Exercises for Advanced Graphics
Lectures 9–12

Rafał Mantiuk

Lent term 2016/17

1 Reflection models

1. Check the example of BRDF rendering from `https://www.shadertoy.com/view/XsjSRG` and analyse `getColor` function. Answer the following questions:

   (a) How is the cube map used to compute colour values?

   (b) The name of the shader suggests that this is an example of physically based rendering, which would involve solving the integral of the rendering equation for each pixel. Does the shader solve the rendering equation? Is gloss correctly computed for the given illumination map.

2. Check another shader from: `https://www.shadertoy.com/view/4sfyRM`. This time the sphere is illuminated by 50 lights, which are the samples from the original illumination map. The cube map (or illumination map) is sampled according to potential contribution of each texel to the final illumination: texels that are brighter are more likely to be sampled. This is called importance sampling.

   (a) Modify the parameters of the Phong reflection model. Try to achieve more or less glossy materials.

   (b) Made a small change in the shader to replace the Phong BRDF with the Phong-Blinn BRDF. Observe the difference in the rendered results.

(c) (Optional) Replace the Phong reflection model with another BRDF of your choice (Ward, Cook-Torrance).

2 Global illumination

The question for the global illumination lecture can be found in the second supervision sheet `http://bentonian.com/Lectures/AdvGraph1617/Exercises-2.pdf`.

3 Light and colour

1. It is recommended that you complete this exercise in Matlab or GNU Octave. Clone the github repository `https://github.com/mantiuk/advanced_graphics` and check `colour/compute_rgb.m`. The script loads an emission spectrum from one of the files `spec_?.csv`. Complete the code so that it computes corresponding linear RGB values using the colour matching functions from `cie_xyz_1931.csv`. Write down what colours are represented by the spectra in the files `spec_?.csv`.

   Note that you can use a matlab function `trapz` to compute a discrete approximation of an integral.

2. Note that the values for red, green and blue components, which you computed in the previous question, can be negative. What does it mean if a color component is negative?

3. Two printed colour patches result in a metameric match for a standard observer. What can be said about:

   (a) The CIE XYZ trichromatic values for both patches;

   (b) Their spectral reflectance;

   (c) Their perceived colour when seen under different illumination than one producing the metameric match.

4. Digital cameras often have a white balancing option to correct colours in captured images. Why is this option needed? What perceptual phenomenon does the white balance operation compensate for?

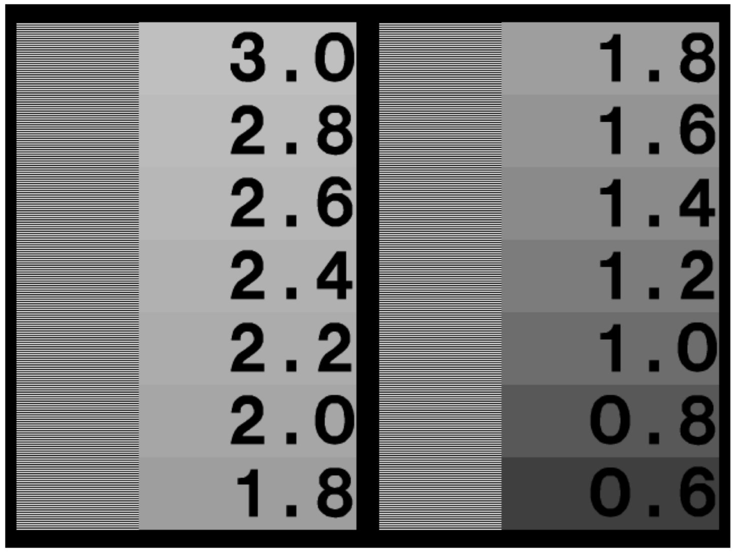

Figure 1: A chart that can be used to visually estimate the *gamma* of a display. Note that you may see aliasing artefacts in the lines unless the figure is enlarged to a sufficiently large resolution.

5. (Optional) A gamma of a display can be found by matching the brightness of alternating white and black lines with the brightness of a uniform gray-level, such as in the gamma chart shown in the Figure 1. Derive a formula for computing the exponent "gamma" given luma (pixel value) of the gray level that matches brightness of the white and black lines.

   Assume that the display black level is 0 and the formula for transforming luma into luminance is $L = V^\gamma$, where L is luminance, V is luma and γ is the "gamma" exponent.

4 Advanced image processing

It is recommended that you complete this exercise in Matlab or GNU Octave.

Clone the github repository `https://github.com/mantiuk/advanced_graphics` and check `grad_domain/grad_domain_example.m`. It contains a code example showing how to do image processing in the gradient domain. Since setting the matrices with a sparse linear system can be tricky, most of the code has been provided for you. However, you are encouraged to analyze the code. The example performs contrast enhancement in the gradient domain by boosting gradients of magnitude close to 0.1. Make the following modifications:

a) You should notice "pinching" artefacts in the resulting image. Design a matrix with weights to reduce or eliminate those artefacts in the result.

b) Experiment with other operations in the gradient domain. For example, remove a flare of a flag pole from the example image by setting gradient values in that image region to 0.