Siddharth Nair
2017196

Assignment 1

Step1:

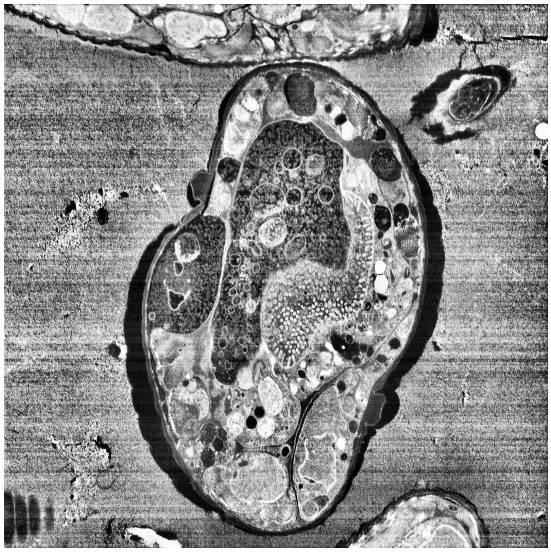
- The probability distribution function(pdf) and cumulative distribution function(cdf) arrays were made of size (ntiles_x * ntiles_y * 256). This implied that each section of 256 corresponds to a tile of the image.
- The pdf was calculated using the *atomicAdd* function.
- For the cdf, a shared integer variable was declared which would store the 0th element of cdf which in turn would be the 0th element of the pdf for a tile.
  Then for each index a '*for loop*' was used to sum the pdf values from index-1 to the thread index. This was then added with the shared variable to finally give the cdf for that particular index.
- The mapping was then calculated using the formula given with the values derived from the previous steps.
- A constant memory of 10*10*256 was declared. The mapping was copied into a variable in the host code which was then used to write the entire mapping into the constant memory.

Step2:

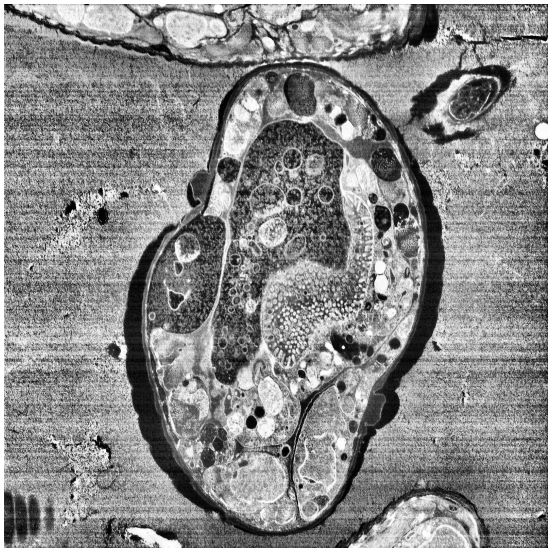
- For the next step, i.e interpolation, the for loops were done away with and a kernel having 256 threads and blocksize of height*width was declared.
- Here the mapping was read from the constant memory and the final output was written to the output array which was then copied back from the device to the host.

The major improvements in speed was observed in the second part (step-2). With only step-1 executing on the gpu the total time required was around 200 ms less than the time taken by the cpu code. However, with step-2 also executing on the gpu, the time taken by the cpu code was 50 times that of the one executing on the gpu.

	CPU	GPU (overall)	GPU (kernel)	Error
Image 1 (earth.png)	2090.79 ms	40.73 ms	9.01 ms	21
Image 2 (celegans.png)	2174.53 ms	58.99 ms	22.73 ms	52



CPU



GPU