Optimizing FPGA-based Accelerator Design for Deep Convolutional Neural Networks

C. Zhang, P. Li, G. Sun, Y. Guan, B. Xiao and J. Cong.

Presented by Maya Gibson
Convolutional Neural Network (CNN)

- Feedforward process for recognition
- Backward path for training
- Composed of two components:
  - A feature extractor
  - A classifier
- Composed of multiple computation layers

[1,2]
Computation Of A Convolutional Layer

Graph of a convolutional layer

Pseudo code

```c
for (row=0; row<R; row++) {
    for (col=0; col<C; col++) {
        for (to=0; to<M; to++) {
            for (ti=0; ti<N; ti++) {
                for (i=0; i<K; i++) {
                    for (j=0; j<K; j++) {
                        L: output_fm[to][row][col] +=
                           weights[to][ti][i][j]*input_fm[to][S*row+i][S*col+j];
                    }
                }
            }
        }
    }
}
```
CNN Applications

- Image Processing
- Video Surveillance
- Mobile Robot Vision
- Natural Language Processing

Fast growth of modern technology based on deep learning algorithms has generated new research & implementations. [1,2,3]
CNN Applications

- For any CNN algorithm implementation, there are a lot of potential solutions that result in a vast design space for exploration.
- There is up to a 90% performance difference between two different solutions with the same logic resource utilisation.

An efficient method is of top priority for exploration of FPGA based CNN design space.
Field-Programmable Gate Arrays (FPGA)

- General purpose processors are not efficient enough for CNN implementations.
- Integrated circuit with the following advantages
  - Good performance
  - High energy efficiency
  - Capability of reconfiguration

[1,2]
Problem!

Although current FPGA accelerators have demonstrated better performance over generic processors, the accelerator design space has not been explored well enough - there are more efficient solutions that have yet to be discovered!
Introduction Summary

- What is your topic?
- Where does it fit in the Computer Science discipline?
- What drives the development of the topic (or what are industry’s needs), what is its background?
- What distinguishes the topic from associated topics?
The Roofline Model

- Relates system performance to off-chip memory traffic and the peak performance provided by the hardware platform.

[1]
Exploring Accelerator Designs

Overview of our accelerator structure
Exploring Accelerator Designs

Computation Optimisation
- Loop Unrolling
- Loop Pipelining
- Tile Size Selection

Memory Access Optimisation
- Local Memory Promotion
- Loop Transformations for Data Reuse
- CTC Ratio.

[1]
Design Space Exploration

(a) Design space of all possible designs
(b) Design space of platform-supported designs

All possible designs
Platform-supported designs

[1]
Implementation Details

Implementation Overview

Proposed Accelerator

[1]
Timing graph

- On-chip buffers overlap data transfer time with computation
- Used to increase the bandwidth utilization
Evaluation

- Accelerator design implemented with Vivado HLS, on a VC707 board in C with a working frequency of 100MHz.
- The software implementation uses a Intel XEON CPU E5-2430 (@2.20GHz) with a 15MB cache.
- The results are very successful!
Comparison to previous implementations

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Precision</strong></td>
<td>fixed point</td>
<td>16bits fixed</td>
<td>48bits fixed</td>
<td>48bits fixed</td>
<td>fixed point</td>
<td>48bits fixed</td>
<td>32bits float</td>
</tr>
<tr>
<td><strong>Frequency</strong></td>
<td>150 MHz</td>
<td>115 MHz</td>
<td>125 MHz</td>
<td>125 MHz</td>
<td>125 MHz</td>
<td>200 MHz</td>
<td>100 MHz</td>
</tr>
<tr>
<td><strong>FPGA chip</strong></td>
<td>Virtex6 VLX240T</td>
<td>Virtex5 LX330T</td>
<td>Spartan-3A DSP3400</td>
<td>Virtex4 SX35</td>
<td>Virtex5 SX240T</td>
<td>Virtex5 SX240T</td>
<td>Virtex7 VX485T</td>
</tr>
<tr>
<td><strong>FPGA capacity</strong></td>
<td>37,680 slices 768 DSP</td>
<td>51,840 slices 192 DSP</td>
<td>23,872 slices 126 DSP</td>
<td>15,360 slices 192 DSP</td>
<td>37,440 slices 1056 DSP</td>
<td>37,440 slices 1056 DSP</td>
<td>75,900 slices 2800 DSP</td>
</tr>
<tr>
<td><strong>LUT type</strong></td>
<td>6-input LUT</td>
<td>6-input LUT</td>
<td>4-input LUT</td>
<td>4-input LUT</td>
<td>6-input LUT</td>
<td>6-input LUT</td>
<td>6-input LUT</td>
</tr>
<tr>
<td><strong>CNN Size</strong></td>
<td>2.74 GMAC</td>
<td>0.53 GMAC</td>
<td>0.26 GMAC</td>
<td>0.26 GMAC</td>
<td>0.53 GMAC</td>
<td>0.26 GMAC</td>
<td>1.33 GFLOP</td>
</tr>
<tr>
<td><strong>Performance</strong></td>
<td>8.5 GMACS</td>
<td>3.37 GMACS</td>
<td>2.6 GMACS</td>
<td>2.6 GMACS</td>
<td>3.5 GMACS</td>
<td>8 GMACS</td>
<td>61.62 GFLOPS</td>
</tr>
<tr>
<td></td>
<td>17 GOPS</td>
<td>6.74 GOPS</td>
<td>5.25 GOPS</td>
<td>5.25 GOPS</td>
<td>7.0 GOPS</td>
<td>16 GOPS</td>
<td>61.62 GOPS</td>
</tr>
<tr>
<td><strong>Performance Density</strong></td>
<td>4.5E-04 GOPs/Slice</td>
<td>1.3E-04 GOPs/Slice</td>
<td>2.2E-04 GOPs/Slice</td>
<td>3.42E-04 GOPs/Slice</td>
<td>1.9E-04 GOPs/Slice</td>
<td>4.3E-04 GOPs/Slice</td>
<td>8.12E-04 GOPs/Slice</td>
</tr>
</tbody>
</table>
Comparisons to CPU

Power Consumption and Energy

Performance Comparison

<table>
<thead>
<tr>
<th></th>
<th>CPU 2.20GHz (ms)</th>
<th>FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>float</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32 bit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>layer 1</td>
<td>98.18</td>
<td>7.67</td>
</tr>
<tr>
<td>layer 2</td>
<td>94.66</td>
<td>5.35</td>
</tr>
<tr>
<td>layer 3</td>
<td>77.38</td>
<td>3.79</td>
</tr>
<tr>
<td>layer 4</td>
<td>65.58</td>
<td>2.88</td>
</tr>
<tr>
<td>layer 5</td>
<td>40.70</td>
<td>1.93</td>
</tr>
<tr>
<td>Total</td>
<td>376.50</td>
<td>21.61</td>
</tr>
<tr>
<td>Overall</td>
<td>3.54</td>
<td>61.62</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CPU 2.20GHz (ms)</th>
<th>FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 thread -O3</td>
<td>16 threads -O3</td>
</tr>
<tr>
<td>Power (Watt)</td>
<td>95.00</td>
<td>95.00</td>
</tr>
<tr>
<td>Comparison</td>
<td>5.1x</td>
<td>5.1x</td>
</tr>
<tr>
<td>Energy (J)</td>
<td>35.77</td>
<td>9.83</td>
</tr>
<tr>
<td>Comparison</td>
<td>89.4x</td>
<td>24.6x</td>
</tr>
</tbody>
</table>
Pros and Cons

- Other application accelerators do not balance the physical limitations between bandwidth and computational power. This solution is balanced.
- The implemented CNN accelerator achieves a performance of 61.62 GFLOPS - the highest performance among existing accelerators.
Conclusion

- The paper describes a “roofline-model-based method for convolutional neural network FPGA acceleration”.
- Through optimising, modeling using the roofline model, finding the best layer/cross layer design and then through actual implementation.
- A unique implementation that sheds light into the design space, further solutions can build on this implementation.
References


References