CSCE 689: Computational Photography, Spring 2019
Homework 4

Ankur Roy Chowdhury
ankurrc@tamu.edu

April 20, 2019

Algorithm

For recovering the *Camera Response Function (**CRF**)*, we essentially need to ascertain the relation between the exposure (H) and the mapped pixel value (Z).

Preliminary

The following notation is established to represent the respective quantities:

1. $$Z_{ij} = f(E_i \Delta t_j) \tag{1}$$

   where f is the function mapping the irradiance of i-th pixel in the j-th image, captured with shutter speed $\frac{1}{t_j}$ to the pixel value, Z_{ij}.

2. Inverting the relation, and taking log on both sides, we get the following relation:

   $$\log f^{-1}(Z_{ij}) = g(Z_{ij}) = \log E_i + \log \Delta t_j \tag{2}$$

3. Our objective function then becomes:

   $$argmin_{g,E_i} \sum_{i=1}^{N} \sum_{j=1}^{P} [g(Z_{ij}) - \log E_i - \log \Delta t_j]^2 \tag{3}$$

   for N pixels and P images.

4. To ensure smoothness, we regularize the terms in g. We take the second derivative of each term in g, such that the local neighbourhood around each term is encouraged to be smooth.

   $$argmin_{g,E_i} \sum_{i=1}^{N} \sum_{j=1}^{P} [g(Z_{ij}) - \log E_i - \log \Delta t_j]^2 + \lambda \sum_{z=Z_{min+1}}^{Z_{max-1}} g''(z)^2 \tag{4}$$

5. To account for uncertainty around the extrema of the pixel values, we introduce a weighting function (w) such that pixel values around the center are given more weight than the ones around the extremes:

$$w(z) = \begin{cases} z - Z_{min}, \text{ if } z \leq (Z_{max} - Z_{min} + 1)/2) \\ Z_{max} - z, \text{ otherwise} \end{cases} \tag{5}$$

6. Finally, we incorporate the weighting into the objective function as:

$$argmin_{g,E_i} \sum_{i=1}^{N} \sum_{j=1}^{P} [w(Z_{ij})(g(Z_{ij}) - \log E_i - \log \Delta t_j)]^2 + \lambda \sum_{z=Z_{min+1}}^{Z_{max-1}} [w(z) g''(z)]^2 \tag{6}$$

Steps

1. **Load Images and Exposure times**
   (a) Sample N random pixel locations, such that:
   $$N > \frac{(Z_{max} - Z_{min} + 1)}{P - 1}$$
   (b) Also, store the exposure time Δt_j for each image.
2. **Build Linear System**
   (a) Loop through all the images and get the Z_{ij} values according to the sampled pixel locations.
   (b) Also, get the corresponding exposure time Δt_j.
   (c) Construct the linear system of equations by filling in each entry of equation 6.
3. **Solve Linear System**
   (a) Solve the linear system using SVD.
4. **Reconstruct Radiance Map**
   (a) Using the g values obtained, use the following equation to compute the relative radiance of each pixel in the image:
   $$L_i = \frac{\sum_{j=1}^{P} w(Z_{ij})[g(Z_{ij}) - \log \Delta t_j]}{\sum_{j=1}^{P} w(Z_{ij})}$$
5. **Tonemap**
   (a) For the obtained Radiance map, use the global tonemapping operation using the Reinhart equation:
   $$Z' = \frac{\alpha L}{1 + \alpha L}$$
6. **Repeat**
   (a) Repeat step 2-5 for all channels.
7. **Merge**
   (a) Depth-wise merge the results obtained from step 5, we get the final HDR image.

Challenges

The following challenges were faced:

1. The regularization factor λ had to be discovered.

2. Global tonemapping does not give the best results, as a result of which is was crucial to set an image-wise α parameter that saturates pixel values to lower regions for smaller α values and higher regions for higher values.

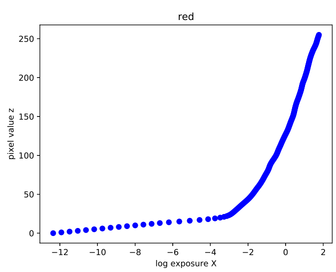
3. Had to be careful while avoiding the divide-by-zero error.

Results

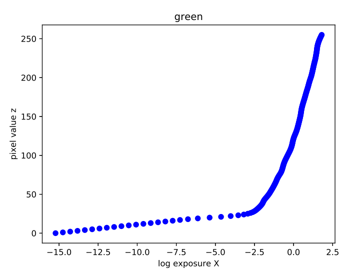
0.1 CRF

0.1.1 Camera 0

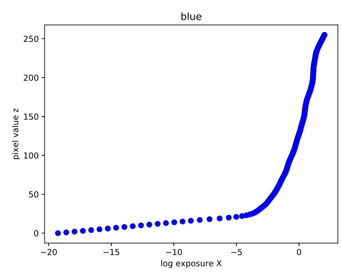
Fig. 1 shows the CRF for each channel of camera 0.



(a) Red



(b) Green

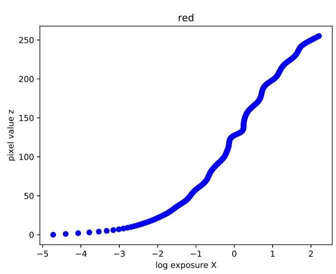


(c) Blue

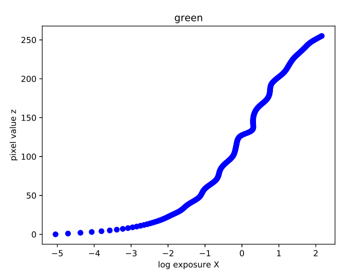
Figure 1: CRF for R, G and B channels. $\lambda = 20$

0.1.2 Camera 1

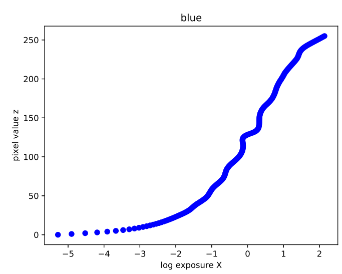
Fig. 2 shows the CRF for each channel of camera 1.



(a) Red



(b) Green



(c) Blue

Figure 2: CRF for R, G and B channels. $\lambda = 20$

0.2 HDR Reconstructions

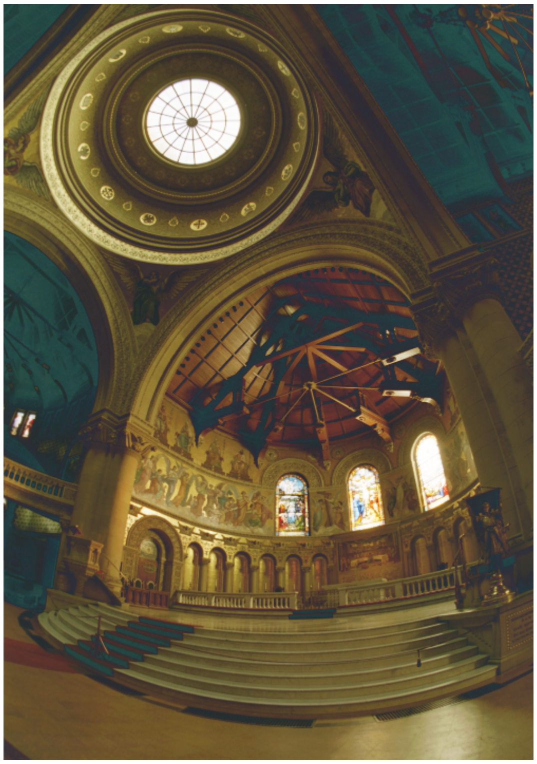
Local tone mapping was achieved using OpenCV's Durand algorithm. Parameters for Durand are: $(\gamma, contrast, saturation, sigma\_space, sigma\_color)$.

(a) Global $alpha = 1$

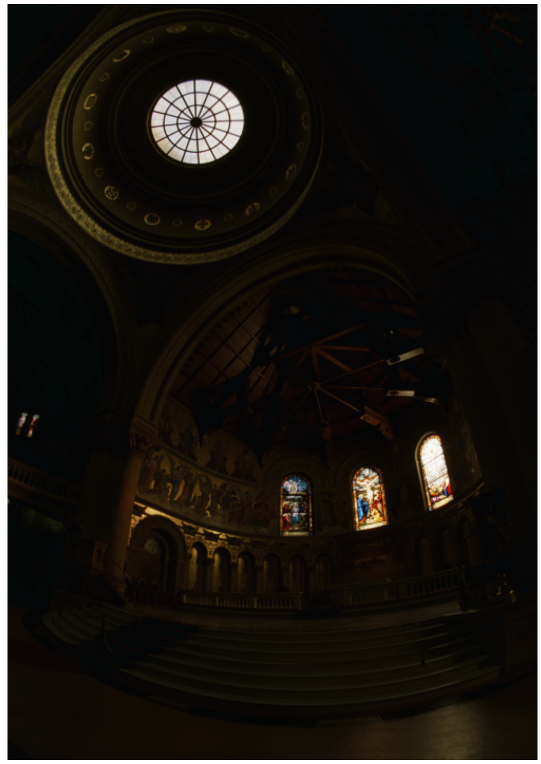
(b) Local $(1.5, 4, 1.0, 1, 1)$

Figure 3: HDR reconstruction for image 1.

Note: The locally tonemapped image appears too saturated for the red channel primarily because of a poor choice of parameters in Durand's algorithm. Because in the program I did not expose an interface to change the parameters, we have the not-so-great looking HDR image.



(a) Global $alpha = 1$



(b) Local $(1.5, 4, 1.0, 1, 1)$

Figure 4: HDR reconstruction for image 1, using CRF 1.

HDR reconstructions for image 1 using the CRF for camera 1, we get an image that does not capture the relative radiance map well because we effectively use the wrong mapping function to go from pixel space to radiance space. The local tone-mapper does not fair well either.

The rest of the results are as expected. Better 'looking' pictures for locally tone-mapped images than the globally tone-mapped ones.

(a) Global $alpha = 0.1$

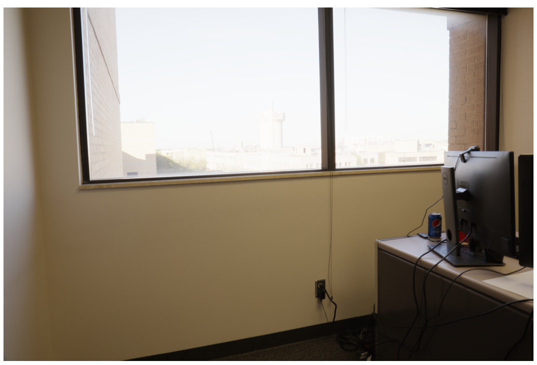
(b) Local $(1.5, 4, 1.0, 1, 1)$

Figure 5: HDR reconstruction for image 2, using CRF 1.

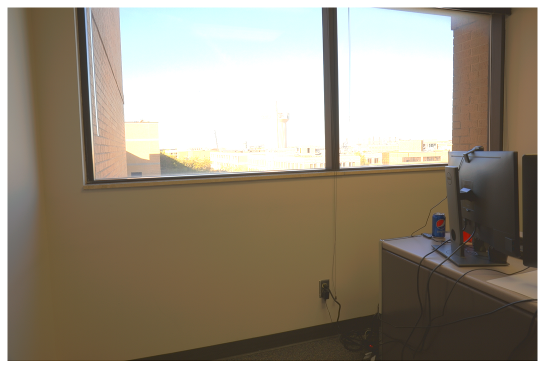
(a) Global $alpha = 0.1$

(b) Local $(1.5, 4, 1.0, 1, 1)$

Figure 6: HDR reconstruction for image 3, using CRF 1.



(a) Global $alpha = 0.1$



(b) Local $(1.5, 4, 1.0, 1, 1)$

Figure 7: HDR reconstruction for image 4, using CRF 1.