High Performance, Low Power, CNN implementation on Ultra-Low Power embedded platforms

Brendan Barry
VP of Engineering
Movidius
Why are CNN’s interesting?

- Deep-Learning: the Great Disruptor
  - Companies using CNNs can enable sought after features otherwise difficult or impossible
Why are CNNs interesting to Movidius?

### DRONES
- Sense and Avoid
- GPS-denied hovering
- Scene labeling

### SERVICE ROBOTICS
- Volumetric mapping
- Indoor visual navigation

### AR/VR
- 6DOF positional tracking
- Depth sensing
- Gesture & Eye tracking
- See-through

### SECURITY CAMERAS
- Detection
- Recognition
- Identification
Deep Convolutional Neural Networks (CNNs):

- Training on HPC cluster
- Inference on embedded platform
- What about training on embedded platform?
AlexNet CNN (Visualization)
Complexity of a Standard Network: GoogLeNet

Computational cost is increased by less than 2X compared to AlexNet (<1.5Bn operations/evaluation)

5M parameters
The Deeper The Better... Yet With Increased Complexity

EXCESS

ImageNet Accuracy % over Time

<table>
<thead>
<tr>
<th>Year</th>
<th>Layers</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>8</td>
<td>83.60%</td>
</tr>
<tr>
<td>2013</td>
<td>16</td>
<td>88.80%</td>
</tr>
<tr>
<td>2014</td>
<td>58</td>
<td>93.33%</td>
</tr>
<tr>
<td>2015</td>
<td>158</td>
<td>95.06%</td>
</tr>
</tbody>
</table>
Incremental Accuracy and the Power Efficiency Cost: Is It Worth It?

Brendan Barry
Where is the energy efficiency lost?

- Data movement burns much more power than arithmetic operations
- Focus is on reducing operations, not on reducing data movement

\[(\text{pJ} / \text{normalized to ADD}=1)\]

- Add
- 64 word RF=1.3
- Multiply=3.4
- 4k SRAM = 44
- 32k SRAM = 61
- DRAM = 3556
Power cost of data movement

- Will this get better?
- Yes but...
  - Logic power decreases with each node
  - SRAM power decrease is less each time
  - DRAM iterations are slower/less impactful

[Chart showing the comparison of LPDDR, SRAM, and Logic over time]
Solutions — 3 Pillars

- Optimized Software
- Appropriate Algorithms
- Processing Platform Architecture
Consider processing platform vs. this pyramid

More local memory allows expensive memory to be kept “Dark”
- Less power consumption
- More scope for software dataflow optimization

Recent paper on “Dark Memory”\[1\] shows this formally

Algorithms: Some are Better than Others

- Kernel Level: Ratio of processing cycles to load/store cycles
  - Matrix-Matrix better than Matrix-Vector
  - CNN Example: Batched AlexNet better than AlexNet
  - CNN Example: GoogleNet better than AlexNet

- Optimize for locality of access
  - Transform algorithm solutions to predictable dataflow
  - Keep intermediate data as local as possible
  - Move data once and complete all processing possible

- Cache systems can help, but can also be counter productive
  - Pre-fetching can burn more power
  - Cache trashing will also burn more power
What is Optimal energy efficiency?

What do we want to do?
- Do as much compute as possible on as smallest span of data
- Keep higher bandwidth data as local as we can

How can we do this (quickly!)?
- Manually optimized code
- Directed Acyclic Graphs (DAG)
- Deferred execution frameworks

How do you know you are actually improving power efficiency?
EXCESS – Energy monitoring

- HPC (x86 and K40) systems system energy monitoring
- Tegra TK1, TX1 energy monitoring
- Myriad2 compute cluster energy monitoring
Directed Graph — Specifics

- OpenVx™: Khronos industry standard supported by variety of platforms
  - Directed graph API
  - Some built-in, vision-specific, common functions
- Tensorflow
  - General compute API
  - Directed graph type approach without full graph API — key ideas
    - ‘Process this’
    - ‘Process this with the output from that earlier task’
- Other viable Directed Graph approaches
  - Android: RenderScript Graph builder
  - Various vendor proprietary Frameworks
**Deferred Execution**

- **Key idea:** Define all of the problem and give it to a tool. Tool can analyze the data flow
  - **Pros:** more abstract API
  - **Cons:** tool needs to do more work, less predictable

- **Specific examples**
  - Halide for image processing
  - OpenCL
Worked example: GoogleNet Implementation

- **GoogLe-Net V1**
  - Lots of compute per input
  - Lots of intermediate data
  - Lots of weights (but pretty good in relative terms)
GoogleNet Implementation

- Optimization 1: Kernel Fusion for Key operations
  - Conv + Bias + RELU → Single Operation
  - Conv + Bias + RELU + Pool → More work but good returns
GoogleNet Implementation

- Optimization2: 8 bit weights
- Optimization3: Keep activations in local SRAM
  - Reduce DDR BW to ~7MB/sec/inference
CNN implementation problems

- Identified need
- General focus on performance not energy efficiency
- Identified industry narrow solutions
- Identified issues with general solutions

- How does EXCESS help?
How Movidius is Deploying Standard CNNs at the Network Edge?

EXCESS

CNN Model Description

Weights

Fathom

For Integration

Drones
Robotics

PARSES CNN Model Scheduler

EXCESS influence

Efficient Execution of Kernel Function Libraries

Brendan Barry

22
GoogleNet Relative GFLOPS/W Performance Results

GoogLeNet Single Inference (Batch = 1) no Heatsink or Fan

<table>
<thead>
<tr>
<th></th>
<th>nm Process</th>
<th>fps</th>
<th>Power (W)</th>
<th>GFLOPS</th>
<th>GFLOPS/W</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPU-A</td>
<td>28</td>
<td>15</td>
<td>7.5</td>
<td>51.37</td>
<td>6.85</td>
</tr>
<tr>
<td>GPU-B</td>
<td>20</td>
<td>22</td>
<td>7.5</td>
<td>75.34</td>
<td>10.04</td>
</tr>
<tr>
<td>Myriad2</td>
<td>28</td>
<td>25</td>
<td>1.2</td>
<td>85.61</td>
<td>71.34</td>
</tr>
</tbody>
</table>
Fathom Neural Compute Stick: Deep Learning’s Newest Form Factor

TX1, NVIDIA Tegra K1 Mobile AP/GPU

TX1, NVIDIA Tegra X1 Mobile AP/GPU

High throughput embedded inference platforms available today
Where to next?

- CNN, very fast changing research area
  - New results frequently change research direction

- Exploration of optimised energy efficient inference
  - Is it possible to change training to produce an energy efficient inference?

- Is it possible to iterate training after each encounter of new data?
  - Current systems cloud based
  - What above live re-learning?
Acknowledgements

Links APIs
- http://www.tensorflow.org
- https://www.khronos.org/openvx/resources
- http://www.movidius.org
- http://halide-lang.org/

Academic references
1. Dark Memory and Accelerator-Rich System Optimization in the Dark Silicon Era
2. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems
4. EMBEDDED DEEP NEURAL NETWORKS “The Cost of Everything and the Value of Nothing” David Moloney, CTO Hot Chips 28, Cupertino, California