Light and Optics

Slides by Sarah Jobalia and Kevin Li, with modifications by Kate Baker

Housekeeping

- HW2 released! Due Thursday during grading sessions.
- See course website for live grading missing policy.
- Slight change in live grading times (2:00pm-7:30pm)
  - Updated Zoom and queue links
- Project Proposal assignment released!
  - Due July 23

Revisiting Questions

- Check Ed for two lecture 4 clarifications!
- Human eyes can have lens flares!
    - Microscopic imperfections in lens and cornea can scatter light as it passes through the eye

Revisiting Questions

- Check Ed for two lecture 4 clarifications!

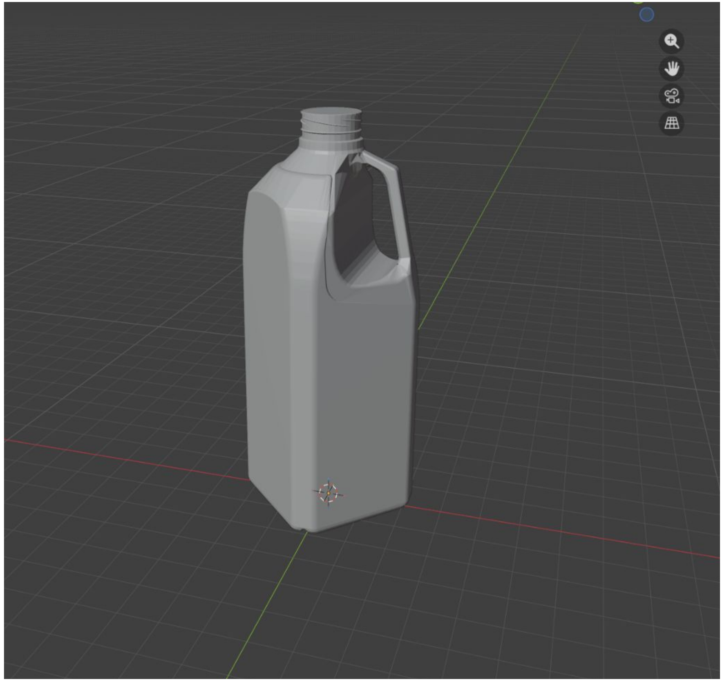
- Human eyes can have lens flares!
  - Microscopic imperfections in lens and cornea can scatter light as it passes through the eye

- Long exposure time for pinhole
  - Tiny aperture
  - Film/sensor needs an amount of light energy

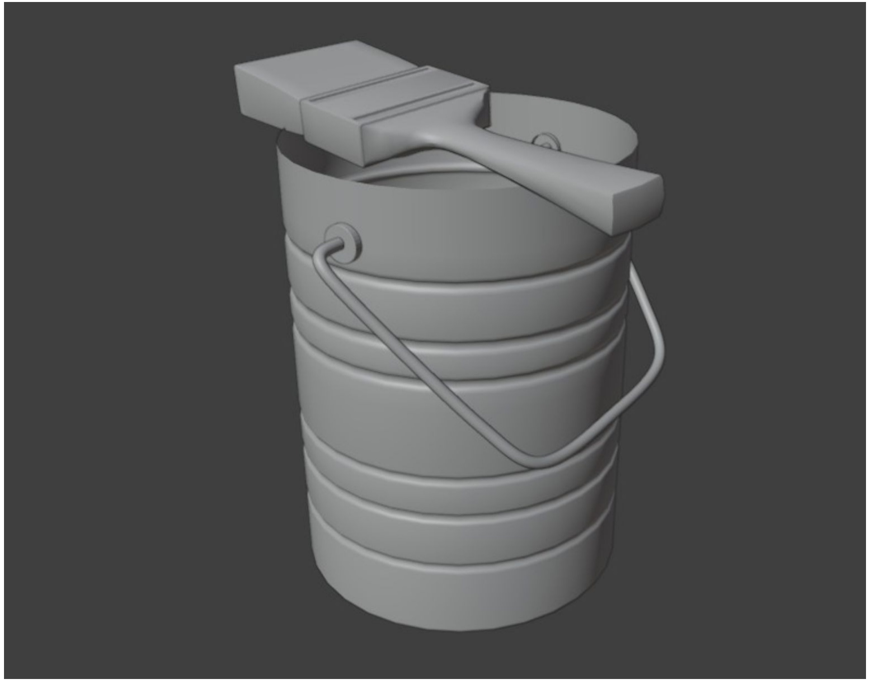
Something Cool!

- The “Backrooms” movie used Blender!
- Check out my Ed post for more detail
- Pre visualizing sets before building them

Student Unique Geometry Highlights!



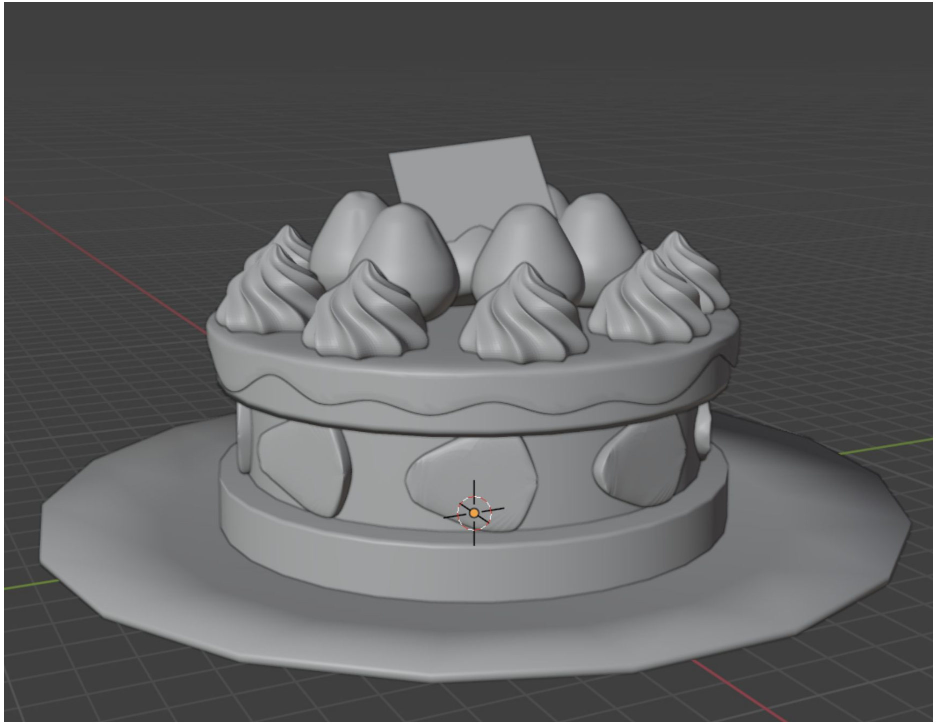
Ryan Griarte



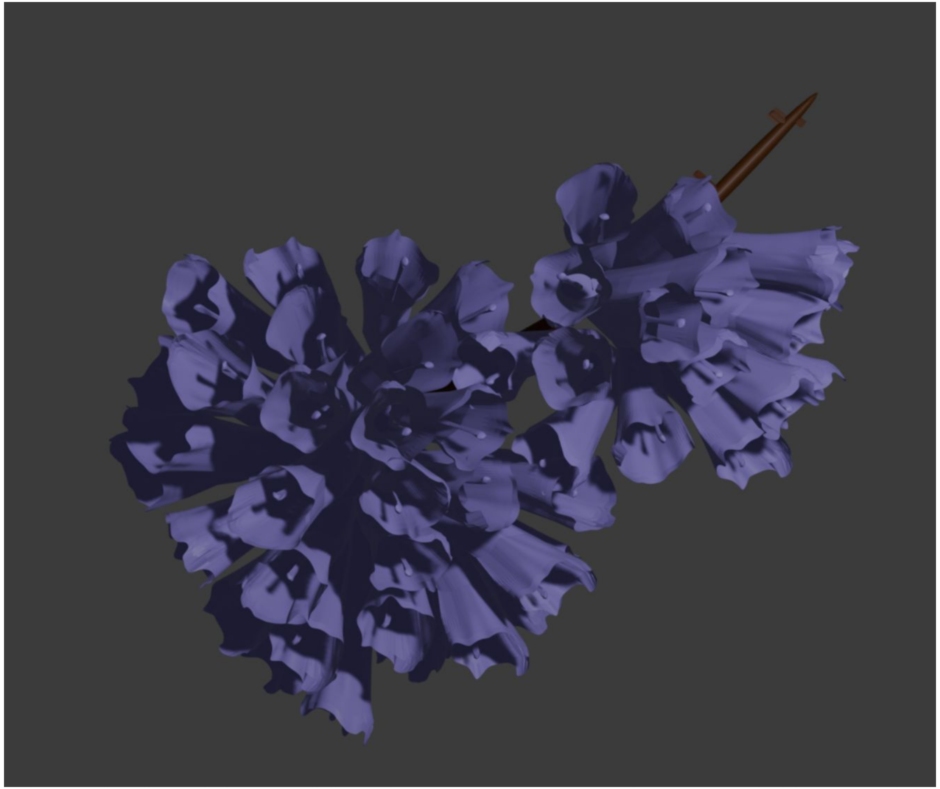
Ji Ryoo

Student Unique Geometry Highlights!

Joshua Starks



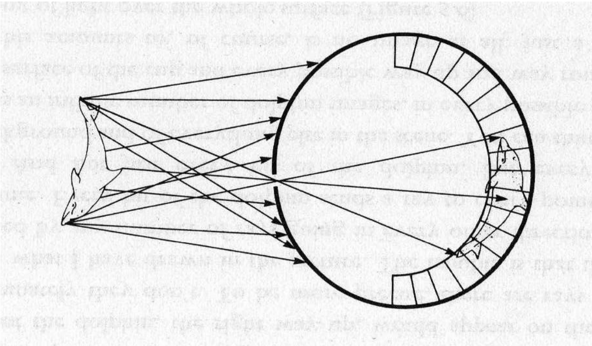
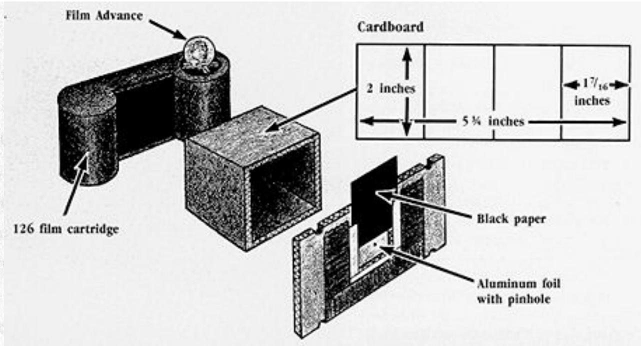
Student Unique Geometry Highlights!



Andrea Matteo

Last time…

- Many, many topics!
- The human eye, pinhole cameras, coordinate frames, projection math…
- Rasterization pipeline done!

Soon… Ray Tracing!

- But first: we need to understand how we model light

Soon… Ray Tracing!

- But first: we need to understand how we model light

Rasterization (Scanline Rendering)

Ray Tracing

Lecture Outline

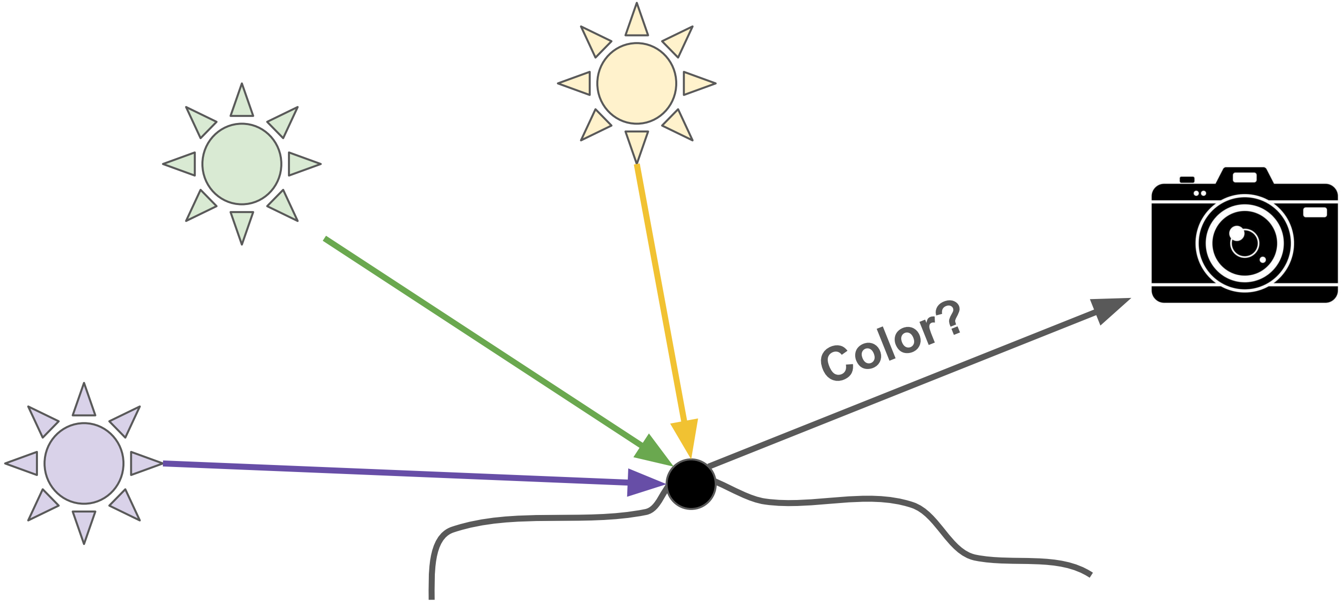
- Outgoing light
  - Radiant intensity, irradiance, radiance
- Incoming light
  - Color bleeding
- Modeling materials
  - BRDF, BTDF, BSSRDF
- The lighting equation

Lecture Outline

- Outgoing light
  - Radiant intensity, irradiance, radiance
- Incoming light
  - Color bleeding
- Modeling materials
  - BRDF, BTDF, BSSRDF
- The lighting equation

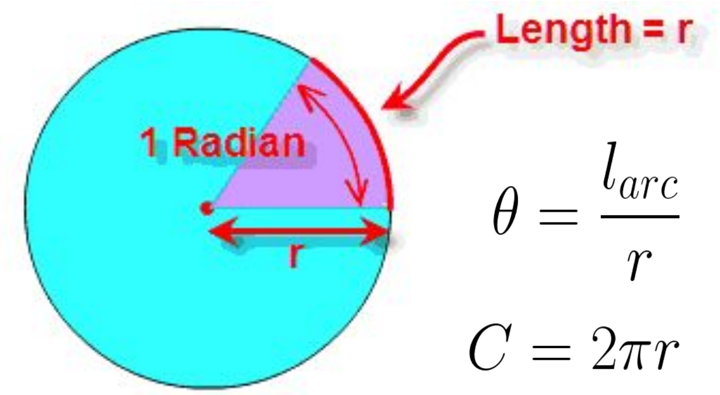
Lighting Geometry

Goal:



Lighting Geometry

- 2D: radian

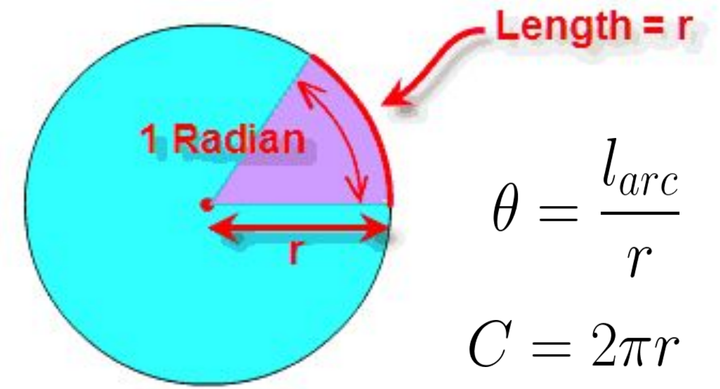


$$\theta = \frac{l_{arc}}{r}$$

$$C = 2\pi r$$

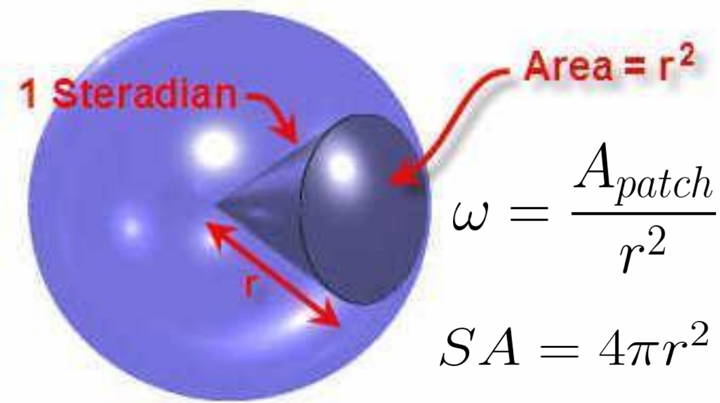
Lighting Geometry

- 2D: radian
- 3D: **steradian**
    - A cone-like volume (called a **solid angle**)
    - Defined by a point and surface area patch
- 2π radians in a circle, 4π steradians in a sphere



$$\theta = \frac{l_{arc}}{r}$$

$$C = 2\pi r$$



$$\omega = \frac{A_{patch}}{r^2}$$

$$SA = 4\pi r^2$$

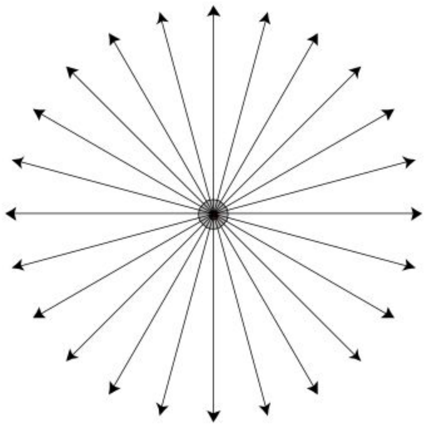
Derivative review

- Every time you see something like: *dX/dY*
- Read it as “how much of X is packed into a tiny sliver of Y, per unit of Y
- You need derivatives to describe how concentrated the X is at a particular spot of Y

Radiant Intensity for a Light Source

- **Radiant Intensity (I)** = power of a light source per solid angle

Where power is measured in watts (energy in joules per second)



Radiant Intensity for a Light Source

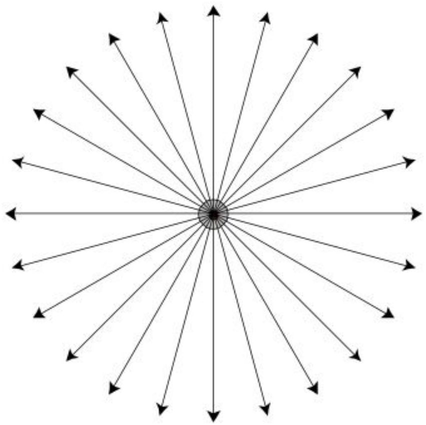
- **Radiant Intensity (I)** = power of a light source per solid angle:

$$I(\omega) = \frac{dP}{d\omega}$$

← **how much is radiating**

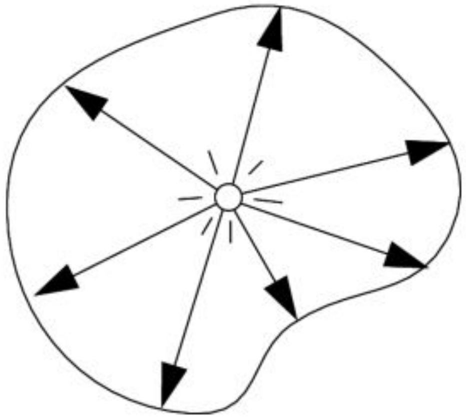
← **radius we care about**

Where power is measured in watts (energy in joules per second)



Radiant Intensity for a Light Source

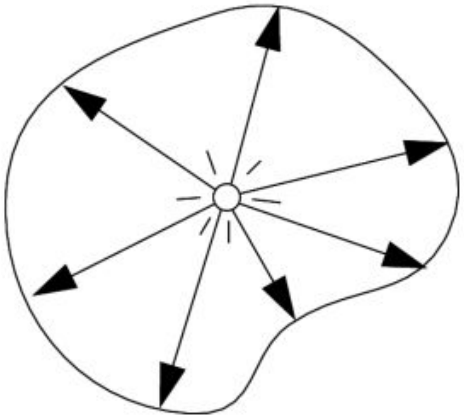
- **Anisotropic** light sources
  - Radiant intensity varies across the light and is, therefore, a function of steradians



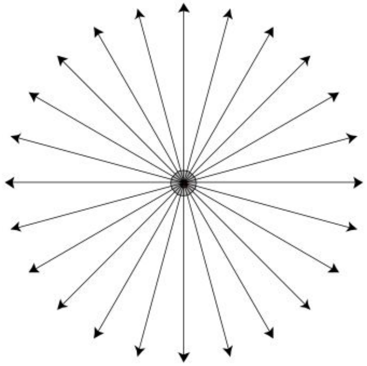
anisotropic light

Radiant Intensity for a Light Source

- **Anisotropic** light sources
  - Radiant intensity varies across the light and is, therefore, a function of steradians
- **Isotropic** point lights
  - Reminder: $$I(\omega) = \frac{dP}{d\omega}$$



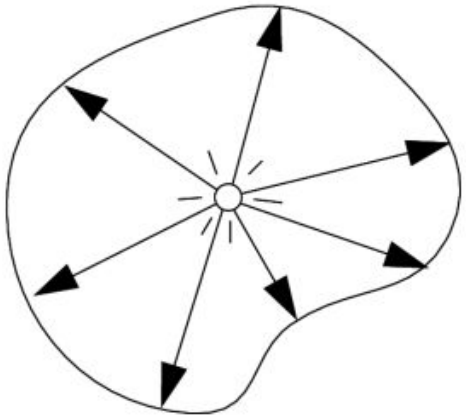
anisotropic light



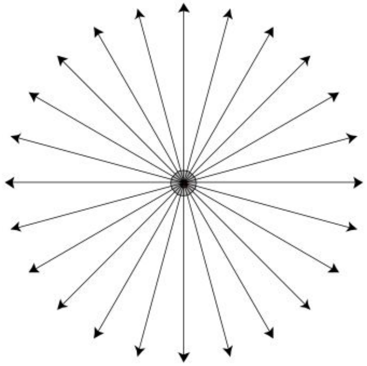
Isotropic point light

Radiant Intensity for a Light Source

- **Anisotropic** light sources
  - Radiant intensity varies across the light and is, therefore, a function of steradians
- **Isotropic** point lights
  - Reminder: $$I(\omega) = \frac{dP}{d\omega}$$
  - If we integrate:
    $$dP = I d\omega$$
    we get:
    $$P = \int_{sphere} I d\omega = 4\pi I$$



anisotropic light

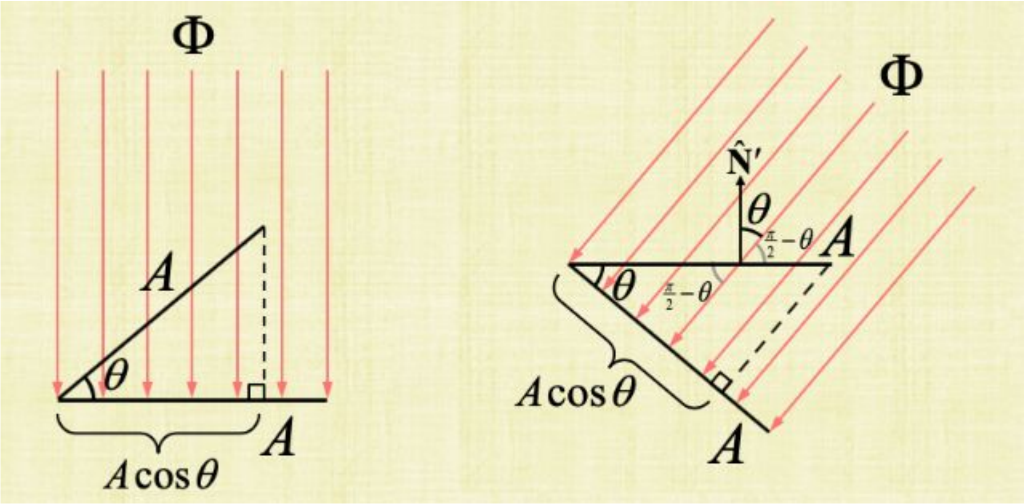


Isotropic point light

Irradiance on a Surface

- **Irradiance (E)** = power per unit surface area of the object that the light is hitting:

$$E = \frac{dP}{dA}$$

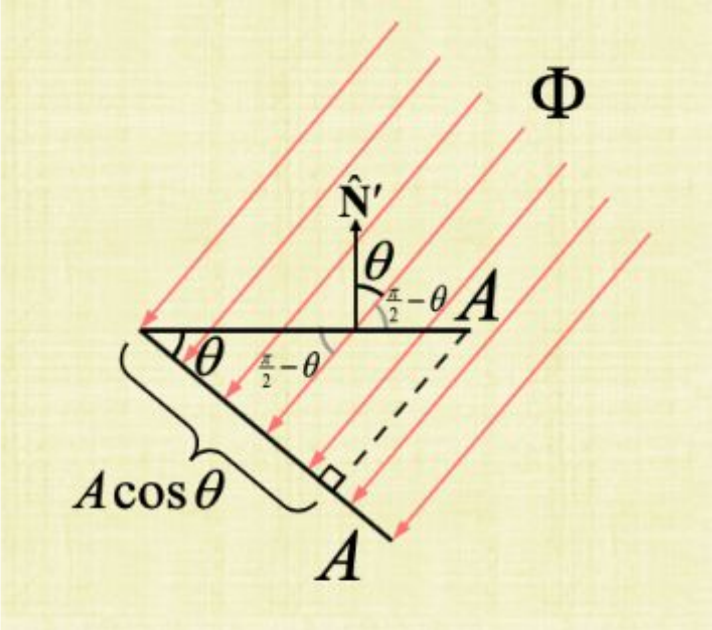


Irradiance on a Surface

- Irradiance decreases as you tilt the surface

$$E = \frac{dP}{dA} \rightarrow E_{tilted} = \frac{\frac{A\cos\theta}{A}P}{A} = E\cos\theta$$

- We saw this cosine appear when talking about diffuse lighting

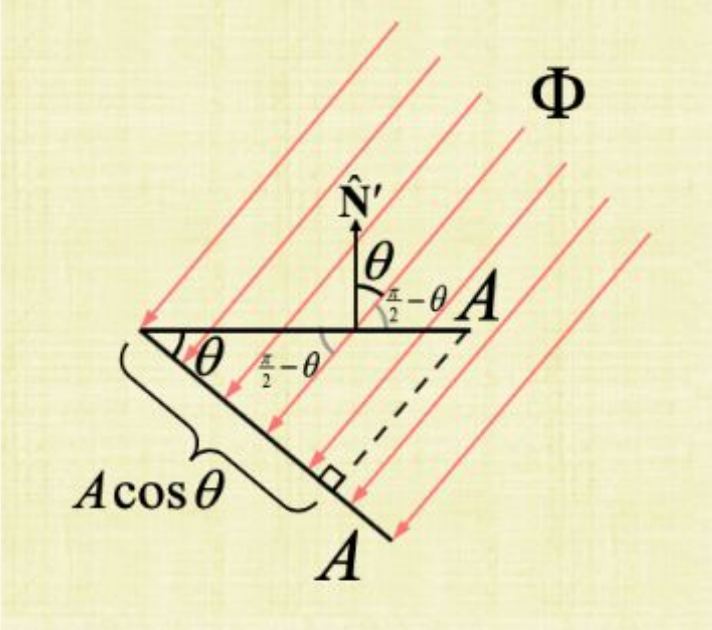


Irradiance on a Surface

- Irradiance decreases as you tilt the surface

$$E = \frac{dP}{dA} \rightarrow E_{tilted} = \frac{\frac{A\cos\theta}{A}P}{A} = E\cos\theta$$

- We saw this cosine appear when talking about diffuse lighting
- What happens to E_{tilted} if θ is zero?

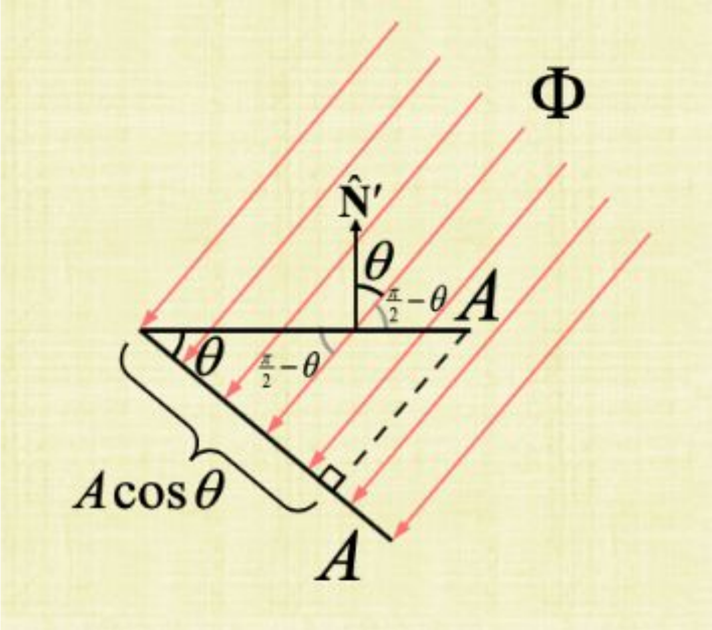


Irradiance on a Surface

- Irradiance decreases as you tilt the surface

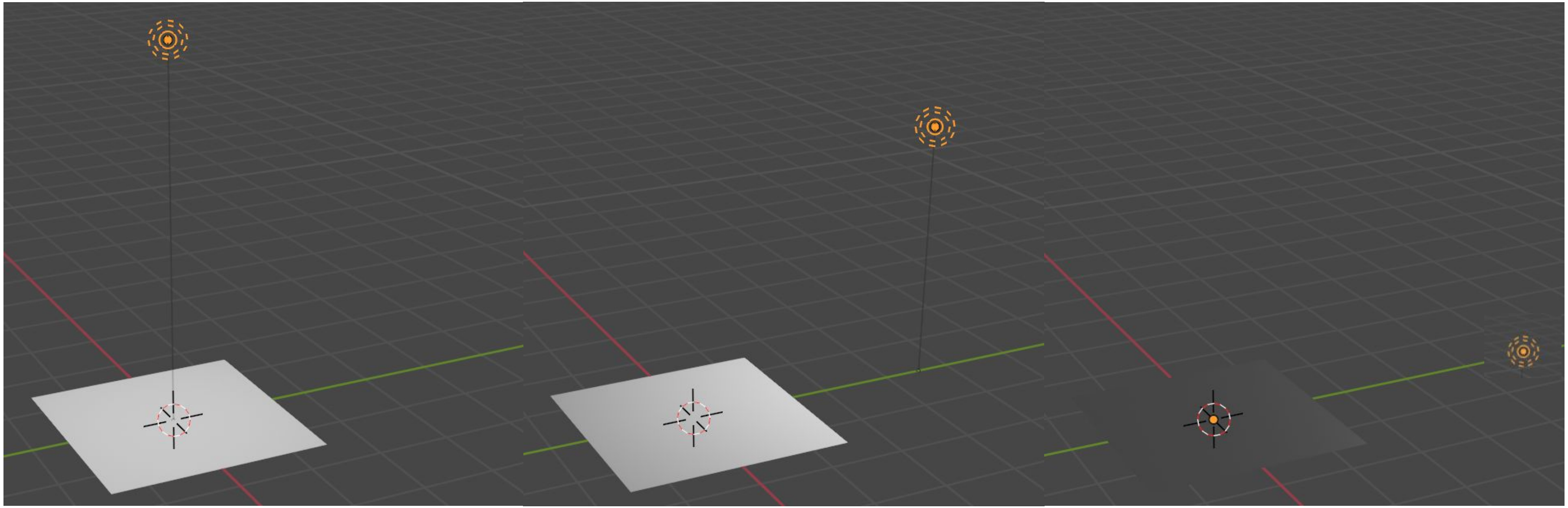
$$E = \frac{dP}{dA} \rightarrow E_{tilted} = \frac{\frac{A\cos\theta}{A}P}{A} = E\cos\theta$$

- We saw this cosine appear when talking about diffuse lighting
- What happens to E_{tilted} if θ is zero?
    - Maximum irradiance!



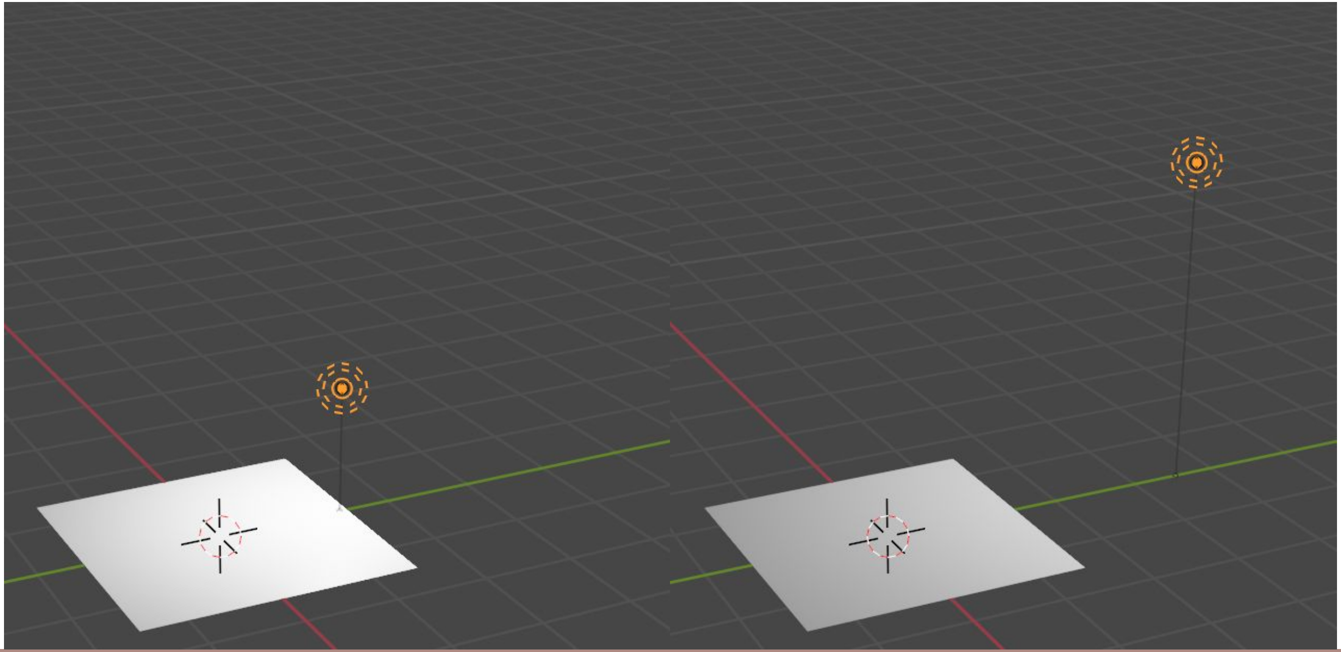
Irradiance on a Surface

- Irradiance varies based on the **tilt angle of the surface** with light



Irradiance on a Surface

- Irradiance varies based on **distance from light**



Summary so far

- **Radiant intensity**
  - Power (coming from a light) per unit surface area
  - Measures the strength of a (point) light source
- **Irradiance**
  - Light power (hitting a surface) per unit surface area
  - Measures how much light is hitting a surface
  - Varies based on tilt angle of surface with light
  - Varies based on distance from light

Radiance: Area Lights

- In the real world, light is emitted **per unit area** (not from a point)
  - E.g. computer screen (rectangle area), light bulb (surface patch)

Radiance: Area Lights

- In the real world, light is emitted **per unit area** (not from a point)
  - E.g. computer screen (rectangle area), light bulb (surface patch)

- Approximate area light by breaking it into small area chunks
- Each area chunk emits light in a solid angle direction

Radiance: Area Lights

- In the real world, light is emitted **per unit area** (not from a point)
  - E.g. computer screen (rectangle area), light bulb (surface patch)

- Approximate area light by breaking it into small area chunks
- Each area chunk emits light in a solid angle direction
- So… define **radiance (L)** as radiant intensity per area chunk:

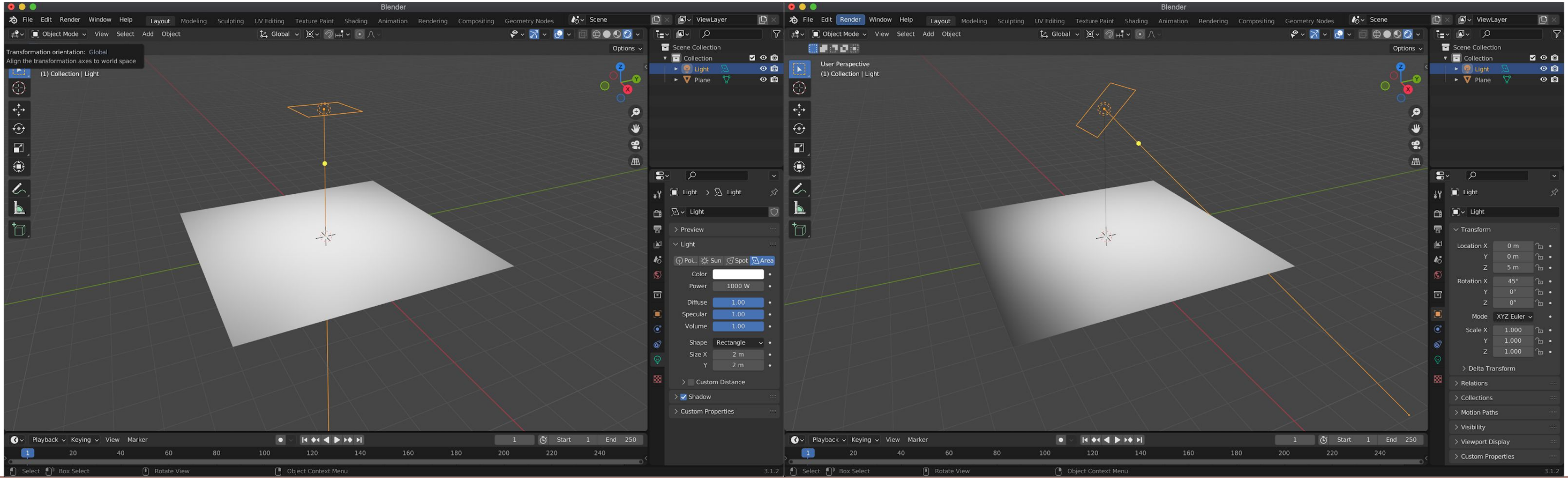
$$L = \frac{dI}{dA \cos\theta}$$

A = area chunk of the light
θ = tilt angle of the light
I = radiant intensity (power)

Radiance: Area Lights

- **Radiance** is radiant intensity per area chunk:

$$L = \frac{dI}{dA\cos\theta} = \left(\frac{d(dP)}{d\omega dA\cos\theta} = \frac{dE}{d\omega\cos\theta} \right)$$



Lecture Outline

- Outgoing light
  - Radiant intensity, irradiance, radiance
- Incoming light
  - Color bleeding
- Modeling materials
  - BRDF, BTDF, BSSRDF
- The lighting equation

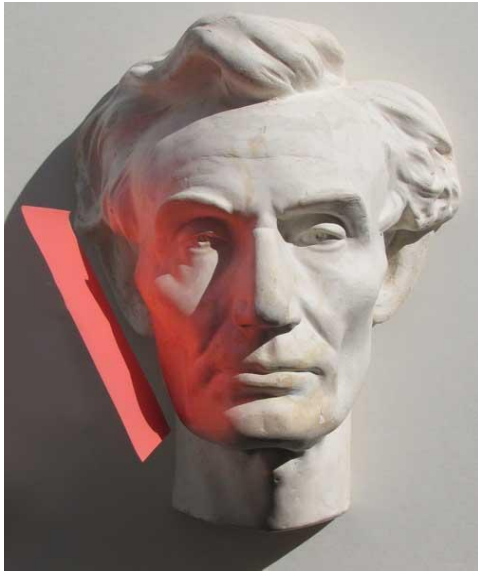
Incoming Light: Color Bleeding

- In the real world, light doesn’t come from just light sources
- Light comes from **all visible objects** in the world
- Each area chunk of each object acts as a source of light

- Ex: the tree opposite the car acts
  as a light shining its radiance on the car

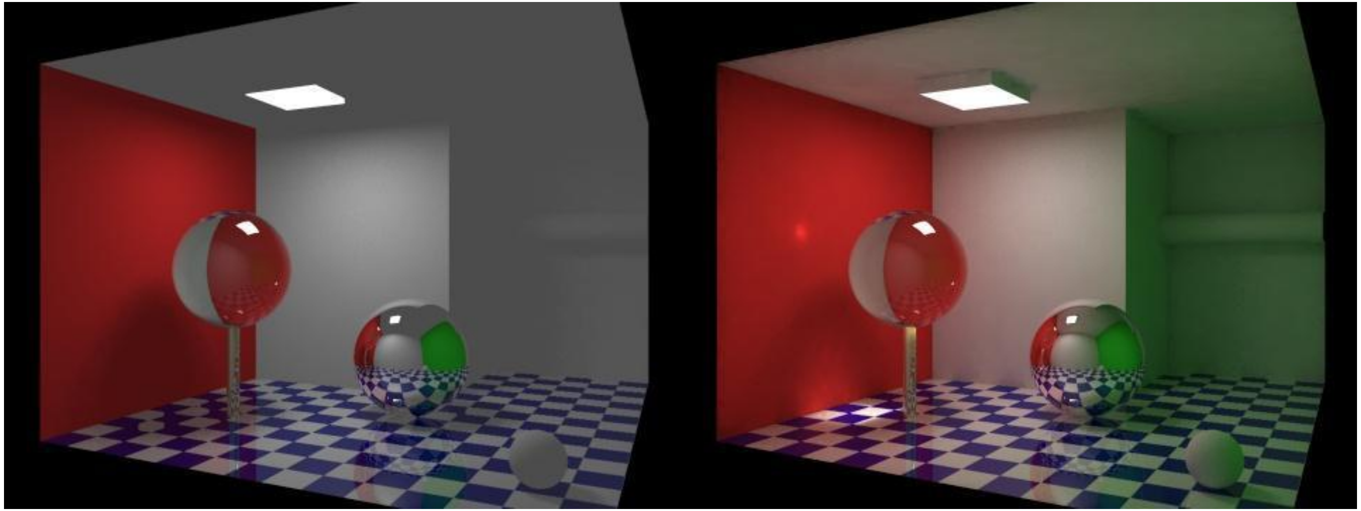
Incoming Light: Color Bleeding

- In the real world, light doesn't come from just light sources
- Light comes from **all visible objects** in the world
- Each area chunk of each object acts as a source of light



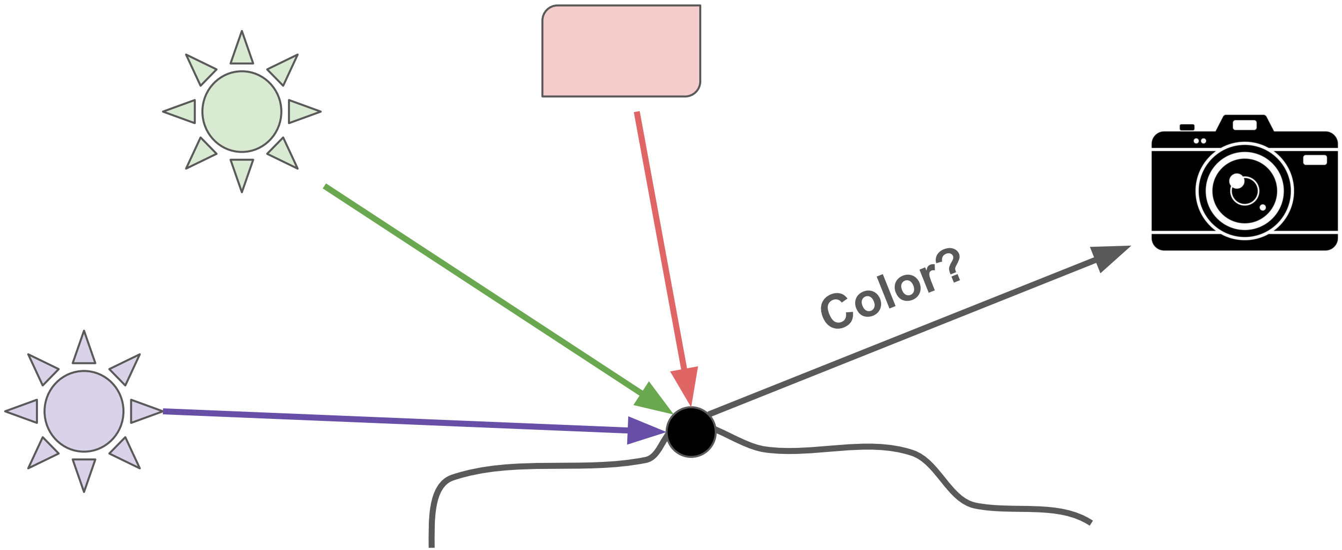
Incoming Light: Color Bleeding

- In the real world, light doesn't come from just light sources
- Light comes from **all visible objects** in the world
- Each area chunk of each object acts as a source of light



using only light from the light source   using incoming light from all directions

Lighting Geometry



Measuring Incoming Light

- **Light probe** = a small reflective chrome sphere
- Photograph it to record the intensity of incoming light

Measuring Incoming Light

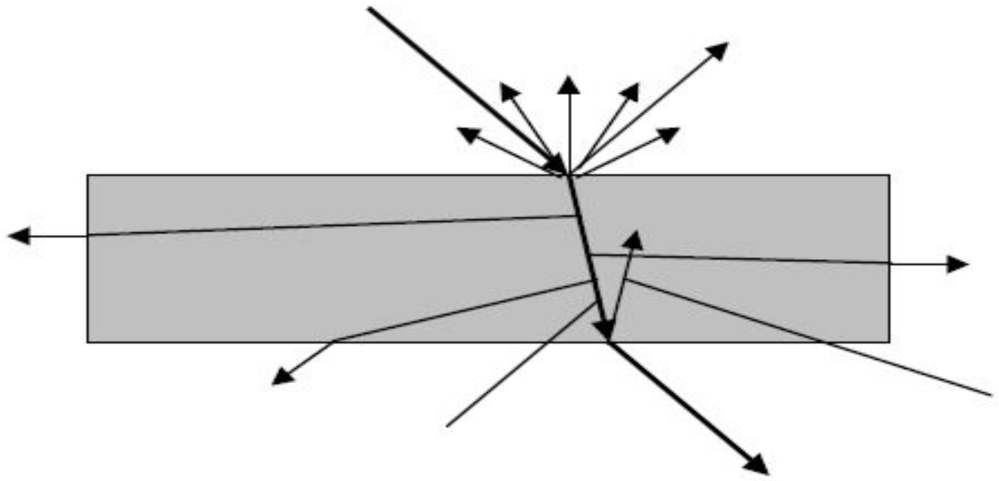
- **Light probe** = a small reflective chrome sphere
- Measuring incoming light lets us model a synthetic object in CGI and render it as though it belonged in the original scene

Lecture Outline

- Outgoing light
  - Radiant intensity, irradiance, radiance
- Incoming light
  - Color bleeding
- Modeling materials
  - BRDF, BTDF, BSSRDF
- The lighting equation

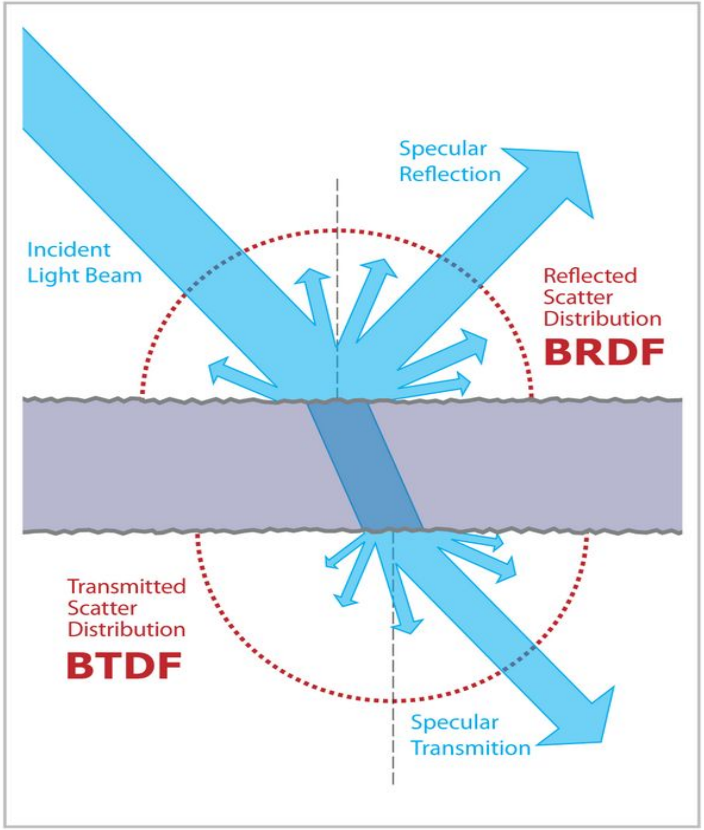
Light/Object Interaction

- Depending on the material of the object, when light hits it, it may be: **absorbed, reflected,** or **transmitted**
- We typically refer to the light as **absorbed** or **scattered**



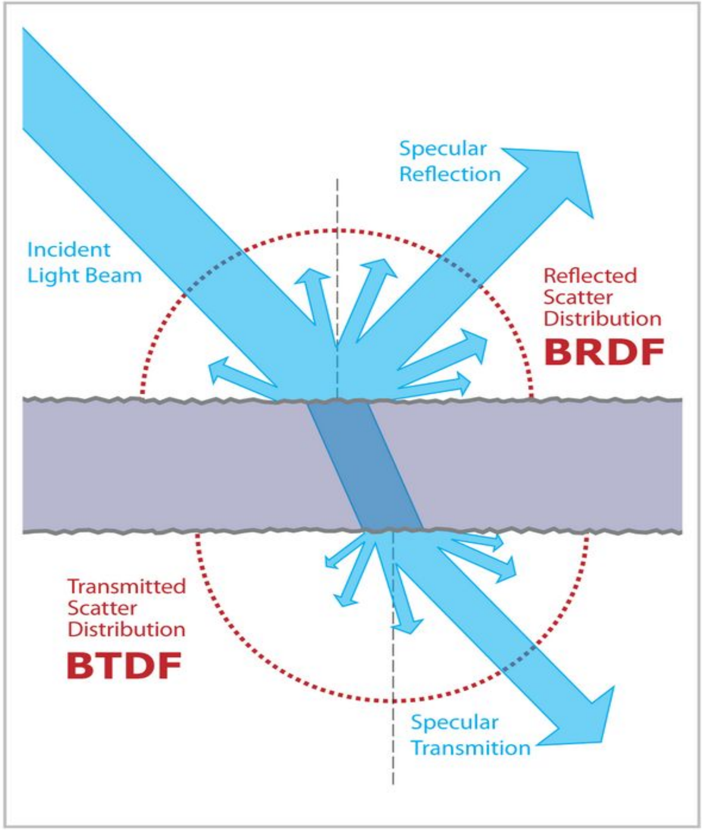
Approximations for Modeling Materials

- **BRDF**
  - Bidirectional **Reflectance** Distribution Function



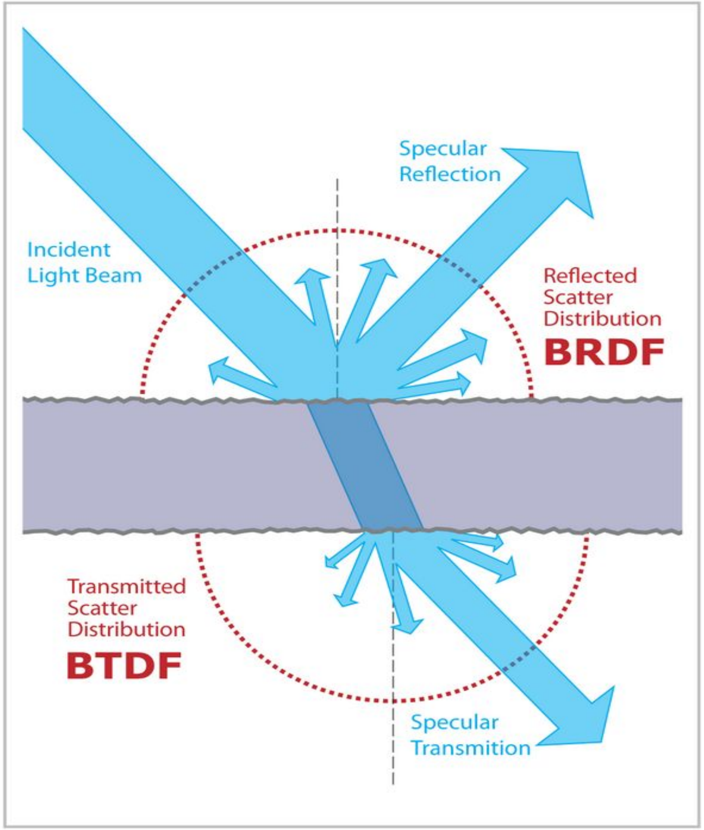
Approximations for Modeling Materials

- **BRDF**
  - Bidirectional **Reflectance** Distribution Function
- **BTDF**
  - Bidirectional **Transmittance** Distribution Function

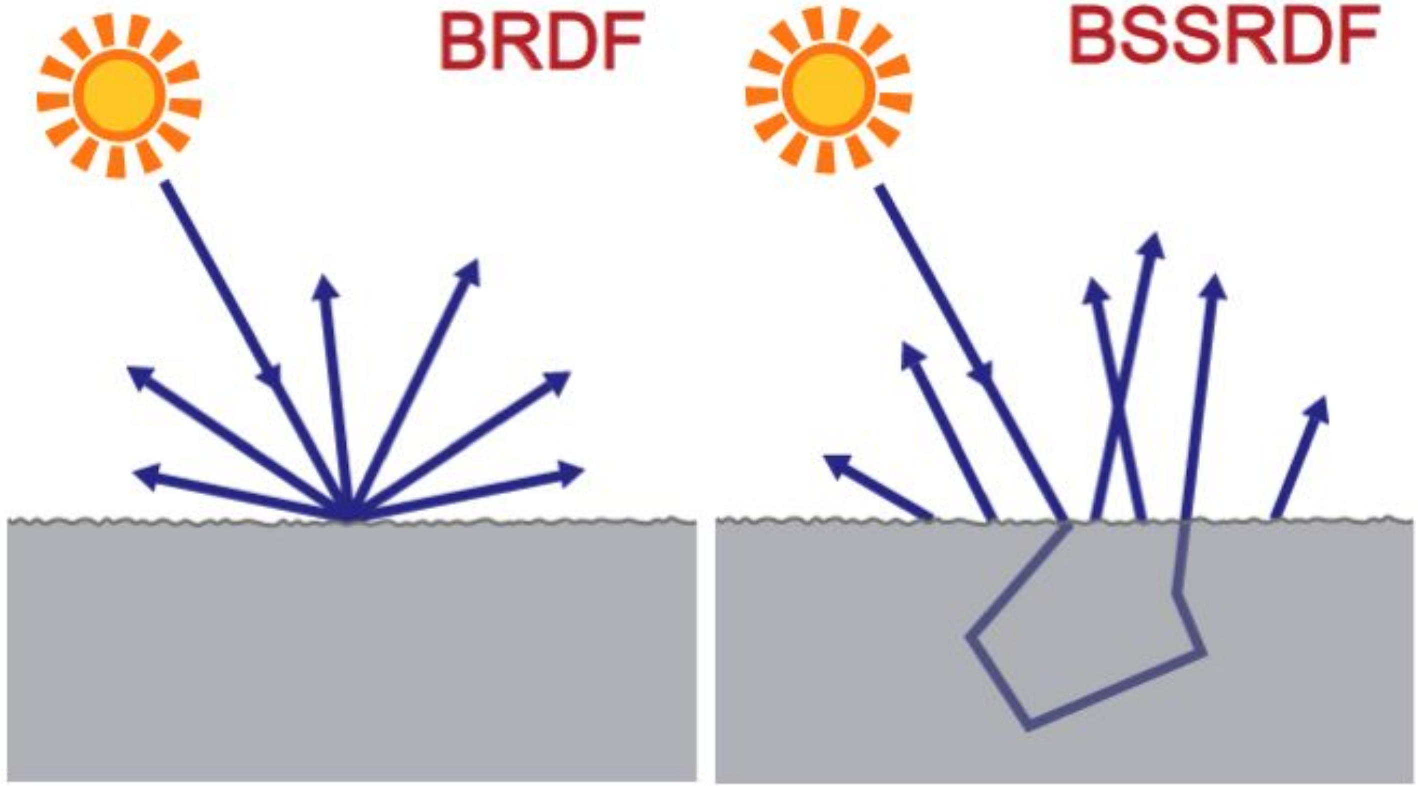


Approximations for Modeling Materials

- **BRDF**
  - Bidirectional **Reflectance** Distribution Function
- **BTDF**
  - Bidirectional **Transmittance** Distribution Function
- **BSSRDF**
  - Bidirectional Surface Scattering Reflectance Distribution Function
  - Combines reflection and transmission models

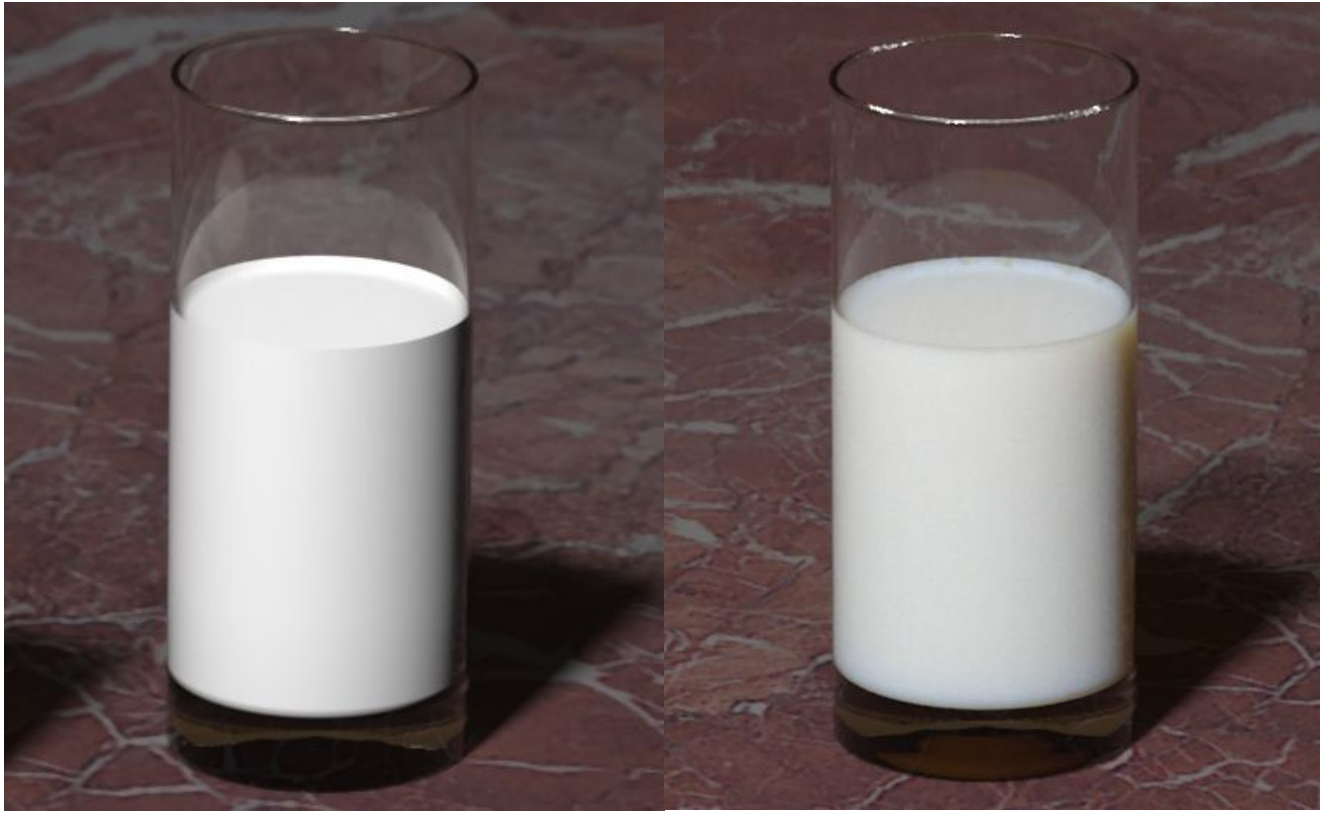


Opaque (BRDF) vs. Translucent (BSSRDF)



Opaque (BRDF) vs. Translucent (BSSRDF)

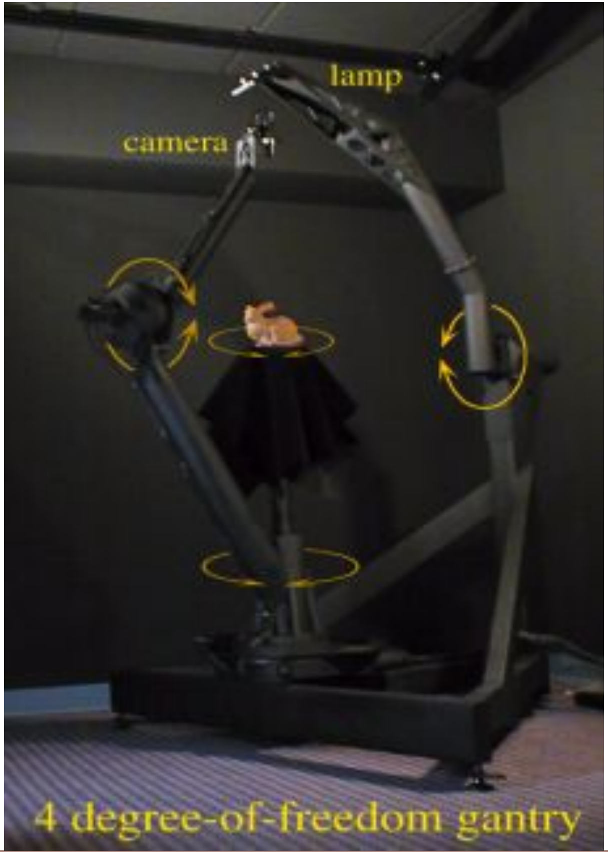
Opaque (BRDF) vs. Translucent (BSSRDF)



Opaque (BRDF) vs. Translucent (BSSRDF)

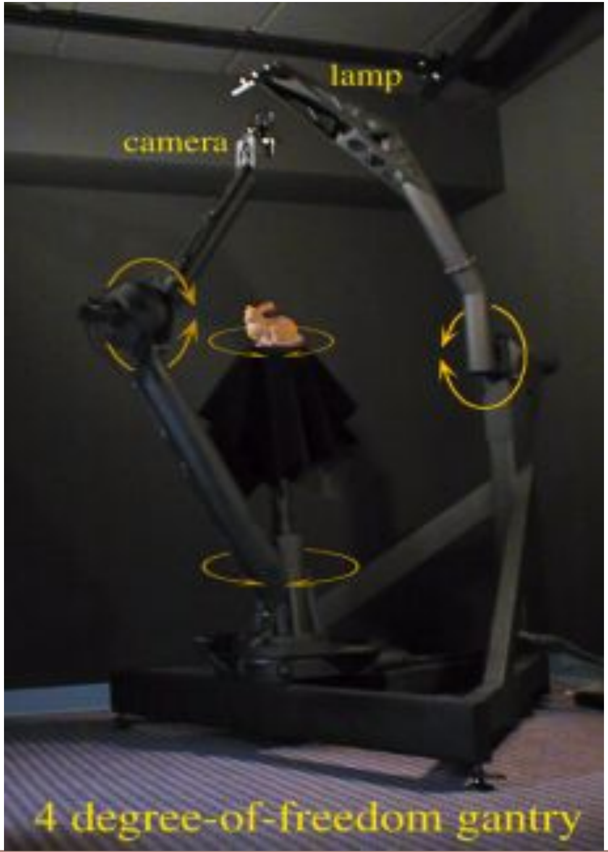
Measuring & Approximating Material Models

- Can measure material data for an object using a gonioreflectometer



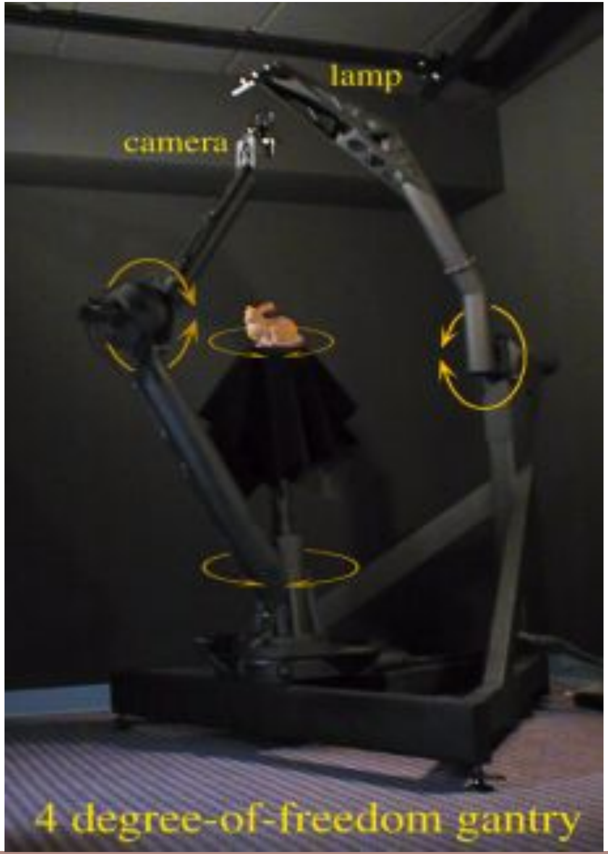
Measuring & Approximating Material Models

- Can measure material data for an object using a gonioreflectometer
- OR (simple) analytical models (BRDFs)
  - Blinn-Phong:
    - Simplest, most general (plastic)
    - Recall Phong Reflection Model!



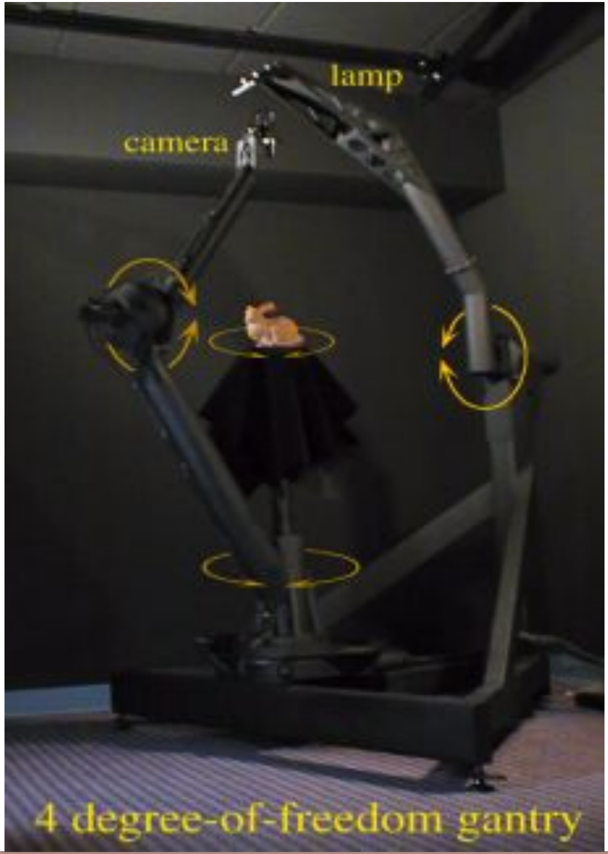
Measuring & Approximating Material Models

- Can measure material data for an object using a gonioreflectometer
- OR (simple) analytical models (BRDFs)
  - Blinn-Phong:
    - Simplest, most general (plastic)
    - Recall Phong Reflection Model!
  - Cook-Torrance:
    - Better specular (metal)



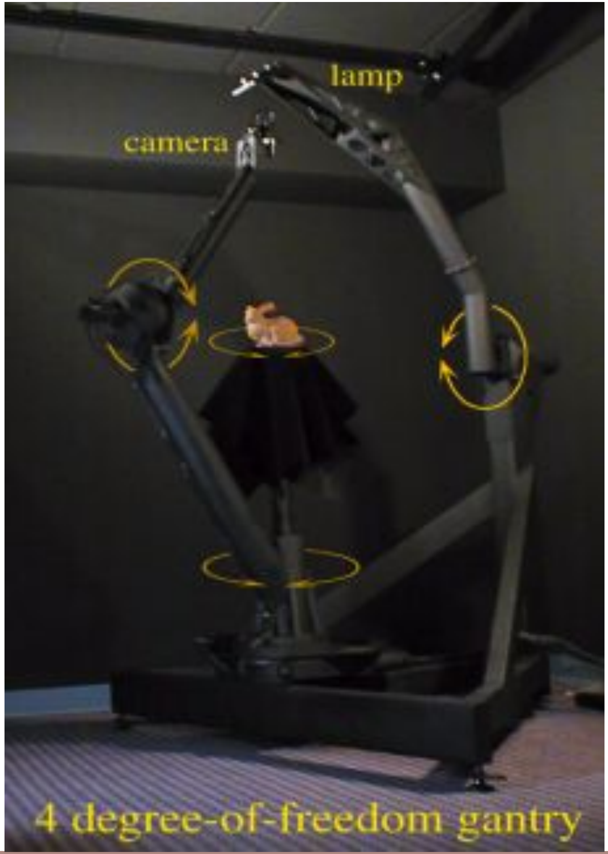
Measuring & Approximating Material Models

- Can measure material data for an object using a gonioreflectometer
- OR (simple) analytical models (BRDFs)
  - Blinn-Phong:
    - Simplest, most general (plastic)
    - Recall Phong Reflection Model!
  - Cook-Torrance:
    - Better specular (metal)
  - Ward:
    - Anisotropic (brushed metal, hair)



Measuring & Approximating Material Models

- Can measure material data for an object using a gonioreflectometer
- OR (simple) analytical models (BRDFs)
  - Blinn-Phong:
    - Simplest, most general (plastic)
    - Recall Phong Reflection Model!
  - Cook-Torrance:
    - Better specular (metal)
  - Ward:
    - Anisotropic (brushed metal, hair)
  - Oren-Nayar:
    - Non-matte (concrete, plaster)



Measuring & Approximating Material Models

- Activity: Which objects/surfaces share the similar material appearances?
  - E.g. water and glass

BRDF

- BRDF $(\lambda, \omega_i, \omega_o, u, v)$
  - λ (wavelength) ⇒ use RGB for 3 BRDFs, 1 for each color component

BRDF

- BRDF $(\lambda, \omega_i, \omega_o, u, v)$
  - λ (wavelength) ⇒ use RGB for 3 BRDFs, 1 for each color component
  - $\omega_i(\theta_i, \phi_i), \omega_o(\theta_o, \phi_o)$ ⇒ the incoming/outgoing light directions,
    Solid angles parameterized using spherical coordinates

BRDF

- BRDF $(\lambda, \omega_i, \omega_o, u, v)$
  - λ (wavelength) ⇒ use RGB for 3 BRDFs, 1 for each color component
  - $\omega_i(\theta_i, \phi_i), \omega_o(\theta_o, \phi_o)$ ⇒ the incoming/outgoing light directions,

    Solid angles parameterized using spherical coordinates
  - (u, v) ⇒ coordinates on the object surface

BRDF

- BRDF $(\lambda, \omega_i, \omega_o, u, v)$
  - λ (wavelength) ⇒ use RGB for 3 BRDFs, 1 for each color component
  - $\omega_i(\theta_i, \phi_i), \omega_o(\theta_o, \phi_o)$ ⇒ the incoming/outgoing light directions,

    Solid angles parameterized using spherical coordinates
  - (u, v) ⇒ coordinates on the object surface
- Therefore, we have 3 functions of 4 variables, 1 for each color component:

$$BRDF_R(\omega_i, \omega_o), BRDF_G(\omega_i, \omega_o), BRDF_B(\omega_i, \omega_o)$$

$$BRDF_R(\theta_i, \phi_i, \theta_o, \phi_o), BRDF_G(\theta_i, \phi_i, \theta_o, \phi_o), BRDF_B(\theta_i, \phi_i, \theta_o, \phi_o)$$

BRDF

- The BRDF also relates the **incoming light that hits a surface patch** (irradiance) to the **outgoing light emitted from the surface patch** (radiance):

BRDF

- The BRDF also relates the **incoming light that hits a surface patch** (irradiance) to the **outgoing light emitted from the surface patch** (radiance):

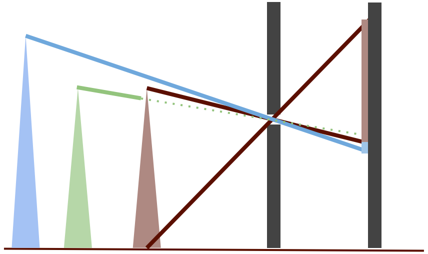
$$BRDF(\omega_i, \omega_o) = \frac{dL_o(\omega_o)}{dE_i(\omega_i)} \begin{array}{l} \leftarrow \textbf{radiance (outgoing light)} \\ \leftarrow \textbf{irradiance (incoming light)} \end{array}$$

Lecture Outline

- Outgoing light
  - Radiant intensity, irradiance, radiance
- Incoming light
  - Color bleeding
- Modeling materials
  - BRDF, BTDF, BSSRDF
- The lighting equation

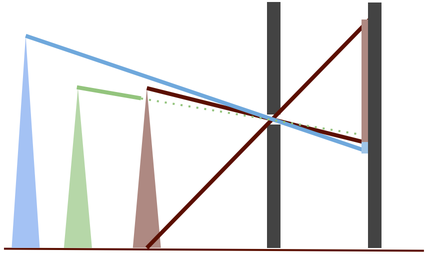
Lighting Equation

- Putting it all together!
- Given a point on an object, we have:
  - light from incoming directions ω_i hitting the point
  - for each incoming direction ω_i, light reflects outwards with ω_o



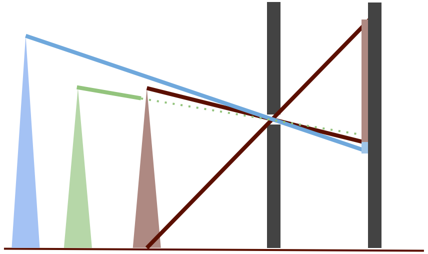
Lighting Equation

- Putting it all together!
- Given a point on an object, we have:
  - light from incoming directions ω_i hitting the point
  - for each incoming direction ω_i, light reflects outwards with ω_o
  - BRDF models how much light gets reflected in each outward direction ω_o given an incoming direction ω_i



Lighting Equation

- Putting it all together!
- Given a point on an object, we have:
  - light from incoming directions ω_i hitting the point
  - for each incoming direction ω_i, light reflects outwards with ω_o
  - BRDF models how much light gets reflected in each outward direction ω_o given an incoming direction ω_i
- We render from the perspective of the camera
- Need to determine which of the outgoing light rays reaches the pixels on our camera film plane



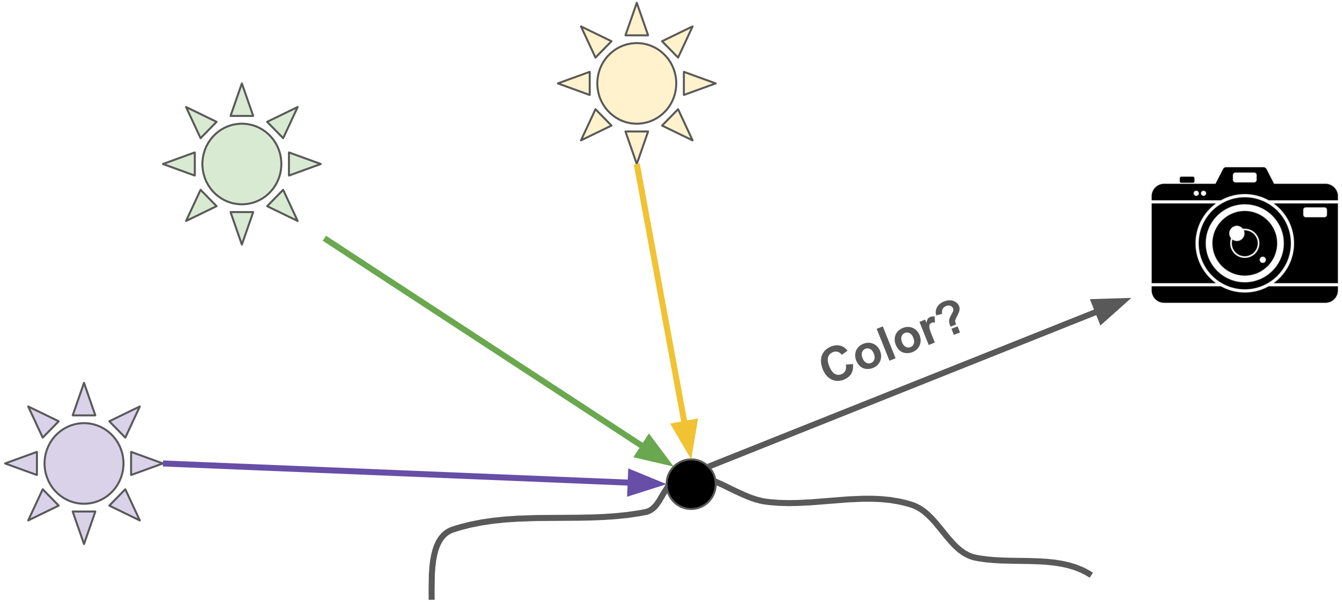
Lighting Equation

- $L_o(\omega_o) = \sum_{i \in in} L_o(\omega_i, \omega_o)$
  - For each incoming ray, compute how much it contributes to the outgoing direction
  - Then sum
- $L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) dE_i(\omega_i)$

Lighting Equation

- $L_o(\omega_o) = \sum_{i \in in} L_o(\omega_i, \omega_o)$
  - For each incoming ray, compute how much it contributes to the outgoing direction
  - Then sum
- $L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) dE_i(\omega_i)$
  - Similar idea, but more precise
  - Light can come from continuous directions (integrate)

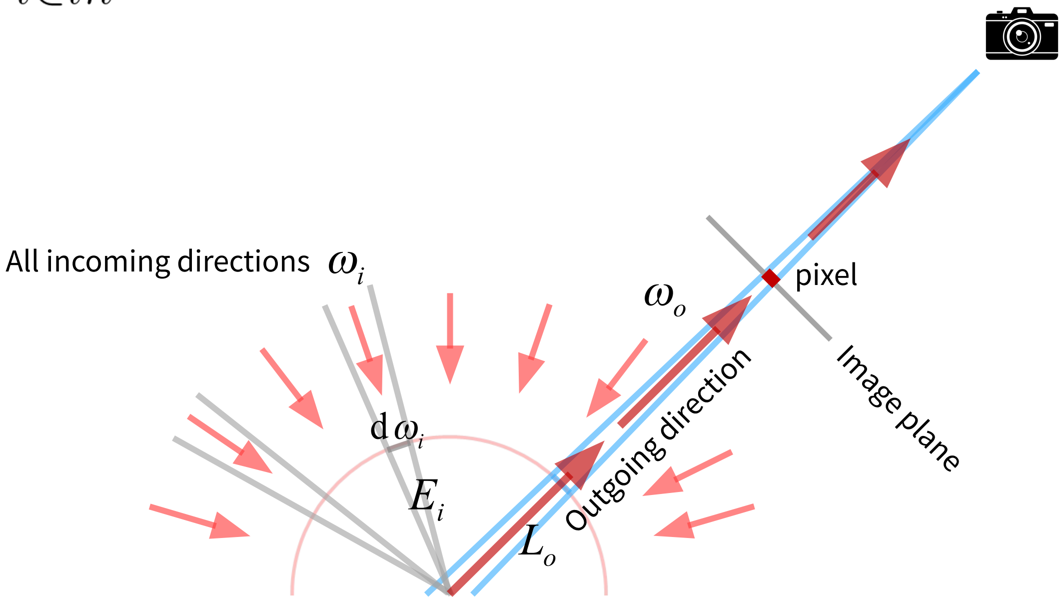
Lighting Geometry



Lighting Equation

$$L_o(\omega_o) = \sum_{i \in in} L_o(\omega_i, \omega_o)$$

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) dE_i(\omega_i)$$



Lighting Equation

- Our equation relates incoming irradiance to outgoing radiance

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) dE_i(\omega_i)$$

Lighting Equation

- Our equation relates incoming irradiance to outgoing radiance

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) dE_i(\omega_i)$$

- But recall:

$$L = \frac{dI}{dA \cos\theta} = \left(\frac{d(dP)}{d\omega dA \cos\theta} = \frac{dE}{d\omega \cos\theta} \right)$$

Lighting Equation

- Our equation relates incoming irradiance to outgoing radiance

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) dE_i(\omega_i)$$

- But recall:

$$L = \frac{dI}{dA \cos\theta} = \left(\frac{d(dP)}{d\omega dA \cos\theta} = \frac{dE}{d\omega \cos\theta} \right)$$

- So we can write: $dE = L d\omega \cos\theta$

Lighting Equation

- Our equation relates incoming irradiance to outgoing radiance

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) dE_i(\omega_i)$$

- But recall:

$$L = \frac{dI}{dA \cos\theta} = \left(\frac{d(dP)}{d\omega dA \cos\theta} = \frac{dE}{d\omega \cos\theta} \right)$$

- So we can write: $dE = L d\omega \cos\theta$
- Thus, to simplify the equation, we write the **lighting equation** as:

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) L_i \cos\theta_i d\omega_i$$

Lighting Equation: Final Version

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) L_i \cos\theta_i d\omega_i$$

- The **BRDF** is a function that **models the material of our object**
  - E.g. Phong model

Lighting Equation: Final Version

$$\boxed{L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) L_i \cos\theta_i d\omega_i}$$

- The **BRDF** is a function that **models the material of our object**
  - E.g. Phong model
- The result of the BRDF is **scaled by the incoming radiance**
  - Example: If our light is red, then even if the BRDF computes e.g. blue, we would still multiply the blue by a factor of red

Lighting Equation: Final Version

$$\boxed{L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) L_i \cos\theta_i d\omega_i}$$

- The **BRDF** is a function that **models the material of our object**
  - E.g. Phong model
- The result of the BRDF is **scaled by the incoming radiance**
  - Example: If our light is red, then even if the BRDF computes e.g. blue, we would still multiply the blue by a factor of red
- The incoming radiance is **scaled by a cosine term**
  - Radiance depends on the tilt of the object and the tilt of the (area) light

Lighting Equation: Final Version

$$\boxed{L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) L_i \cos\theta_i d\omega_i}$$

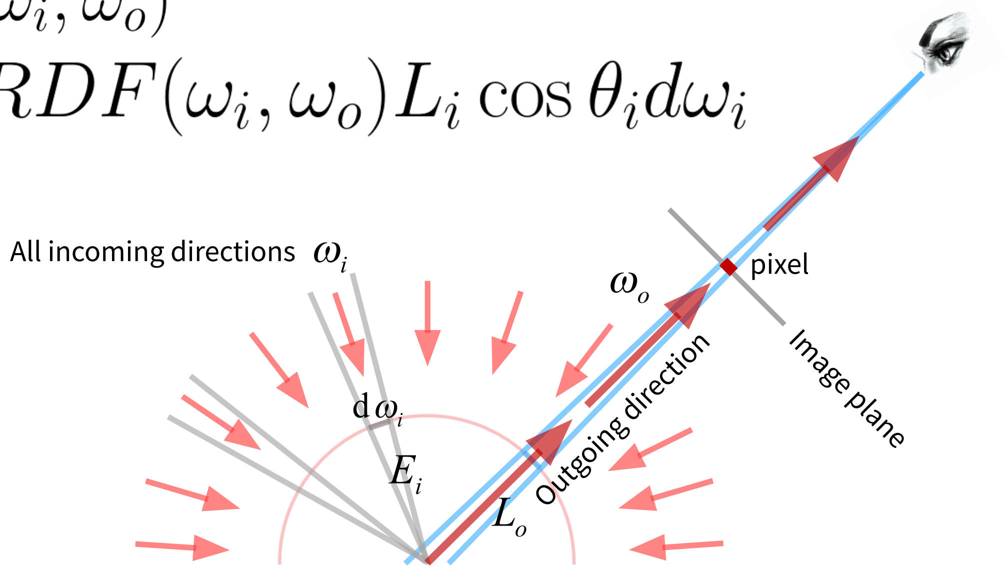
- The **BRDF** is a function that **models the material of our object**
  - E.g. Phong model
- The result of the BRDF is **scaled by the incoming radiance**
  - Example: If our light is red, then even if the BRDF computes e.g. blue, we would still multiply the blue by a factor of red
- The incoming radiance is **scaled by a cosine term**
  - Radiance depends on the tilt of the object and the tilt of the (area) light
- For <u>each</u> (for loop!) **incoming light direction** (omega) on a point on our surface, **compute** the above, **then sum** (integrate) them all up

Next time…

- We use this equation to determine the color of a surface point
- This requires understanding the rays involved!

$$L_o(\omega_o) = \sum_{i \in in} L_o(\omega_i, \omega_o)$$

$$L_o(\omega_o) = \int_{i \in in} BRDF(\omega_i, \omega_o) L_i \cos\theta_i d\omega_i$$



Additional Resources + Exploring!

- VFX Firsts: Chrome Ball
- Gonioreflectometer specifications
- Subsurface Scattering
- Anisotropic Chemistry!
- Ed Post about Blender Backrooms

- Fundamentals of Computer Graphics, Steve Marschner and Peter Shirley
    - Preview, Book