# General Reading

## FPGA-based accelerator for convolution operations

* <https://ezproxyprod.ucs.louisiana.edu:2373/document/9172934>
* Maps convolution to matrix convolution for flexibility/continuity, mapping this way allows for the acceleration of the convolution to ignore system design constraints
* Proposes a Systolic Array of Processing Elements (PE) to accelerate CNN
* Each PE shifts data and completes MAC operation
* Low level approach to convolution

## Angel-Eye: A Complete Design Flow for Mapping CNN Onto Embedded FPGA

* <https://ezproxyprod.ucs.louisiana.edu:2373/document/7930521>
* Has previous work (Here is a presentation of it): <http://nicsefc.ee.tsinghua.edu.cn/media/publications/2016/FPGA2016_None_slide.pdf>
* Surveys current CNN implementations (as of 2017)
* Proposes a programmable and flexible CNN accelerator architecture
* States that many CNN architectures are too bulky for embedded systems/IoT
* Describes CNN Layers
  + Convolution layer
    - Applies a trained filter value to an input feature map to extract local features. Usually cascade several layers to extract many features.
  + Fully Connected layer
    - Usually, a classifier stage
  + Nonlinearity layer
    - Help with fitting, usually ReLU
  + Pooling layer
    - Down sampling, usually average or max
* States modern CNN prefer smaller (3x3/5x5) kernel size
* Chooses to compress the CNN model
  + States that an effective method is to limit bit width to ~16b or 12b
* This paper is not exactly talking about reconfigurable hardware per say but a design flow to reconfigure a CNN model for different hardware applications

## A FPGA-based Accelerator of Convolutional Neural Network for Face Feature Extraction

* <https://ezproxyprod.ucs.louisiana.edu:2373/document/8754067>
* Quantizes to fixed point
* Propose an RTL-designed hardware architecture to accelerate the entire DeepID CNN module on FPGA target.
* Choose DeepID for face feature extraction
* Pre-trained weights and image data are stored on flash
* Coarse grained parallelism is achieved by allowing a multi-channel input map to be used and apply multi-channel weights to create a single output
* Talks about layers of CNN
  + Pooling
  + ReLU (Non-Linear)
  + Fully Connected
    - Has the most data access
* Proposed model is described in Verilog and implemented on Quartus 2. Have implemented a functional simulation

## Reconfigurable Instruction-Based Multicore Parallel Convolution and Its Application in Real-Time Template Matching

* <https://ieeexplore-ieee-org.ezproxyprod.ucs.louisiana.edu/document/8375740>
* Recent (as of June 2018) space missions have increased volume of data production sparking a need for significantly increasing data throughput of data processing tech
  + Specifically need real-time convolutions processors that are feasible
* Propose a convolution instruction to optimize convolution computing
* Other researchers have proposed application specific acceleration engines on ASIC / FPGA / GPU
* Again this is not really reconfigurable hardware, creates an architecture based around an instruction set

## An Energy-Efficient and Flexible Accelerator based on Reconfigurable Computing for Multiple Deep Convolutional Neural Networks

* <https://ieeexplore-ieee-org.ezproxyprod.ucs.louisiana.edu/document/8565823>
* Common CNN architecture is not very flexible, this is an issue as the layer size diversity is drastic in larger models
* Propose a Reconfigurable Neural Accelerator (RNA) is designed for adapting to neural network evolution and can easily change CNN shapes like AlexNet, VGG, and Lenet-5.

## Reconfigurable Convolution Architecture for Heterogeneous Systems-on-Chip

* <https://ieeexplore-ieee-org.ezproxyprod.ucs.louisiana.edu/document/9134344>

## A software controlled hardware acceleration architecture for image processing using an embedded development board

* <https://ezproxyprod.ucs.louisiana.edu:2373/document/7942352>

## Optimizing the Convolution Operation to Accelerate Deep Neural Networks on FPGA

* <https://ezproxyprod.ucs.louisiana.edu:2373/document/8330049>

## A FPGA-based Hardware Accelerator for Multiple Convolutional Neural Networks

* <https://ezproxyprod.ucs.louisiana.edu:2373/document/8565657>

## A Method for Accelerating Convolutional Neural Networks Based on FPGA

* <https://ezproxyprod.ucs.louisiana.edu:2373/document/9151535>

## Hardware Implementation of Reconfigurable Separable Convolution

* <https://ieeexplore-ieee-org.ezproxyprod.ucs.louisiana.edu/document/8429372>

## A Reconfigurable Streaming Deep Convolutional Neural Network Accelerator for Internet of Things

* <https://ieeexplore-ieee-org.ezproxyprod.ucs.louisiana.edu/document/8011462>

## Methodology for Efficient Reconfigurable Architecture of Generative Neural Network

* <https://ieeexplore-ieee-org.ezproxyprod.ucs.louisiana.edu/document/8702807>

## ImageNet classification with deep convolutional neural networks

* <https://dl.acm.org/doi/abs/10.1145/3065386>

# Using PYNQ Architecture API

Reconfigurable Real-Time Video Pipelines on SRAM-based FPGAs

* <https://ieeexplore-ieee-org.ezproxyprod.ucs.louisiana.edu/document/8994814>