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

Jürgen P. Schulze, Ph.D.
University of California, San Diego
Fall Quarter 2017

Announcements

- Project 2 due next Friday at 2pm
- This Friday: late grading 2-3:15pm
- Projects in need of help:
  - WIFIRE app: CSE 199
    - https://wifire.ucsd.edu/
  - ARCore tweaking: CSE 199
  - SPAWAR (see following slides): internship

Overview:

The purpose of this task is to provide analytical and/or technical support in the fields of Computer Science, Engineering, and Cognitive Science to assist with the Unmanned Underwater Vehicles (UUV) Lab. This project looks to explore and develop autonomy solutions for the warfighter.

Responsibilities:

- Reviewing, evaluating and determining ways in which new technologies will increase mission performance.
- Designing, exercising and/or testing interaction between the operator and the autonomous agents.
- Supporting data analyses and documenting findings.

Qualifications:

- Programming experience in Java or C/C++
- Additional Qualifications desired: Experience with eye tracking software and analysis, human factors/human-machine interface skills

Students positions are limited to 20 hours a week, except for the summer. We are interested in getting senior undergraduate and graduate students. A SECRET clearance is required for this position (will be done by SPAWAR)

Send your resume to maria.rodas@navy.mil, or email me

Application deadline:
October 31, 2017

http://nreip.asee.org/promotional_poster

NREIP

Naval Research Enterprise Internship Program

Office of Naval Research

Naval Research Enterprise Internship Program

http://nreip.asee.org/apply

nreip@asee.org 202-649-3833

Program Information

- 10 week summer research opportunities at a Naval Research Laboratory or a Naval System Command Laboratory
- Stipends: $5,400 31-60 Credit Hours
  $8,100 61+ Credit Hours
  $10,800 Graduate Students

Eligibility Requirements

- Must be a U.S. citizen (dual citizens and permanent residents will be considered at some labs)
- Must have 31 or more college credits
- Must be currently enrolled at a 4 year U.S. accredited college/university (students attending 2 year schools may be eligible at the laboratory's discretion)

Participating Laboratories

- Arizona, California, Connecticut, DC, Florida, Hawaii, Indiana, Maryland, Mississippi, New Jersey, Pennsylvania, Rhode Island, South Carolina, Texas, Virginia, Washington

Application Opens: August 21, 2017

Office of Naval Research ONR Science & Technology

ASEE

The Department of the Navy is an Equal Employment Opportunity employer. All qualified candidates will receive consideration without regard to race, color, religion, sex, national origin, age, disability, marital status, political affiliation, sexual orientation, or any other non-merit factors. The Department of the Navy provides reasonable accommodation to applicants with disabilities.

Lecture Overview

- Phong Reflection Model
- Light Sources

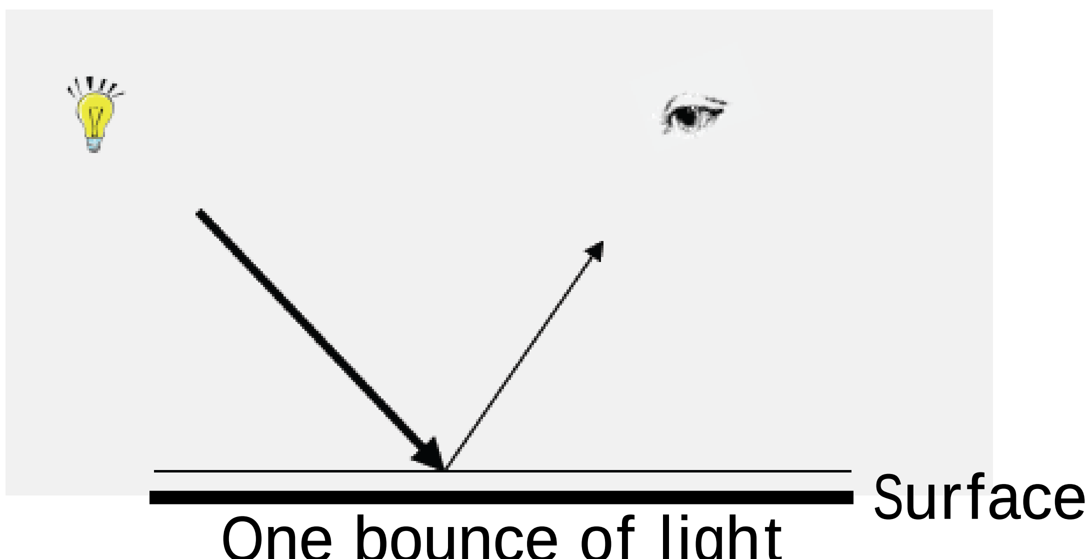
Shading

- 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

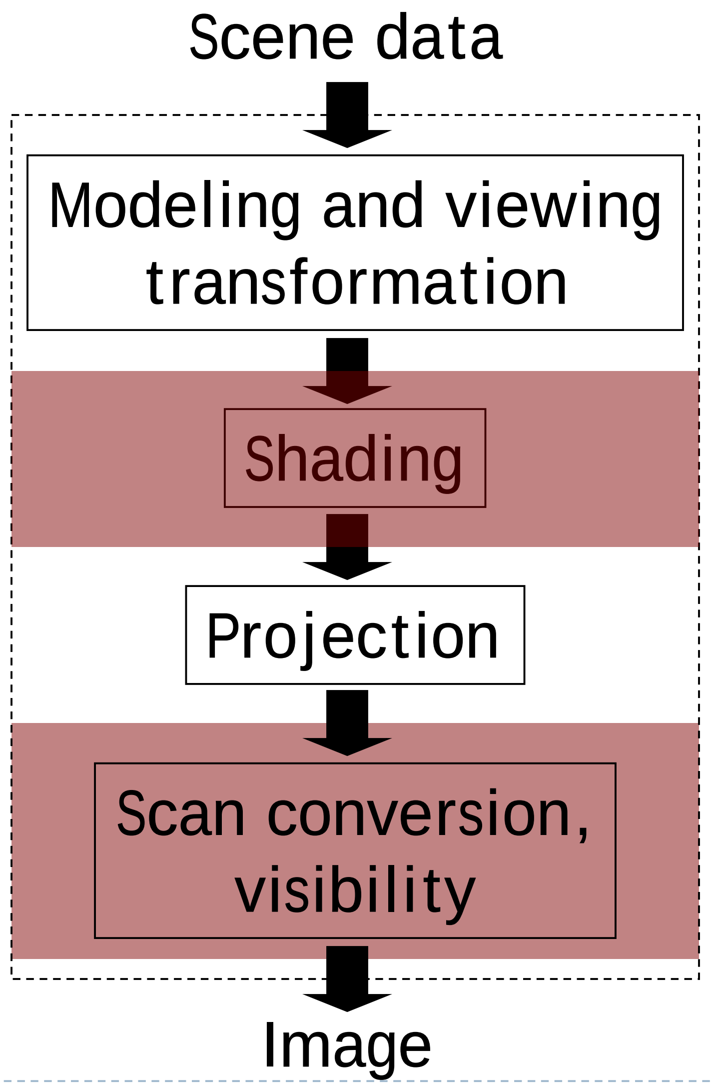
Interactive Applications

- 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



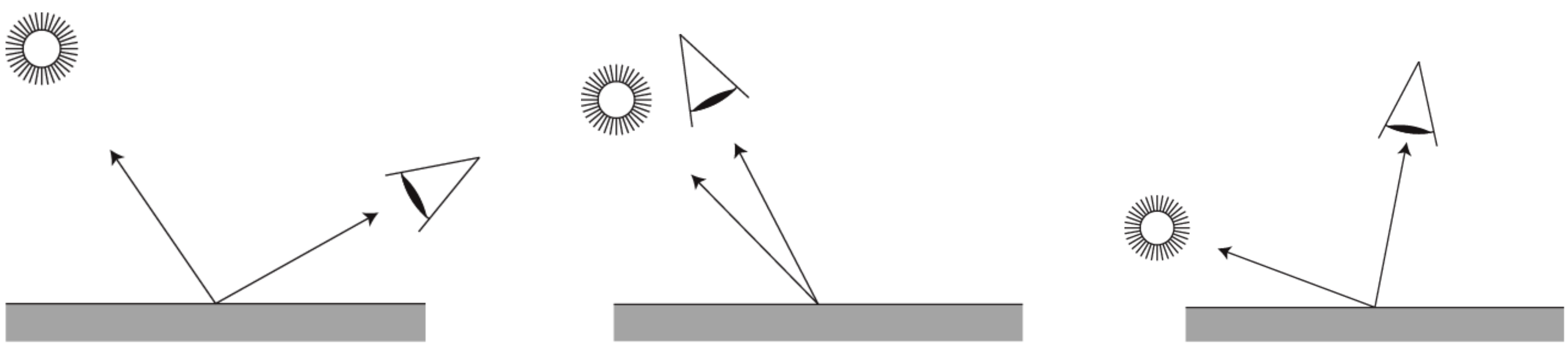
- Position object in 3D
- Determine colors of vertices
  - Per vertex shading
- Map triangles to 2D
- Draw triangles
  - Per pixel shading

Local Illumination

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

Local Illumination

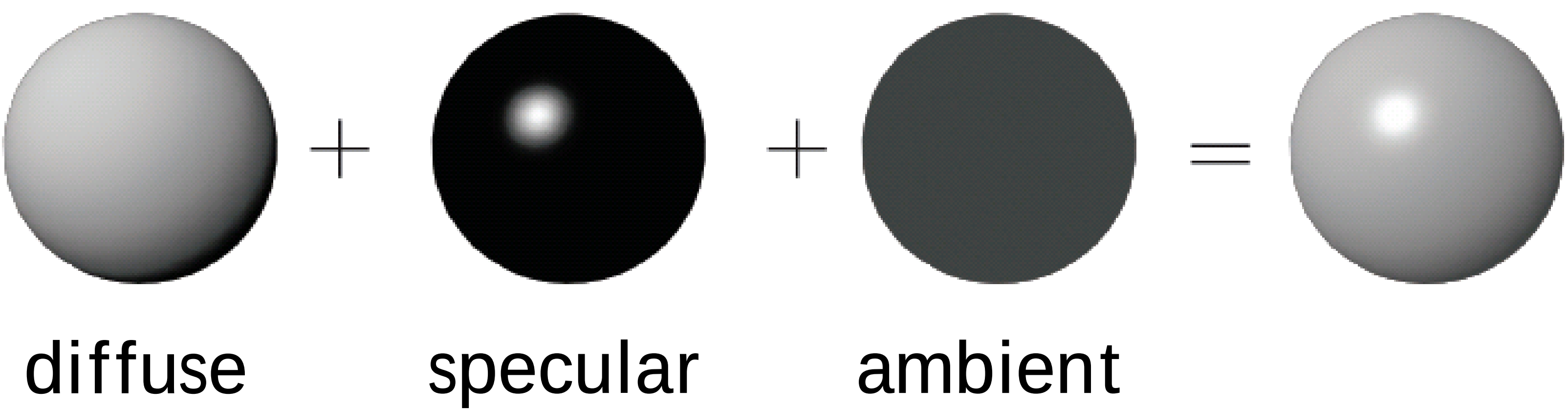
- Model reflection of light at surfaces
  - Assumption: no subsurface scattering
- Bidirectional reflectance distribution function (BRDF)
  - Given light direction, viewing direction, how much light is reflected towards the viewer
  - For any pair of light/viewing directions!



Local Illumination

Simplified model

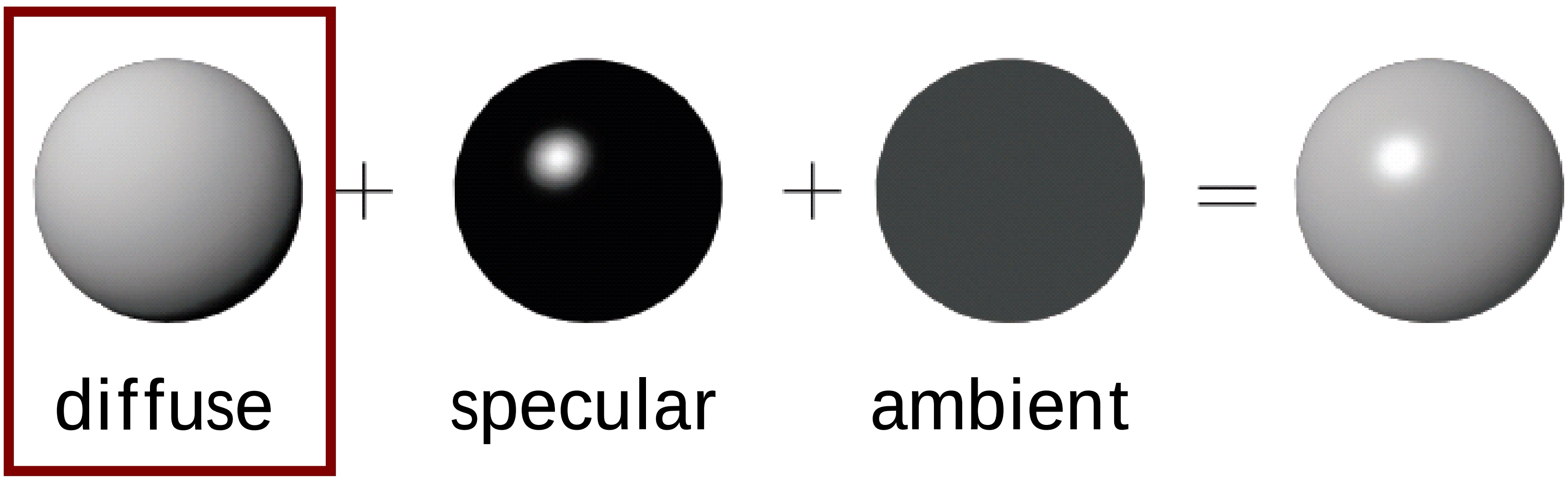
- Sum of 3 components
- Covers a large class of real surfaces



Local Illumination

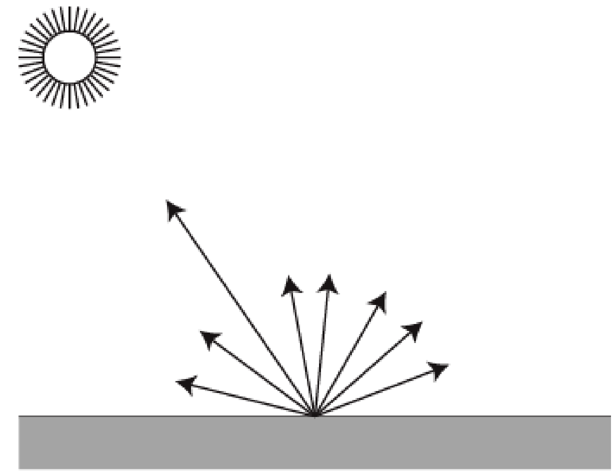
Simplified model

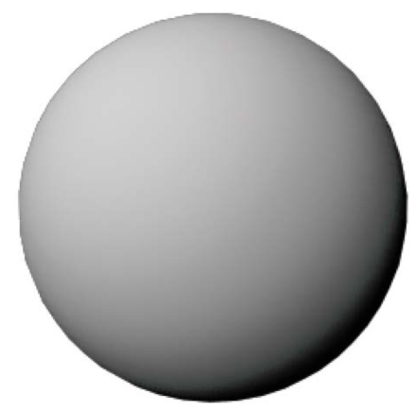
- Sum of 3 components
- Covers a large class of real surfaces



Diffuse Reflection

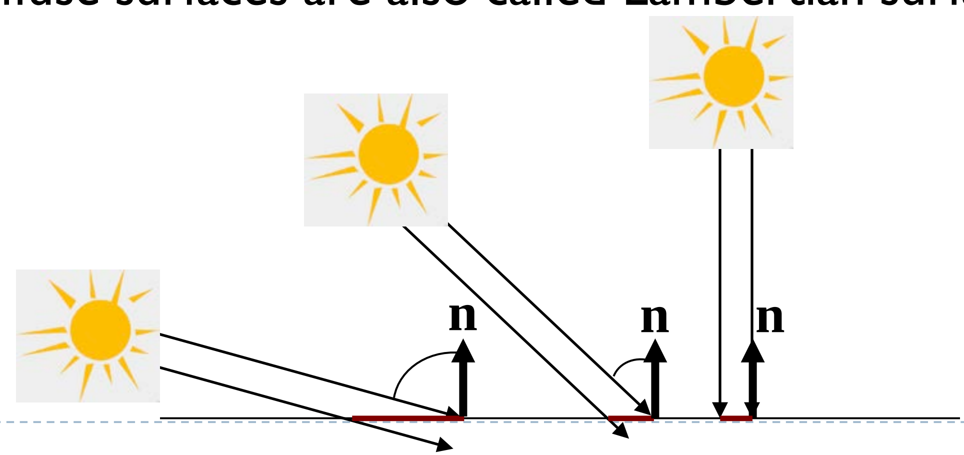
- Ideal diffuse material reflects light equally in all directions
- View-independent
- Matte, not shiny materials
  - Paper
  - Unfinished wood
  - Unpolished stone

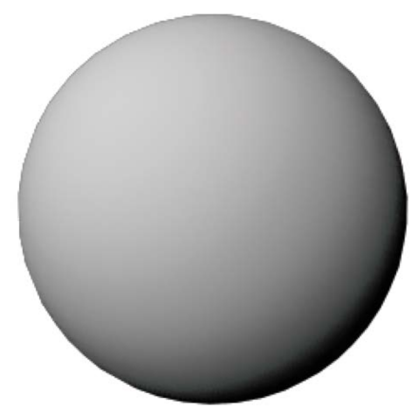




Diffuse Reflection

- 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





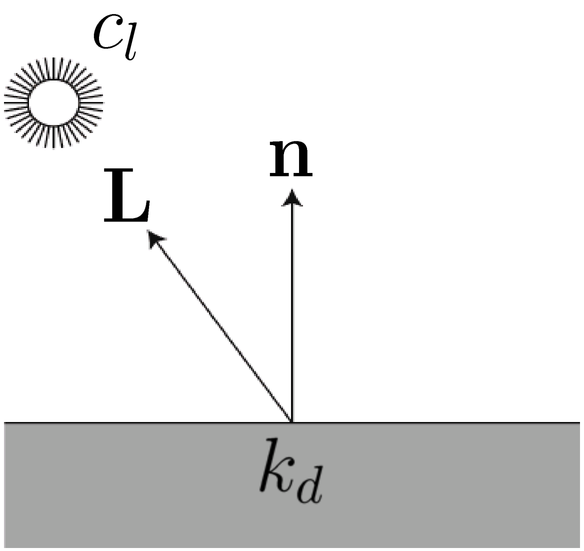
Diffuse Reflection

- 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:

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

Proportional to cosine between normal and light



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

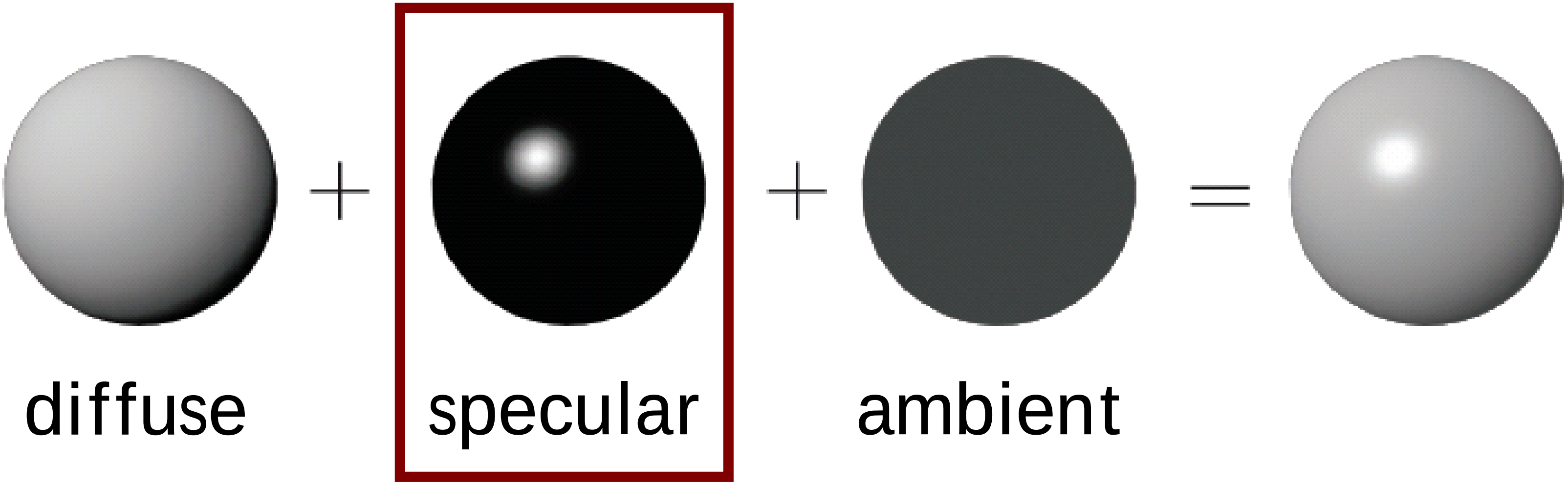
- Provides visual cues
  - Surface curvature
  - Depth variation

Lambertian (diffuse) sphere under different lighting directions

Local Illumination

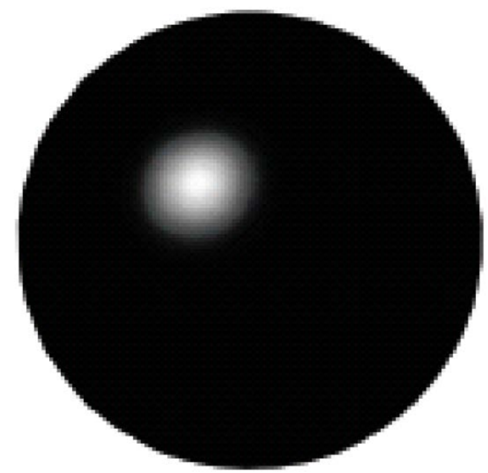
Simplified model

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



Specular Reflection

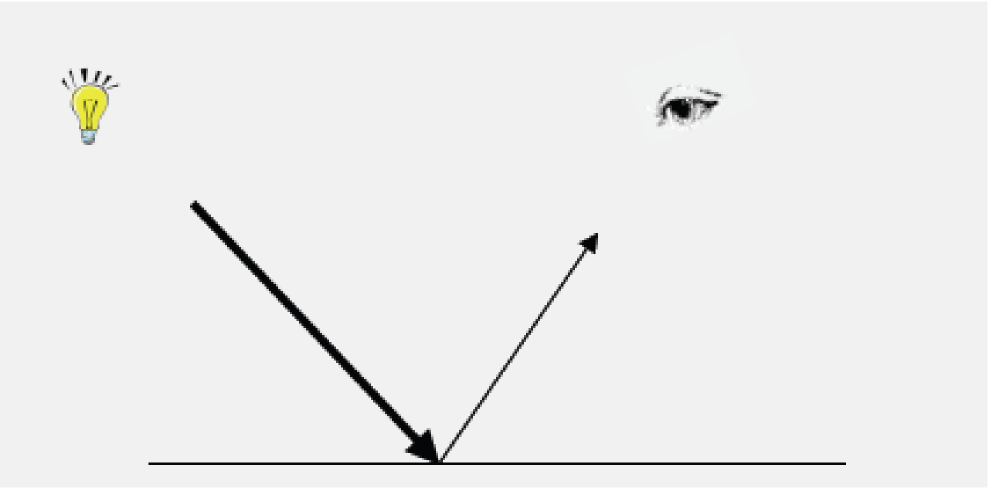
- Shiny surfaces
  - Polished metal
  - Glossy car finish
  - Plastics
- Specular highlight
  - Blurred reflection of the light source
  - Position of highlight depends on viewing direction



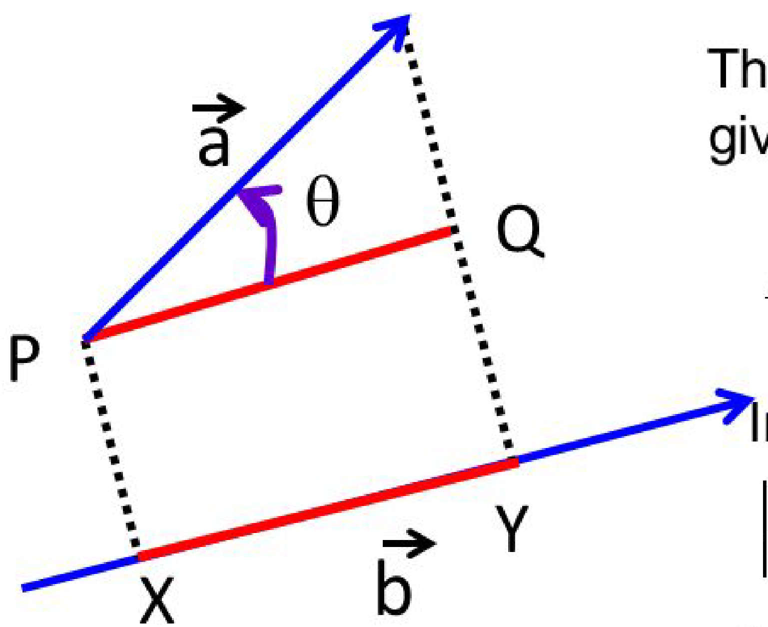
Specular highlight

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 $\frac{b}{|b|}$

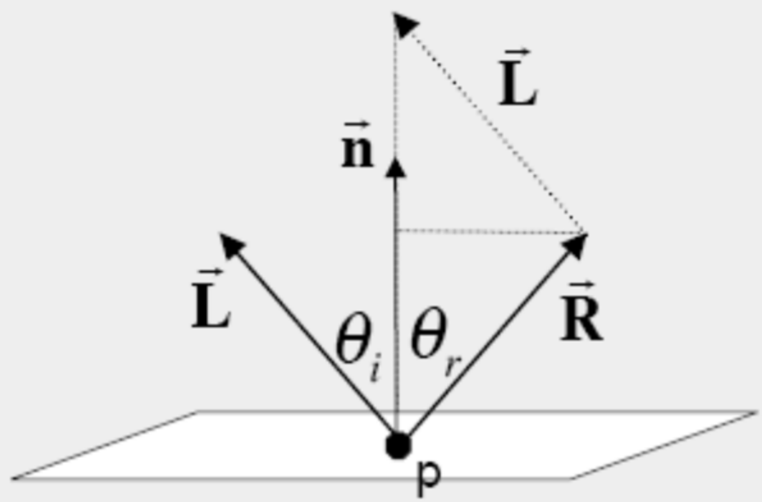
It is called the scalar component of **a** in the direction of **b**

Law of Reflection

- Angle of incidence equals angle of reflection

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

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

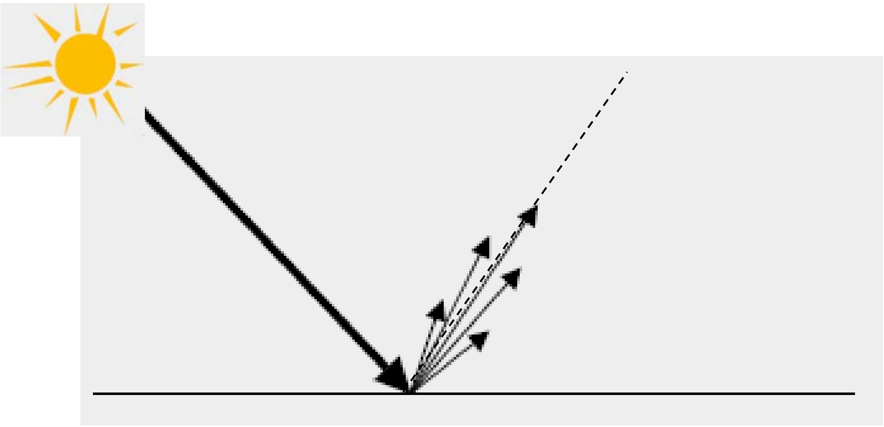


$$\theta_r = \theta_i$$

UCSD

Specular Reflection

- Many materials are not perfect mirrors
  - Glossy materials



Glossy teapot

Glossy Materials

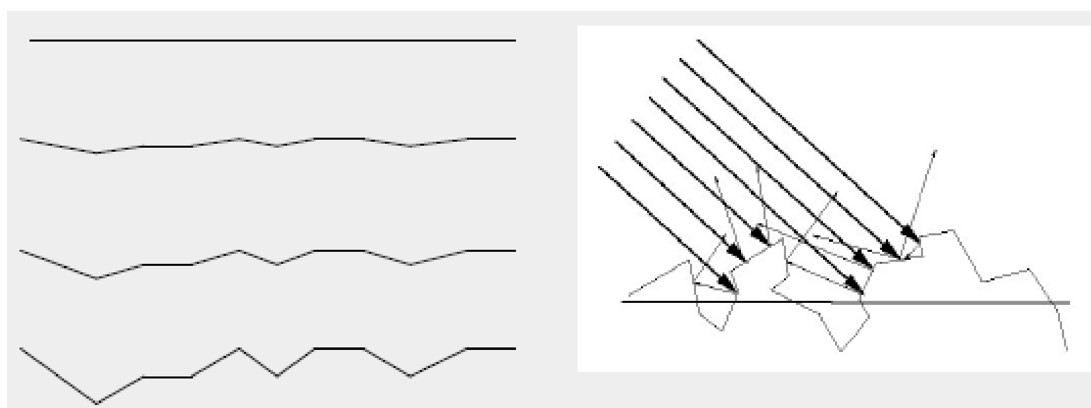
- Assume surface composed of small mirrors with random orientation (micro-facets)
- Smooth surfaces
  - Micro-facet normals close to surface normal
  - Sharp highlights
- Rough surfaces
  - Micro-facet normals vary strongly
  - Blurry highlight

Polished

Smooth

Rough

Very rough

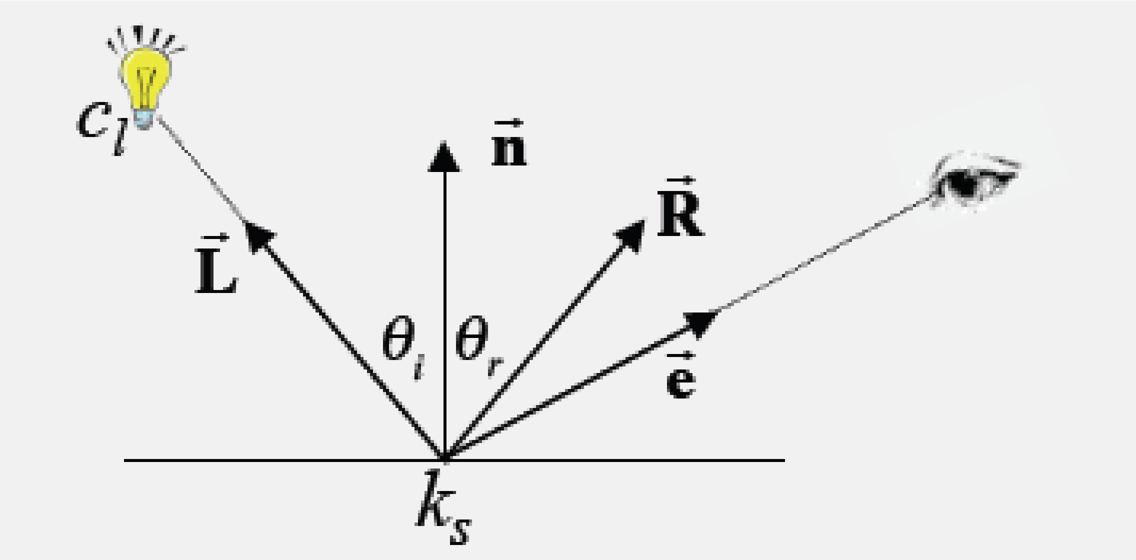


Glossy Surfaces

- 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

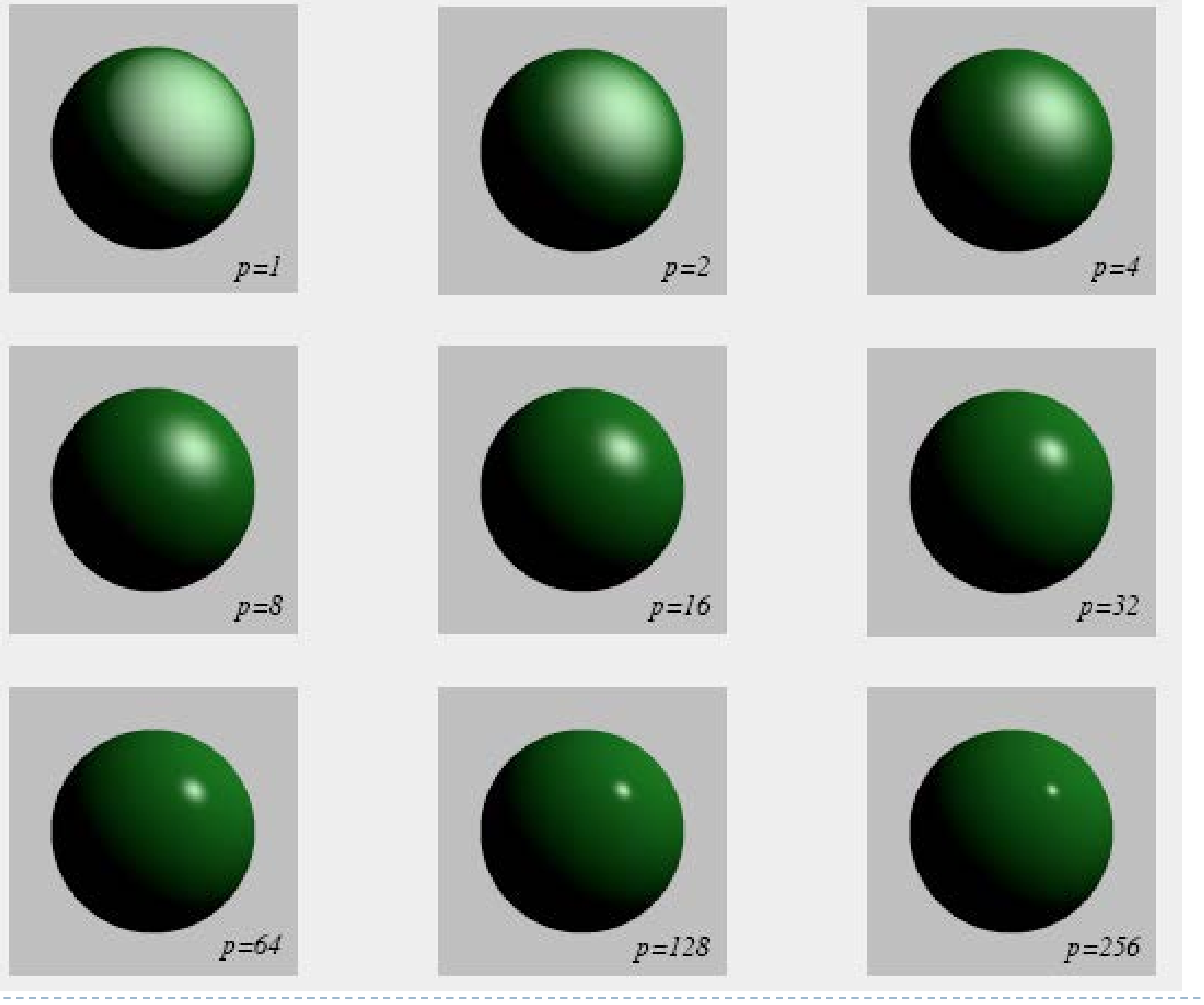
Phong Shading Model

- Developed by Bui Tuong Phong in 1973
- Specular reflectance coefficient k_s
- Phong exponent p
  - Greater p means smaller (sharper) highlight



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

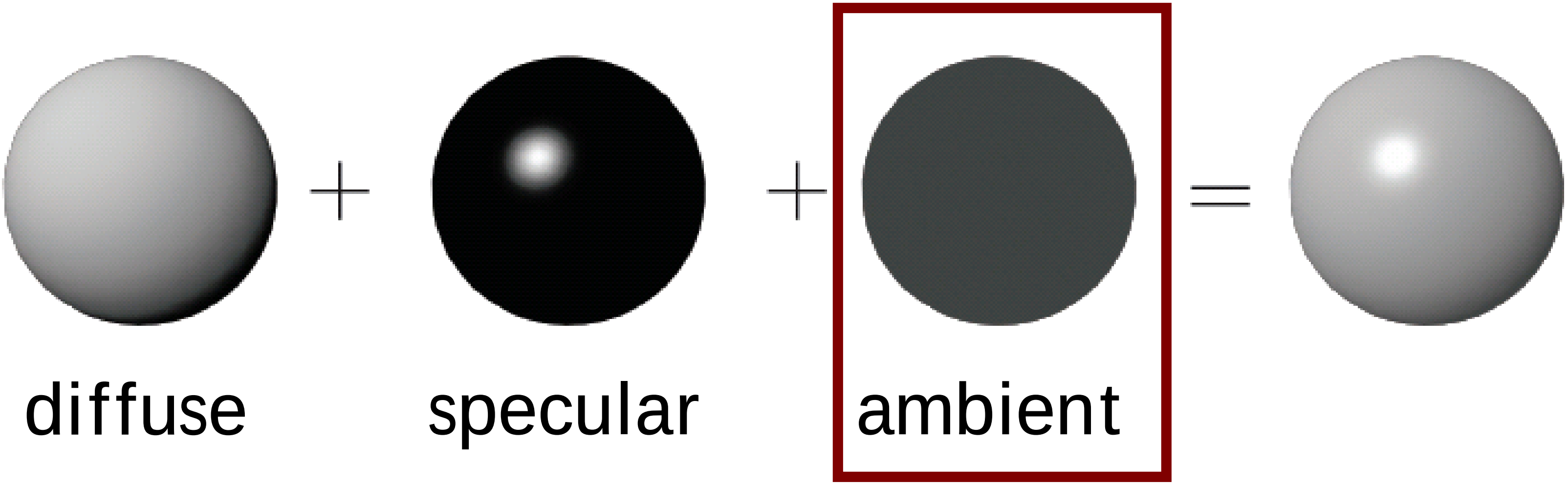
Phong Shading Model



Local Illumination

Simplified model

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



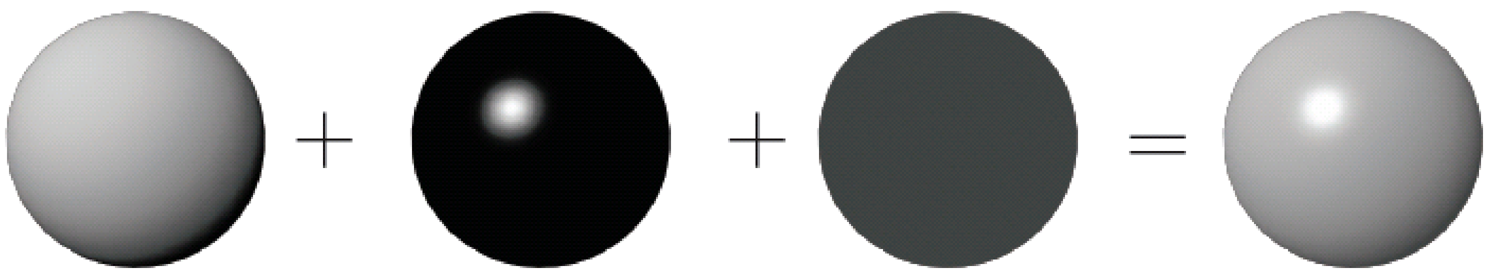
Ambient Light

- 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

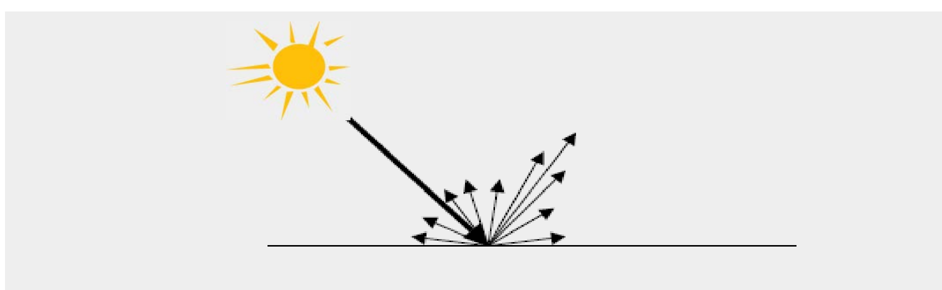
Complete Phong Shading Model

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

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



diffuse specular ambient



Lecture Overview

- Illumination
- Light Sources

Light Sources

- Real light sources can have complex properties
  - Geometric area over which light is produced
  - Anisotropy (directionally dependent)
  - Reflective surfaces act as light sources (indirect light)







- We have to use a drastically simplified model to allow real-time rendering

Types of Light Sources

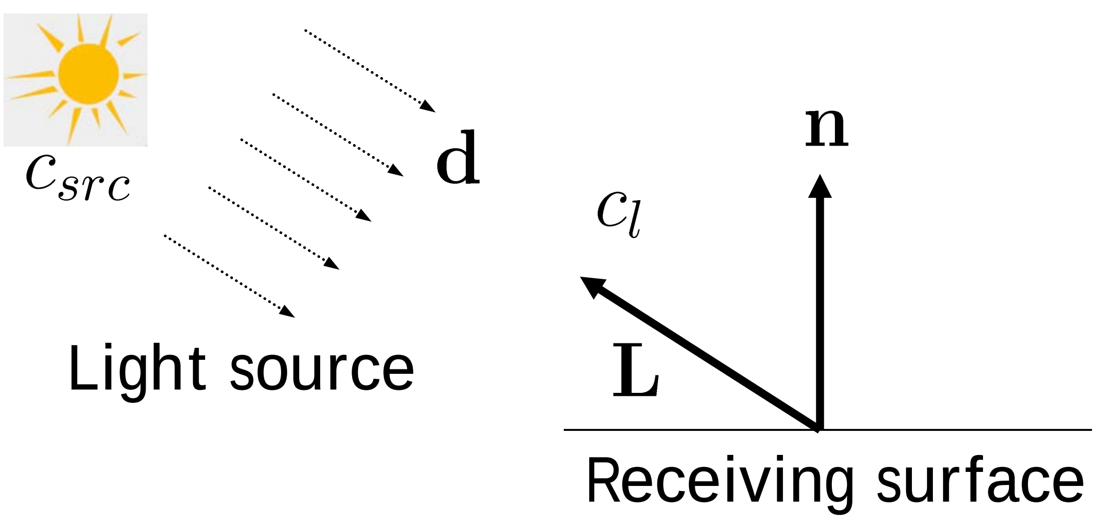
- At each point on surfaces we need to know
  - Direction of incoming light (the $\mathbf{L}$ vector)
  - Intensity of incoming light (the c_l values)
- Three light types:
  - Directional: from a specific direction
  - Point light source: from a specific point
  - Spotlight: from a specific point with intensity that depends on direction

Lecture Overview

- Illumination
- Light Sources
  - Directional Lights
  - Point Lights
  - Spot Lights

Directional Light

- Light from a distant source
  - Light rays are parallel
  - Direction and intensity are the same everywhere
  - As if the source were infinitely far away
  - Good approximation of sunlight
- Specified by a unit length direction vector, and a color



$$\mathbf{L} = -\mathbf{d}$$

$$c_l = c_{src}$$

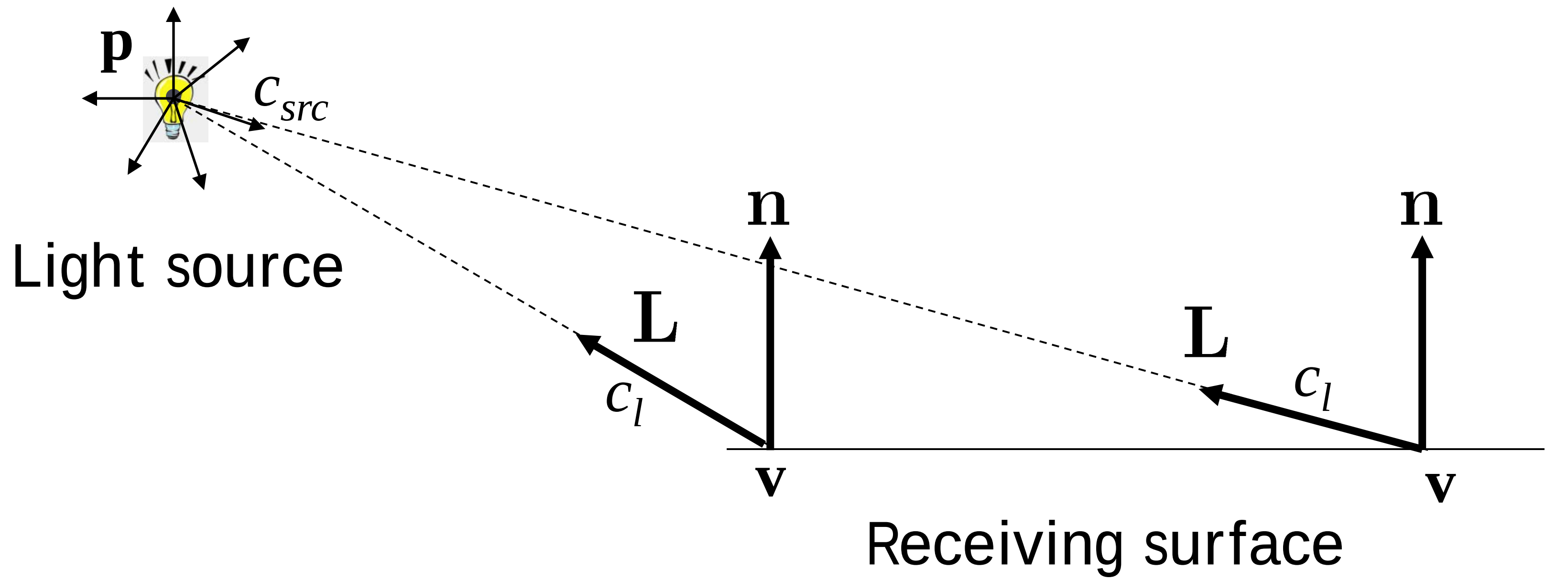
Lecture Overview

- Illumination
- Light Sources
  - Directional Lights
  - Point Lights
  - Spot Lights

Point Lights

- Similar to light bulbs
- Infinitely small point radiates light equally in all directions
  - Light vector varies across receiving surface
  - What is light intensity over distance proportional to?
  - Intensity drops off proportionally to the inverse square of the distance from the light
    - Reason for inverse square falloff:
      Surface area A of sphere:
      $A = 4 \pi r^2$

Point Light Math



At any point v on the surface:

$$\mathbf{L} = \frac{\mathbf{p}-\mathbf{v}}{\|\mathbf{p}-\mathbf{v}\|}$$

Attenuation:

$$c_l = \frac{c_{src}}{\|\mathbf{p}-\mathbf{v}\|^2}$$

Light Attenuation

- Adding constant factor k to denominator for better control
- Quadratic attenuation: $k*(p-v)^2$
  - Most computationally expensive, most physically correct
- Linear attenuation: $k*(p-v)$
  - Less expensive, less accurate
- Constant attenuation: k
  - Fastest computation, least accurate

Lecture Overview

- Illumination
- Light Sources
  - Directional Lights
  - Point Lights
  - Spot Lights

Spotlights

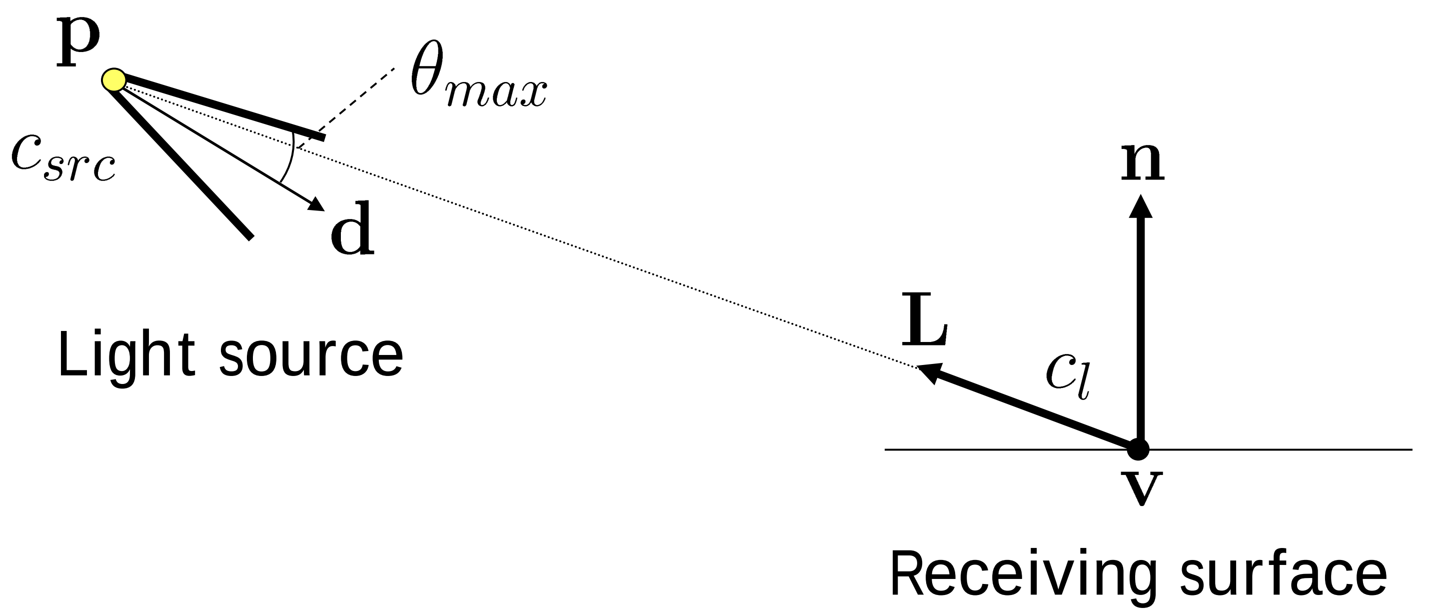
- Like point light, but intensity depends on direction

Parameters

- Position: location of light source
- Spot direction: center axis of light source
- Intensity falloff:
  - Beam width (cone angle)
  - The way the light tapers off at the edges of the beam (cosine exponent)

UCSD

Spotlights



$$\mathbf{L} = \frac{\mathbf{p} - \mathbf{v}}{\|\mathbf{p} - \mathbf{v}\|}$$

$$c_l = \begin{cases} 0 & \text{if} \quad -\mathbf{L} \cdot \mathbf{d} \leq \cos(\theta_{max}) \\ c_{src}\,(-\mathbf{L} \cdot \mathbf{d})^f & \text{otherwise} \end{cases}$$