RISC-V Convolutional Neural Network

Alex Heinle, Farris Nefissi

Rensselaer Polytechnic Institute

Introduction

Neural networks are energy and resource intensive, and typical computer hardware is not optimized for such workloads. By designing an application specific architecture with a RISC-V ISA, the execution of a convolutional neural network (CNN) **can be made significantly more efficient than a general-purpose central processing unit**. RISC-V Convolutional Neural Network (RV-CNN) is an extension to the open-source RISC-V ISA to allow for more cycle-efficient execution of convolutional neural networks. In other words, this will be capable of executing a convolutional neural network program in fewer cycles than a general-purpose RISC-V architecture. This extension will include a framework for a convolutional neural network optimized to operate on this hardware.

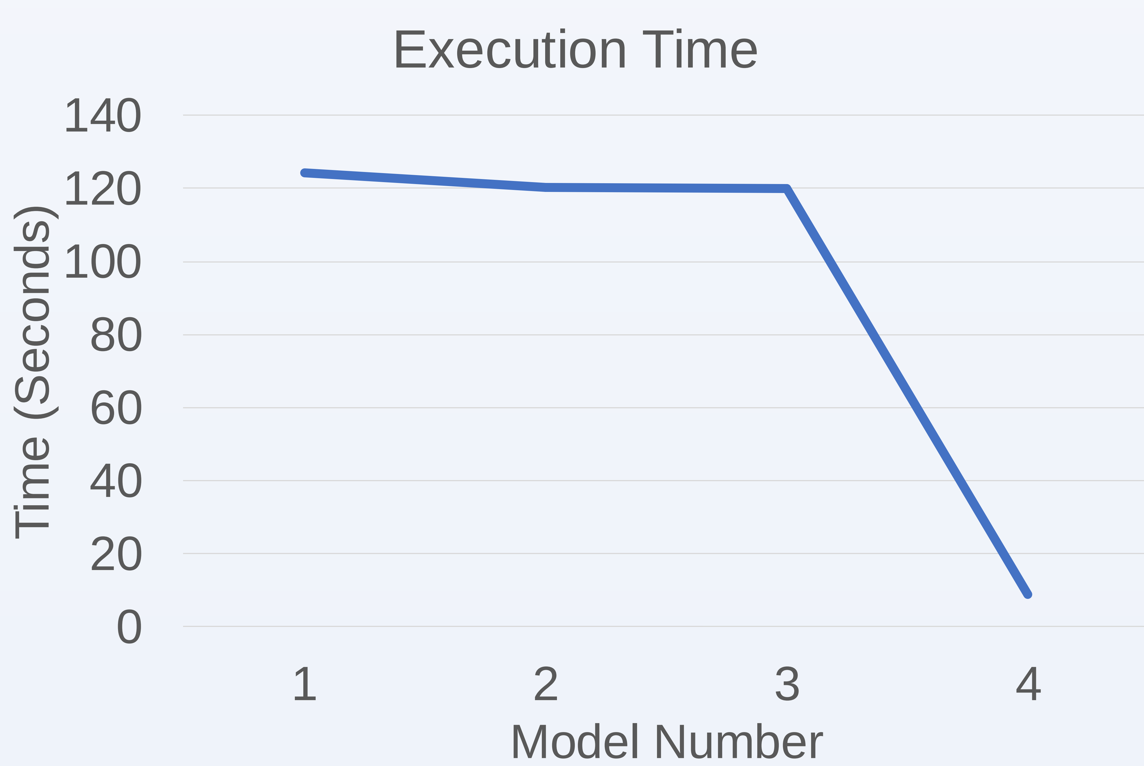
Objectives

The objectives of this project are to:

- Design a framework for a standard convolutional neural network
  - Create **efficient** functions to perform convolutions, pooling, and perform fully connected layers
  - **Minimize** number of layers and neurons required to achieve high efficiency computation
- Implement **application specific** parallelized hardware to be compatible with 5-stage pipeline
  - Can be implemented with minimal modification of the RISC-V ISA
  - Reduces highly repetitive instructions to occur in parallel to increase the cycle efficiency

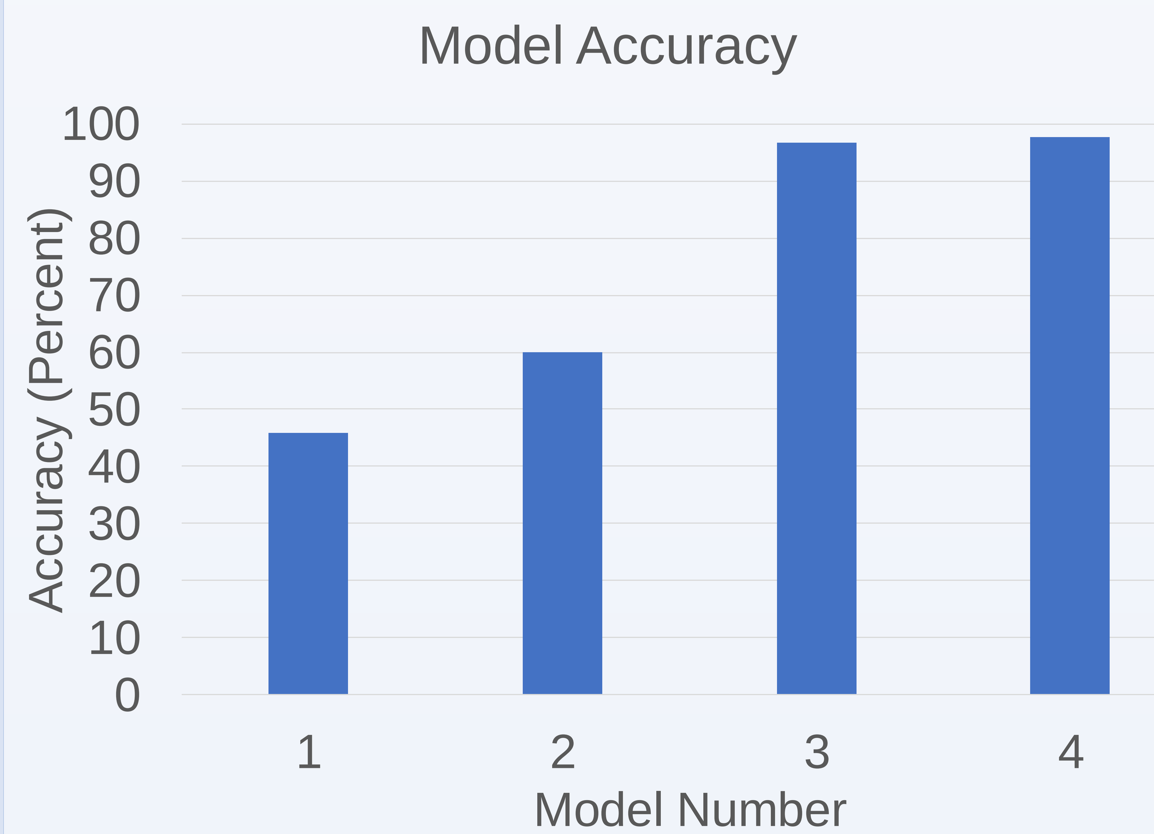
Materials and Methods

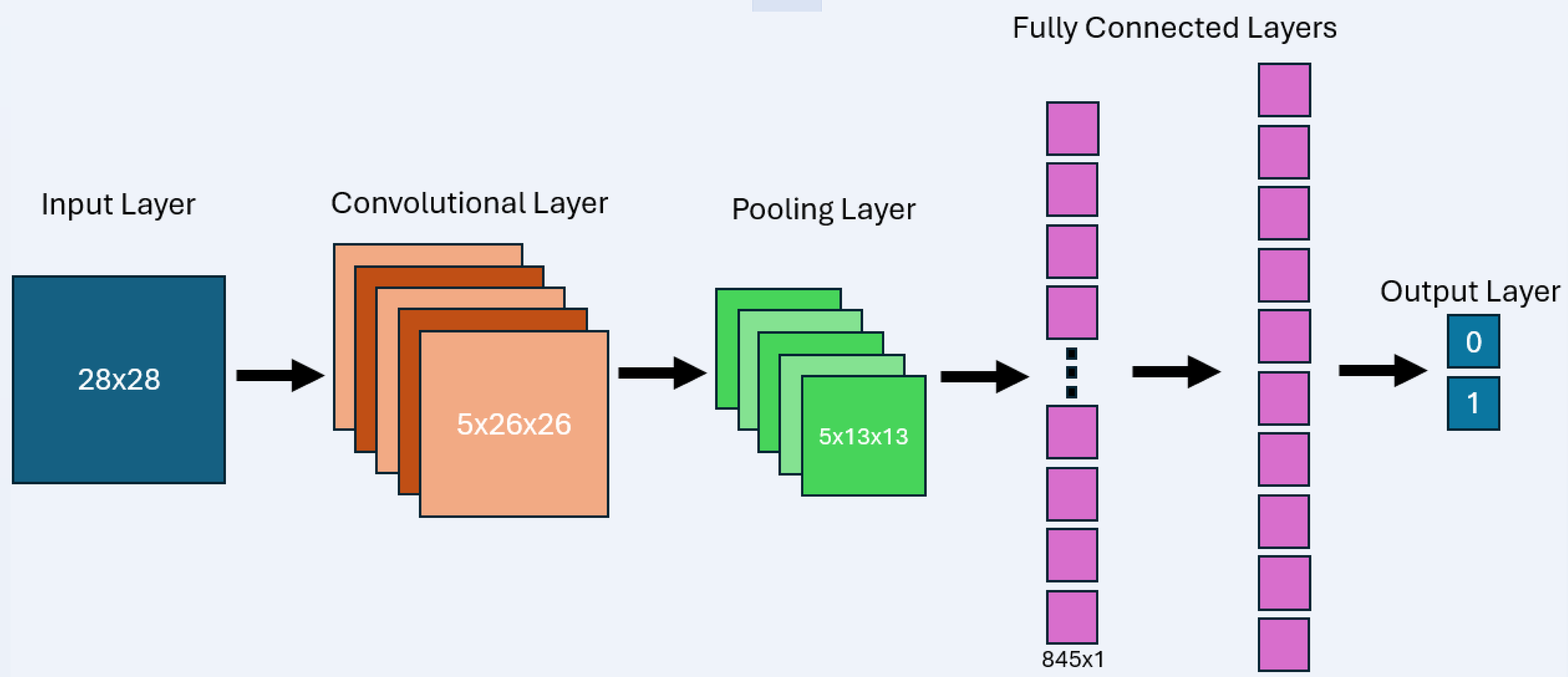
The first step of this project was to determine what architecture the convolutional neural network was going to use. There are many tradeoffs to be made in order to maximize efficiency, so we decided to make a small network with the goal of identifying handwritten 1's and 0's from the MNIST dataset. We began with a TensorFlow model that had a number of optimizations and stripped it back until we started noticing real efficiency losses. We then set about to design this model using standard python libraries, so we designed simple functions that collectively would achieve the same results. We finally converted it to C++ so that it could be compiled to the standard RISC-V ISA.



Results

At the conclusion of this semester, the design has resulted in a 97.65% accuracy for determining if an image is a handwritten 1 or 0. The **first model** had less than random accuracy, at **45.8%**. The second model had 60% accuracy, showing progress, and then the 3rd model was 96.67% accurate, and then after removing the number of decimal points in floats and converting it to C++, the **fourth** design achieved the final accuracy of **97.65%**. Additionally, converting the model to C++ made the program execute **12 times faster** than the equivalent python program.





Conclusions

At this point in time, it is critical that hardware and software design works together to make efficient processes. By making systems on chips to target common workloads, both power usage and time can be minimized.

From the results, we can see the strength of this image classification neural network and the efficiency that transitioning from Python to C++ had. While backpropagation was not implemented in the C++ program due to time constraints, using the weights and biases from our initial Python program, shows that there is a strong correlation between the models and adding the back-propagation network would just widen the scope of the project, but not fundamentally change the results that we obtained.

Some further implementations of this project would be to implement this code on an FPGA board or synthesizing a RISC-V processor.

References

Computerphile. (2015). How Blurs & Filters Work - Computerphile [Video]. YouTube. http://www.youtube.com/watch?v=C_zFhWdM4ic

Computerphile. (2016). CNN: Convolutional Neural Networks Explained - Computerphile [Video]. YouTube. http://www.youtube.com/watch?v=py5byOOHZM8

Computerphile. (2016). Deep Learning - Computerphile [Video]. YouTube. http://www.youtube.com/watch?v=l42lr8AlrHk

Sanderson, G. (n.d.). But what is a Neural Network? 3Blue1Brown. Retrieved April 8, 2025, from https://www.3blue1brown.com/lessons/neural-networks

Sanderson, G. (n.d.). Gradient descent, how neural networks learn. 3Blue1Brown. Retrieved April 8, 2025, from https://www.3blue1brown.com/lessons/gradient-descent

Sanderson, G. (n.d.). What is backpropagation really doing? 3Blue1Brown. Retrieved April 8, 2025, from https://www.3blue1brown.com/lessons/backpropagation

References and Acknowledgments

Thanks Wes.