CS 33400, Fall 2014
Voicu Popescu

Due: Wednesday November 26th at 6:00am

A9—Environment Mapping

In a nutshell

Enhance your interactive renderer with environment mapping, applied to rendering distant geometry and reflections. Environment mapping should be supported both in SW and in HW (GPU shaders).

Details

1. Scene
    - Auditorium model + teapot model floating in mid-air inside auditorium.
    - Teapot is made of perfectly reflecting "chrome".
2. Environment map modeling
    - Cube map acquired at teapot center using HW rendering
    - Teapot not rendered as the faces of the cube map are rendered (i.e. acquired)
    - Save cube map (face images + camera parameters)
3. SW
    - Environment map implementation: a class that
        i. Stores 6 framebuffers and 6 ppcs, one of each per face
        ii. 512x512 resolution for each face
        iii. Constructor from saved images and camera parameters
        iv. Given a direction it returns a color
    - Distant geometry, i.e. auditorium, rendered by environment mapping
        i. For each pixel, lookup ray (from eye to pixel center) into environment map
    - Reflections on teapot rendered by environment mapping
        i. For each pixel of a reflective triangle, compute reflected ray, lookup reflected ray into environment map
4. HW (GPU shaders)
    - Environment mapping implemented using samplerCube (see Cg or other doc.)
    - Distant geometry, i.e. auditorium, rendered by environment mapping:
        i. A GPU fragment shader running on image frame quad
        ii. Same algorithm as in SW
    - Reflections on teapot rendered by environment mapping
        i. A GPU fragment shader
        ii. Same algorithm as in SW
5. Demo
    - Render only environment and reflective teapot. Do not render auditorium geometry. Auditorium rendered through environment mapping.
    - View navigation restricted to revolving around teapot (3 degrees of freedom)

- Define a camera path (camera revolves around teapot), save to file; have GUI button that animates camera along path.
- Make a 10s 30fps video of your scene with camera navigating according to the path.

6. Extra credit, all through GPU programs:
    - When user clicks on teapot, teapot is deformed locally, around the clicked point. Deformation is dynamic, deformation amplitude decreases to 0 in 3 seconds (i.e. the teapot goes back to its un-deformed state). 3%

Turn in

- Code including GPU programs.
- Camera path file.
- Movie file.
- A README.txt description of your GUI.