Reducing Dynamic Power in Streaming CNN Hardware Accelerators by Exploiting Computational Redundancies

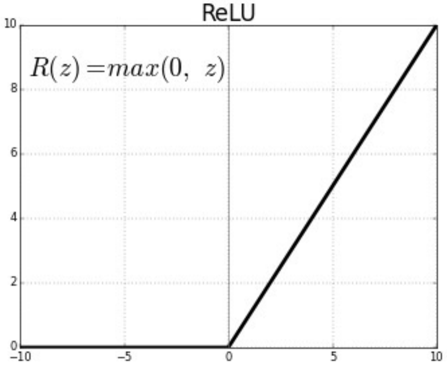
Duvindu Piyasena, Rukshan Wickramasinghe, Debdeep Paul,
Siew-Kei Lam and Meiqing Wu
School of Computer Science and Engineering (SCSE)
Nanyang Technological University (NTU)
Singapore

Email: siewkei_lam@pmail.ntu.edu.sg

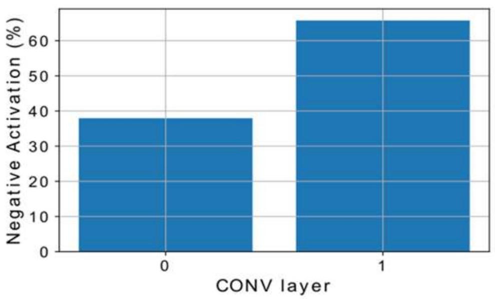
NANYANG
TECHNOLOGICAL
UNIVERSITY

Motivation

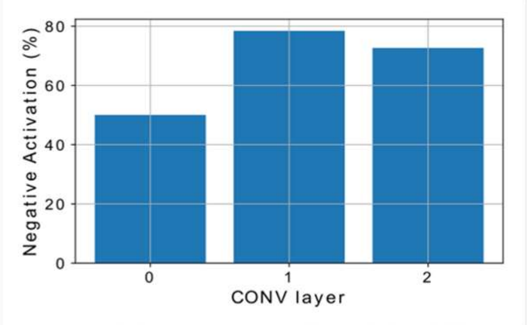
- ReLU discards negative convolution activations causing **high computational redundancy in CNNs**.
- Widely-used CNN models discard **30%-90%** CONV activations in a given layer.



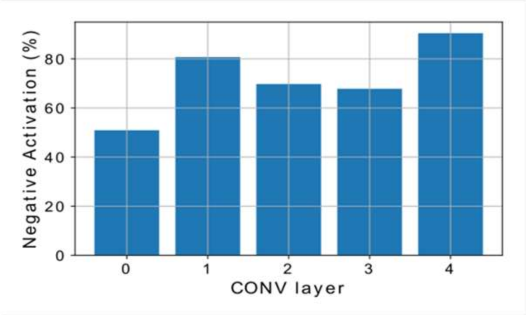
ReLU activation function



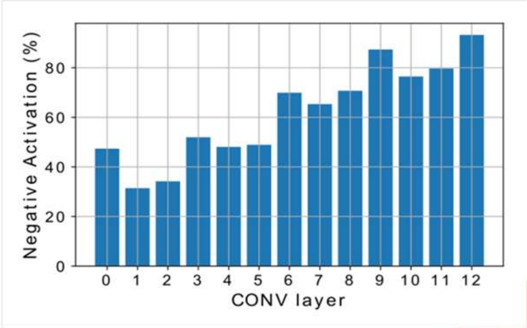
Lenet (MNIST)



CIFAR10-Quick (CIFAR10)



AlexNet (Imagenet)

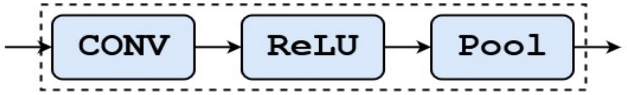


VGG16 (Imagenet)

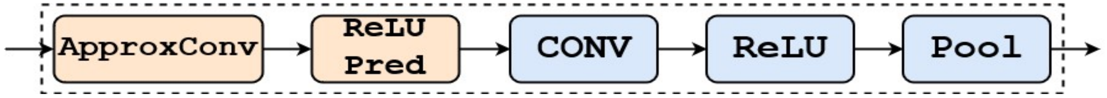
Proposed Method

- We propose a method to eliminate the computational redundancies to save dynamic power in ***FPGA stream-based CNN accelerators***
- Eliminates the computational redundancies arising from ReLU activation **by predicting the positive/negative CONV activations using a low-cost approximation scheme**.

Conventional CNN layer



Proposed CNN layer



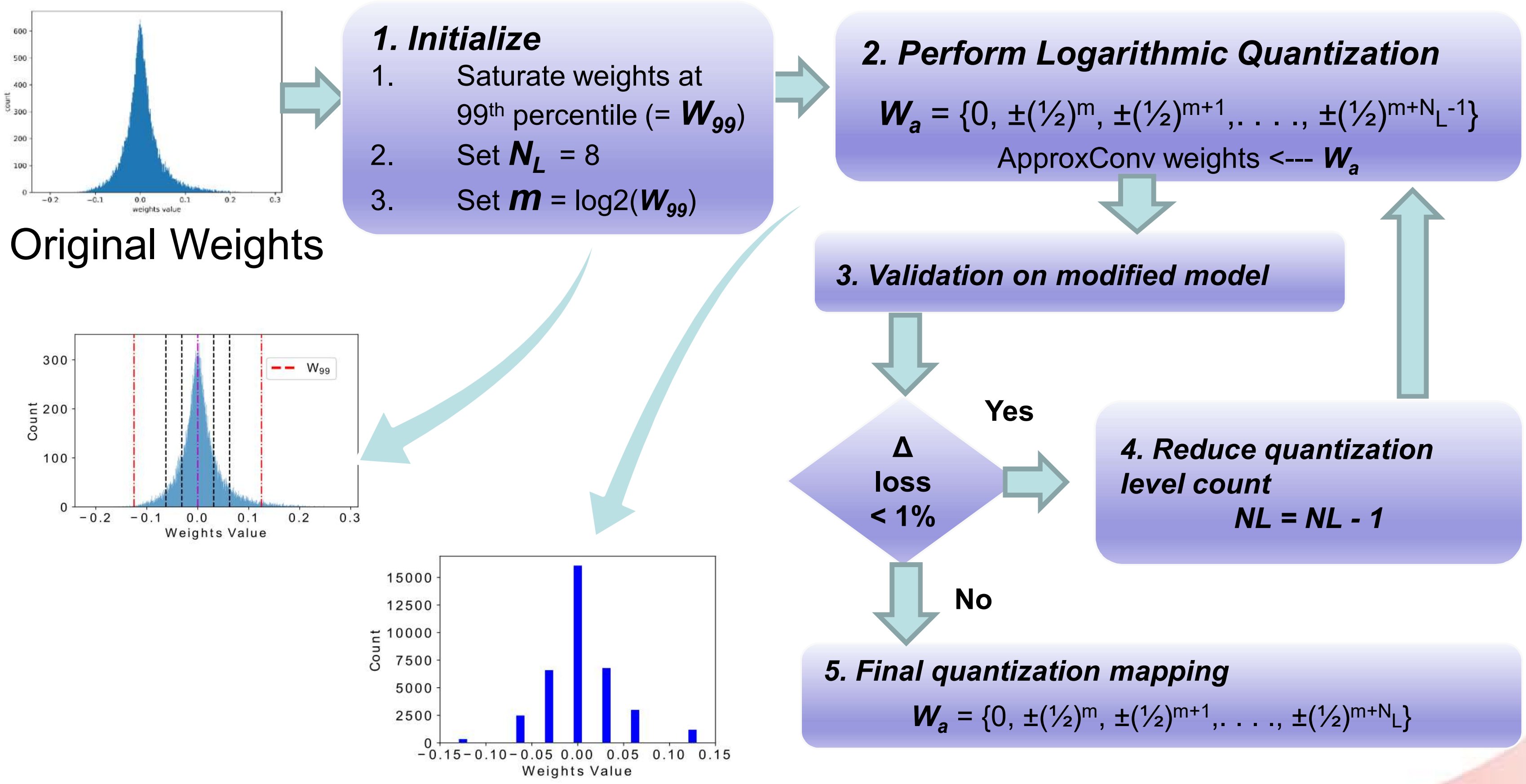
Contribution

- We propose a **hardware-friendly convolution approximation method** that rely on power-of-two quantized weights.

- We show that the proposed methodology can be applied to various CNN models to **significantly reduce the convolution operations, without compromising on the accuracy or retraining**.

- We propose a **streaming CNN FPGA accelerator** that integrates our approximation method and demonstrate that **notable power/energy savings can be achieved**.

Proposed Method



Original Weights

1. Initialize

1. Saturate weights at 99th percentile (= $\boldsymbol{W_{99}}$)
2. Set $\boldsymbol{N_L}$ = 8
3. Set $\boldsymbol{m}$ = log2($\boldsymbol{W_{99}}$)

2. Perform Logarithmic Quantization

$\boldsymbol{W_a}$ = $\{0, \pm(½)^{m}, \pm(½)^{m+1}, ..., \pm(½)^{m+N_L-1}\}$

ApproxConv weights <--- $\boldsymbol{W_a}$

3. Validation on modified model

Δ loss < 1%

4. Reduce quantization level count

NL = NL - 1

5. Final quantization mapping

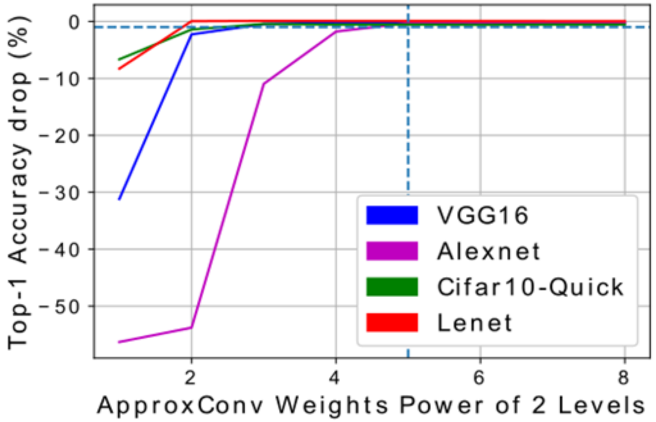
$\boldsymbol{W_a}$ = $\{0, \pm(½)^{m}, \pm(½)^{m+1}, ..., \pm(½)^{m+N_L}\}$

NANYANG TECHNOLOGICAL UNIVERSITY

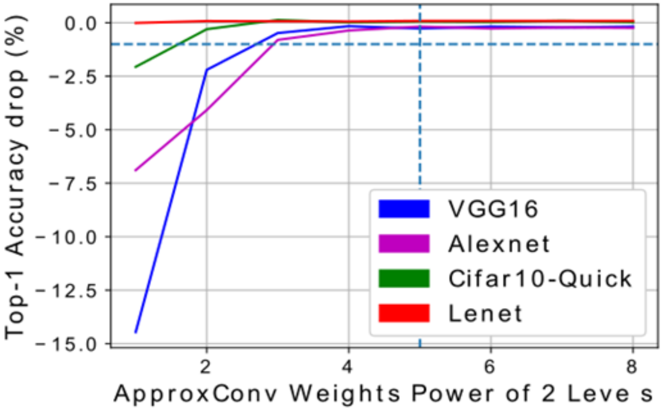
Implementation

- Quantization level search

  Evaluated designs :
  - ***Prop - 1*** : Approximation applied across all-layers
  - ***Prop - 2*** : Approximation applied across all-layers except 1st



Prop-1



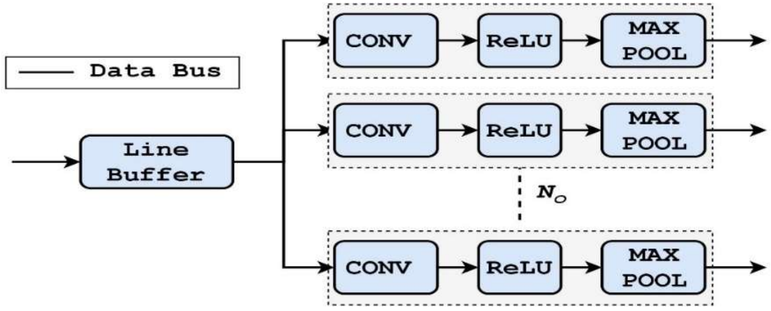
Prop-2

NANYANG
TECHNOLOGICAL
UNIVERSITY

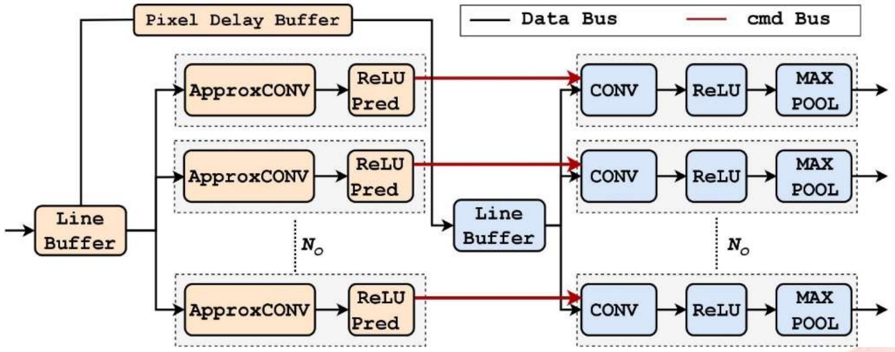
Implementation

- Implementation done in Verilog HDL for Lenet
  - **Operating Frequency** : 100Mhz
  - **Device** : Xilinx Virtex Ultrascale+ xcvu9p
  - **Synthesize tool** : Xilinx Vivado 2018.3
  - **Simulator** : Mentor Modelsim 10.3
  - **Power Estimation Mode** : Post-Synthesis Timing Simulations

- Power Gains achieved by clock gating CONV circuitry via ApproxConv predictions



Baseline HW (single layer)



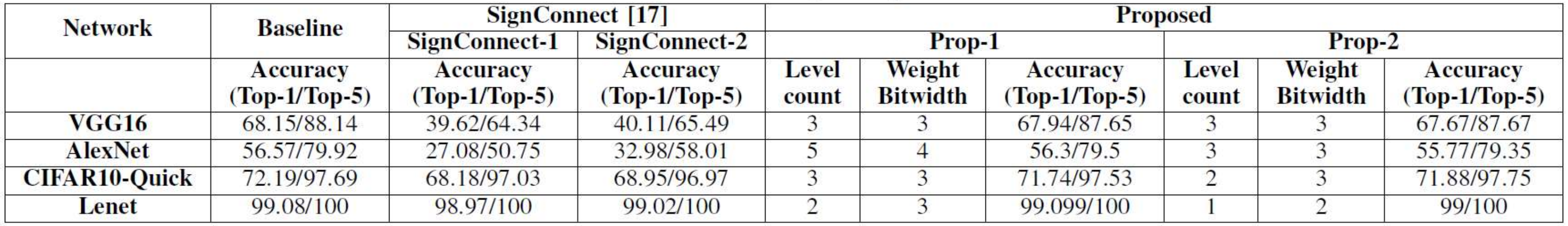
Proposed HW (single layer)

NANYANG TECHNOLOGICAL UNIVERSITY

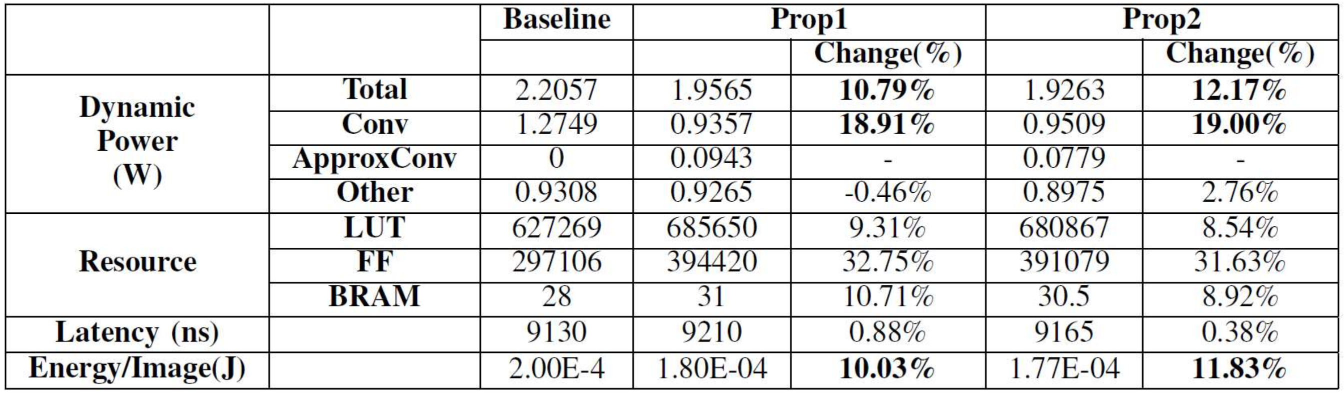
Accuracy and Hardware Evaluations

- Compared with ***Signconnect*** proposed in previous work(*), which uses the sign of the weights to perform the approximations
  - ***SignConnect-1*** **:** Approximation applied across all-layers
  - ***SignConnect-2*** **:** Approximation applied across all-layers except 1st

TABLE I. Accuracy comparisons



Network	Baseline	SignConnect [17]		Proposed					
		SignConnect-1	SignConnect-2	Prop-1			Prop-2		
	Accuracy (Top-1/Top-5)	Accuracy (Top-1/Top-5)	Accuracy (Top-1/Top-5)	Level count	Weight Bitwidth	Accuracy (Top-1/Top-5)	Level count	Weight Bitwidth	Accuracy (Top-1/Top-5)
VGG16	68.15/88.14	39.62/64.34	40.11/65.49	3	3	67.94/87.65	3	3	67.67/87.67
AlexNet	56.57/79.92	27.08/50.75	32.98/58.01	5	4	56.3/79.5	3	3	55.77/79.35
CIFAR10-Quick	72.19/97.69	68.18/97.03	68.95/96.97	3	3	71.74/97.53	2	3	71.88/97.75
Lenet	99.08/100	98.97/100	99.02/100	2	3	99.099/100	1	2	99/100



		Baseline	Prop1		Prop2	
				Change(%)		Change(%)
Dynamic Power (W)	**Total**	2.2057	1.9565	**10.79%**	1.9263	**12.17%**
	Conv	1.2749	0.9357	**18.91%**	0.9509	**19.00%**
	ApproxConv	0	0.0943	-	0.0779	-
	Other	0.9308	0.9265	-0.46%	0.8975	2.76%
Resource	**LUT**	627269	685650	9.31%	680867	8.54%
	FF	297106	394420	32.75%	391079	31.63%
	BRAM	28	31	10.71%	30.5	8.92%
Latency (ns)		9130	9210	0.88%	9165	0.38%
Energy/Image(J)		2.00E-4	1.80E-04	**10.03%**	1.77E-04	**11.83%**

* T. Ujiie, M. Hiromoto, and T. Sato, "Approximated prediction strategy for reducing power consumption of convolutional neural network processor," in *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2016, pp. 870–876

Summary

- Methodology to determine the minimal number of power-of-two quantization levels for realizing lightweight convolution approximations that can predict the positive and negative convolution activations.
- Proposed a streaming CNN FPGA accelerator that integrates our approximation method.
- FPGA synthesis results show that the dynamic power can be reduced by 10-12% while maintaining good accuracy.

Thank You

NANYANG
TECHNOLOGICAL
UNIVERSITY