COMPUTER VISION - LAB 3

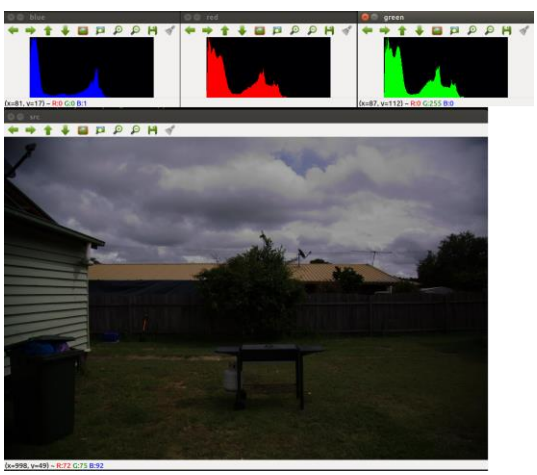
P.Zanuttigh, S. Ghidoni, M. Carraro, G. Agresti

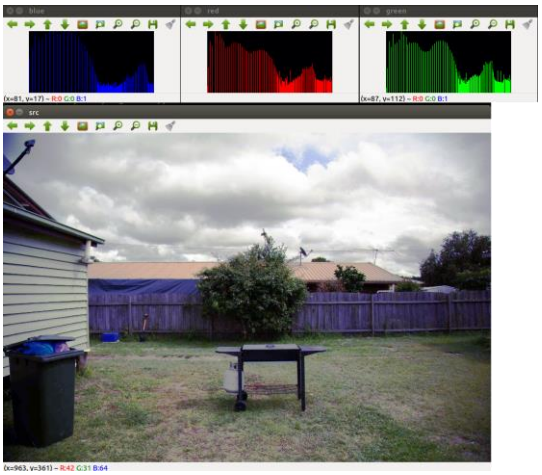
Topics: Image Equalization, Histograms, Filters, Morphological Operators

Part 1: Histogram Equalization

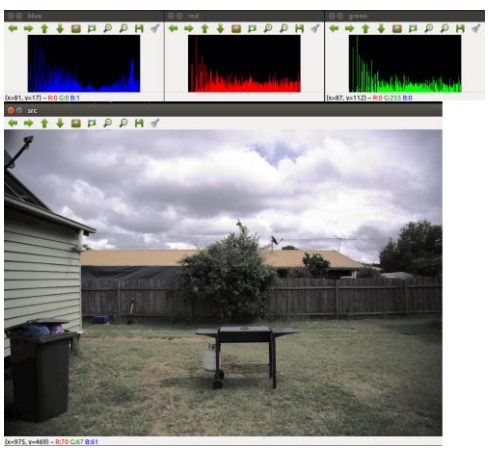
Write a program that:

1. Loads an image (e.g., one of the provided images like “image.jpg” or “countryside.jpg”)
2. Prints the histograms of the image (with 256 bins and [0, 255] as range for all the three channels). Notice that you need to use the calcHist() function separately on the 3 channels. You can use the provided function to visualize the data.
3. Equalizes the R,G and B channels by using `cv::equalizeHist()`
4. Prints the equalized image and the histogram of its channels.





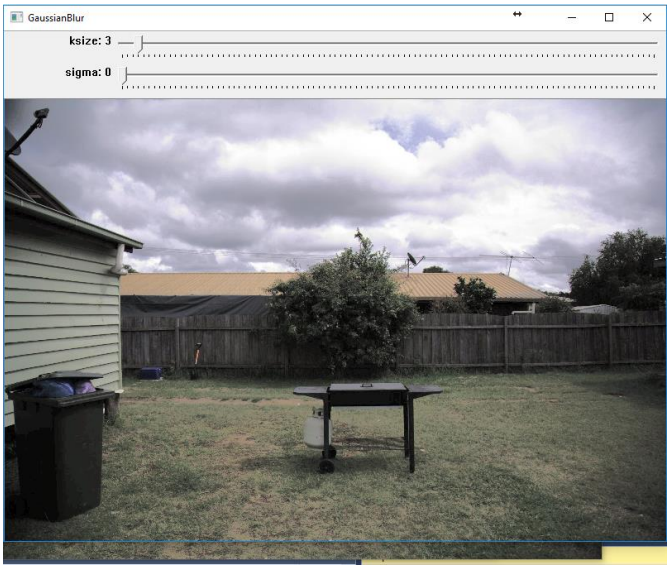
5. Is it possible to obtain a better equalization than the one in 4? Try to work using a different color space, e.g. Lab (use `CV_BGR2Lab` with `cv::cvtColor()`), and equalize only one channel (which one do you choose?)



Part 2: Image Filtering

Generate a denoised version of the image computed in 5. You should try different filters and parameter values.
Write a program that shows the result with some trackbars to vary the parameters of each filter (see the example in the figure). The Table instead specifies the requested filters to test and the parameters to be controlled with the trackbars.

- In order to generate the trackbars you can use the `cv::createTrackbar()` function.
- In order to pass the image and the parameters to the callback of the trackbar create a class containing the image and the filter parameters.
- For the filter parameters create a base class using the provided source code and extend it creating subclasses for the various filters.



`cv::medianFilter()`	• ksize
`cv::GaussianBlur()`	• ksize (keep it square) • sigmaX (=sigmaY)
`cv::bilateralFilter()`	• ksize (trackbar not required, use a fixed value or use the $6s_s$ rule) • sigmaColor • sigmaSpace

Part 3: Morphological Operators (optional, not required for the homework)

By using one or more morphological operators (in particular the erode and dilate ones, that correspond to min/max filters), remove the electric cables and the handle of the barbecue from of the image without damaging too much the rest of the image. Recall that the max/min filters correspond to the morphological operators on greyscale images with a square window as structuring element. In OpenCV you can use the dilate/erode morphological filters.

The functions to be used are `cv::erode()`, `cv::dilate()` and `cv::morphologyEx()` with `cv::MOP_OPEN` and `cv::MOP_CLOSE` as operators (in this case, you may want to try different structuring elements with the function `cv::getStructuringElement()`).

Which is the filter that provides the desired result?