199: Natural world and CG: rendering

Karan Singh

dgp
Dynamic Graphics Project
University of Toronto
www.dgp.toronto.edu

Computer Graphics: the trinity

- Modeling:
  How do we represent (2D or 3D) objects & environments?
  How do we build these representations?

- Animation:
  How do we represent the way objects move?
  How do we define & control their motion?

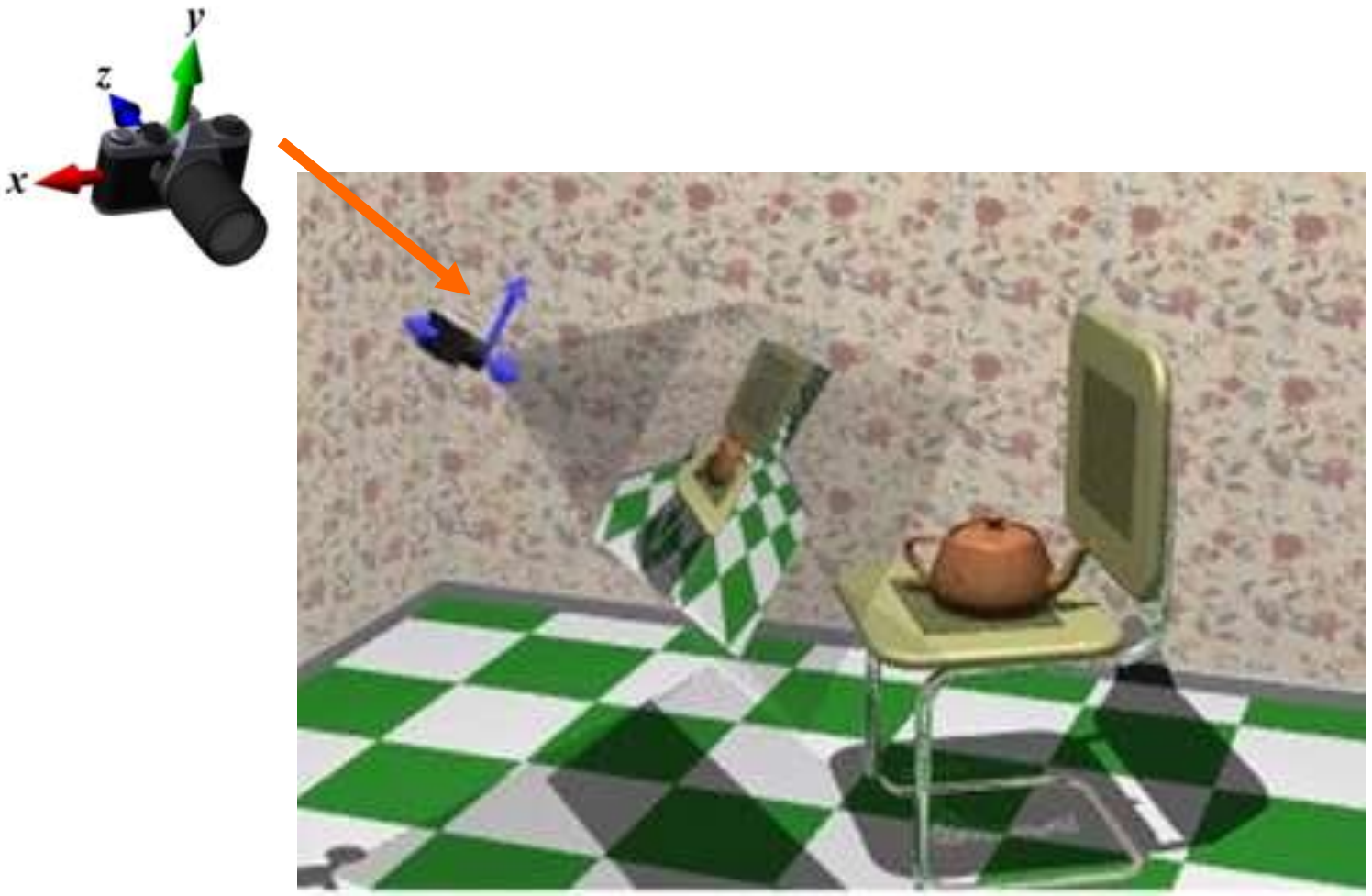
- Rendering:
  How do we represent the appearance of objects?
  How do we simulate the image-forming process?

bunny

Rendering: How?

Light emanating from an emissive source, bounces around objects, entering, refracting, absorbing, scattering, reflecting and eventually enters the eye or camera, where it excites a photosensitive region on an image to define a color.

Camera model

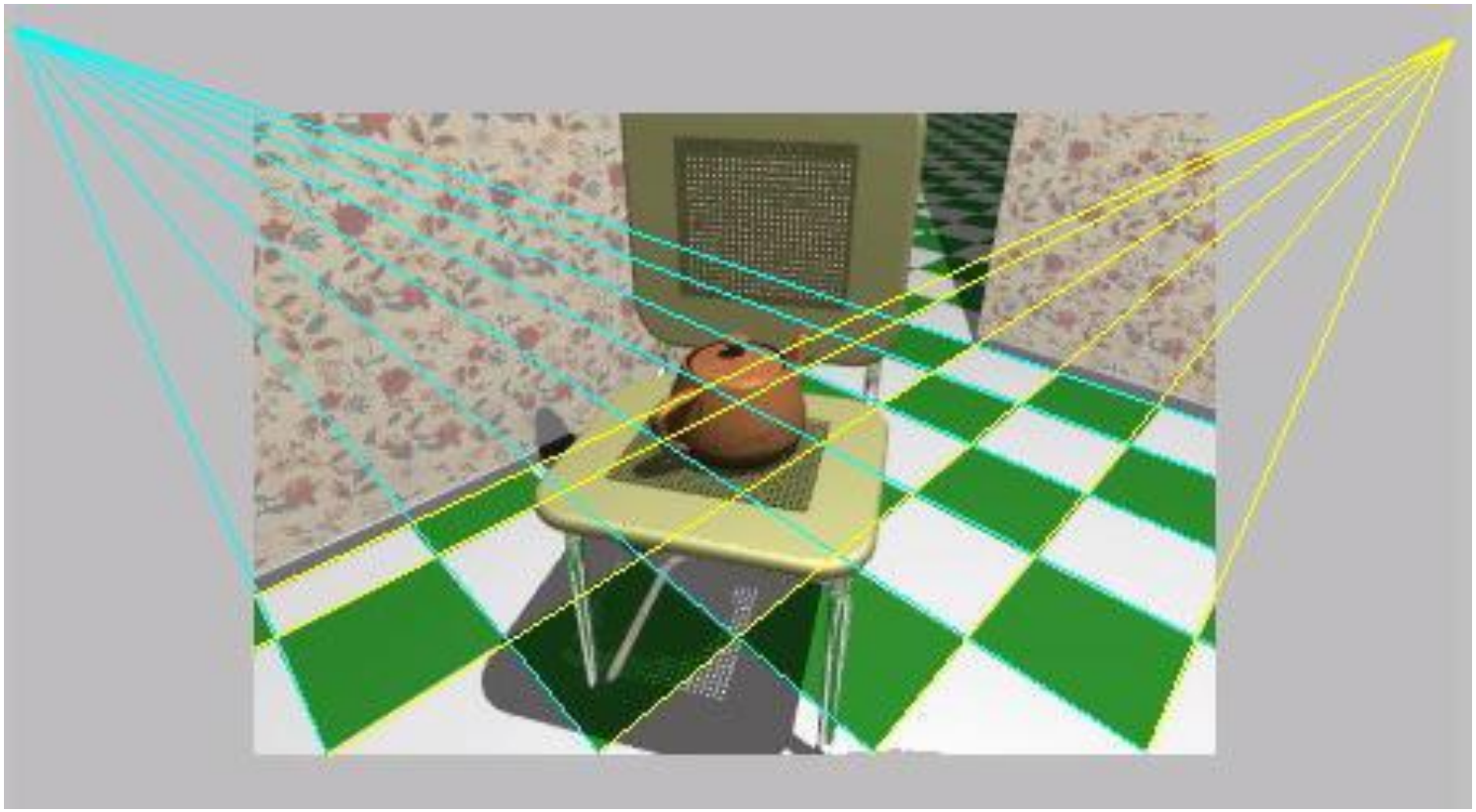


Camera model

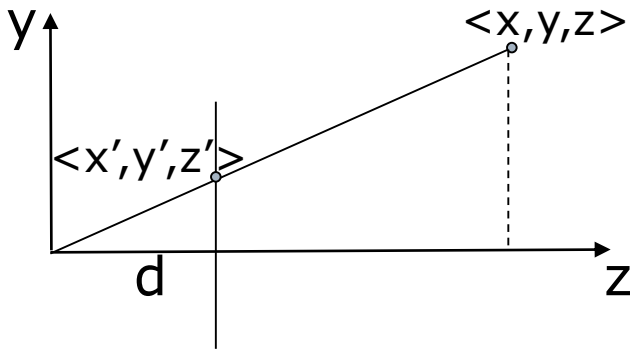
What is the difference between these images?

Camera model

Perspective Projection



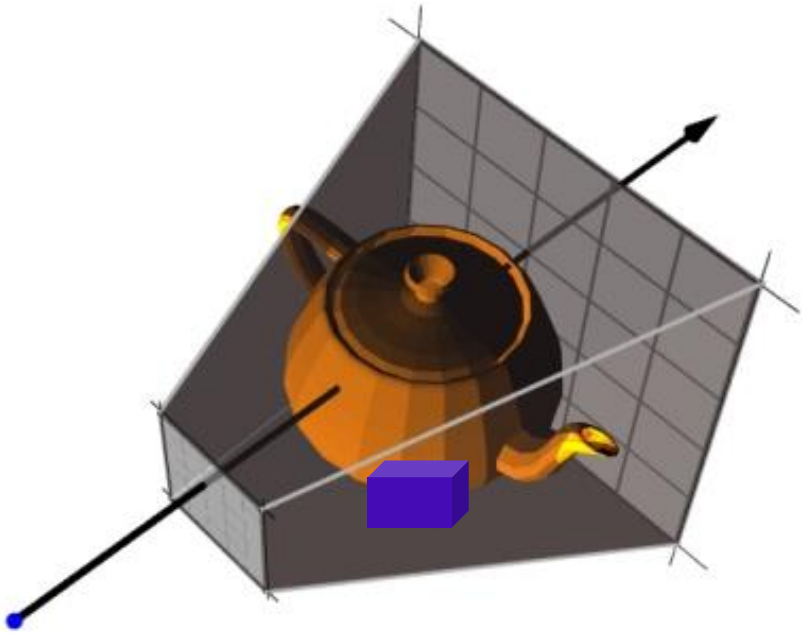
Simple Perspective



- **y′= yd/z**
- **x′= xd/z**
- **z′=d**

Visibility Problem

- **What is NOT visible?**



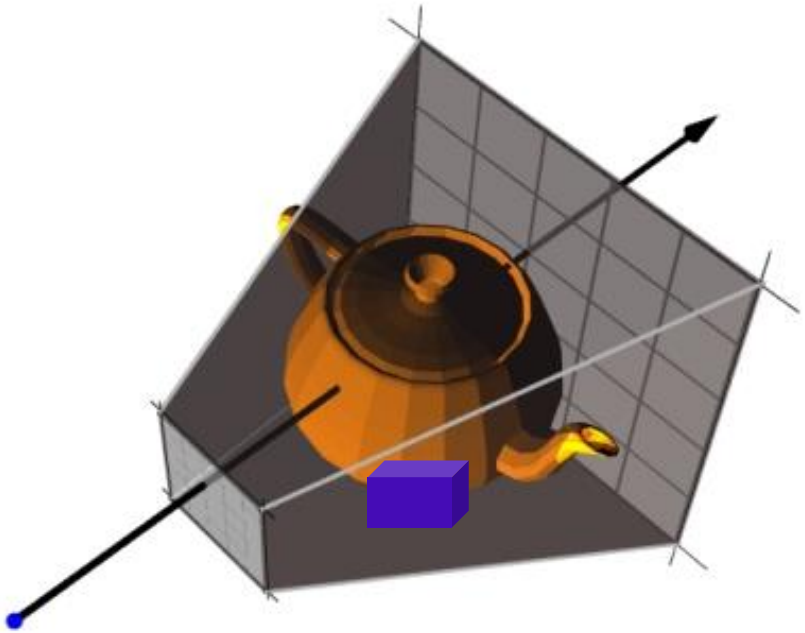
Visibility Problem

- **What is NOT visible?**

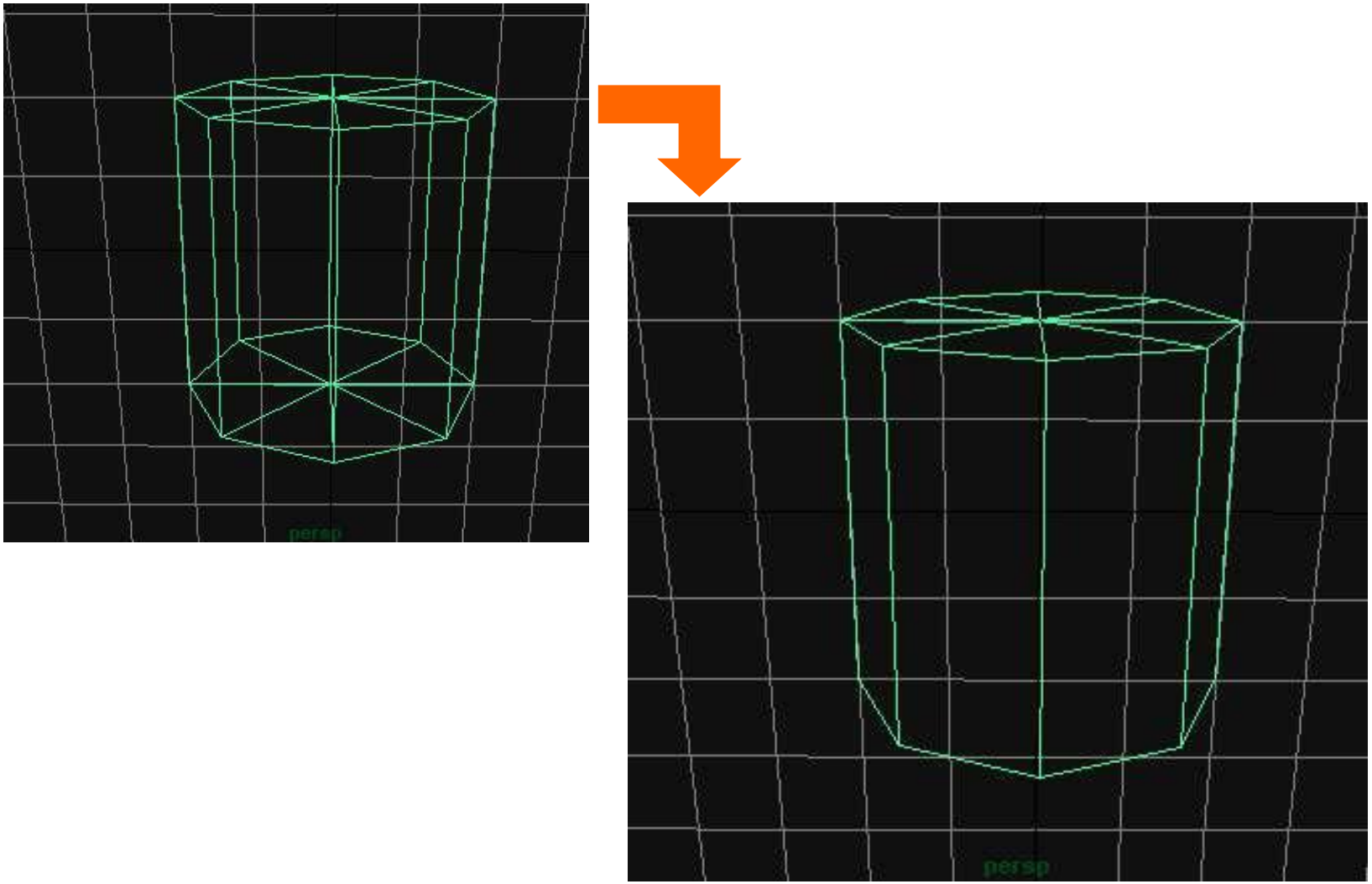
primitives outside of the field of view

back-facing primitives

primitives occluded by other objects closer to the camera

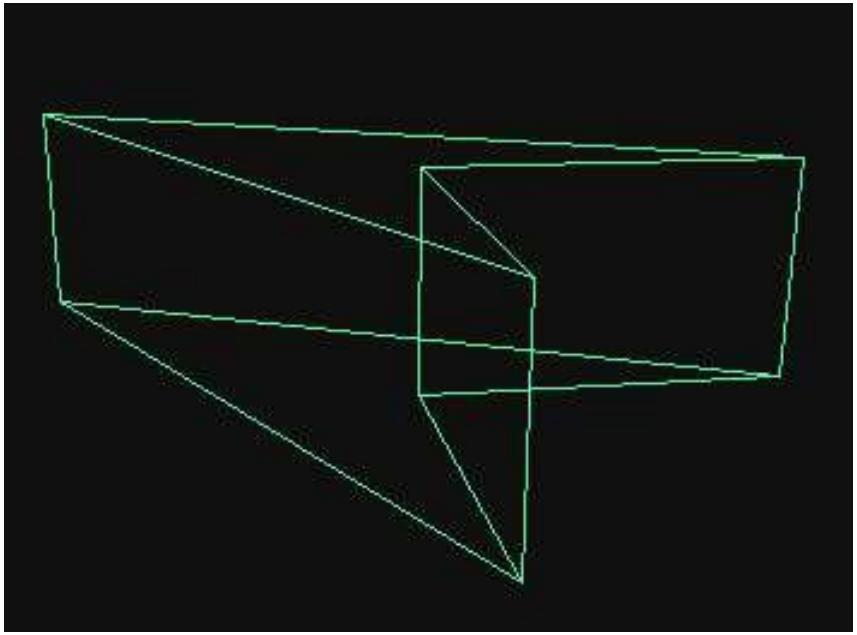


Backface culling



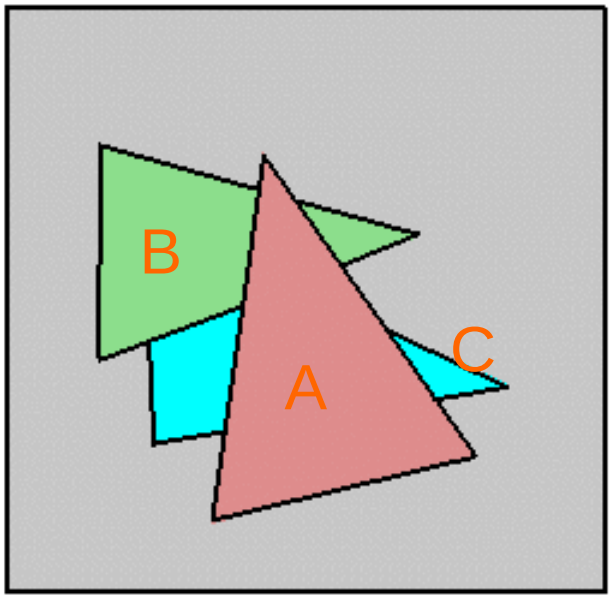
Occluded faces

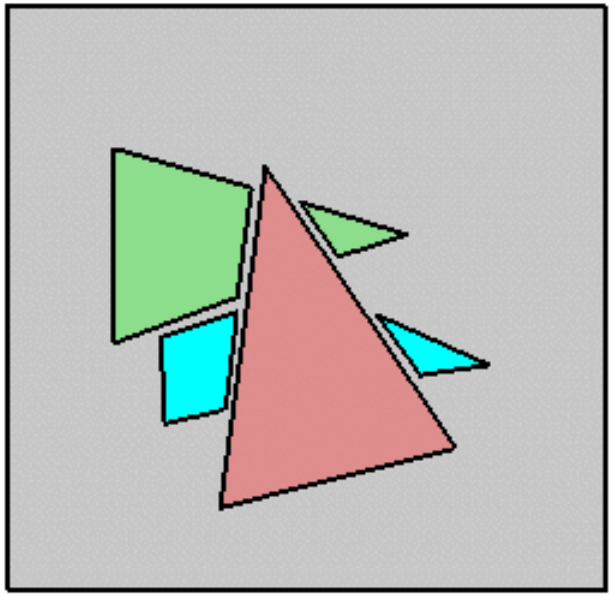
Does backface culling always determine visibility completely for a single object?



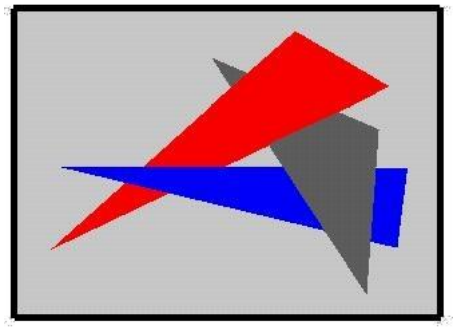
Painters Algorithm

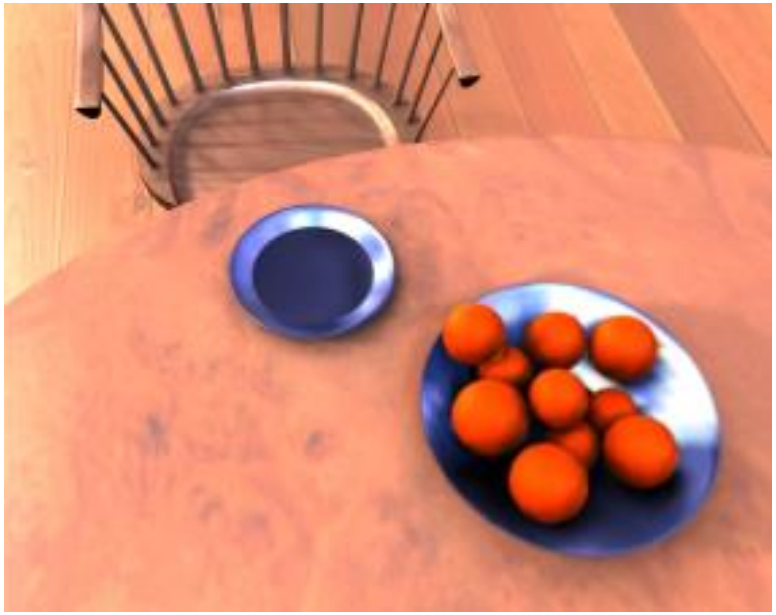
- **Sort primitives in Z.**
- **Draw primitives back to front (CBA).**

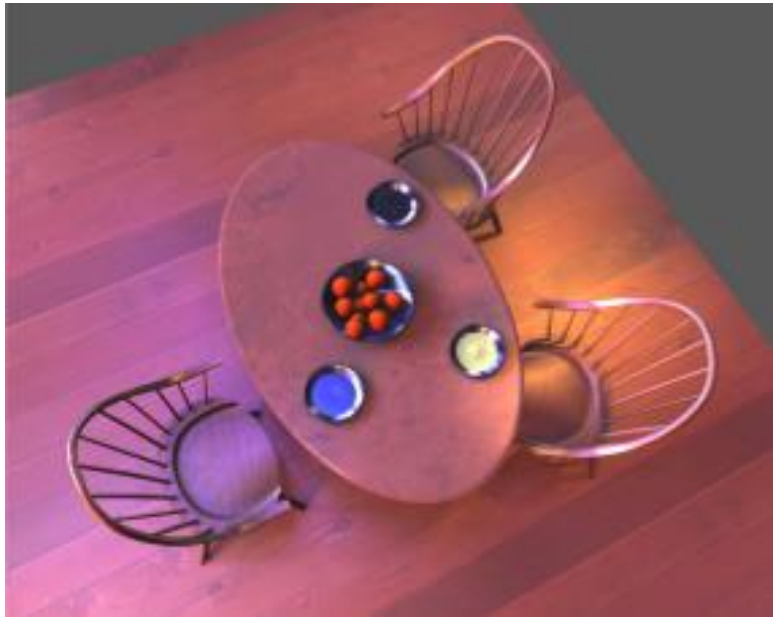




Painters Algorithm Problem





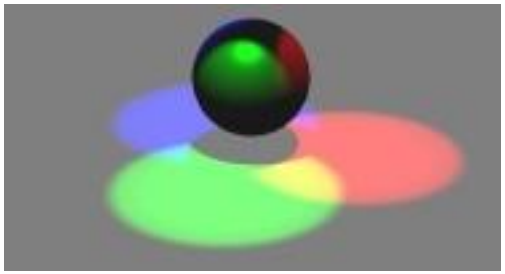


Terminology

- **Illumination**
  - The transport of luminous flux from light sources between points via direct and indirect paths
- **Lighting**
  - The process of computing the luminous intensity reflected from a specified 3-D point
- **Shading**
  - The process of assigning a color to a pixel
- **Illumination Models**
  - Simple approximations of light transport
  - Physical models of light transport

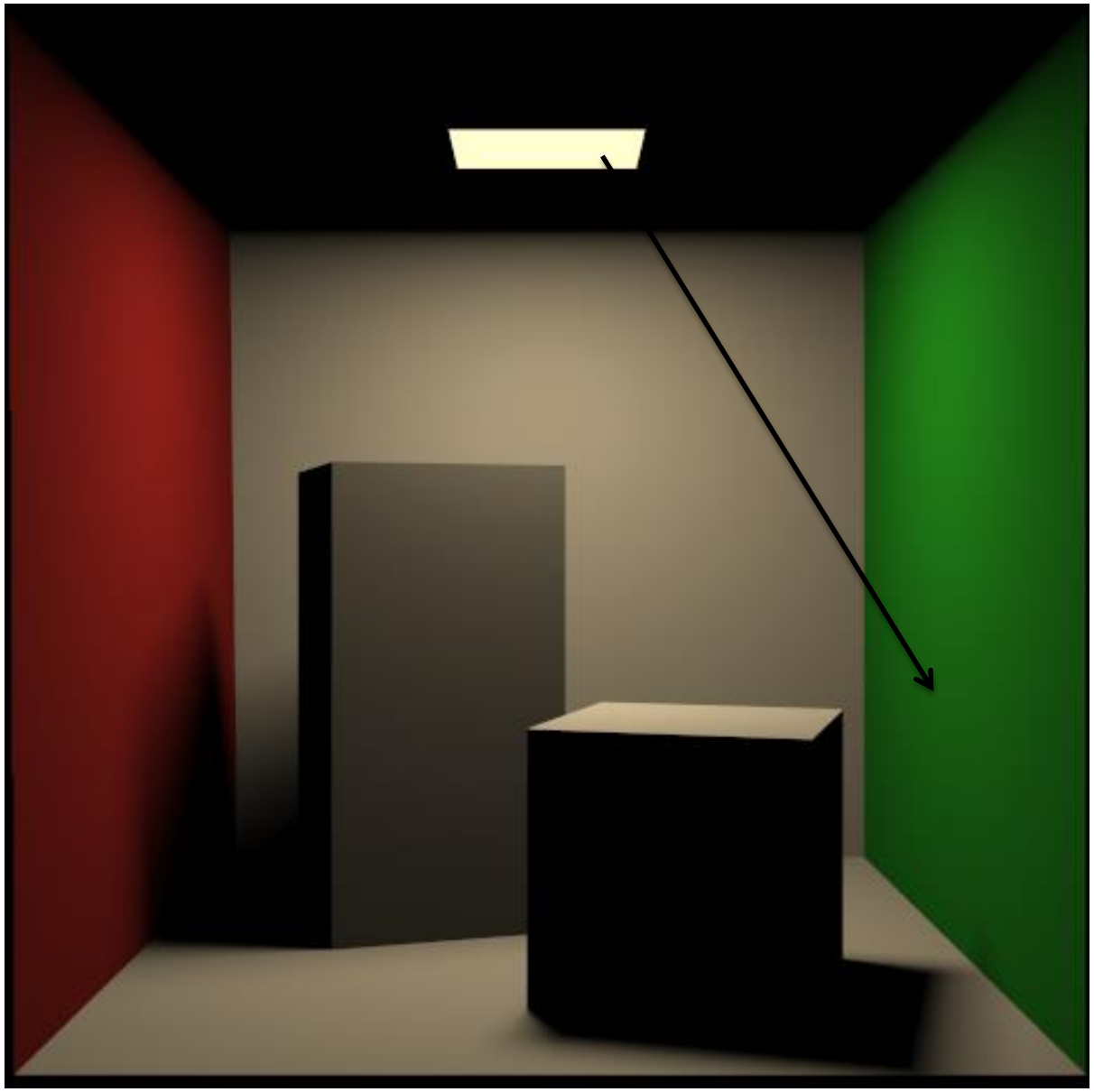
Two Components of Illumination

- **Light Sources**
  - Emission Spectrum (color)
  - Geometry (position and direction)
  - Directional Attenuation

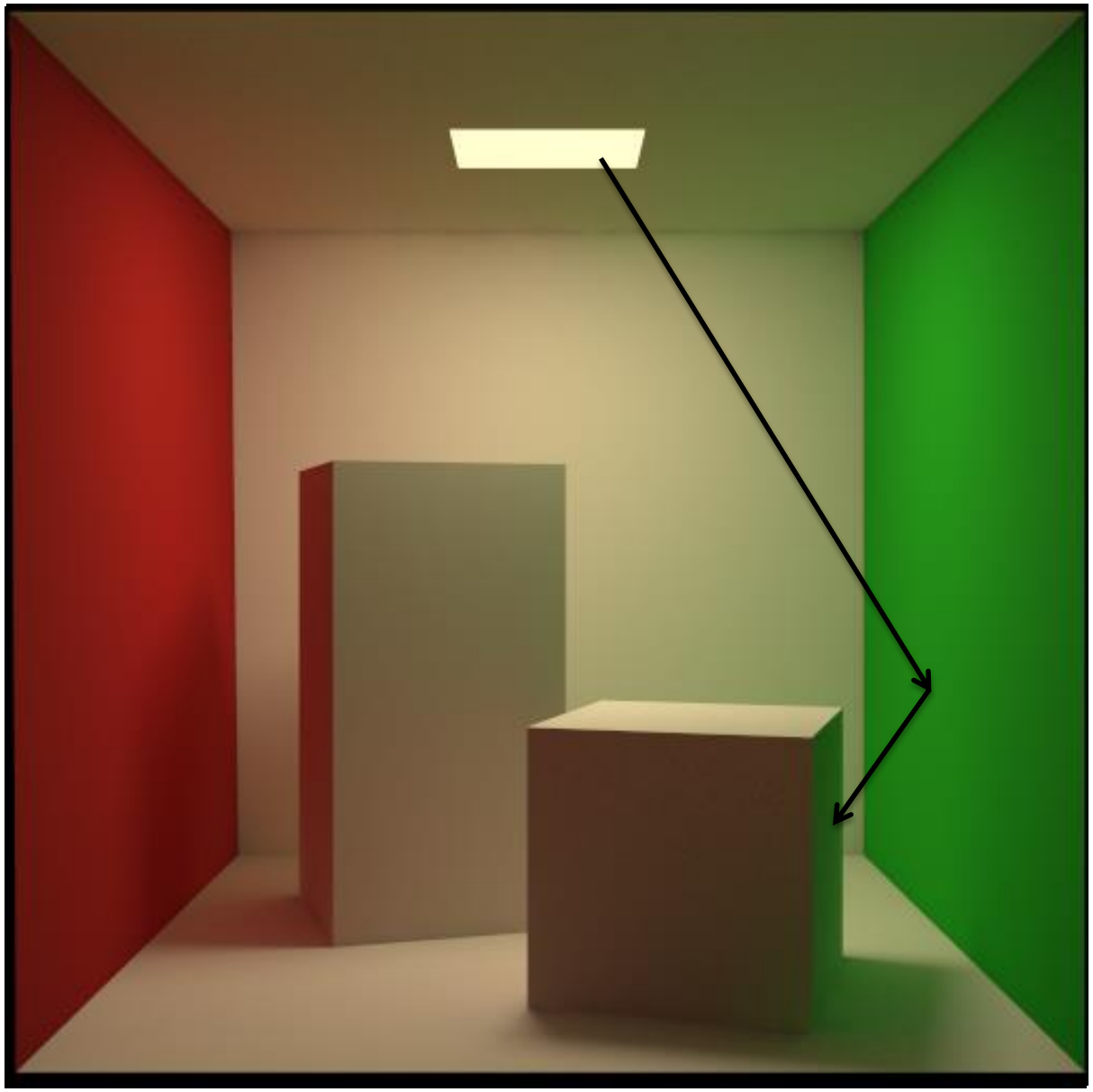


- **Surface Properties** (Reflectors)
  - Reflectance Spectrum (color)
  - Geometry (position, orientation, and micro-structure)
  - Absorption
  - Transmission

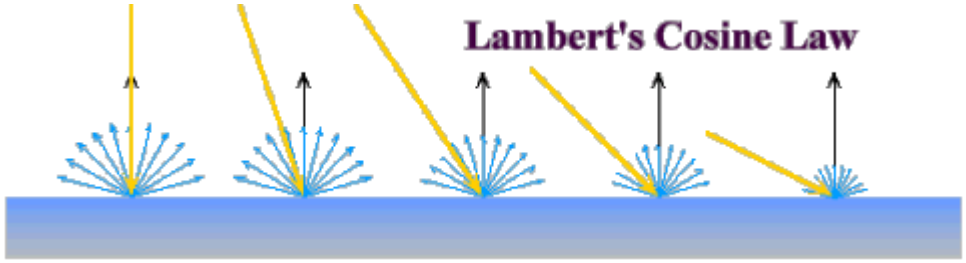
Area Light Source: Direct Lighting



Area Light Source: Indirect Lighting



Lambert's Cosine Law



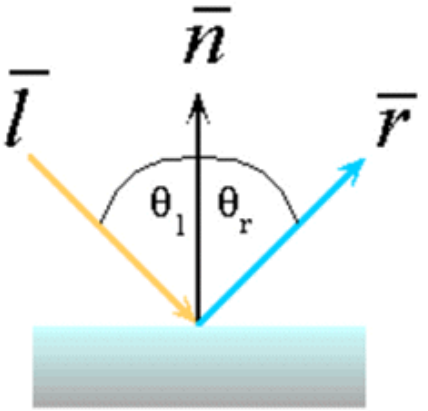
Ideal diffuse reflectors reflect light according to *Lambert's cosine law*, Lambert's law states that the reflected energy in a particular direction is proportional to cosine of the angle between that direction and the surface normal.

Snell's Law

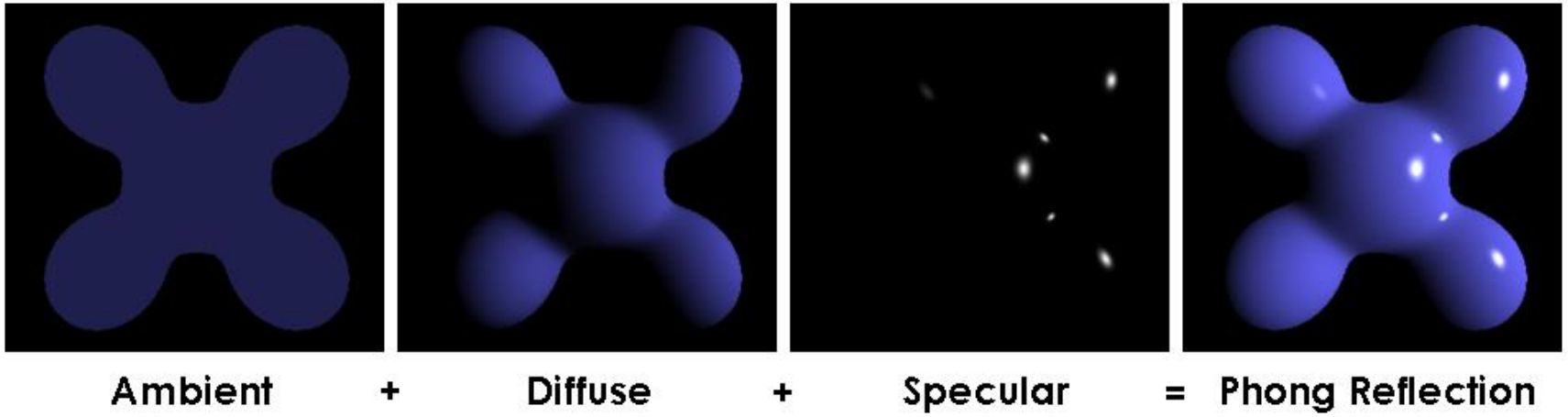
- The incoming ray, the surface normal, and the reflected ray all lie in a common plane.

- The angle that the reflected ray forms with the surface normal is determined by the angle that the incoming ray forms with the surface normal, and the relative speeds of light of the mediums in which the incident and reflected rays propagate according to the following expression.
(Note: n_l and n_r are refractive indices).

$$n_l \sin\theta_l = n_r \sin\theta_r$$



Phong Illumination: general equation



$$I_o(N,V,L) = k_a I_a + I_i [\, k_d \, max(0, N.L) + k_s \, max(0,(V.R)^n)]$$

Flat Shading

- Apply only one illumination calculation for each face.

- Which point on the facet do we illuminate?
- Pros?
- Cons?

Gouraud Shading

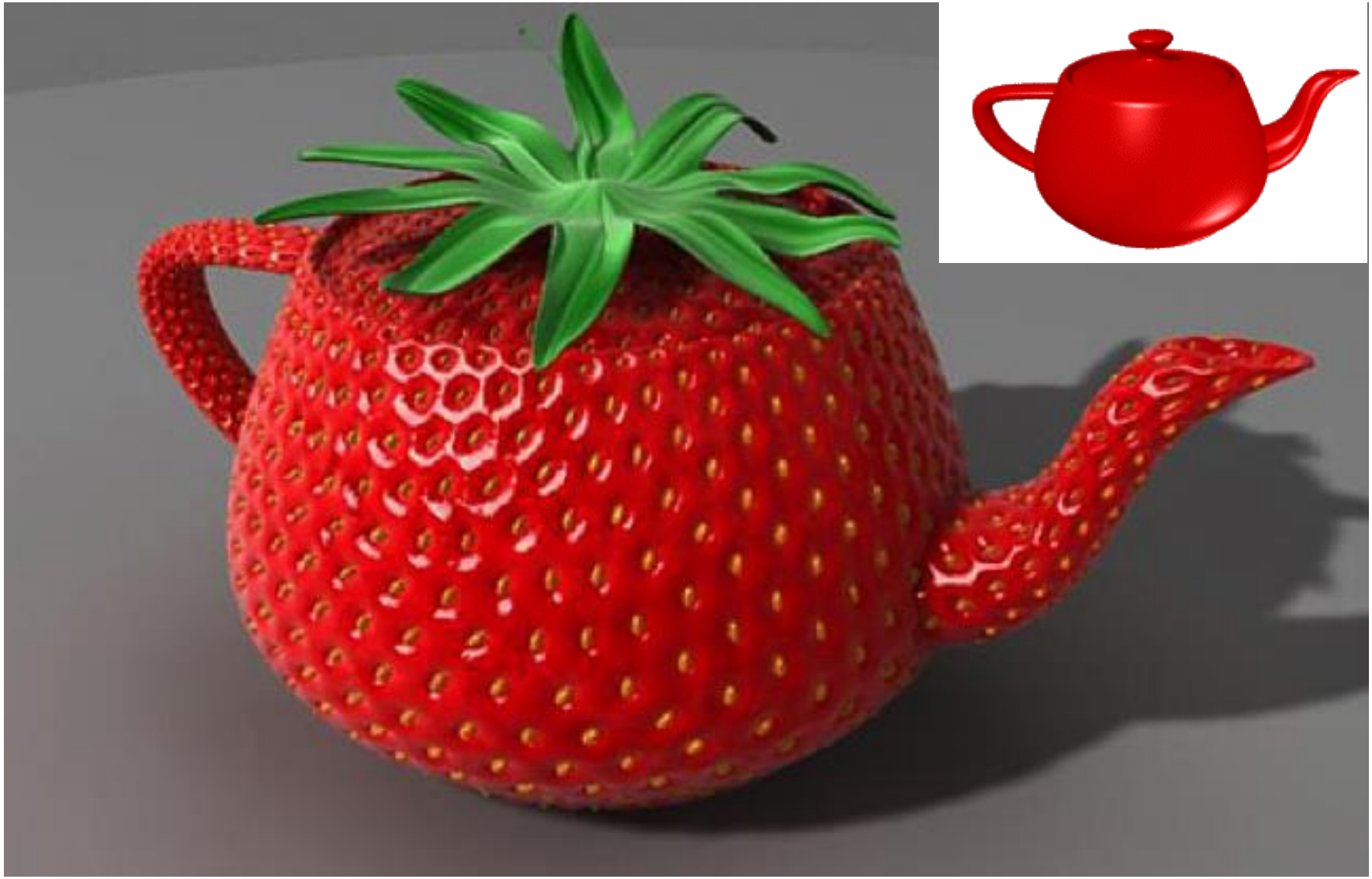
- Apply the illumination model at vertices and interpolate the color intensity across faces.

- Remember bilinear interpolation?
- Pros?
- Cons?

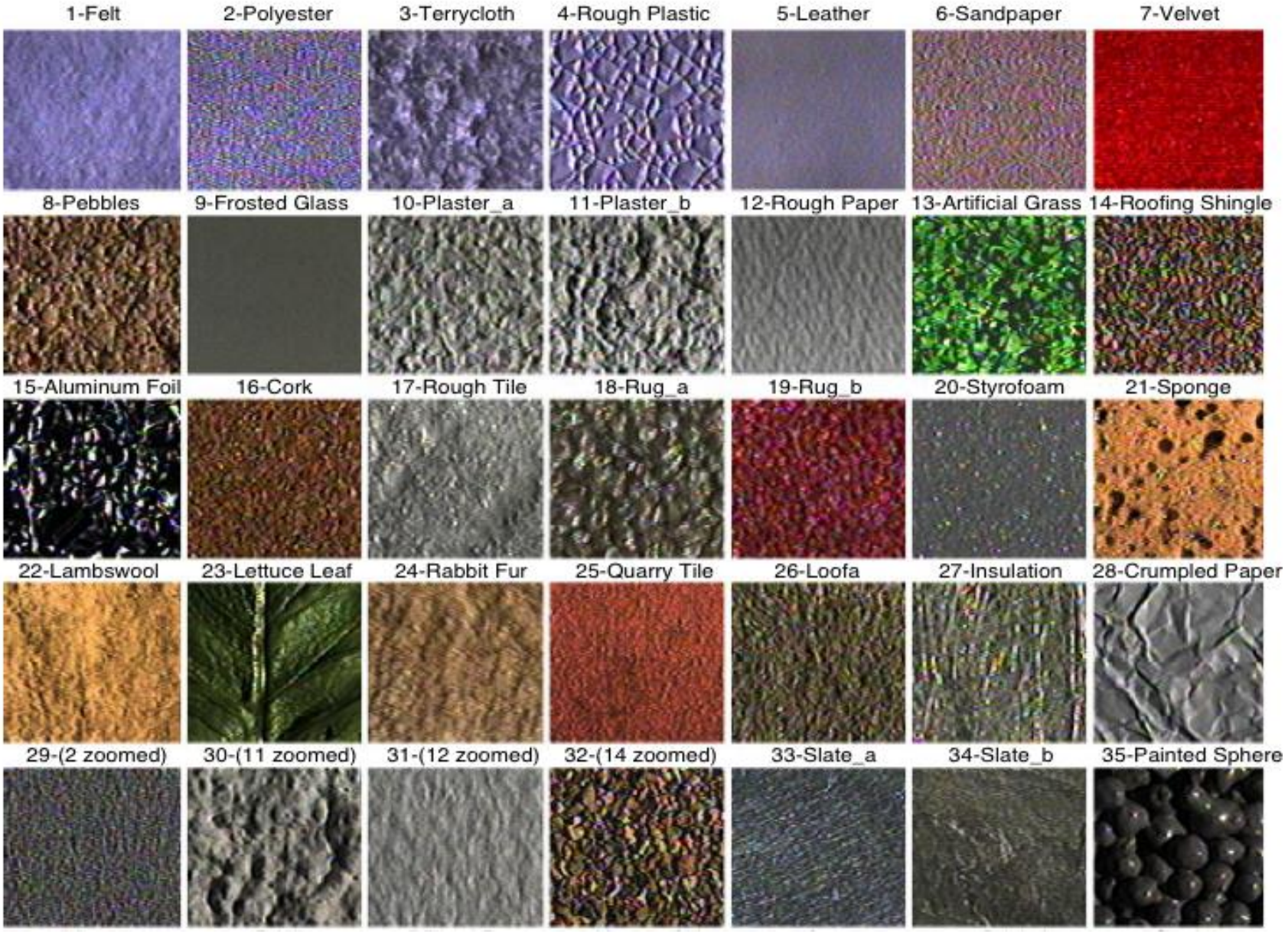
Phong Shading

- Not to be confused with Phong Illumination model.
- Apply the illumination model at every point on the face.
- Calculate the normal at any point in a face by interpolating the vertex normals.of that face.

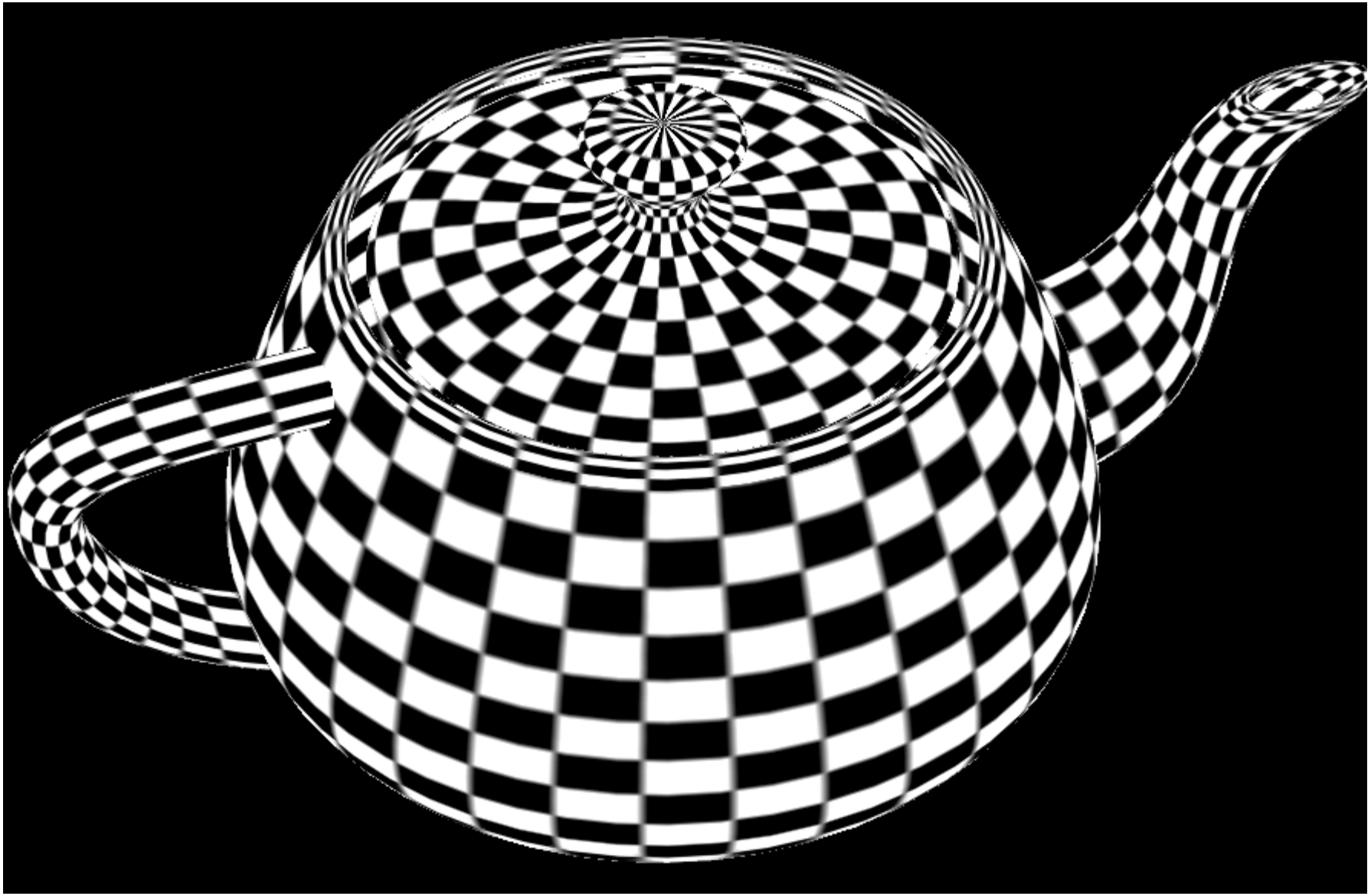
- Pros?
- Cons?
- Silhouettes?



Photographs



Procedural

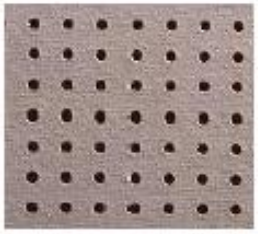


Solid textures

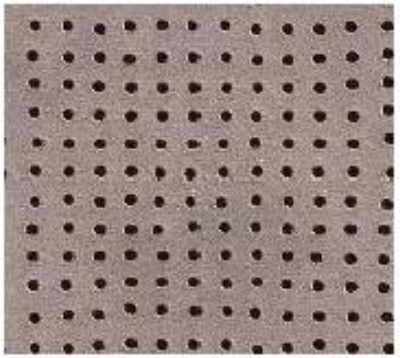
Synthesized

(e)

(f)



(g)



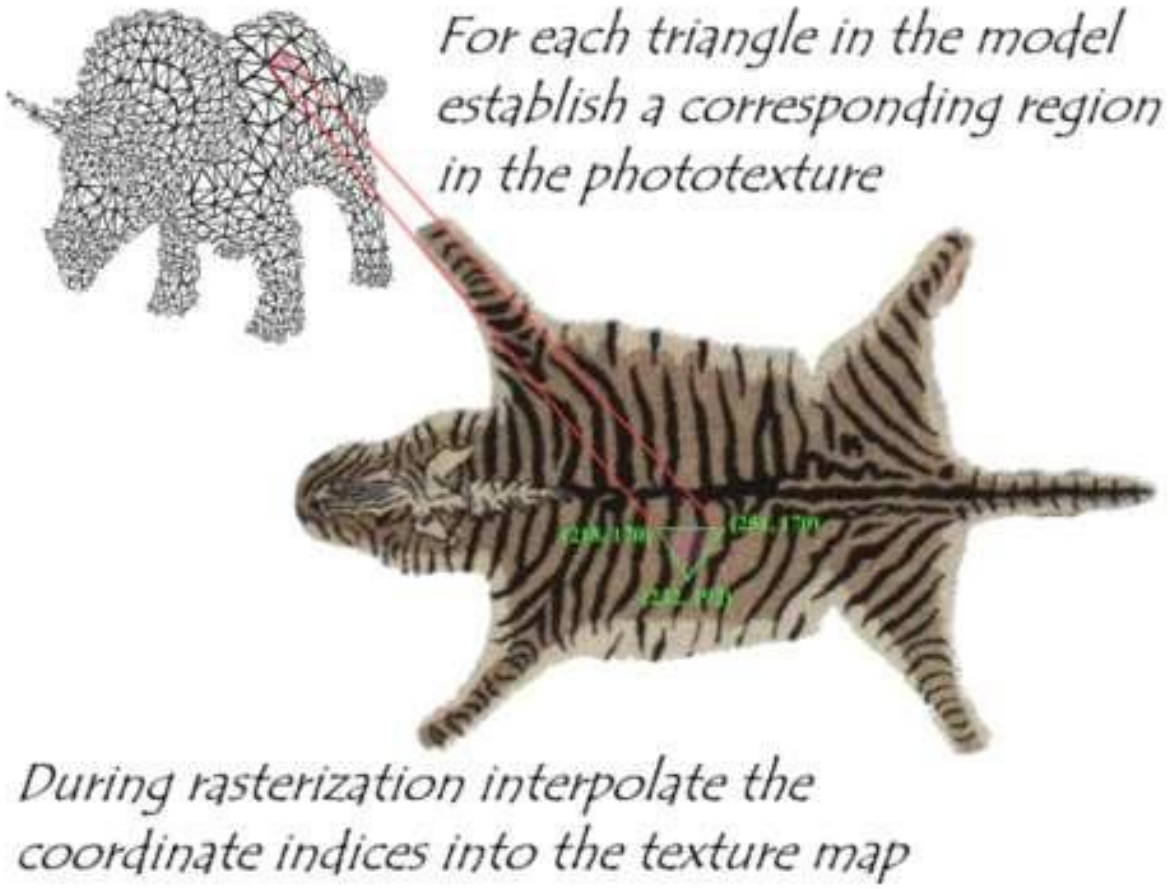
(h)

(i)

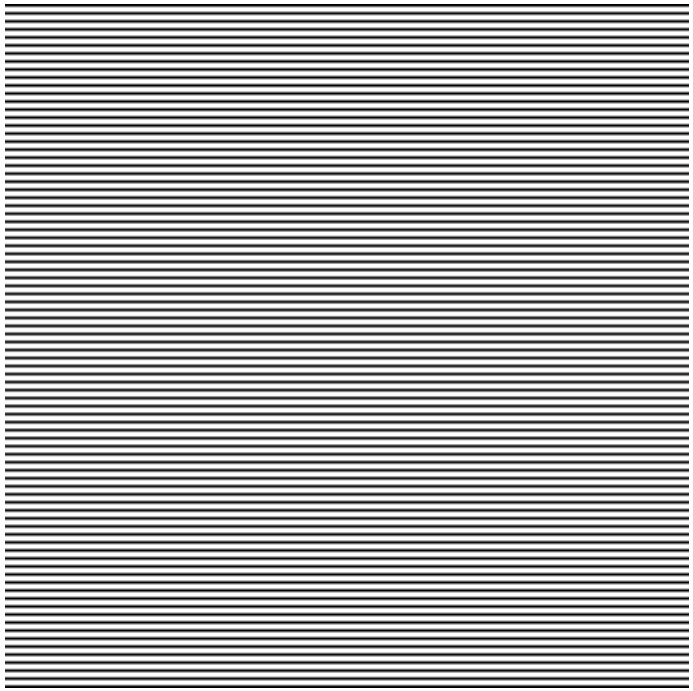
(j)

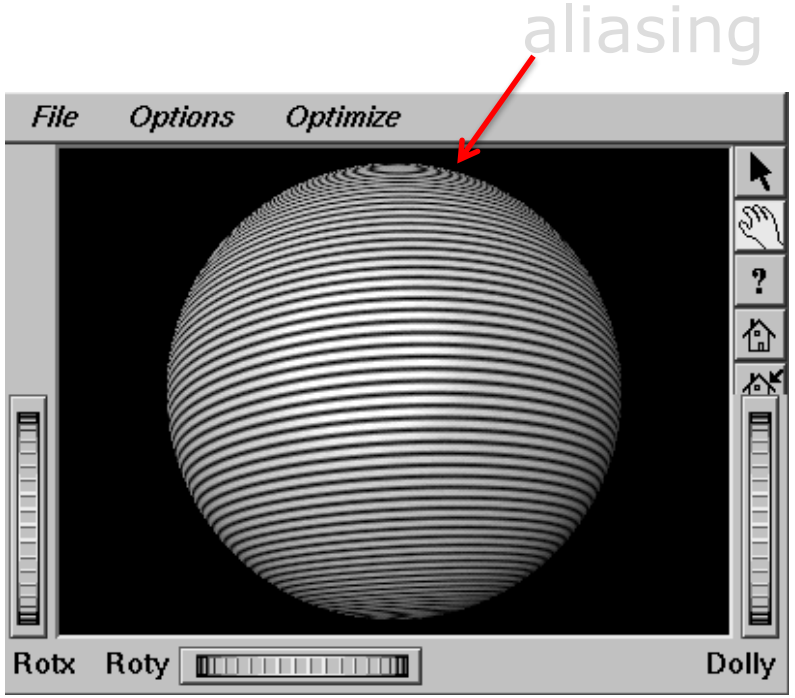
Texture coordinates

How does one establish correspondence? (UV mapping)

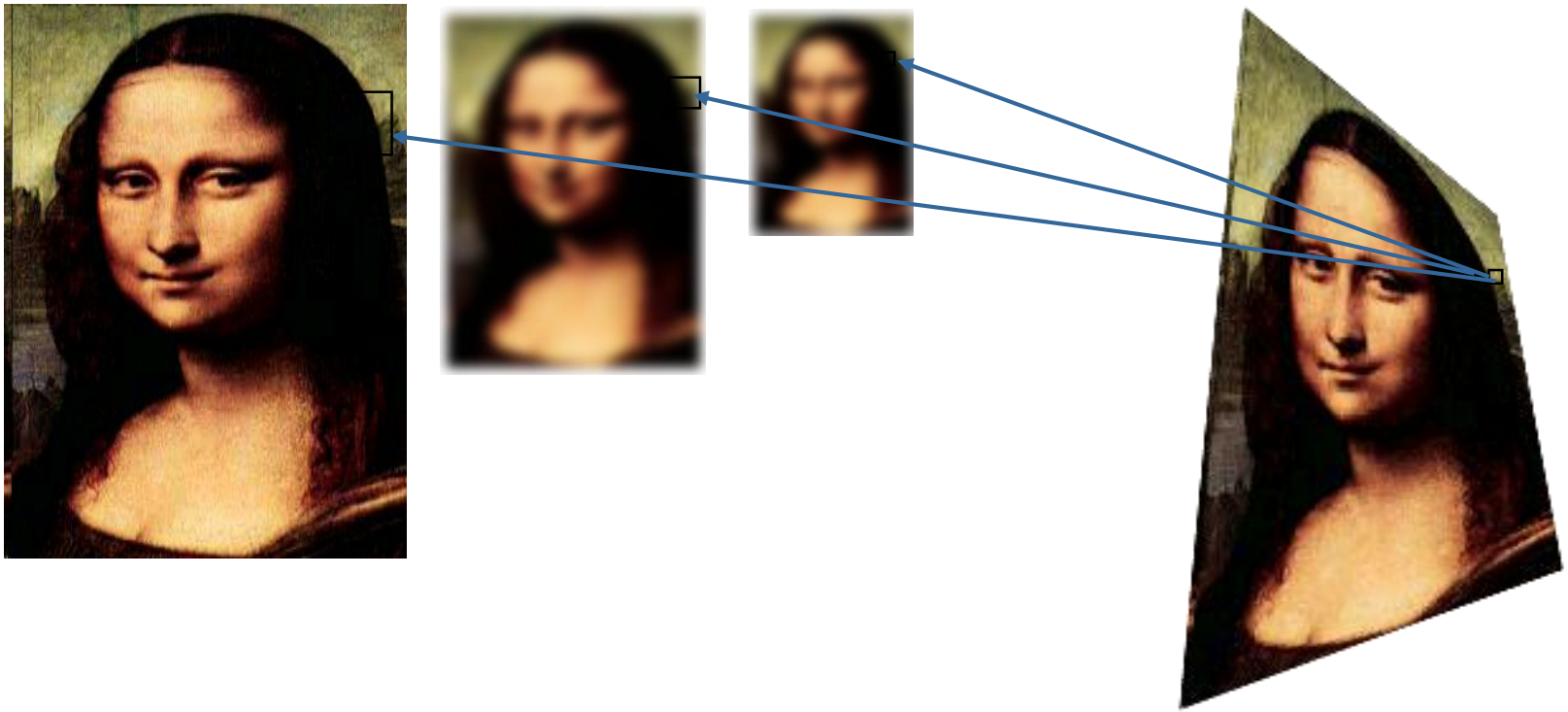


Aliasing During Texture Mapping



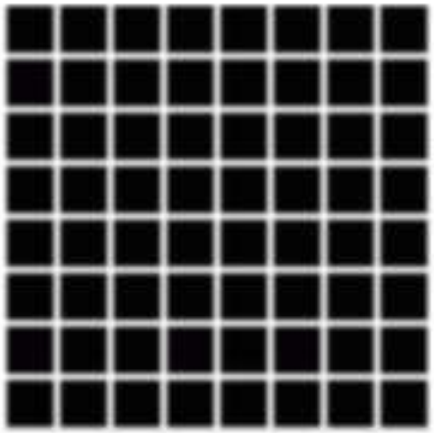


MIP-Mapping: Basic Idea



Given a polygon, use the texture image, where the projected polygon best matches the size of the polygon on screen.

Bump mapping



Bump Map

Categories of light transport

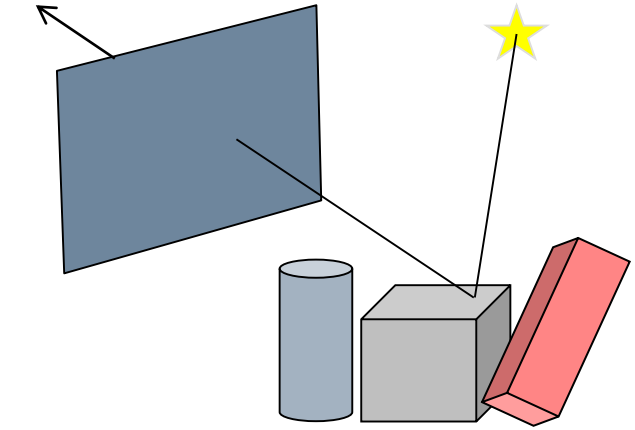
- Specular-Specular
- Specular-Diffuse
- Diffuse-Diffuse
- Diffuse-Specular

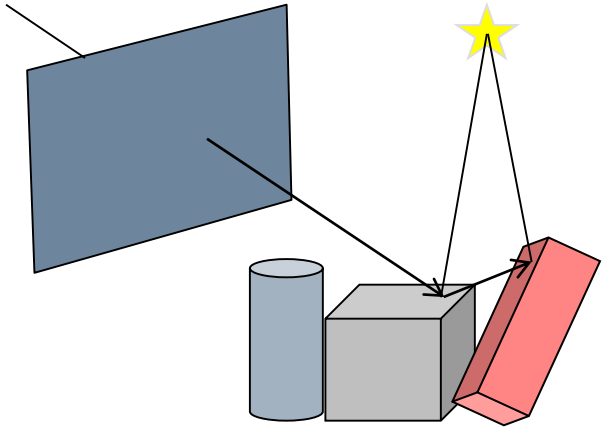
Ray Tracing

- Traces path of specularly reflected or transmitted (refracted) rays through environment
- Rays are infinitely thin
- Don’t disperse
- Signature: shiny objects exhibiting sharp, multiple reflections

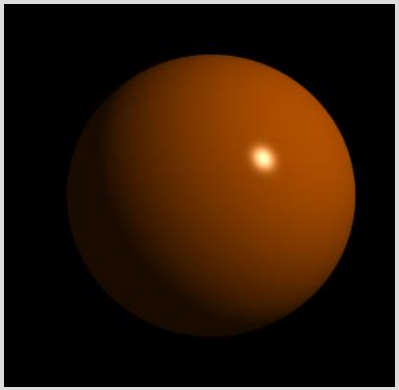
- Transport E - S – S – S – D – L.

Ray Tracing: Basic Idea

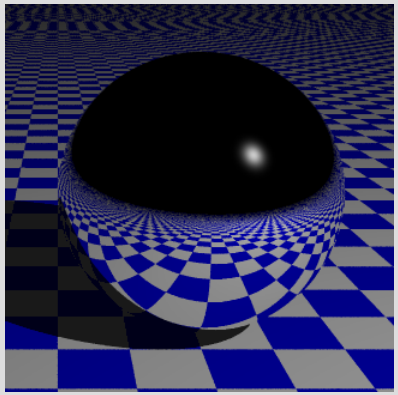




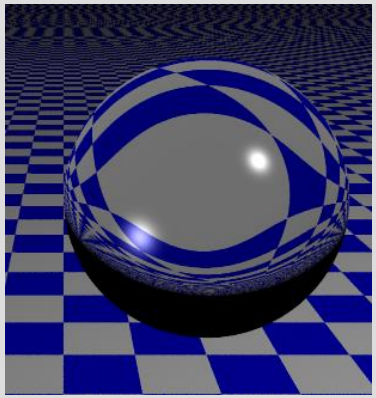
local illumination

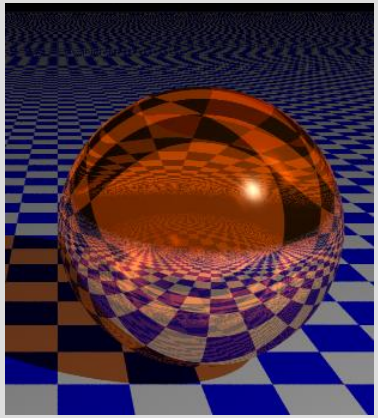


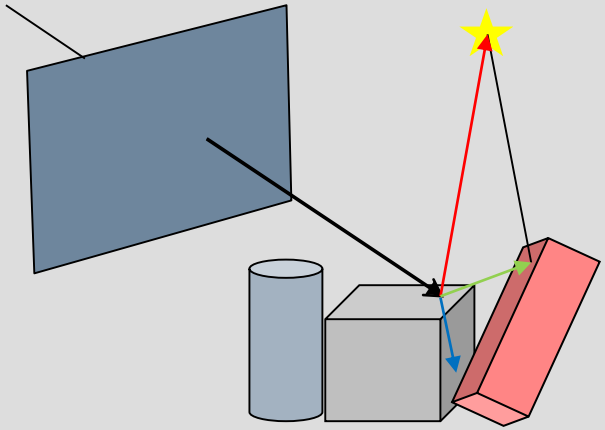
reflection



refraction







Ray Tracing: Basic Algorithm

For each pixel *q*
{
  compute ***r***, the ray from the eye through ***q***;
  find first intersection of ***r*** with the scene, a point ***p***;
  estimate light reaching ***p****;*
  estimate light transmitted from ***p*** to ***q*** along ***r***;
}

Ray Tracing Imagery

Ray Tracing Improvements: Caustics

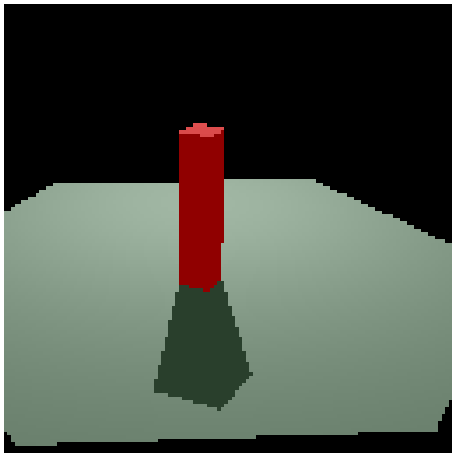
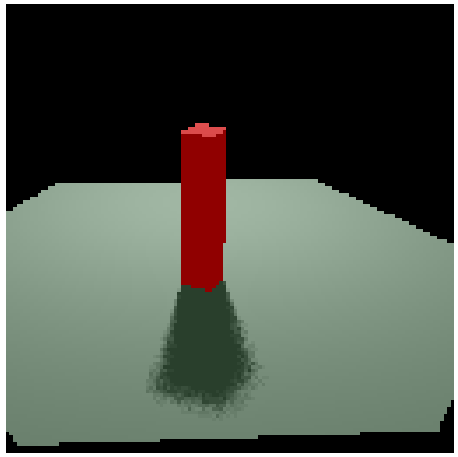
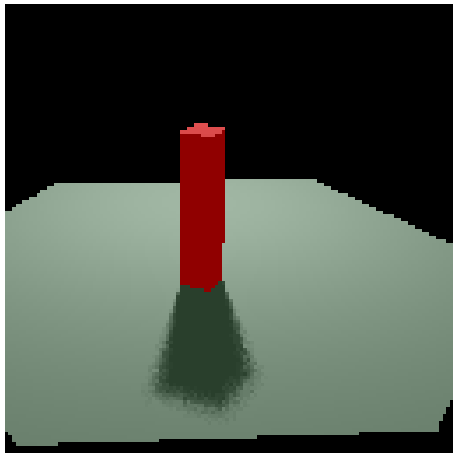
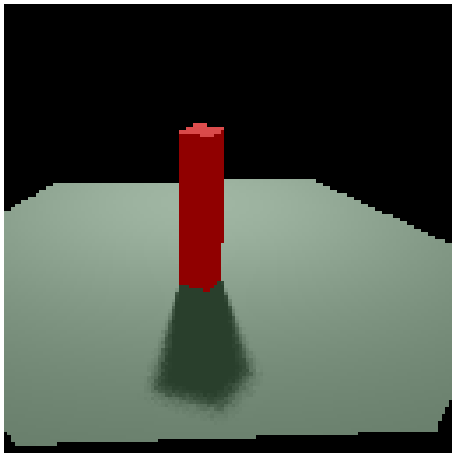
How many rays do you need?

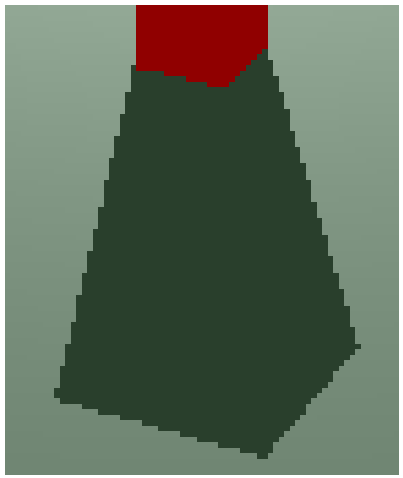
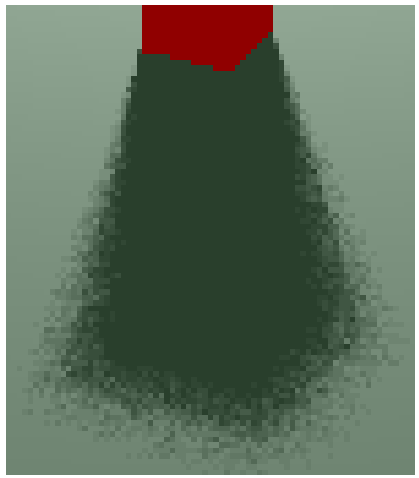
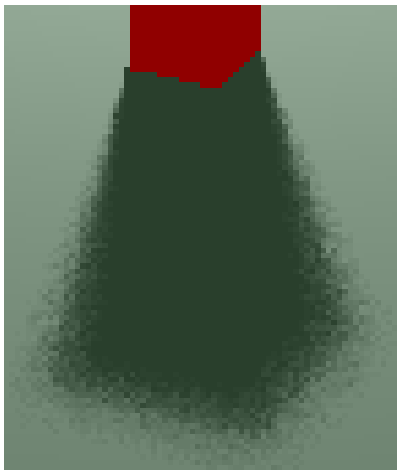
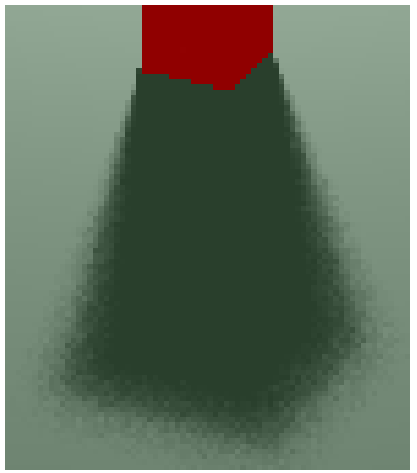
1 ray/light

10 ray/light

20 ray/light

50 ray/light

Ray Tracing vs. Radiosity

Subsurface scattering

Diffraction

Next: Interaction