CMP SCI 635: Modern Computer Architecture

A Configurable Cloud-Scale DNN Processor for Real-Time AI

Name: Kunjal Panchal Date: 3rd Dec, 2019 Student ID: 32126469

Paper: 22 – Data Parallel [Fowers18]

Strengths:

1. NPUs are optimized for high volume 8-Bit operations and fast accumulators that are able to store the product of large matrix-matrix multiplications (n=64). They possess no hardware rasterizer nor hardware texture mapping. Essentially aimed for fast inference calculations. They sound practical for Deep Learning in Computer Vision.

2. An NPU can transcribe speech to text as a virtual assistant. It can provide voice assistance without streaming audio to the cloud; eventually we would be able to use it without the internet.

Weaknesses:

1. NPUs "approximate" results and might hence be used for applications like image processing or pattern recognition but would fail to provide a valid calculator. It seems like architecture is somewhat at a loss as to how to approach the multicore era, in particular the dark silicon problem. So, the community has begun exploring architectures that may have appeared too risky in the past.

2. Consumers will still want Microsoft to offer cloud servers with the FPGA exposed to them for their own designs, like Amazon.

Questions/Assertions:

1. An NPU, about the size of a thumb's nail, built with a total 5.5 billion transistors, can deliver 25 times the performance with 50 times greater efficiency compared to a quad core Cortex A73 processor for AI computing tasks.

2. FPGAs can be faster at inference than state of the art GPUs when properly designed, primarily since you can target much lower precision with them than is currently possible on GPUs (and ASICs like TPU). In datacentres, power and cooling is also very important, and given that FPGAs can beat GPUs at a mere 40W power envelope compared to 250W, there's a real niche for FPGA ML.

3. While some are interested in machine learning on hardware, few may not be as interested in just getting a model running. Ideas can involve dividing up the network pipeline in a way that takes advantage of an FPGA's capability for parallelism while dancing around the relatively limited number of DSPs. Because of this, we might need full control of what goes on the FPGA and not use any existing frameworks. Project Brainwave has a limited amount of preconfigured options to choose from.

4. For the hundreds of thousands of reconfigurable processors running at unheard of real time input speeds configured to run a vast array of easily downloadable preconfigured open source machine learning models; all someone would have to do is load one in a clever way and suddenly all those FPGA chips are trying to break free of their bounds and in runaway.

5. There's a benchmark comparison between their solution of an NN on a CPU versus the same task with tensorflow on NVidia P100 GPU, and the conclusions state "…end-to-end we were able to outperform optimized Tensorflow implementations by 1.5×- 2.3×." This proves that CPUs can be a competitive alternative when training neural nets.