Fall 2015

CSCI 420: Computer Graphics

11.1 Global Illumination

Hao Li / Chloe LeGendre

http://cs420.hao-li.com

Global Illumination

The Blue Umbrella © Pixar 2013

Global Illumination

The Blue Umbrella © Pixar 2013

Global Illumination

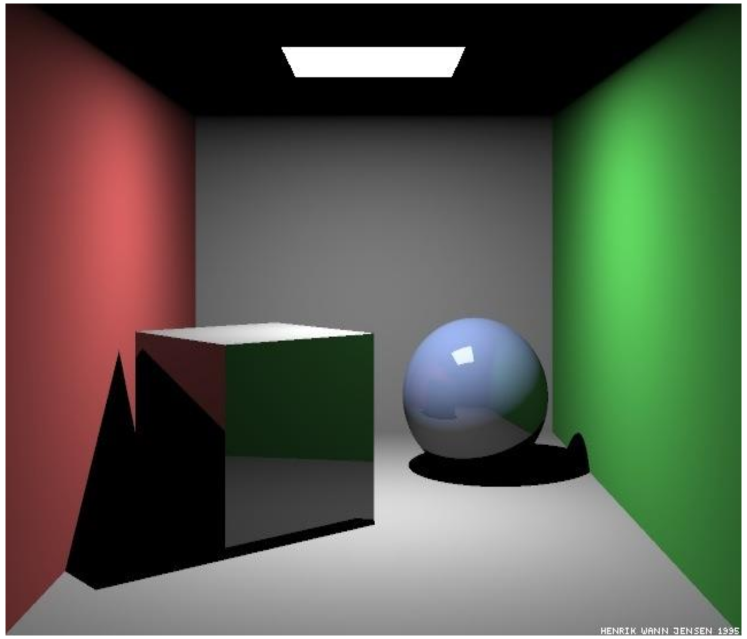
Le Parapluie Café

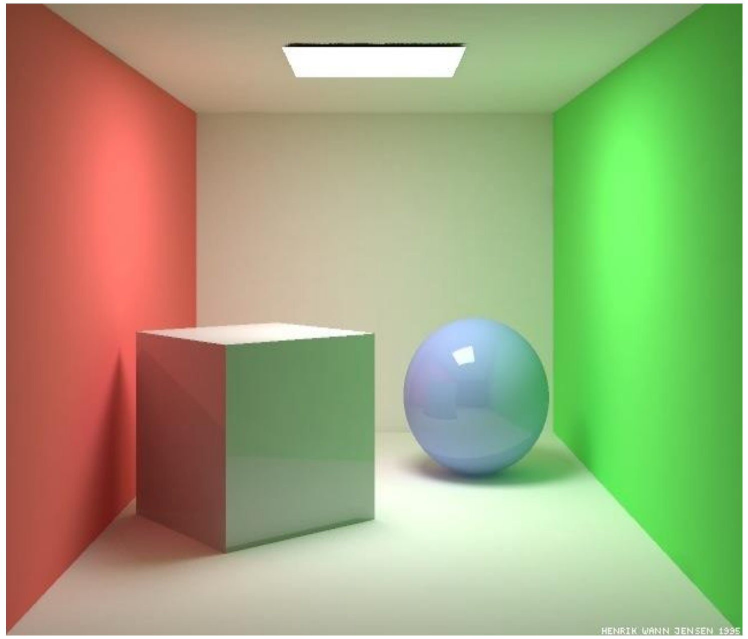
The Blue Umbrella © Pixar 2013

Global Illumination

- Lighting based on the full scene
- Lighting based on physics (optics)
- Traditionally represented by two algorithms
  - Raytracing – 1980
  - Radiosity – 1984
- More modern techniques include photon mapping and many variations of raytracing and radiosity ideas

Direct Illumination vs. Global Illumination

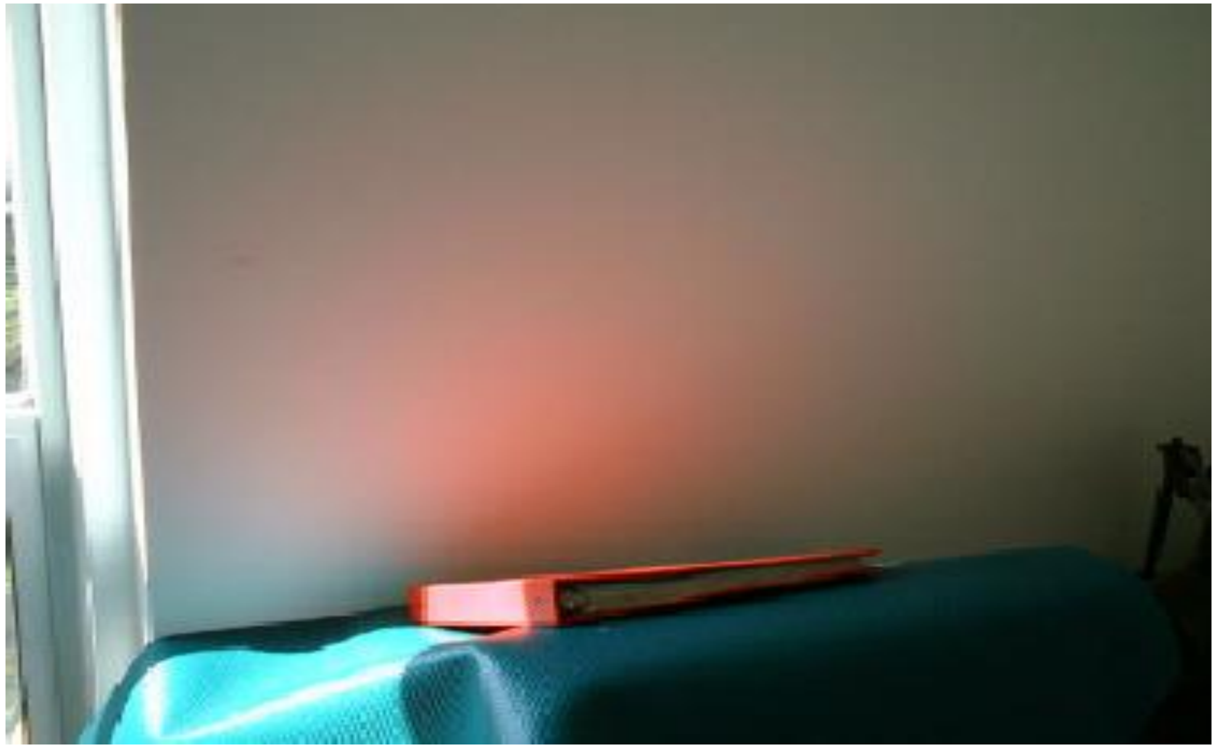




- single (or few) bounces of the light only
- for example, ray casting
- no recursion (or shallow recursion only)
- fast lighting calculations based on light and normal vectors

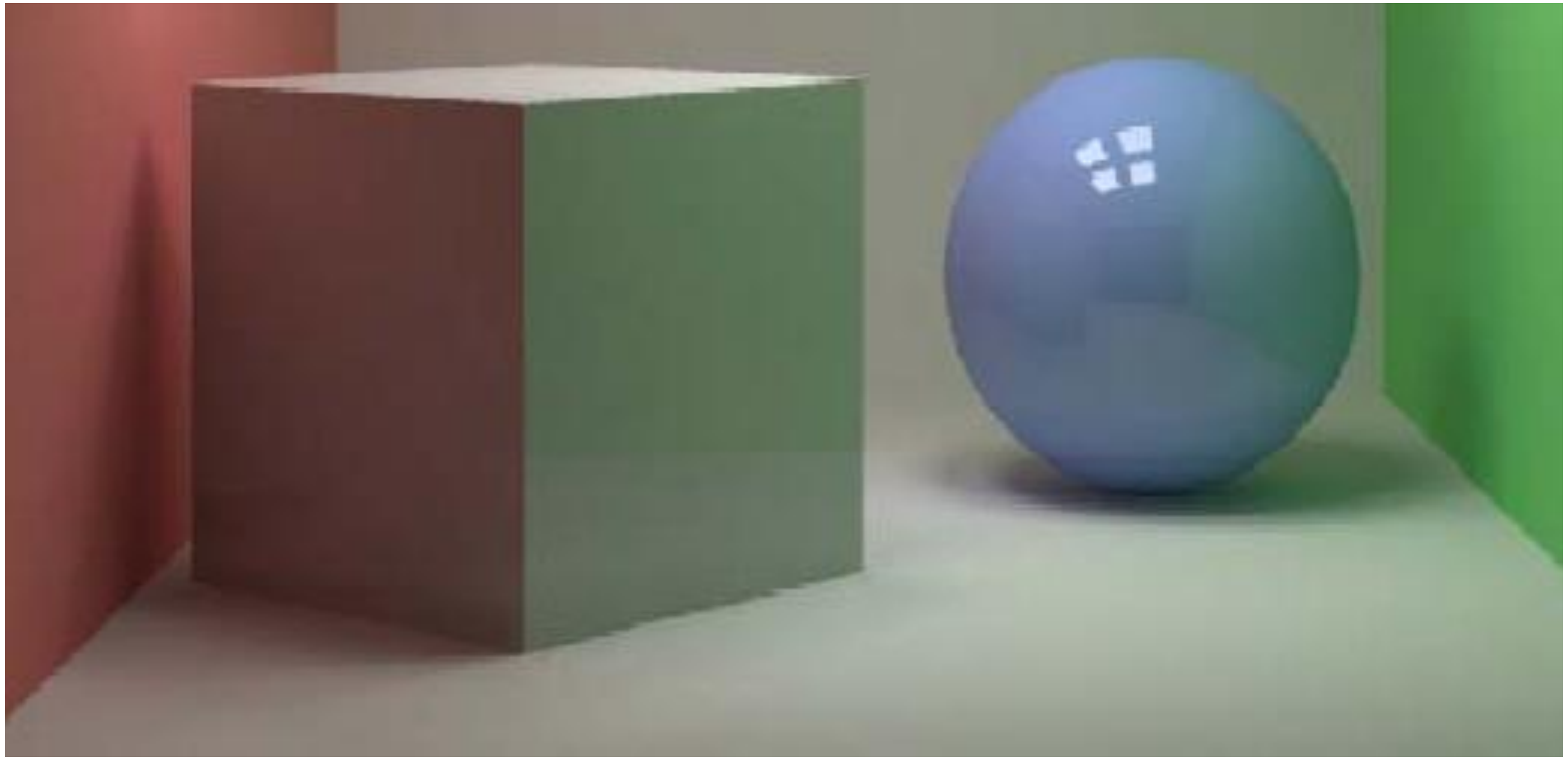
- reflected, scattered and transmitted light
- many (infinite) number of bounces
- physically based light transport

Indirect Illumination



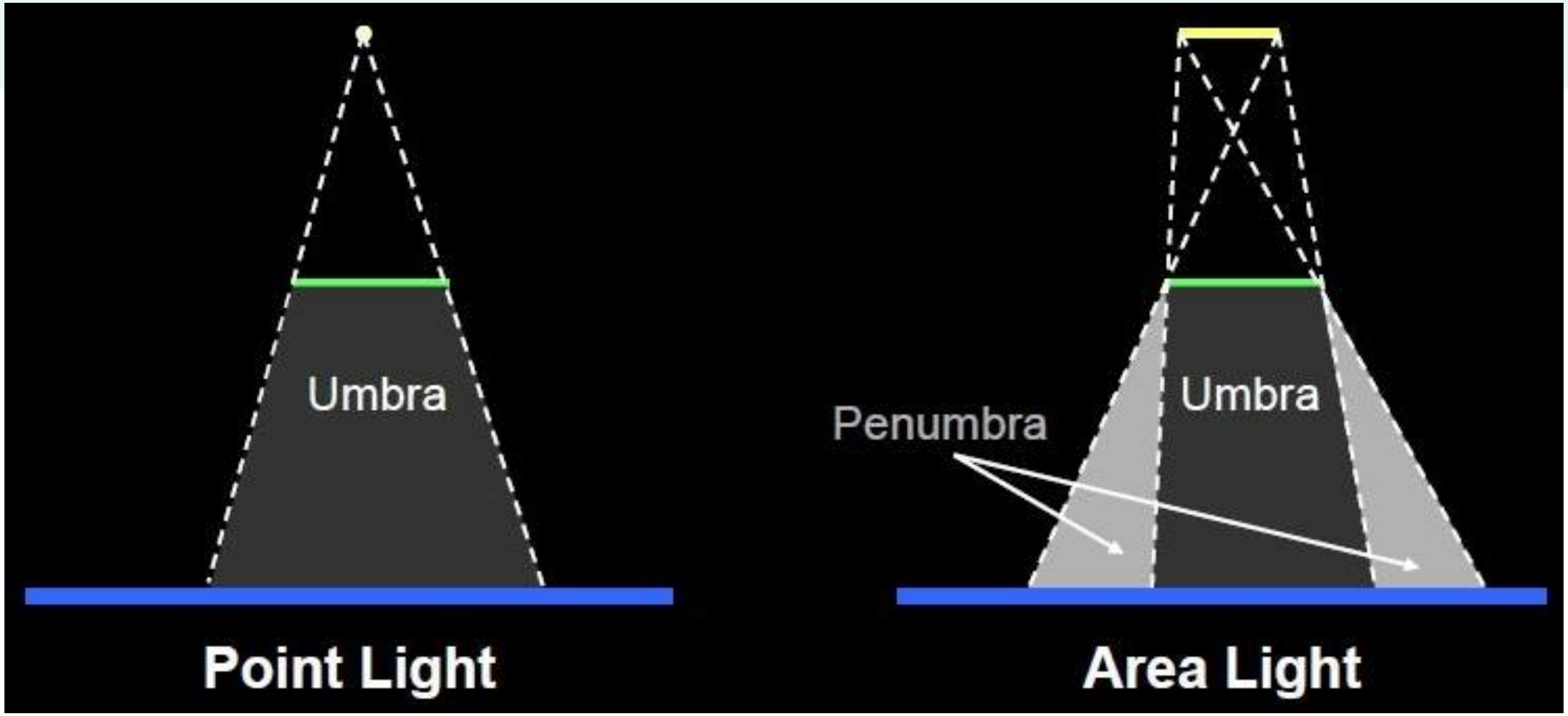
Color Bleeding

Soft Shadows



Shadows are much darker where the direct and indirect illuminations are occluded. Such shadows are important for “sitting” the sphere in the scene. They are difficult to fake without global illumination.

Soft Shadows



Soft shadows must be faked when using point light sources.

Caustics

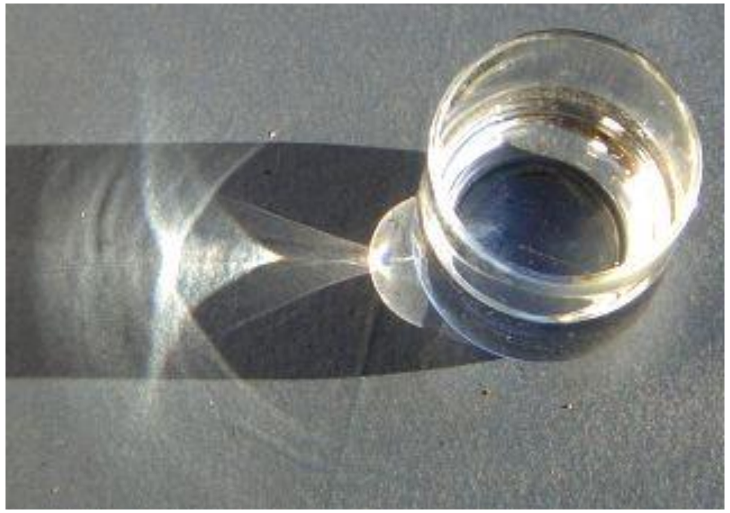
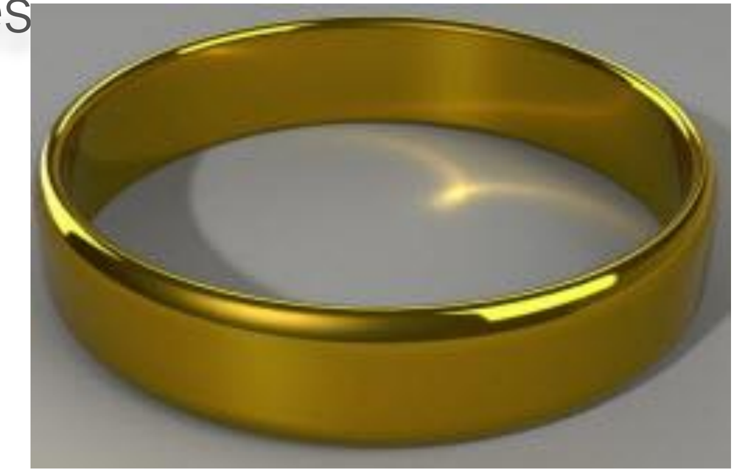
- Refractive:

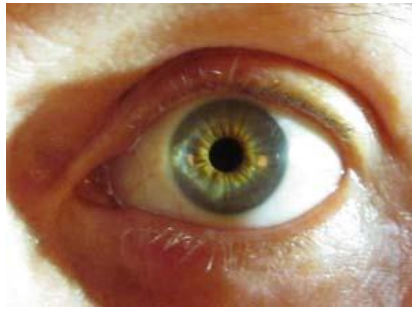
Transmitted light that refocuses on a surface, usually in a pretty pattern

- Reflective:

Reflected light that refocuses on a surface

- Not present with direct illumination

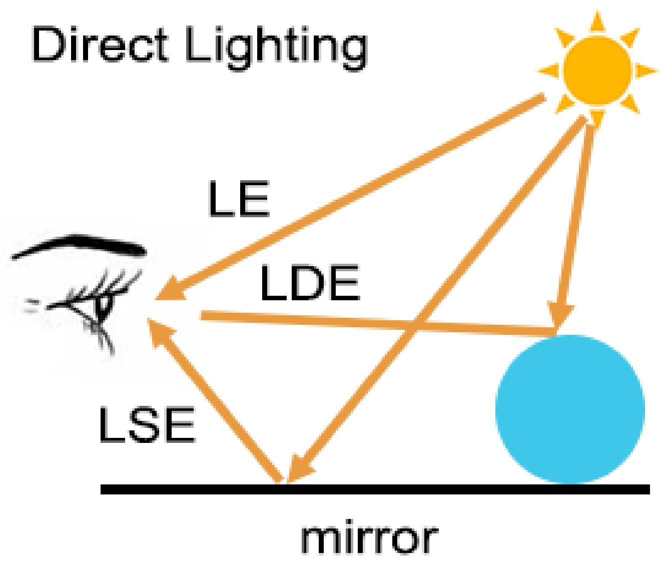


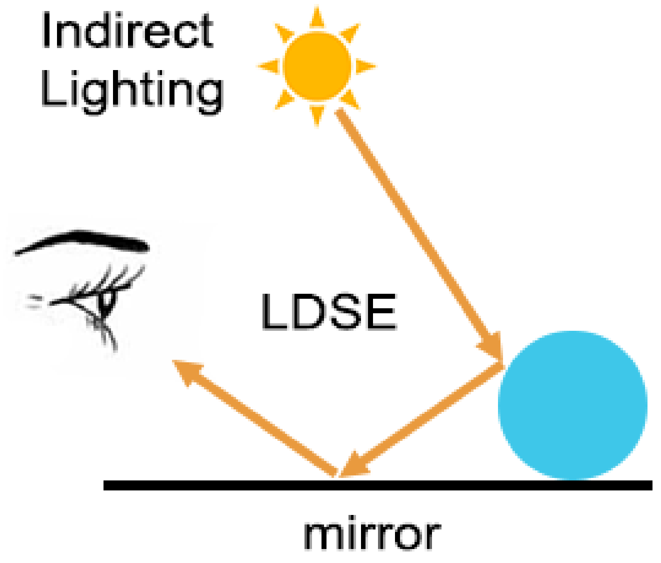
Caustics

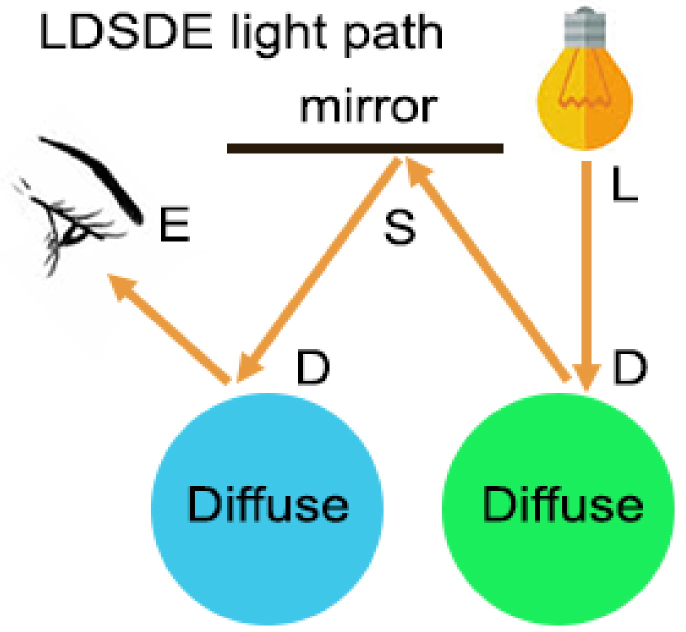
Caustics on the set of Bladerunner 2049

Light Paths

L = Light
E = Eye (camera)
D = Diffuse
S = Specular







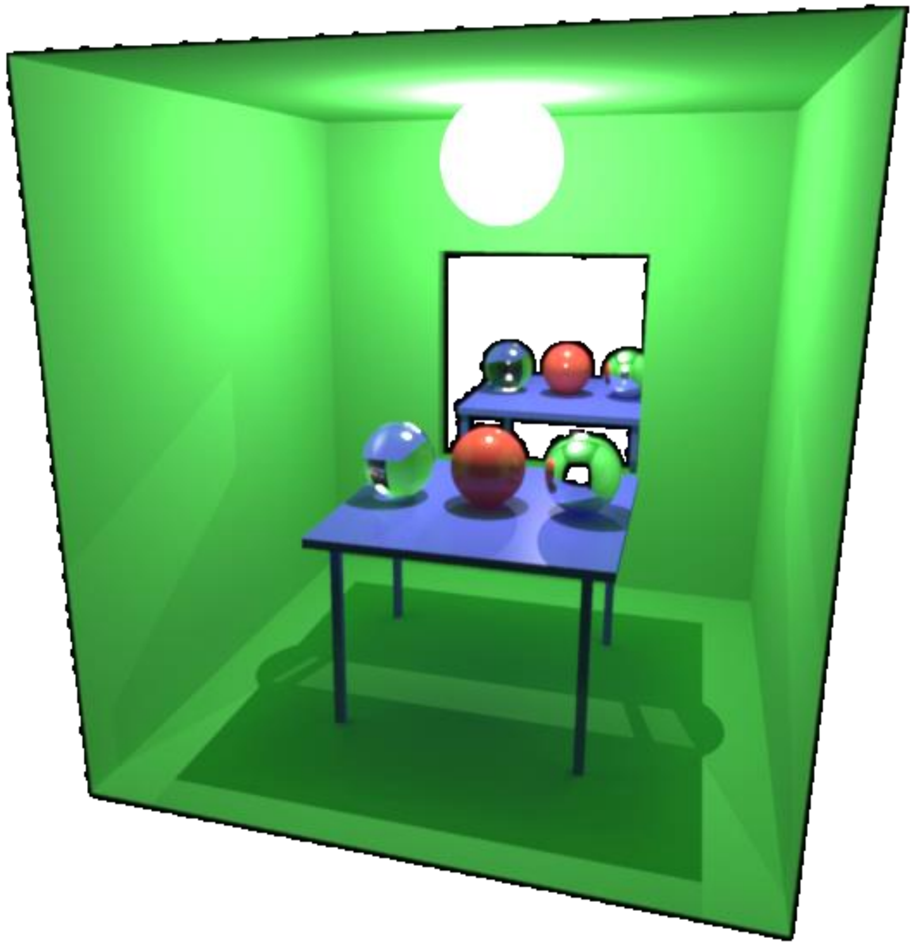
www.scratchapixel.com

Light Transport and Global Illumination

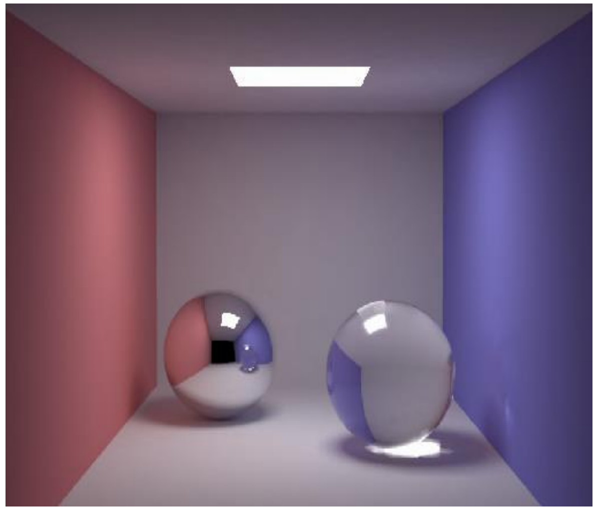
- Diffuse to diffuse → (LD*E)
- Diffuse to specular → (LDSE)
- Specular to diffuse →(LSDE)
- Specular to specular → (LSSE)
- Ray tracing (viewer dependent)
  - Light to diffuse [backward: EDL]
  - Specular to specular … [backward: E(S*)(D)L]
- Radiosity (viewer independent)
  - Diffuse to diffuse [ED*L]

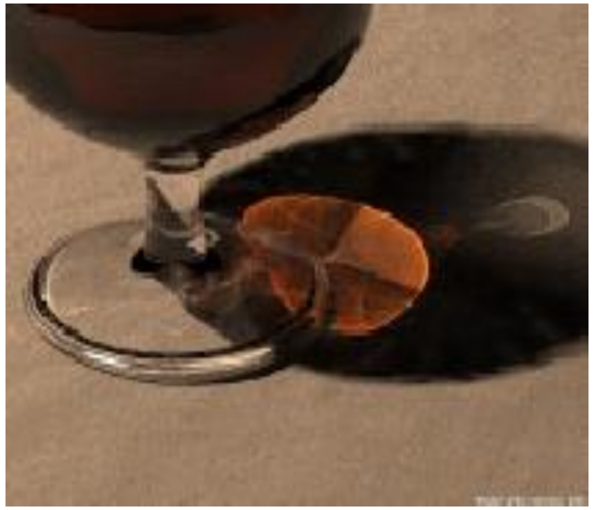
Path Types (behavior of global illumination)

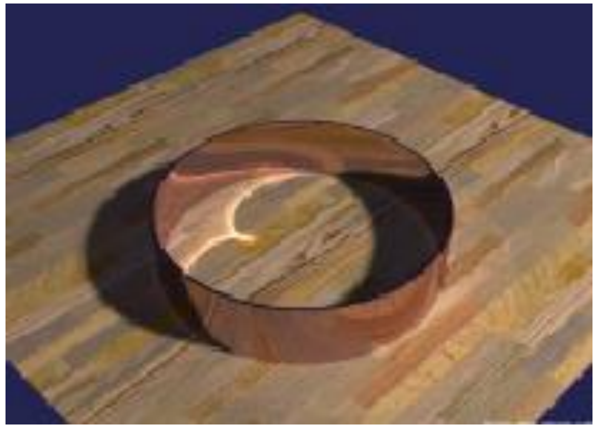
- OpenGL ("local")
  - L(D|S)E
- Ray Tracing
  - ES*DL
- Radiosity
  - ED*L
- Path Tracing
  - attempts to trace "all rays" in a scene



Images Rendered With Global Illumination







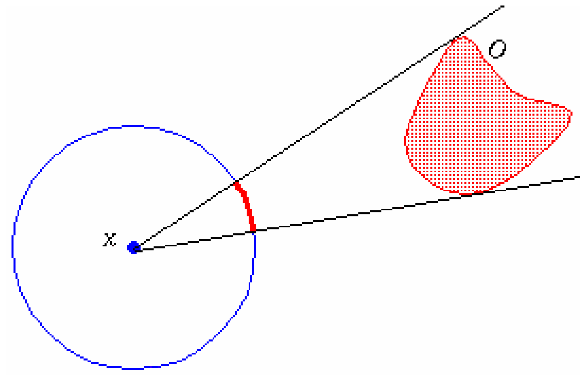
- Caustics
- Color bleeding
- Area light sources and soft shadows

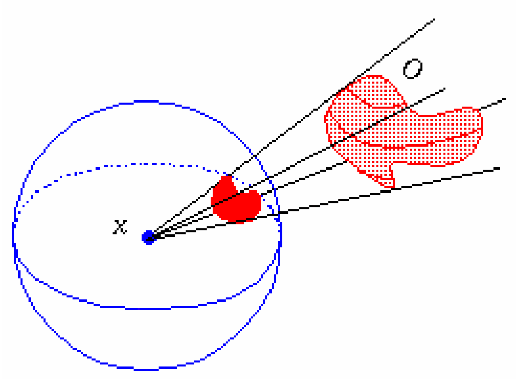
Outline

- Direct and Indirect Illumination
- Bidirectional Reflectance Distribution Function
- Raytracing and Radiosity
- Subsurface Scattering
- Photon Mapping

Solid Angle

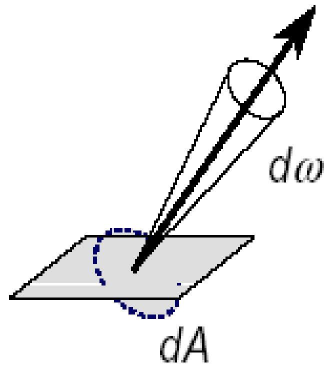
- 2D angle subtended by object O from point x:
  - Length of projection onto unit circle at x
  - Measured in radians (0 to 2π)
- 3D solid angle subtended by O from point x:
  - Area of of projection onto unit sphere at x
  - Measured in steradians (0 to 4π)





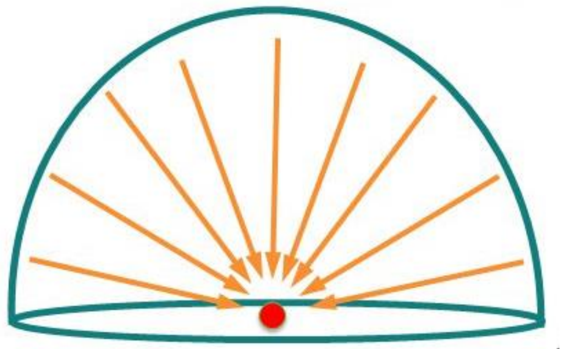
Light and Surfaces

- Radiance (L): Power (φ) per unit area per unit solid angle
  - Measured in W / m^{2}str
  - dA is projected area (perpendicular to given direction)



$$L = \frac{d\Phi}{dA\,d\omega}$$

- Irradiance: Power received per unit area, integrated over all directions
  - Measured in W / m^2



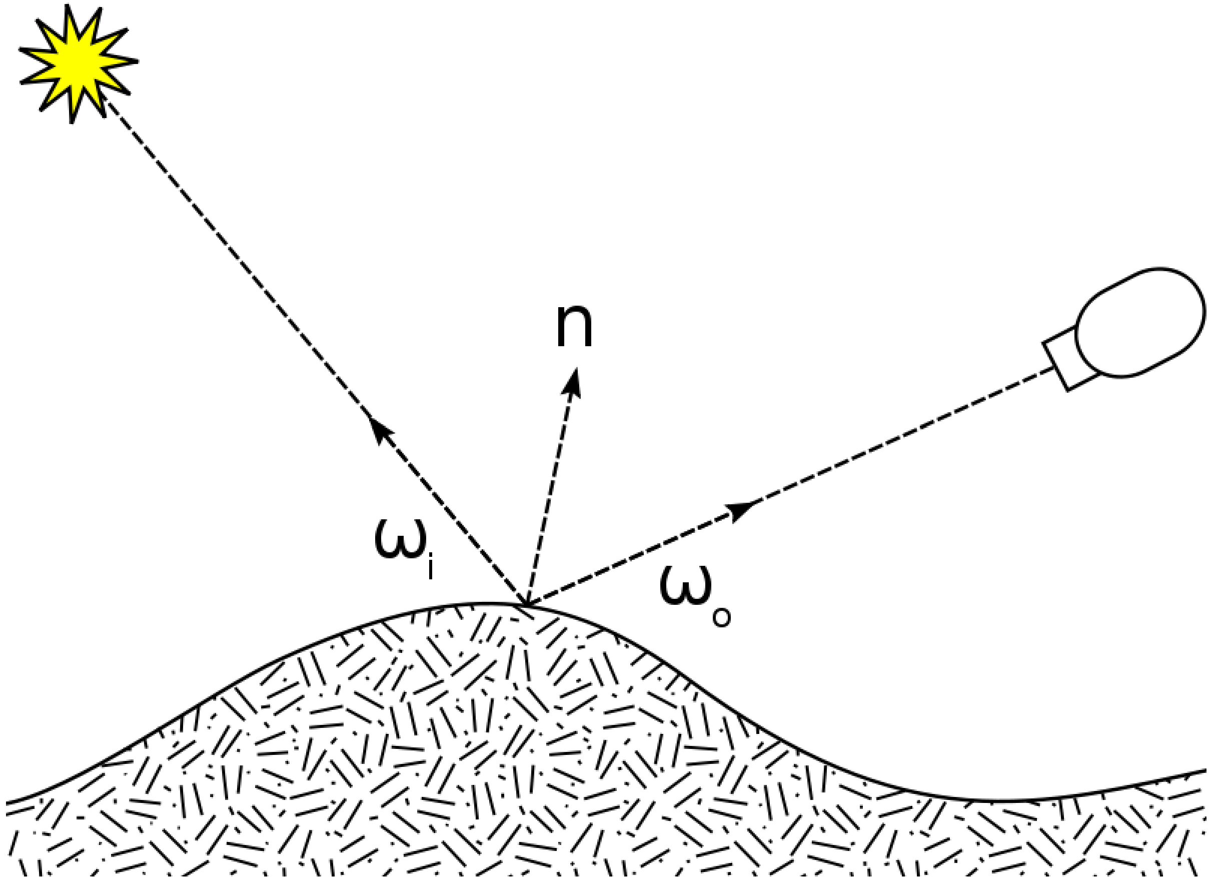
- Radiosity (B): Power emitted/reflected/transmitted per unit area, integrated over all directions
  - Measured in W / m^2

$$B = \int_{\Omega} L(\theta,\phi)\cos\theta\, d\omega$$

Bidirectional Reflectance Distribution Function (BRDF)

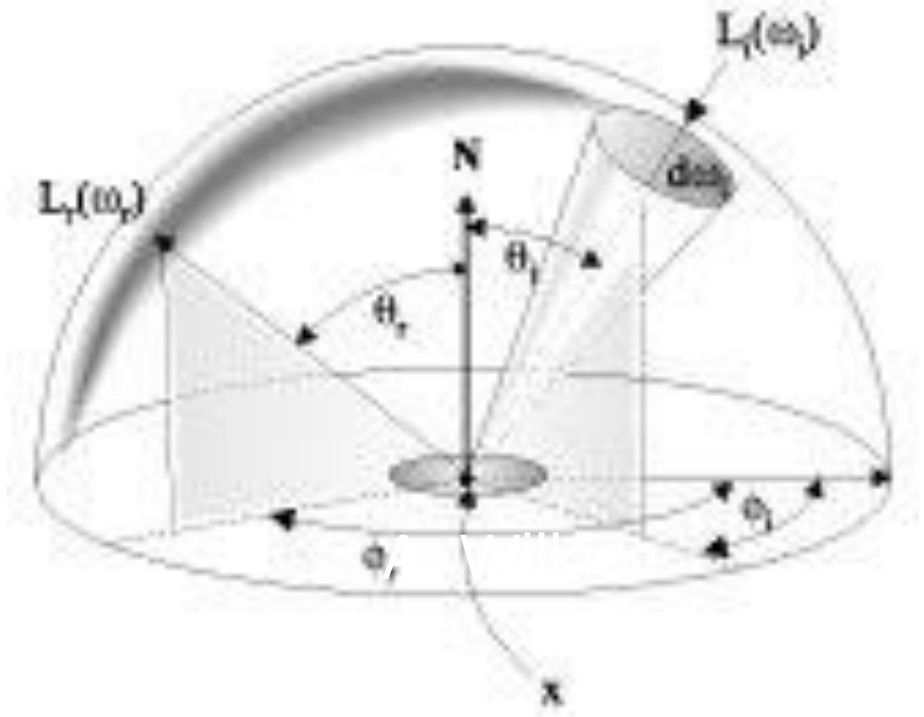
If a ray hits a surface point at angle ω_i, how much light bounces into the direction given by angle ω_o?

It depends on the type of material.

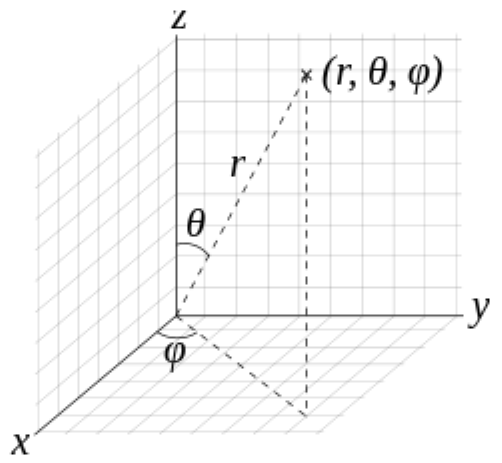


Bidirectional Reflectance Distribution

- General model of light reflection
- Hemispherical function
- 6-dimensional (2-DOF location, 4 angles)

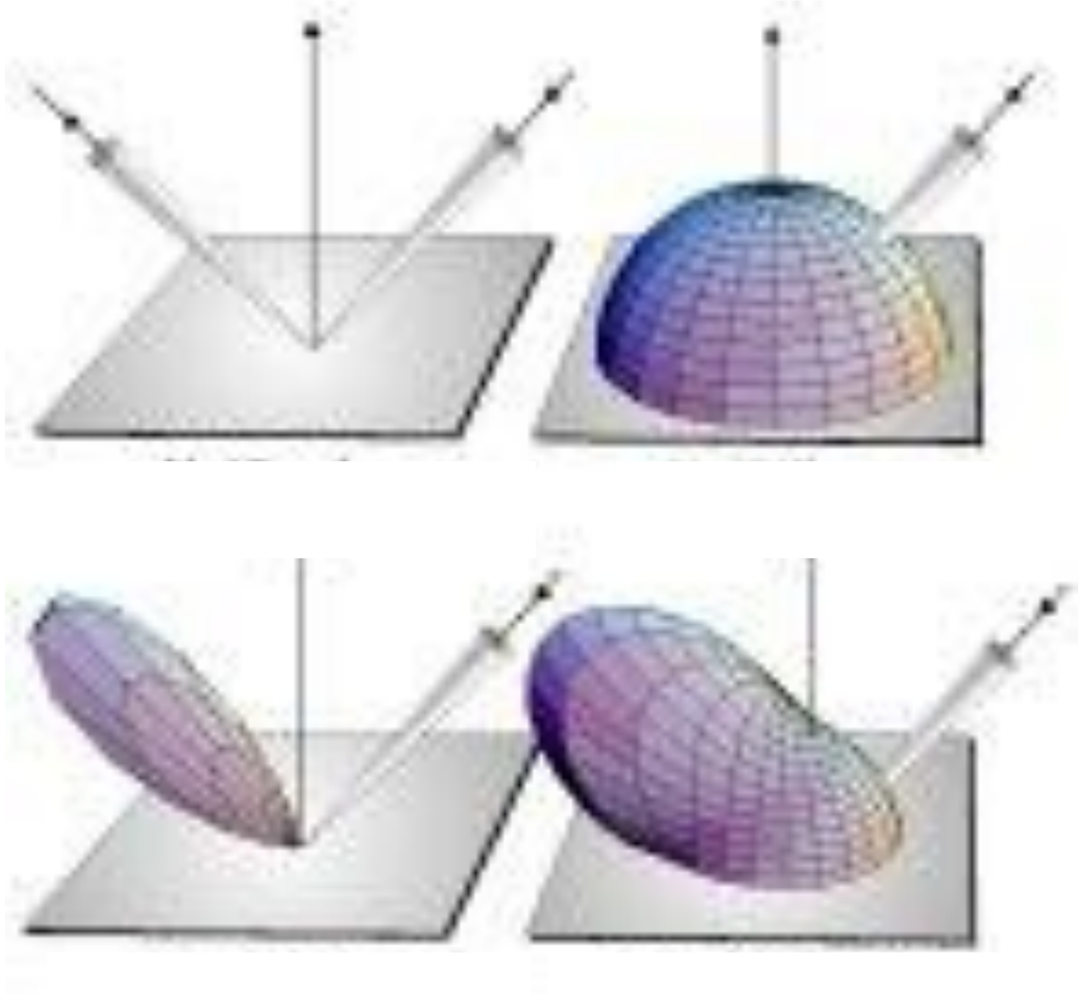


x,y
θ , Φ incoming
θ , Φ outgoing



BRDF Examples

- BRDF is a property of the material
- There is no formula for most materials
- Measure BRDFs for different materials (and store in a table)



Material Examples

Marschner et al. 2000

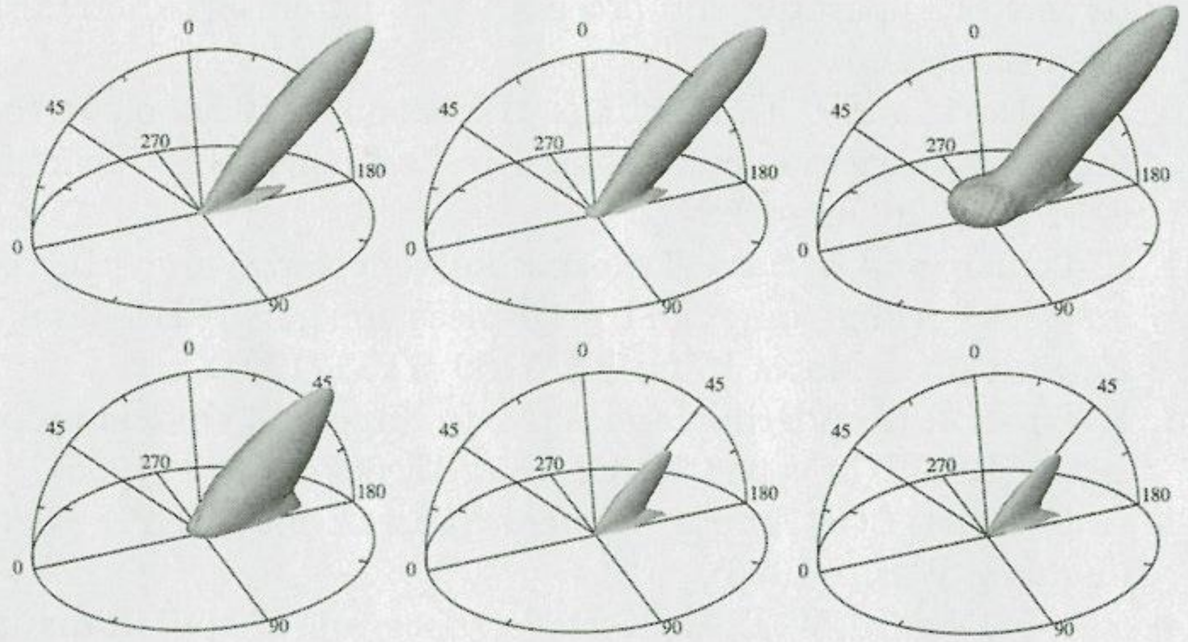
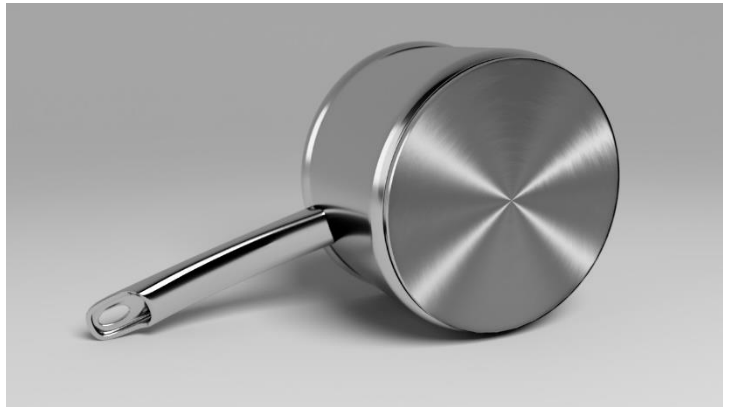


Fig. 16. Resampled scattering diagrams of the BRDF measurements of two paints: a blue enamel (top row) and a red automotive lacquer (bottom row). The RGB color measurements are shown from left to right.

BRDF Isotropy

- Rotation invariance of BRDF
- Reduces 4 angles to 2
- Holds for a wide variety of surfaces
- Anisotropic materials
  - Brushed metal
  - Others?



Rendering Equation

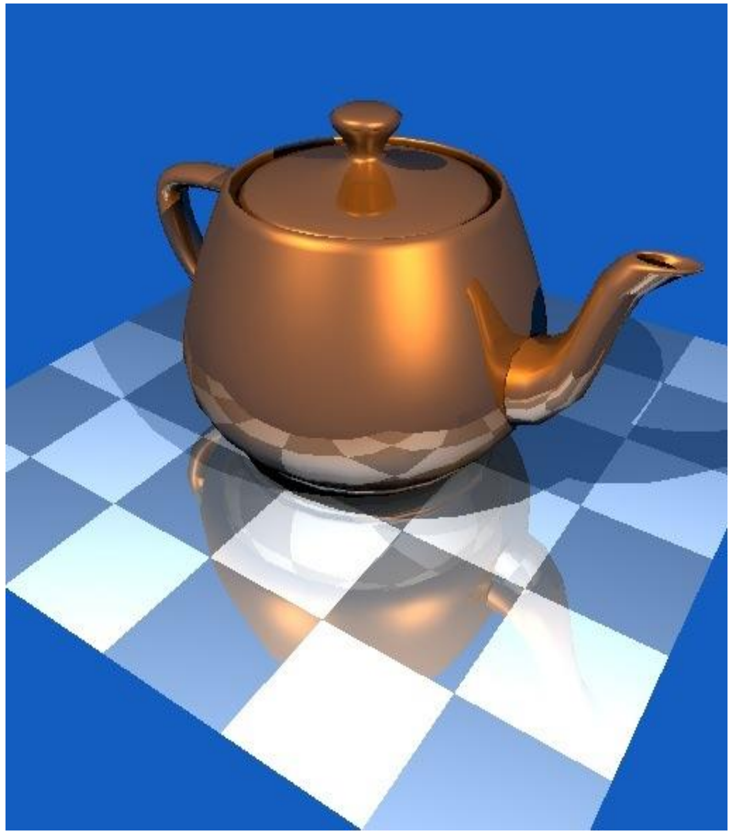
$$L(\mathbf{x},\omega) = E(\mathbf{x},\omega) + \int f_r(\mathbf{x},\omega,\omega')G(\mathbf{x},\mathbf{x}')V(\mathbf{x},\mathbf{x}')L(\mathbf{x}',\omega')dA'$$

- *L* is the radiance from a point *x* on a surface in a given direction *ω*
- *E* is the emitted radiance from a point: *E* is non-zero only if *x* is emissive
- f_r is the BRDF (6 DOF)
- *G* is a geometry term, which depends on the geometric relationship (such as distance) between the two surfaces *x* and *x'*
- *V* is a visibility term: 1 when the surfaces are unobstructed along the direction *ω*, 0 otherwise
- It includes contributions from light bounced many times off surfaces

Outline

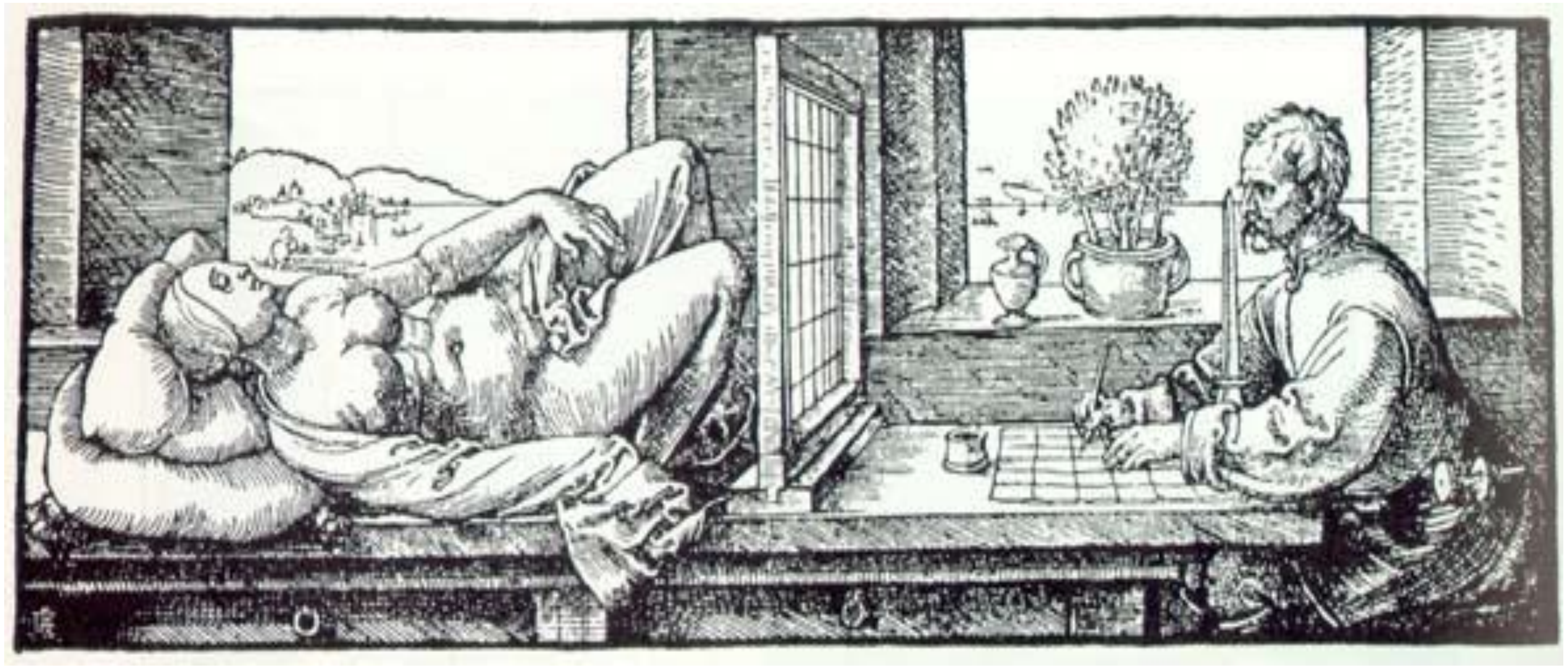
- Direct and Indirect Illumination
- Bidirectional Reflectance Distribution Function
- Raytracing and Radiosity
- Subsurface Scattering
- Photon Mapping

Raytracing



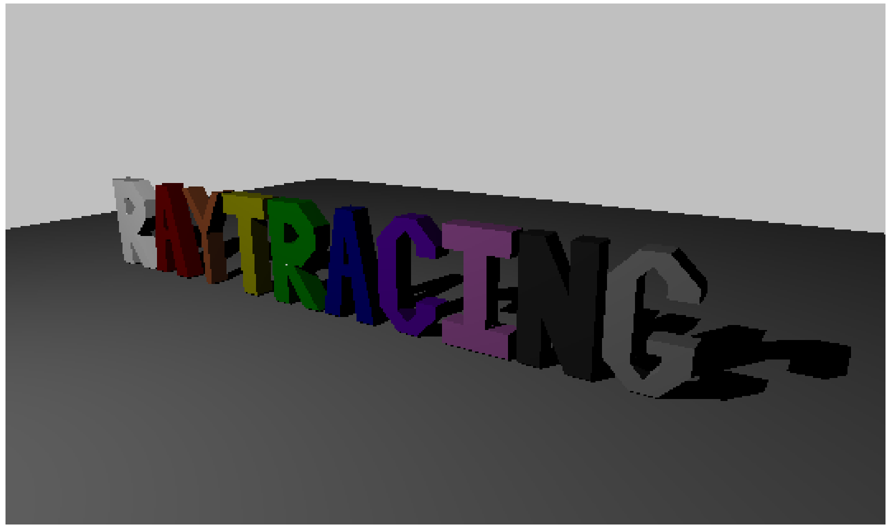
http://jedi.ks.uiuc.edu/~johns/raytracer/raygallery/stills.html

Raytracing

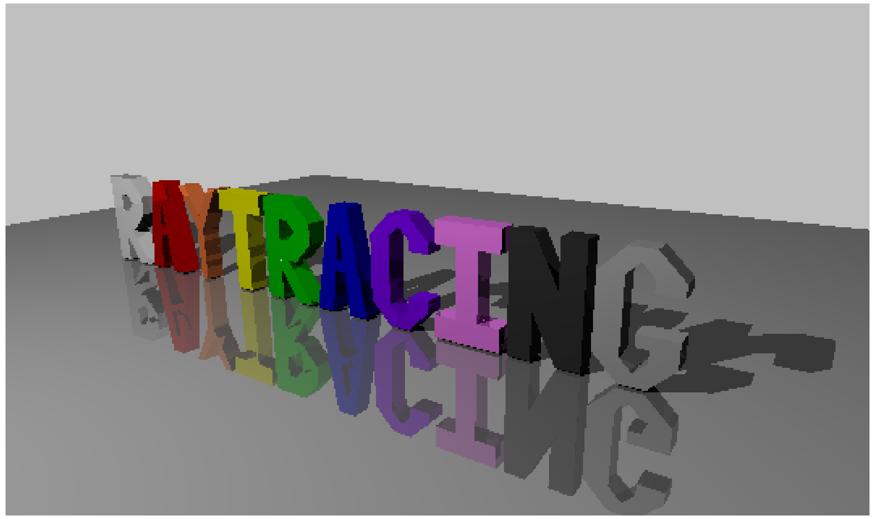


Albrecht DÜRER,
Underweysung der Messung mit dem Zirkel und Richtscheyt
(Nurenberg, 1525), Book 3, figure 67.

Raycasting vs. Raytracing



Raycasting



Raytracing

Raytracing: Pros

- Simple idea and nice results

- Inter-object interaction possible
  - Shadows
  - Reflections
  - Refractions (light through glass, etc.)

- Based on real-world lighting

Raytracing: Cons

- Slow
- Speed often highly scene-dependent
- Lighting effects tend to be abnormally sharp, without soft edges, unless more advanced techniques are used
- Hard to put into hardware

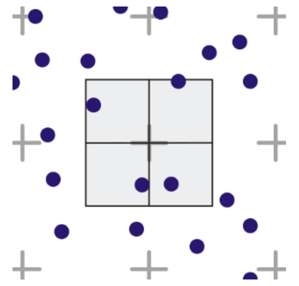
Supersampling I

- Problem: Each pixel of the display represents one single ray
  - Aliasing
  - Unnaturally sharp images

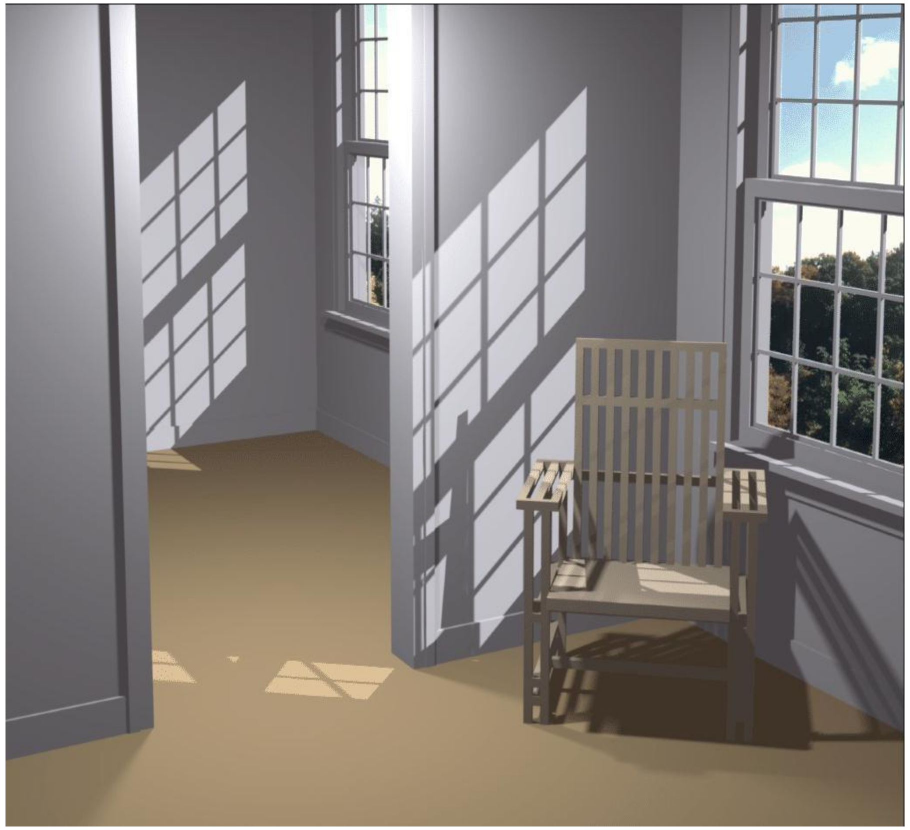
- Solution: Send multiple rays through each “pixel” and average the returned colors together

Supersampling II

- Direct supersampling
  - Split each pixel into a grid and send rays through each grid point
- Adaptive supersampling
  - Split each pixel only if it’s significantly different from its neighbors (image space)
- Jittering
  - Send rays through randomly selected points within the pixel



The Radiosity Method



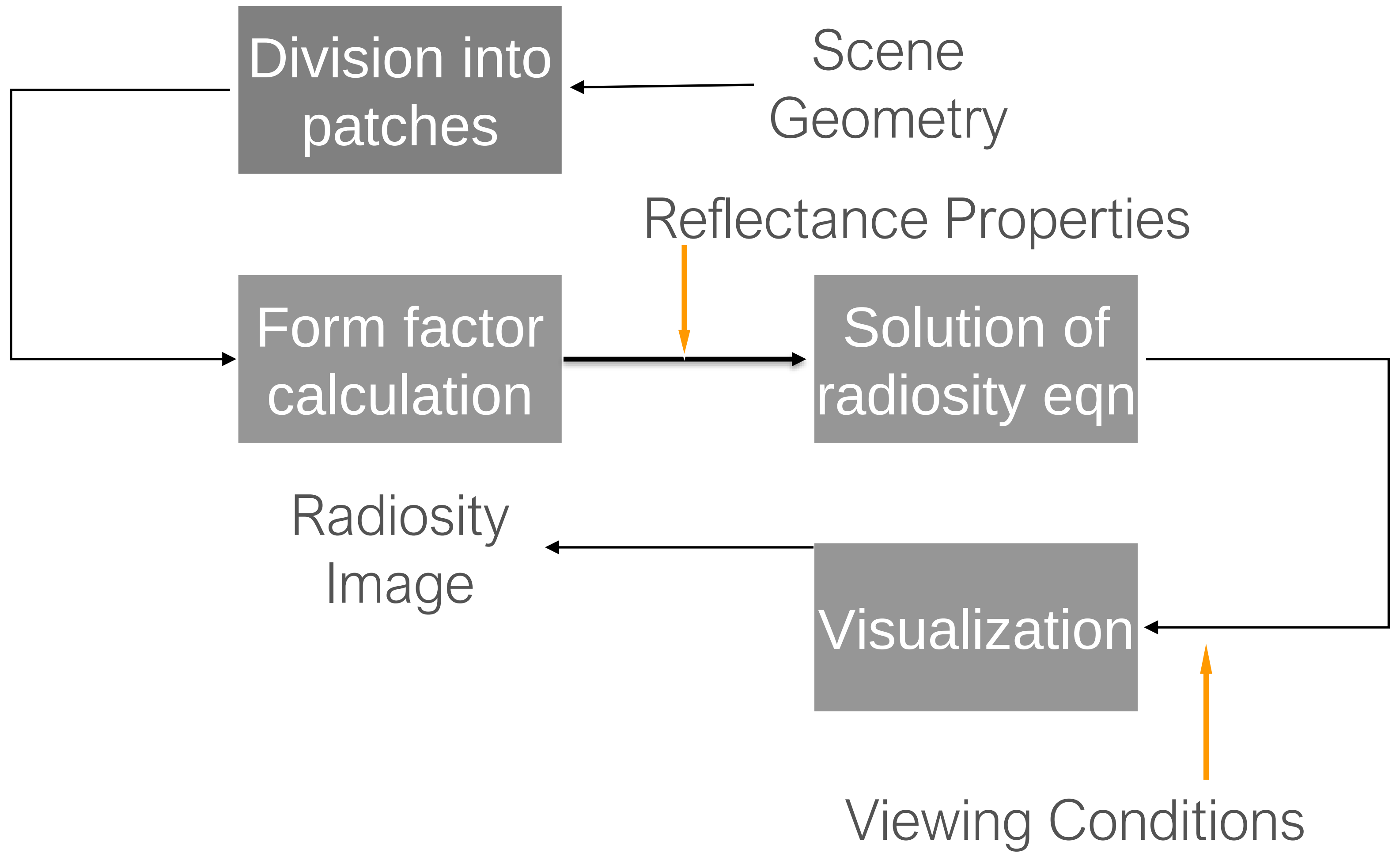
The Radiosity Method

- Divide surfaces into patches
  (e.g., each triangle is one patch)

- Model light transfer between patches as system of linear equations

- Important assumptions:
  - **Diffuse reflection only → E(D*)L**
  - No specular reflection
  - No participating media (no fog)
  - No transmission (only opaque surfaces)
  - Radiosity is constant across each patch
  - Solve for R, G, B separately

(Idealized) Radiosity Computation



Radiosity: Pros

- Can change camera position and re-render with minimal re-computation (viewpoint independence)

- Inter-object interaction possible
  - Soft shadows
  - Indirect lighting
  - Color bleeding

- Accurate simulation of energy transfer

Radiosity: Cons

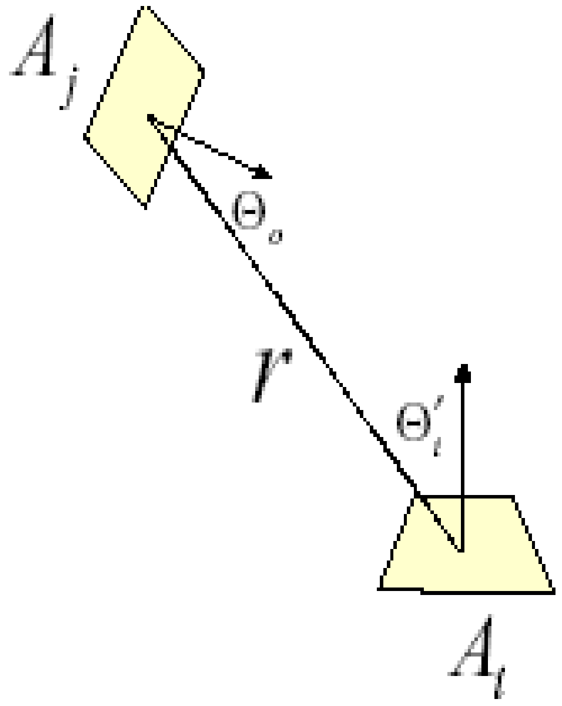
- Precomputation must be re-done if *anything* moves
- Large computational and storage costs
- Non-diffuse light not represented
  - Mirrors and shiny objects hard to include
- Lighting effects tend to be “blurry” (not sharp)
- Not applicable to procedurally defined surfaces

Radiosity Equation

- For each patch i:
$$\begin{aligned} B_i &= E_i + \rho_i \sum_j (F_{ji} A_j / A_i) B_j \\ &= E_i + \rho_i \sum_j F_{ij} B_j \end{aligned}$$
- Variables
  - B_i = radiosity (unknown)
  - E_i = emittance of light sources (given; some patches are light sources)
  - ρ_i = reflectance (given)
  - F_{ij} = form factor from i to j (computed)
    fraction of light emitted from patch i arriving at patch j
  - A_i = area of patch i (computed)

The Form Factor

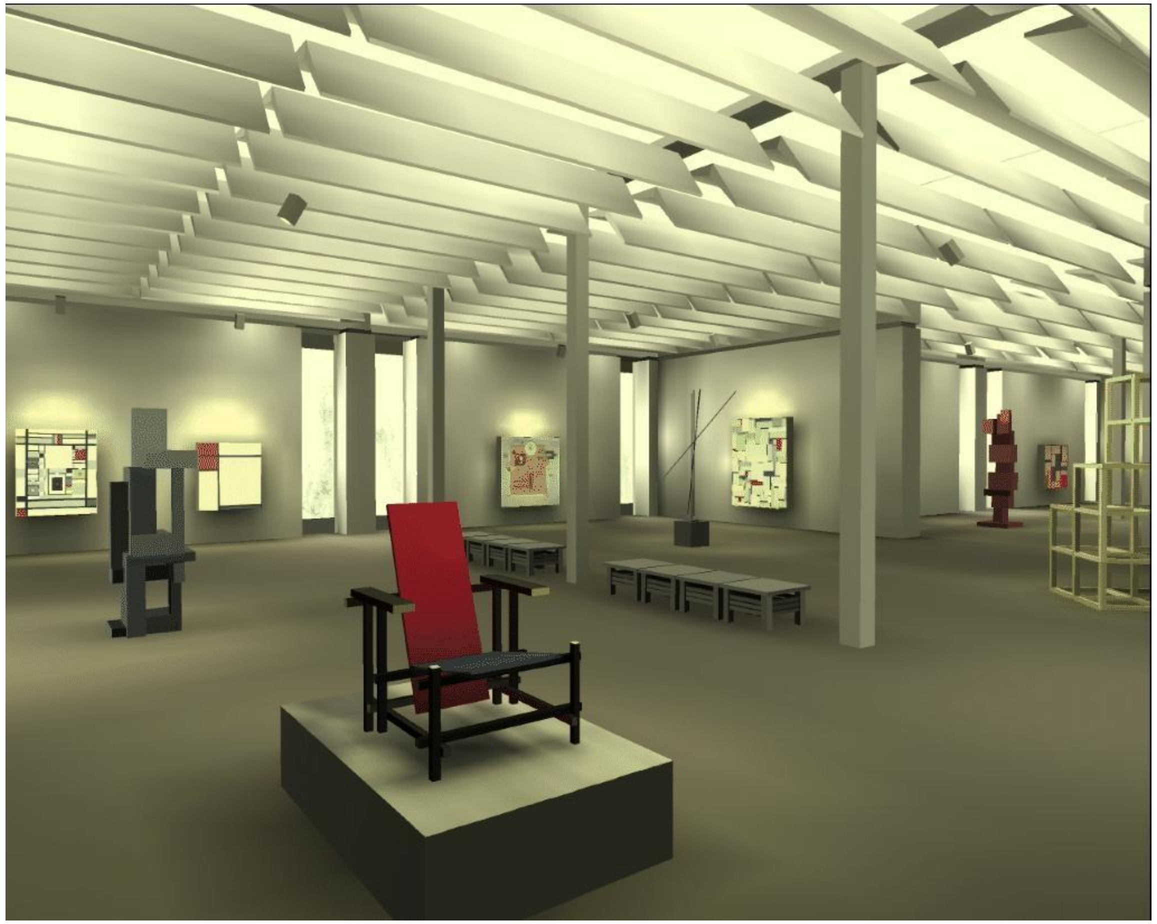
$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{V_{ij} \cos\phi_i \cos\phi_j}{\pi r^2} dA_j dA_i$$



F_{ij} is dimensionless

V_{ij} = 0 if occluded
1 if not occluded
(visibility factor)

Radiosity Example



Museum simulation. Program of Computer Graphics, Cornell University.
50,000 patches. Note indirect lighting from ceiling.

Outline

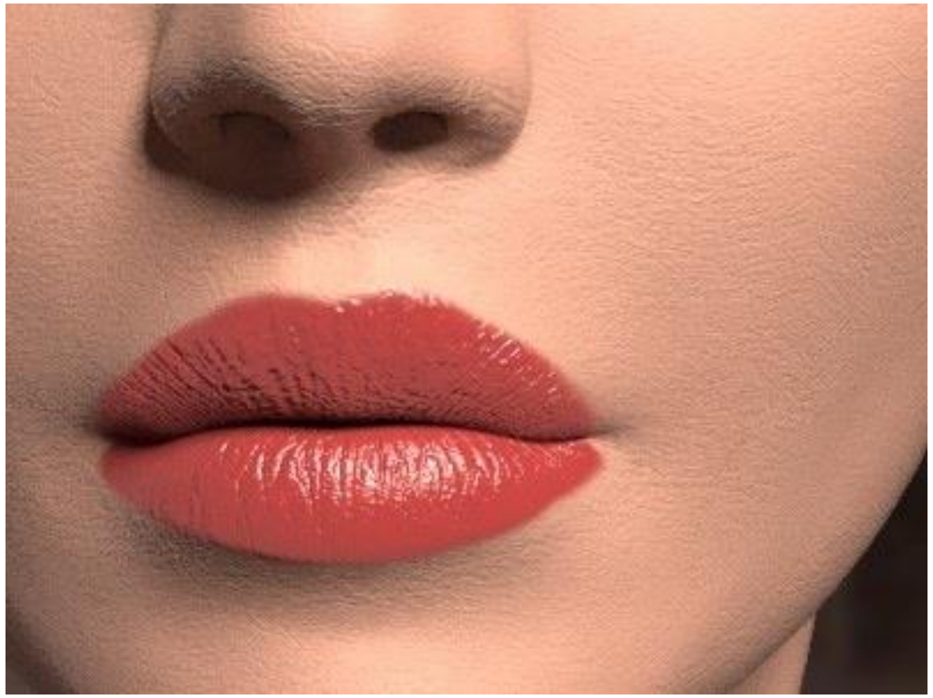
- Direct and Indirect Illumination
- Bidirectional Reflectance Distribution Function
- Raytracing and Radiosity
- Subsurface Scattering
- Photon Mapping

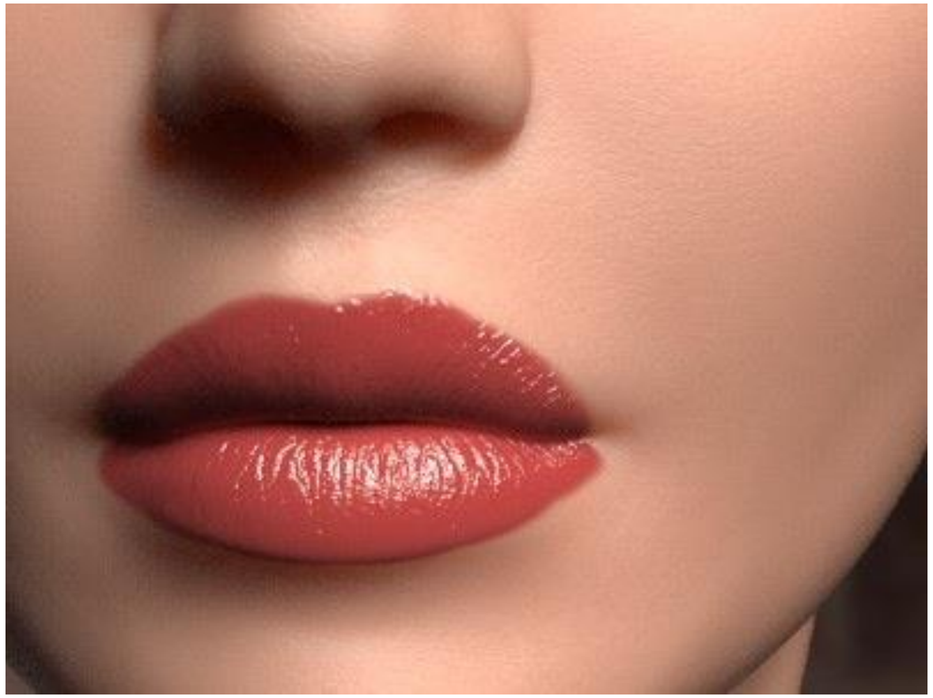
Subsurface Scattering

- Translucent objects: skin, marble, milk
- Light penetrates the object, scatters and exits
- Important and popular in computer graphics

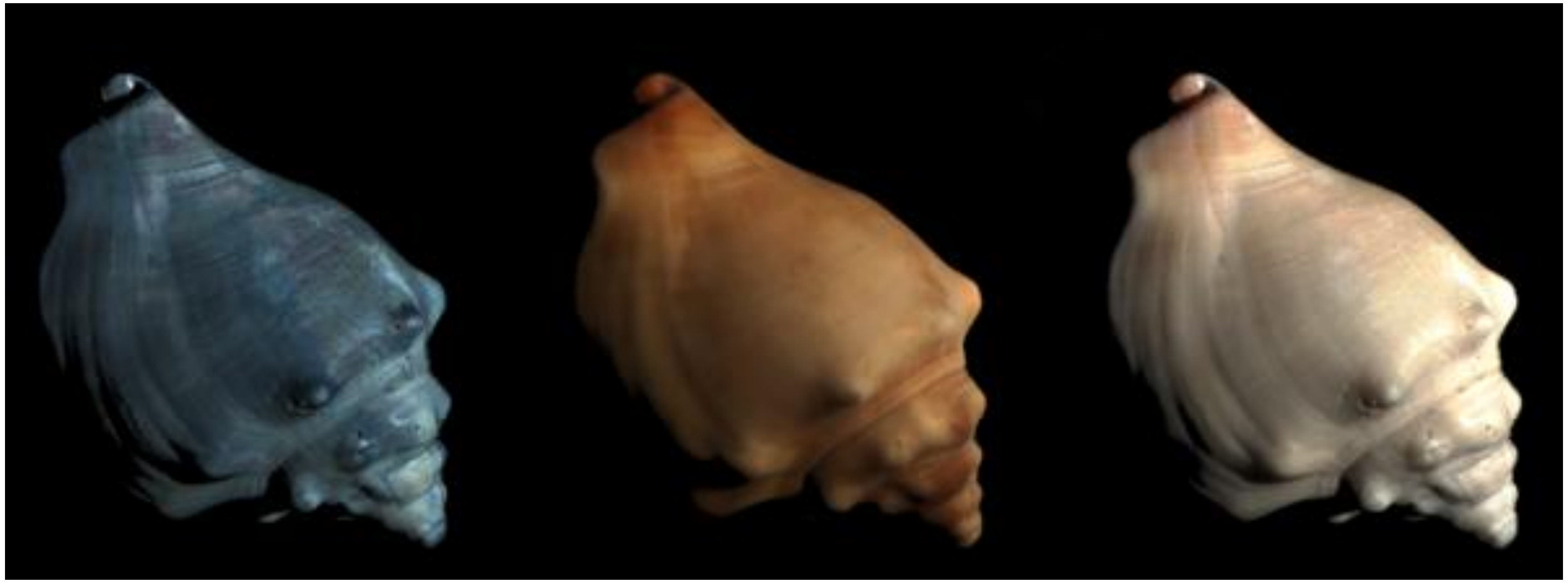
Subsurface Scattering

- Jensen et al. 2001


Using only BRDF


With subsurface light transport

Subsurface Scattering



direct only

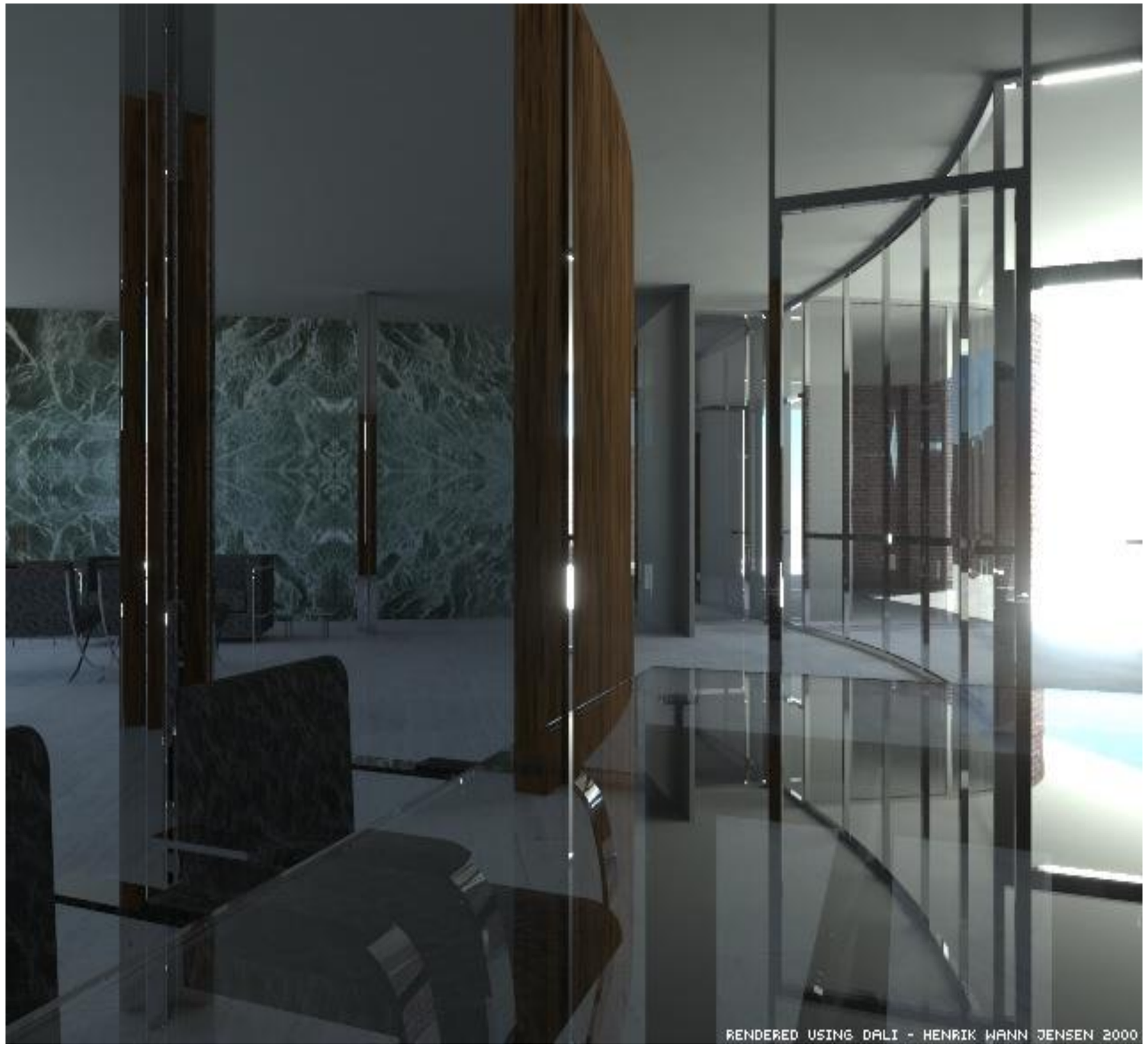
subsurface
scattered only

combined

Outline

- Direct and Indirect Illumination
- Bidirectional Reflectance Distribution Function
- Raytracing and Radiosity
- Subsurface Scattering
- Photon Mapping

Photon Mapping

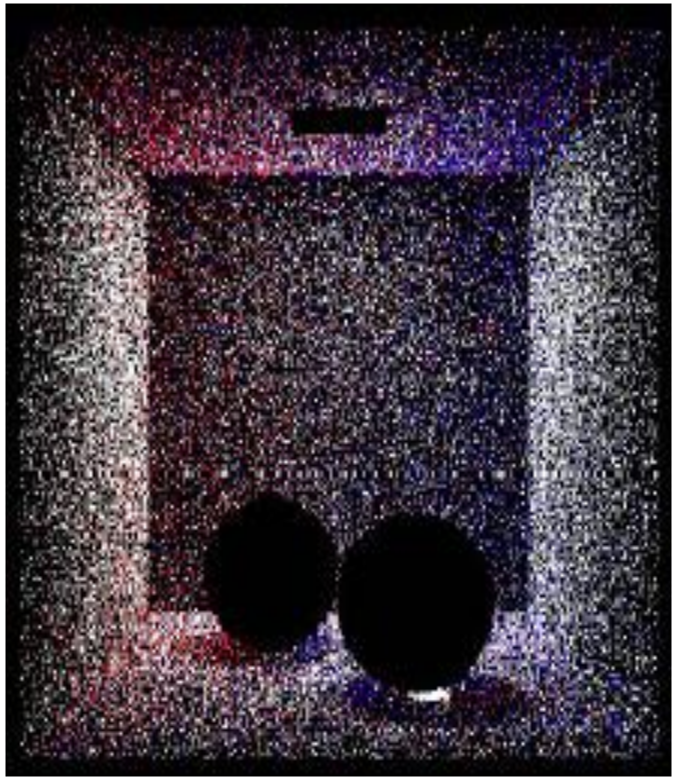


Photon Mapping Basics

- Enhancement to raytracing
- Can simulate caustics
- Can simulate diffuse inter-reflections
  (e.g., the "bleeding" of colored light from a red wall onto a white floor, giving the floor a reddish tint)
- Can simulate clouds or smoke

Photon Mapping

- “Photons” are emitted (raytraced) from light sources
- Photons either bounce or are absorbed
- Photons are stored in a *photon map,* with both position and incoming direction
- Photon map is decoupled from the geometry (often stored in a kd-tree)



Photon Mapping

- Raytracing step uses the closest *N* photons to each ray intersection and estimates the outgoing radiance

- Specular reflections can be done using “usual” raytracing to reduce the number of photons needed

- Numerous extensions to the idea to add more complex effects

Photon Mapping: Pros

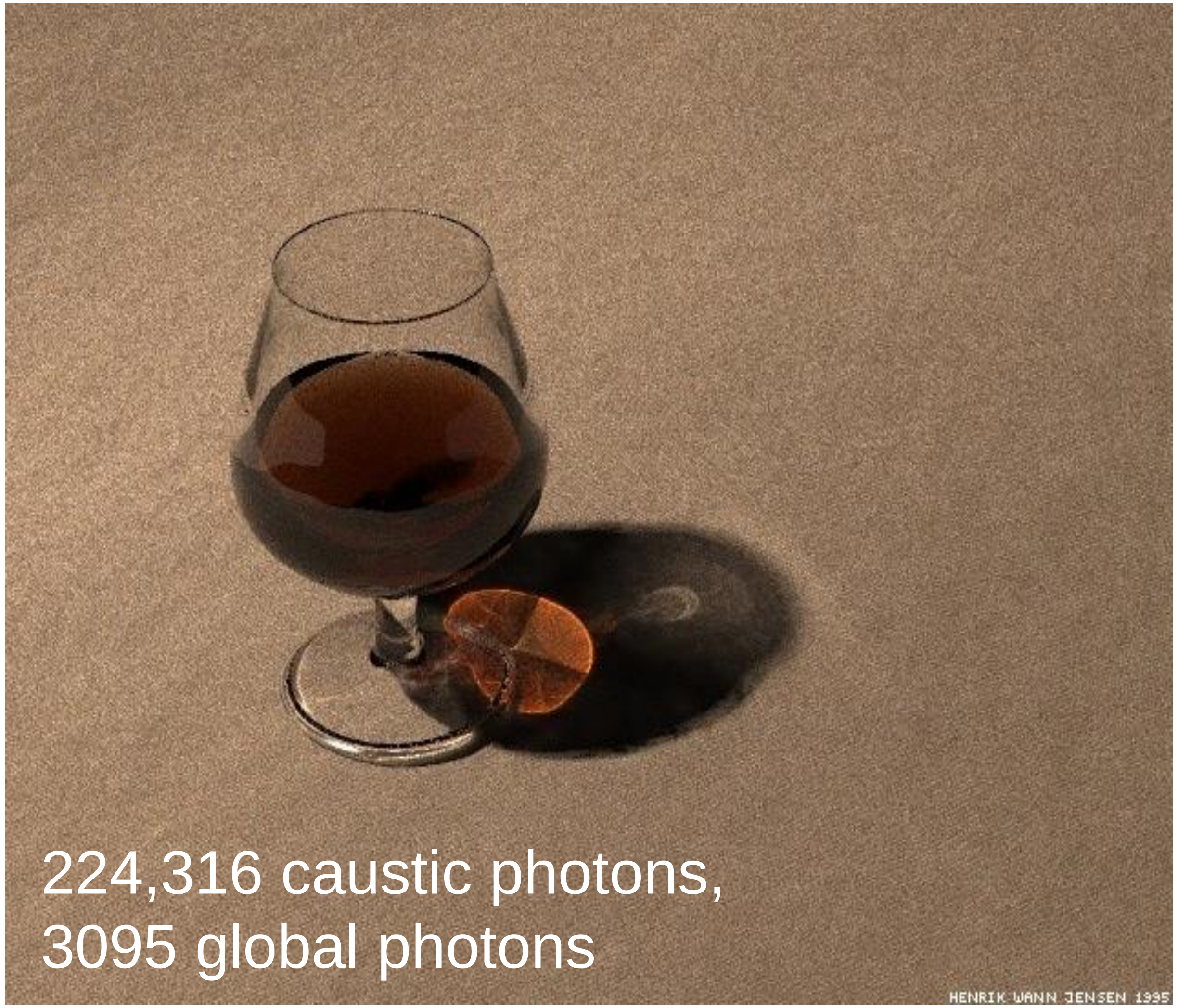
- Preprocessing step is view independent, so only needs to be re-done if the lighting or positions of objects change
- Inter-object interaction includes:
  - Shadows
  - Indirect lighting
  - Color bleeding
  - Highlights and reflections
  - Caustics – *current method of choice*
- Works for procedurally defined surfaces

Photon Mapping: Cons

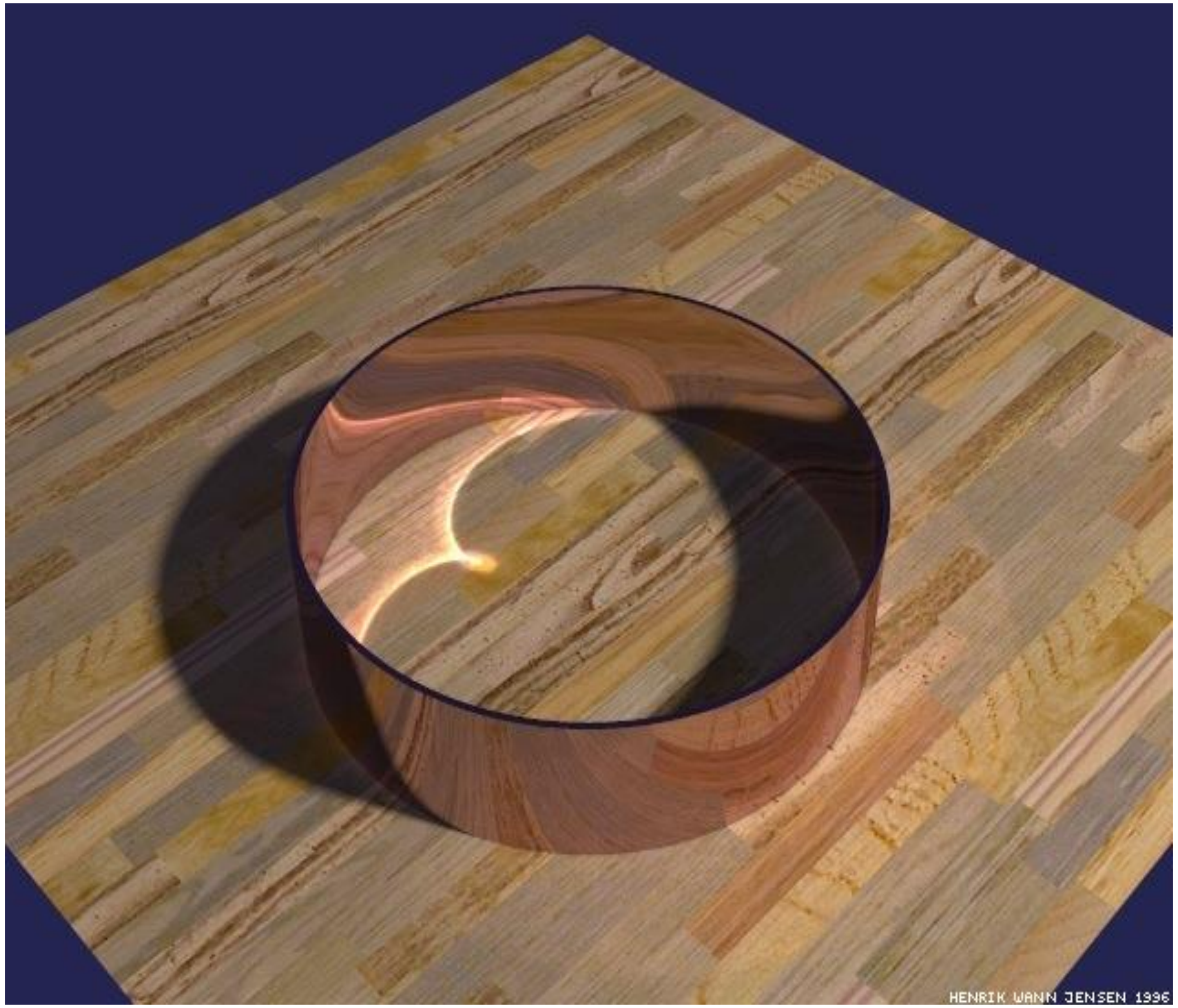
- Still time-consuming, although not as bad as comparable results from pure raytracing

- Photon map not easy to update if small changes are made to the scene

Photon Mapping Example


224,316 caustic photons,
3095 global photons

Photon Mapping Example



Summary

- Direct and Indirect Illumination
- Bidirectional Reflectance Distribution Function
- Raytracing and Radiosity
- Subsurface Scattering
- Photon Mapping

Extras

- Global Illumination for games / real-time:
https://docs.unity3d.com/Manual/GIIntro.html
    "baking light maps" – pre-compute GI for real-time applications for any static scene elements

- Ambient Occlusion:
  How exposed is each point in a scene to ambient lighting?
  (not the ambient term in Phong shading)

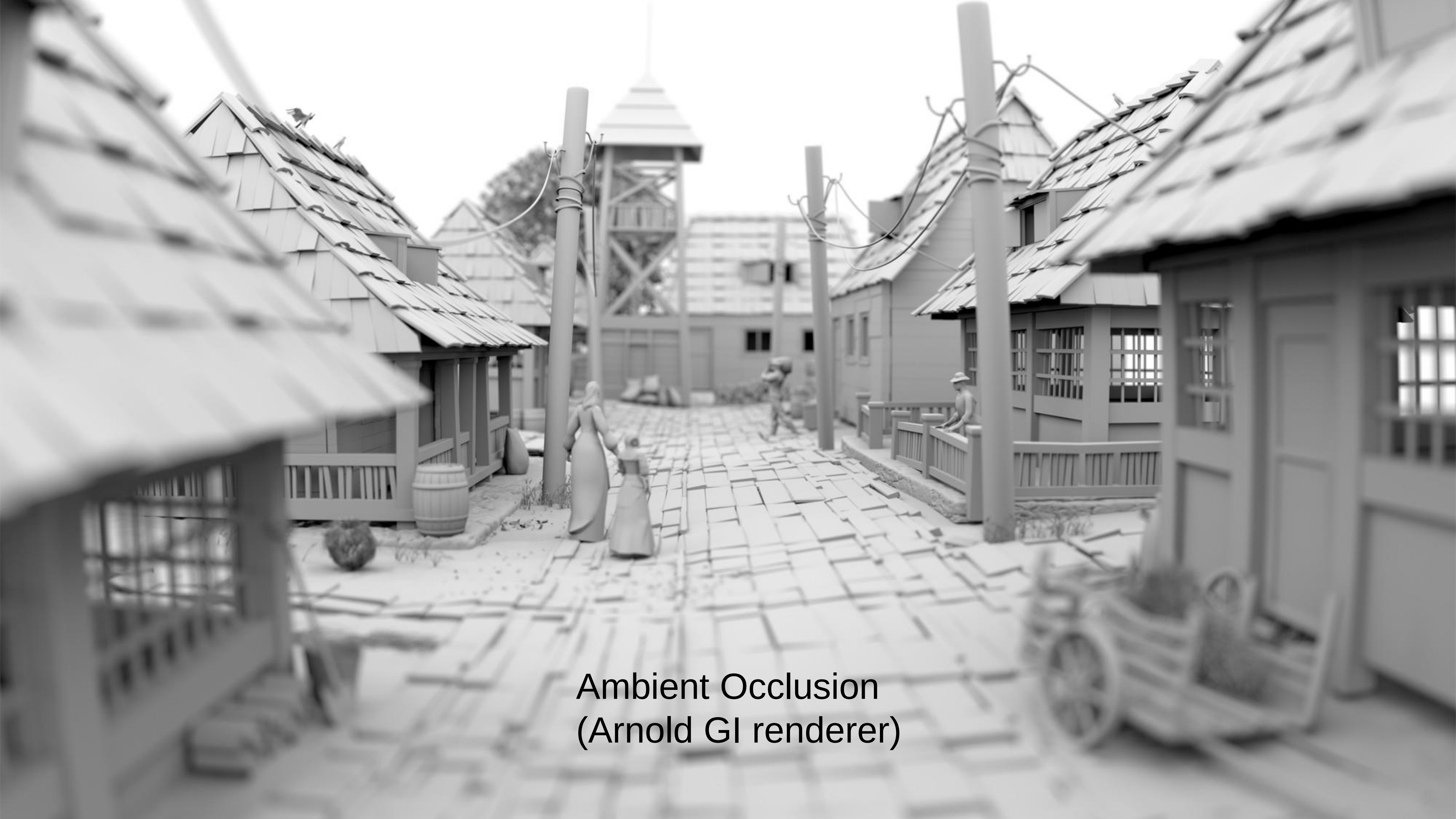
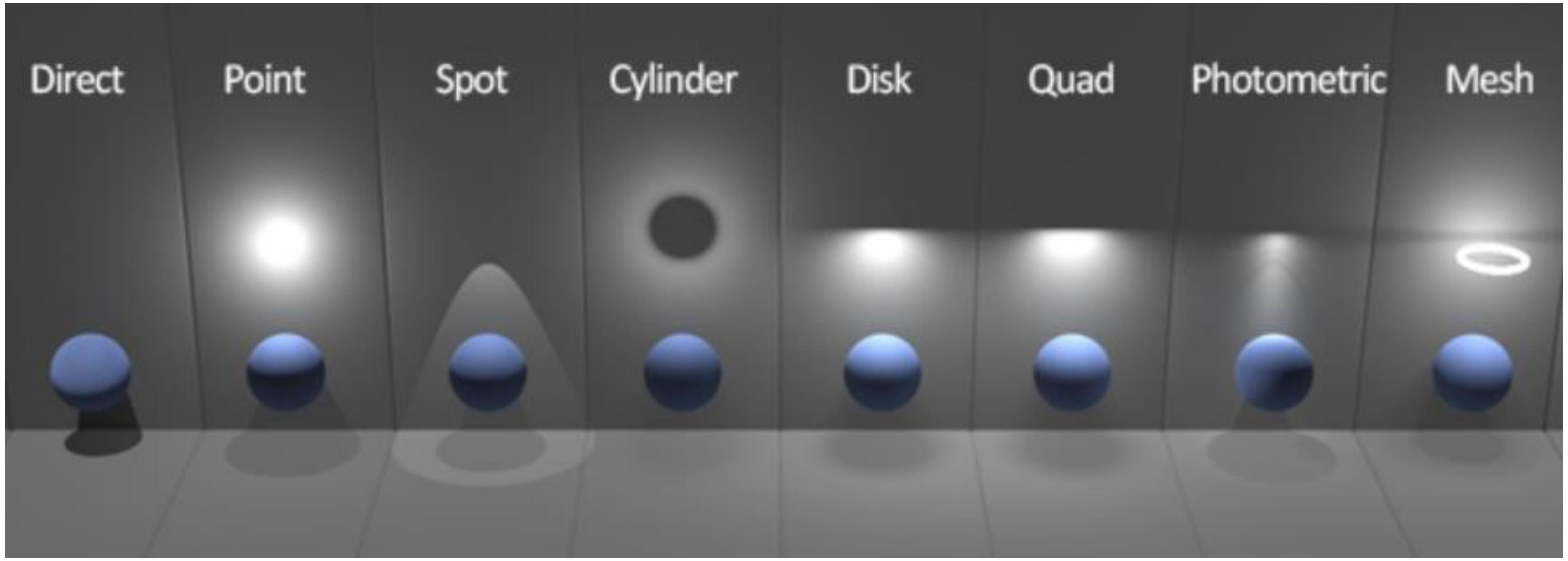

Ambient Occlusion
(Arnold GI renderer)

Image-based Lighting

Image-based Lighting
(Photographs capture
real-world light, which
can be used for rendering)

[Paul Debevec, USC ICT]

GI Light Types



[Arnold renderer]