CS562 Project 2

Synopsis

Continuing from the previous project, adding an implementation of the exponential shadow map. Experiment with different blurring techniques and several amounts of blurring: small amount for anti-aliasing and large amount to simulate diffuse lighting.

Instructions

The basic algorithm is very straightforward:

- Store e^{cd} instead of just depth d in the shadow map.
- Blur (filter) the shadow map.
- For the shadow test, project the pixel p onto the shadow map as before, and compute the shadow factor as $(filtered\,depth\,at\,p) * e^{-cz}$ for pixel depth z.
- If the shadow factor is greater than one, set it to one.

In both uses of depth (object depth d and pixel depth z), the depth should be mapped to a a standard range, say 0...1: $(depth - minDepth)/(maxDepth - minDepth)$. Then the paper's suggestion of a $c = 60$ works well.

Blur filter

Implement a blur filter in a compute shader. (If you don't have access to a graphics card that supports compute shaders, this may be done in a pixel/fragment shader.) See the Moodle lecture notes for an explanation of this algorithm and some notes on programming OpenGL compute shaders. A nice set of $2w + 1$ Gaussian weights (i.e., a bell curve) is

$$\beta\, e^{-\frac{i^2}{2s^2}} \quad \text{for i in range } -w \ldots w,$$

where the normalization factor β is chosen so the weights sum to 1, and s=w/2 controls the width of the bell curve. (The usual recommendation of $s = w/3$ puts too many near-zeros at the ends of the weight array which is inefficient for real-time uses.)

What to display

You should have a way to display your (exponential) depth buffer, both before and after the blur. Since the contents of the depth buffer are exponential values, you may want to take their log when you display them. Beware that the log(blur(exponential(depth))) looks subtly different than blur(depth).

Report

Submit a zip file containing the relevant code and a project report. Your report should contain sufficient screen captures (and accompanying text) to demonstrate the correctness of your project, including:

- Images of the depth buffer (or log thereof) both before and after the blur.
- Some discussion of the c parameter and its effect on the algorithm.
- Final images showing a range of shadows, from just enough blurring to be anti-aliased, to lots of blurring to produce very soft shadows simulating a wide light source.