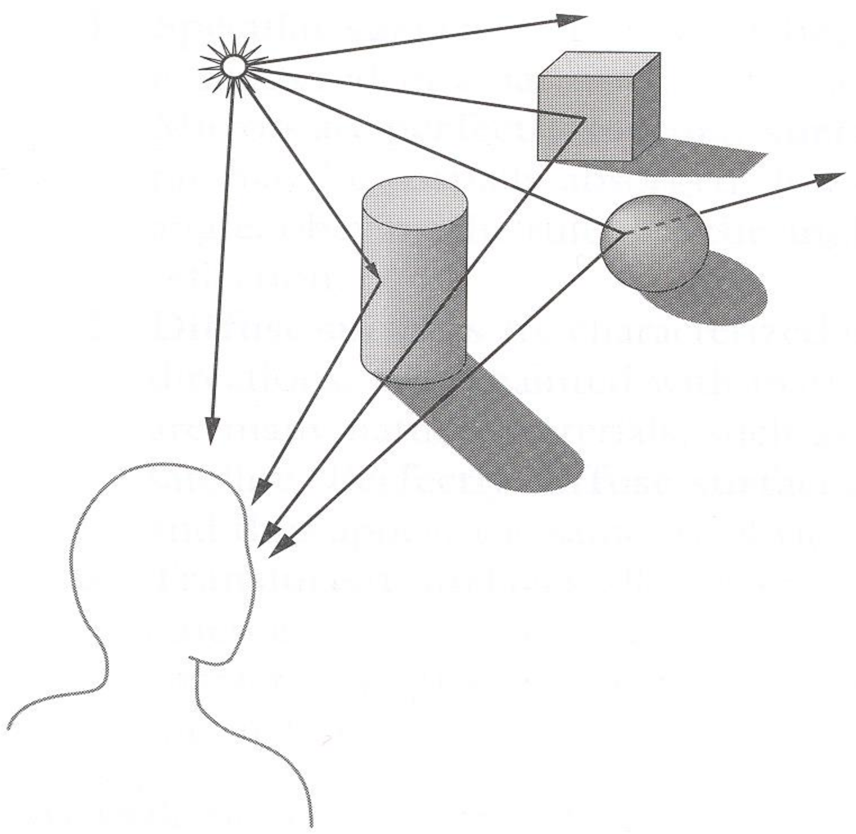
Lighting

Partly based on Georgia Technical University lectures

Illumination

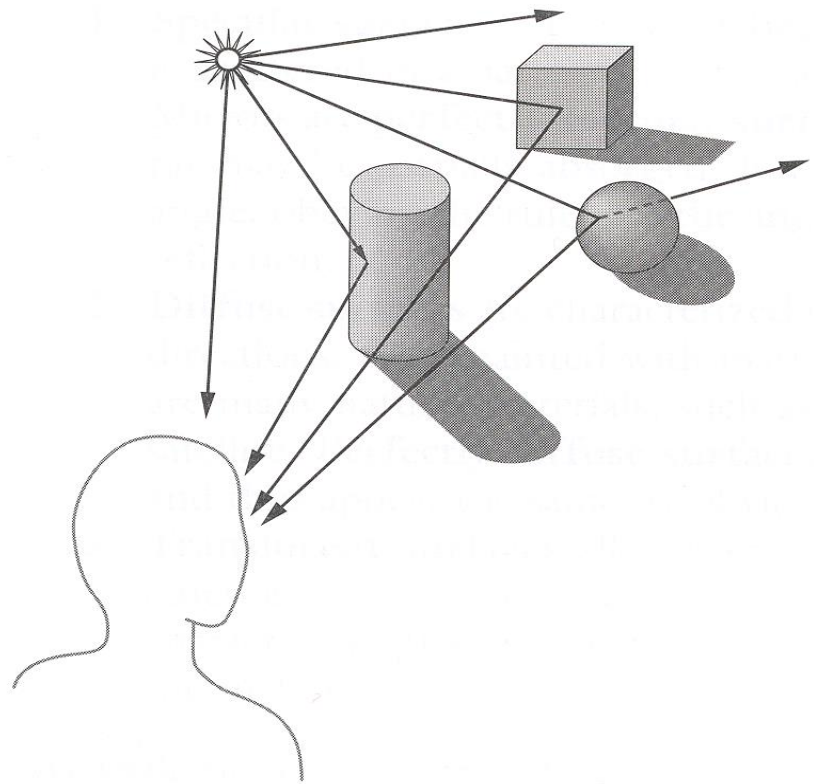
- How do we compute radiance for a sample ray?



Angel Figure 6.2

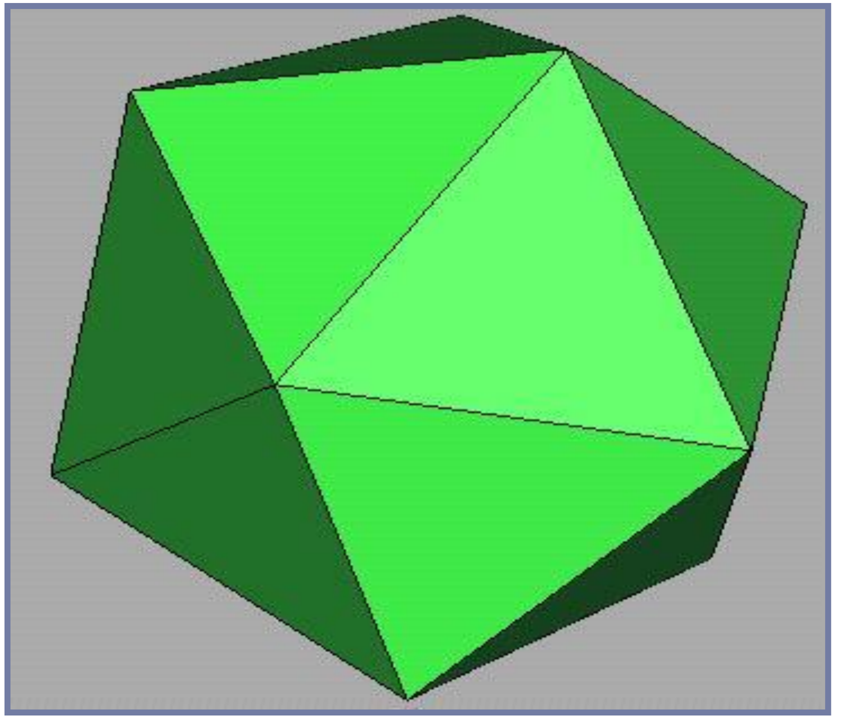
Goal

- Must derive computer models for ...
  - Emission at light sources
  - Scattering at surfaces
  - Reception at the camera
- Desirable features …
  - Concise
  - Efficient to compute
  - “Accurate”



Overview

- Direct (Local) Illumination
  - Emission at light sources
  - Scattering at surfaces
- Global illumination
  - Shadows
  - Refractions
  - Inter-object reflections



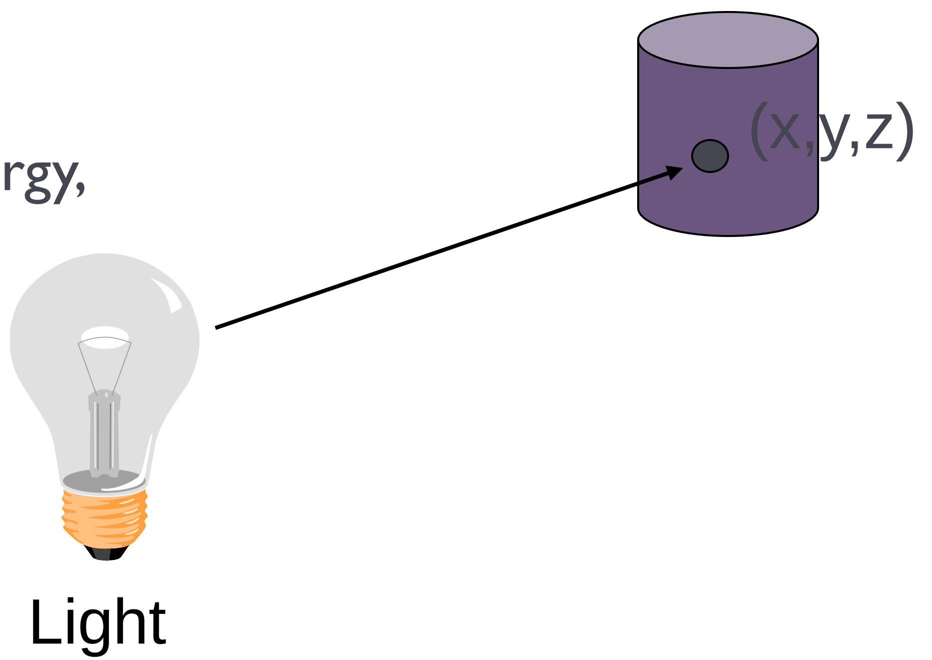
Direct Illumination

Modeling Light Sources

- In general may be quite complicated…

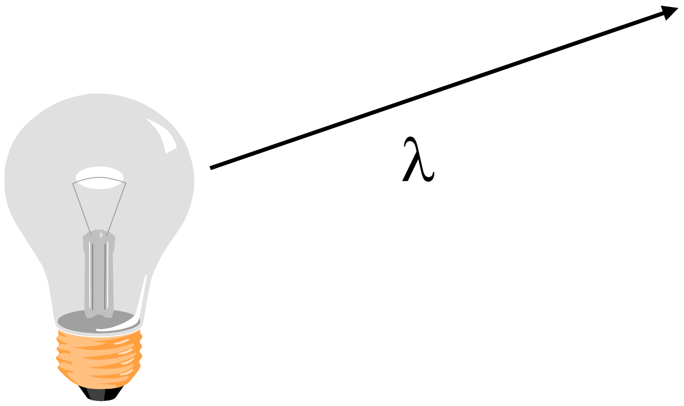
$$I_L = (x, y, z, \theta, \phi, \lambda)$$

- where...
  - I_L describes the intensity of energy,
  - leaving a light source, …
  - arriving at location(x,y,z), ...
  - from direction (θ,ϕ), ...
  - with wavelength λ



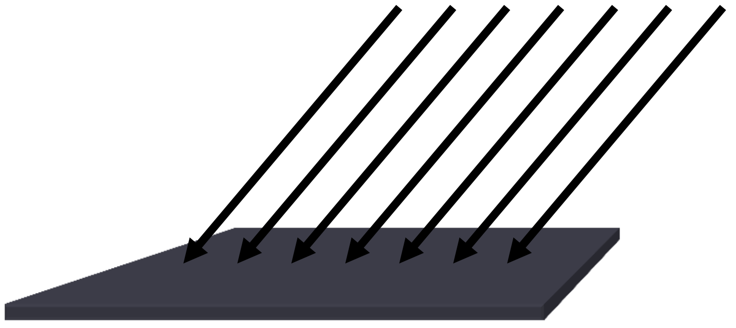
Empirical Models

- Ideally measure irradiant energy for "all" situations
  - Too much storage
  - Difficult in practice



Directional Light Sources

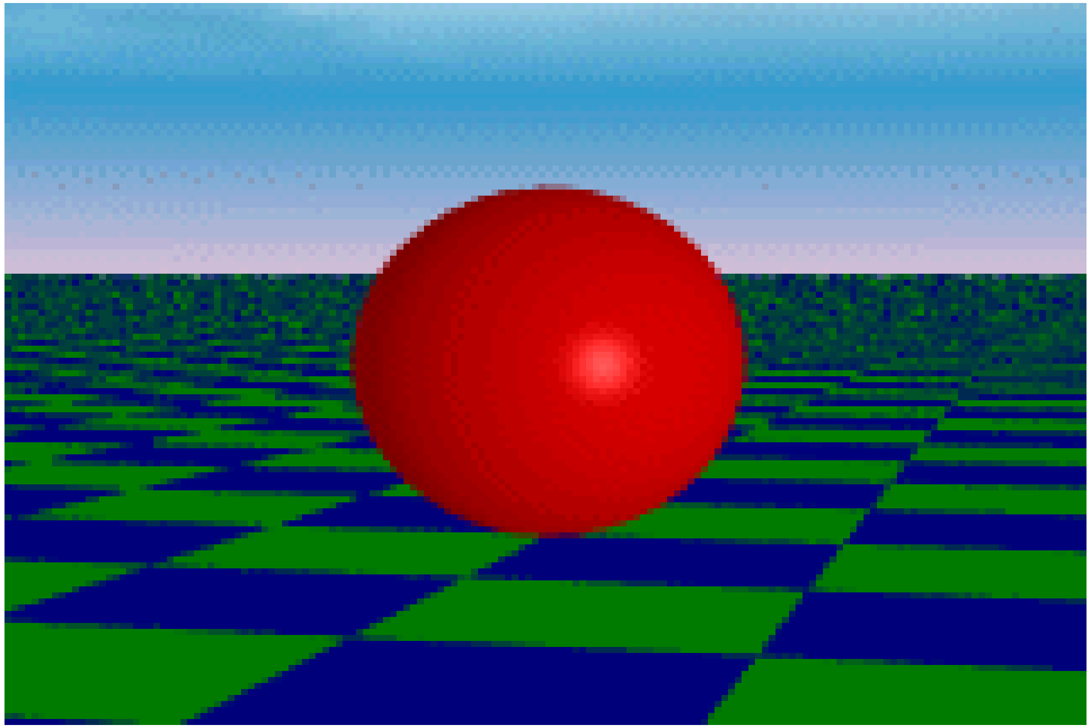
- For a directional light source we make simplifying assumptions
  - Direction is constant for all surfaces in the scene
  - All rays of light from the source are parallel
    - As if the source were infinitely far away from the surfaces in the scene
    - A good approximation to sunlight



- The direction from a surface to the light source is important in lighting the surface

Directional Light Sources

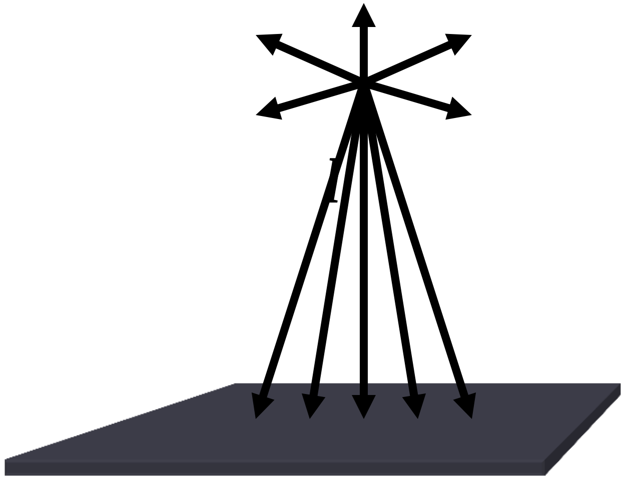
- The same scene lit with a directional and an ambient light source



Point Light Sources

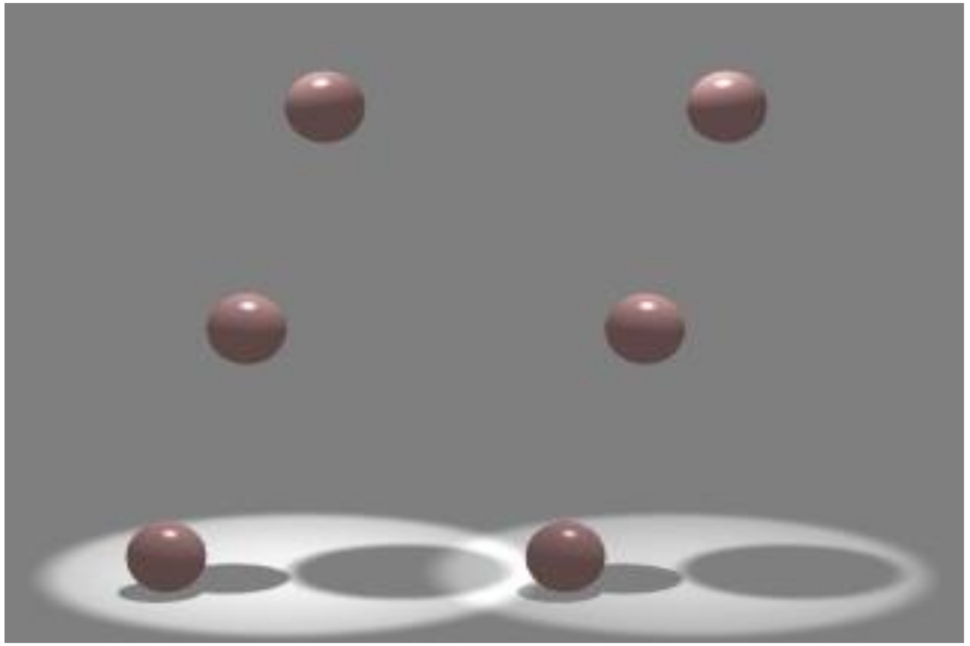
- A point light source emits light equally in all directions from a single point
- The direction to the light from a point on a surface thus differs for different points:
  - So we need to calculate a normalized vector to the light source for every point we light:

$$\bar{d} = \frac{\bar{p} - \bar{l}}{\left\| \bar{p} - \bar{l} \right\|}$$



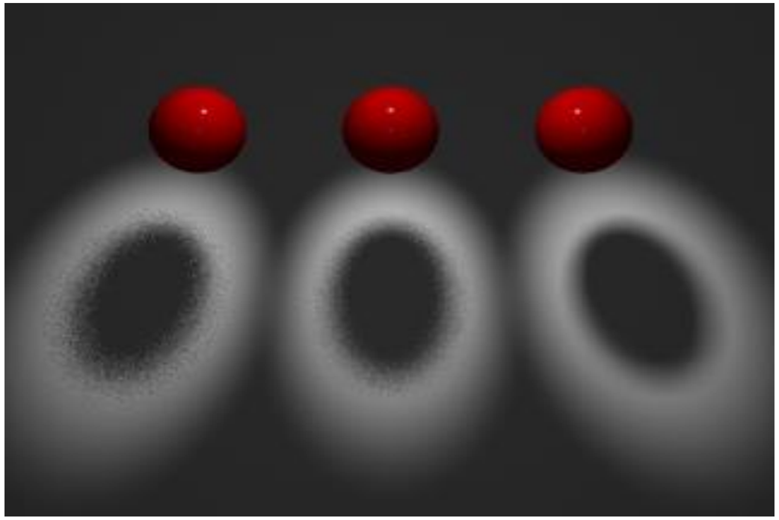
Other Light Sources

- Spotlights are point sources whose intensity falls off directionally.
  - Requires color, point direction, falloff parameters
  - Supported by Three.js



Other Light Sources

- Area light sources define a 2-D emissive surface (usually a disc or polygon)
  - Good example: fluorescent light panels
  - Capable of generating *soft shadows (why?*)



Ambient Light Sources

- Objects not directly lit are typically still visible
  - e.g., the ceiling in this room, undersides of desks
- This is the result of indirect illumination from emitters, bouncing off intermediate surfaces
- Too expensive to calculate (in real time), so we use a hack called an ambient light source
  - No spatial or directional characteristics; illuminates all surfaces equally
  - Amount reflected depends on surface properties

Ambient Light Sources

- For each sampled wavelength (R, G, B), the ambient light reflected from a surface depends on
  - The surface properties, k_A
  - The intensity, $I_{A,}$ of the ambient light source (constant for all points on all surfaces)

$$I_R = k_A \cdot I_A$$

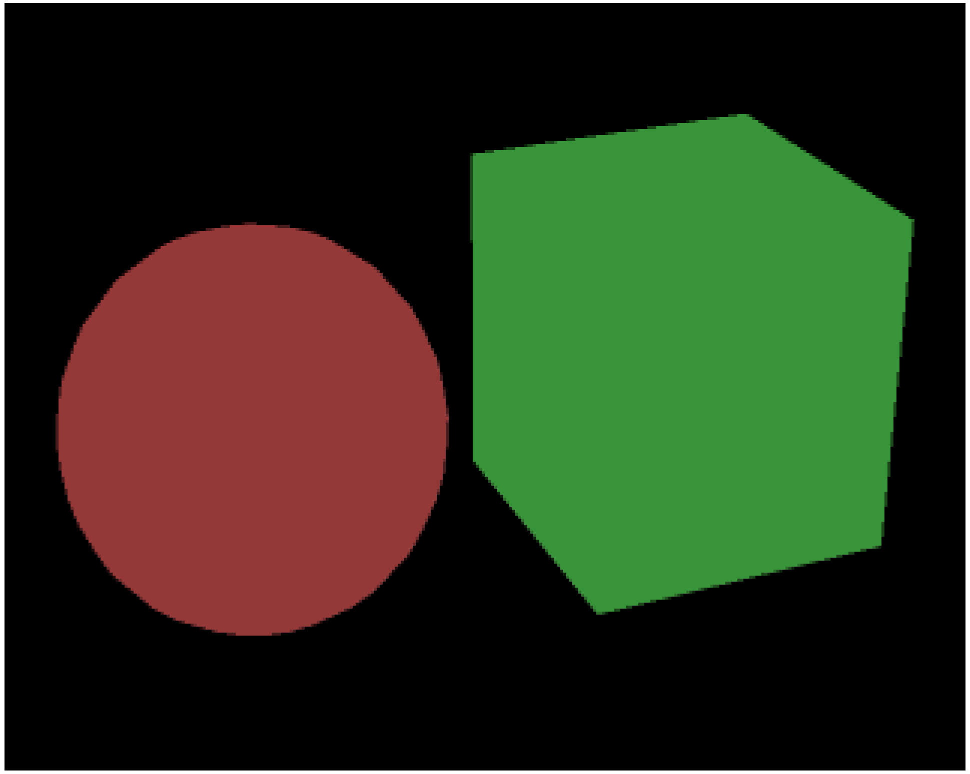
$$I_R^R = k_A^R \cdot I_A^R$$

$$I_R^G = k_A^G \cdot I_A^G$$

$$I_R^B = k_A^B \cdot I_A^B$$

Ambient Light Sources

- A scene lit only with an ambient light source:



Light Position
Not Important

Viewer Position
Not Important

Surface Angle
Not Important

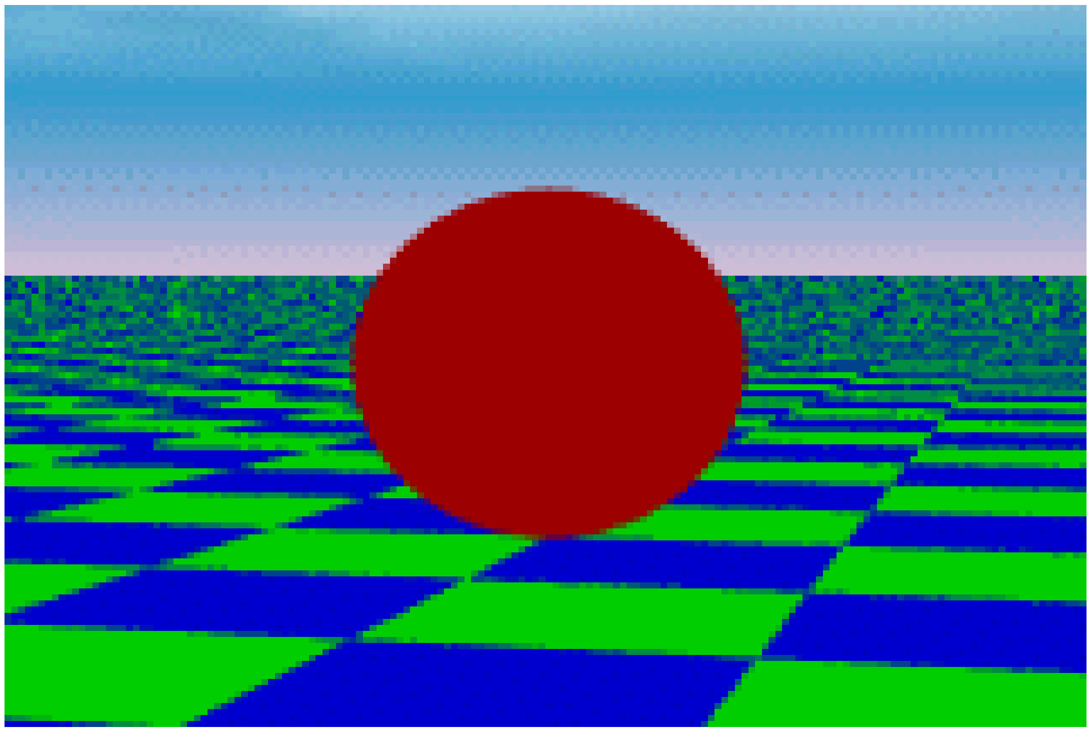
Ambient Term

- Represents reflection of all indirect illumination

This is a total hack (avoids complexity of global illumination)!

Ambient Light Sources

- A scene lit only with an ambient light source:



Light Position
Not Important

Viewer Position
Not Important

Surface Angle
Not Important

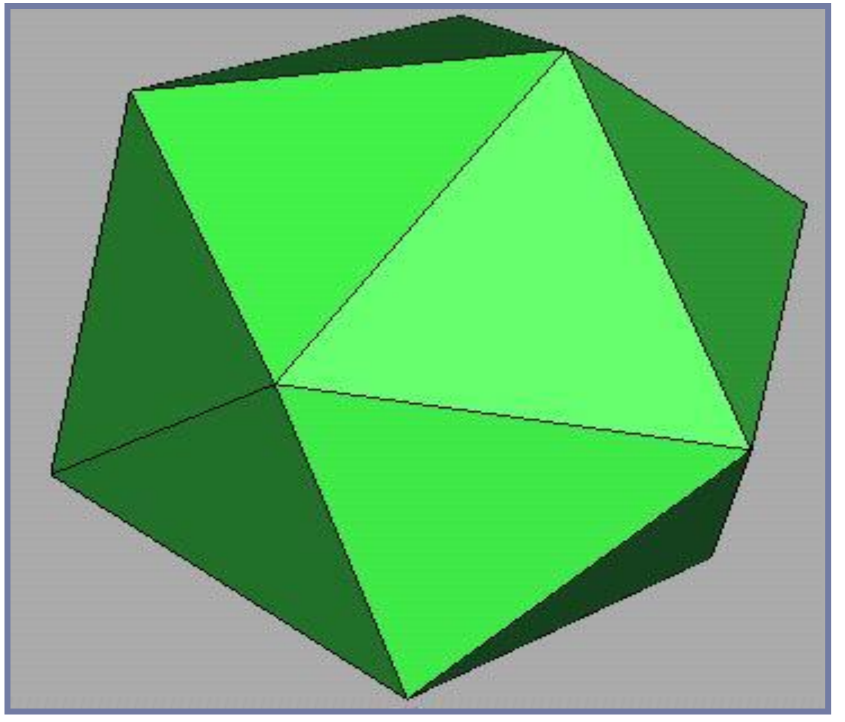
Ambient Term

- Represents reflection of all indirect illumination

This is a total hack (avoids complexity of global illumination)!

Overview

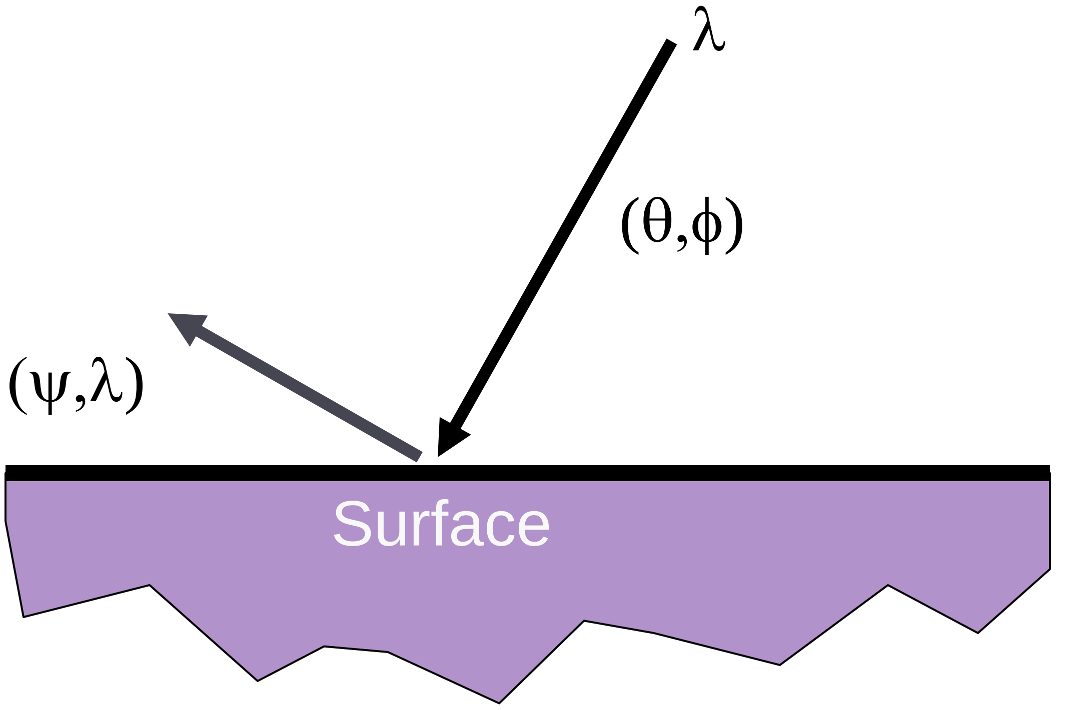
- Direct (Local) Illumination
  - Emission at light sources
  - Scattering at surfaces
- Global illumination
  - Shadows
  - Refractions
  - Inter-object reflections



Direct Illumination

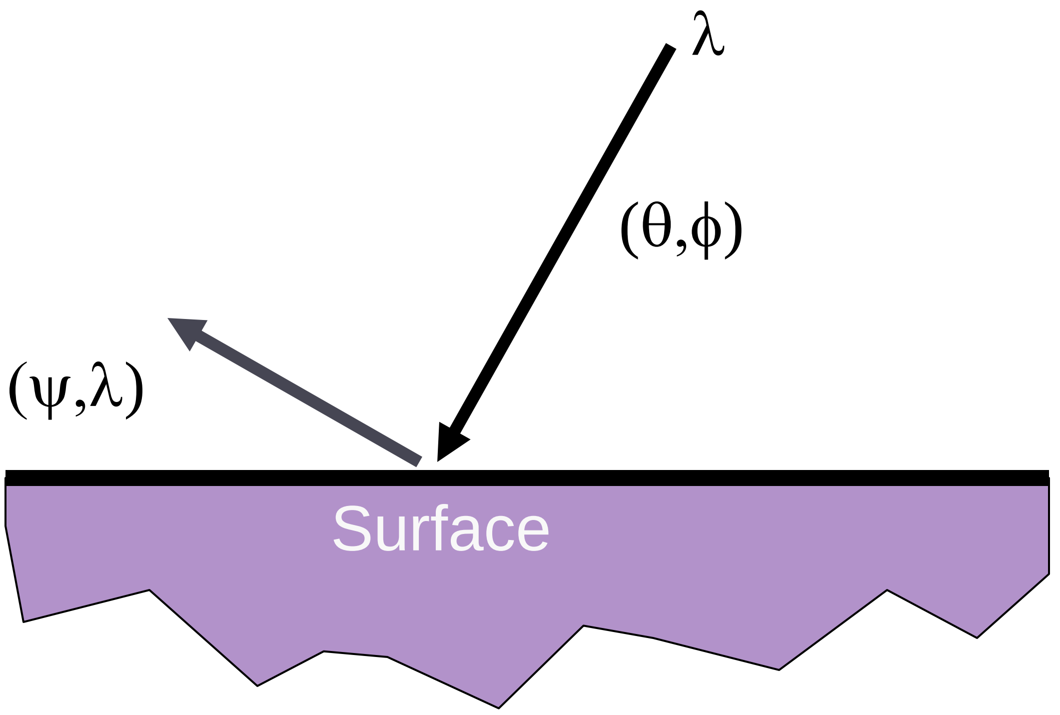
Modeling Surface Reflectance

- $R_s(\theta,\phi,\gamma,\psi,\lambda)$ ...
  - describes the amount of incident energy,
  - arriving from direction (θ,ϕ), ...
  - leaving in direction (γ,ψ), …
  - with wavelength λ



Empirical Models

- Ideally measure radiant energy for "all" combinations of incident angles
  - Too much storage
  - Difficult in practice

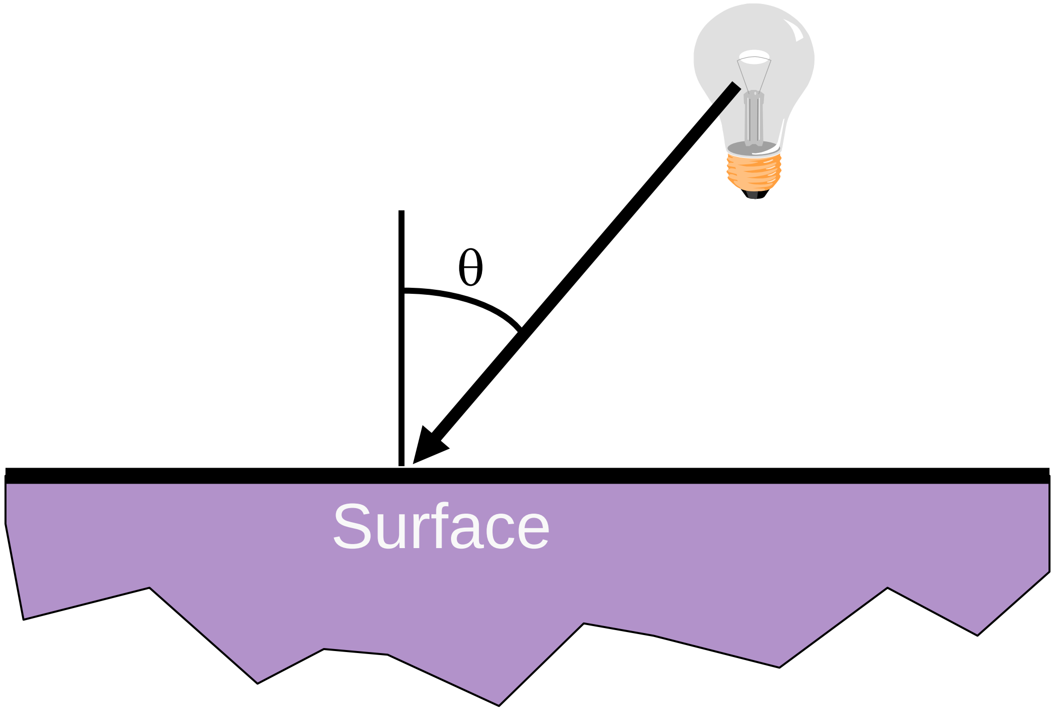


The Physics of Reflection

- Ideal diffuse reflection
  - An *ideal diffuse reflector*, at the microscopic level, is a very rough surface (real-world example: chalk)
  - Because of these microscopic variations, an incoming ray of light is equally likely to be reflected in any direction over the hemisphere:
  - 

Diffuse Reflection

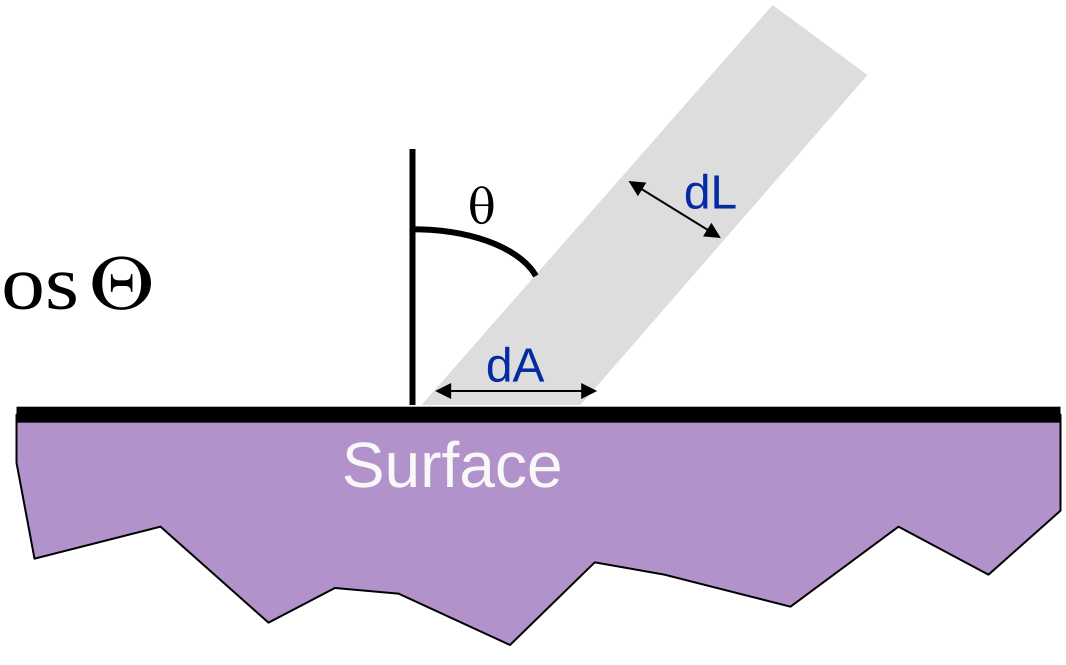
- How much light is reflected?
  - Depends on angle of incident light



Diffuse Reflection

- How much light is reflected?
  - Depends on angle of incident light

$$dL = dA \cos \Theta$$



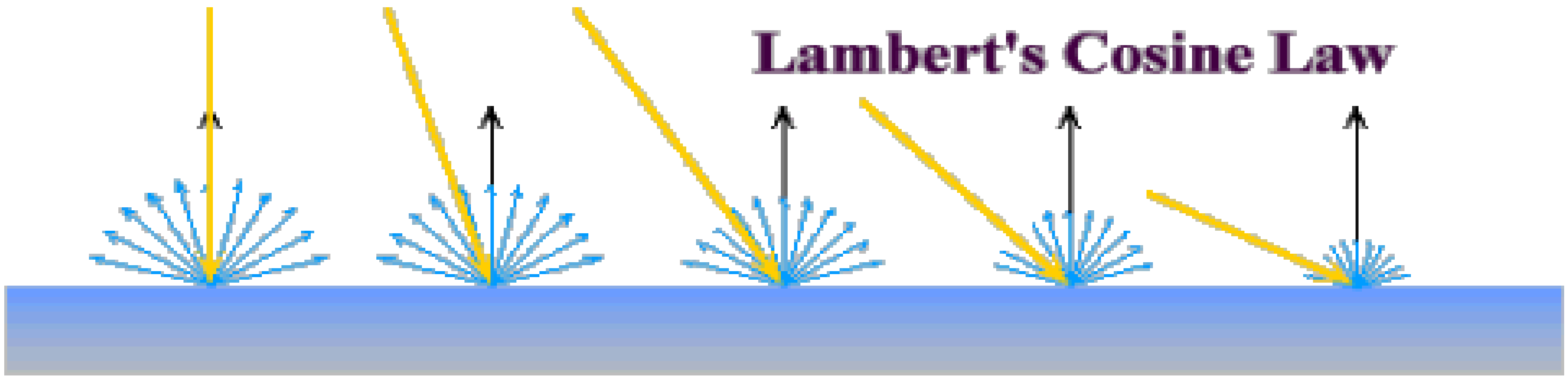
Lambert's Cosine Law

- Ideal diffuse surfaces reflect according to Lambert's cosine law:

  The energy reflected by a small portion of a surface from a light source in a given direction is proportional to the cosine of the angle between that direction and the surface normal

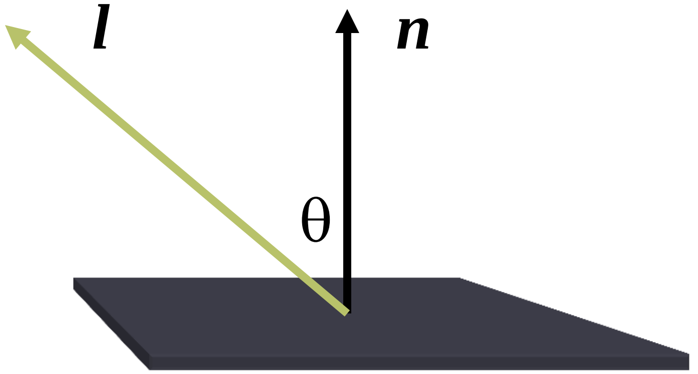
- These are often called Lambertian surfaces
- Note that the reflected intensity is independent of the viewing direction, but does depend on the surface orientation with regard to the light source

Lambert's Law`



Computing Diffuse Reflection

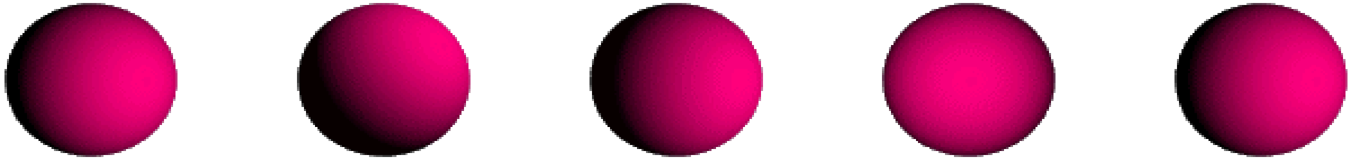
- The angle between the surface normal and the incoming light is the angle of incidence:



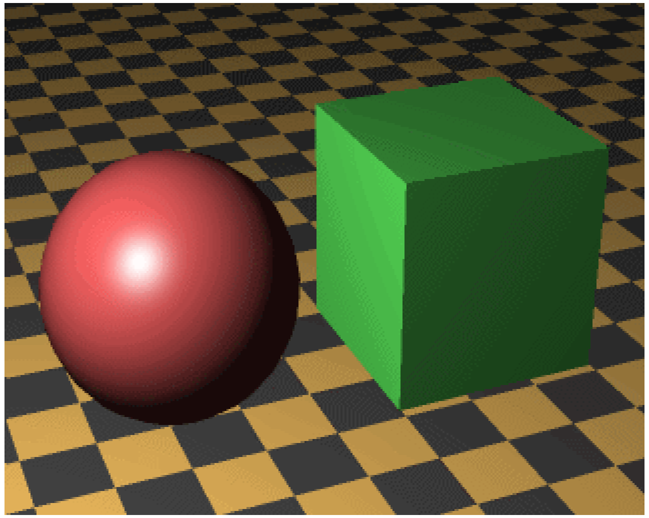
  - $I_{diffuse} = k_d \, I_{light} \cos \theta$
- In practice we use vector arithmetic:
  - $I_{diffuse} = k_d \, I_{light} \, (n \bullet l)$

Diffuse Lighting Examples

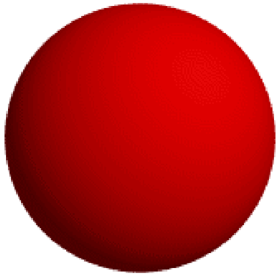
- We need only consider angles from 0° to 90° (Why?)
- A Lambertian sphere seen at several different lighting angles:



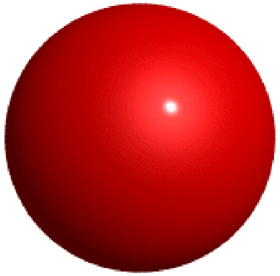
Specular Reflection



- Shiny surfaces exhibit specular reflection
  - Polished metal
  - Glossy car finish



Diffuse Lighting



Plus Specular Highlight

- A light shining on a specular surface causes a bright spot known as a specular highlight

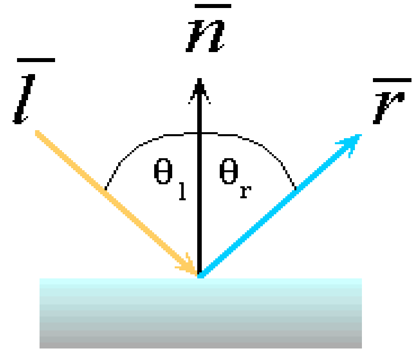
- Where these highlights appear is a function of the viewer's position, so specular reflectance is view dependent

The Physics of Reflection

- At the microscopic level a specular reflecting surface is very smooth
- Thus rays of light are likely to bounce off the microgeometry in a mirror-like fashion
- The smoother the surface, the closer it becomes to a perfect mirror

The Optics of Reflection

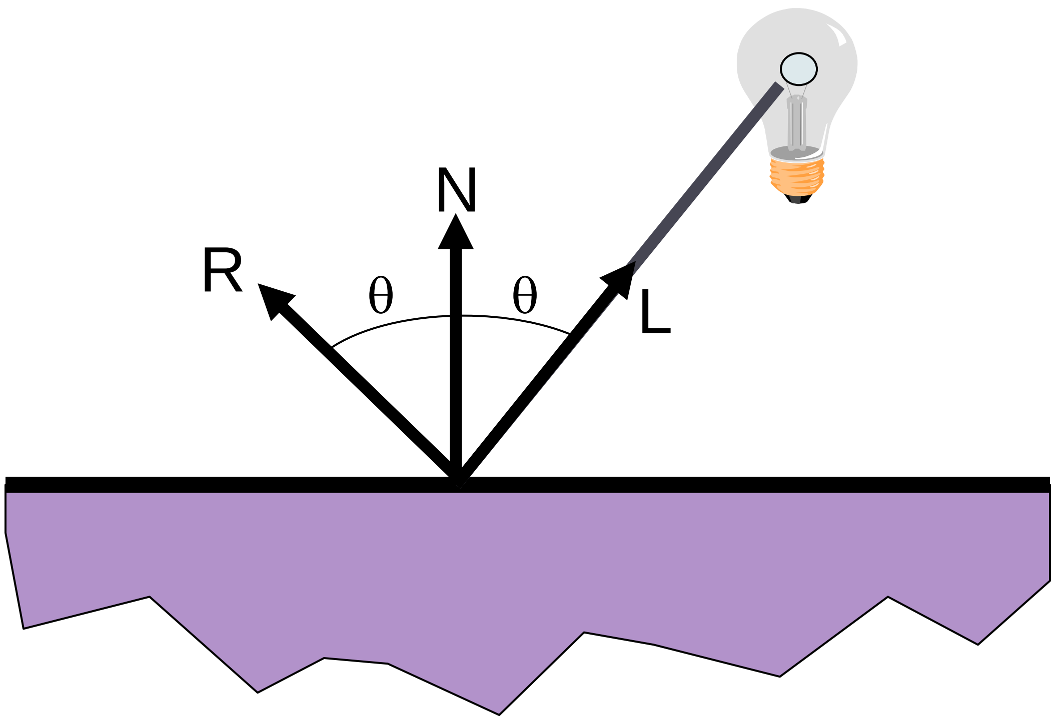
- Reflection follows Snell's Laws:
  - The incoming ray and reflected ray lie in a plane with the surface normal
  - The angle that the reflected ray forms with the surface normal equals the angle formed by the incoming ray and the surface normal:



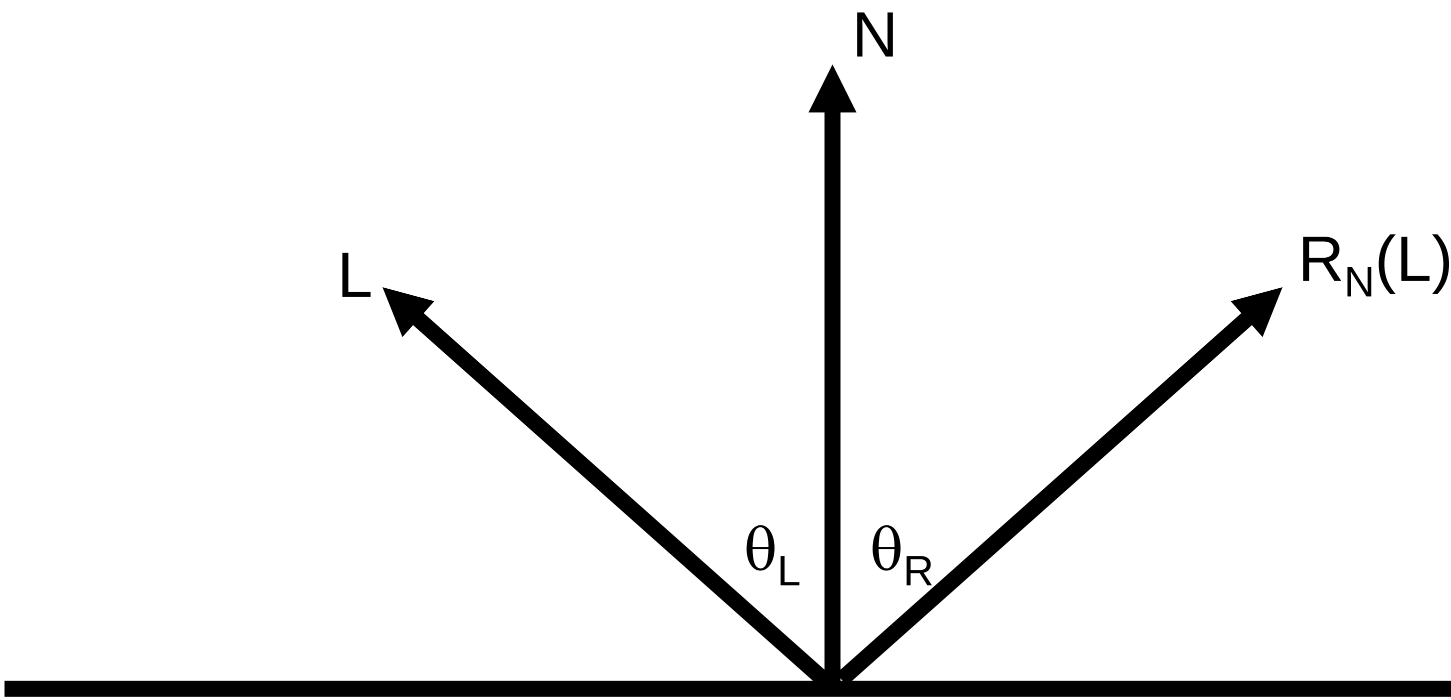
$$\theta_{(l)ight} = \theta_{(r)eflection}$$

Specular Reflection

- Reflection is strongest near mirror angle
  - Examples: mirrors, metals

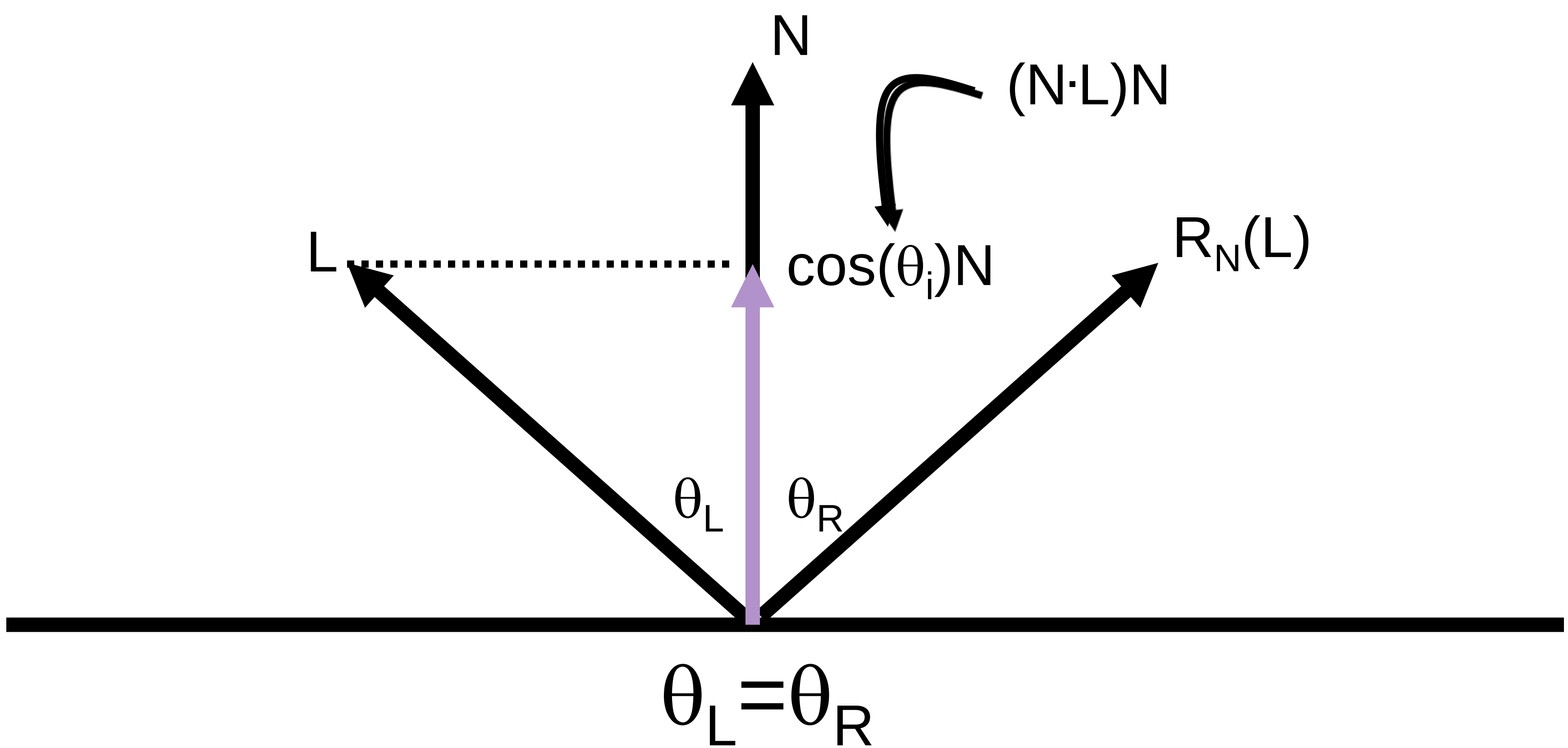


Geometry of Reflection

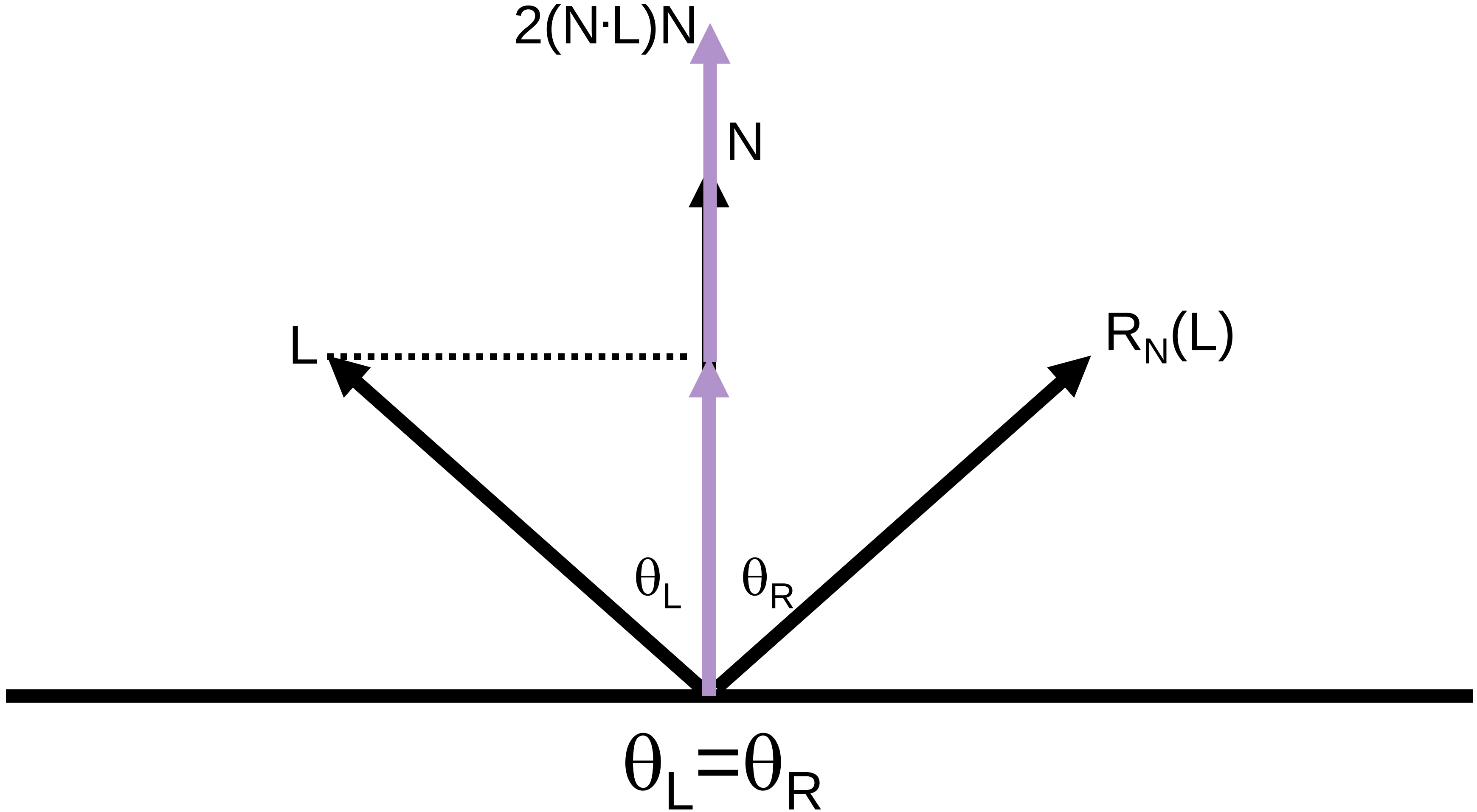


$$\theta_L = \theta_R$$

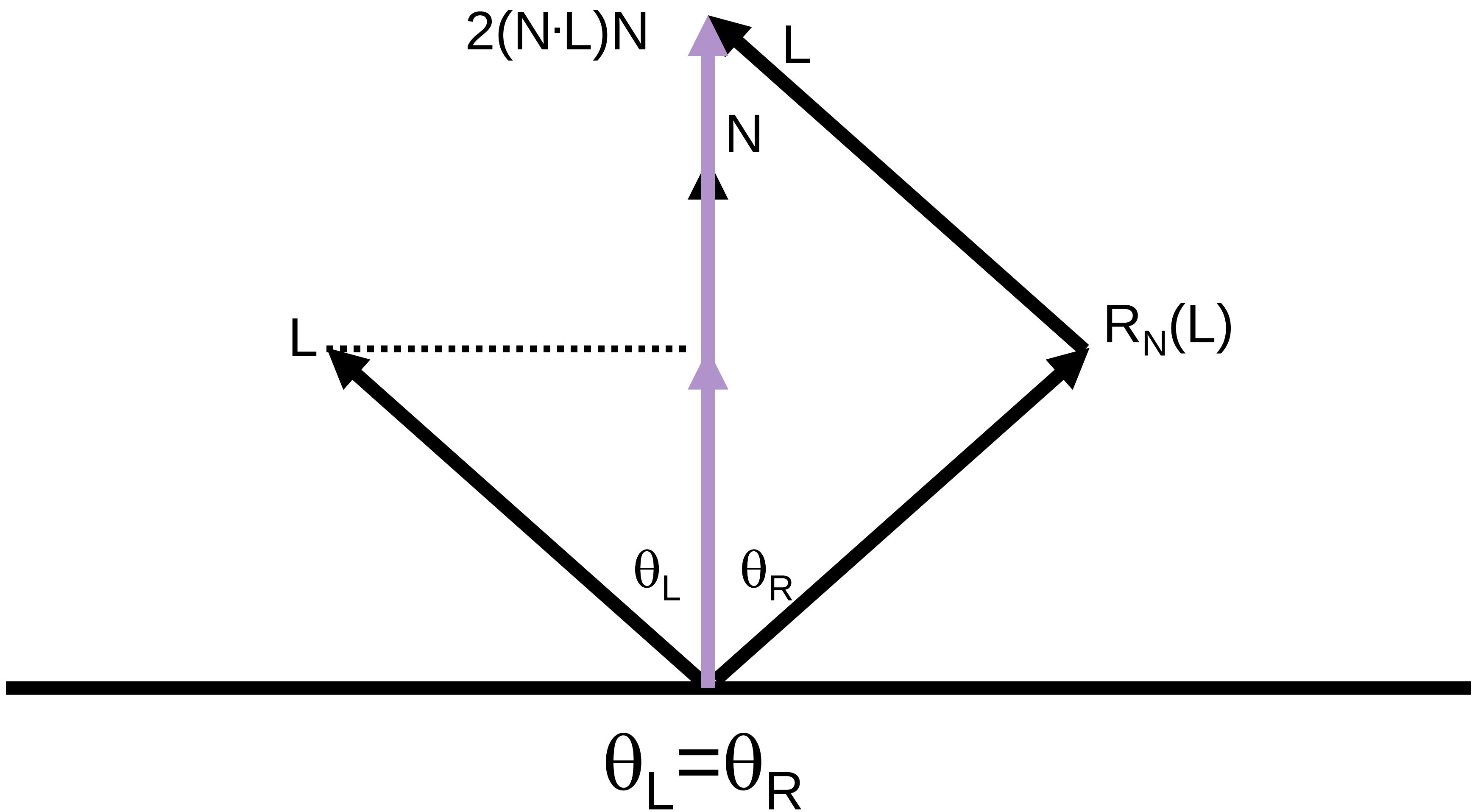
Geometry of Reflection



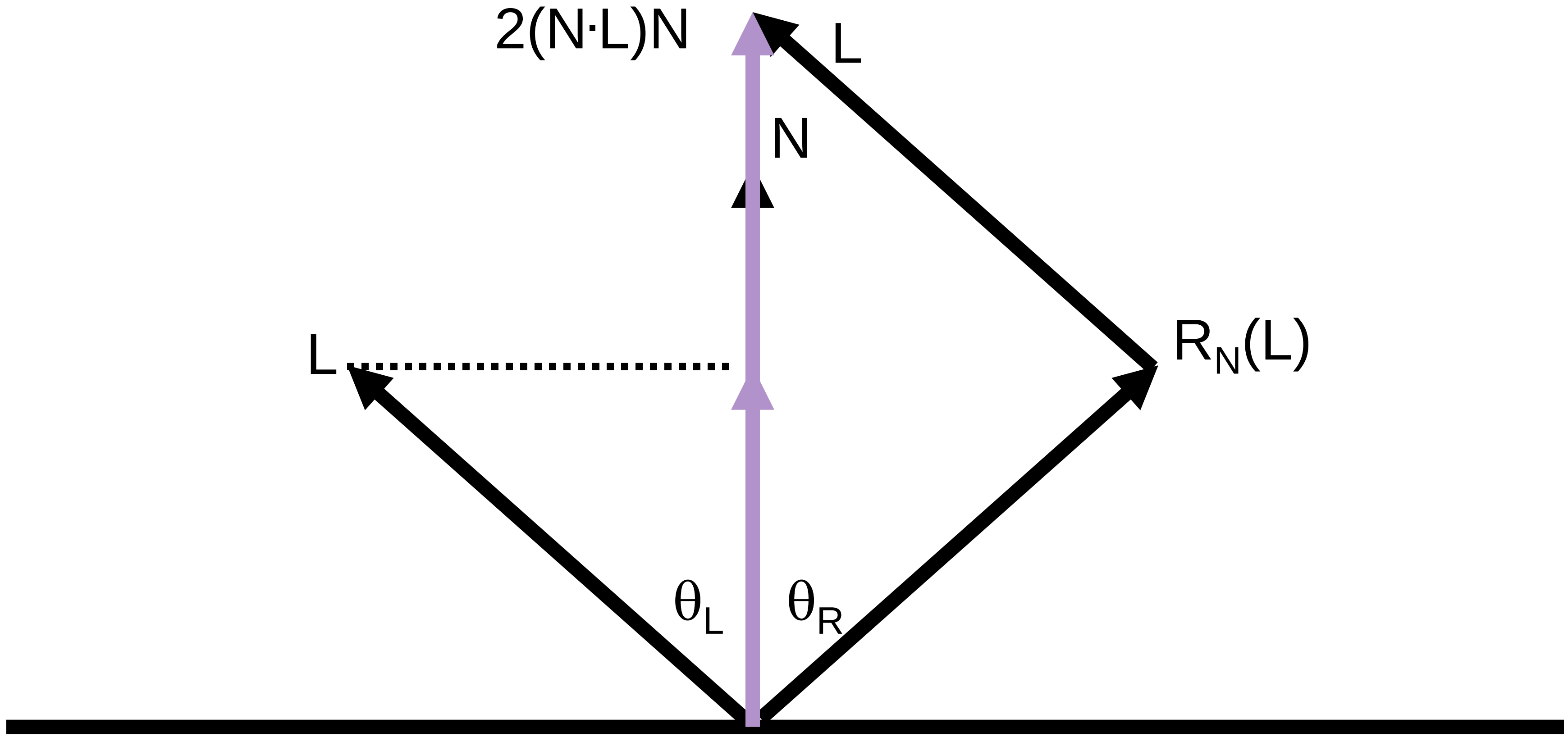
Geometry of Reflection



Geometry of Reflection



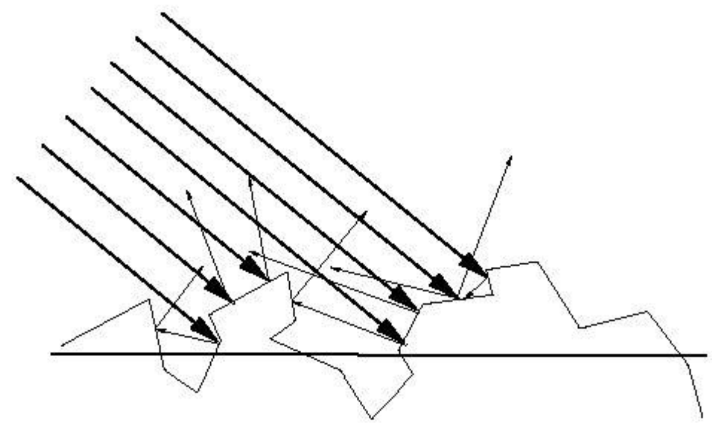
Geometry of Reflection



$\theta_L = \theta_R$

Non-Ideal Specular Reflectance

- Snell's law applies to perfect mirror-like surfaces, but aside from mirrors (and chrome) few surfaces exhibit perfect specularity
- How can we capture the "softer" reflections of surface that are glossy rather than mirror-like?



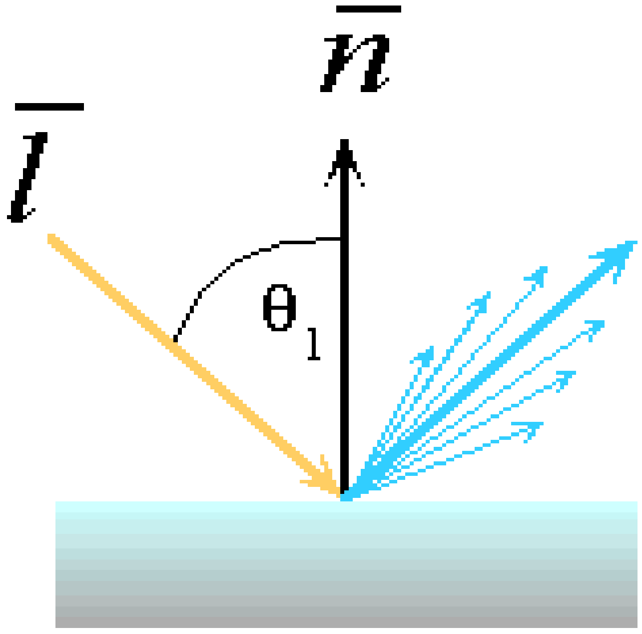
- One option: model the microgeometry of tl bounce rays off of it
- Or…

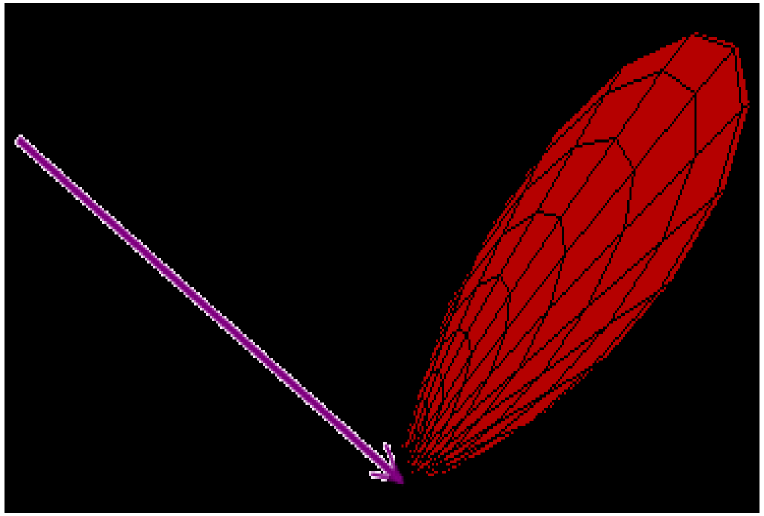
Non-Ideal Specular Reflectance: An Empirical Approximation

- Hypothesis: most light reflects according to Snell's Law
  - But because of microscopic surface variations, some light may be reflected in a direction slightly off the ideal reflected ray
- Hypothesis: as we move from the ideal reflected ray, some light is still reflected

Non-Ideal Specular Reflectance: An Empirical Approximation

- An illustration of this angular falloff:





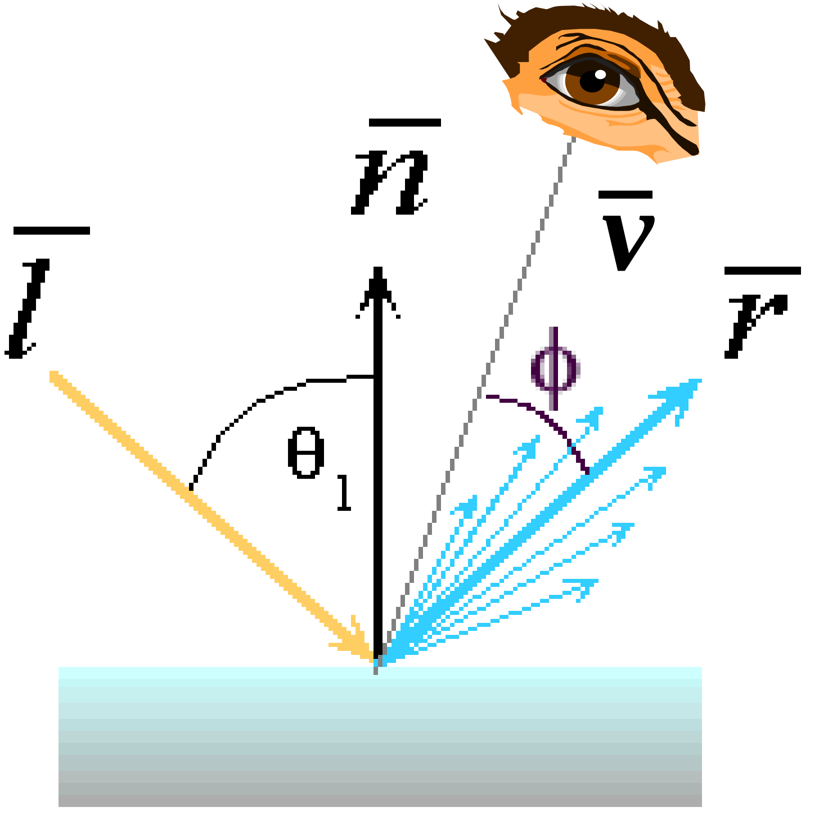
Phong Lighting

- The most common lighting model in computer graphics was suggested by Phong:

$$I_{specular} = k_s I_{light} (\cos\phi)^{n_{shiny}}$$

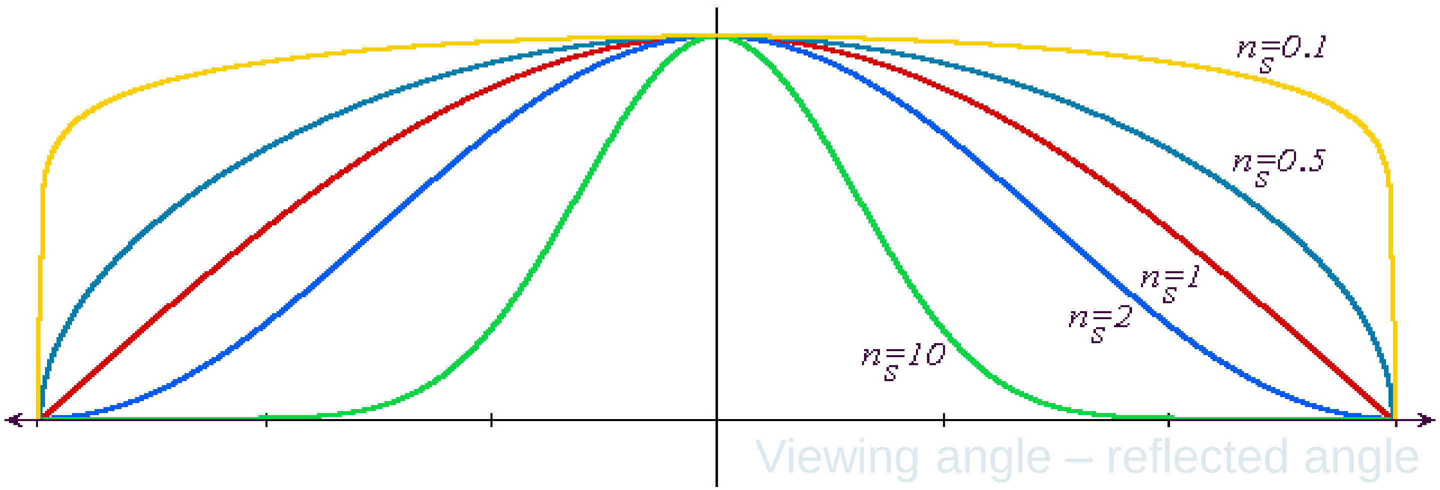
The $\mathbf{n_{shiny}}$ term is a purely empirical constant that varies the rate of falloff

Though this model has no physical basis, it works (sort of) in practice



Phong Lighting: The n_{shiny} Term

- This diagram shows how the Phong reflectance term drops off with divergence of the viewing angle from the ideal reflected ray:



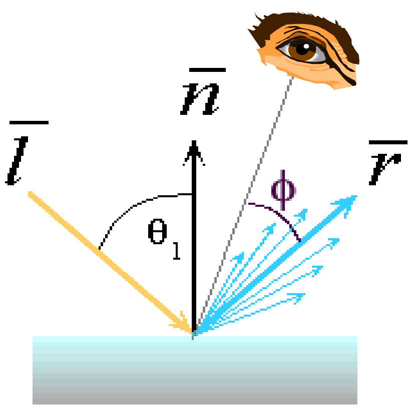
- What

Calculating Phong Lighting

- The cos term of Phong lighting can be computed using vector arithmetic:

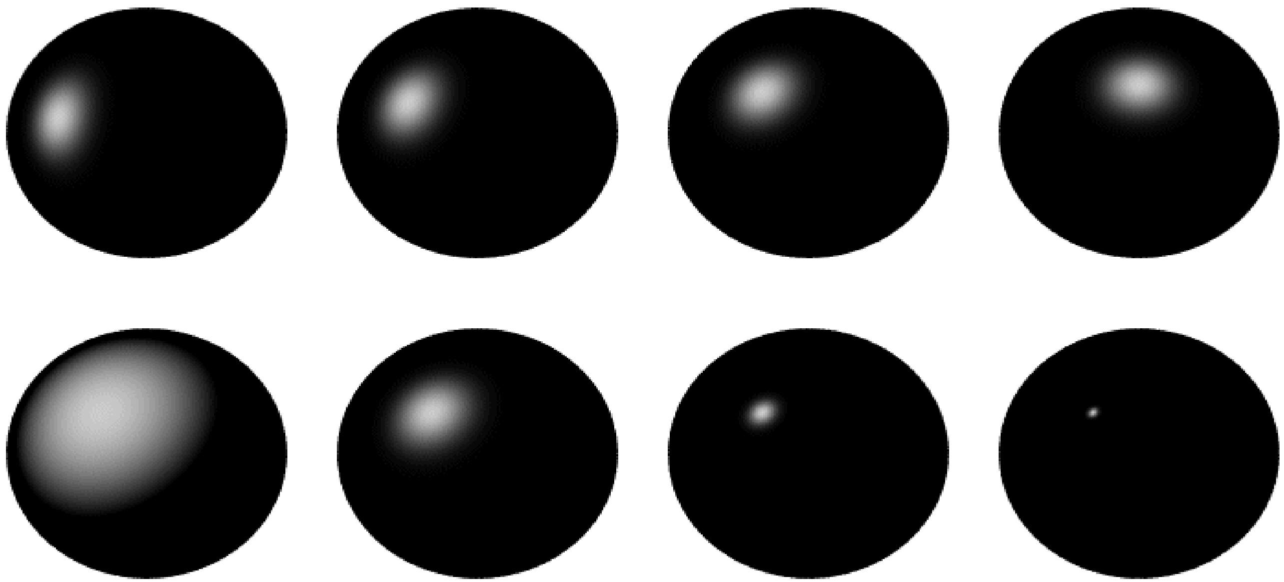
$$I_{specular} = k_s I_{light} \left(\bar{v} \cdot \bar{r} \right)^{n_{shiny}}$$

  - v is the unit vector towards the viewer
  - r is the ideal reflectance direction



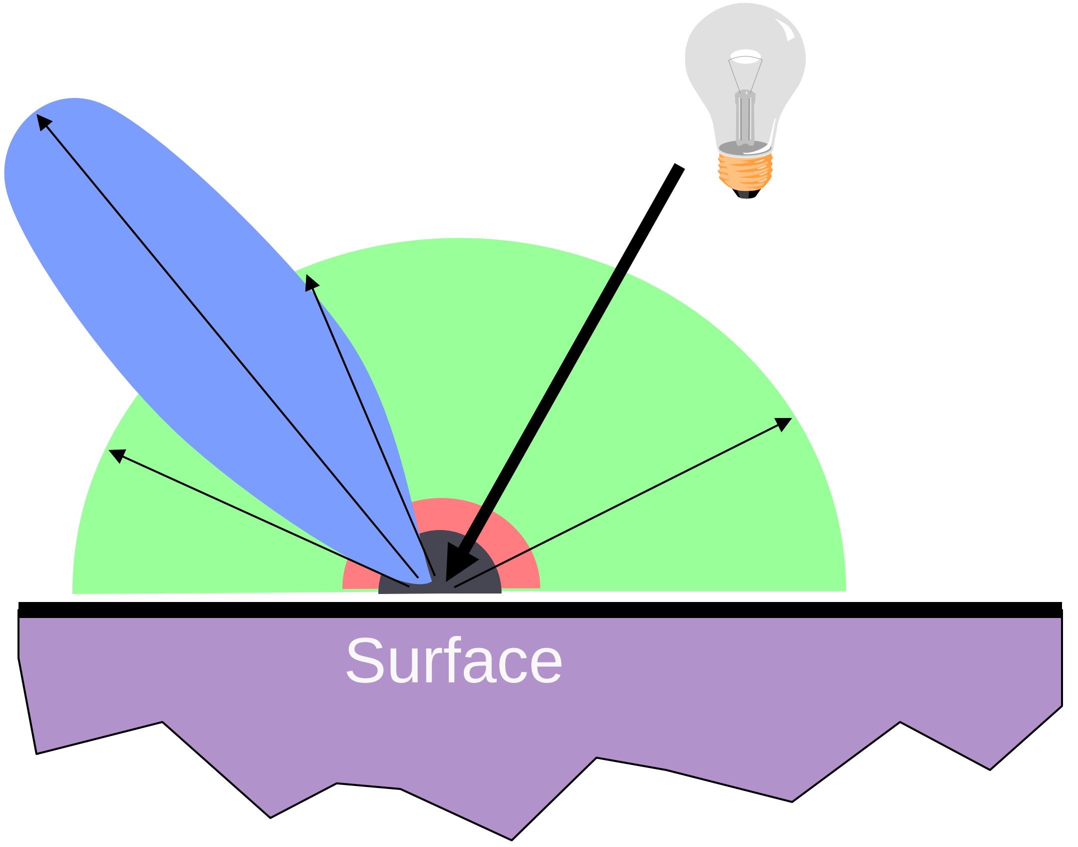
Phong Examples

- These spheres illustrate the Phong model as l and n_{shiny} are varied:



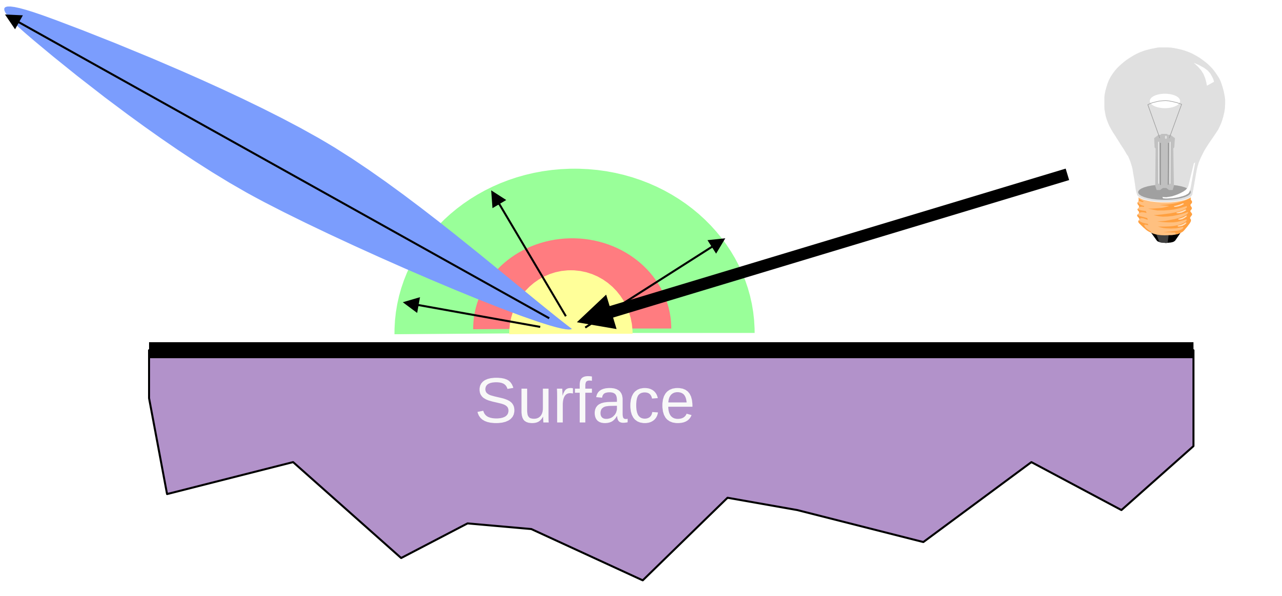
Combining Everything

- Simple analytic model:
  - diffuse reflection +
  - specular reflection +
  - emission +
  - “ambient”



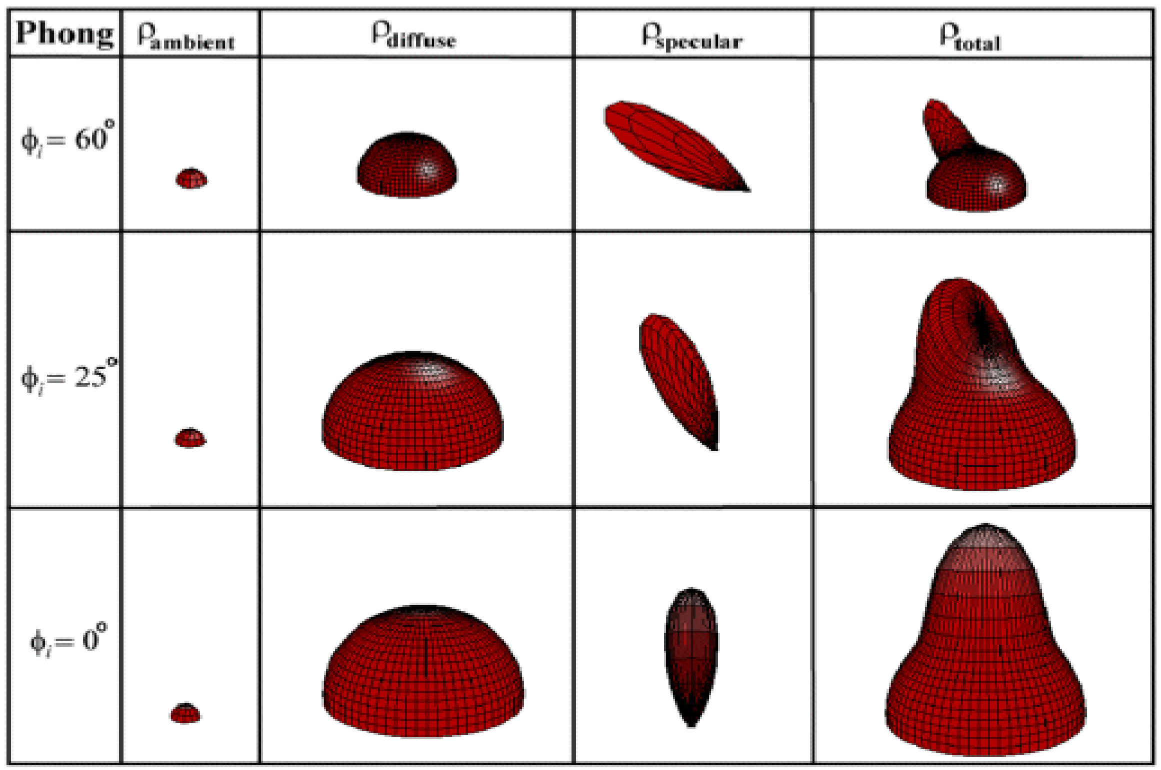
Combining Everything

- Simple analytic model:
  - diffuse reflection +
  - specular reflection +
  - emission +
  - "ambient"



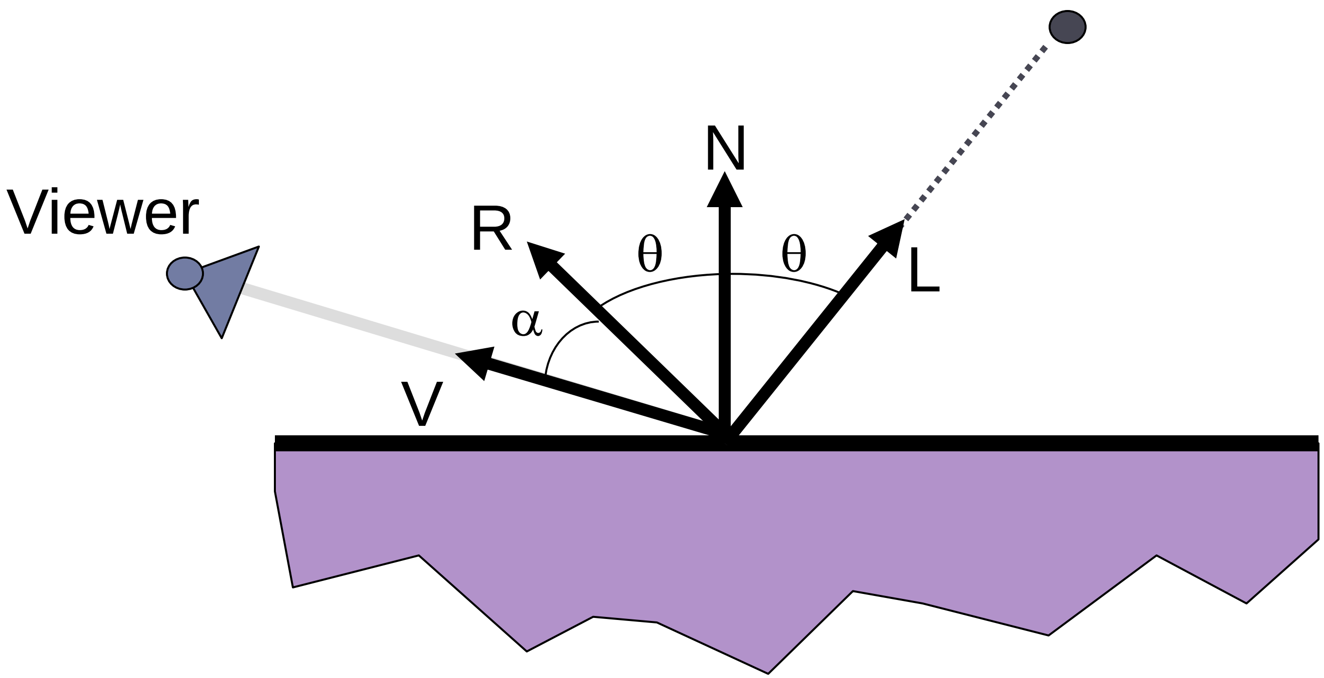
OpenGL/WebGL Reflectance Model

- Sum diffuse, specular, emission, and ambient



The Final Combined Equation

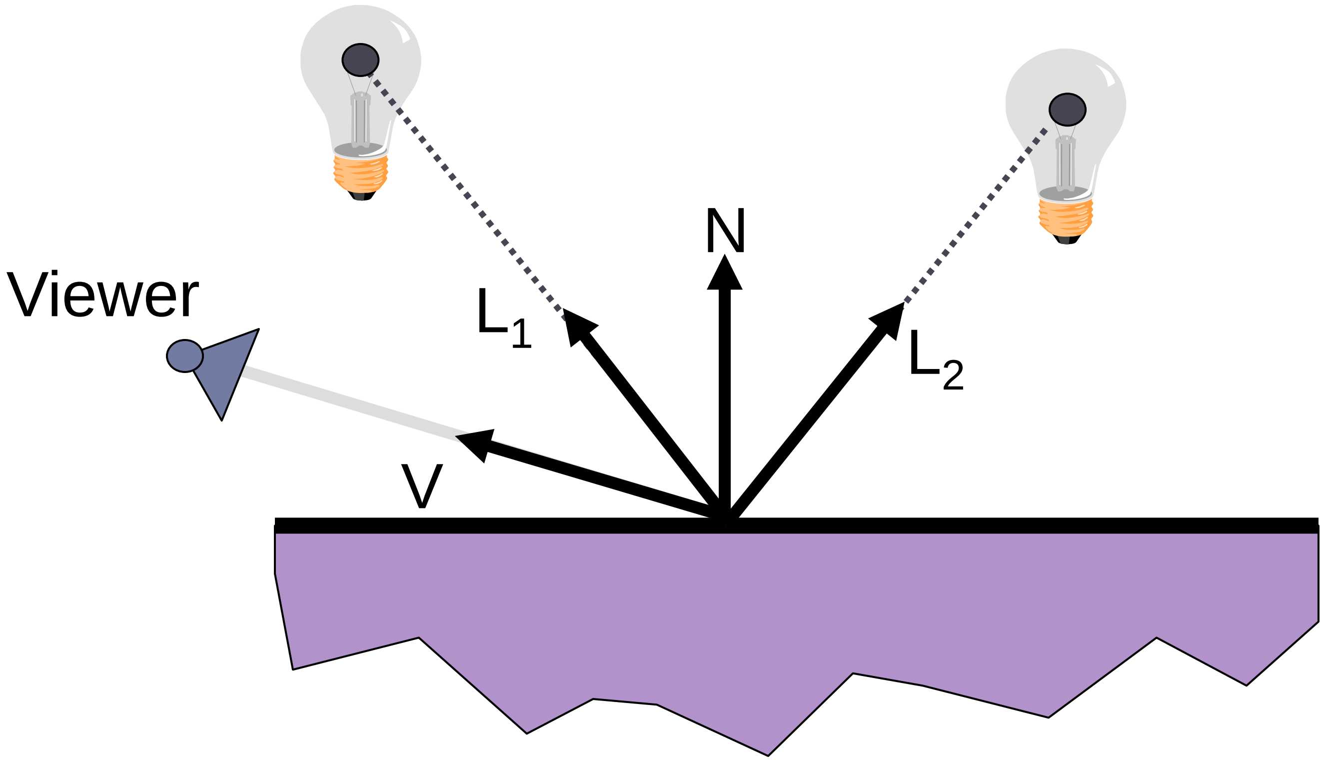
- Single light source:



$$I = I_E + K_A I_{AL} + K_D (N \bullet L) I_L + K_S (V \bullet R)^n I_L$$

Final Combined Equation

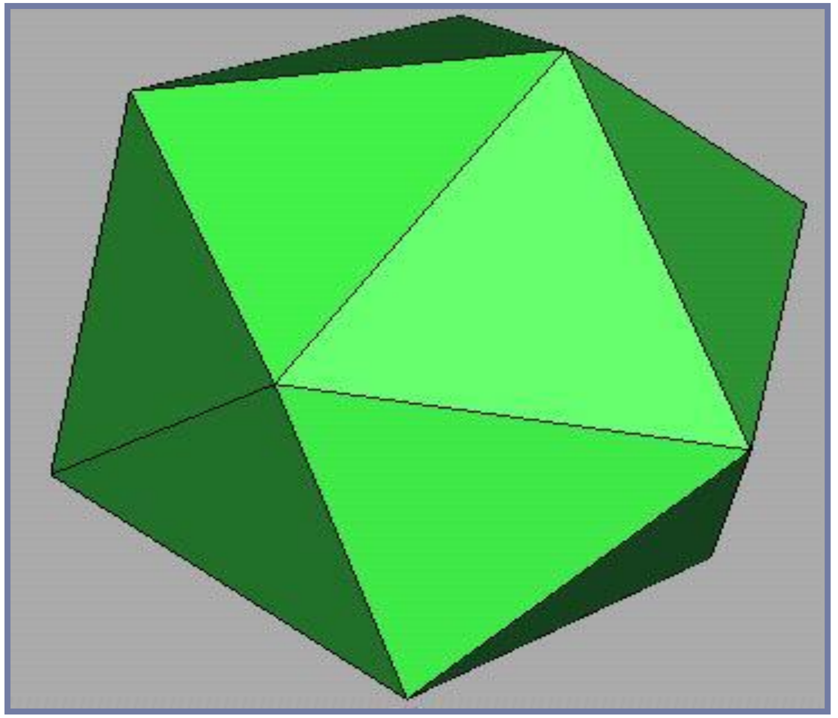
- Multiple light sources:



$$I = I_E + K_A I_{AL} + \sum_i \left(K_D (N \bullet L_i) I_i + K_S (V \bullet R_i)^n I_i\right)$$

Overview

- Direct (Local) Illumination
  - Emission at light sources
  - Scattering at surfaces
- Global illumination
  - Shadows
  - Refractions
  - Inter-object reflections



Direct Illumination

Global Illumination

- We've glossed over how light really works
- And we will continue to do so…
- One step better

- Global Illumination
  - The notion that a point is illuminated by more than light from local lights; it is illuminated by all the emitters and reflectors in the global scene

The ‘Rendering Equation’

- Jim Kajiya developed this in 1986 – we are not going to use it

$$I(x,x') = g(x,x')\left[\varepsilon(x,x') + \int_S \rho(x,x',x'')I(x',x'')dx''\right]$$

- $I(x, x')$ = total intensity from point x’ to x
- $g(x, x')$ = 0 when x/x’ are occluded
  = 1/d^2 otherwise (d = distance between x and x’)
- $\varepsilon(x, x')$ = intensity emitted by x’ to x
- $\rho(x, x', x'')$ = intensity of light reflected from x” to x through x’
- S = all points on all surfaces

The 'Rendering Equation'

- The light that hits x from x' is the direct illumination from x' and all the light reflected by x' from all x''
- To implement:
  - Must handle recursion effectively
  - Must support diffuse and specular light
  - Must model object shadowing