Parallel Visual Computing 2012/13

Assignment 4

November, 23 2012

1. **Framework improvements**

   Do this first, it will help you to work!

   a. **Timing** (4 points)
      Remember that you need to call `glFinish` before taking the time.
   b. **Reloading** (4 points)
      When pressing a key, reload all shaders
   c. **Readback** (4 points)
      Make a function that reads back the framebuffer into an image. Use `glReadPixels`.

2. **Performance instrumentation**

   Please make tables, with Excel or ASCIII. Have absolute values and improvement ratio to a baseline you select.

   a. **Analysis** (8 points)
      i. Speed vs. image size for both your best CPU and GPU bilateral
      ii. Speed vs. kernel size for both your best CPU and GPU bilateral
      iii. Speed vs. image formats (RGBA8, half float, float)
   b. **What is the fastest kernel**? (4 points)
      i. Full Gauss with exp()
      ii. Tabulated in uniform array
      iii. A const float array in the shader
   c. **Approximations** (3 points)
      How do approximations perform in terms of quality and speed?
      i. Smoothstep
      ii. Tent
      iii. Box
   d. **Separability** (3 points)
      You cannot do a separable Gaussian Kernel in this framework, but can you still find out approximately at what kernel size separable wins on a GPU? Note, that going to the length of implementing it separable is not required here.

3. **Application: Joint Bilateral Upsampling** (optional, 10 points)

   Joint bilateral Upsampling is great to improve quality and save time. Implement this in a separate shader `JointBilateralUpsamplingFragment`. Look at the paper, it is really simple: http://johanneskopf.de/publications/jbu/

As test data we provide two dataset:

a. A low-resolution chrominance map and a high-resolution luminance map. Result should be a high-resolution RGB map.
b. A low-resolution depth map and a high-res color image. Result should be a high-resolution depth map.