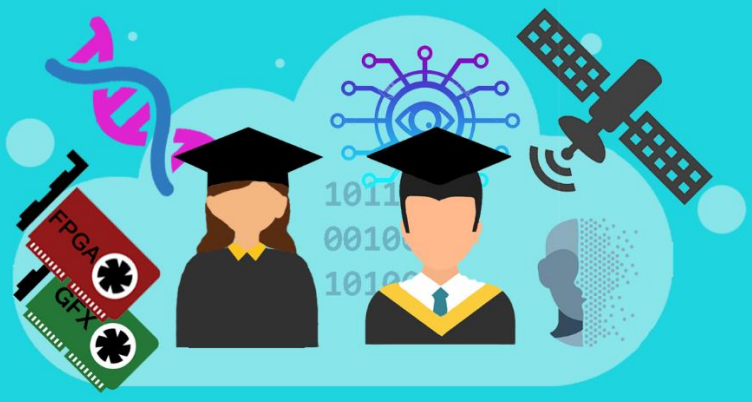
Diploma Thesis

Microproccessors and Digital Systems Laboratory

Academic year 2020-2021



Developing a framework that automatically produces OpenCL accelerator description from CNN model specification

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. CNNs are broadly used in applications such as image classification and speech recognition. However, the computational complexity of neural networks far exceeds traditional computer vision algorithms and today's computational efficiency is not enough for efficient execution of these models. Therefore, it is necessary to implement CNNs in a faster way. For that purpose, various acceleration methods have been developed, such as CNN architecture compression, algorithm optimization, and hardware-based improvement. The latter category includes GPU and FPGA acceleration, using for example the OpenCL programming model. However, there is a wide range of user-defined inference models and each one requires a customized hardware acceleration solution, e.g. a unique OpenCL description. As it is not feasible to manually generate an OpenCL description for all possible CNN models, a tool is required that translates model descriptions to hardware implementations and renders a system flexible enough to execute different neural network models.

The scope of this diploma thesis is to develop an automatic tool that bridges this gap. Caffe, TensorFlow, and other deep learning frameworks provide the user with a unified template for the model description (including number and type of layers e.t.c). There are also available open-source OpenCL implementations of critical layers of CNNs, such as convolution layers, fully-connected layers etc. In this thesis, an automatic tool will be developed that parses the template description of a customized model, identifies layers and operators and generates an OpenCL description for this model based on a library of OpenCL functions for the various types of CNN layers.

PREREQUISITES: OpenCL/HDL programming, Python, Scripting Skills

Academic Advisor1: Konstantina Koliogeorgi, konstantina@microlab.ntua.gr

Academic Advisor2: Sotirios Xydis, sxydis@microlab.ntua.gr

Academic Advisor3: Dimitrios Soudris, dsoudris@microlab.ntua.gr