Lab3 Report

Ke Xu 604761427

Briefly explain the parallel strategies you applied in this lab. How is it parallelized? What is the expected communication overhead?

I just partition the multiplication part of the convolutional layer into 256 different work items, which is only one dimension and each work item get the computation of 1 group of filters. For the pooling layer, I implement it sequentially. The communication overhead only includes the transfer of data whose size is 256*224*224, 256*228*228, 256*256*5*5 from the host memory to local memory and data whose size is 256*224*224 from local memory to host memory.

Evaluate your program in terms of the execution time. Please make a comparison with the serial version, and discuss the scalability of your parallel implementation using 1, 2, 4, 8, 16, 32 processors.

It is about 3.85s. For the serial version, it is about 58s and only 2 GFlop/s. For the different numbers of processors, the execution time is following:

1: 60.7s
2: 32.34s
4: 18.22s
8: 10.45s
16: 4.34s
32: 3.85s

Challenges you encountered and how you solved them.

I have a lot of segmentation errors and it is because I pass the data from the host memory to the local memory and the size is larger than the local memory. So I just modify the parallel part and get the data fitting the memory.

Also I find that only when I change the cnn.c and make can I correctly get the result of the current kernel.cl function. If I only modify the kernel.cl and make, it will not change. This takes me a lot of time.