Neural Net Implemntation on FPGA's: Goals

Cory Nezin, Brenda So

September 6, 2017

1. Design and implement hardware optimized inference on an FPGA.
   (a) Create a framework to convert symbolic neural nets to FPGA code
      i. ReLU, Sigmoid, Threshold
      ii. Fully Connected (matrix multiply - with Strassen algorithm?)
      iii. Sparsely Connected? (Sparse martrix multiply)
      iv. 1D (2D and 3D also?) Convolution - just FIR filters
      v. Max pooling, average pooling, downsampling
      vi. Recurrent elements?
   (b) Implement weight compression to decrease neural net size.
      i. Use multiplierless multiplication?
      ii. Use SVD for fully connected compression?
      iii. Implement Optimal Brain Damage.
      iv. Implement Optimal Brain Surgeon.
      v. Implement Deep Compression for specialized hardware.
2. Benchmark performance of FPGA's against GPU's and CPU's.
3. Build an "analog" front end.
   (a) Radar - Gender id, people in room
   (b) Style Transfer - live feed
   (c) Audio - Frank's speech denoising
   (d) Video - lipnet, live blackboard transcription
   (e) Image - Handwriting transcription

(f) Lidar - Car or pedestrian detection

(g) Wireless - Signal classification, smart jamming?