Implementation of a Pseudo-HDR Algorithm Using Cuda

GPGPU Course

Winter 2019

Javad Khashabi Zade

Erfan Bahmani

Problem Description

High-dynamic-range imaging is a high dynamic range technique used in imaging and photography to reproduce a greater dynamic range of luminosity than is possible with standard digital imaging or photographic techniques. The original algorithm requires multiple images of an exactly same scene with variable exposures.

But there are some techniques that can produce nearly (not exactly) same results on a single image with bad exposure range.

There are different algorithms to make it happen [1]. We chose **Histogram Adjustment** or **Histogram Equalization**. A technique which is also used in Adobe Photoshop.

The main purpose of this technique is to distribute the luminance variance in a better way.

Implementation Details

An overall look to the process would be like:

1- Loading the image
2- Calculating the histogram
3- Calculating the cumulative distribution
4- Applying the CDF to the input image

Loading the image is nothing by retrieving a 1d array of separated channels of pixels of the image.

The structure would be [r_1, g_1, b_1, r_2, g_2, b_2, …].

Next step is calculating the mentioned histogram. RGB color space has a limited range; So, it's not that suitable for distinguishing the differences between pixels. To overcome this problem, converting the space to xyY is good. What we need is Y which is known as the luminance. So, by keeping x and y and modifying the Y through the process, we can change only the luminance or brightness of a color in an effective way. As it's not possible to convert RGB to xyY directly, it's required to convert the RGB to XYZ and the to xyY. These conversions are done by some formulas which are accessible at [2].

By converting to xyY, calculating the histogram over Y becomes easy.

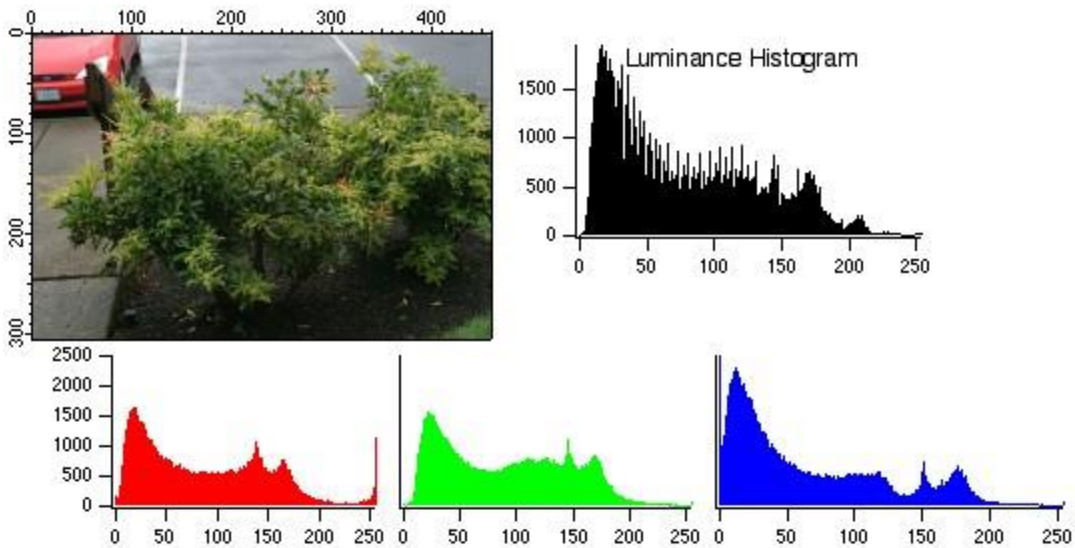
This calculation is done using equation:

$$\mathrm{lum\_range} = \mathrm{lum\_max} - \mathrm{lum\_min}$$

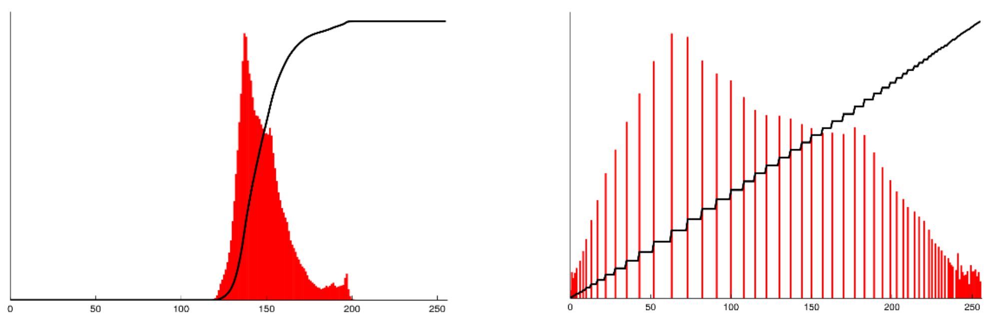
$$bin = \frac{pixel_{luminance} - lum\_min}{lum\_range} * bin\_count$$

lum_max and *lum_min* is maximum and minimum luminance values of pixels. *Bin_count* is the number of separations. By calculating the histogram, we classify pixels according to their luminance and the *bin* will be the index of each pixel's class.

To calculate *lum_max* and *lum_min*, we used reduction technique to make the process parallel as the nature of calculating min/max is sequential.



Next step is calculating CDF from current luminance histogram. This CDF helps to figure out how relevant a pixel of a class is, according to its previous classes.
By normalizing the CDF, we'll see all values are between 0 and 1 and if we consider them as luminance values, we'll see that they are adjusted according to darker pixels comparing to each pixel.



By retrieving Y values from CDF and recalculating the XYZ space, and then converting it to RGB, we have and image with adjusted luminance of each pixel according to a global luminance.

References

[1] Kuang, Jiangtao, et al. "Evaluating HDR rendering algorithms." *ACM Transactions on Applied Perception (TAP)* 4.2 (2007): 9.
[2] http://www.brucelindbloom.com/
[3] https://en.wikipedia.org/wiki/Histogram_equalization