# CSE 167:
# Introduction to Computer Graphics
# Lecture #6: Shading

Jürgen P. Schulze, Ph.D.
University of California, San Diego
Fall Quarter 2018

# Announcements

- Project 2 due this Friday at 2pm
  - Grading in CSE basement labs B260 and B270
  - This time using Autograder (no whiteboard)
  - Upload code to TritonEd by 2pm

# Internship Opportunity

- Jurgen –

  I actually may need your help sooner then later.  I'm actually looking to get one or two interns into this new company to help with the application.  It's a web based SAAS app and I'm looking for a full stack developer that knows react and node.  They use meteor (which is just a package of custom java scripts), but finding anyone with meteor experience is rare and not mission critical if they know how to trace the information.

  The company is called "SimpleForms" (www.simpleforms.com).  If you know any students that may be interested please let me know.

  My email is danlipsky0@gmail.com

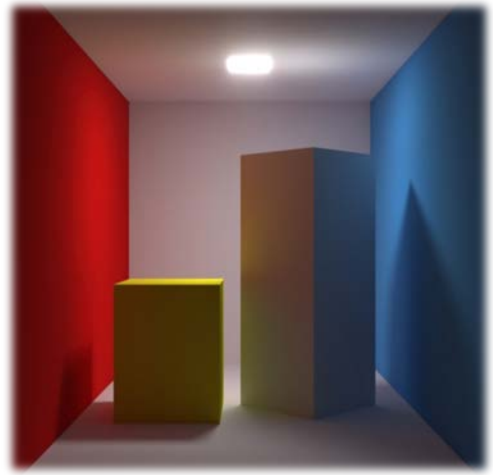
  …Dan

# Shading

# Shading

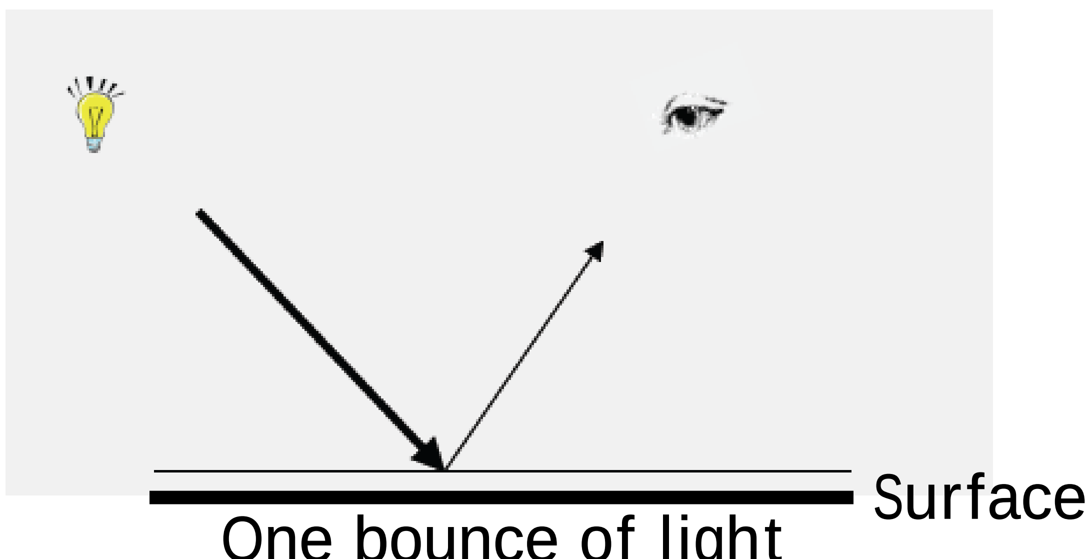
- Compute interaction of light with surfaces
- Requires simulation of physics
- "Global illumination"
  - Multiple bounces of light
  - Computationally expensive, minutes per image
  - Used in movies, architectural design, etc.

# Global Illumination



# Interactive Applications

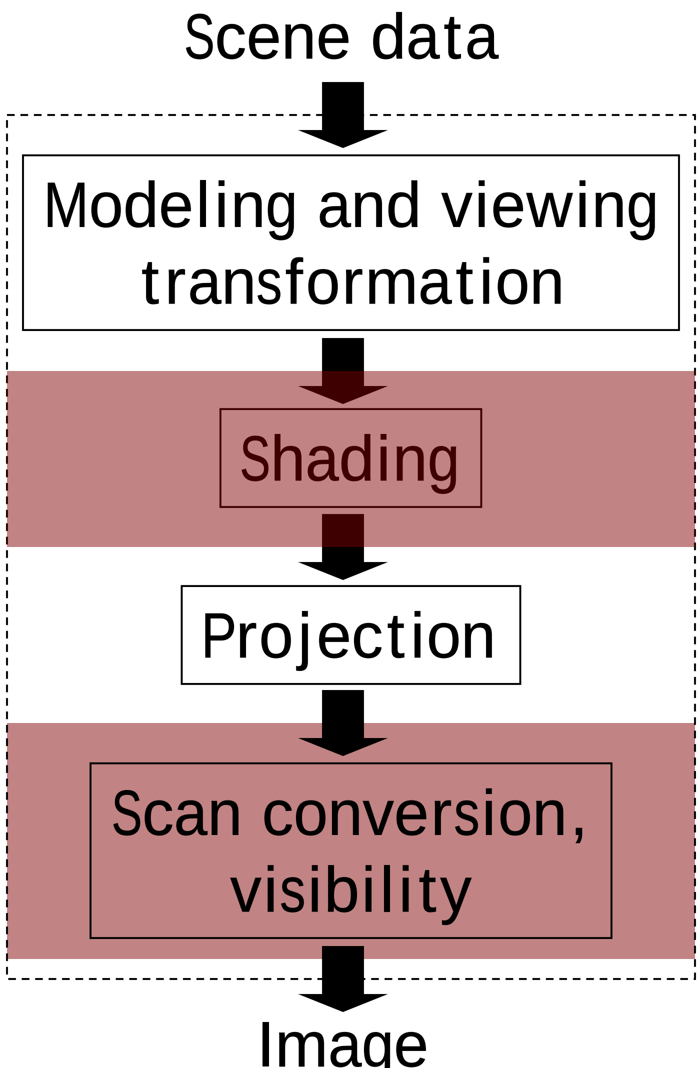
- No physics-based simulation
- Simplified models
- Reproduce perceptually most important effects
- Local illumination
  - Only one bounce of light between light source and viewer






UCSD

# Rendering Pipeline



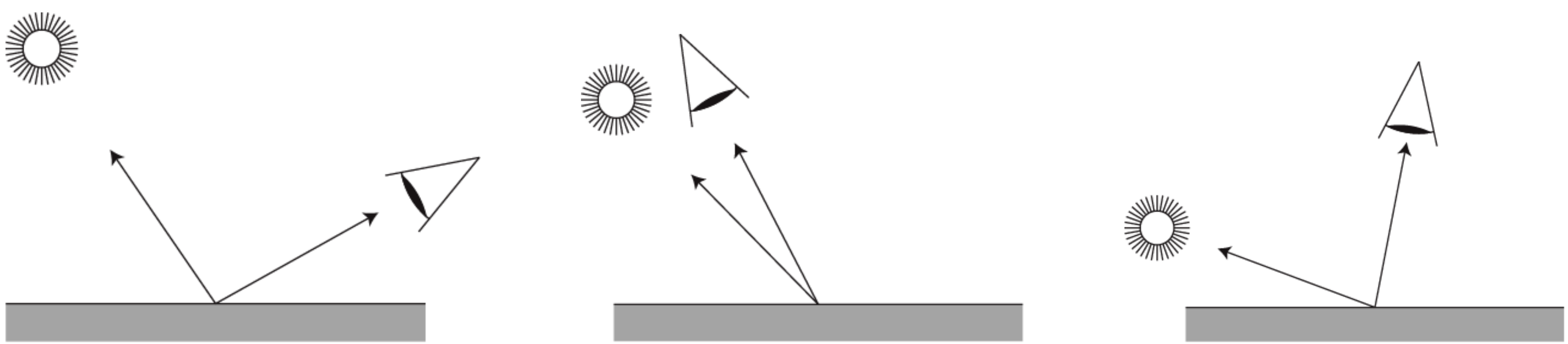

- Position object in 3D
- Determine colors of vertices
  - Per vertex shading
- Map triangles to 2D
- Draw triangles
  - Per pixel shading

# Local Illumination

- Gives material its color
- Light can be reflected by
  - Mirror
  - White wall
  - Glossy metal
  - etc.

# Local Illumination

- Model reflection of light at surfaces
  - Assumption: no subsurface scattering
- Bidirectional reflectance distribution function (BRDF)
  - Given light direction, viewing direction, how much light is reflected towards the viewer
  - For any pair of light/viewing directions!



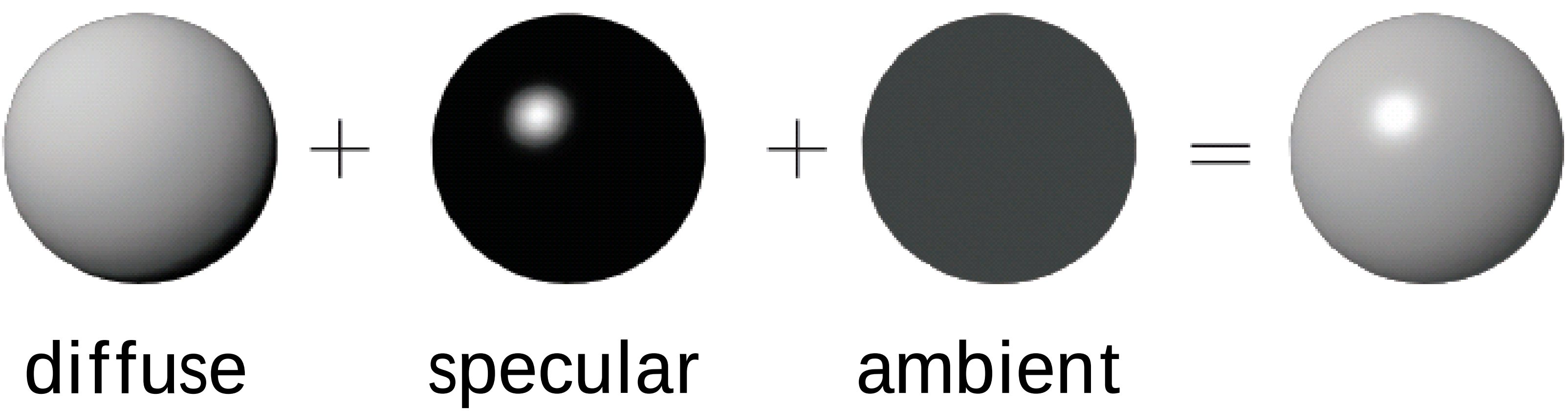


UCSD

# Local Illumination

**Simplified model**

- Sum of 3 components
- Covers a large class of real surfaces



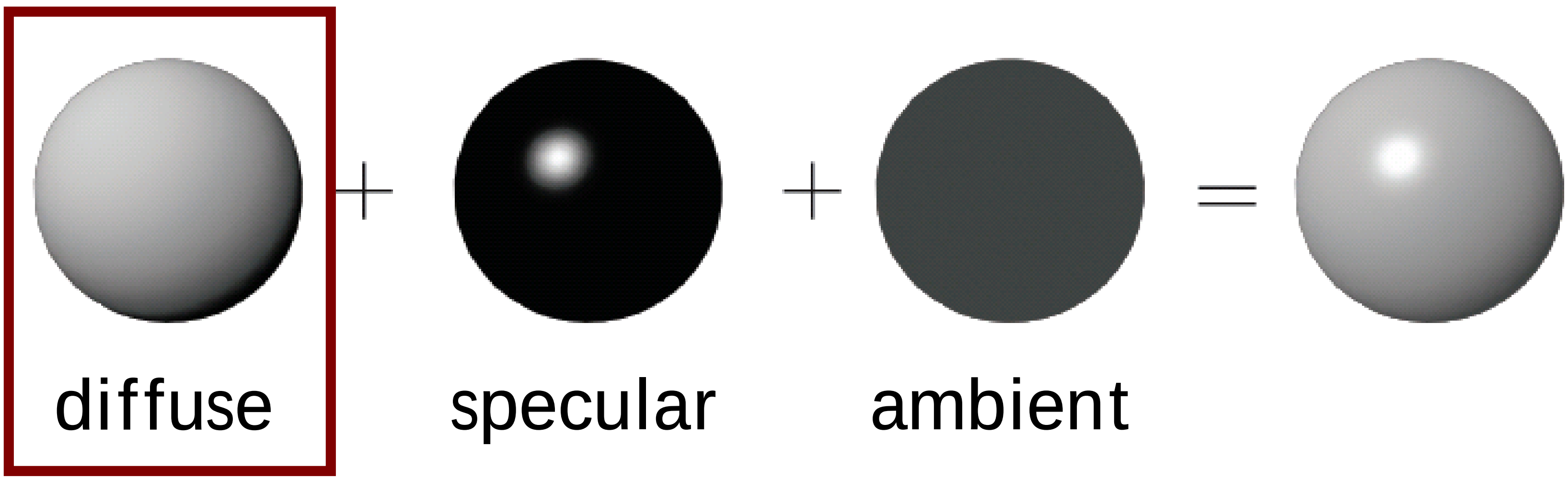



UCSD

# Local Illumination

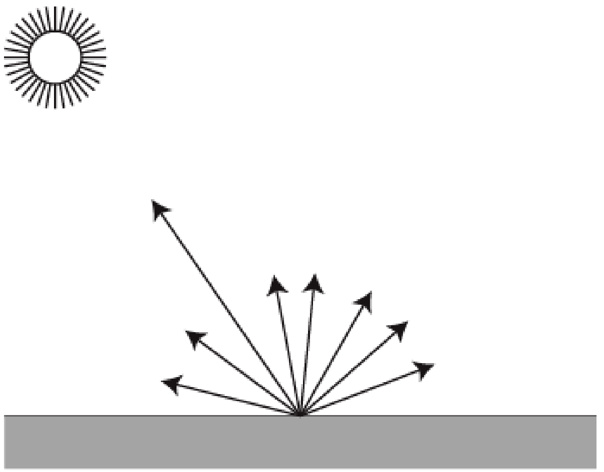
## Simplified model

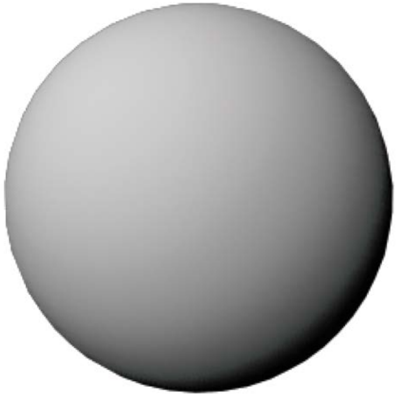
- Sum of 3 components
- Covers a large class of real surfaces



# Diffuse Reflection

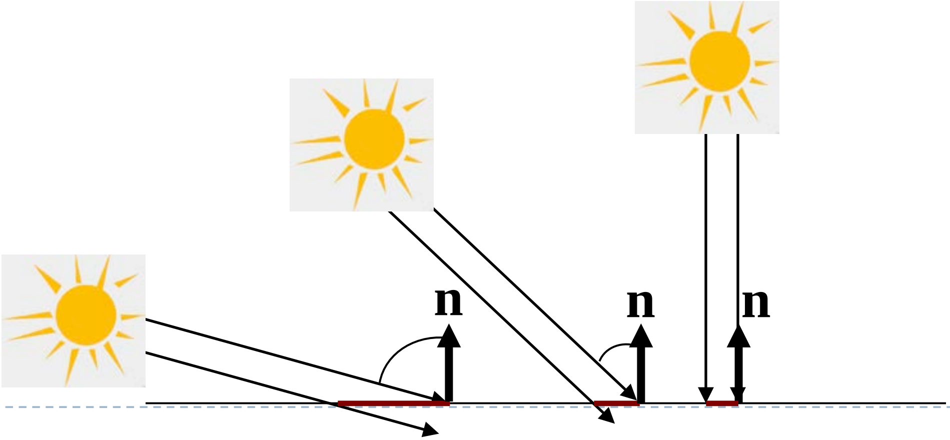
- Ideal diffuse material reflects light equally in all directions
- View-independent
- Matte, not shiny materials
  - Paper
  - Unfinished wood
  - Unpolished stone





# Diffuse Reflection

- Beam of parallel rays shining on a surface
  - Area covered by beam varies with the angle between the beam and the normal
  - The larger the area, the less incident light per area
  - Incident light per unit area is proportional to the cosine of the angle between the normal and the light rays
- Object darkens as normal turns away from light
- Lambert's cosine law (Johann Heinrich Lambert, 1760)
- Diffuse surfaces are also called Lambertian surfaces






UCSD

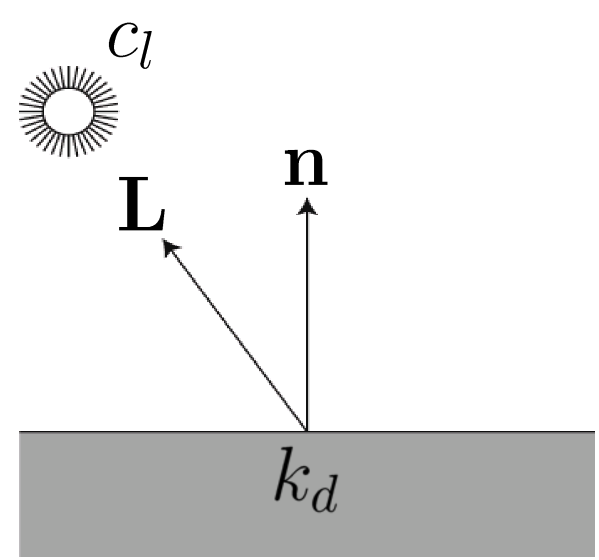
# Diffuse Reflection

- Given
  - Unit (normalized!) surface normal **n**
  - Unit (normalized!) light direction **L**
  - Material diffuse reflectance (material color) $k_d$
  - Light color (intensity) $c_l$

- Diffuse color $c_d$ is:

$$c_d = c_l k_d (\mathbf{n} \cdot \mathbf{L})$$

Proportional to cosine between normal and light






UCSD

# Diffuse Reflection

**Notes**

- Parameters $k_d$, $c_l$ are r,g,b vectors
- Need to compute r,g,b values of diffuse color $c_d$ separately
- Parameters in this model have no precise physical meaning
  - $c_l$: strength, color of light source
  - $k_d$: fraction of reflected light, material color

# Diffuse Reflection

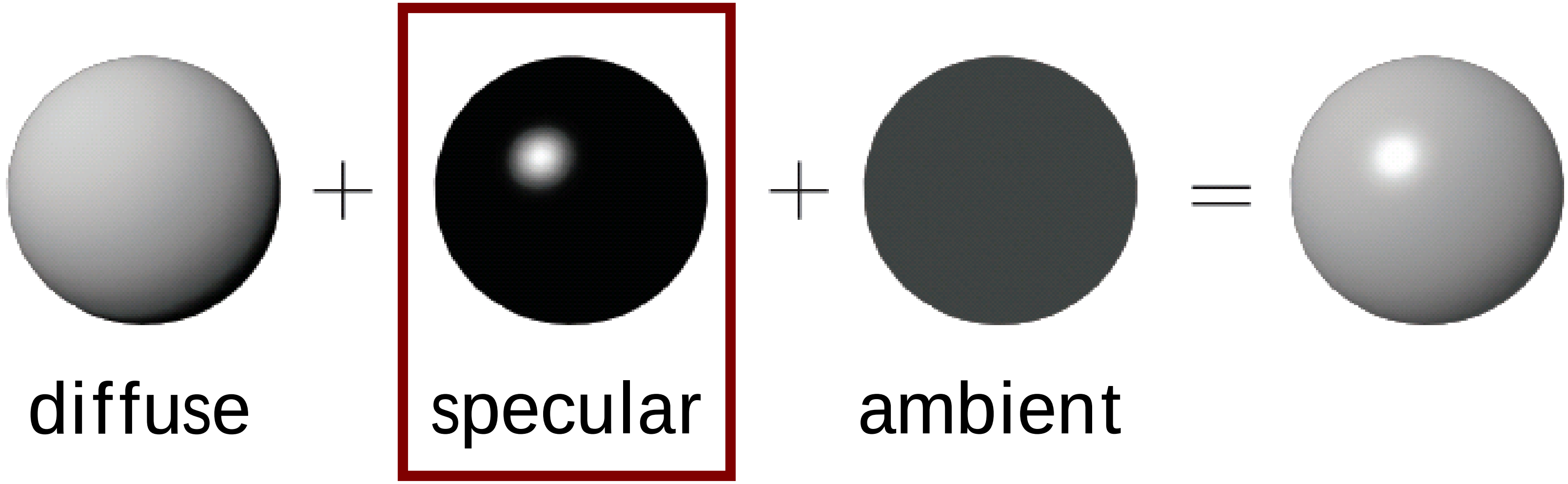
- Provides visual cues
  - Surface curvature
  - Depth variation

Lambertian (diffuse) sphere under different lighting directions

# Local Illumination

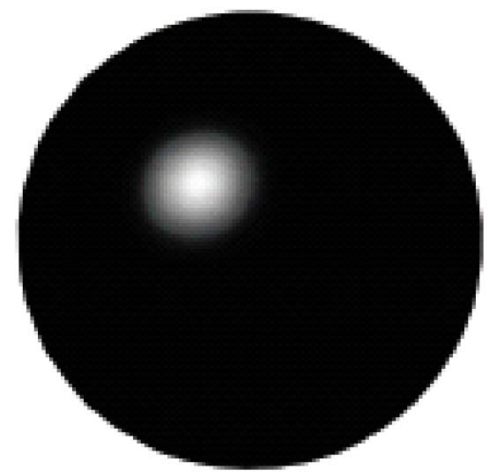
**Simplified model**

- Sum of 3 components
- Covers a large class of real surfaces



# Specular Reflection

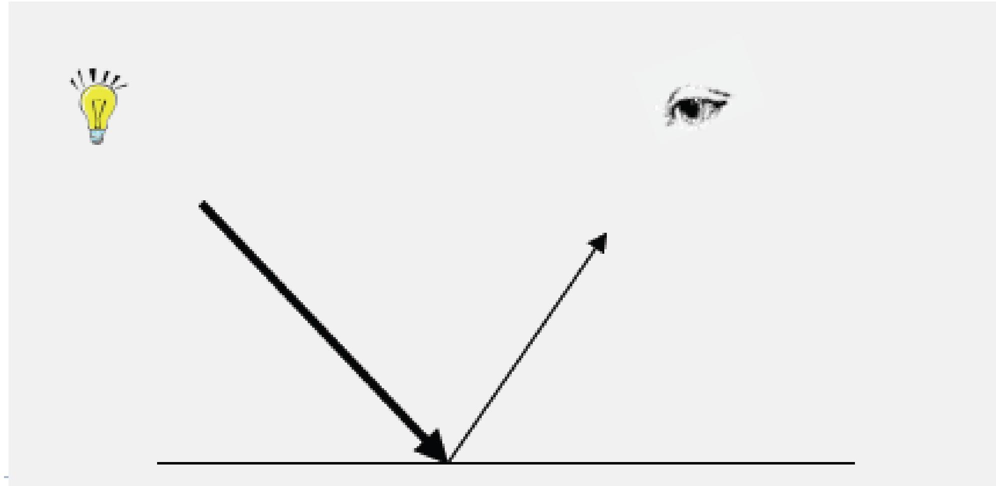
- Shiny surfaces
  - Polished metal
  - Glossy car finish
  - Plastics
- Specular highlight
  - Blurred reflection of the light source
  - Position of highlight depends on viewing direction



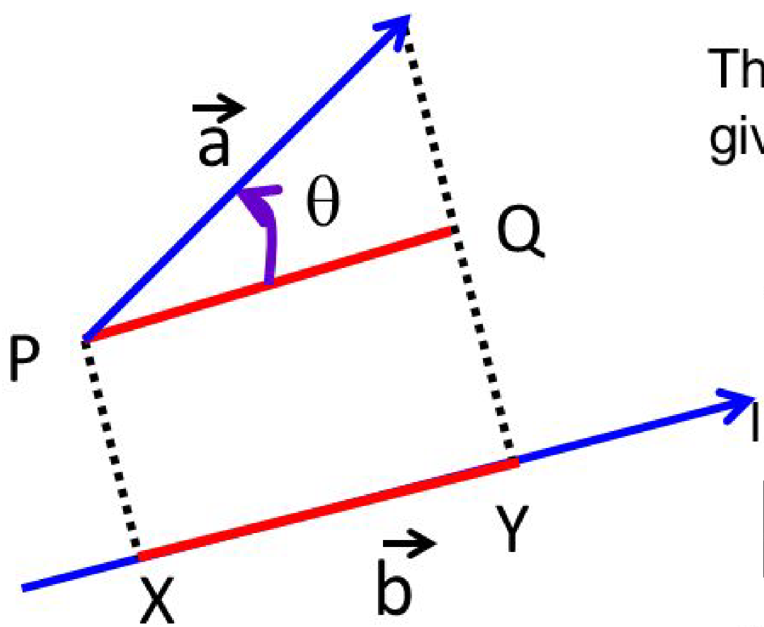
Specular highlight

# Specular Reflection

- Ideal specular reflection is mirror reflection
  - Perfectly smooth surface
  - Incoming light ray is bounced in single direction
  - Angle of incidence equals angle of reflection



# Projection of vector on another vector




Projection of $\vec{a}$ on $\vec{b}$ is XY

The projection of **a** onto **b** will be given by:

$$proj_b a = |a| \cos\theta \frac{\mathrm{b}}{|\mathrm{b}|}$$

In summary, the $proj_a$ b has length $|a|\cos\theta$, and direction $\frac{b}{|b|}$

It is called the scalar component of **a** in the direction of **b**



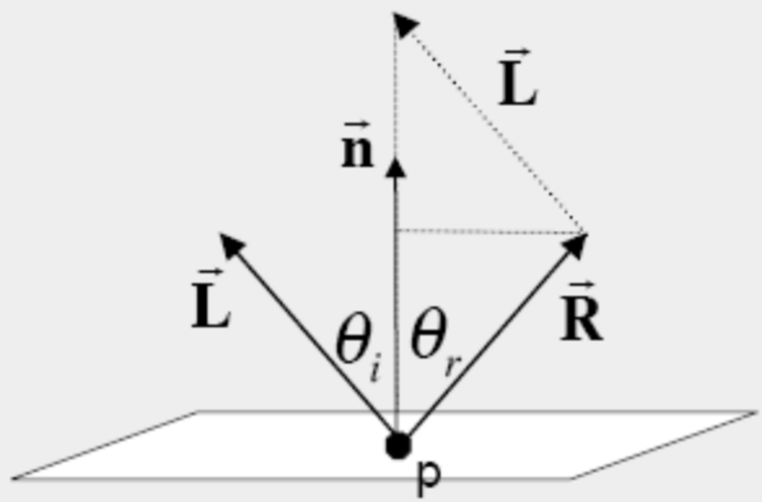
UCSD

# Law of Reflection

- Angle of incidence equals angle of reflection

$$\vec{\mathbf{R}} + \vec{\mathbf{L}} = 2\cos\theta\ \vec{\mathbf{n}} = 2\left(\vec{\mathbf{L}} \cdot \vec{\mathbf{n}}\right)\vec{\mathbf{n}}$$

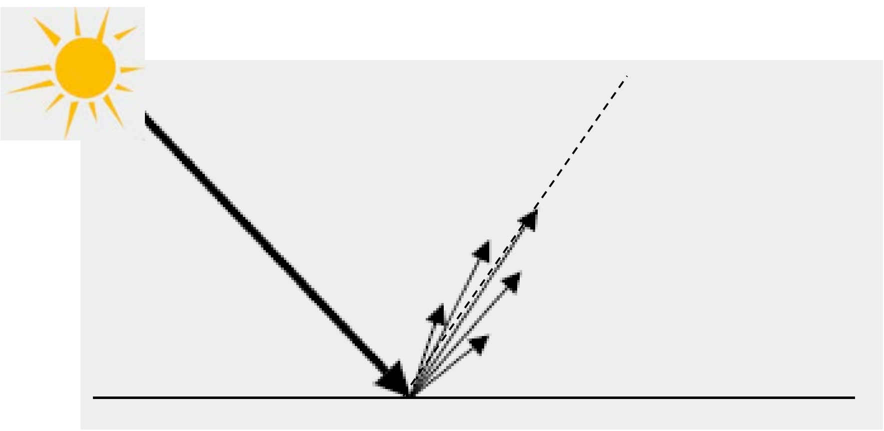
$$\vec{\mathbf{R}} = 2\left(\vec{\mathbf{L}} \cdot \vec{\mathbf{n}}\right)\vec{\mathbf{n}} - \vec{\mathbf{L}}$$




$$\theta_r = \theta_i$$

# Specular Reflection

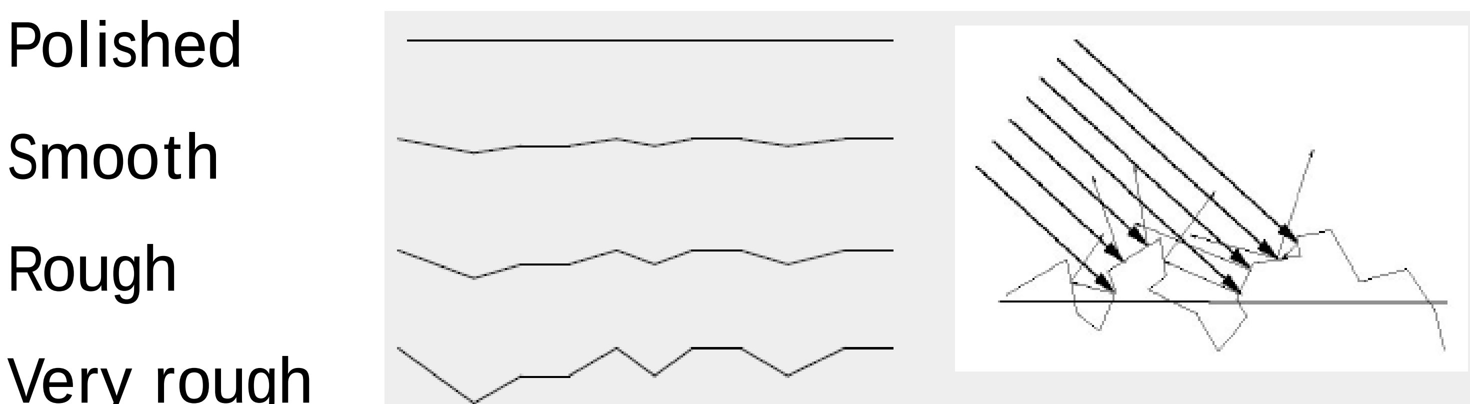
- Many materials are not perfect mirrors
  - Glossy materials



Glossy teapot

# Glossy Materials

- Assume surface composed of small mirrors with random orientation (micro-facets)
- Smooth surfaces
  - Micro-facet normals close to surface normal
  - Sharp highlights
- Rough surfaces
  - Micro-facet normals vary strongly
  - Blurry highlight





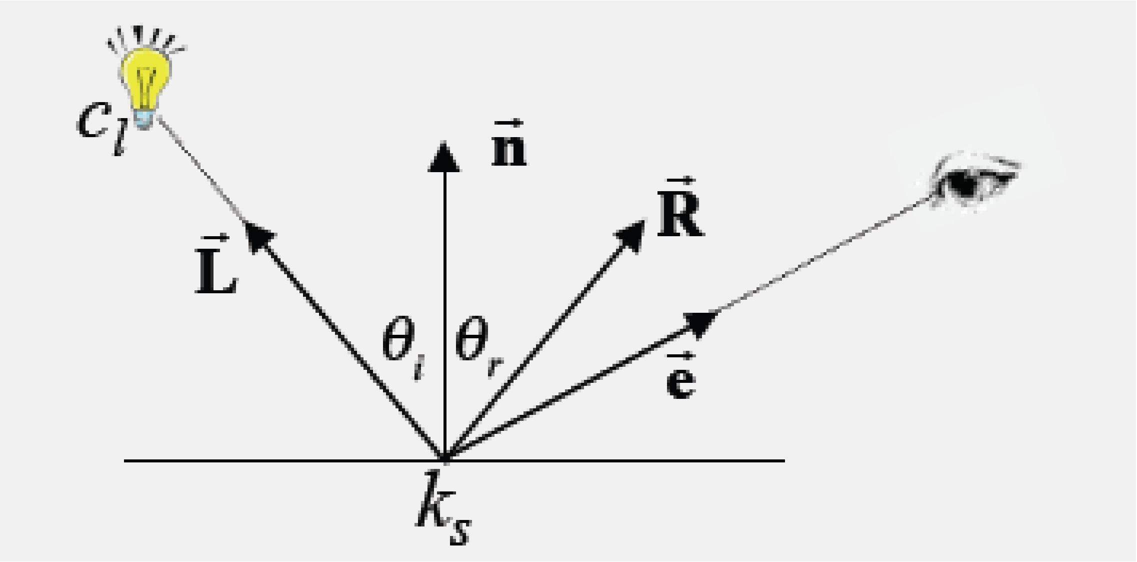

UCSD

# Glossy Surfaces

- Expect most light to be reflected in mirror direction
- Because of micro-facets, some light is reflected slightly off ideal reflection direction
- Reflection
  - Brightest when view vector is aligned with reflection
  - Decreases as angle between view vector and reflection direction increases

# Phong Shading Model

- Developed by Bui Tuong Phong in1973
- Specular reflectance coefficient $k_s$
- Phong exponent $p$
  - Greater $p$ means smaller (sharper) highlight

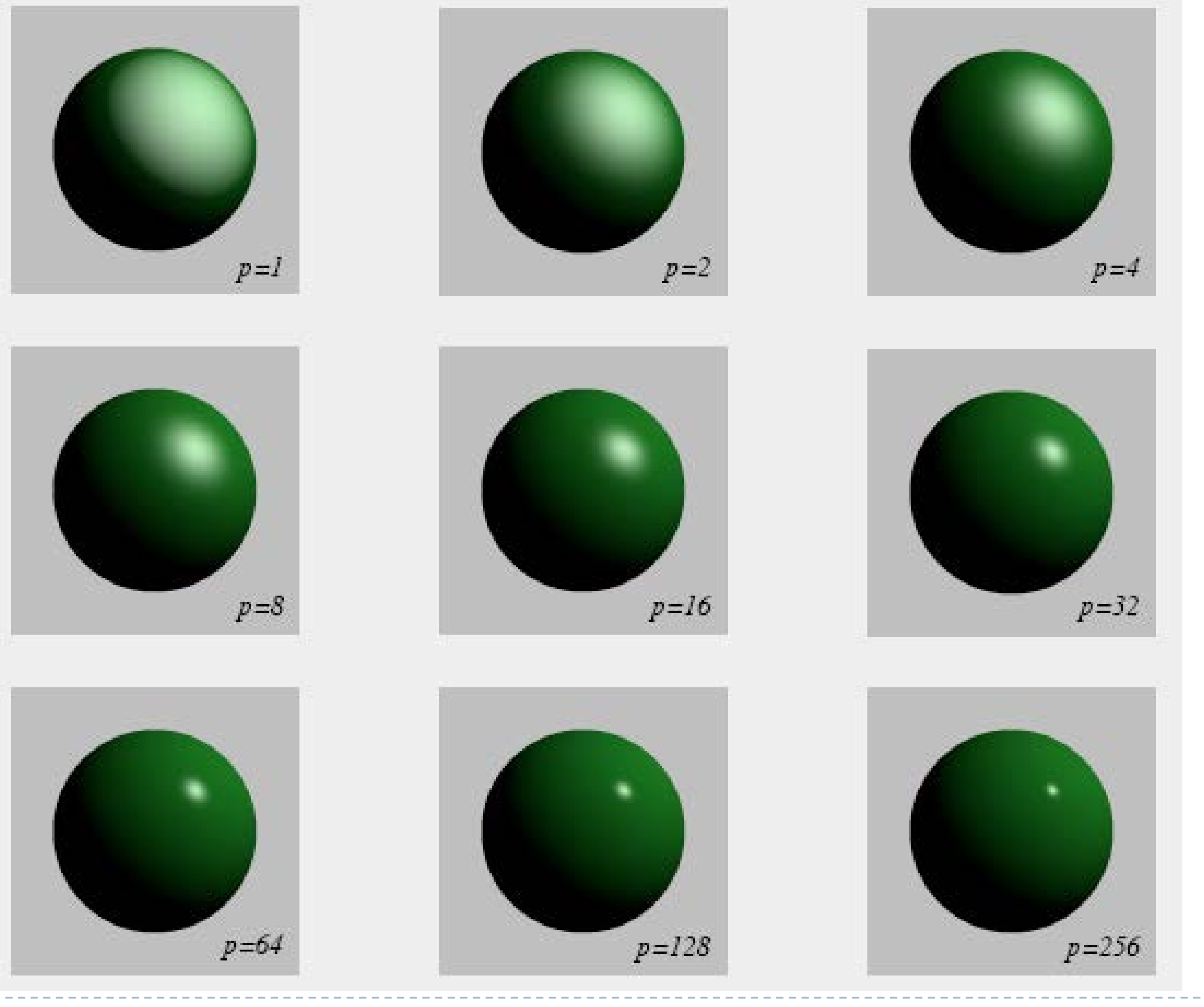



$$c = k_s c_l \left(\mathbf{R} \cdot \mathbf{e}\right)^p$$



UCSD

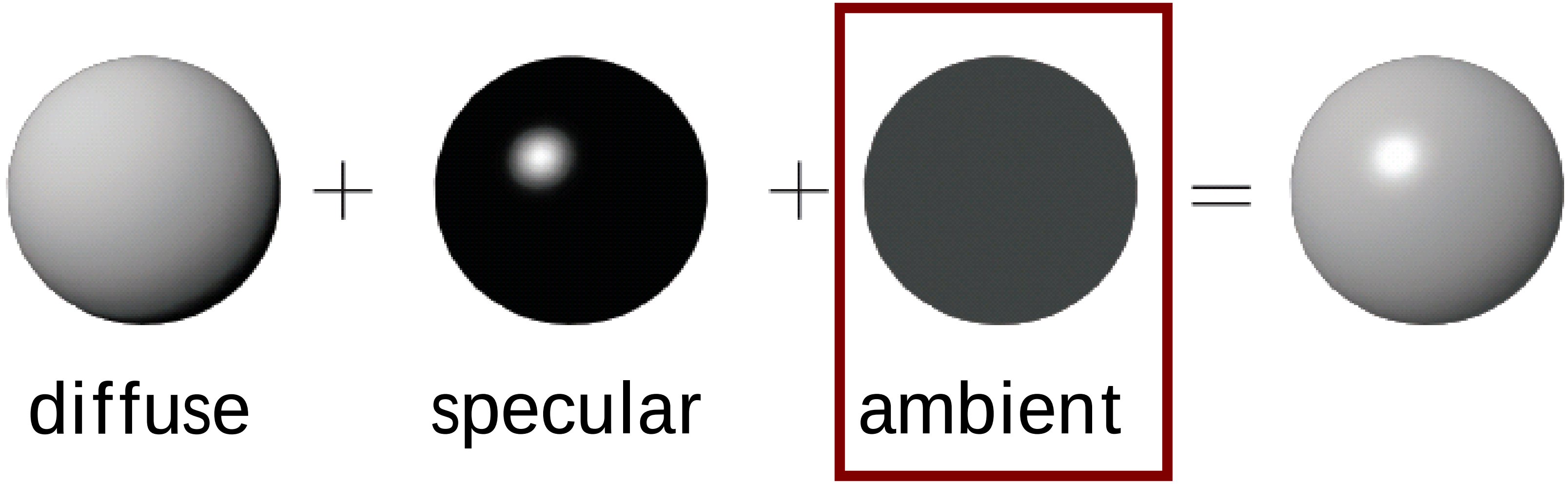
# Phong Shading Model



# Local Illumination

**Simplified model**

- Sum of 3 components
- Covers a large class of real surfaces



# Ambient Light

- In real world, light is bounced all around scene
- Could use global illumination techniques to simulate
- Simple approximation
  - Add constant ambient light at each point: $k_a c_a$
  - Ambient light color: $c_a$
  - Ambient reflection coefficient: $k_a$
- Areas with no direct illumination are not completely dark

# Complete Phong Shading Model

- Phong model supports multiple light sources
- All light colors *c* and material coefficients *k* are 3-component vectors for red, green, blue

$$c = \sum_i c_{l_i}(k_d(L_i \cdot n) + k_s(R \cdot e)^p + k_a)$$

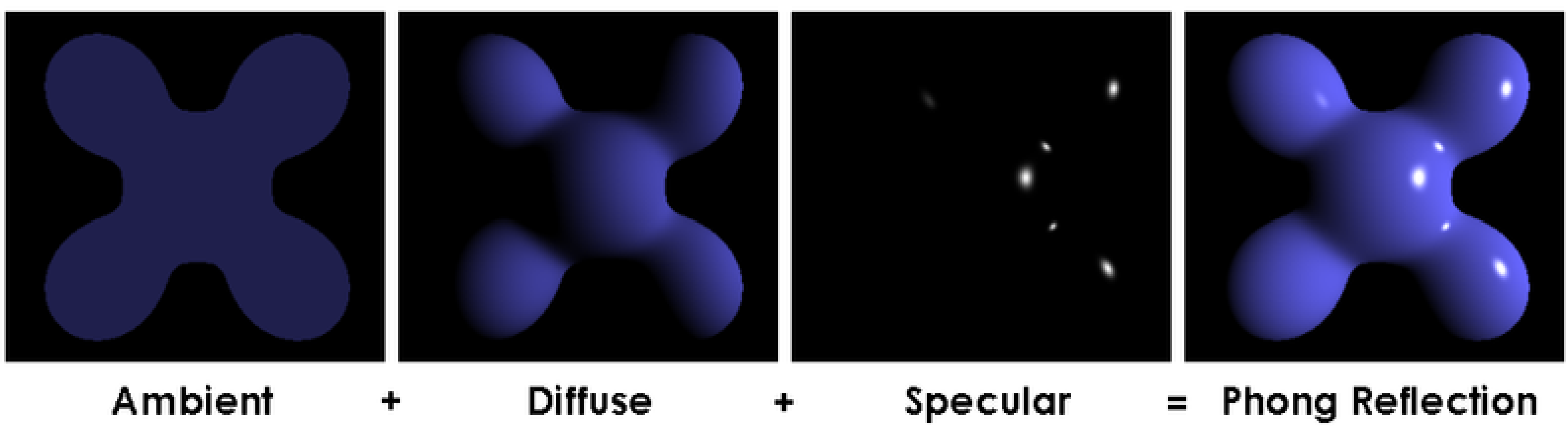



*Image by Brad Smith*


UCSD