

PC-2020/21 Histogram Equalization

Lorenzo Arena

lorenzo.arena@stud.unifi.it

Abstract

This project was made as the final work for the “Parallel Programming” exam at the University of Florence. The objective was to create three version of a software to equalize the histogram of a given JPEG image. The first version had to be implemented as a sequential program, while the other two had to be parallel; for the parallel versions one has been implemented using OpenMP on top of the already implemented sequential code and the other is built using CUDA to take advantage of a GPU computing power. The projects has been developed in C on a Ubuntu machine; all tests were made on a Ryzen 7 1700 CPU and a NVIDIA Quadro P2000.

Future Distribution Permission

The author(s) of this report give permission for this document to be distributed to Unifi-affiliated students taking future courses.

1. The algorithm

Histogram equalization in image processing can be used to increase an image’s contrast. This is accomplished by spreading the most frequent intensities values across the whole histogram. It works especially well on images which presents foreground and background which are both dark or both bright.

1.1. implementation

We can start by considering a grayscale image of n pixels, in which n_i is the number of occurrences of gray level i . The probability of an occurrence of a pixel of level i in the image is:

$$p_x(i) = \frac{n_i}{n}$$

2. The sequential implementation

3. The OpenMP implementation

4. The CUDA implementation

5. Results