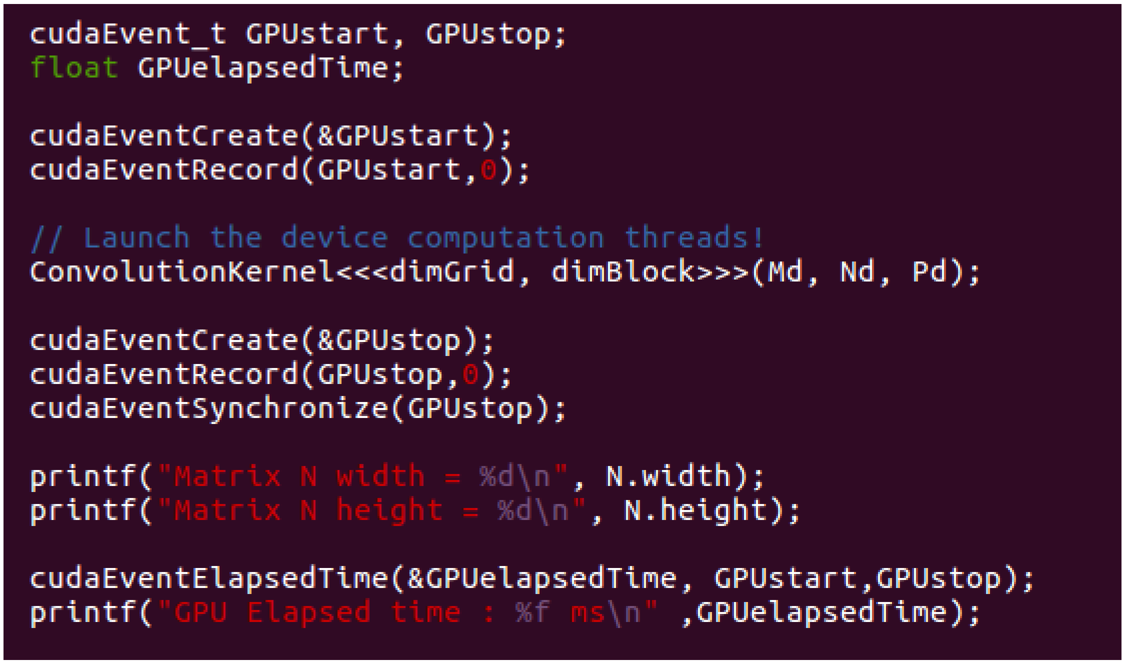
Homework 4 Answer

1. CUDA event API is used to measure execution time of GPU and CPU code. Implemented by recording a time by creating an event just before and after the kernel function is called and then taking the difference of start time and end time recorded as shown below.



```
cudaEvent_t GPUstart, GPUstop;
float GPUelapsedTime;

cudaEventCreate(&GPUstart);
cudaEventRecord(GPUstart,0);

// Launch the device computation threads!
ConvolutionKernel<<<dimGrid, dimBlock>>>(Md, Nd, Pd);

cudaEventCreate(&GPUstop);
cudaEventRecord(GPUstop,0);
cudaEventSynchronize(GPUstop);

printf("Matrix N width = %d\n", N.width);
printf("Matrix N height = %d\n", N.height);

cudaEventElapsedTime(&GPUelapsedTime, GPUstart,GPUstop);
printf("GPU Elapsed time : %f ms\n" ,GPUelapsedTime);
```

Results obtained for different tests inputs are as follows

Test Input I : (Random Size)
Size : 281 x 80
GPU Elapsed time : 0.048480 ms
CPU Elapsed time : 2.256992 ms
Total number of floating point operations : 2 * 5 * 5 * 281 * 80 = 1124000

GPU time for per floating point operations = 0.048480 / 1124000 = 43.13 x 10 $^{-9}$ ms

CPU time for per floating point operations = 2.256992 / 1124000 = 2 x 10^{-6} ms

Test Input II :
Size : 32 x 32
GPU Elapsed time : 0.044768 ms
CPU Elapsed time : 0.099488 ms
Total number of floating point operations : 2 * 5 * 5 * 32 * 32 = 51200

GPU time for per floating point operations = 0.044768 / 51200 = 87.43 x 10 $^{-9}$ ms

CPU time for per floating point operations = 0.099488 / 51200 = 1.95 x 10 $^{-6}$ ms

Test Input III :
Size : 64 x 64

GPU Elapsed time : 0.043488 ms
CPU Elapsed time : 0.434176 ms
Total number of floating point operations : 2 * 5 * 5 * 64 * 64 = 204800

GPU time for per floating point operations = 0.043488 / 204800 = 212.34 x 10 $^{-9}$ ms

CPU time for per floating point operations = 0.434176 / 204800 = 2.1 x 10 $^{-6}$ ms

Test Input IV :
Size : 128 x 128
GPU Elapsed time : 0.047488 ms
CPU Elapsed time : 1.629376 ms
Total number of floating point operations : 819200

GPU time for per floating point operations = 0. 047488 / 819200 = 57.96 x 10 $^{-9}$ ms

CPU time for per floating point operations = 1.629376 / 819200 = 2 x 10 $^{-6}$ ms

Test Input V :
Size : 256 x 256
GPU Elapsed time : 0.060352 ms
CPU Elapsed time : 6.535136 ms
Total number of floating point operations : 2 * 5 * 5 * 256 * 256 = 3276800

GPU time for per floating point operations = 0.060352 / 3276800 = 18.42 x 10 $^{-9}$ ms

CPU time for per floating point operations = 6.535136 / 3276800 = 2 x 10 $^{-6}$ ms

Test Input V :
Size : 512 x 512
GPU Elapsed time : 0.113568 ms
CPU Elapsed time : 26.200705 ms
Total number of floating point operations : 2 * 5 * 5 * 512 * 512 = 13107200

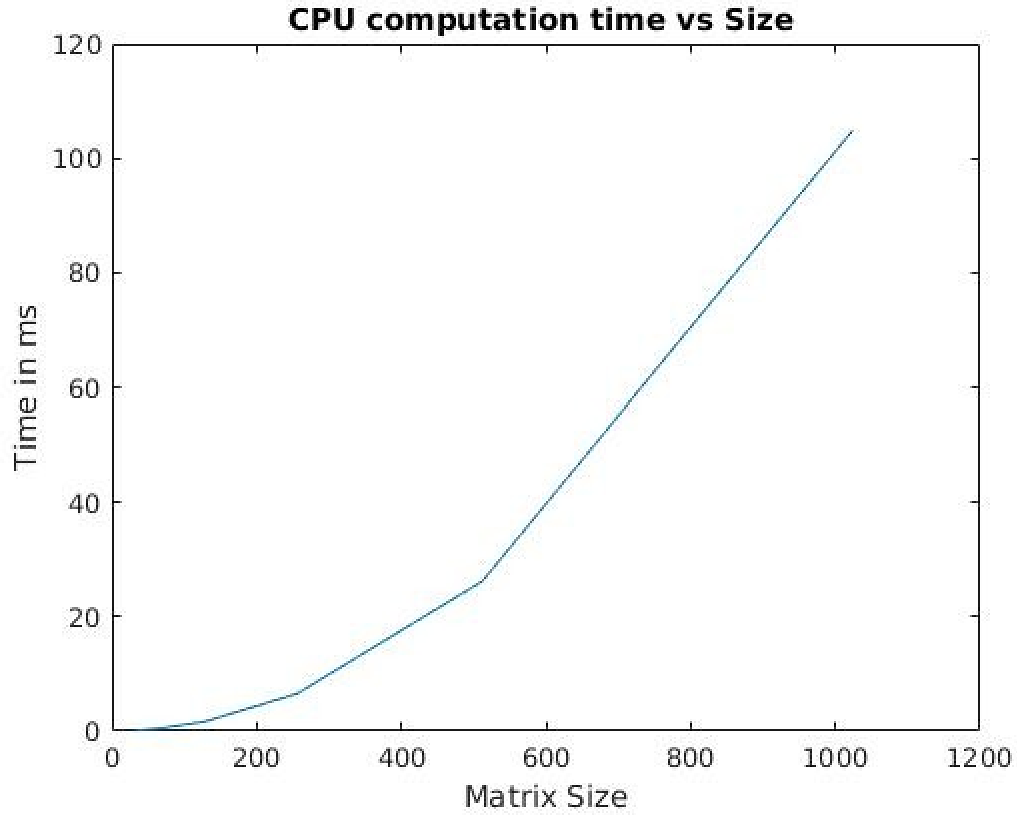
GPU time for per floating point operations = 0.113568 / 13107200 = 8.66 x 10^{-9} ms

CPU time for per floating point operations = 26.200705 / 13107200 = 2 x 10 $^{-6}$ ms

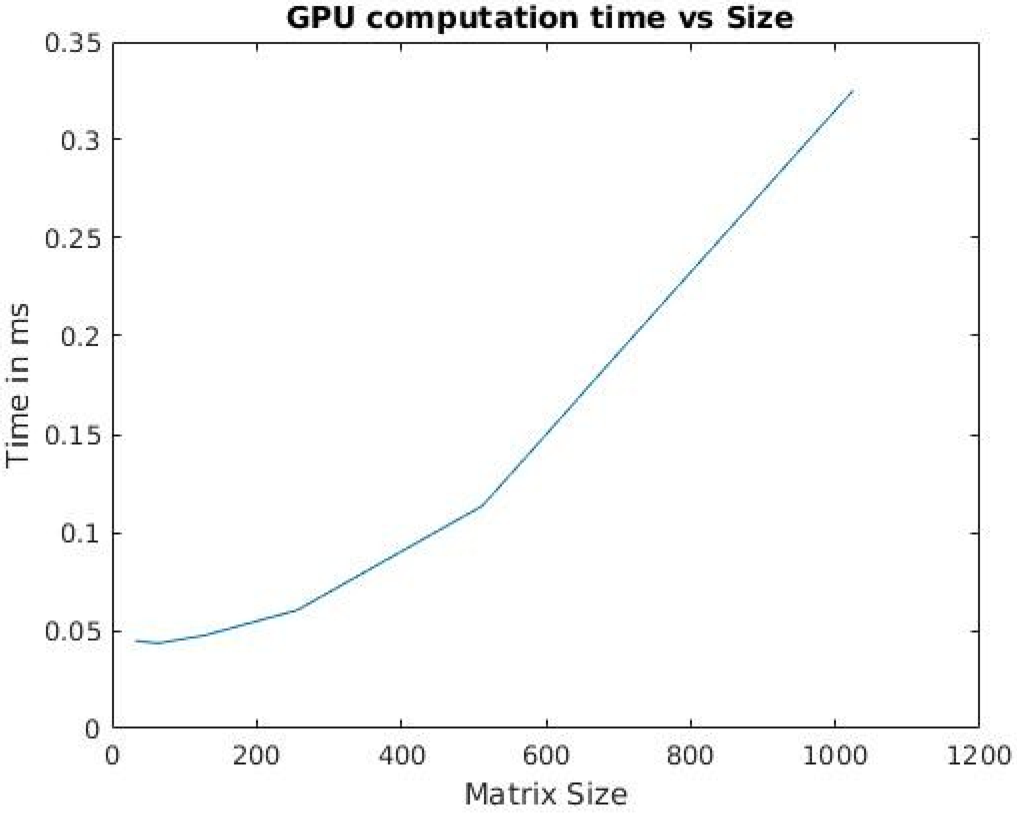
Test Input VI :
Size : 1024 x 1024
GPU Elapsed time : 0.325024 ms
CPU Elapsed time : 104.868607 ms
Total number of floating point operations : 2 * 5 * 5 * 1024 * 1024 = 52428800
GPU time for per floating point operations = 0.325024 / 52428800 = 6.20 x 10 $^{-9}$ ms
CPU time for per floating point operations = 104.868607/ 52428800 = 2 x 10 $^{-6}$ ms

CPU computation time scales with size as shown below:

Homework 4 Answer



GPU computation time scales with size as shown below:

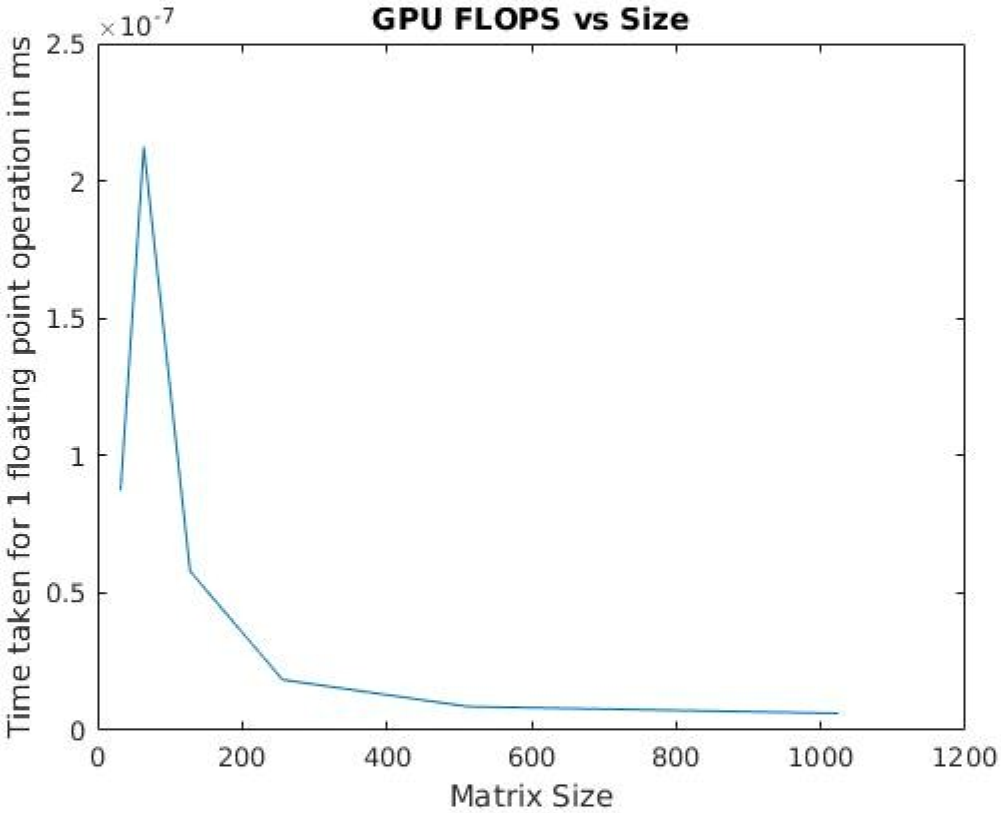


Homework 4 Answer

Image I Test :
Size : 32 x 32
GPU Elapsed time : 0.044768 ms
CPU Elapsed time : 0.099488 ms
Total number of floating point operations : 2 * 5 * 5 * 32 * 32 = 51200

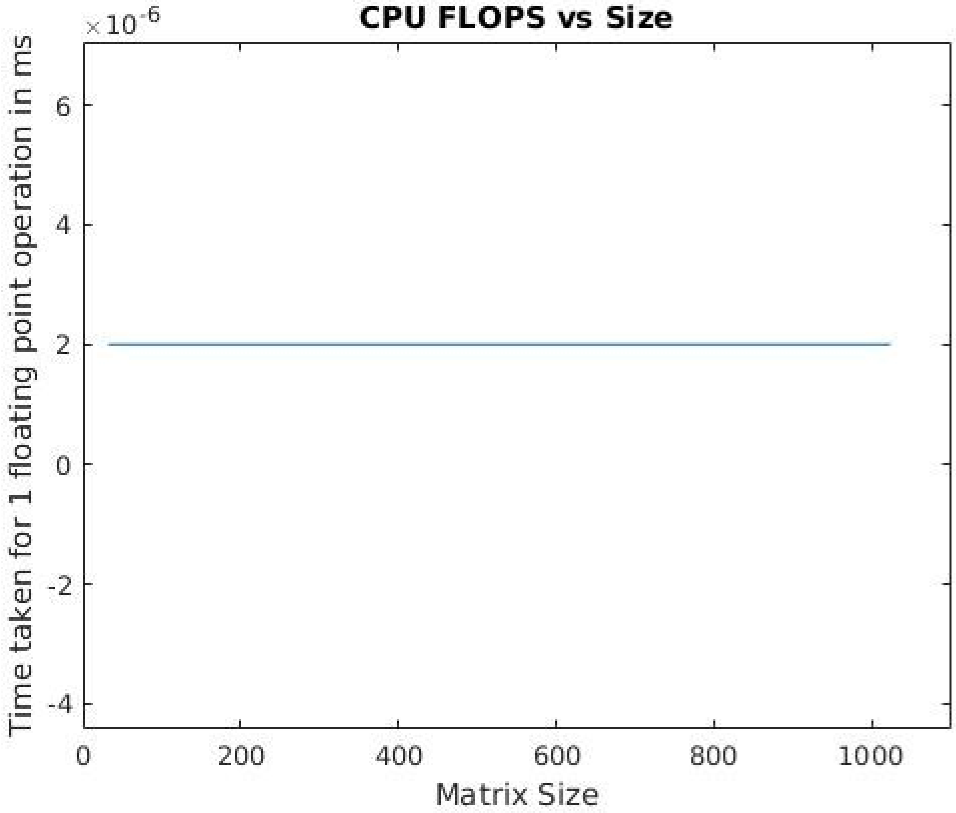
Image II Test :
Size 1024 x 1024
GPU Elapsed time : 0.313120 ms
CPU Elapsed time : 105.495361 ms
Total number of floating point operations : 2 * 5 * 5 * 1024 * 1024 = 52428800

GPU FLOPS scaling with respect to input



Homework 4 Answer

CPU FLOPS scaling with respect to input:



2. Overhead cost for using the GPU for computation can be given by the difference between Overhead time and GPU Elapsed time.

Test Input I :
Size : 80 x 281
GPU Elapsed time : 0.048512 ms
Time including GPU computation : 1.144768 ms
Therefore, Overhead time = 1.144768 ms - 0.048512 ms = 1.096256 ms

Test Input II :
Size : 32 x 32
GPU Elapsed time : 0.043424 ms
Time including GPU computation : 0.897504 ms
Therefore, Overhead time = 0.897504 ms - 0.043424 ms = 0.85408 ms

Test Input III :
Size : 64 x 64
GPU Elapsed time : 0.043328 ms
Time including GPU computation : 0.941120 ms
Therefore, Overhead time = 0.941120 ms - 0.043328 ms = 0.897792 ms

Homework 4 Answer

Test Input IV :
Size : 128 x 128
GPU Elapsed time : 0.046880 ms
Time including GPU computation : 1.114784 ms
Therefore, Overhead time = 1.114784 ms - 0.046880 ms = 1.067904 ms

Test Input V :
Size : 256 x 256
GPU Elapsed time : 0.060128 ms
Time including GPU computation : 1.498880 ms
Therefore, Overhead time = 1.498880 ms - 0.060128 ms = 1.438752 ms

Test Input VI :
Size : 512 x 512
GPU Elapsed time : 0.126304 ms
Time including GPU computation : 3.621952 ms
Therefore, Overhead time = 3.621952 ms - 0.126304 ms = 3.495648 ms

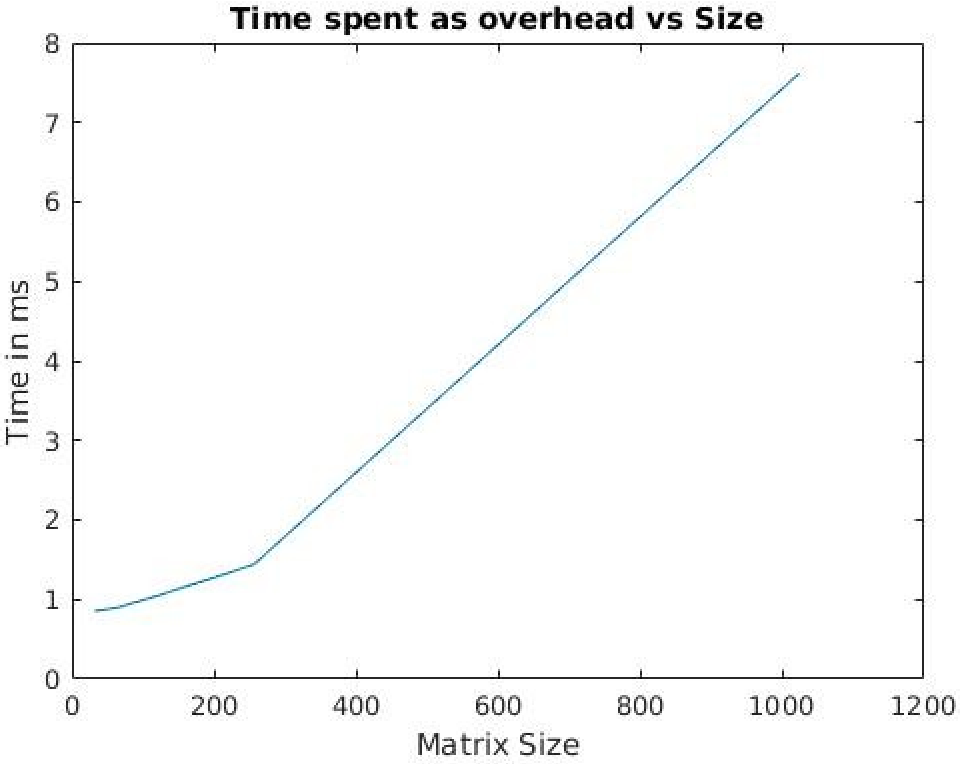
Test Input VI :
Size : 1024 x 1024
GPU Elapsed time : 0.323392 ms
Time including GPU computation : 7.937344 ms
Therefore, Overhead time = 7.937344 ms - 0.323392 ms = 7.613952 ms

Test Image I :
Size : 32 x 32
GPU Elapsed time : 0.053504 ms
Time including GPU computation : 0.996128 ms
Therefore, Overhead time = 0.996128 ms - 0.053504 ms = 0.942624 ms

Test Image II :
Size 1024 x 1024
GPU Elapsed time : 0.324096 ms
Time including GPU computation : 7.857888 ms
Therefore, Overhead time = 7.533792 ms

Homework 4 Answer

Overhead scale with the size as shown below


Time spent as overhead vs Size
Time in ms
Matrix Size