Introduction

A Convolutional Neural Network (CNN) is a form of artificial intelligence primarily used for image recognition and, in turn, requires the use of high-end processing computers. This work explores creating custom pipelined hardware for the three main stages of a CNN: convolution, rectification (ReLU), and pooling for image classification. The ultimate goal is to create custom hardware for all CNN stages in order to optimize the hardware for analyzing and detecting objects in images.

VHDL is used to design the custom hardware components (convolution, ReLU, and pooling); then this hardware description is mapped onto a field-programmable gate array (FPGA). For hardware verification, the outputs of the implemented custom hardware are compared to the outputs of a floating-point model in MATLAB.

Tools

Software

- Vivado Design Suite 2018.1
- MATLAB

Hardware

- ZYBO Zynq-7000 Development Board

Results

The outputs of the custom hardware loaded onto the FPGA were the same as the outputs from the MATLAB code. An example of inputs/outputs can be seen in Figure 5.

If completely optimized, the designed custom CNN hardware is able to, from the first enable to valid output, complete in 14 clock cycles.

Each input required one cycle, and from the last input to the output is 10 cycles, totaling 14 clock cycles. This can be seen in Figure 6.

Conclusion

In this work, hardware has been designed for the convolution, ReLU, and pooling stages. The designs have been verified through simulations and running it on an FPGA. These steps have contributed to the creation of a fully operating Convolutional Neural Network on custom hardware.

A future goal for this project is to investigate the feasibility of designing custom hardware to implement the fully connected layer (neural network), which will enable the design of a complete hardware architecture for a CNN.