Lab 3 - Image Equalization, Histograms, Filters

Luca Scattolaro

15 April 2021

1 Report

To do this Laboratory I follow primarly the opencv online guide and the theoretical slides of the lab.
On the slide there are some hints about the way to procede and the functions to use to get the right results and to understand the meaning of all of them and also to use them properly I readed the guide.

Implementation choices
I decide to organize all the lab using a "Menu" in order to let the tester decide what to do during the execution.

```
choose 1 image among:
 0. ../data/countryside.jpg
 1. ../data/image.jpg
 2. ../data/lena.png
 3. ../data/overexposed.jpg
0
HSV equalization, chose:      (if you type a wrong number the default channel chosen will be V)
  0. H
  1. S
  2. V
2
How do you want to see the Images of Results and Histograms? Type:
  0. I don't want to see anything
  1. All togheter
  2. One image for Result
1
Chose Type of Filter:
  0: Exit
  1: Median
  2: Gaussian
  3: Bilateral
3
```

Figure 1: Menu view

In the first part you can choose which image to work with.
in the second part you can choose which channel among H, S and V to equalize.
The third one is just to get only one image with all the execution steps put together or 1 image for each step. (This is just for a better visualization (best choice is 1)).
The forth one allow the user to apply a filter on the equalized image using trackbars to change values of the filter's parameters.

1.1 Histogram Equalization

After the user had chosen the image to process I loaded it and then split it using $cv :: split()$ in order to get the values of the 3 channels (RGB).
For each channel I computed the histogram using the function $cv :: calcHist()$. After that I equalized all the 3 channel using the function $cv :: equalizeHist()$ and with $cv :: merge()$ I reconstructed the equalized image.
Then I convert the input image from RGB to HSV color space and I equalized only one channel among H,S,V chosen by the user through the menu. I reconstructed the image from the 3 channels and I reconverted it into the RGB color space.
You can see on the image below an example of the result obtained using "image.jpg" and by equalizing the channel V.
Left image: input image with its histograms.
Center image: equalized image (all the 3 channels R,G,B) and its histograms.
Right image: equalized image (only the channel V, in HSV color space) and its histograms.

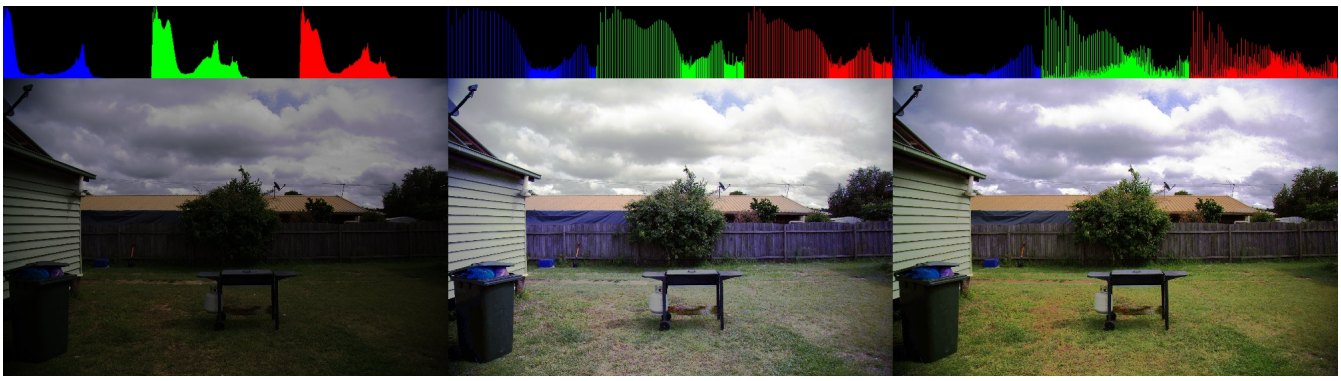

Figure 2: Result of Histogram Equalization

To being able to get this result I modify the showHistogram function: it return the image of the histograms.

1.2 Image Filtering

For the second part of the lab I used the class Filter. I modify it and I create also 3 subclass (1 for each filter we need to apply):

- MedianFilter
- GaussianFilter
- BilateralFilter

In the main for each filter I create trackbars in order to let the user modify the parameters of the filter choosen and show him the respective result.
The parameter that the user can modify through the trackbars are:
Median Filter:

- Kernel Size

Gaussian Filter:

- Kernel Size
- sigma

Bilateral Filter:

- sigma Range
- sigma Space

You can see on the image below an example of what you can do in this part:
Left image: example of the application of a median filter.
Center image: example of the application of the gaussian filter.
Right image: example of the application of the bilateral filter.

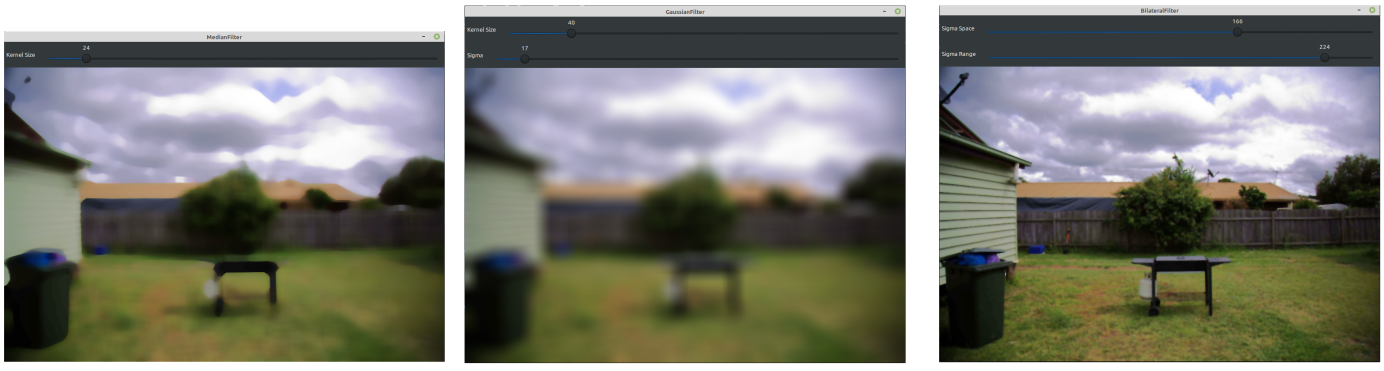

Figure 3: Median, Gaussian, Bilateral