ABSTRACT

Convolutional Neural Network (CNN), a popular machine learning algorithm, has been proven as a highly accurate and effective algorithm that has been used in a variety of applications such as handwriting digit recognition, visual recognition, and image classification. State-of-the-art CNNs are computationally intensive, yet their parallel and modular nature make platforms like Field Programmable Gate Arrays (FPGAs) well suited for the acceleration process. Typically, Convolutional Neural Networks take a very long development round to be implemented or accelerated using FPGAs, hence in this thesis, we propose a VHDL generation tool (VGT), which through VHDL code (CNN architecture) can be on the fly generated for different CNN models (benchmarked and hand-tuned). The generated code or architecture is highly optimized, where it is modular, highly parallel, reconfigurable, scalable, fully pipelined, and adaptive to different CNN models. We demonstrate the automatic VHDL generation tool and its adaptability by implementing a small-scale CNN model “LeNet-5” and a large-scale one “AlexNet”. The generated code for the small-scale model does not incorporate any external memory management for the CNN parameters, whereas parameters are automatically hard-coded as constants unlike how it is typically done for large-scale models. On a Xilinx Virtex-7 running at 200 MHZ, the system is capable of processing up to 125k 28×28 Images per second for LeNet-5 and achieved a peak performance of 611.52 GOP/s for AlexNet.