Light Reflection and Advanced Shading

CS 4620 Lecture 17

Thursday, November 13th:

4:30pm - 6:00pm - Nathan Loofbourrow's Rigging Workshop in the Rhodes 551
6:30pm - 7:30pm - Marilyn's DWA Company Overview presentation in Rhodes 551

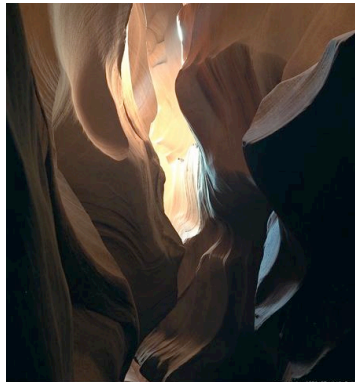
Friday, November 14th:
9:00am - 12:00pm - Student Interviews
12:45pm - 1:45pm - Student Interviews

Visual cues to 3D geometry

- size (perspective)
- occlusion
- shading

Shading

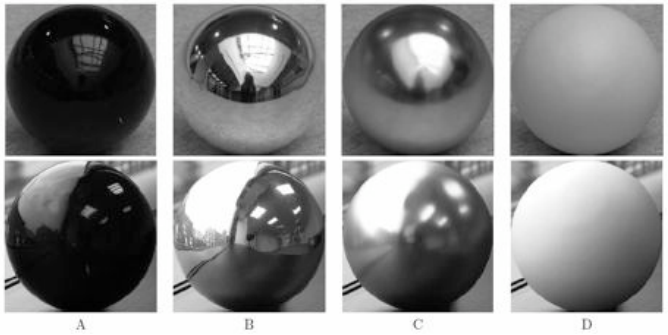
- Variation in observed color across an object
  - strongly affected by lighting
  - present even for homogeneous material
- caused by how a material reflects light
  - depends on
    - geometry
    - lighting
    - material
  - therefore gives cues to all 3



[Philip Greenspun]

Recognizing materials

- Human visual system is quite good at understanding shading



[Dror, Adelson, & Willsky]

© 2006 Steve Marschner •

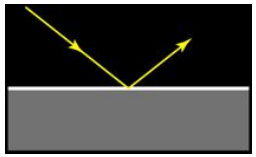
Shading for Computer Graphics

- Need to compute an image
  - of particular geometry
  - under particular illumination
  - from a particular viewpoint
- Basic question: how much light reflects from an object toward the viewer?

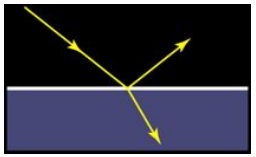
© 2006 Steve Marschner •

Simple materials

metal

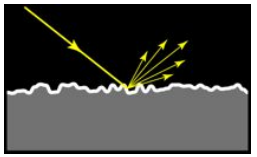


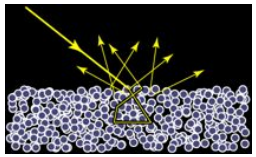
dielectric



© 2006 Steve Marschner •

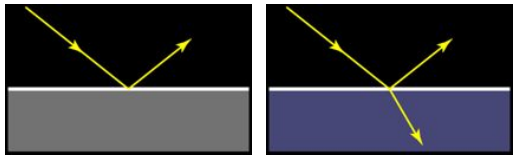
Adding microgeometry



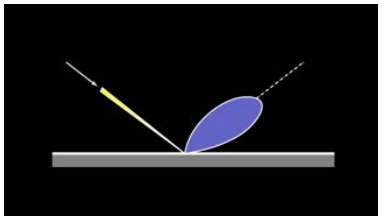


© 2006 Steve Marschner •

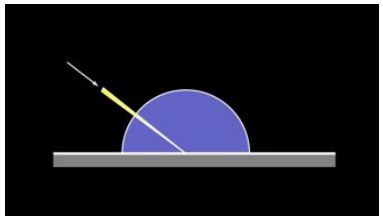
Classic reflection behavior



ideal specular (Fresnel)



rough specular

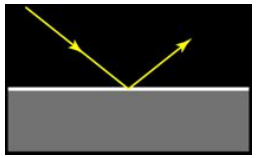


Lambertian

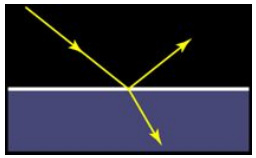
© 2006 Steve Marschner •

Specular reflection

- Smooth surfaces of pure materials have ideal specular reflection (said this before)
  - Metals (conductors) and dielectrics (insulators) behave differently
- Reflectance (fraction of light reflected) depends on angle



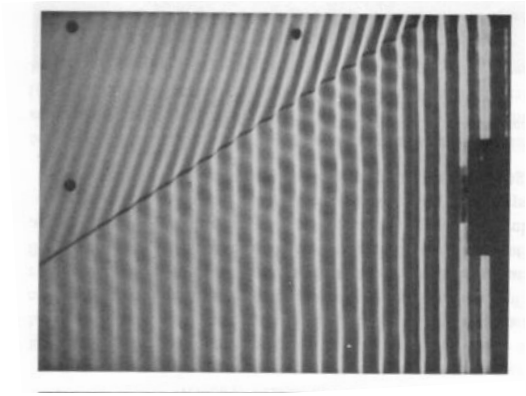
metal



dielectric

© 2008 Steve Marschner • 10

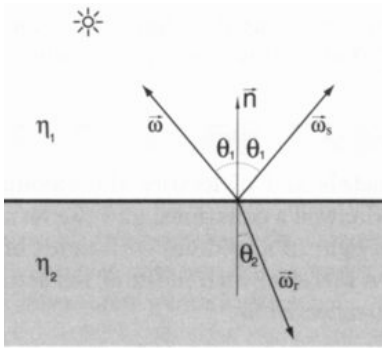
Refraction at boundary of media



© 2008 Steve Marschner • 11

Snell's Law

- Tells us where the refracted ray goes
- Computation
  - ratio of sines is ratio of in-plane components
  - project to surface; scale by eta ratio; recompute normal-direction component
  - total internal reflection



$$\eta_1 \sin\theta_1 = \eta_2 \sin\theta_2$$

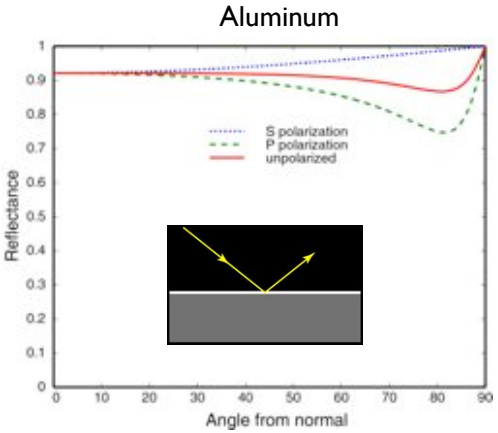
© 2008 Steve Marschner • 12

Ray tracing dielectrics

- Like a simple mirror surface, use recursive ray tracing
- But we need two rays
  - One reflects off the surface (same as mirror ray)
  - The other crosses the surface (computed using Snell's law)
    - Doesn't always exist (total internal reflection)
- Splitting into two rays, recursively, creates a ray tree
  - Very many rays are traced per viewing ray
  - Ways to prune the tree
    - Limit on ray depth
    - Limit on ray attenuation

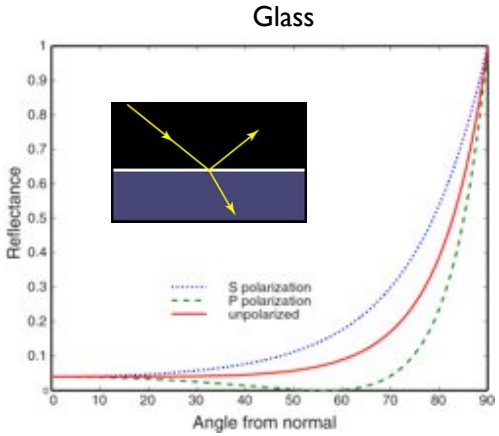
Specular reflection from metal

- Reflectance does depend on angle
  - but not much
  - safely ignored in basic rendering



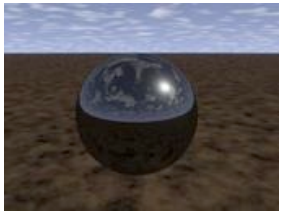
Specular reflection from glass/water

- Dependence on angle is dramatic!
  - about 4% at normal incidence
  - always 100% at grazing
  - remaining light is transmitted
- This is important for proper appearance

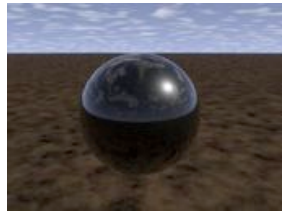


Fresnel reflection

- Black glazed sphere
  - reflection from glass surface
  - transmitted ray is discarded



constant reflectance



Fresnel reflectance

Fresnel's formulas

- They predict how much light reflects from a smooth interface between two materials
  - usually one material is empty space

$$F_p = \frac{\eta_2 \cos\theta_1 - \eta_1 \cos\theta_2}{\eta_2 \cos\theta_1 + \eta_1 \cos\theta_2}$$

$$F_s = \frac{\eta_1 \cos\theta_1 - \eta_2 \cos\theta_2}{\eta_1 \cos\theta_1 + \eta_2 \cos\theta_2}$$

$$R = \frac{1}{2}\left(F_p^2 + F_s^2\right)$$

  - R is the fraction that is reflected
  - $(1 - R)$ is the fraction that is transmitted

Schlick's approximation

- For graphics, a quick hack to get close with less computation:

$$\tilde{R} = R_0 + (1 - R_0)(1 - \cos\theta)^5$$

- R_0 is easy to compute:

$$F_p = \frac{\eta_2 - \eta_1}{\eta_2 + \eta_1}$$

$$F_s = \frac{\eta_1 - \eta_2}{\eta_1 + \eta_2}$$

$$R_0 = \left(\frac{\eta_2 - \eta_1}{\eta_2 + \eta_1}\right)^2$$

[Josh Wills | 2003 UCSD Rendering Competition]

Fresnel reflection

[Mike Hill & Gaain Kwan | Stanford cs348 competition 2001]

Basic ray tracing

- Many advanced methods build on the basic ray tracing paradigm
- Basic ray tracer: one sample for everything
  - one ray per pixel
  - one shadow ray for every point light
  - one reflection ray, possibly one refraction ray, per intersection

© 2003 Steve Marschner •

Discontinuities in basic RT

- Perfectly sharp object silhouettes in image
  - leads to aliasing problems (stair steps)
- Perfectly sharp shadow edges
  - everything looks like it's in direct sun
- Perfectly clear mirror reflections
  - reflective surfaces are all highly polished
- Perfect focus at all distances
  - camera always has an infinitely tiny aperture
- Perfectly frozen instant in time (in animation)
  - motion is frozen as if by strobe light

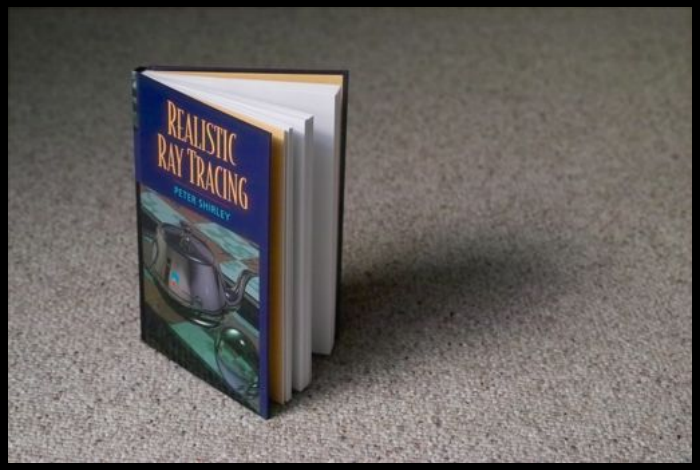
© 2003 Steve Marschner •

Basic ray traced image

[Glassner 89]

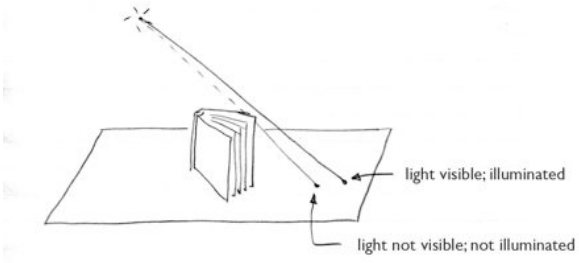
© 2003 Steve Marschner •

Soft shadows



© 2003 Steve Marschner •

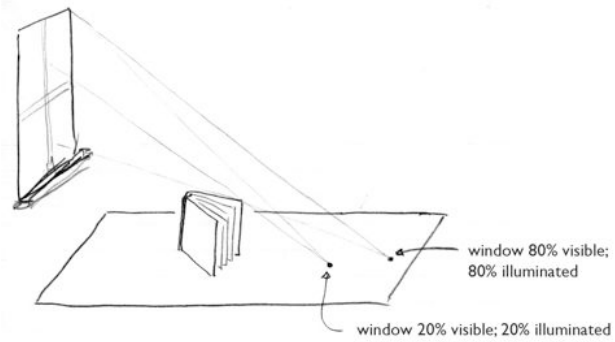
Cause of soft shadows



point lights cast hard shadows

© 2003 Steve Marschner •

Cause of soft shadows



area lights cast soft shadows

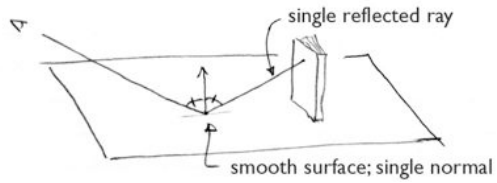
© 2003 Steve Marschner •

Glossy reflection

[Lafortune et al. 97]

© 2003 Steve Marschner •

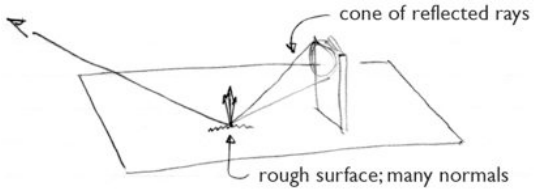
Cause of glossy reflection



smooth surfaces produce sharp reflections

© 2003 Steve Marschner •

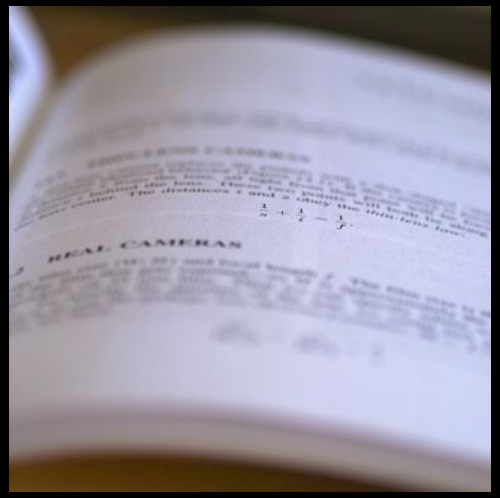
Cause of glossy reflection



rough surfaces produce soft (glossy) reflections

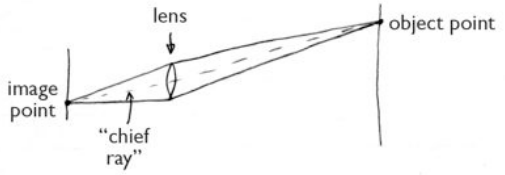
© 2003 Steve Marschner •

Depth of field



© 2003 Steve Marschner •

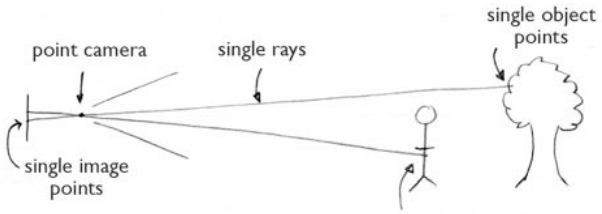
Cause of focusing effects



what lenses do (roughly)

© 2003 Steve Marschner •

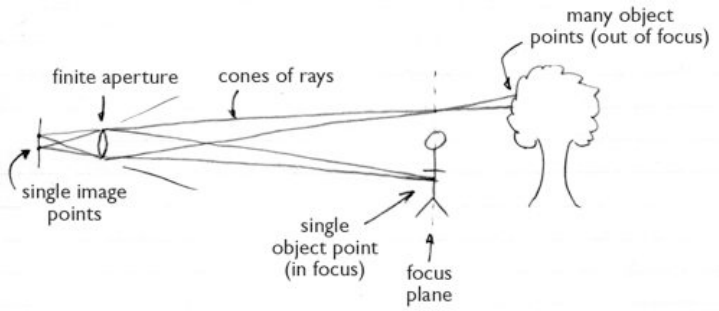
Cause of focusing effects



point aperture produces always-sharp focus

© 2003 Steve Marschner •

Cause of focusing effects



finite aperture produces limited depth of field

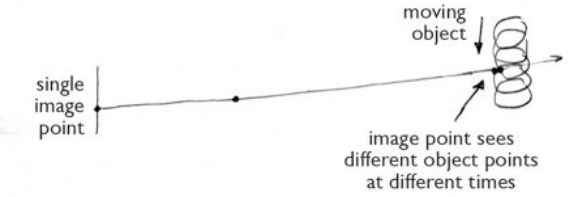
© 2003 Steve Marschner •

Motion blur

[Cook, Porter, Carpenter 1984]

© 2003 Steve Marschner •

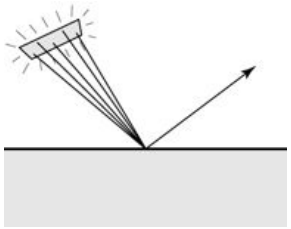
Cause of motion blur



© 2003 Steve Marschner •

Creating soft shadows

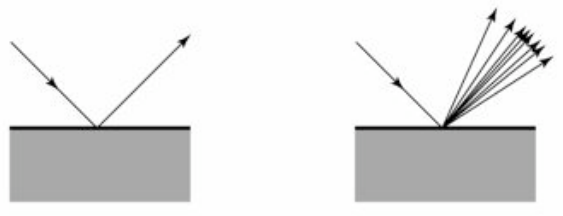
- For area lights: use many shadow rays
  - and each shadow ray gets a different point on the light
- Choosing samples
  - general principle: start with uniform in square



© 2003 Steve Marschner •

Creating glossy reflections

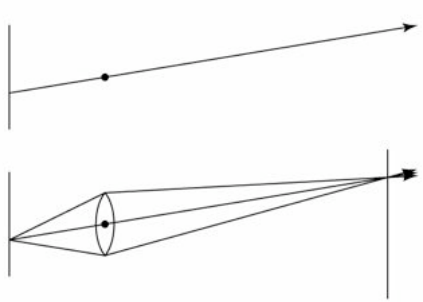
- Jitter the reflected rays
  - Not exactly in mirror direction; add a random offset
  - Can work out math to match Phong exactly
  - Can do this by jittering the normal if you want



© 2003 Steve Marschner •

Depth of field

- Make eye rays start at random points on aperture
  - always going toward a point on the focus plane



© 2003 Steve Marschner •

Motion blur

- Caused by finite shutter times
  - strobing without blur
- Introduce time as a variable throughout the system
  - object are hit by rays according to their position at a given time
- Then generate rays with times distributed over shutter interval

© 2003 Steve Marschner •

Generating samples

- A complicated question in general
- Basic idea: start with random points in a square
- Monte Carlo methods—CS 667

© 2003 Steve Marschner •