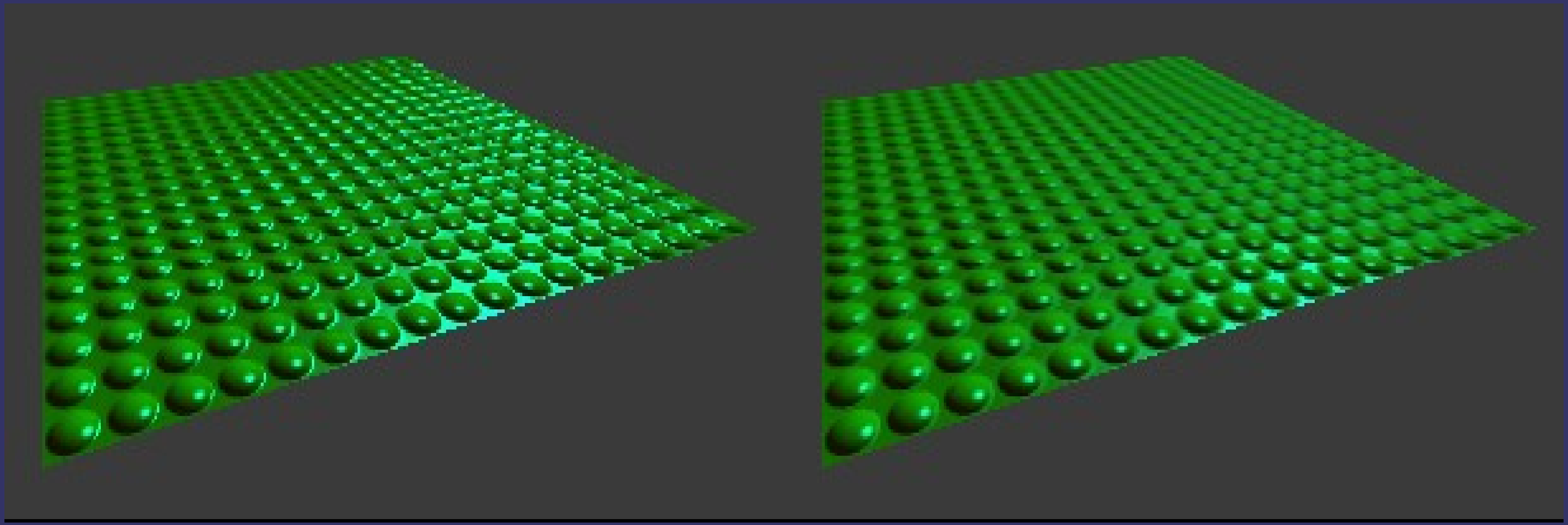
Computer Graphics Programming II

- Agenda:
  - Mipmapping normal maps
  - Environment map pre-filtering for lighting
  - Fresnel reflection
  - BRDF introduction
    - Common terminology and notation
    - Cook-Torrance

Mipmapping Normal Maps

- Traditional mipmaps fail for normal maps
  - Averaging multiple vectors yields a vector less than unit length, which leads to lighting artifacts.



Original image from
http://developer.nvidia.com/object/mipmapping_normal_maps.html

Theory

- By knowing how much less than unit length the averaged normal is, we can *improve* the lighting
  - Decrease the amount of specular reflection
  - Make the specularity more diffuse-like → decrease the specular exponent

Toksvig Factor

- Update Blinn's lighting equation:

$$f_t = \frac{|N_a|}{|N_a| + s\left(1 - |N_a|\right)}$$

$$K_s = \frac{1 + f_t s}{1 + s}\left(\frac{N_a \cdot H}{|N_a|}\right)^{f_t s}$$

  - f_t is the “Toksvig Factor”
  - N_a is the averaged normal
  - s is the original specular exponent

Environment Maps

- Last week we used env. maps for lighting
  - Calculate ideal reflection vector, use that to look up light color from environment map
- What is the limitation of this approach and why?

Environment Maps

- Last week we used env. maps for lighting
  - Calculate ideal reflection vector, use that to look up light color from environment map
- What is the limitation of this approach and why?
  - Really only works for perfectly mirror-like surfaces
    - Surfaces where the specular exponent approaches ∞
  - Only one point of the environment is sampled

Env. Map Sampling

- If the problem is that only one point in the original environment map is sampled, what can we do?

Env. Map Sampling

- If the problem is that only one point in the original environment map is sampled, what can we do?
  - Obvious answer: take multiple samples
  - Sample R and a few vectors around R, and take a weighted average

Env. Map Sampling (cont.)

- What's the problem with this technique?

Env. Map Sampling (cont.)

- What's the problem with this technique?
  - Taking *enough* samples to get good results is slow
  - Taking few enough samples to be fast will give poor looking results
- Does this remind you of something?

Env. Map Sampling (cont.)

- What's the problem with this technique?
  - Taking *enough* samples to get good results is slow
  - Taking few enough samples to be fast will give poor looking results
- Does this remind you of something?
  - Sounds like texture minification
  - To solve that problem we store pre-averaged (filtered) versions of the texture
    - Mipmaps

Env. Map Pre-filtering

- For a cube map:
  - Render a single quad for each face
  - Each fragment is a unique R vector
    - Calculate the matching V vector
  - Sample some number of surrounding texels
    - Each texel is an L vector
    - The contribution of each texel is based on its vector
  - Calculate weighted average of all sampled texels, write result

Env. Map Pre-filtering (cont.)

- Notes / caveats:
  - This filtered env. map *only* includes the specular component
    - Create a second similar env. map for diffuse
      - This is called an *irradiance map*
    - Can apply essentially *any* lighting model to this technique
  - Other env. map types can be handled similarly

Break

Fresnel Reflection

- Named after French physicist Augustin-Jean Fresnel
  - Pronounced *fray-NELL*
- Light moves at different speeds through different materials
  - The ratio of the speed of light in a vacuum to the speed in a particular material is the *refractive index* of that material
  - Glass has an index of refraction of about 1.5

Fresnel Reflection

- When light passes through a material with one refractive index to a material with another, several things happen:
  1. The light changes velocity
  2. The light changes direction
     - Since light behaves like a wave, #1 causes #2.
  3. Some of the light is reflected
  4. The remaining light is refracted
     - That is, passes into the material

Fresnel and Materials

- Dielectric materials (i.e., things that do *not* conduct electricity) exhibit a strong Fresnel factor
  - Plastic, glass, autopaint, etc.
  - Metal *does not*
  - This fact will be important later...

Reflection vs. Refraction

- The amount of reflection depends on the angle between the light and the normal
  - The larger the angle between the normal and the light, the more light is reflected
  - The effect is like a rock skipping on water

Reflection Math

- The amount of reflection, *R(θ)*, is:

$$c = n_i / n_t (\cos\theta)$$

$$g = \sqrt{1 + c^2 - (n_i / n_t)^2}$$

$$R(\theta) = \frac{1}{2}\left(\frac{(g-c)}{(g+c)}\right)^2 \left(1 + \left(\frac{c(g+c) - (n_i/n_t)^2}{c(g-c) + (n_i/n_t)^2}\right)^2\right)$$

  - n_i is the refractive index of the first material
  - n_t is the refractive index of the second material
  - θ is the angle between the surface normal and the light vector

Reflection Approximation

- That math is complex *and* expensive
- A good approximation exists:

$$R_a(\theta) = R(0) + (1 - R(0))\left(1 - \cos(\theta)\right)^5$$

  - *R(0)* is calculated in the application and passed into the shader as a uniform

How is it used?

- Simulate a diffuse surface with a shinny coating:

$$K = (1 - F) K_d + F K_s$$

  - The Fresnel term determines what part of the light is reflected by the specular coating
  - Whatever light is left is reflected by the diffuse part

References

http://www.graphics.cornell.edu/~westin/misc/fresnel.html

http://physics-animations.com/Physics/English/rays_txt.htm

http://en.wikipedia.org/wiki/Fresnel_equations

Break

BRDFs

- Bi-direction reflection distribution function
  - Notation is $f(\omega_o, \omega_i)$

    "...describes the ratio of reflected radiance exiting from a surface in a particular direction (defined by the vector ω_o) to the irradiance incident on the surface from direction ω_i over a particular waveband."

BRDFs In Lighting

- The amount of light reflected for a particular angle is:

$$L(\omega_o) = f(\omega_o, \omega_i) L(\omega_i)(N \cdot \omega_i)$$

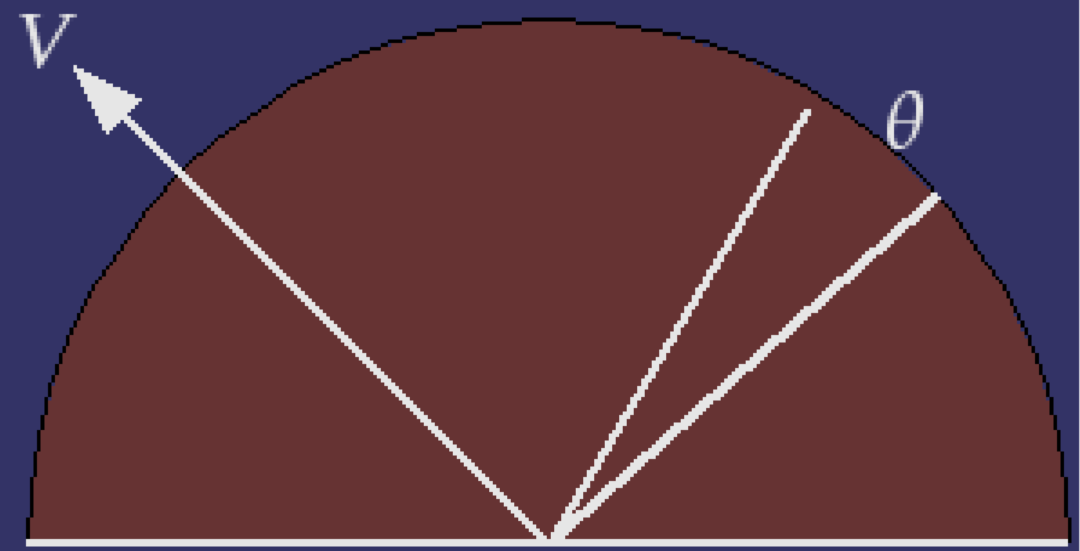
  - $L(\omega_o)$ is the outgoing light
  - $L(\omega_i)$ is the incoming light
  - N is the surface normal

Notes About Notation

- ω (omega) is used for vectors
  - This is the notation used for BRDFs in physics literature, and all papers about BRDFs use it as well
- Exchange ω_i and ω_o and get the same result
  - This is called Helmholtz Stereopsis or Helmholtz reciprocity
  - It's the "bi-directional" part of BRDF
- Measured as a solid angle

Solid Angle?

- Each measurement is flux times solid angle
  - To get total light output, integrate over the entire hemisphere
  - Results in an extra scale by the total angle, π
    - Pretty much all BRDFs have a $1/\pi$ factor to account for this



Where does f come from?

- Several common sources:
  - Complex camera and light setups can be used to sample a BRDF
    - Oregon BRDF Library (and other) have a lot of this raw data available
  - Analytic methods can be used to derive a BRDF
    - We'll cover several common ones this term
    - Most include a Fresnel term

Cook-Torrance

- One of the oldest BRDFs used in graphics
  - Developed while Robert Cook was at Lucasfilm and Ken Torrance was at Cornel
  - Published in 1982
- Based on “microfacets”

Microfacet Primer

- Surfaces are comprised of numerous infinitesimal subsurfaces that act as perfect mirrors
  - Distribution of the subsurface normals determines how specular a surface is
  - Microfacets can obscure other microfacets from both the light and from the viewer
- We'll cover both these elements in *much* more detail next week

Cook-Torrance BRDF

$$f(\omega_o,\omega_i)=K_d f_d+K_s f_s(\omega_o,\omega_i)$$

$$f_d=1/\pi$$

$$f_s=1/\pi\,\frac{F\times D(N\cdot H)\times G(N\cdot\omega_i,N\cdot H,N\cdot\omega_o)}{(N\cdot\omega_i)(N\cdot\omega_o)}$$

Microfacet Distribution

- D represents the distribution of microfacet normals
  - Several models exist
  - Cook-Torrance uses the Beckmann Distribution

$$D(N \cdot H) = \frac{1}{4m^2 (N \cdot H)^4} e^{-\left(\frac{1-(N \cdot H)^2}{(N \cdot H)^2 m^2}\right)}$$

m is a constant on [0, 1] that controls the smoothness of the surface

Geometry Factor

- Represents facet-facet occlusion of light source and / or vierwer

$$G(N\cdot\omega_i, N\cdot H, N\cdot\omega_o) = min\left(1, \frac{2(N\cdot H)(N\cdot\omega_o)}{\omega\cdot H}, \frac{2(N\cdot H)(N\cdot\omega_i)}{\omega\cdot H}\right)$$

- Why aren't there subscripts on ω in the denominators?
  - Hint: remember ω_i is the light and ω_o is the viewer

Geometry Factor

- Represents facet-facet occlusion of light source and / or vierwer

$$G(N\cdot\omega_i, N\cdot H, N\cdot\omega_o) = min\left(1, \frac{2(N\cdot H)(N\cdot\omega_o)}{\omega\cdot H}, \frac{2(N\cdot H)(N\cdot\omega_i)}{\omega\cdot H}\right)$$

- Why aren't there subscripts on ω in the denominators?
  - Hint: remember ω_i is the light and ω_o is the viewer
  - H is half-way between ω_i and ω_o, so $(H \bullet \omega_i) = (H \bullet \omega_o)$

Cook-Torrance Diffuse Factor

- Cook-Torrance diffuse factor:

$$f_d = 1/\pi$$

- “Typical” diffuse factor:

$$K_d = N \cdot L$$

Cook-Torrance Diffuse Factor

- Cook-Torrance diffuse factor:

$$f_d = 1/\pi$$

- “Typical” diffuse factor:

$$K_d = N \cdot L$$

- Remember how the BRDF equation is used:

$$L(\omega_o) = f(\omega_o, \omega_i) L(\omega_i)(N \cdot \omega_i)$$

  - We just want to scale the incoming energy by the total angle and let the built-in ($N \bullet \omega_i$) do the rest
  - Remember ω_i <u>is</u> L

References

- Measured BRDF data:
  - http://math.nist.gov/~FHunt/appearance/obl.html
  - http://www.graphics.cornell.edu/online/measurements/

Next week...

- More BRDFs
  - Visit microfacet theory in more depth
- Assignment #2 due

Legal Statement

- This work represents the view of the authors and does not necessarily represent the view of IBM or the Art Institute of Portland.
- OpenGL is a trademark of Silicon Graphics, Inc. in the United States, other countries, or both.
- Khronos and OpenGL ES are trademarks of the Khronos Group.
- Other company, product, and service names may be trademarks or service marks of others.