Lab Session 1
HDR Imaging

Computational Imaging. Universidad de Zaragoza.

Introduction

In this assignment, you will learn how to generate and display High Dynamic Range images. You will write software to automatically combine multiple exposures into a single HDR radiance map, a floating-point precision image that linearly map to scene radiance values. Then, you will implement different tone mapping algorithms to compress the dynamic range of the radiance map to a standard 8-bit range, allowing you to display the image on a screen.

The three sections of this assignment corresponds to the three papers you will have to (partially) implement [1] [3] [2]. We provide you with a very brief overview of the method in this document, but you will have to carefully read the pointed sections of the original papers in order to implement them.

You will first apply your work on the set of provided image in the data folder : the name of each image reflects the exposure used to capture it as a fraction. Then, you will have to capture your own image and apply your pipeline to it.

1 HDR imaging (40 points)

In the first part of this assignment, you will transform a stack of images taken at different exposures into a single HDR image. To do so, you will have to partially implement the method from Debevek and Malik [1].

1.1 Linearize images (30 points)

Unlike the radiance map we want to obtain, the rendered JPEG images are not linear. The first step is thus to linearize the images. To do so, you will have to find the camera response function g as explained in Section 2.1 of the paper. Below is very brief explanation of the process, refer to the paper for more details.

Let Z_{ij} be the observed pixel values for pixel i in image j taken with exposure Δt_j. This observed value is an unknown function of the scene radiance E_i at pixel i and the exposure duration:

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

.

Both f and E_i are unknown, but we will use the fact that E_i is constant over our stack of images to retrieve f. More precisely, instead of directly solving for f, we will solve for $g = ln(f^{-1})$ which maps the pixel values to the logarithm of the exposure values (Equation 2 in [1]).

We can recover g by solving a least square problem involving the pixels from our different exposure images (Equation 4 in [1]). In this linear system, you will have to solve for the value of g over all possible pixel values as well as the scene radiance at each pixel.

Once we have the response function g, it is straightforward to map the pixels from each picture to the log radiance of the scene :

$$ln(E_i) = g(Z_{ij}) - ln(\Delta t_j)$$

In the report, plot the response function obtained with the full method but also without the smoothness term (you can use a very low λ) and without using the weighting scheme. Comment on the functions that you obtained with these different parameters.

Note that you will have to subsample the images in order to solve the least square problem. You can regularly sample the images every 20 pixels for example when constructing your least square problem.

1.2 Obtain the radiance map (10 points)

Now that you can get the radiance from each image, you can retrieve a HDR image by averaging the values from all the images from the stack. This is described in Section 2.2 of the paper.

Reconstruct the radiance of the scene and store it as a HDR image. Again, try with and without the tent weighting scheme and analyze the difference in the final radiance map.

In the report, show the radiance map obtained with and without weighting (you can use the `imagesc` function).

2 Global tone mapping (20 points)

In this section, you will implement the global tone mapping operator from Reinhard et al. [3] to be able to display the HDR image you just constructed. Read carefully Section 3 from the paper and implement their method.

As you can see, this algorithm is an improved version of the basic tone-mapping operator $\frac{L}{1+L}$. In the report, show how the tone mapped image differs from using this simple operator, experience with different parameters and show how they influence the final image.

3 Local tone mapping (30 points)

Finally, you will implement a local tone mapping algorithm : the operation applied on each pixel in not exactly the same and depends on local contrast and details. You will implement a simplified version of Durand et al. [2]. The paper in itself focus a lot on the implementation of a fast bilateral filter but you can find a brief description of their method in Section 6.

In practice, you will extract a base layer from your image by filtering the intensity with a bilateral filter and apply contrast reduction in the log domain. Then, you will bring back the details that were removed by the filter and the colors. The intensity can be computed as the average of all color channels. Be careful that in the linear domain, you will have to divide (or multiply) to isolate the colors of your image and get the intensity back; whereas in the log domain, you will have to subtract (or add) to isolate the details and put them back.

The contrast reduction operation is a scaling of the image

$$I' = (I - o) * s$$

where $o = max(I)$ and $s = dR/(max(B) - min(B))$. The offset o is such that the maximum intensity will be 1 in the linear domain, the scale s is such that the output intensity has dR stops of dynamic range. In practice, you can experiment with values between 2 and 8.

For the kernel sizes of the bilateral filter, you can use the values recommended in the paper (σ_s and σ_r).

In the report, show the decomposition of the image (base and detail layer) as well as the tone mapped images. Experiment the contrast reduction operator applied naively on the whole image (without using the filtering to conserve details) and compare with the result from the full method.

4 Try out with your own pictures (30 points)

Now that you have all the methods to generate a HDR image from a stack of LDR images and tonemap it, try with your own pictures. First, choose a scene with high dynamics : typically any scene that contains both indoor and outdoor (a room with a window) or two places with very different illuminations (an illuminated hallway and a dark room), but also many outdoor scenes on a very sunny day. Be careful to choose a scene that requires HDR: if the LDR images already contain all the details because the dynamic range is low, it will have no interest.

Photograph the scene several times while varying only the shutter speed (your camera should be in Manual mode). Your camera should not move between the pictures so use a tripod or put your camera on any stable support (a table). I you don't have a camera that allows you to set up the shutter speed, you can borrow one from the lab.

In the report, show two or three representative images from your stack and explain why you choose this scene. Show the radiance map that you obtained from Section 1 and your favorite tone mapped image for both algorithm along with the parameters that produce them.

Deliverables

Your solution should be an archive (e.g., a ZIP file) that includes the following:

- All of your Matlab code, including commented code for all the pipeline stages, as well as a README file explaining how to use the code and commenting design choices, if any.

- A PDF report that includes final and intermediate results for each of the section. Comment on the different parameters we ask you to explore and explain what are their effects and which image you prefer.

Acknowledgements

This assignment is adapted from James Tompkin' Computational Photography course, Brown University and Ioannis Gkioulekas' Computational Photography course, Carnegie Mellon University. The provided images are taken from http://www.pauldebevec.com/Research/HDR/ .

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*, pages 369–378, 1997.

[2] Frédo Durand and Julie Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 257–266, 2002.

[3] Erik Reinhard, Michael Stark, Peter Shirley, and James Ferwerda. Photographic tone reproduction for digital images. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 267–276, 2002.