Technische Hochschule Ingolstadt
Prof. Dr.-Ing. Richard Membarth

17.05.2023

GPU Programming
Assignment 6

Submission deadline for the exercises: 05.06.2023

HDR Pipeline: Adaptive Exposure

The goal of this exercise is to move from a problem that is trivially parallelizable on to a slightly more advanced problem that requires dealing with data dependencies of a certain kind. Your task is to extend the tone mapping example from the previous assignment by adding automatic exposure control. The tone mapping operator implemented in the previous assignment requires the user to explicitly specify what exposure value to use for the purpose of rendering the image. One way to automatically determine what exposure to use for a given scene is by computing the average luminance of the scene and then picking the exposure that will end up mapping the average luminance to medium gray (and, thus, hopefully mapping most of the image to the midtone range).

The perceived luminance L of a color given by its red, green, and blue components r, g, and

b can be computed as the weighted average[1]:

$$L = 0.2126 \cdot r + 0.7152 \cdot g + 0.0722 \cdot b. \tag{1}$$

To compute the average luminance of a scene as a whole, we use the log-average[2]

$$\overline{L} = \exp\left(\frac{1}{N}\sum_{(x,y)} \ln L_{(x,y)}\right) \tag{2}$$

where $L_{(x,y)}$ is the luminance of the pixel at location (x, y) and N the total number of pixels in the image.

Once we have calculated the average luminance, the last step is to compute the exposure parameter a to map our average luminance to 18% gray[3]:

$$e = a \cdot \frac{0.18}{\overline{L}}. \tag{3}$$

The *key a* serves as a tunable parameter for the user to influence the overall look of the result.

HDR Pipeline Task 2: Adaptive Exposure

Modify the function `HDRPipeline::process` in «`src/hdr_pipeline.cpp`» such that it automatically determines what exposure to use prior to tone mapping following the approach described above. Use the GPU to compute the log-average luminance value needed as part of this process. Rather than specifying the exposure directly, the parameter `exposure` should now serve as the key a in Equation (3). Feel free to extend the class `HDRPipeline`, e.g., by adding members in case you need to allocate additional memory buffers to use during your computation.

---

[1]this is assuming an RGB color space with BT.709 primary colors
[2]which corresponds to the geometric mean
[3]https://en.wikipedia.org/wiki/Middle_gray