Lighting

Wolfgang Heidrich

Course News

Assignment 1
- Due today!

Assignment 2
- Out later today
- Due Monday, Feb 28
- More later…

Homework 3
- Out today
- Discussed in labs next week

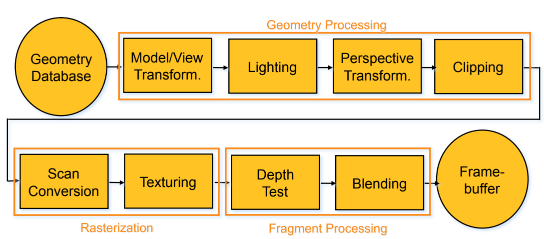
Course News

Quiz 1
- Discussed in labs this week

Reading
- Chapter 9, 3

Out of Town this Friday
- For grant review meeting
- Anika will fill in for me

The Rendering Pipeline



Lighting

Goal
- Model the interaction of light with surfaces to render realistic images

Contributing Factors
- Light sources
  - *Shape and color*
- Surface materials
  - *How surfaces reflect light*
- Transport of light
  - *How light moves in a scene (global illumination, later in the course)*

Light Sources and Materials

Appearance depends on
- Light sources, locations, properties
- Material (surface) properties
- Viewer position

Local illumination
- Compute at material, from light to viewer

Global illumination (later in course)
- Ray tracing: from viewer into scene
- Radiosity: between surface patches

Materials

Analyzing surface reflectance

- Illuminate surface point with a ray of light from different directions
- Observe how much light is reflected in all possible directions

Materials

Light is linear

- If two rays illuminate the surface point the result is just the sum of the individual reflections for each ray
- For N directions the reflection is the sum of the individual N reflections
- For light arriving from a *continuum* of directions, the reflection is the *integral* over the reflections caused by the individual directions
  - *More on this when we talk about global illumination at the end of the term*

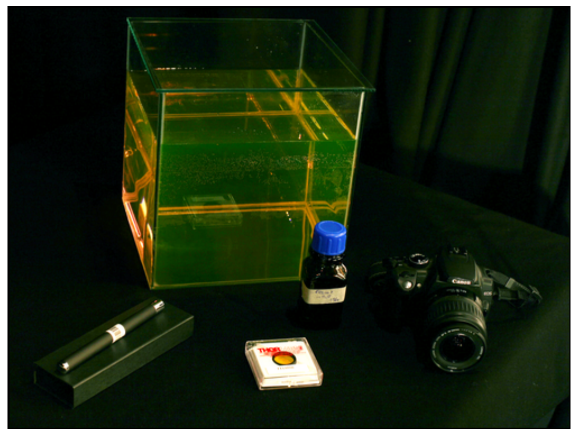
Experiment

Goal:

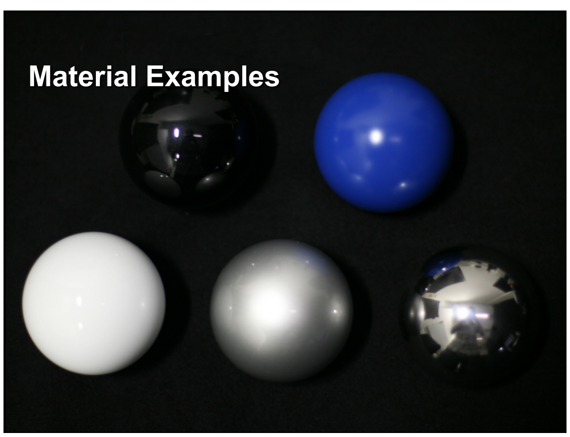
- Visualize reflected light distribution for a given illuminating ray

Physical setup:

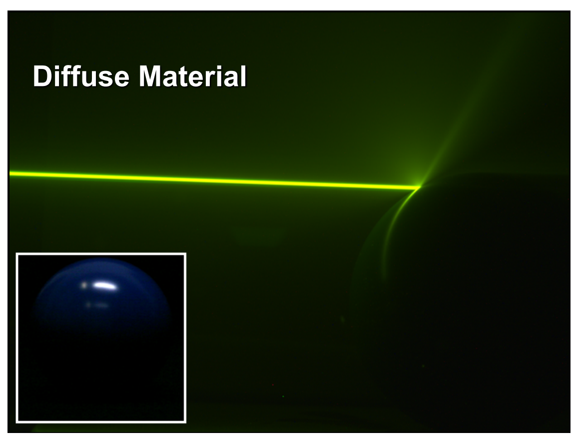
- Laser illumination
- Water tank with fluorescent dye
  - *Makes laser light visible as it travels through "empty" space*

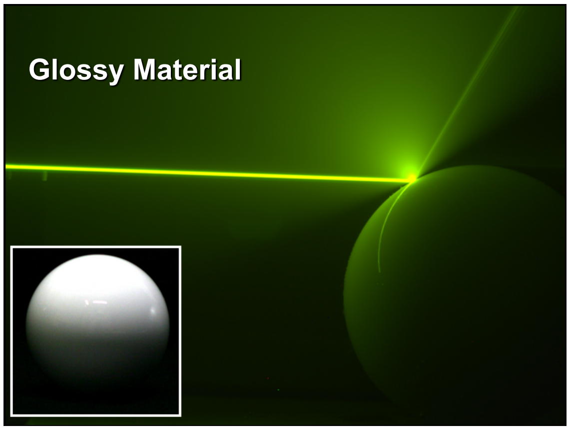


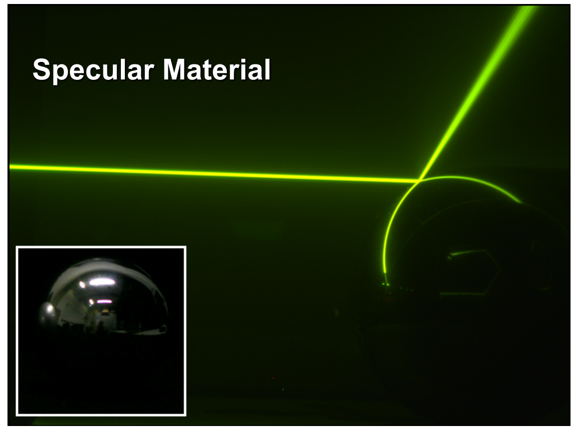
Material Examples

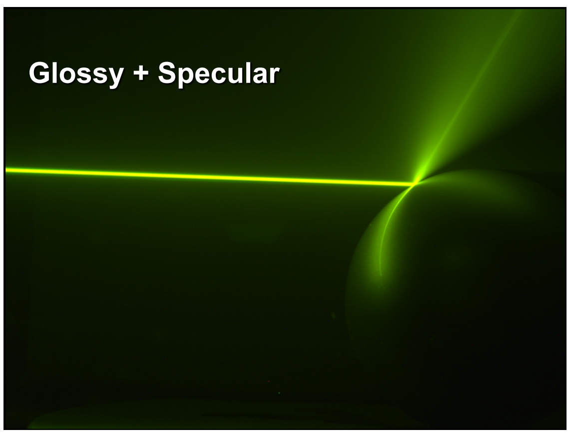


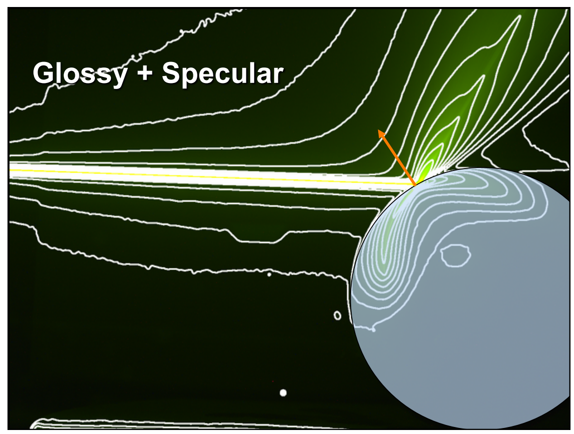
Diffuse Material











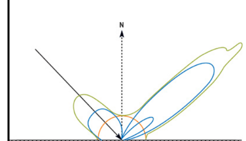
BRDF

Model for all these effects:

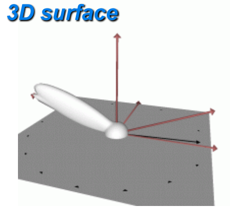
- Bi-directional
  - *i.e. dependent on 2 directions: incident, exitant*
- Reflectance
  - *A model for surface reflection (not transmission)*
- Distribution
  - *Light is distributed over different exitant directions*
- Function

BRDF lobe plots

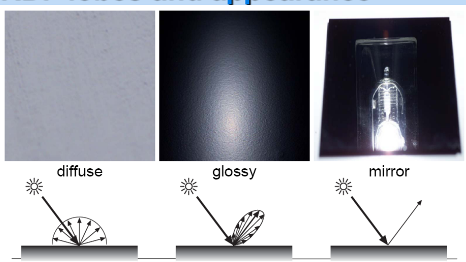
2D slice



3D surface



BRDF lobes and appearance



Limitations of the BRDF Model

BRDFs cannot describe

- Light of one wavelength that gets absorbed and re-emitted at a different wavelength
  - *(fluorescence)*
- Light that gets absorbed and emitted much later
  - *(phosphorescence)*
- Light that penetrates the object surface, scatters in the interior of the object, and exits at a different point form where it entered
  - *(subsurface scattering)*

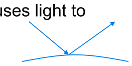
Materials

Practical Considerations

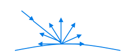
- In practice, we often simplify the BRDF model further
- Derive specific formulas that describe different reflectance behaviors
  - *E.g. diffuse, glossy, specular*
- Computational efficiency is also a concern

Types of Reflection

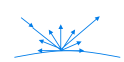
- *Specular* (a.k.a. *mirror* or *regular*) reflection causes light to propagate without scattering.



- *Diffuse* reflection sends light in all directions with equal energy.

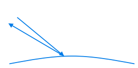


- *Mixed* reflection is a weighted combination of specular and diffuse.

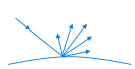


Types of Reflection

- *retro-reflection* occurs when incident energy reflects in directions close to the incident direction, for a wide range of incident directions.



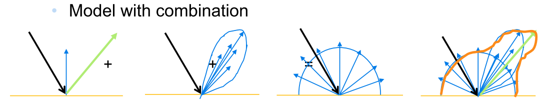
- *gloss* is the property of a material surface that involves mixed reflection and is responsible for the mirror like appearance of rough surfaces.



Reflectance Distribution Model

Most surfaces exhibit complex reflectances

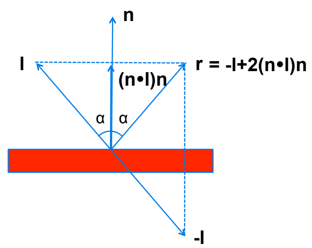
- Vary with incident and reflected directions.
- Model with combination



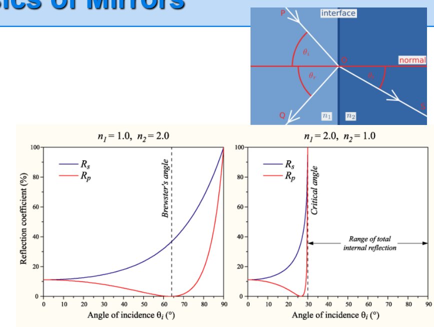
specular + glossy + diffuse = reflectance distribution

Specular Reflection

Geometry of specular (mirror) reflection



Fresnel Reflection
Physics of Mirrors

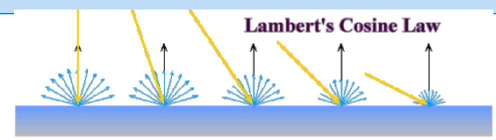


Physics of Diffuse Reflection

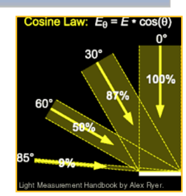
Ideal diffuse reflection

- Very rough surface at the microscopic level
  - *Real-world example: chalk*
- Microscopic variations mean incoming ray of light equally likely to be reflected in any direction over the hemisphere
- Reflected intensity only depends on light direction!

Lambert's "Law"

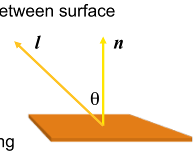


Intuitively: cross-sectional area of the "beam" intersecting an element of surface area is smaller for greater angles with the normal.



Computing Diffuse Reflection

- Depends on angle of incidence: angle between surface normal and incoming light
  - $I_{diffuse} = k_d I_{light} \cos\theta$
- In practice use vector arithmetic
  - $I_{diffuse} = k_d I_{light} (\mathbf{n} \bullet \mathbf{l})$
- Always normalize vectors used in lighting
  - ***n**, **l** should be unit vectors*
- Scalar (B/W intensity) or 3-tuple or 4-tuple (color)
  - k_d: *diffuse coefficient, surface color*
  - I_{light}: *incoming light intensity*
  - $I_{diffuse}$: *outgoing light intensity (for diffuse reflection)*



Diffuse Lighting Examples

- Lambertian sphere from several lighting angles:

- need only consider angles from 0° to 90°

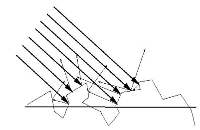
Physics of Glossy Reflection

- At the microscopic level a glossy 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

Glossy 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, not mirror-like?
- One option: model the microgeometry of the surface and explicitly bounce rays off of it

or…

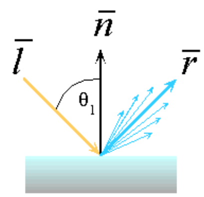


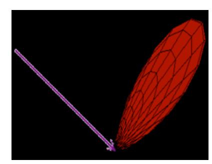
Empirical Approximation

- We expect most reflected light to travel in direction predicted by Snell's Law
- But because of microscopic surface variations, some light may be reflected in a direction slightly off the ideal reflected ray
- As angle from ideal reflected ray increases, we expect less light to be reflected

Empirical Approximation

Angular falloff





how might we model this falloff?

Phong Lighting

Most common lighting model in computer graphics

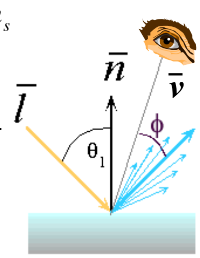
- *(Phong Bui-Tuong, 1975)*

$$\mathbf{I}_{\text{specular}} = \mathbf{k}_s \mathbf{I}_{\text{light}} (\cos\phi)^{n_s}$$

n_s : purely empirical constant, varies rate of falloff

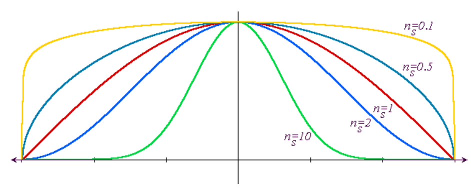
k_s: specular coefficient, highlight color

no physical basis, works ok in practice



Phong Lighting: The n_s Term

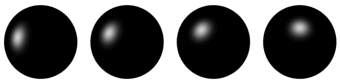
- Phong reflectance term drops off with divergence of viewing angle from ideal reflected ray



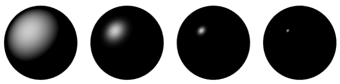
Viewing angle – reflected angle

Phong Examples

varying l



varying n_{shiny}

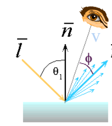


Calculating Phong Lighting

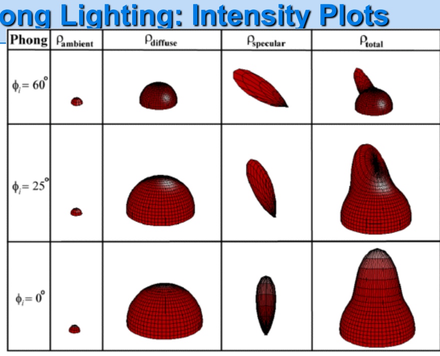
compute cosine term of Phong lighting with vectors

$$\mathbf{I}_{\text{specular}} = \mathbf{k}_s \mathbf{I}_{\text{light}} (\mathbf{v} \bullet \mathbf{r})^{n_s}$$

- v: unit vector towards viewer/eye
- r: ideal reflectance direction (unit vector)
- k_s: specular component
  - *highlight color*
- I_{light}: incoming light intensity



Phong Lighting: Intensity Plots

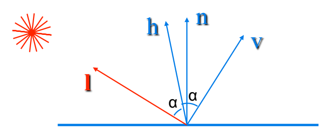


Alternative Model

Blinn-Phong model (Jim Blinn, 1977)

- Variation with better physical interpretation
  - **h**: *halfway vector; r: roughness*

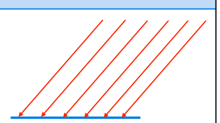
$$I_{out}(\mathbf{x}) = k_s \cdot (\mathbf{h} \cdot \mathbf{n})^{1/r} \cdot I_{in}(\mathbf{x}); \text{ with } \mathbf{h} = (\mathbf{l} + \mathbf{v})/2$$

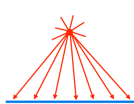


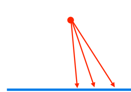
Simple Light Sources

Types of light sources

- Directional/parallel lights
  - *E.g.sun*
  - *Homogeneous vector*
- (Homogeneous) point lights
  - *Same intensity in all directions*
  - *Homogeneous point*
- Spot lights
  - *Limited set of directions*
  - *Point+direction+cutoff angle*



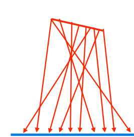




Light Sources

Area lights:

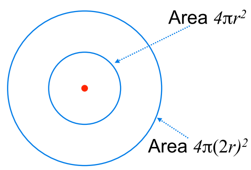
- Light sources with a finite area
- Can be considered a continuum of point lights
- Not available in many rendering systems



Light Source Falloff

Quadratic falloff (point- and spot lights)

- Brightness of objects depends on power per unit area that hits the object
- The power per unit area for a point or spot light decreases quadratically with distance



Wolfgang Heidrich

Light Source Falloff

Non-quadratic falloff

- Many systems allow for other falloffs
- Allows for faking effect of area light sources
- OpenGL / graphics hardware
  - I_o: *intensity of light source*
  - $\mathbf{x}$: *object point*
  - r: *distance of light from* $\mathbf{x}$

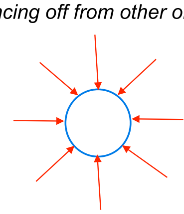
$$I_{in}(\mathbf{x}) = \frac{1}{ar^2 + br + c} \cdot I_0$$

Wolfgang Heidrich

Light Sources

Ambient lights

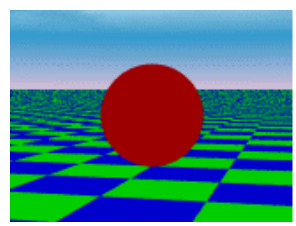
- No identifiable source or direction
- Hack for replacing true global illumination
  - *(light bouncing off from other objects)*



Wolfgang Heidrich

Ambient Light Sources

- Scene lit only with an ambient light source



Light Position Not Important

Viewer Position Not Important

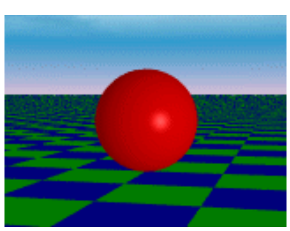
Surface Angle Not Important

Wolfgang Heidrich

Directional Light Sources

- Scene lit with directional and ambient light

Surface Angle Important



Light Position Not Important

Viewer Position Not Important

Wolfgang Heidrich

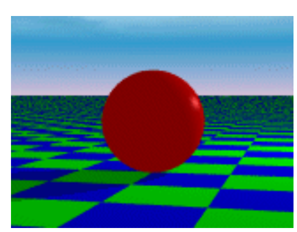
Point Light Sources

- Scene lit with ambient and point light source

Light Position Important

Viewer Position Important

Surface Angle Important



Wolfgang Heidrich

Light Sources & Transformations

Geometry: positions and directions

- Standard: world coordinate system
  - *Effect: lights fixed wrt world geometry*
  - *Demo: http://www.xmission.com/~nate/tutors.html*
- Alternative: camera coordinate system
  - *Effect: lights attached to camera (car headlights)*
- Points and directions undergo normal model/view transformation

Illumination calculations: camera coords

Wolfgang Heidrich

Lighting Review

Lighting models

- Ambient
  - *Normals don't matter*
- Lambert/diffuse
  - *Angle between surface normal and light*
- Phong/specular
  - *Surface normal, light, and viewpoint*

Wolfgang Heidrich

Lighting in OpenGL

Light source: amount of RGB light emitted

- Value represents percentage of full intensity
  E.g., (1.0,0.5,0.5)
- Every light source emits ambient, diffuse, and specular light

Materials: amount of RGB light reflected

- Value represents percentage reflected
  e.g., (0.0,1.0,0.5)

Interaction: multiply components

- Red light (1,0,0) x green surface (0,1,0) = black (0,0,0)

Wolfgang Heidrich

Lighting in OpenGL

```
glLightfv(GL_LIGHT0, GL_AMBIENT, amb_light_rgba );
glLightfv(GL_LIGHT0, GL_DIFFUSE, dif_light_rgba );
glLightfv(GL_LIGHT0, GL_SPECULAR, spec_light_rgba );
glLightfv(GL_LIGHT0, GL_POSITION, position);
glEnable(GL_LIGHT0);

glMaterialfv( GL_FRONT, GL_AMBIENT, ambient_rgba );
glMaterialfv( GL_FRONT, GL_DIFFUSE, diffuse_rgba );
glMaterialfv( GL_FRONT, GL_SPECULAR, specular_rgba );
glMaterialfv( GL_FRONT, GL_SHININESS, n );
```

Wolfgang Heidrich

Coming Up:

Wednesday

- Shading

Friday

- Clipping; scan conversion

Wolfgang Heidrich