Johns Hopkins
Engineering for Professionals

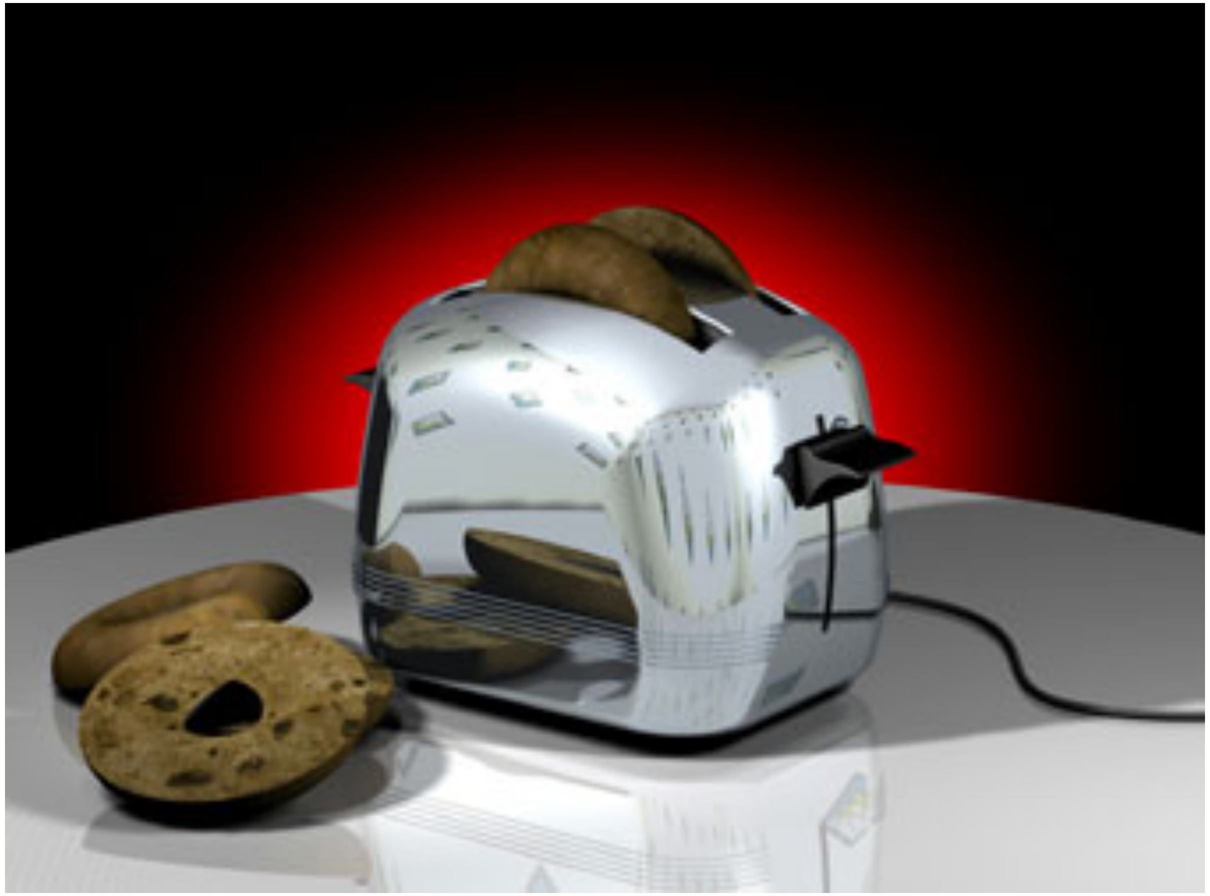
605.767 Applied Computer Graphics

Brian Russin

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Module 9F
Environment Maps

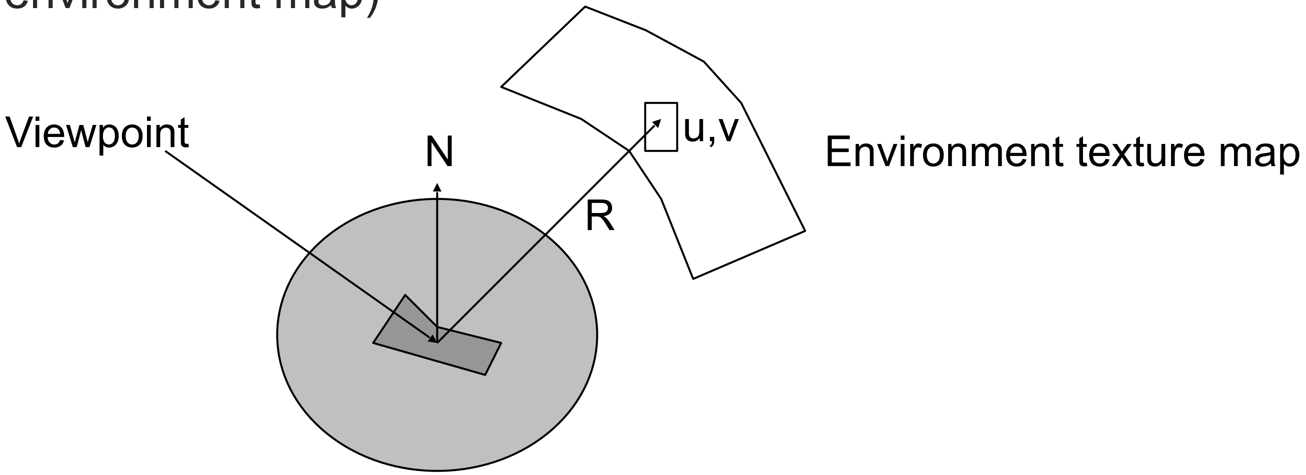
Environment Mapping



SolidWorks 2000 (from NVIDIA web site)

Environment Mapping

- Environment mapping refers to the process of reflecting the surrounding environment in a shiny object
  - Texture mapping technique where the pattern mapped onto an object depends on the view vector
  - Simplification of ray tracing where only the reflected ray is traced
    - Process is terminated once it intersects texture
- Reflect the view ray from the object surface
  - Intersect this ray with a sphere containing the image of the environment to be reflected (called an environment map)



Steps in Environment Mapping

- Generate or load a 2D image of the environment
  - Environment maps are usually prefiltered
- For each pixel containing a reflective object, compute its normal
- Compute the reflection vector from the view vector and the normal

$$R = 2(N \cdot V)N - V$$

- Use the reflection vector to index into the environment map
- Use texel data from the environment map to color the pixel
- Several projection functions are available
  - Blinn and Newell’s method
  - Sphere map
    - Improves on Blinn and Newell
    - Used in many early OpenGL demos
  - Cubic environment mapping
  - Other environment mapping methods

Flat Surfaces

- Note on Environment Mapping
  - Large flat surfaces do not work as well
    - Reflected rays do not vary as much as off curved surfaces
    - Small part of environment map is mapped onto a relatively large surface
      - Individual texels of environment map become visible

Blinn and Newell’s Method

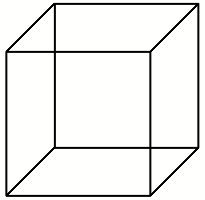
- Use the reflected view vector to index into latitude, longitude environment map
  - Rather than a globe viewed from the outside it is like a map of constellations in the sky
  - Single equirectangular texture map can be created from real world environments using photography techniques
    - 360-degree camera
- Conversion of the reflected ray $r=(r_x, r_y, r_z)$ into latitude-longitude
  - Latitude varies from 0 to π radians
    - $Lat = \arccos(-r_z)$
  - Longitude varies from 0 to 2*π radians
    - $Lng = \operatorname{atan2}(r_y, r_x)$
- Problems
  - Non-uniform density – more texels near the poles
  - Singularities
    - Poles
    - Issues where left and right edges meet

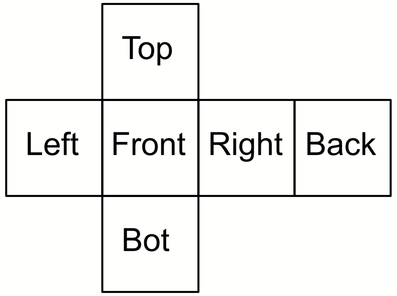
Spherical Environment Mapping

- 1st Environment mapping technique supported in hardware
- OpenGL Programming Guide gives a method for creating an environment map
  - Take a photograph of a large silvered sphere in a surrounding environment
  - Camera located infinite distance away, lens with infinite focal length
  - Scan the photograph
  - Alternatively: use an extremely wide angle (fish-eye) lens
- With OpenGL use GL_SPHERE_MAP texgen mode
- Disadvantage – valid only for a single view direction
  - Can map other views but can get artifacts where parts of the sphere map become magnified
- Good Resources
  - Section 10.4.2 (8.4.2 in 3rd Edition)
  - http://www.debevec.org/ReflectionMapping/
  - http://www.ozone3d.net/tutorials/glsl_texturing_p04.php#part_41

Cubic Environment Mapping

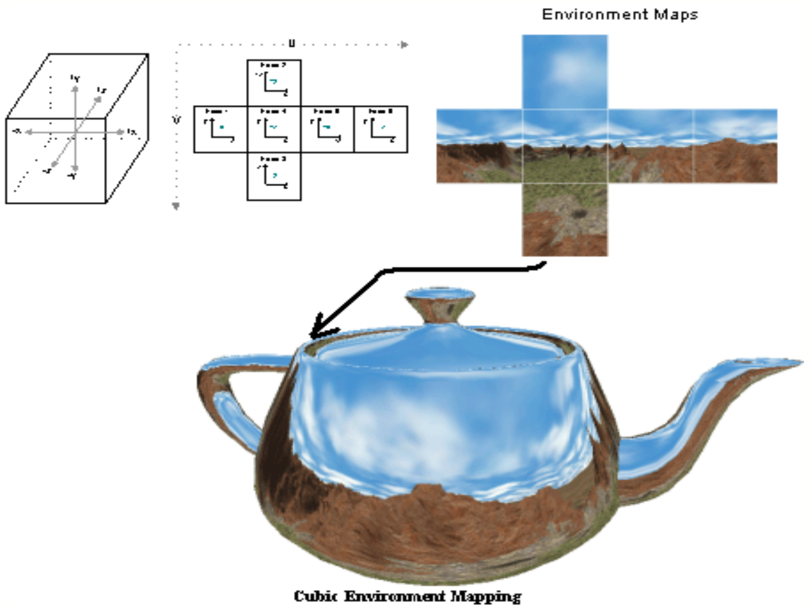
- Many graphics accelerators offer cubic environment mapping support
  - Support in OpenGL and other graphics APIs
- Environment map is 6 maps that form the faces of a cube
  - View point is fixed at the center, 6 views are rendered
  - View independent
- For a single reflection vector (normalized)
  - Find the face it intersects (which map)
    - Reflection vector with the largest magnitude selects the face
      - E.g. (-0.2, 0.5, -0.84) selects the -z face
  - Map the components of the reflected vector into (u,v)
    - Example of intersection with negative z axis:
      - $u = r_x + 0.5$
      - $v = -r_z + 0.5$





Cubic Environment Mapping

- Advantages
  - Environment maps can be rendered in real-time
    - Place camera at center of cube
    - Project the scene onto the 6 faces of a cube
  - More uniform sampling than spherical methods
    - Spherical methods have more texels near poles than equator



http://www.ocworkbench.com/hardware/ati/radeon/radeonp3.htm [link inactive]

Using Cube Maps with GLSL

- Sampler function: samplerCube
- Texture access function: textureCube
  - Provide the cubeMap and the reflection vector
- OpenGL setup

```
glEnable(GL_TEXTURE_CUBE_MAP);
glGenTextures(1, &cubemap_texture);
glBindTexture(GL_TEXTURE_CUBE_MAP, cubemap_texture);
glTexParameteri(GL_TEXTURE_CUBE_MAP, GL_TEXTURE_WRAP_S, GL_REPEAT);
glTexParameteri(GL_TEXTURE_CUBE_MAP, GL_TEXTURE_WRAP_T, GL_REPEAT);
glTexParameteri(GL_TEXTURE_CUBE_MAP, GL_TEXTURE_WRAP_R, GL_REPEAT);
glTexParameteri(GL_TEXTURE_CUBE_MAP, GL_TEXTURE_MAG_FILTER, GL_LINEAR);
glTexImage2D(GL_TEXTURE_CUBE_MAP_POSITIVE_X, 0, GL_RGBA, 256,
                      256,0, GL_RGBA, GL_UNSIGNED_BYTE, image1);
glTexImage2D(GL_TEXTURE_CUBE_MAP_NEGATIVE_X, 0, GL_RGBA, 256,
                      256,0, GL_RGBA, GL_UNSIGNED_BYTE, image2);
glTexImage2D(GL_TEXTURE_CUBE_MAP_POSITIVE_Y, 0, GL_RGBA, 256,
                      256,0, GL_RGBA, GL_UNSIGNED_BYTE, image3);
glTexImage2D(GL_TEXTURE_CUBE_MAP_NEGATIVE_Y, 0, GL_RGBA, 256,
                      256,0, GL_RGBA, GL_UNSIGNED_BYTE, image4);
glTexImage2D(GL_TEXTURE_CUBE_MAP_POSITIVE_Z, 0, GL_RGBA, 256,
                      256,0, GL_RGBA, GL_UNSIGNED_BYTE, image5);
glTexImage2D(GL_TEXTURE_CUBE_MAP_NEGATIVE_Z, 0, GL_RGBA, 256,
                      256,0, GL_RGBA, GL_UNSIGNED_BYTE, image6);
```

Cube Maps

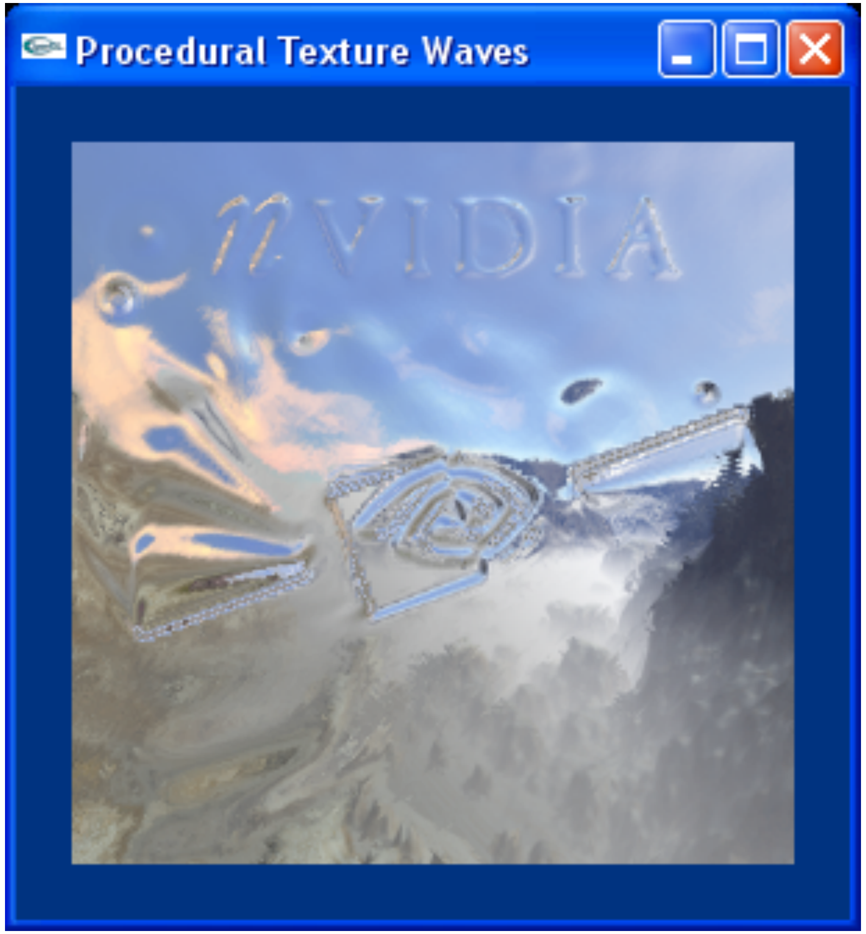
- Downloadable cube maps:
  - https://www.cleanpng.com/free/cube-mapping.html
- Dynamic cube map generation
  - Can generate cube maps for your scene by drawing all objects except reflecting object with camera placed in center of reflecting object
    - Single pass to render all 6 different views
      - Possible in Direct X Shader Model 4 and some OpenGL extensions
      - https://registry.khronos.org/OpenGL/extensions/ARB/ARB_geometry_shader4.txt
      - http://stackoverflow.com/questions/462721/rendering-to-cube-map
- Tutorial
  - http://www.ozone3d.net/tutorials/glsl_texturing_p05.php#part_51

Other Environment Mapping

- Dual Parabaloid Mapping
  - Similar to spherical mapping but uses parabaloid
  - More uniform texel sampling than other methods
- Octahedral Mapping
  - Similar to cube mapping
  - Projects onto octahedron
  - Can easily be unfolded to fit in one image file

Bump Environment Mapping

- Combination of bump mapping and environment mapping
- Perturbs environment mapping coordinates by the normal differentials found in the bump map
  - Effect is to perturb the reflection vector
- Requires 2 additional maps: bump map and an environment map



http://developer.nvidia.com/view.asp?IO=ogl_dynamic_bumpreflection [link inactive]