CS 435 - Computational Photography

Assignment 2 - High Dynamic Range (HDR) Images

Introduction

Digital cameras are unable to capture the full dynamic range of real scenes (especially those with sunlight). We can capture the full dynamic range of these real scenes by taking multiple exposures.

HDR images have a large dynamic range. That is, pixels aren't limited to one of 256 values. In some cases, each pixel is represented by a single precision floating-point number.

In this assignment we'll look at combining images taken at different exposure lengths to form a *high dynamic range* (HDR) image. In addition, we'll look at how to convert an HDR image to a standard dynamic range (SDR) image.

Grading

Theory Questions	15pts
Plotting pixel value vs log exposure	15pts
Finding Response Curves	30pts
Generating HDR Images	20pts
Tonemapping HDR Images	20pts
TOTAL	100pts

Table 1: Grading Rubric

1 (15pts) Theory Questions

1. (5pts) Apply a 3×3 mean filter to the following 2D matrix. You may assume that the filter is only applied to areas of the data that have a full 9 samples to process. Feel free to use Matlab to help you compute this, however, realize that you may be asked to do this without a calculator on an exam.

$$\begin{bmatrix} 7 & 7 & 6 & 3 & 3 & 4 & 2 & 2 \\ 3 & 7 & 2 & 6 & 4 & 4 & 5 & 7 \\ 5 & 4 & 7 & 5 & 1 & 1 & 2 & 2 \\ 2 & 1 & 3 & 4 & 1 & 3 & 5 & 6 \\ 6 & 2 & 2 & 7 & 4 & 2 & 5 & 4 \\ 2 & 2 & 2 & 3 & 6 & 6 & 6 & 7 \\ 4 & 6 & 5 & 6 & 7 & 3 & 4 & 1 \\ 5 & 2 & 4 & 6 & 1 & 4 & 1 & 4 \end{bmatrix}$$

2. (5pts) What is the kernel function for a 5×5 Gaussian function with $\sigma = 1$? Normalize the filter so that its elements sum to one. Feel free to use Matlab to help you compute this, however, realize that you may be asked to do this without a calculator on an exam (leaving things in terms of e).

3. (5pts) Given the following 2D kernels, what is the magnitude and direction of the gradient at the center pixel in I? Feel free to use Matlab to help you compute this, however, realize that you may be asked to do this without a calculator on an exam.

$$\frac{\partial}{\partial x} = \begin{bmatrix} -1/3 & 0 & 1/3 \\ -1/3 & 0 & 1/3 \\ -1/3 & 0 & 1/3 \end{bmatrix} \tag{1}$$

$$\frac{\partial}{\partial y} = \begin{bmatrix} -1/3 & -1/3 & -1/3 \\ 0 & 0 & 0 \\ 1/3 & 1/3 & 1/3 \end{bmatrix} \tag{2}$$

$$I = \begin{bmatrix} 7 & 7 & 6 \\ 3 & 7 & 2 \\ 5 & 4 & 7 \end{bmatrix} \tag{3}$$

2 (15 points) Plotting pixel value vs log exposure

On BBlearn you have been provided with a directory, *memorial*. This directory contains a file *images.txt* that provides a list of images in that directory, as well as their exposure lengths. Your first task will be to parse the image.txt file to get the list of file names and associated exposure times, and then load all the images in that directory.

Next, select *three* pixel locations and plot the values in their *red channel* as a function of Δt_j, where Δt_j is the exposure length for image j. This is akin to plotting the *log irradiance* as a function of the exposure length, but with an identity log irradiance function.

You image should look something like Figure 1.

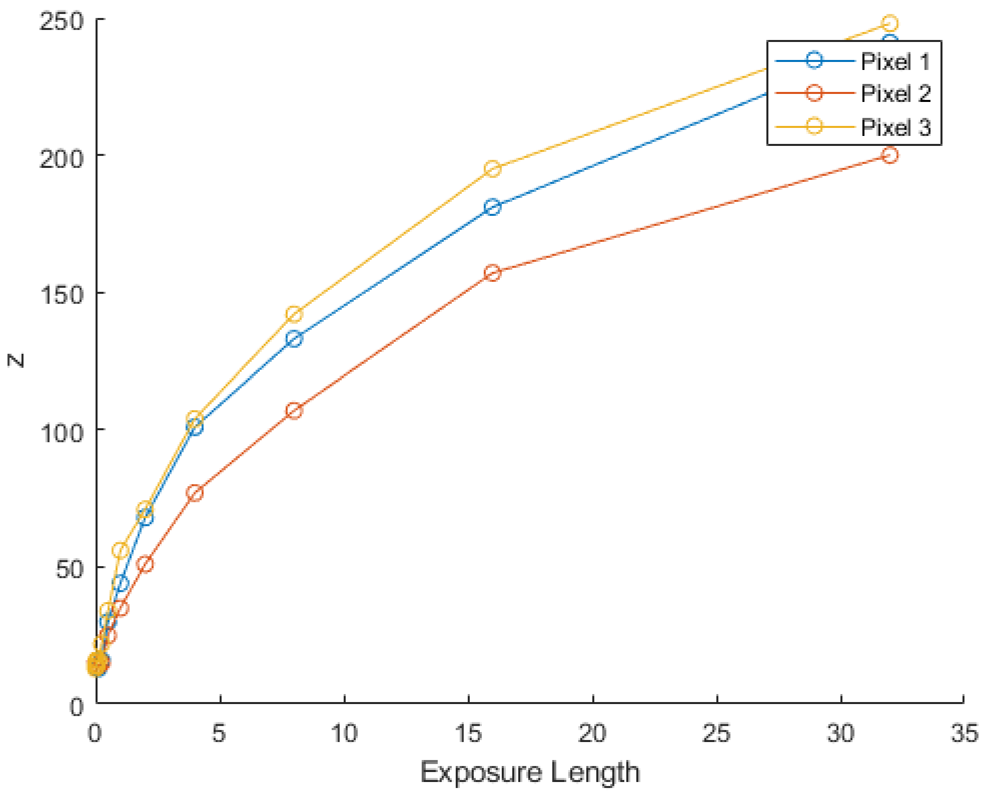

Figure 1: Observed red intensity vs Exposure Length

3 (30 points) Finding and plotting the Log Irradance Functions

Using the technique discussed in class, find the log irradiance function $g(z_{ij})$ for each color channel. Then repeat the plot from the previous section. You image should look something like Figure 2.

NOTE: The more pixels you use to solve the system the better. That being said, the more pixels you use the larger the matrix to invert will becomes. Therefore experiment with how many pixels to use.

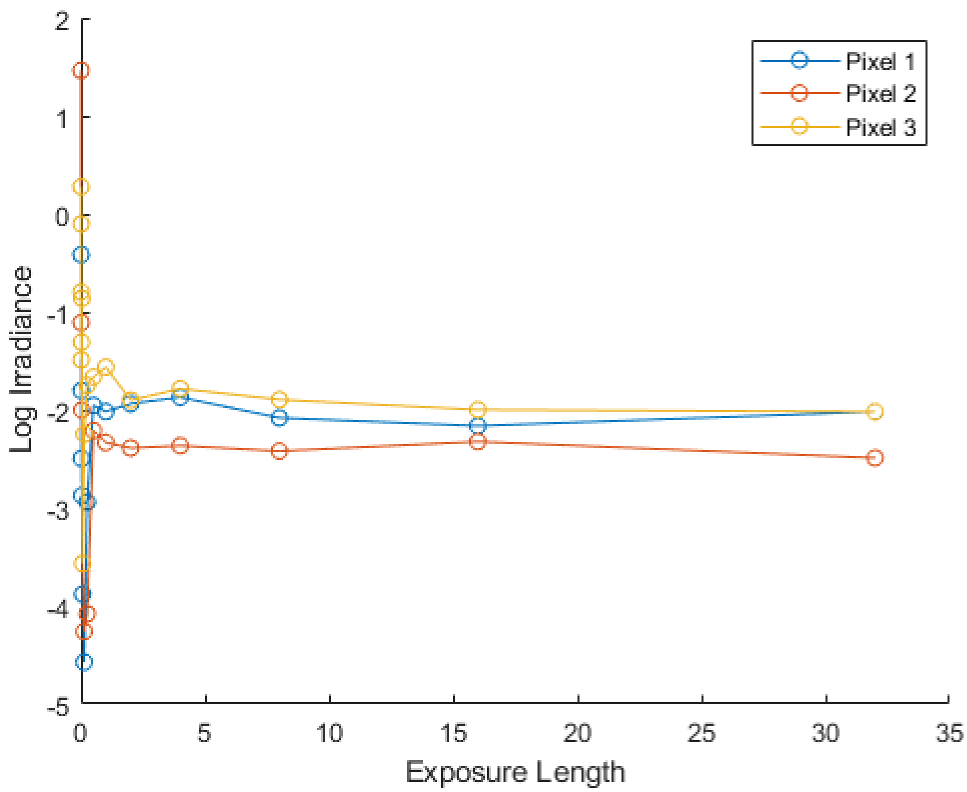

Figure 2: Log Irradiance vs Exposure Length with learned log irradiance function

4 (20 points) Generate HDR Images

Now that we have our log irradiance functions we can combine our images taken with different exposure times!

For each color channel, go through all the pixel locations and compute the new pixel value to be the average of the pixel's irradiance values from the different exposure length images (making use of the associated channel's log irradiance function, and that image's exposure time).

As a quick reference, from the lecture slides, the equations to do this are:

$$ln(R_i) = \frac{1}{P}\sum_{j=1}^{P}(g(z_{ij}) - ln(\Delta t_j)) \tag{4}$$

$$R_i = e^{ln(R_i)} \tag{5}$$

5 (20 points) Tone Mapping an HDR Image

HDR images cannot be viewed on devices that only support low dynamic range. To be able to view all the details of dark and bright areas at once, the image must be *tonemapped*.

Tonemap each channel of your HDR image by compressing its values using the compresion function $f(x) = \frac{x}{1+x}$, then scaling its values to $[0, 255]$, then casting it as an unsigned 8-bit integer.

Submission

For your submission, upload to Blackboard a single zip file containing:

1. PDF writeup that includes:
   (a) Your answer to the theory question(s).
   (b) Your plot for Part 2.
   (c) Your plot for Part 3.
   (d) Your HDR image for Part 4.
   (e) Your HDR$\rightarrow$SDR tonemapped image for Part 5.
2. A README text file (**not** Word or PDF) that explains:
   (a) Features of your program
   (b) Name of your entry-point script
   (c) Any instructions on how to run your script
3. Your source files.