Worksheet 6

Path tracing starts at the eye and traces rays through the scene. Every position reached in the scene is connected to the light sources. In some cases this is not the optimal strategy. Sometimes it is beneficial to start from the light source, trace rays through the scene, and connect every encountered position to the eye. This procedure is called *light ray tracing* and it is, among other things, good for rendering caustics. Photon mapping is the golden middle way where you trace some rays from the light and some from the eye. The following exercises are about implementing photon mapping.

Learning Objectives

- Implement photon mapping.
- Do direct visualisation of a photon map.
- Do final gathering.
- Use a global photon map with final gathering and a separate photon map for caustics.

Photon Mapping

In Worksheet 5, you rendered the two classic Cornell box scenes using path tracing. This means that you have reference images for these two scenes which you can compare your photon mapping results with. Consequently, the following exercises will involve the same two scenes. The main problem in path tracing is bright dots that sometimes appear as a consequence of a low probability event being chosen by Russian roulette. These dots take "forever" to mean out. Soft caustics are particularly prone to this problem. The goal of the following exercises is to show that photon mapping is one way to overcome this path tracing problem.

- Implement the algorithm that builds photon maps. For these exercises you only need to take photon emission from an area light source into account. (In the `pathtrace` project of the course framework, implement the `emit` function in `AreaLight.cpp`. Set the desired number of photons in `RenderEngine.cpp`, lines 72–73, to ensure that some photons are emitted. The framework takes care of the tracing and the photon map construction.)
- Load the Cornell box (`CornellBox.obj`) and the blocks inside it (`CornellBlocks.obj`) into your ray tracer. Set the background illumination to 0 and render the photons in your photon map as dots. Save the resulting image. (In the framework, the photon map is rendered as dots in preview when you choose the `PhotonLambertian` shader. This shader is used when you press '5' on the keyboard.)
- Now implement the radiance estimate for the global photon map. Shade the scene using this radiance estimate. Your implementation works when the result compares qualitatively to the path traced Cornell box. Save the resulting image. (In the framework, implement the `shade` function in `PhotonLambertian.cpp` and toggle off final gathering by pressing 'g' on the keyboard before you render the scene. Set the photon mapping parameters in the `Globals` section of `RenderEngine.cpp`.)
- Unless a very high number of photons is traced and stored, the radiance estimate has a lot of low frequency noise. Avoid the worst low frequency noise by implementing final gathering. Render the scene using final gathering and the global photon map. Note the rendering time, the number of samples, and the configuration used for the radiance estimate, that is, the maximum look-up distance and number of photons in the radiance estimate. Save the resulting image. (In the framework, implement the `split_shade` function in `PhotonLambertian.cpp`.)

- When final gathering is used, caustics are handled separately by a caustics photon map. Implement a shader for the caustics photon map. (In the framework, implement the `shade` function in `PhotonCaustics.cpp`.)
- Replace the blocks in the Cornell box by a silver sphere (`CornellLeftSphere.obj`) and a glass sphere (`CornellRightSphere.obj`). Render the photons in your caustics photon map as dots. Save the resulting image. (In the framework, the `PhotonCaustics` shader is used when you press '4' on the keyboard.)
- Render the radiance estimate from the caustics photon map only. Save the resulting image.
- Render the scene using the complete photon mapping solution. Note the rendering time, the number of samples, and the configuration used for the radiance estimates. Save the resulting image.

Worksheet 6 Deliverables

Two series of images: one series showing the photons, the radiance estimate, and the final result for the Cornell box with blocks; one series showing the caustics photons, the caustics radiance estimate, and the final result for the Cornell box with two spheres. Include relevant code and render log (number of triangles, number of samples, render time, number of splits, and likewise). Compare the results to path traced reference images and discuss the reasons for the differences. Answer the following question:

Why is photon mapping a biased method?

Reading Material

The curriculum for Worksheet 6 is

P Section 16–16.2. *The Path-Space Measurement Equation* and *Stochastic Progressive Photon Mapping*.

Alternative literature available online or uploaded to CampusNet:

- Jensen, H. W., and Christensen, N. J. *A Practical Guide to Global Illumination Using Photon Maps*, ACM SIGGRAPH 2000 Course Notes, Course 8, 2000.