15-663 Homework Assignment 2 Report

Jingguo Liang

1 HDR imaging

Develop RAW images

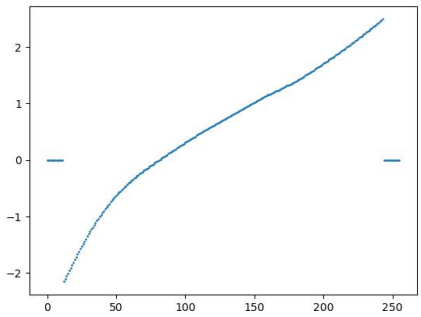
I use the following command to convert the RAW `.NEF` images into `.tiff` images:

```
.\dcraw.exe -w -o 1 -q 3 -4 -T .\exposurex.NEF
```

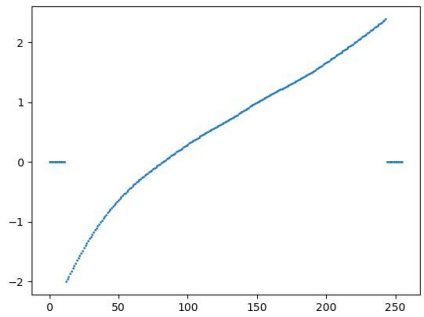
Where '`x`' in the filename is replaced by indices of the images.

Linearize rendered image

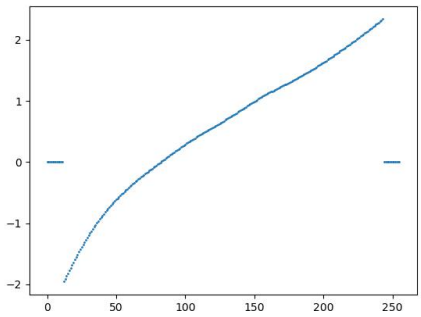
I sample from the images uniformly with distance between two neighbouring samples being 250 pixels. Figure 1 shows the g functions recovered under four weighting schemes.



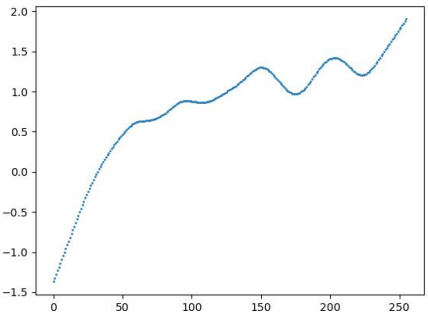
(a) Uniform weight



(b) Tent weight



(c) Gaussian weight



(d) Photon weight

Figure 1: Function g under different weighting schemes

Weighting schemes

I use the recommended $Z_{\min} = 0.05$ and $Z_{\max} = 0.95$.

Make your pick

I choose the setting using rendered images input, logarithmic merging, and tent weighting for the following reasons:

1. There seems to be an issue with the RAW input such that some patches on the color checker looks very black. This creates a problem in the following color correction step. In rendered images such problem is not observed.

2. I really cannot tell the difference of different merging methods and weighting schemes by just looking at the intermediate results. Thus, I decide to go with whichever method used by Debevec and Malik in their paper, which turns out to be tent weighting and logarithmic merging.

2 Color correction and white balalncing

Figure 3 shows the intermediate result presented using scale-and-clip method after color correction and white balancing, with scales being 1 and 0.02, respectively. The white balancing is performed such that in patch 19 (the white one located bottom right in the image), the average of R and B channel is scaled respectively to be equal to that of G channel. This mitigates the orange looking in the original image, but makes the image look cold and pale. I personally perfer neither of them, so I took a third approach: I perform only the color correction, but no white balancing. I find the resulting style of this most pleasing. Check **??** for the final tonemapped image using this scheme.

3 Photographics tonemapping

Figure 3 is the tonemapped image with $K = 0.025$, $B = 0.85$, using RGB tonemapping method. And figure 4 is the same image, tonemapped using xyY method, with $K = 0.1$, $B = 0.85$. While the one with xyY mapping scheme looks richer in color, it looks much less realistic than the one with RGB mapping scheme. Thus, I prefer the RGB tonemapping method.

4 Create and tonemap your own HDR photo

Figure 5 is one image from the stack of images taken in a spot in campus with different shutter speeds. The shown one has shutter speed of 1/8 second, and the whole stack consists of 11 images with the following shutter speeds: 1/1000 sec, 1/500 sec, 1/250 sec, 1/125 sec,

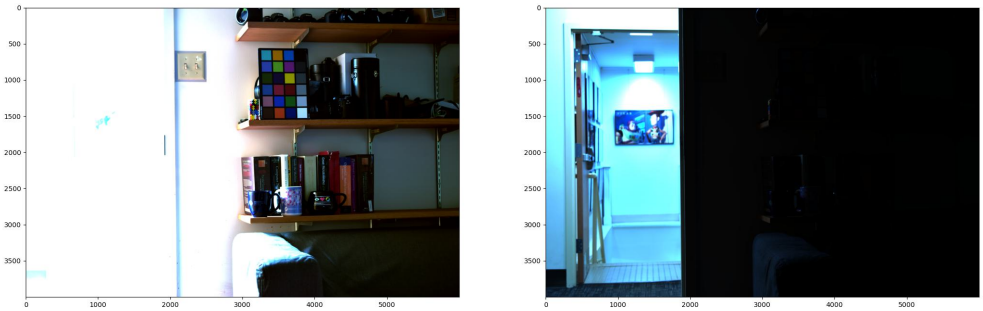


Figure 2: Intermediate result after color correction and white balancing

1/60 sec, 1/30 sec, 1/15 sec, 1/8 sec, 1/4 sec, 1/2 sec, 1 sec. The apeture is f/16, ISO is 100, and the direct sunlight white-balancing preset is used.

I merge the LDR images using the same settings as in previous sections, and figure 6 are the tonemapped HDR images using two tonemapping methods, both with $K = 0.2$ and $B = 0.85$. In this scenenario, I prefer the xyY mapping rather than the RGB mapping because the richer color makes outdoor environment look better (sky, grass, trees, etc.).

5 Noise calibration and optimal weights

Noise calibration

Figure 7 shows the histogram of the pixel values of 3 pixels across 50 dark-frame-subtracted images, located at $(2000, 1500, 0)$, $(2000, 3000, 1)$, $(2000, 4500, 2)$ respectively. We observe that those values have shapes of either Poisson distribution or normal distribution. This is because those two distributions are the ones that noises in the images will follow. In brighter pixels, photon noise, which follows a Poisson distribution, is dominant. While in darker pixels, read noise and ADC noise, which follow normal distribution, will be dominant.

Figure 8 shows the mean-variance plot (blue) and the line fitted to it (red). Its slope, which is the gain, is 0.09081608059983558, and the intercept, which is the additive noise variance, is 0.2669777268833028.

Figure 3: RGB-tonemapped image with color correction but no white balancing, K=0.025, B=0.85

Merging with optimal weights

Figure 9 shows the tonemapped image using optimal weight. It looks very strange, with the tree trunks being completely black. I'll stick with the image with tent weight.

Figure 4: xyY-tonemapped image with color correction but no white balancing, K=0.1, B=0.85

Figure 5: LDR campus image, shutter speed 1/8 sec

(a) RGB-tonemapping (b) xyY-tonemapping

Figure 6: Tonemapped HDR campus images, K=0.2, B=0.85

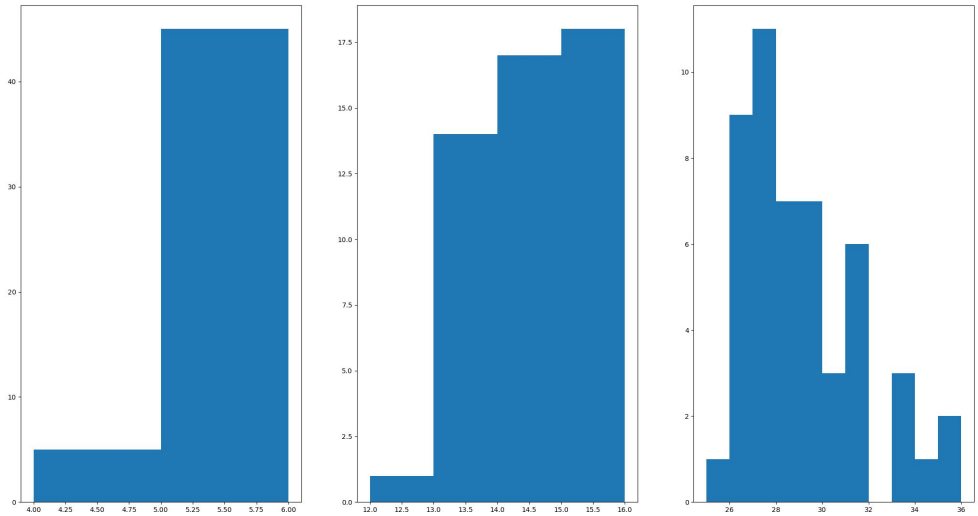


Figure 7: Histogram of 3 pixels, after dark frame subtraction

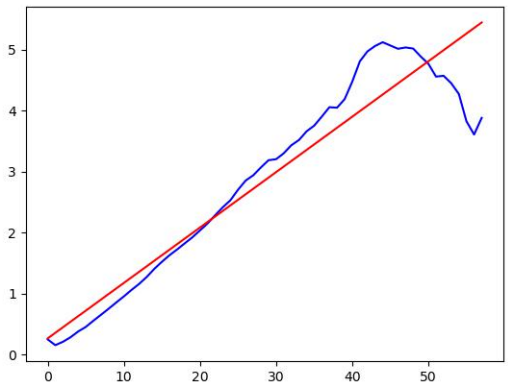


Figure 8: Mean-variance plot and the fitted line

Figure 9: Tonemapped image using optimal weight