

# Dynamic Range Imaging

R08922144 翁紫瑄\* R08922A02 張碩恩\*  
r08922144@ntu.edu.tw r08922a02@ntu.edu.tw

April 2020

## 1 Introduction

This is a homework about reconstructing the LDR image from images with different exposures.

## 2 Alignment

We simply implement the median threshold bitmap (MTB) alignment algorithm [3] for aligning the images. First, we convert the RGB images into grayscale images, then we design a recursive function to evaluate the offset. The size of the input image will be divided by 2 until either it's height or weight is smaller than 200. An example of alignment is given in Figure 1. A tripod was used in the process of photographing, so there is not too much difference between the one with alignment and the other without alignment.

We also found that if we implement the alignment algorithm on some of the images, the result might become shaking. As seen in Figure 2. We inference the reason is that in each images, there are lots of people walking in front of the library, this may influence the error rate in the alignment algorithm.

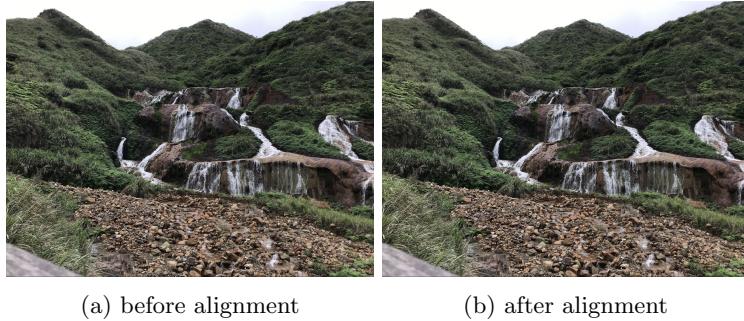
## 3 Inverse mapping function recovery

We follow the method in [1] to do this part. First of all, we construct the matrix and do the least squares approximation. For the purpose of recovering the inverse of the mapping function  $f^{-1}$ , we subsample the pixels by reshaping the images into 10 by 10. That is because we find out that it is a more robust way to choose pixels compare to random choosing 100 pixels in the image. To optimize the curve, the 127th dimension of  $f^{-1}$  is fixed to 0. The weight matrix we adopt is as Eqaution 1, which is depends on the unreliability of pixels with large or small intensity.

$$w_i = \min(i, 256 - i) \quad (1)$$

---

\*equal contribution



(a) before alignment (b) after alignment

Figure 1: Result after alignment.



Figure 2: Our results of NTU library with and without alignment.

We show our evaluated  $f^{-1}$  in Figure 3. To force the response curve be smooth,  $\lambda$  is set to 50. The result shows that the curves of three channels almost overlap with each other. We also plot out our recovered radiance map. As shown in Figure 4. The green radiance distribution shows the highest value over the mountain and the grass. This can be easily understood because the colors of the mountain and the grass are green.

## 4 Tone mapping

We implement the spatial (global/local) method in [2], which first employs a global mapping on the luminance, and then employs a local mapping. The equation of the global mapping in [2] are:

$$\bar{L}_w = \frac{1}{N} \exp\left(\sum_{x,y} \log \epsilon + L_w(x,y)\right) \quad (2)$$

$$L(x, y) = \frac{a}{\bar{L}_w} L_w(x, y) \quad (3)$$

$$L_d(x, y) = \frac{L(x, y)(1 + \frac{L(x, y)}{L_{white}^2})}{1 + L(x, y)} \quad (4)$$

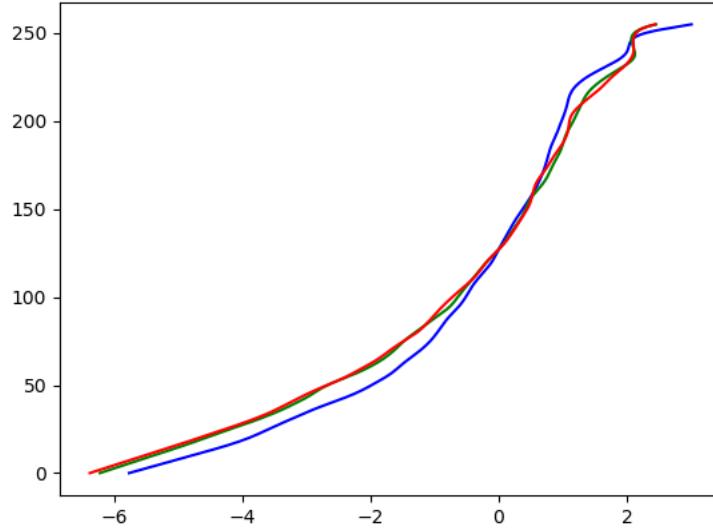


Figure 3: Our inverse function  $f^{-1}$  of  $f$ . Different colors represent different channel of  $f^{-1}$ . red curve: B channel, green curve: G channel, blue curve: R channel.

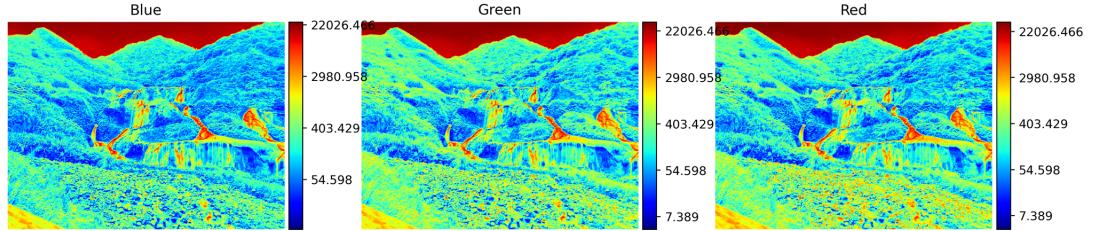


Figure 4: The radiance distribution over the scene. Left: B channel, Middle: G channel, Right: B channel.

To obtain the luminance, we adopt Equation 5. Figure 5 shows some results after employing global mapping, which can find out that details in the images remain invisible. Local mapping is then introduced.

$$L_w = 0.114L_B + 0.587L_G + 0.299L_R \quad (5)$$

We implement GaussianBlur on the radiance map  $L$  with different sizes of kernel 1, 3, 5, 7, 9, 11, 13, 15, then find the maximum size  $s_{max}$  of kernel which

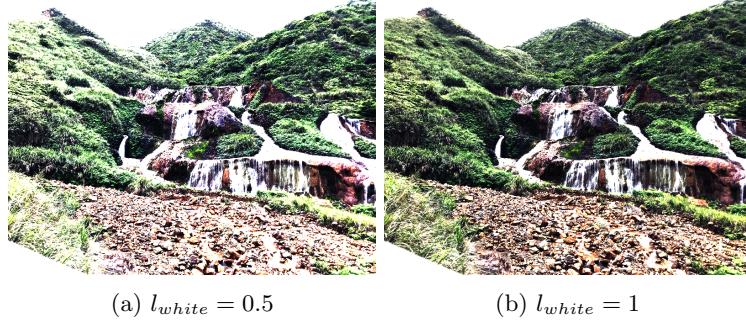


Figure 5: Results after employing the global operator with different values of  $L_{white}$ . Both with  $a = 0.5$ .

can constraint Equation 7 under the threshold  $\epsilon$ . In the following parts,  $\epsilon$  is set to 0.05 and  $\phi$  is set to 8. The results of local mapping are shown in Figure 6.

$$L_s^{blur}(x, y) = L(x, y) \otimes G_s(x, y) \quad (6)$$

$$V_s(x, y) = \frac{L_s^{blur}(x, y) - L_{s+1}^{blur}(x, y)}{2^{\phi} \frac{a}{s^2} + L_s^{blur}(x, y)} \quad (7)$$

$$L_d(x, y) = \frac{L(x, y)}{1 + L_{s_{max}}^{blur}(x, y)} \quad (8)$$

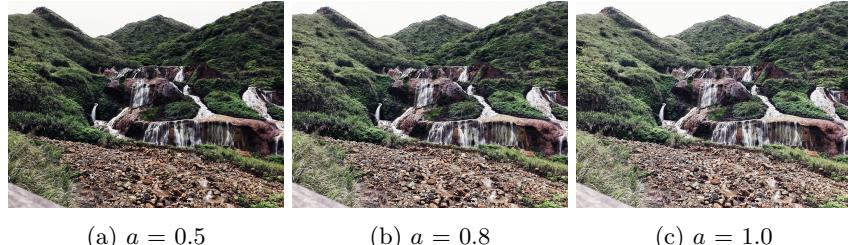


Figure 6: Results after implementing tone mapping with different value of  $a$ .

## 5 Other experiments

We test the code on 10 different datasets which were taken by Apple iPhone 7 plus and Nikon D90. Each of them can get a better picture after we do the HDR reconstruct. All of the results can be found in the folder "result", which has the dataset name following different value of  $a$  for the name of picture.

## References

- [1] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, page 369–378, USA, 1997. ACM Press/Addison-Wesley Publishing Co.
- [2] Erik Reinhard, Michael M. Stark, Peter Shirley, and James A. Ferwerda. Photographic tone reproduction for digital images. *ACM Trans. Graph.*, 21(3):267–276, 2002.
- [3] Greg Ward. Fast, robust image registration for compositing high dynamic range photographs from hand-held exposures. *Journal of Graphics Tools*, 8(2):17–30, 2003.