Assignment : 2 HCSD
Name: Ganesh Biradar
ID : 2023ht01609

*** Finding ways for accelerating ways for CNN's On FPGA using Hw-Sw Co-Design**

CNN :
Convolution neural network is power type of deep learning architecture designed specifically for Image, Video, signal processing tasks. Widely used in computer vision applications for object detection, Image classification, Image segmentation. In other fields like Natural language processing, Medical image analysis, time series analysis.

For accelerating the CNN's on FPGA there could be 2 ways with Hardware& Software both needs to be optimized. First understand the structure & computational requirements of CNN's model you want to accelerate.

1. Hardware Optimizations

A . Pipe-lining & parallelism : By Implementing the parallel architectures allows processing concurrently within different layers of CNN. Spatial parallelism, Layer parallelism, Channels parallelism.

B. Customizable Multiply-Accumulate(MAC) units: Explore configurable MAC units that can adjust data precision based on CNN layers for balance between performance & accuracy

C. Dataflow Optimization: Analyse data-flow between CNN layers & optimize the data movement on FPGA.

2. Software optimization

A. Communication Optimization : design efficient communication protocol between software & hardware. Explore techniques like DMA's

B. Layer Mapping: Analyse the CNN architecture to identify layers most suitable for hardware acceleration with high compute intensity & regular data access patterns.

C. Data Pre & post processing : utilize software running on CPU for preprocessing(normalization & scaling) & postprocessing(converting outputs to possibilities tasks).frees up the FPGA for compute intensive task of CNN.

Co-design techniques:
1. High-Level synthesis :
We can utilize HLS to tools, HLS allows the to describe the CNN algorithm in C-like languages & then translate it into hardware code suitable for FPGA.

2. Runtime reconfiguration :
Exploring the techniques of partially re configuring the FPGA at runtime this allows harware accelerators to adapt different CNN architectures on the fly. We will dicuss this in details below.

Dynamic partial reconfiguration : DPR is advanced technique used for Field programmable gate arrays(FPGAs) to modify specific region of the FPGA fabric. While the rest of the device continues operating. This allows the on fly updates to the functionality of FPGA

DPR Granularity can be achieved by partitioning hardware design into individual modules suitable for dynamic reconfiguration. Define boundaries between static & re configurable regions.

DPR implementation :
A. FPGA design Tools:

B. Reconfigurable controller : implement dedicated hardware module on FPGA to manage reconfiguration process.

C. Synchronization mechanism : Design sync mechanism between static & reconfigurable regions to ensure the sooth handoff during the reconfiguration.

* Each reconfigurable region in FPGA has its own bitstream file which acts as blueprint for reconfiguration. Bit-streams are typically generated using tools provided FPGA vendor. By understanding the bitstream in DPR we can effectively leverage the technique to achieve the efficient & flexible CNN execution.

Design Integration & Testing :
A. Integration &Functional verification : simulate the entire system to ensure the functional correctness of CNN implementation.

B. Performance Profiling : Identify the bottleneck in both hardware & software using profiling tools which will guide us in further optimization .

Summary :
Accelerating CNN's on FPGA using hw-sw co design has 2 major groups as name suggests hardware optimization & software optimization. Any FPGA has4 commonly identified components FlipFlop, Lookup tables(LUT), Digital signal processors(DSPs), & Block RAM all plays major role in its working. Co-design techniques, Dynamic Reconfirmations, HwSw optimizations play major role in Accelerating the CNN's algorithms. With DPR CNN is able to process 3 dimensional input data directly unlike other traditional artificial neural networks.