Chapter 12

Ray-Tracing

Global Illumination Models
- Simple shading methods simulate local illumination models
  - No object interaction
- To simulate global illumination models need more sophisticated & more computation-intensive algorithms
- Ray-tracing deals with
  - Reflectivity
  - Transparency
  - Shadows

Ray-Tracing Algorithm

Reflection and Refraction

Basic Ray-Tracing Algorithm

Sub-Routines
- ReflectRay(r,obj) – computes reflected ray (use obj normal at intersection)
- RefractRay(r,obj) - computes refracted ray
  - Note: ray is inside obj
- Shade(reflect_color,refract_color,obj) – compute illumination given three components
**Ray-Object Intersections**

- Kernel of ray-tracing ⇒ must be extremely efficient

- Usually involves solving a set of equations

**Example:** Ray-Sphere intersection

```plaintext
ray: x(t) = p_v + v_t, y(t) = p_v + v_t, z(t) = p_v + v_t
(unit sphere): x^2 + y^2 + z^2 = 1

quadratic equation in t:
0 = (p_v + v_t)^2 + (p_v + v_t)^2 + (p_v + v_t)^2 - 1
- t^2(v_x^2 + v_y^2 + v_z^2) + 2(p_v.x + p_v.y + p_v.z)
  + (p_v^2 + p_v^2 + p_v^2) - 1
```

**More About Ray-Tracing**

- Algorithm above has a BUG....

- Does not terminate

- Termination Criteria
  - No intersection
  - Contribution of secondary ray attenuated below threshold – each reflection/refraction attenuates ray
  - Maximal depth is reached

**Optimized Ray-Tracing**

- Basic algorithm simple but VERY expensive

- Optimize...
  - Reduce number of rays traced
  - Reduce number of ray-object intersection calculations

- Methods
  - Bounding Boxes
  - Spatial Subdivision
  - Visibility & Intersection
  - Tree Pruning

**Simulating Shadows**

- Trace ray from each ray-object intersection point to light sources
  - If the ray intersects an object in between ⇒ point is shadowed from the light source

```plaintext
shadow = RayTrace(LightRay(obj,r,light));
return Shade(shadow,reflect_color,refract_color,obj);
```
Advanced Phenomena

- Can (not always efficiently) simulate
  - Soft Shadows

- Fog

- Frequency Dependent Light (diamonds & prisms)

- Barely handle S*DS^*
  - S = Specular
  - D = diffuse