CSC 473 – Advanced Rendering Techniques

1. CSC 473 – Advanced Rendering Techniques

2. credit units 4  contact hours 6

3. Course Coordinator: Zoë Wood

4. Textbook (or other required material):
   M. Pharr and Humphreys, “Physically Based Rendering”

5. a. Course Description:
   Illumination models, reflectance, absorption, emittance, Gouraud shading, Phong shading,
   raytracing polyhedra and other modeling primitives, coherence, acceleration methods,
   radiosity, form factors, advanced algorithms. 3 lectures, 1 laboratory.

   b. Prerequisite: CSC/CPE 471.

   c. Required/Elective/Selective Elective for CPE, CSC, EE, SE

<table>
<thead>
<tr>
<th></th>
<th>CSC</th>
<th>CPE</th>
<th>SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elective</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Selective Elective</td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

6. a. Course Learning Objectives
   The student will be able to:
   - Describe the basic problems with photorealistic rendering.
   - Use mathematical approaches for photorealistic rendering of complex scenes.
   - Create programs to generate photorealistic renderings of complex scenes.
   - Describe the basics of Monte Carlo ray tracing (or a similar global illumination algorithm)
   - Correctly implement ray-sphere, ray-plane, ray-triangle intersections (with correctness
demonstrated via unit testing)
   - Describe and apply basic object-oriented design in order to create a well-structured larger
software project (stochastic sampled ray tracer)
   - Program basic data structures to represent geometric objects in a scene (sphere, planes,
triangles), including the application of transforms (translate, scale, rotate) and scene objects
such as lights, and the camera
   - Describe and implement shadow feelers to produce shadows in a software render (ray
tracer)
   - Describe and implement ray traced rendering of reflective surfaces
   - Describe and implement ray traced rendering of refractive surfaces
   - Describe and implement a ‘virtual camera’
   - Describe the basics of Monte Carlo sampling in order to implement an approximation of
global illumination via a distributed ray tracer implementation
   - Describe and implement a few BRDF (Bi-directional radiance distribution functions) to
simulate the reflection of light
   - Translate mathematics into a higher level computer programming language (e.g. C++)
• Exposure to and implementation of some subset of advanced topics: texture mapping, anti-aliasing, depth of field, motion blur, spatial data structures, parallel programming, point based color bleeding, photon mapping, path tracing, radiosity, scripting to produce animation, real-time ray traced rendering via writing results to framebuffers in OpenGL, basics of VR, etc.

b. **Level at which Student Outcomes are addressed**
   (“B” = Basic level, “I” = Intermediate level, “A” = Advanced level)

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSC</td>
<td>A</td>
<td>A</td>
<td></td>
<td></td>
<td>I</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>SE/CPE</td>
<td>A</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>I</td>
</tr>
</tbody>
</table>

7. **Major Topics Covered: (number of lecture hours per)**
   • Raytracing (9)
   • Rendering equation (2)
   • Monte Carlo sampling (distribution ray tracing) (2)
   • Reflection (2)
   • Refraction (4)
   • Global Illumination and Advanced Techniques (8)