GPU-Accelerated Deep Convolutional Neural Network for Object Recognition

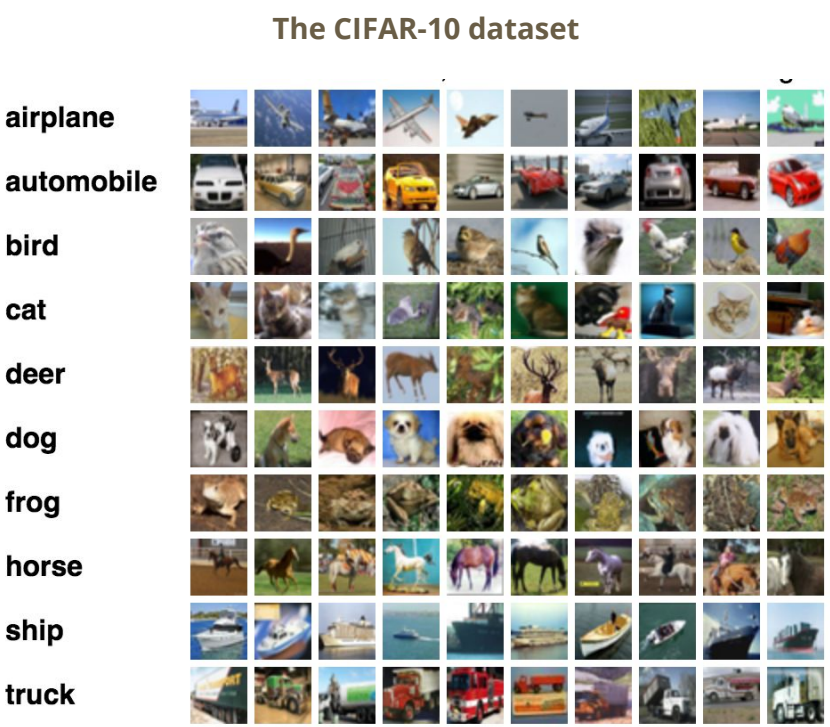
Guan Sun

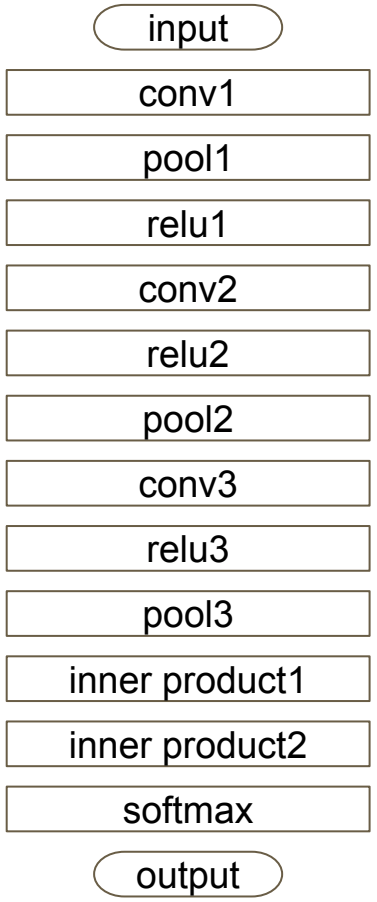
Milestone 3

Train a model for object recognition

Performance Analysis & Optimization

Object Recognition Model

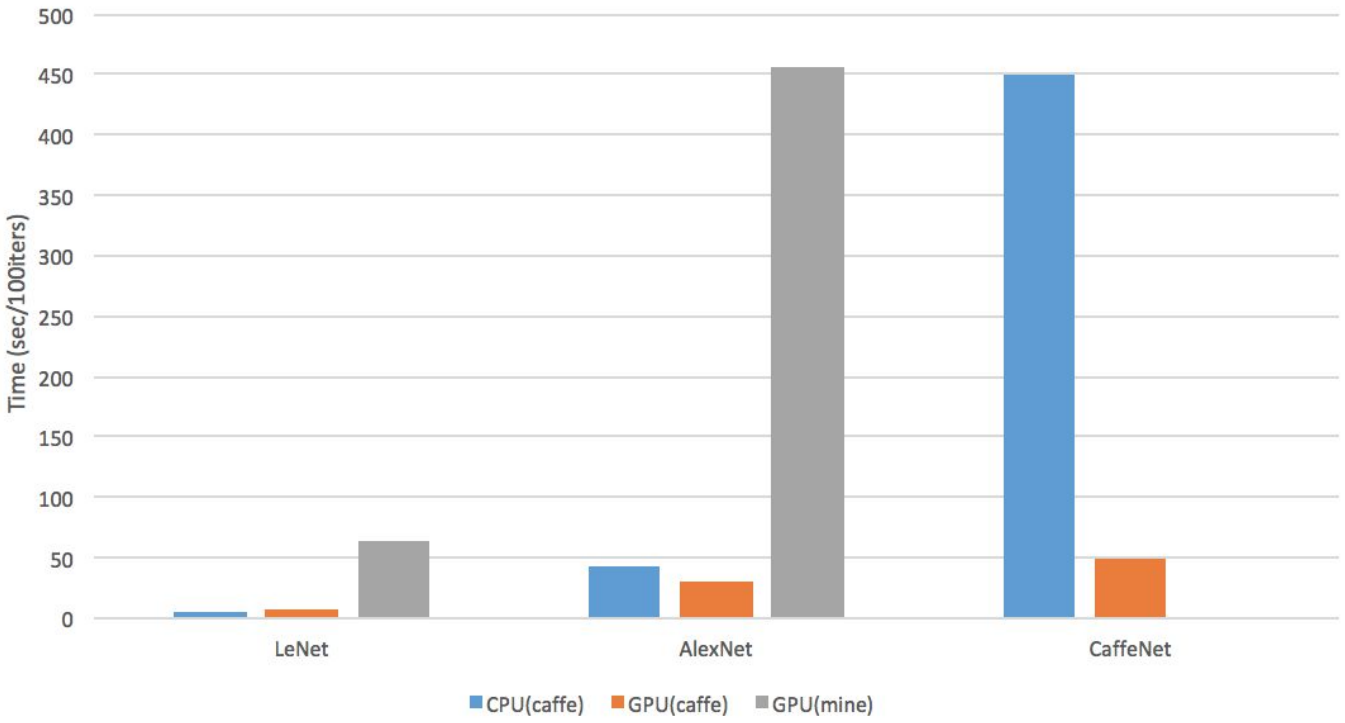





```
I1207 17:07:34.491516 2071388928 solver.cpp:320] Iteration 5000, loss = 0.481036
I1207 17:07:34.491554 2071388928 solver.cpp:340] Iteration 5000, Testing net (#0)
I1207 17:07:44.542856 2071388928 solver.cpp:408]     Test net output #0: accuracy = 0.7621
I1207 17:07:44.542901 2071388928 solver.cpp:408]     Test net output #1: loss = 0.722371 (* 1 = 0.722371 loss)
I1207 17:07:44.542912 2071388928 solver.cpp:325] Optimization Done.
I1207 17:07:44.542920 2071388928 caffe.cpp:215] Optimization Done.
```

Accuracy = 76.21%

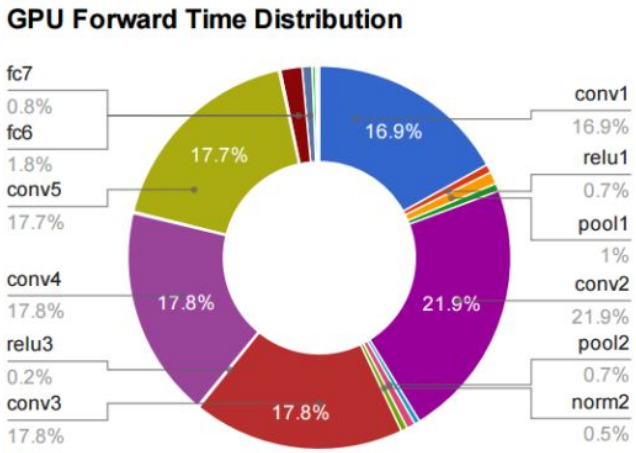
Performance

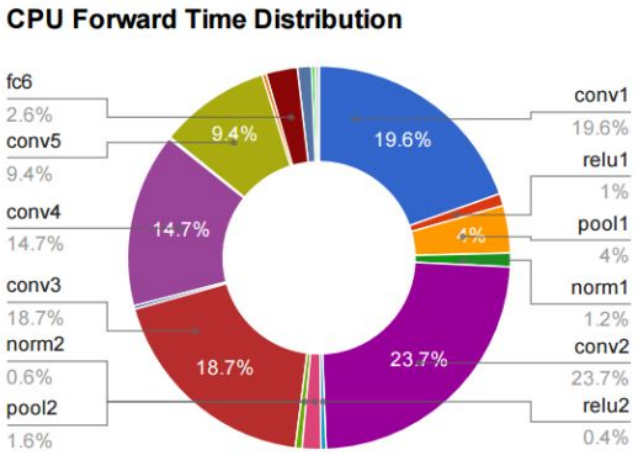


	LeNet	AlexNet	CaffeNet
CPU(caffe)	6.11	42.89	9X
GPU(caffe)	6.89	30.67	1X
GPU(mine)	64.78	456.11	

(seconds/100 training iters)

It's all about Matrix Multiplication

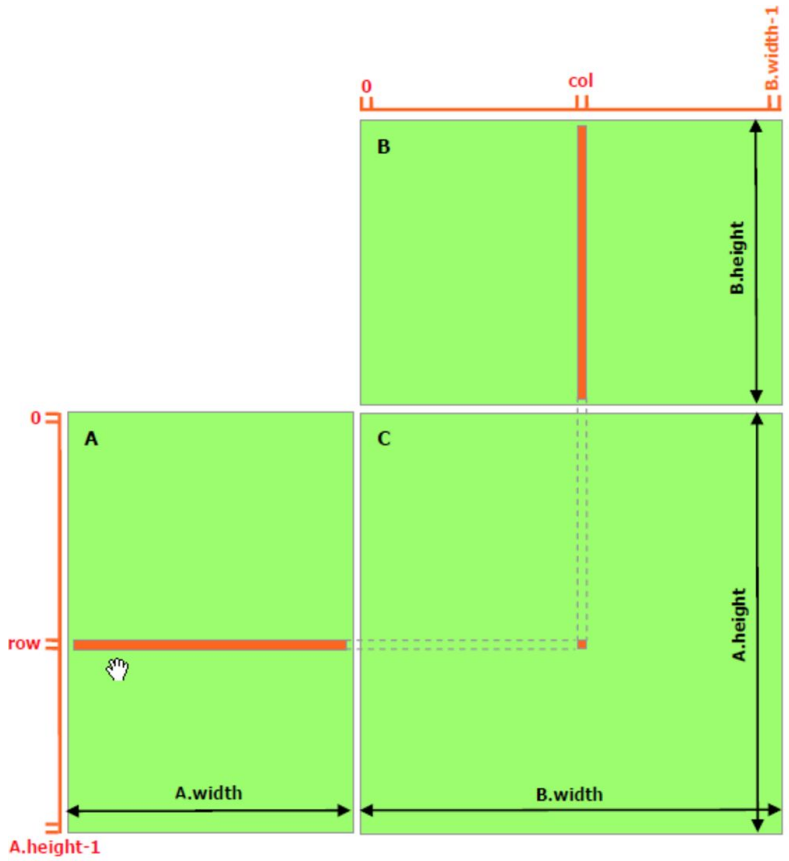




95% of GPU

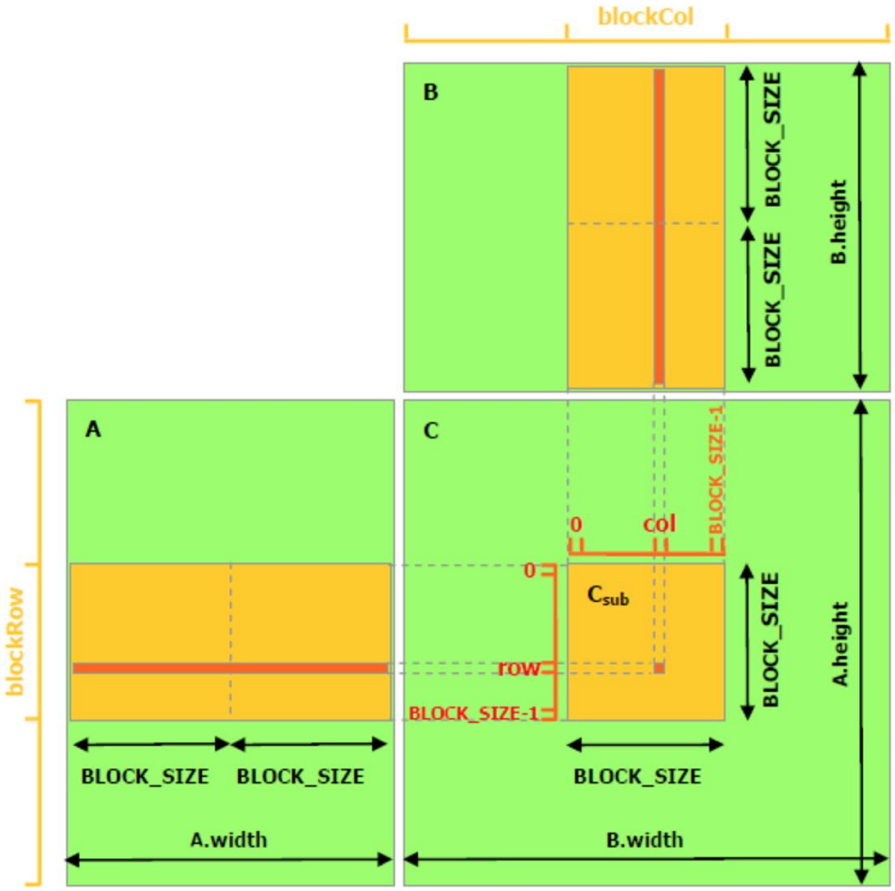
89% of CPU

Matrix Multiplication Optimization



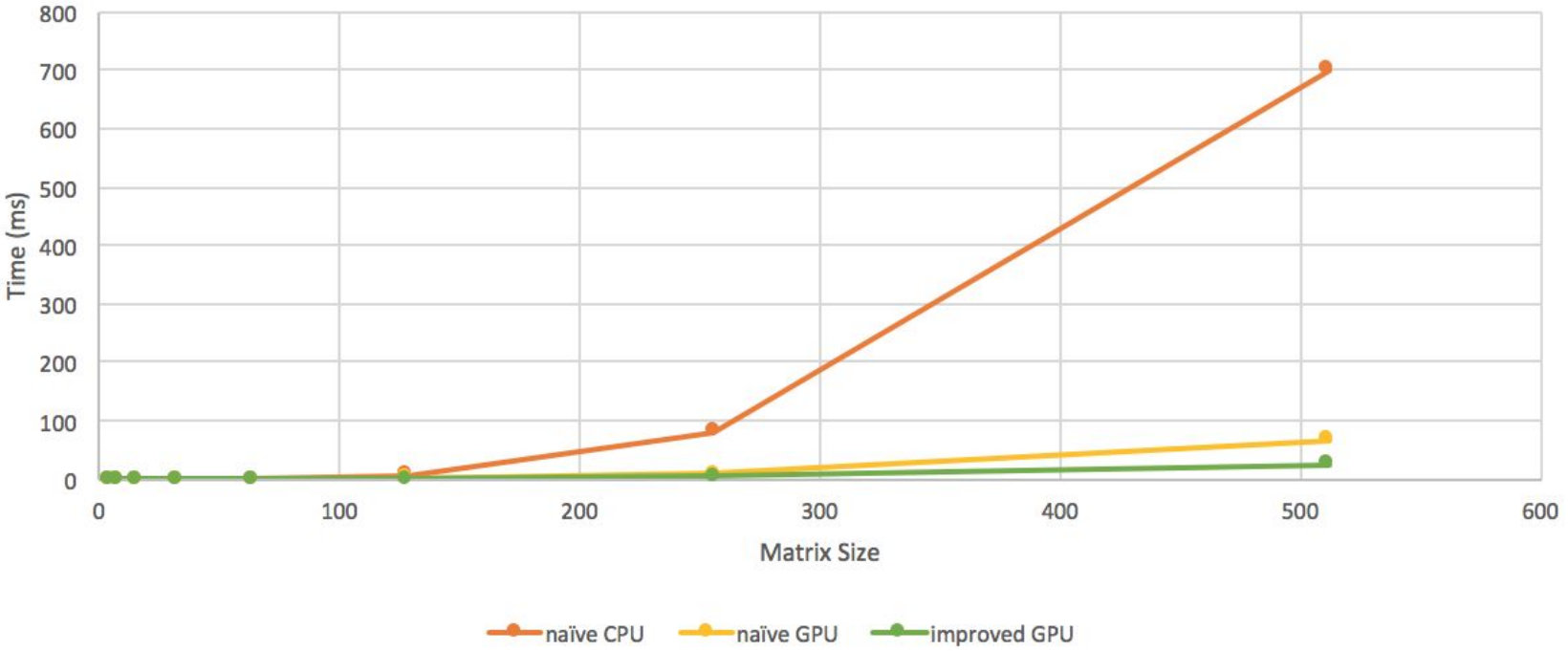
Each element is read N times from global memory!

Matrix Multiplication Optimization



Each element is read N/Block_size times from global memory.
Threads access elements form faster shared memory.

Results after Optimization



	4	8	16	32	64	128	256	512	1024	2048
naïve CPU	0.0030	0.0050	0.0240	0.1690	0.8350	7.2370	80.2250	698.8210	13206.3710	151433.4680
naïve GPU	0.0370	0.0630	0.0578	0.0412	0.1948	1.0418	8.5202	66.1688	340.0051	1714.5858
improved GPU	0.0314	0.0407	0.0291	0.0385	0.0769	0.3827	4.3128	25.8006	78.8657	429.3831

(ms)

Before the Deadline

Run test on larger dataset

Continue working on optimization

Thanks!