Alperen Degirmenci
CS 205 - HW 4
11/9/12

Problem 4

The mean is computed by calculating the sum of the image pixel values using *gpuarray.sum*() and then dividing this value by the size of the image. For calculating the variance, I wrote a ReductionKernel that calculates $(image[i,j]-\mu)^2$ on the map stage, and then adds these values together; dividing the reduction by the size of the image gives us the variance. These values are used to determine if the termination conditions are met.

The serial code was run on *Harvard_Tiny.png*, *Harvard_Small.png*, and *Harvard_Medium.png*, and then the results were extrapolated in order to estimate the computation time for the larger images (see Figure 1a for results).

Sharpening *Harvard_Huge.png* using $P4$ took 386 *ms*: 41 *ms* for data transfer, 246 *ms* for calculating the mean and the variance, and 99 *ms* for sharpening the image. BlockSize was [128, 2, 1], GridSize was [32, 1536]. In comparison, the version where the mean and the variance were computed on the CPU took about 2 seconds. The serial code would have taken close to two hours.

One problem was that this code used *float* data type for images, which reduced precision in computing the mean and led to errors compared to the CPU.

Image Size	Serial (s)	GPU (s)
256 x 192	12.94	0.176761
512 x 384	65.56	0.175699
1024 x 768	323.48	0.209957
2048 x 1536	1326.9/1604.9	0.211576
4096 x 3072	5345.3/7936.4	0.386303

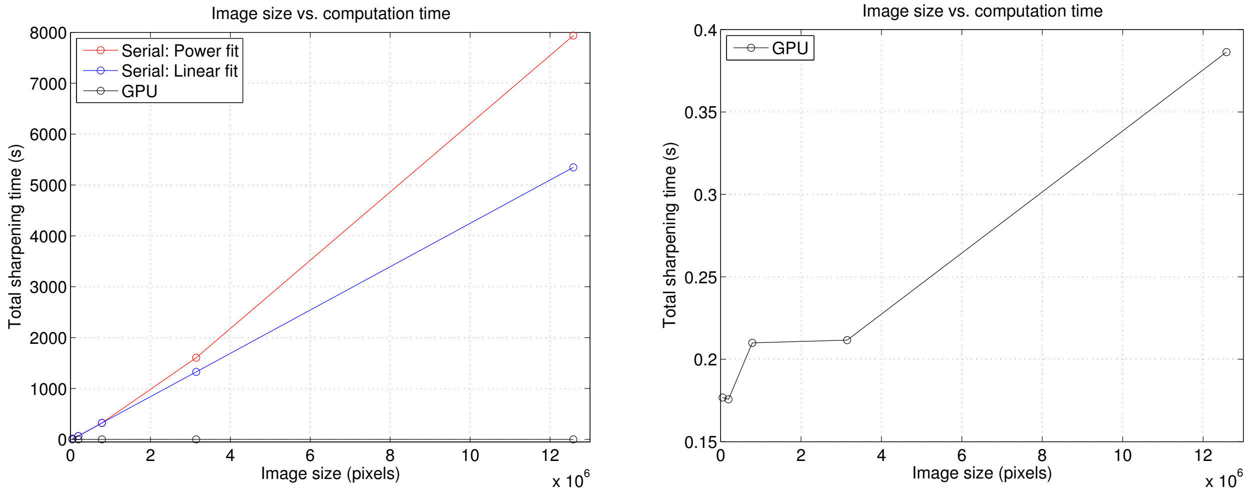

Figure 1: Performance comparison of the serial and GPU algorithm.