02 Shading and Frames

Steve Marschner
CS5625 Spring 2016
Light reflection physics
Radiometry redux

**Power**

**Intensity** power per unit solid angle

**Irradiance** power per unit area

**Radiance** power per unit (solid angle × area)
Sources of light

Point sources
- intensity
- can be directionally varying—spotlights

Area sources
- radiance
- can be spatially varying

Directional sources
- irradiance (normal irradiance)

Environment lighting
- radiance (usually spatially varying)
- sun-sky models
Simple kinds of scattering

**Ideal specular reflection**
- incoming ray reflected to a single direction
- mirror-like behavior
- arises at smooth surfaces

**Ideal specular transmission**
- incoming ray refracted to a single direction
- glass-like behavior
- arises at smooth dielectric (nonmetal) surfaces

**Ideal diffuse reflection or transmission**
- outgoing radiance independent of direction
- arises from subsurface multiple scattering
Ideal specular reflection from metals
Ideal reflection and transmission from smooth dielectrics

Water (ior = 1.33)

Diamond (ior = 2.4)
Two diffuse surfaces
More complex scattering

Rough interfaces
- metal interfaces: blurred reflection
- dielectric interfaces: blurred transmission

Subsurface scattering
- liquids—milk, juice, beer, …
- coatings—paint, glaze, varnish, …
- natural materials—wood, marble, …
- biological materials—skin, plants, …
- low optical density leads to *translucency*
Reflection from rough metal interfaces

Cu ($\alpha = 0.1$)  
Al (anisotropic)
Reflection and refraction at rough dielectric interfaces

Anti-glare glass ($\alpha = 0.02$)

Etched glass ($\alpha = 0.1$)
Translucent materials

“skim milk”

low optical density

high optical density
Modeling complex scattering

**Opaque materials**
- reflection: bidirectional reflectance distribution function (BRDF)
- transmission: bidirectional transmittance distribution function (BTDF)
- both: bidirectional scattering distribution function (BSDF)

**Translucent materials**
- bidirectional subsurface scattering reflectance distribution function (BSSRDF)
- more on this later, maybe
Isotropy vs. anisotropy

isotropic

anisotropic
Types of BRDF/BSDF models

**Ad hoc formulas**
- e.g. Blinn-Phong

**Physics-based analytical models**
- Lambertian
- Microfacet-based models
- Kirchhoff-based models

**Measured data**
- tables of data from pointwise BRDF measurements
- image-based BRDF measurements
Light reflection in shaders
all types of reflection reflect all types of illumination

- diffuse, glossy, mirror reflection
- environment, area, point illumination
## Categories of illumination

<table>
<thead>
<tr>
<th></th>
<th>diffuse</th>
<th>glossy</th>
<th>mirror</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>indirect</strong></td>
<td>soft indirect illumination</td>
<td>blurry reflections of other objects</td>
<td>reflected images of other objects</td>
</tr>
<tr>
<td><strong>environment</strong></td>
<td>soft shadows</td>
<td>blurry reflection of environment</td>
<td>reflected image of environment</td>
</tr>
<tr>
<td><strong>area</strong></td>
<td>soft shadows</td>
<td>shaped specular highlight</td>
<td>reflected image of source</td>
</tr>
<tr>
<td><strong>point/directional</strong></td>
<td>hard shadows</td>
<td>simple specular highlight</td>
<td>point reflections</td>
</tr>
</tbody>
</table>

= easy to compute using standard shaders
How to compute shading

Basic case: point or directional lights; diffuse or glossy BRDF

Type in BRDF model, plug in illumination and view direction
  • can write down model in world space, use world-space vectors
  • can write down model in surface frame, transform vectors
  • really not different

Subtleties are all about what frame to use for shading
Interpolated shading

Coarse triangle meshes are fast
Discontinuities are bad

Therefore: interpolate geometric quantities across triangles
  · goal: shading is smooth across edges

What do we interpolate?
  · what do we need to compute shading?
Shading frames

When we carry around a normal, we are defining a tangent plane

• interpolated normal defines an approximate, smoothly varying tangent plane

For some purposes, the tangent plane is enough

• e.g. computing shading for isotropic BRDFs
• any coordinate system conforming to the normal is equally good

In other cases, need a complete frame

• whenever directions within the plane are inequivalent
• e.g. anisotropic BRDFs
• e.g. tangent-frame normal maps

How to compute these from normals and texture coordinates? (blackboard)
What to interpolate

Need plane: can just interpolate a normal

Need frame: interpolate enough data to define a tangent frame

One and a half vectors rounds up to two
  • normal and one tangent vector
  • two tangent vectors

Rebuilding a frame from the vectors
  • worry about handedness matching texture coordinates (or not)
  • orthonormality gets broken by interpolation (when does that matter?)
What you need for shading

When/why you need full frames

• when you care (or not) what the orientation is
• when you care (or not) about orthonormality

What to interpolate

• underlying math question: representation of frames
• representations that behave well under interpolation

How to author orientation

• with maps
• by following a parameterization

How to deal with corner cases