Computer Vision
Lab 3: image processing

Riccardo De Zen. 2019295

4 May 2021

1 Introduction

This report covers the third lab experience, which was centered around image manipulation using histogram equalization and spatial filters.

2 Implementation

2.1 HSV Equalization

Histogram equalization was very straight forward for the BGR image. The only question left was whether a better equalization was possible or not. I tried using the HSV color space and equalizing only one channel, as suggested. I decided to equalize only the Value channel, since it roughly corresponds to the perceived luminance, and I assumed it would implicitly equalize all three color channels. I later found out that Hue and Saturation channels gave a quite poor overall equalization, confirming my previous assumption.

2.2 Code changes

I made a few changes to the provided code. First of all I changed the **`showHistogram`**: since having nine histograms was a bit confusing, I made it show all three channels in a single window. It also allows for a window suffix to avoid overwriting previous histograms. The other thing I modified was the **`Filter`** class. I did not like the idea of freezing the target image inside the filter object, but I also understand the issue of OpenCV callbacks taking a single pointer. This led me to define the following structure:

```
struct Userdata {
    Mat &current_image;
    Filter &filter;
};
```

Allowing me to keep the filter logic and the image separate, as they should be, while still passing the whole thing to the callback when needed.

3 Results and Discussion

As can be seen in Figure 1, the HSV equalization yields histograms that are closer to a uniform distribution. While the result may not seem impressive, Figure 2 shows that the color gamma appears much wider than BGR equalization. Quite frankly, I expected such a result from equalization over Hue or Saturation, which instead both yielded very dark images.

When experimenting with filters to remove noise I found that the Gaussian and Median filters do not preserve the edges, as expected. The image gets quite blurred even for small filter sizes (e.g. 2×2 Median and 3×3 Gaussian).

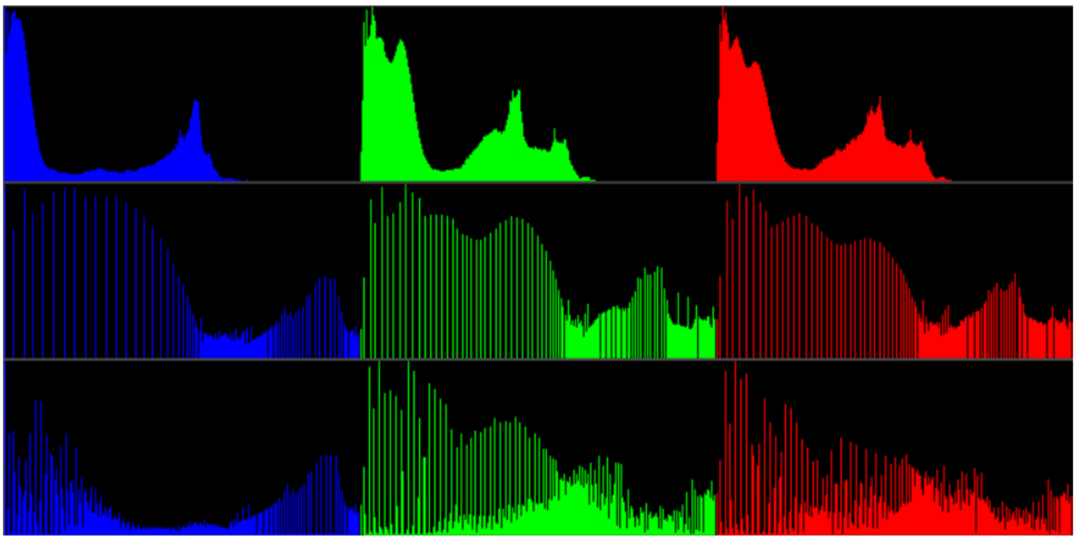

Figure 1: BGR histograms for **`image.jpg`**. Original (top), after BGR equalization (middle), after HSV equalization (bottom).

Figure 2: Result of image equalization on the same image of the above histograms. Original (left), BGR equalization (middle), HSV equalization (right).

Figure 3: Result of equalization on another image. Result is similar as Figure 2, with the HSV equalization yielding the most vivid colors.

The only interesting result are obtained with the Bilateral filter. While the effects are quite small for small sigmas, for higher values such as $\sigma_s = 40, \sigma_r = 40$ we can clearly see its effects in Figures 5 and onwards:

- Edges and heavily detailed areas do not suffer much, because they have a standard deviation which is much higher than average.
- Areas where noise is more moderate, the effects of the filter are more evident (e.g. roof tiles in 5). As sigmas increase further, the result becomes more and more similar to what one would expect from segmentation.
- Some finer details can be lost: in Figures 4 and 5 the electrical cables in the back slightly suffer the effects of the filter.

These results lead me to say that the Bilateral Filter is the most suited for a general purpose noise removal, as expected. This comes at the cost of a higher computational complexity.

While the results were good with images such as `image.jpg` (Figures 4 and 5) or `overexposed.jpg` (Figure 6), this was not the case with other images. An example is `coutryside.jpg` shown in Figure 7, where we can see that after both equalizations, the sky is quite pixelated. While the other images also had this feature, this one is also much more noisy to begin with, meaning even the bilateral filter would detect most of the content as noise and smooth it.

4 Notes

The size of the Bilateral filter had been initially set to be $6 \cdot \sigma_s$, as suggested, but this caused the computational times to explode. Setting the filter size to be fixed at 15×15 gave comparable results.

Figure 4: Smoothing with a $\sigma_s = 40, \sigma_r = 40$ bilateral filter after BGR equalization. Smoothing is evident inside uniform patches, while edges and details are preserved.

Figure 5: Smoothing with a $\sigma_s = 40, \sigma_r = 40$ bilateral filter after HSV equalization. Smoothing is evident inside uniform patches, while edges and details are preserved. Compared to the BGR image, grass and clouds are smoother.

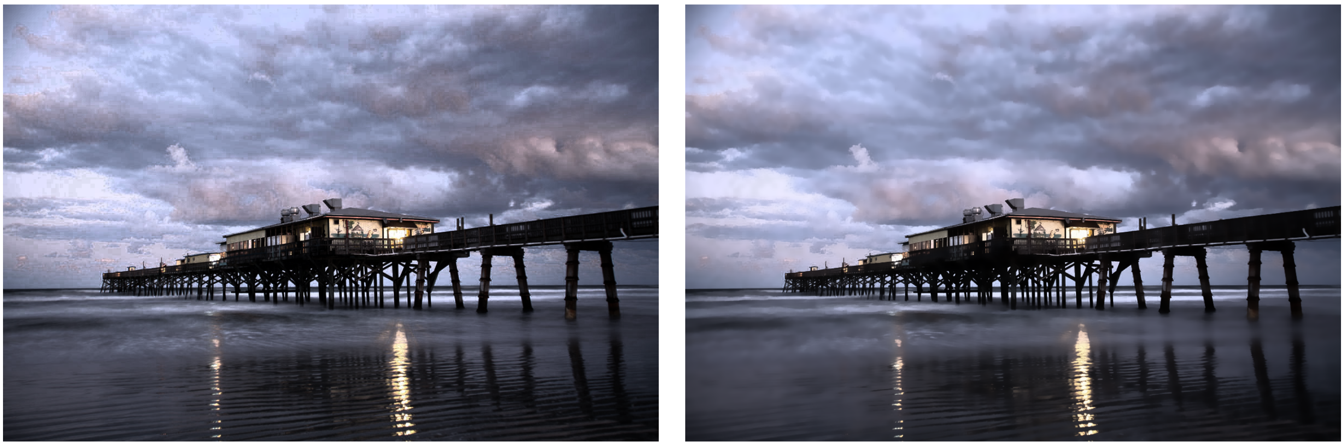

Figure 6: **`overexposed.jpg`** was not particularly noisy so most of the information is preserved.

Figure 7: **`countryside.jpg`** was quite noisy to begin with. We can see on the right that most of the ground and the leaves have been touched by the filter.