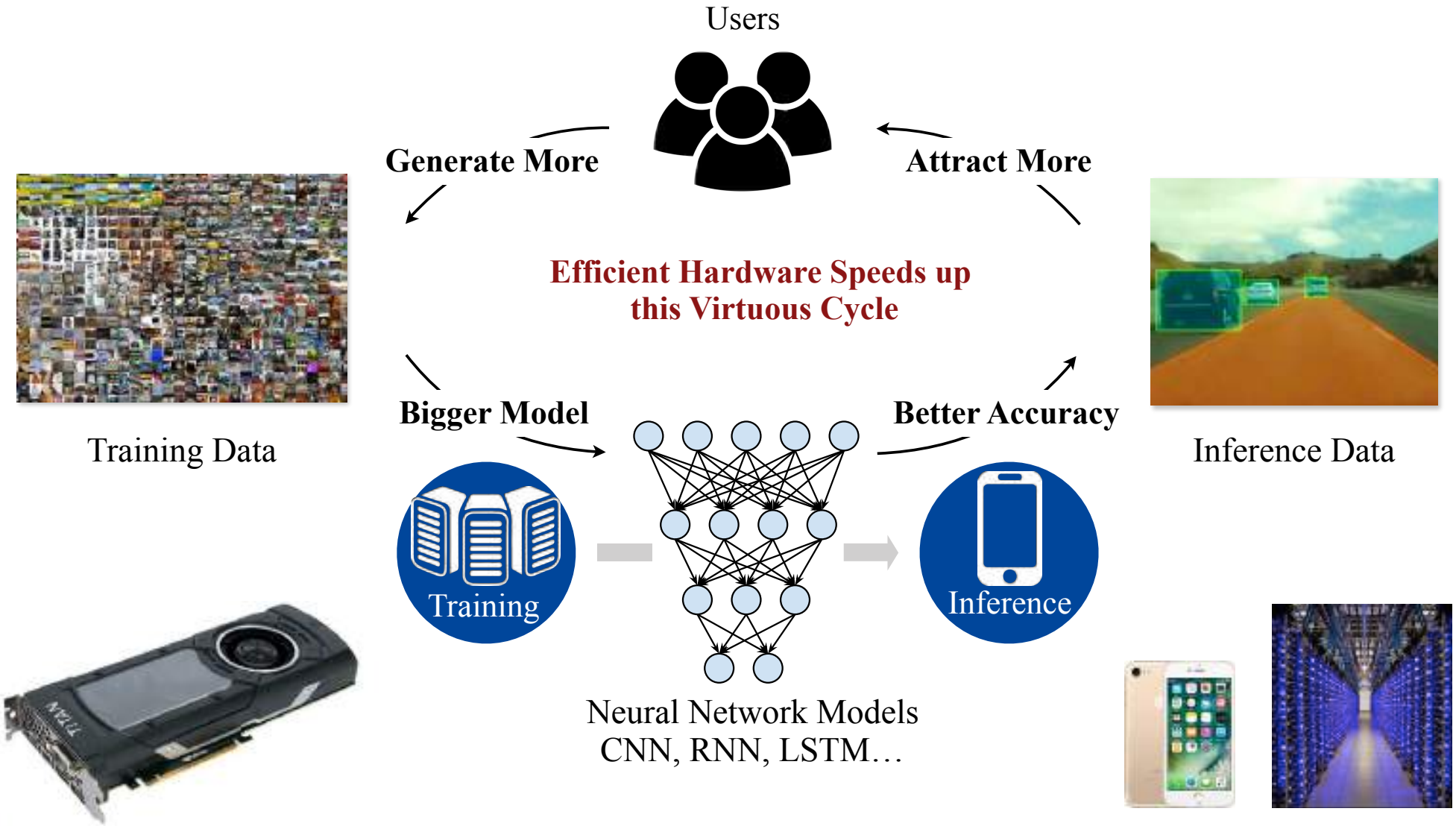
Accelerating Inference at the Edge

Song Han

Assistant Professor
Massachusetts Institute of Technology

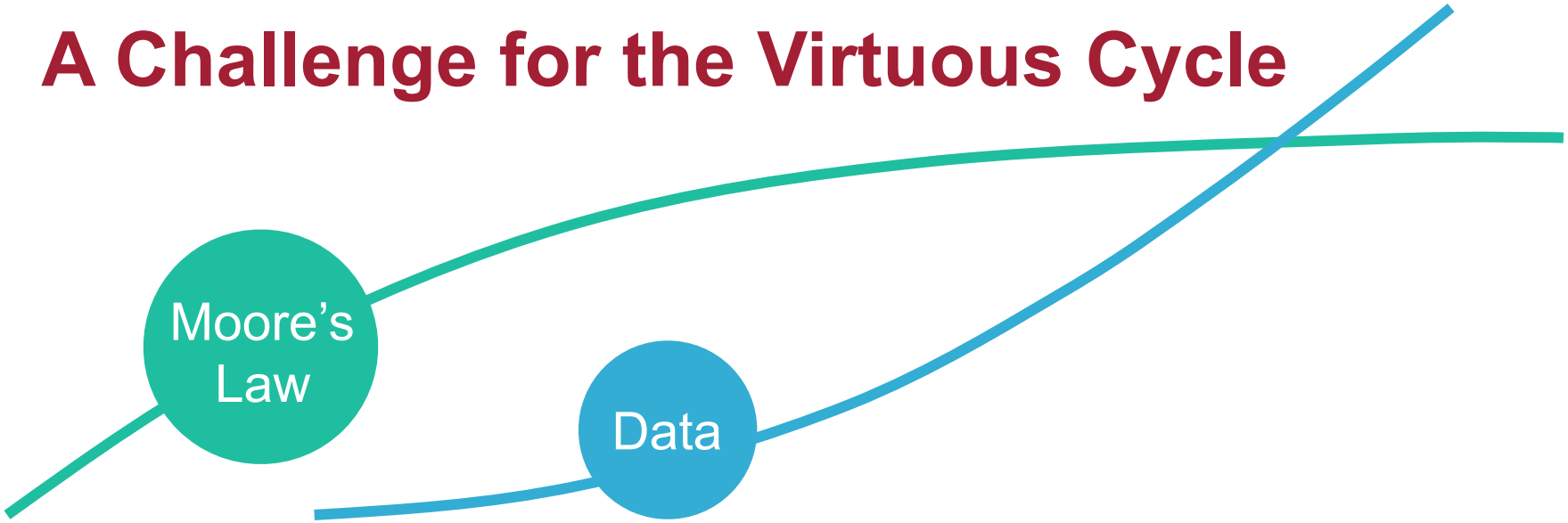
77 Massachusetts Avenue, 38-344
Cambridge, MA, 02139
https://songhan.mit.edu

The Virtuous Cycle of Deep Learning & Hardware



Song Han, Stanford PhD thesis. https://purl.stanford.edu/qf934gh3708

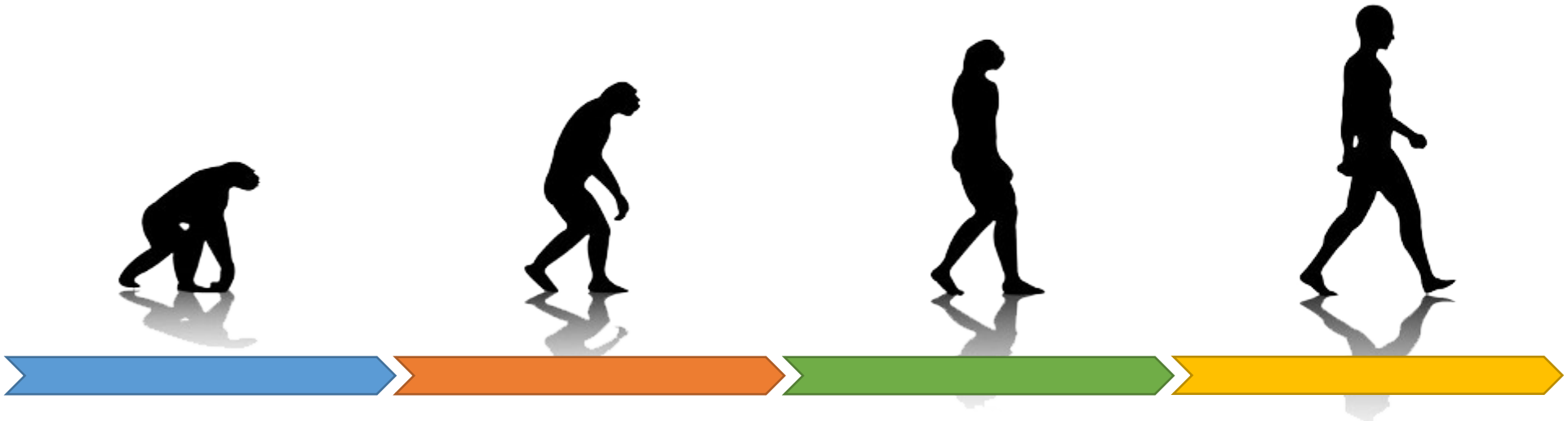
A Challenge for the Virtuous Cycle



- In the **post-ImageNet** era, ImageNet is becoming MNIST; we are solving more complicated AI problems using larger data sets driving the demand for more computation.

- However, we are in the **post-Moore's Law** world where the amount of computation per unit cost and power is no longer increasing at its historic rate.

Evolution of NN Accelerators



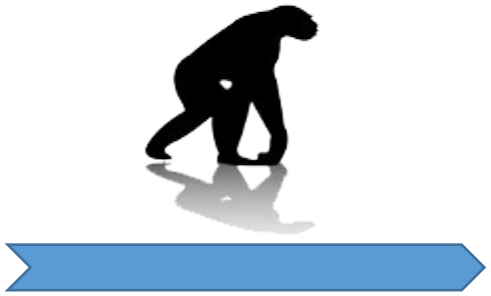
Computation Specialization

Memory-centric Specialization

Compression before Acceleration

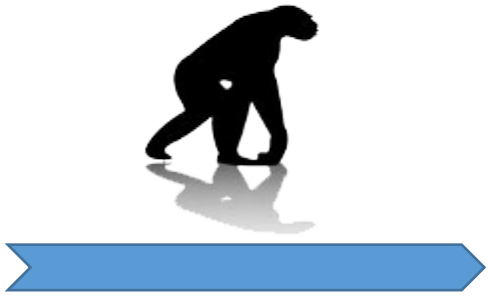
Hardware & NN Architectures Co-Design

Evolution of NN Accelerators

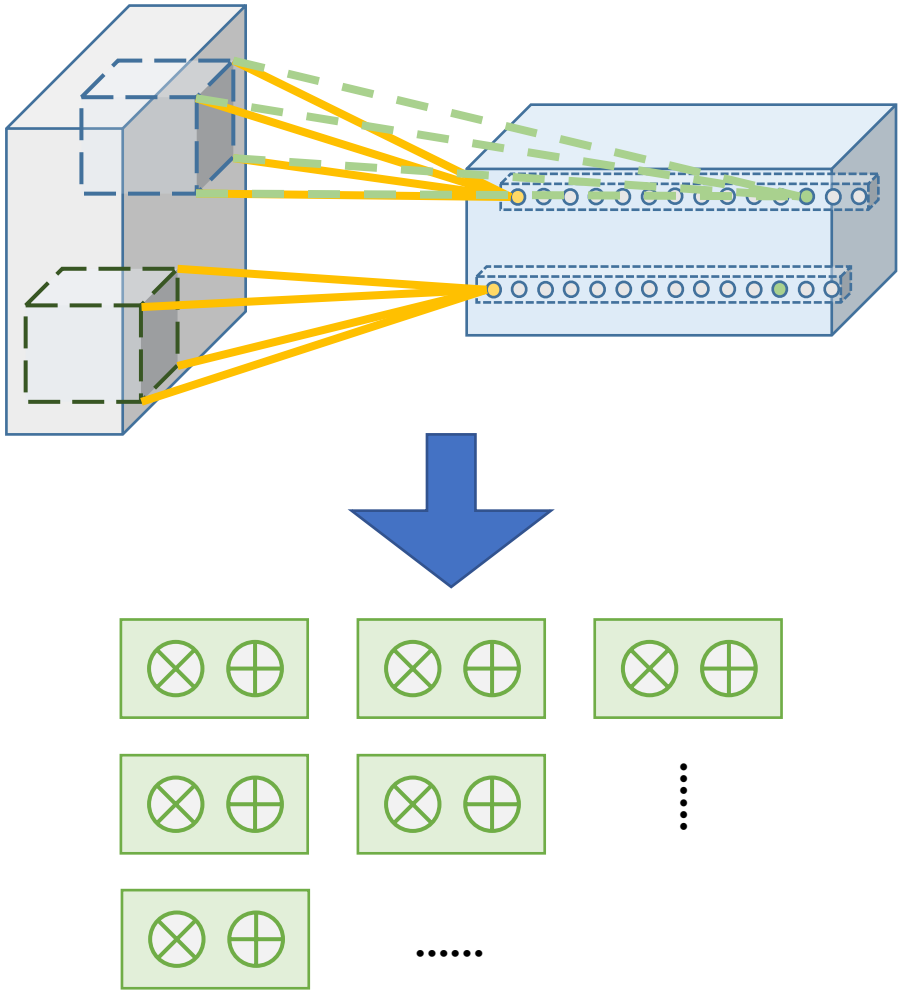


Computation Specialization

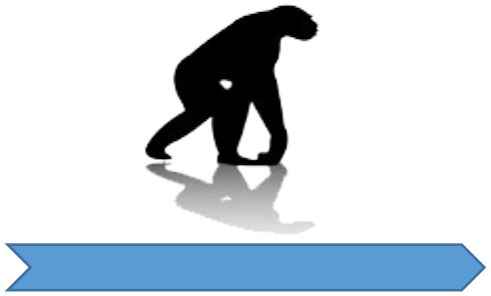
Part 1/4: Computation Specialization



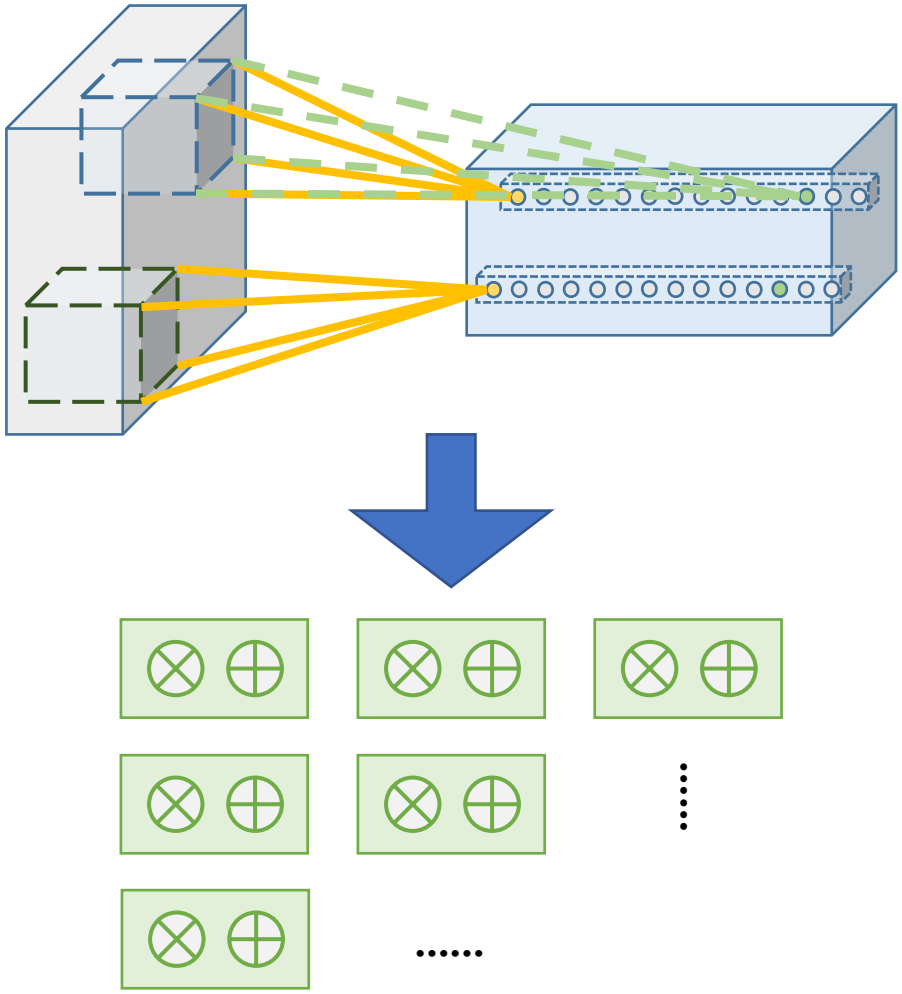
Computation Specialization



Part 1/4: Computation Specialization



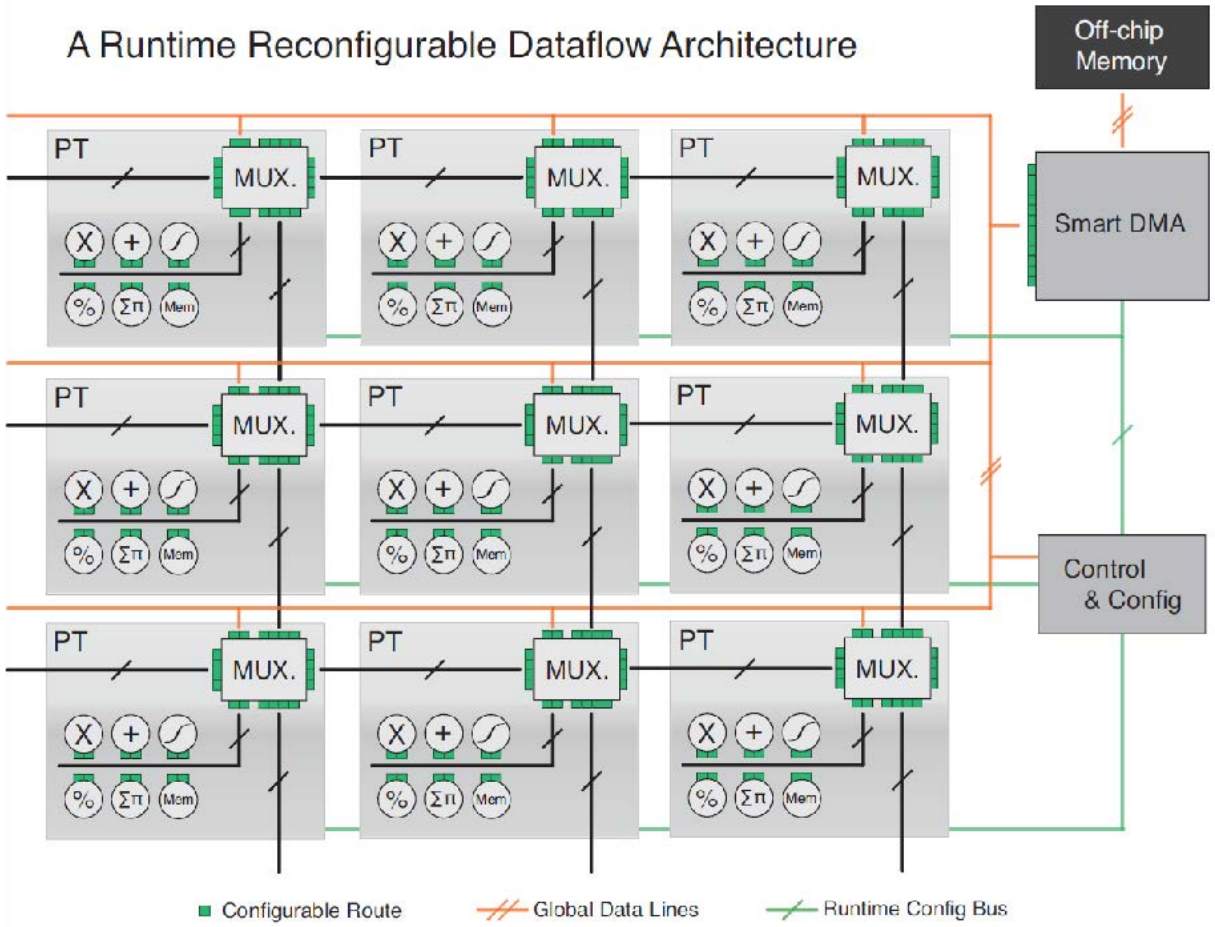
Computation Specialization



- NeuFlow [Farabet, 2011]
- TPU-V1 [Jouppi, production in 2015]

Part 1/4: Computation Specialization

NeuFlow [Farabet, 2011]



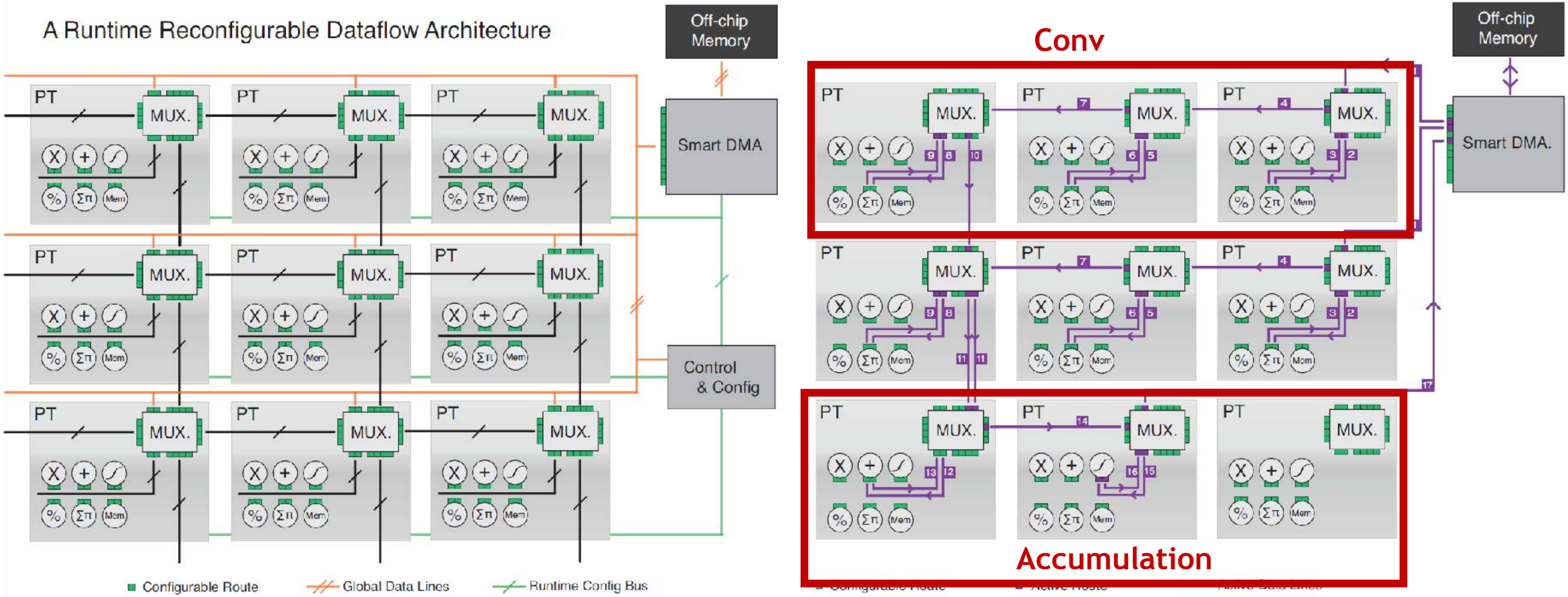
Runtime Reconfigurable Bus

- Connections
- Operations
- etc

Clement Farabet, Berin Martini, Benoit Corda, Polina Akselrod, Eugenio Culurciello, Yann LeCun
"NeuFlow: A Runtime Reconfigurable Dataflow Processor for Vision"

Part 1/4: Computation Specialization

NeuFlow [Farabet, 2011]



Clement Farabet, Berin Martini, Benoit Corda, Polina Akselrod, Eugenio Culurciello, Yann LeCun
"NeuFlow: A Runtime Reconfigurable Dataflow Processor for Vision"

Part 1/4: Computation Specialization

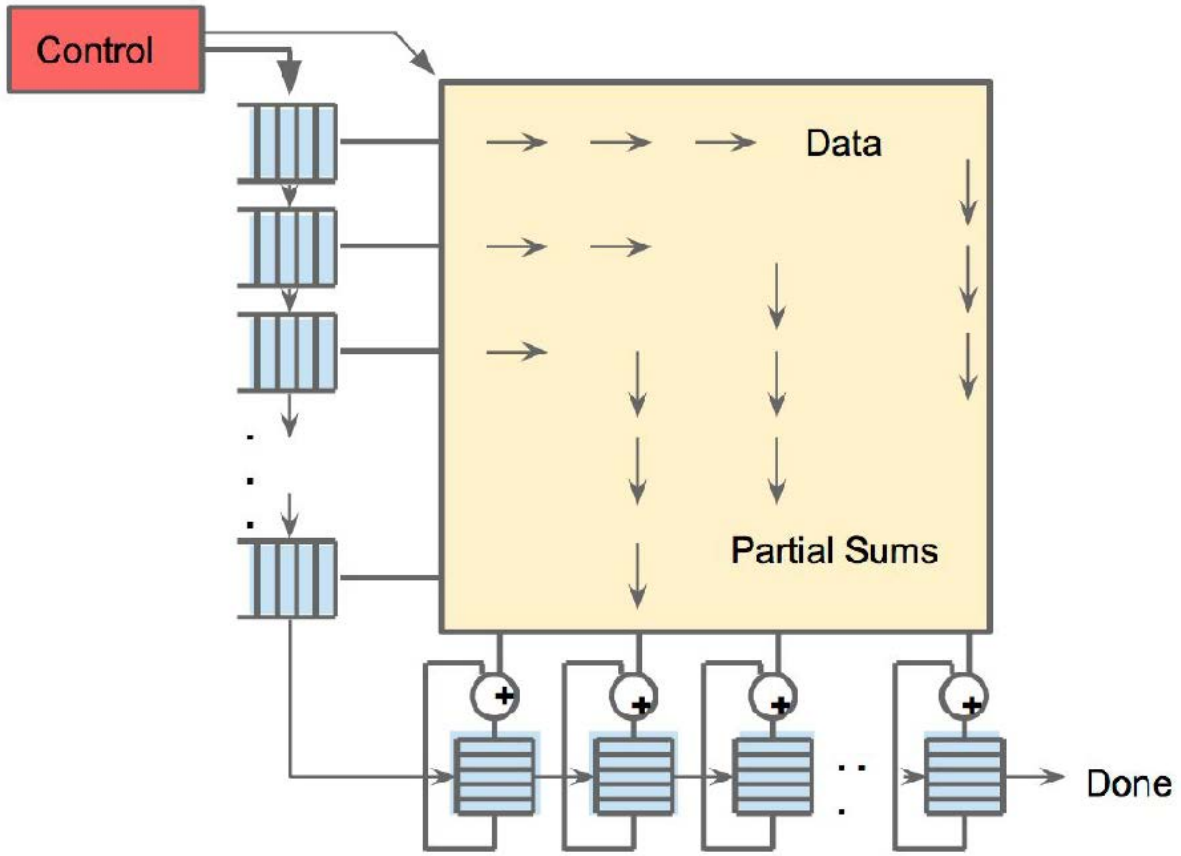
NeuFlow [Farabet, 2011]

	CPU (Intel DuoCore, 2.7GHz)	GPU (GTX480)	mGPU (GT335m)	NeuFlow on Xilinx Virtex 6	NeuFlow on IBM 45 nm process
Peak GOPs	10	1350	182	160	1280
Real GOPs	1.1	294	54	147	1164
Power (W)	30	220	30	10	5
GOPs/W	0.04	1.34	1.8	14.7	230

Clement Farabet, Berin Martini, Benoit Corda, Polina Akselrod, Eugenio Culurciello, Yann LeCun
"NeuFlow: A Runtime Reconfigurable Dataflow Processor for Vision"

Part 1/4: Computation Specialization

TPU [Jouppi, 2017]

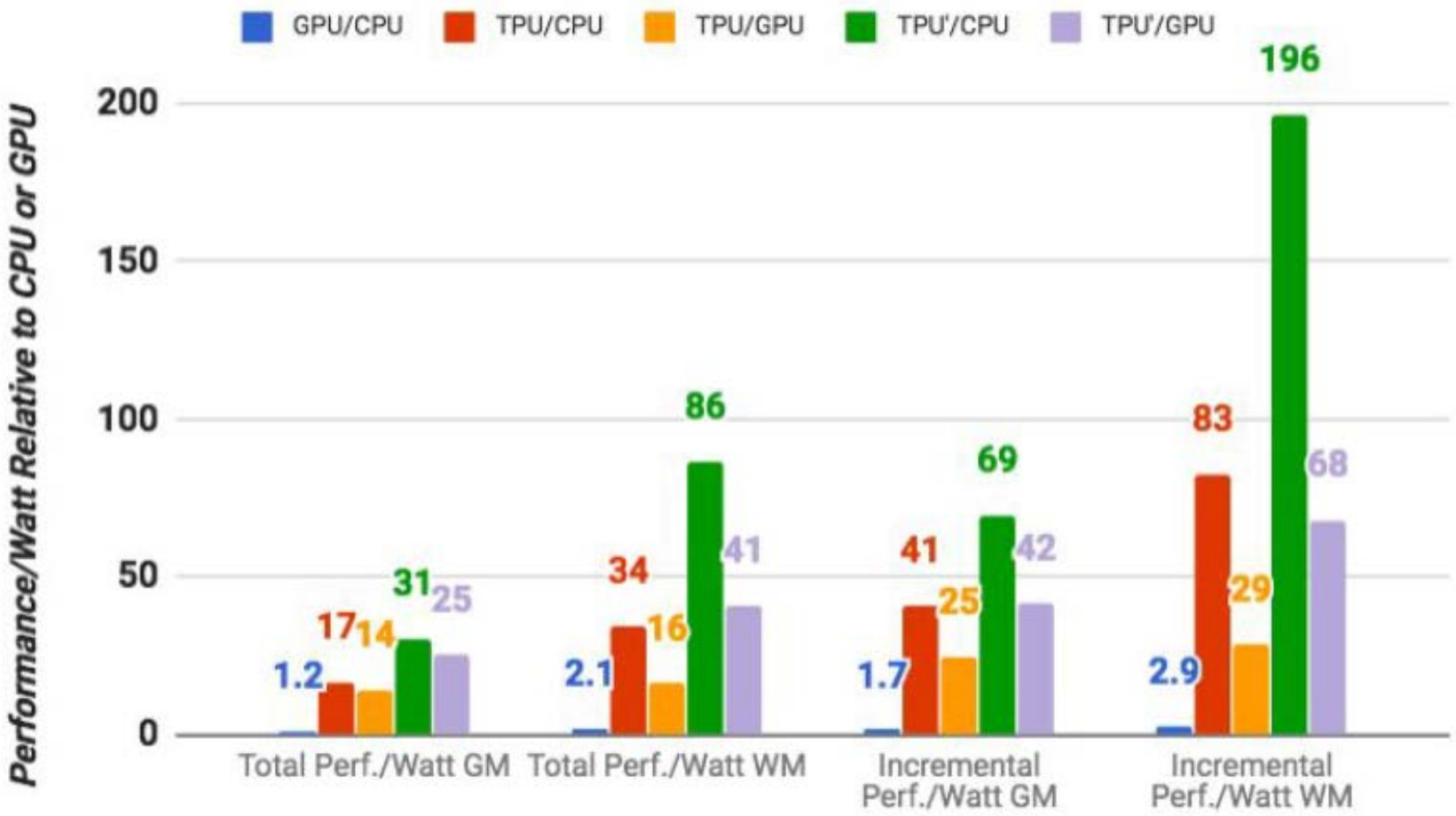


Systolic Array Architecture

- **Flexible: CNN/RNN/LSTM/MLP**
- **Latency matters**
- **Simple silicon, complicated compiler**

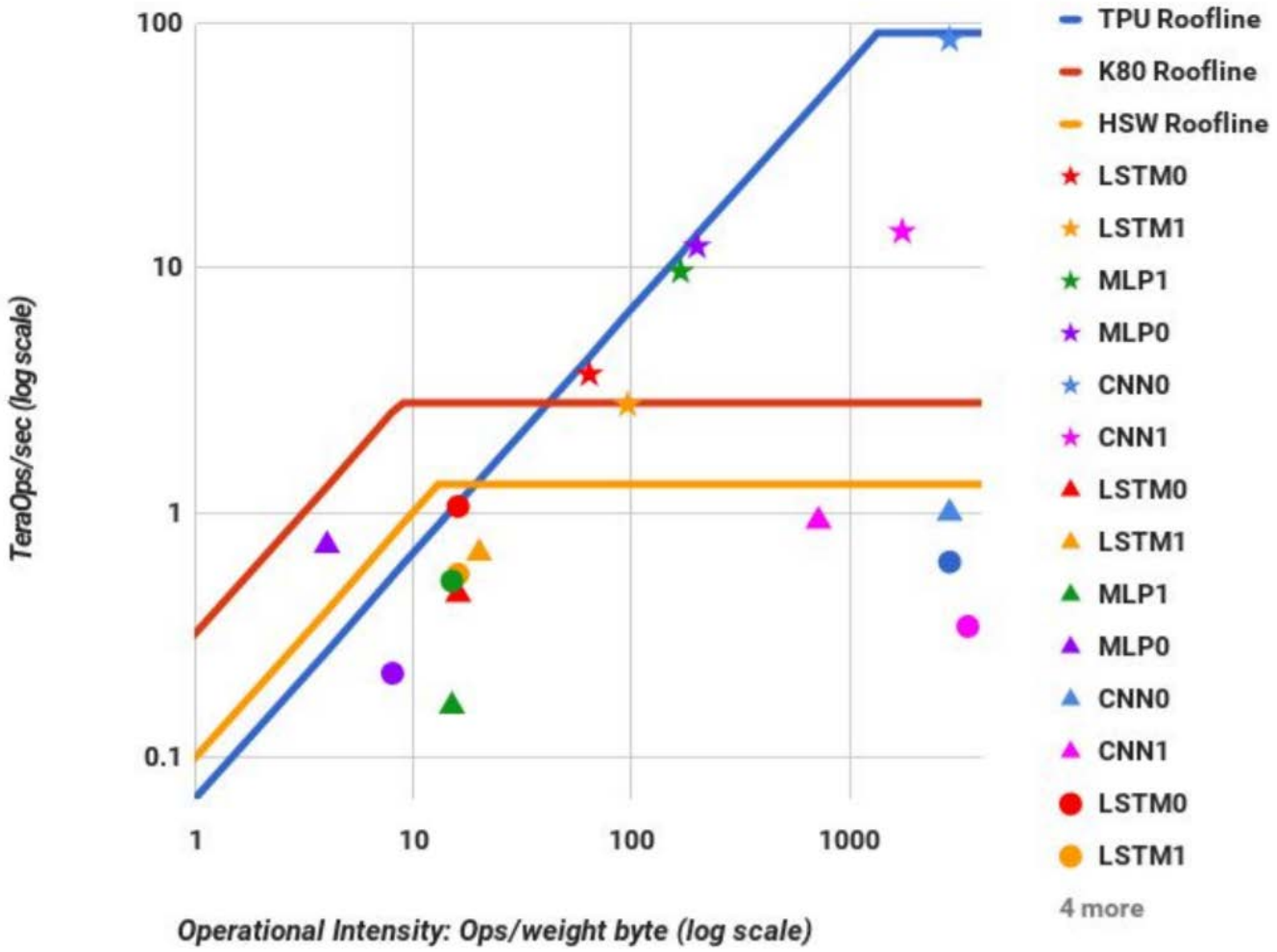
Part 1/4: Computation Specialization

TPU [Jouppi, 2017]



Roof line model

TPU [Jouppi, 2017]



TPU

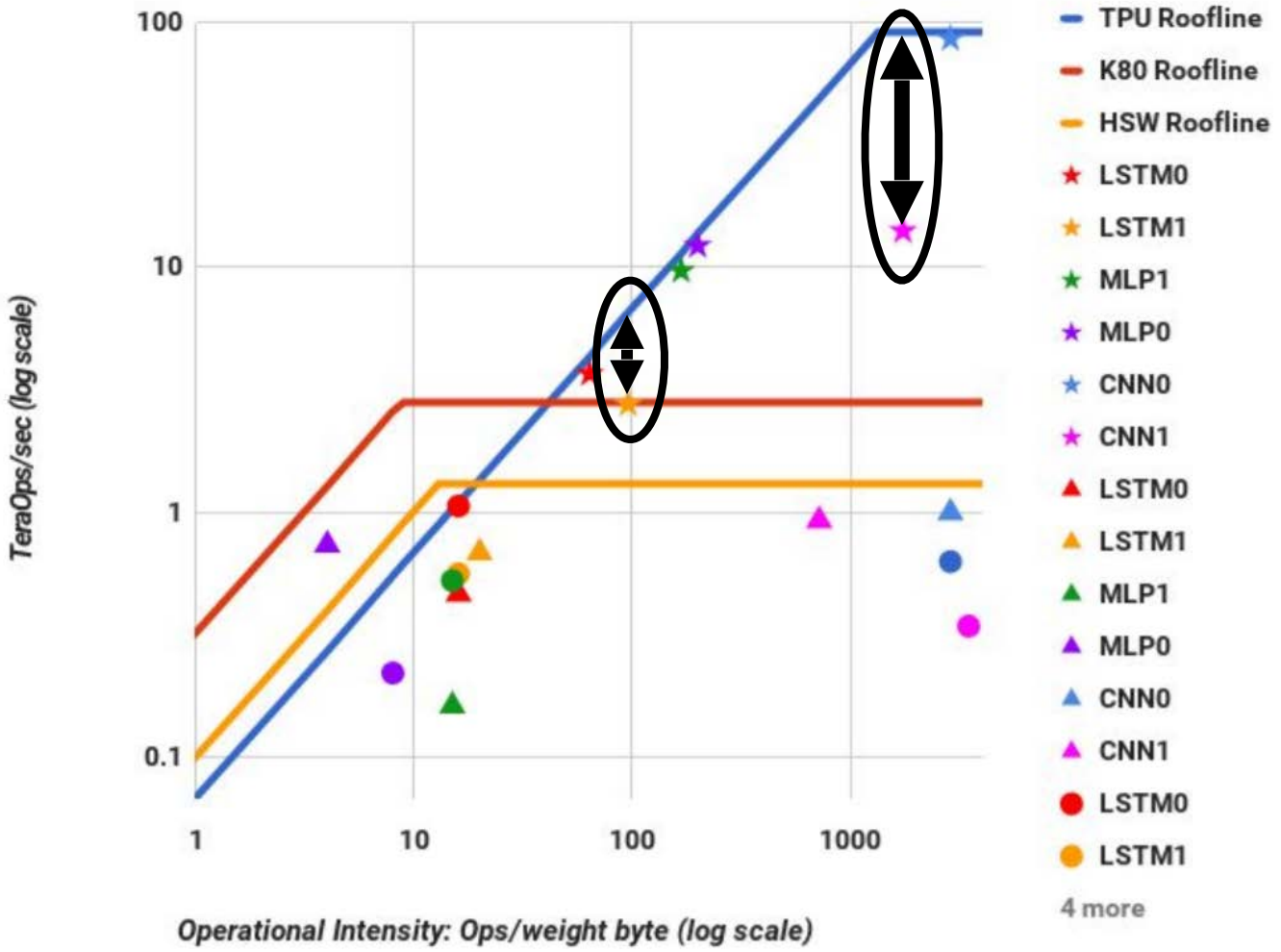
GPU K80

CPU Haswell

In-Datacenter Performance Analysis of a Tensor Processing Unit

Gap between real FLOP and peak FLOP: —— Memory Matters

TPU [Jouppi, 2017]



TPU

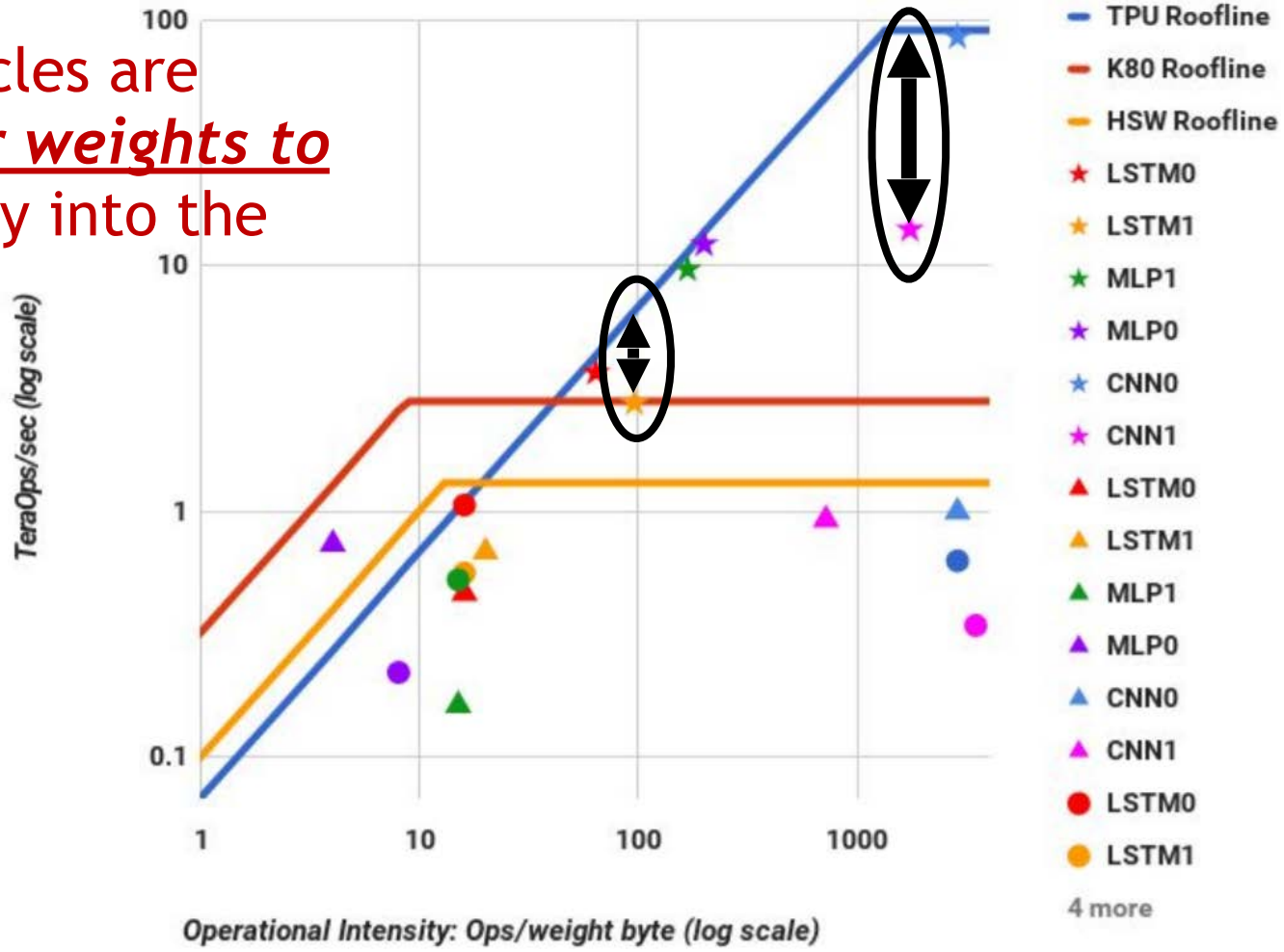
GPU K80

CPU Haswell

Gap between real FLOP and peak FLOP: —— Memory Matters

TPU [Jouppi, 2017]

"About 35% of cycles are spent ***waiting for weights to load*** from memory into the matrix unit …"

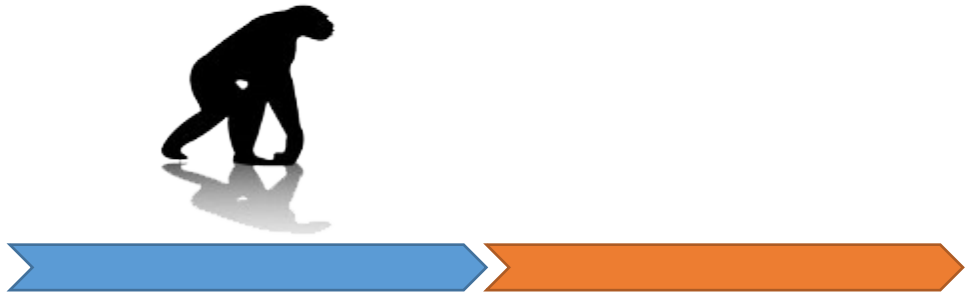


TPU

GPU K80

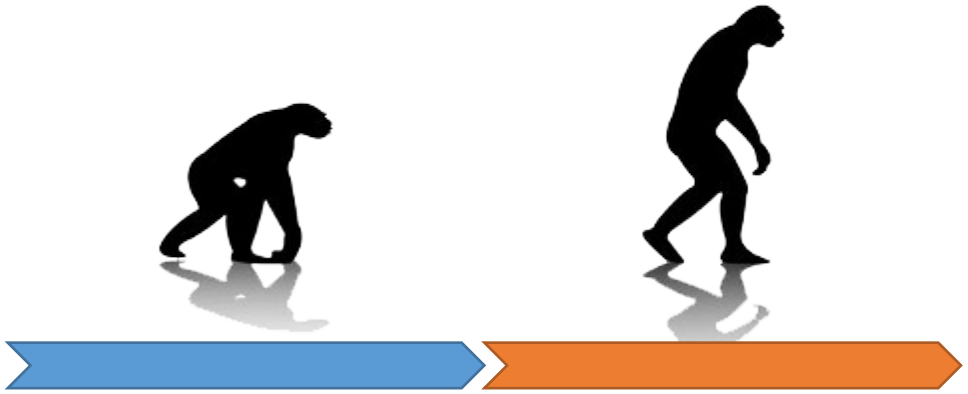
CPU Haswell

Evolution of NN Accelerators



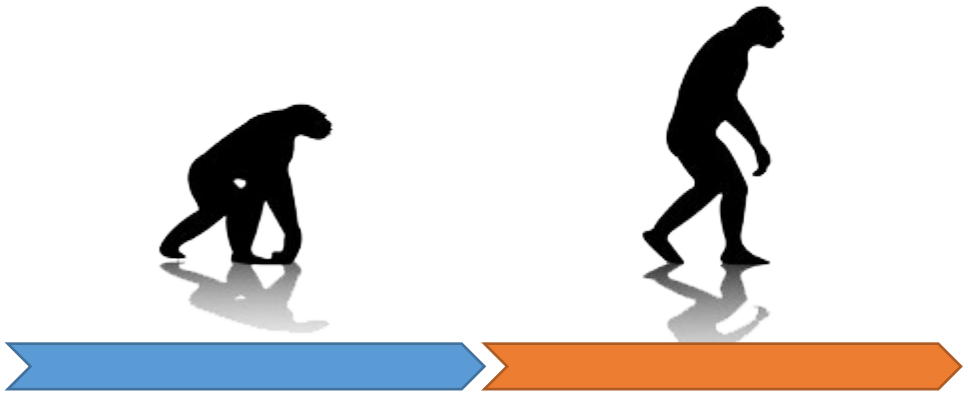
Computation Specialization

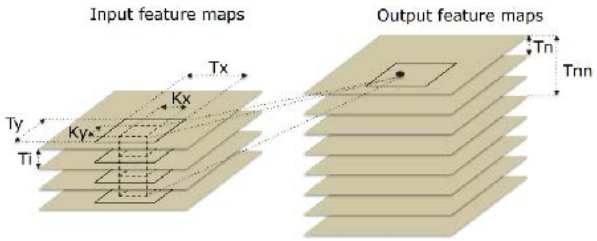
Evolution of NN Accelerators



Memory-centric Specialization

Part 2/4: Memory-centric Specialization



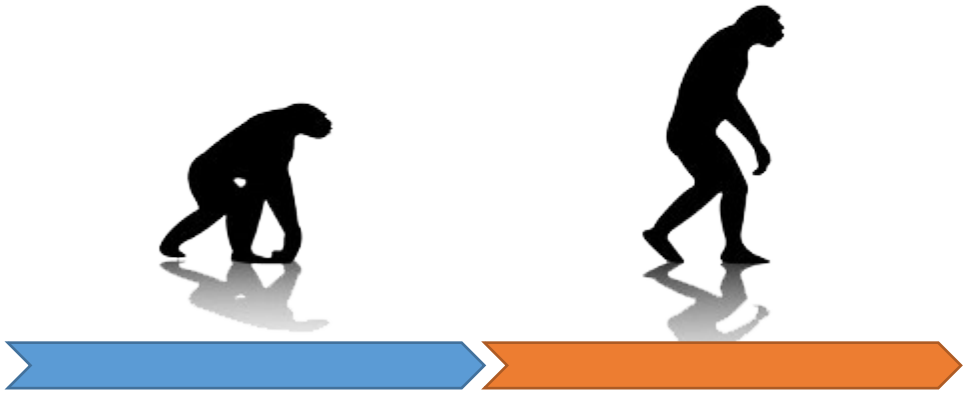


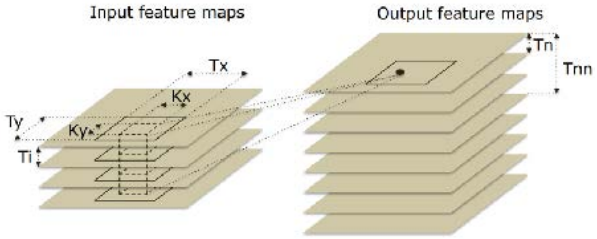
More Tiling

Memory-centric Specialization

- Diannao Family [Chen, 2014-2016]

Part 2/4: Memory-centric Specialization

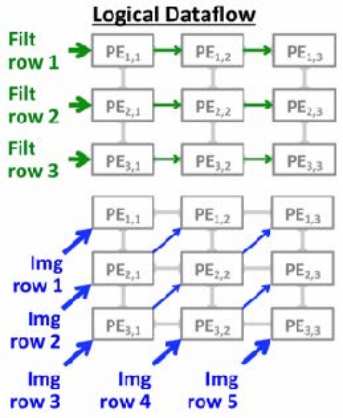




More Tiling

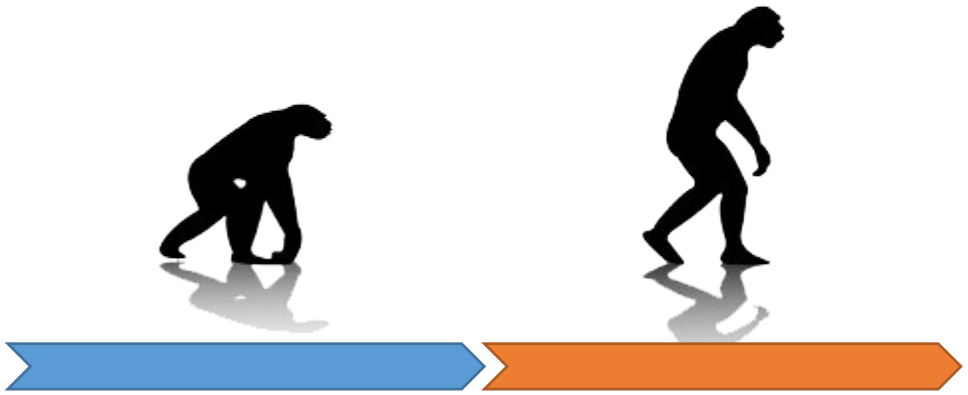
Memory-centric Specialization

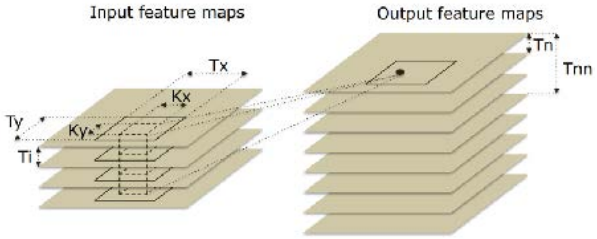
- Diannao Family [Chen, 2014-2016]
- Eyeriss [Chen, 2016]



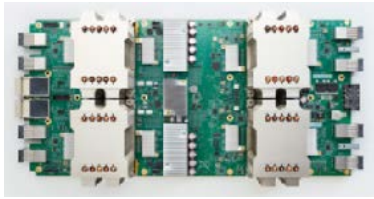
Better Dataflow

Part 2/4: Memory-centric Specialization





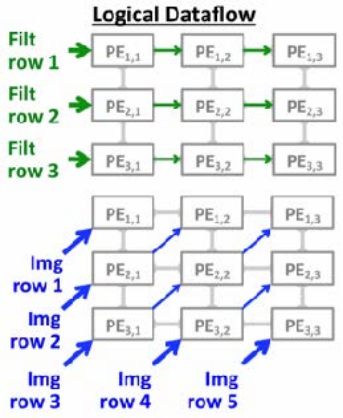
More Tiling



High Bandwidth Memory

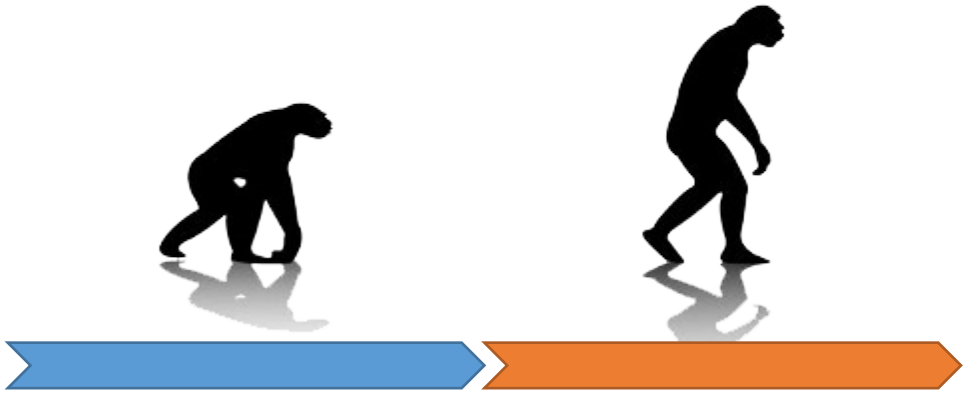
Memory-centric Specialization

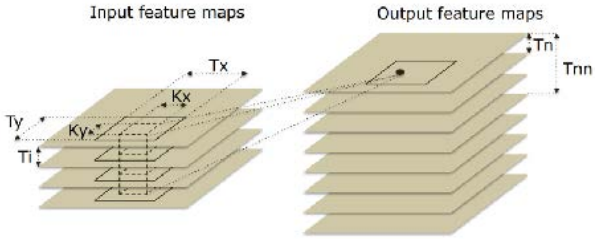
- Diannao Family [Chen, 2014-2016]
- Eyeriss [Chen, 2016]
- TPU v2 [Google, production 2017]



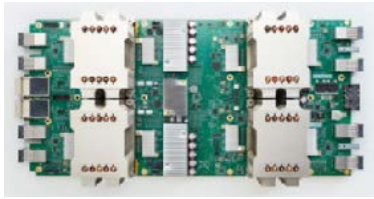
Better Dataflow

Part 2/4: Memory-centric Specialization





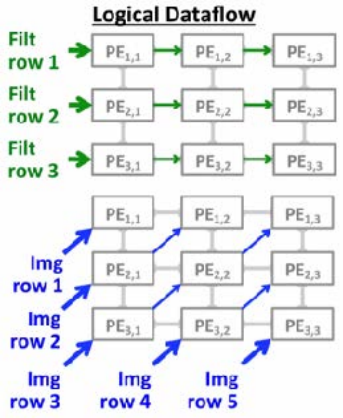
More Tiling



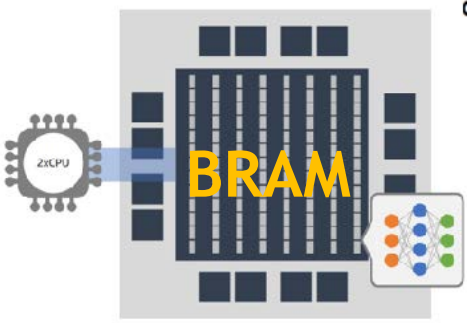
High Bandwidth Memory

Memory-centric Specialization

- Diannao Family [Chen, 2014-2016]
- Eyeriss [Chen, 2016]
- TPU v2 [Google, production 2017]
- Brainwave Project [Microsoft, 2017]



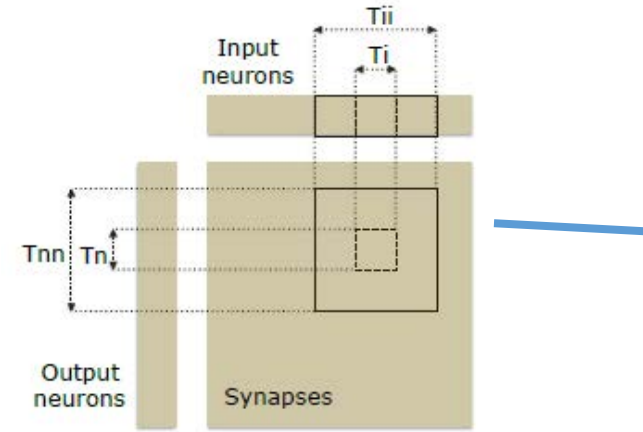
Better Dataflow

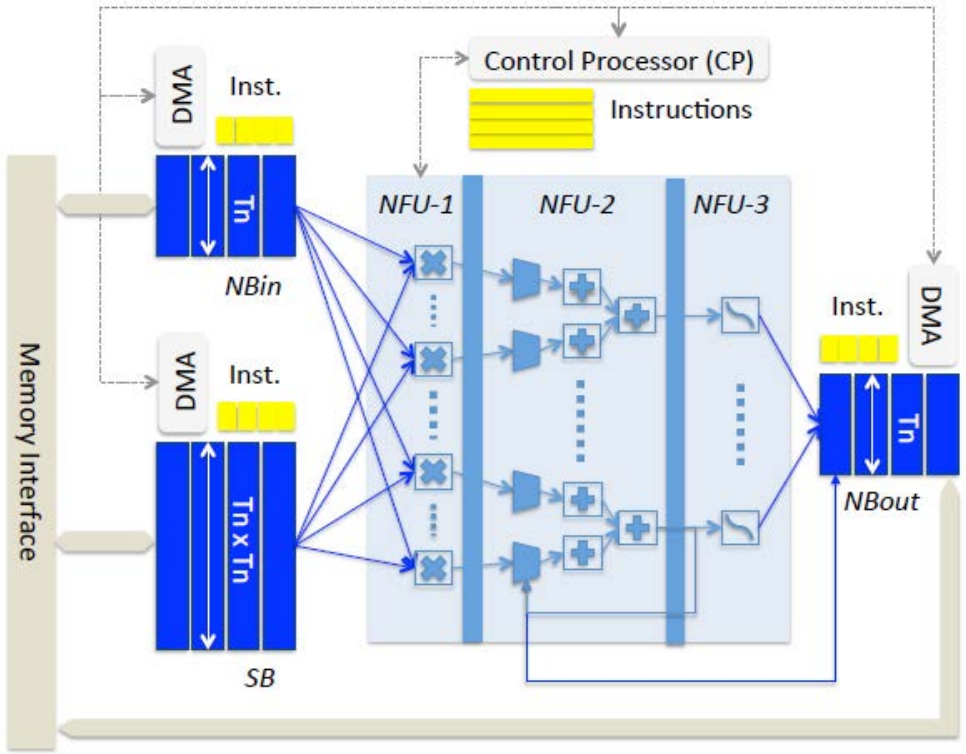


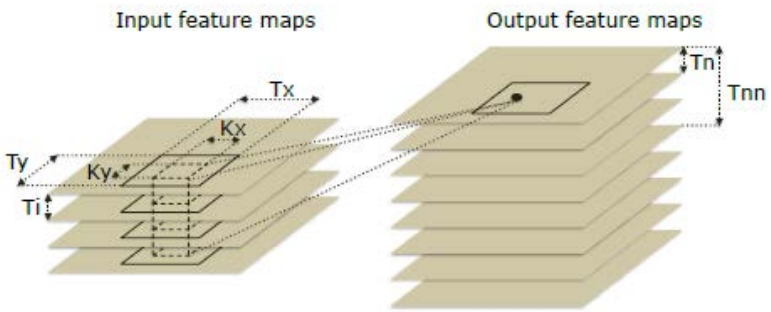
All On-Chip

Part 2/4: Memory-centric Specialization

Diannao Family [Chen, 2014-2016]



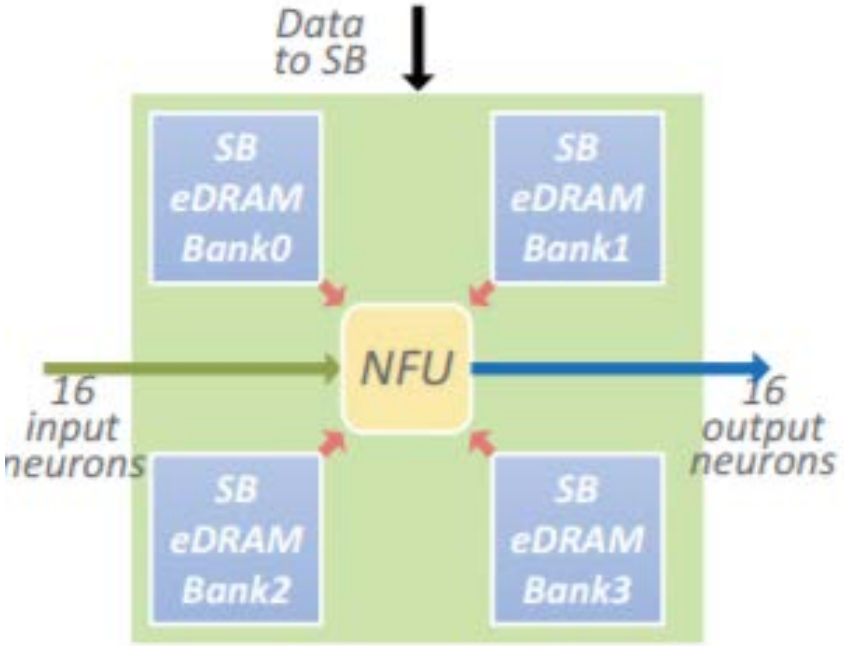


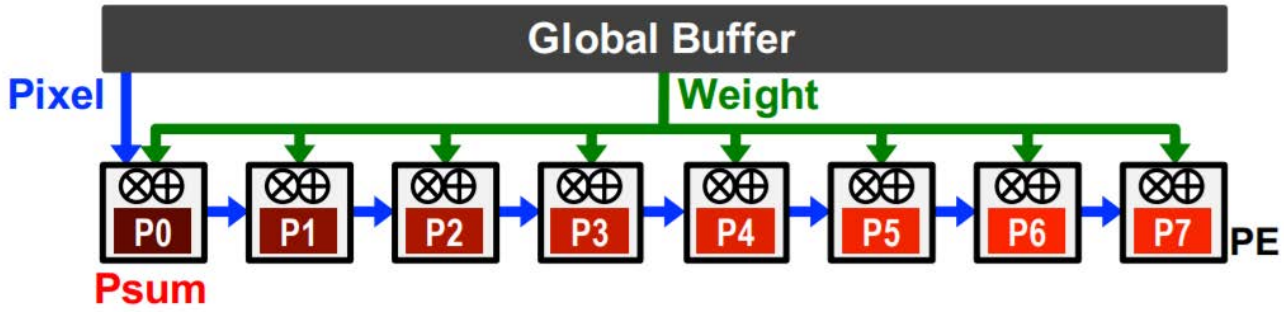


"The spirit of the NFU is to reflect the ***decomposition of a layer into computational blocks*** of inputs/synapses and output neurons …"

Part 2/4: Memory-centric Specialization

Diannao Family [Chen, 2014-2016]





Chen, Yunji, et al. "Dadiannao: A machine-learning supercomputer."
Du, Zidong, et al. "ShiDianNao: Shifting vision processing closer to the sensor."

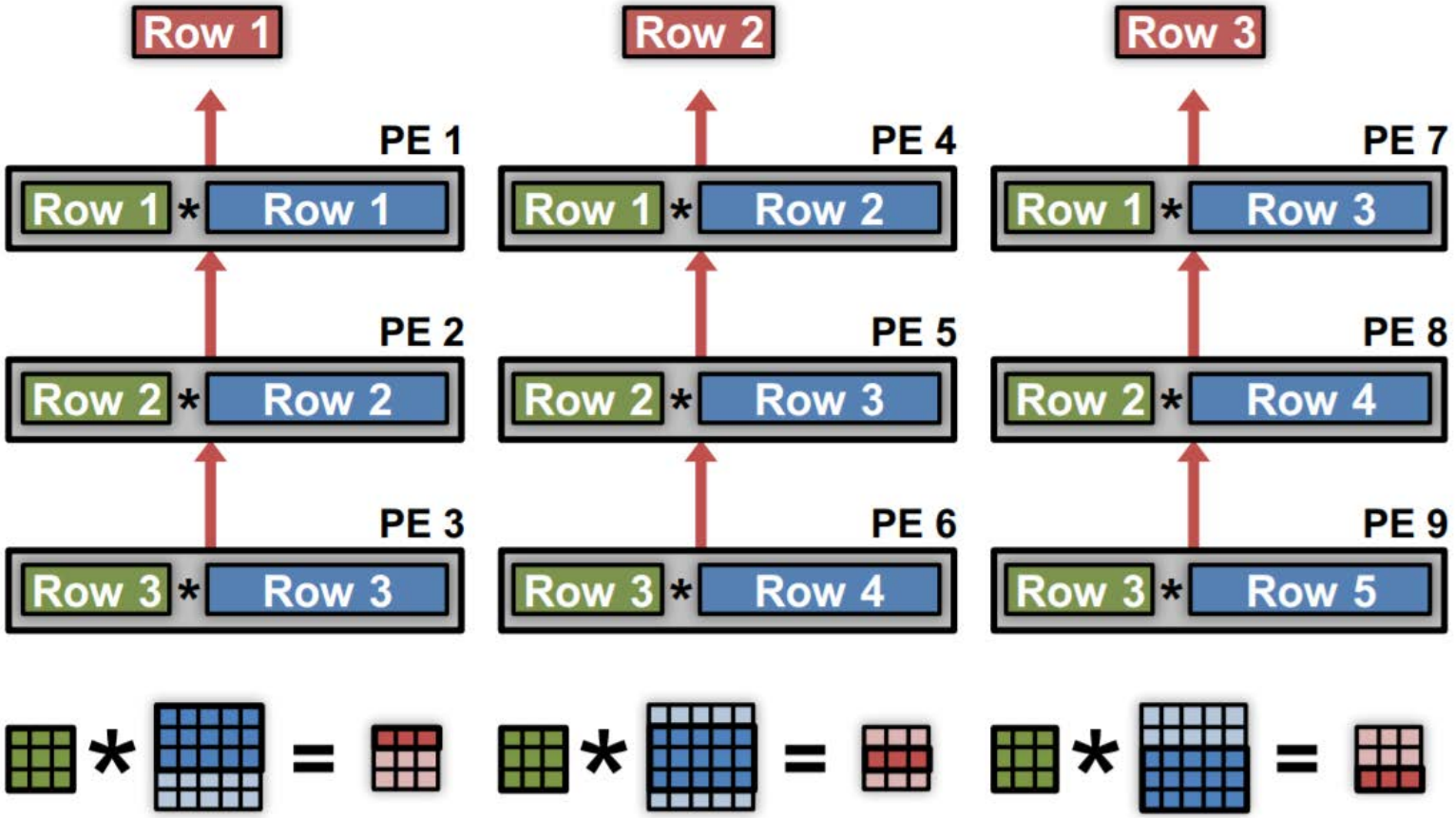
Part 2/4: Memory-centric Specialization

Diannao Family [Chen, 2014-2016]

	Process (nm)	Peak Performance (GOPs)	Peak Power (W)	Area (mm2)
DianNao	65	452	0.485	3.02
DaDianNao	28	5585	15.97	67.73
ShiDianNao	65	194	0.32	4.86

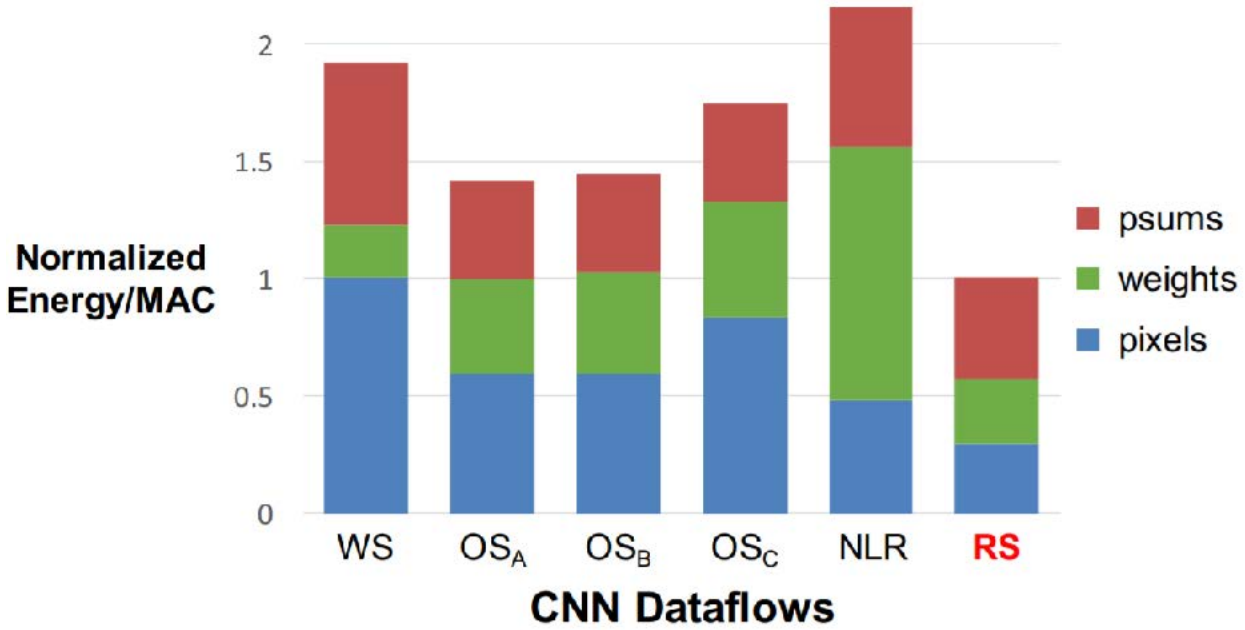
Part 2/4: Memory-centric Specialization

Eyeriss [Chen, 2016]



Part 2/4: Memory-centric Specialization

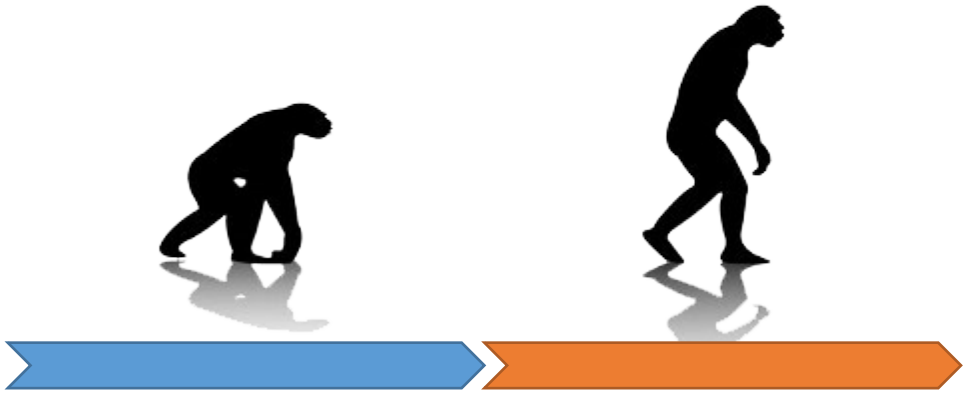
Eyeriss [Chen, 2016]



	AlexNet	VGG-16
Performance (GOPs)	46.2	21.4
Energy Efficiency (GOPs/W)	166.2	90.7

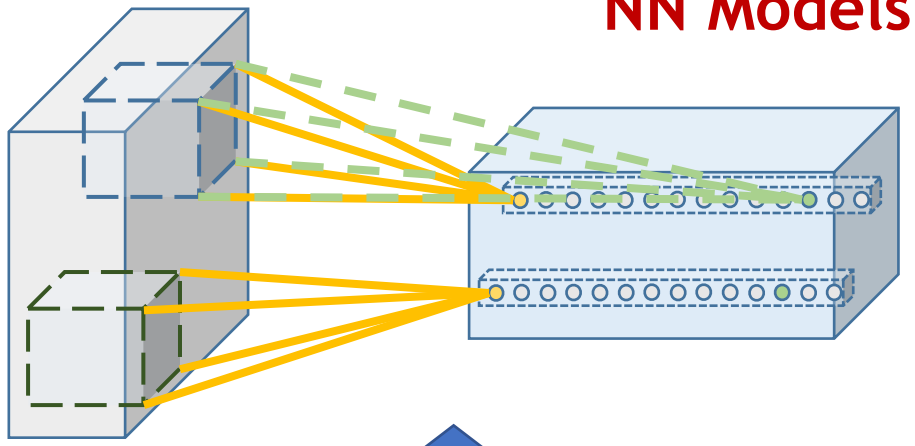
Yu-Hsin Chen, Tushar Krishna, Joel Emmer, Vivienne Sze.
"Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks."

Evolution of NN Accelerators

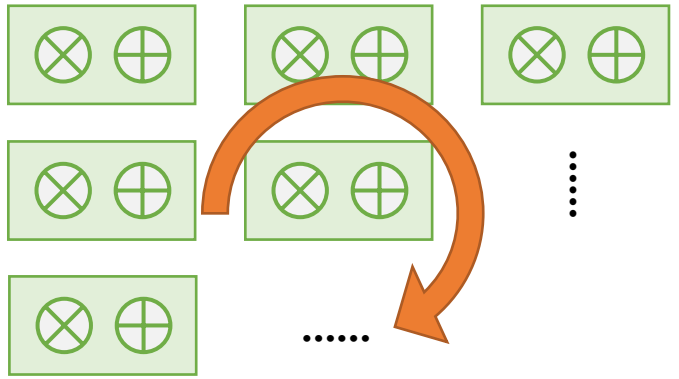


Memory-centric Specialization

NN Models

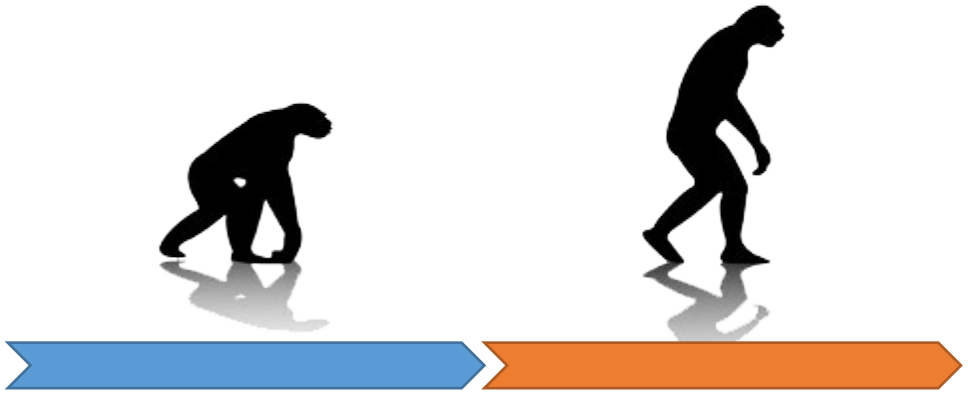


Bottom Up

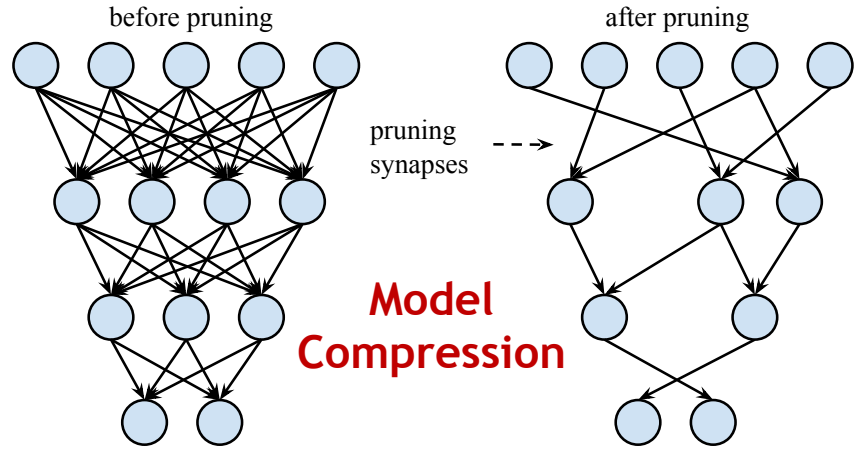


Hardware

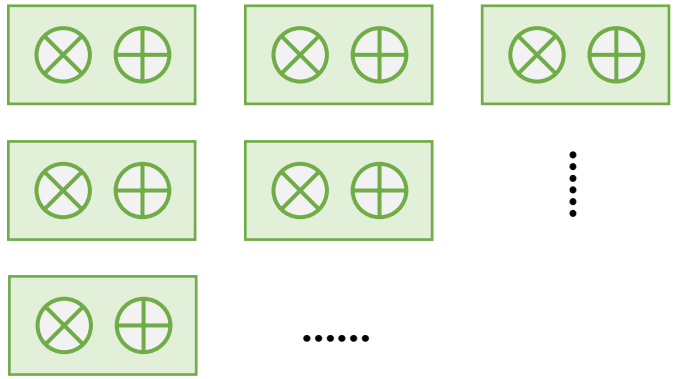
Evolution of NN Accelerators



Memory-centric Specialization

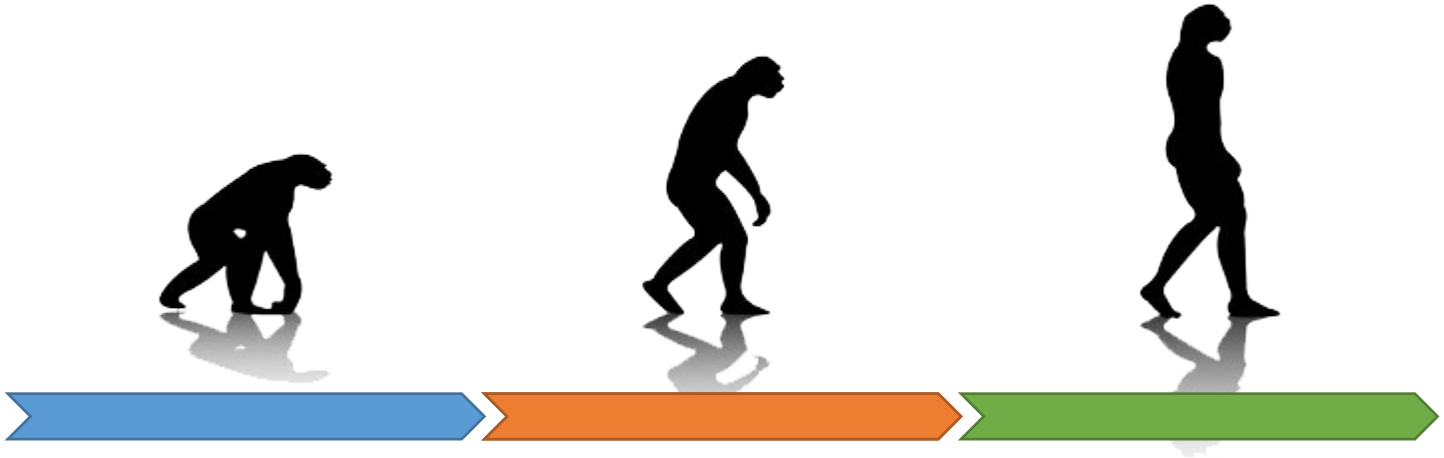


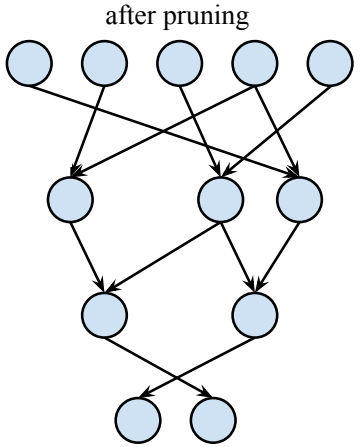
Top Down



Hardware

Evolution of NN Accelerators

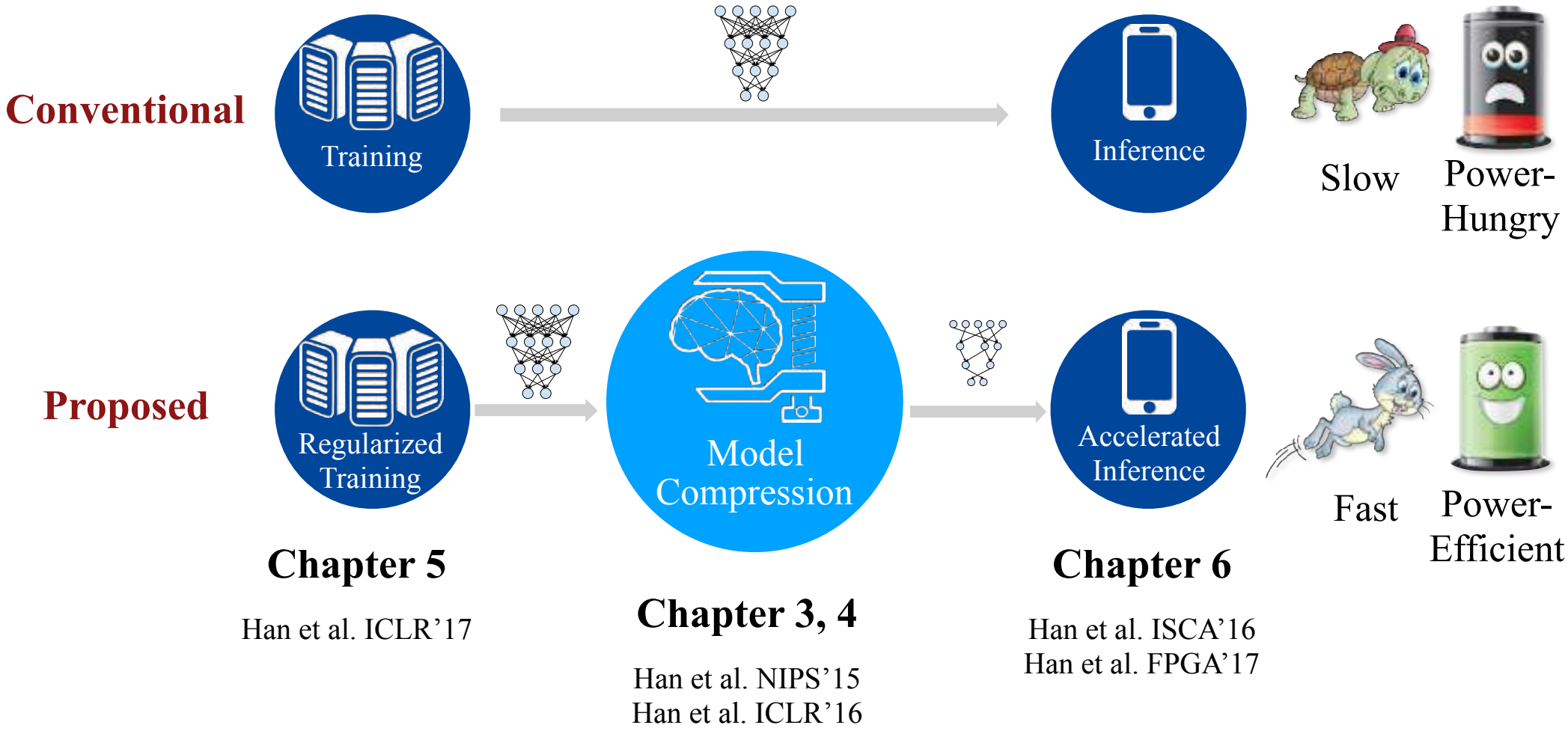




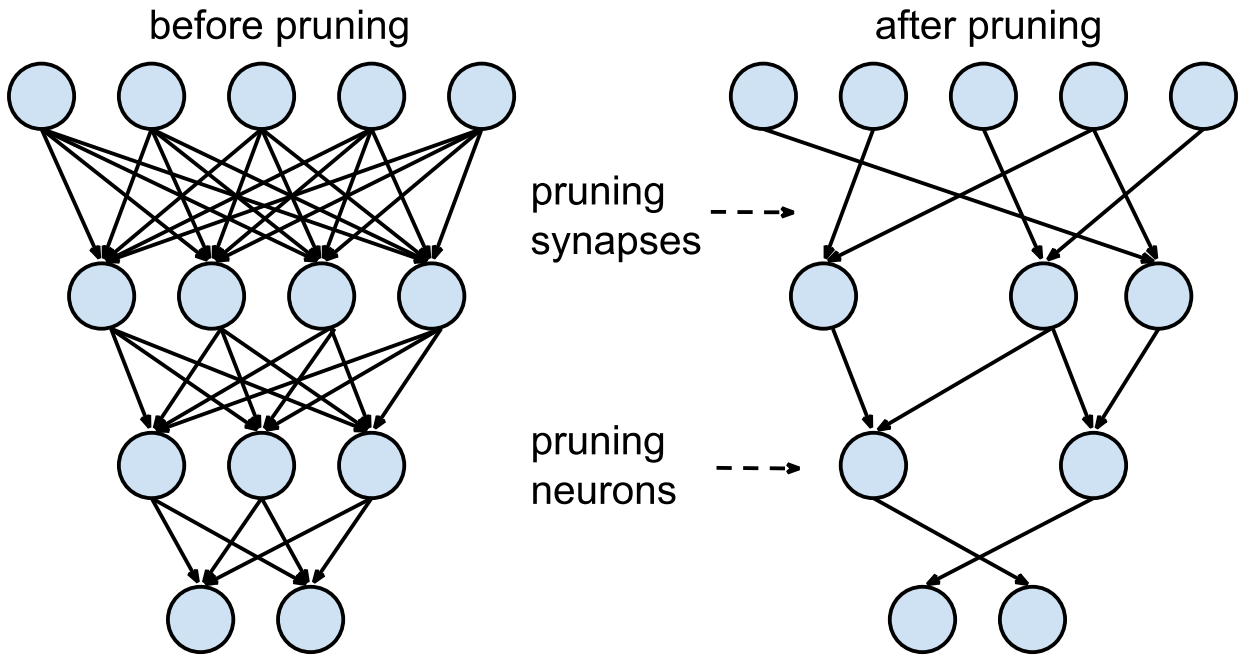
Compression
before Acceleration

In order to understand the 3rd generation of NN accelerator let's switch gear to talk about deep model compression

Compression before Acceleration

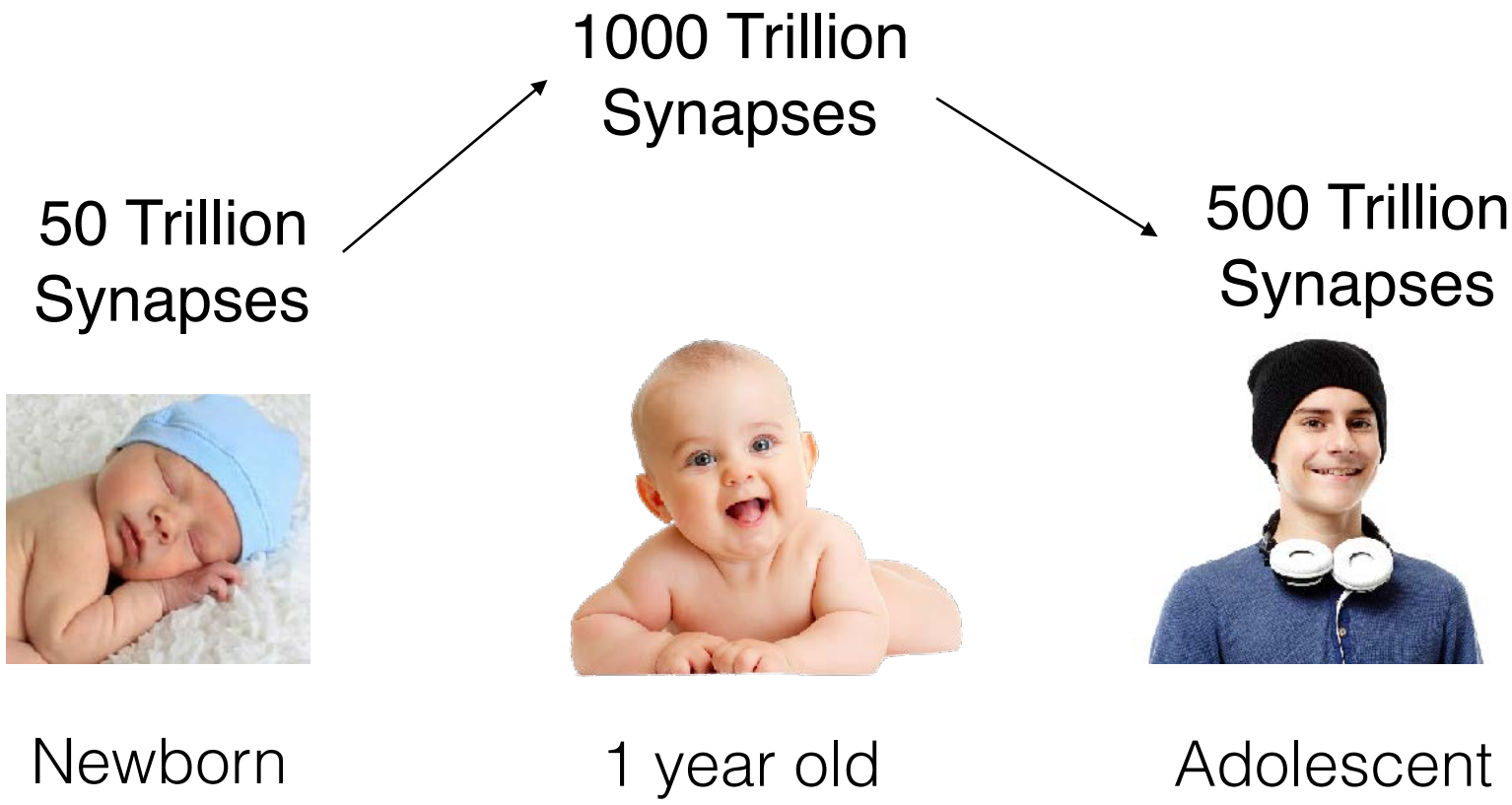


Deep Compression Part1: Pruning



Song Han, Jeff Pool, John Tran, Bill Dally, "Learning both Weights and Connections for Efficient Neural Networks" NIPS'15

Pruning Happens in Human Brain



Christopher A Walsh. Peter Huttenlocher (1931-2013). Nature, 502(7470):172–172, 2013.

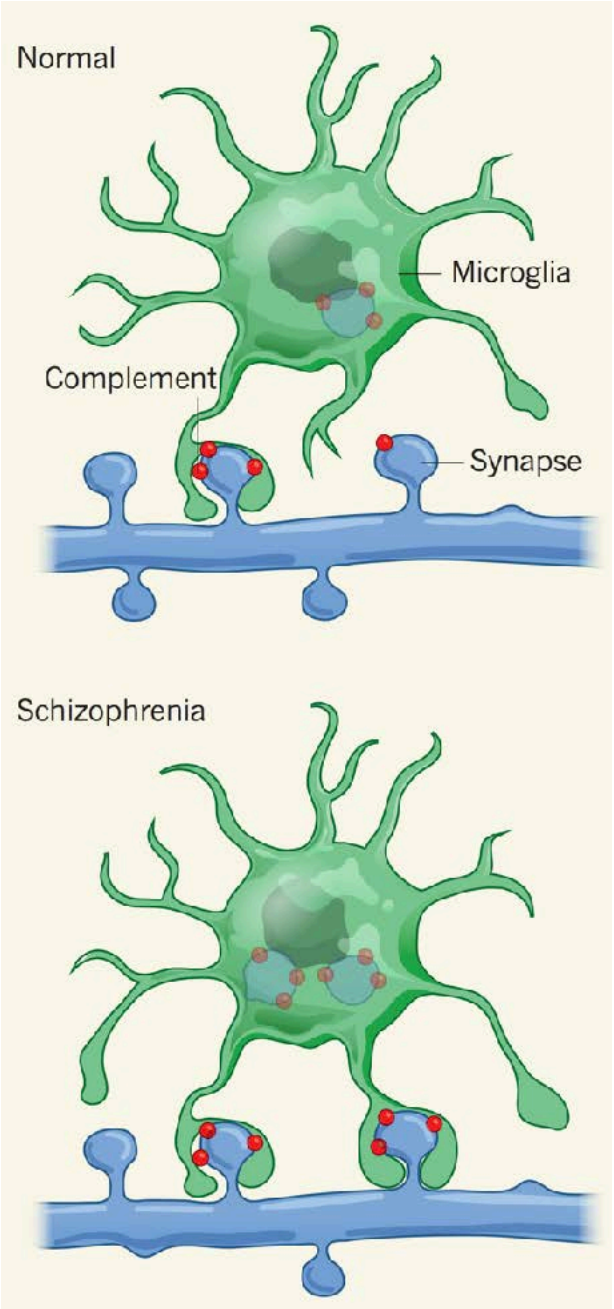


Figure: A growing understanding of brain rewiring in adolescence.

In Retrospect

Pruning hypothesis comes of age

The idea that disrupted pruning of neuronal connections in the brain during adolescence is a cause of schizophrenia was proposed in 1983. This proved prescient, as subsequent imaging, genetic and molecular research has shown.

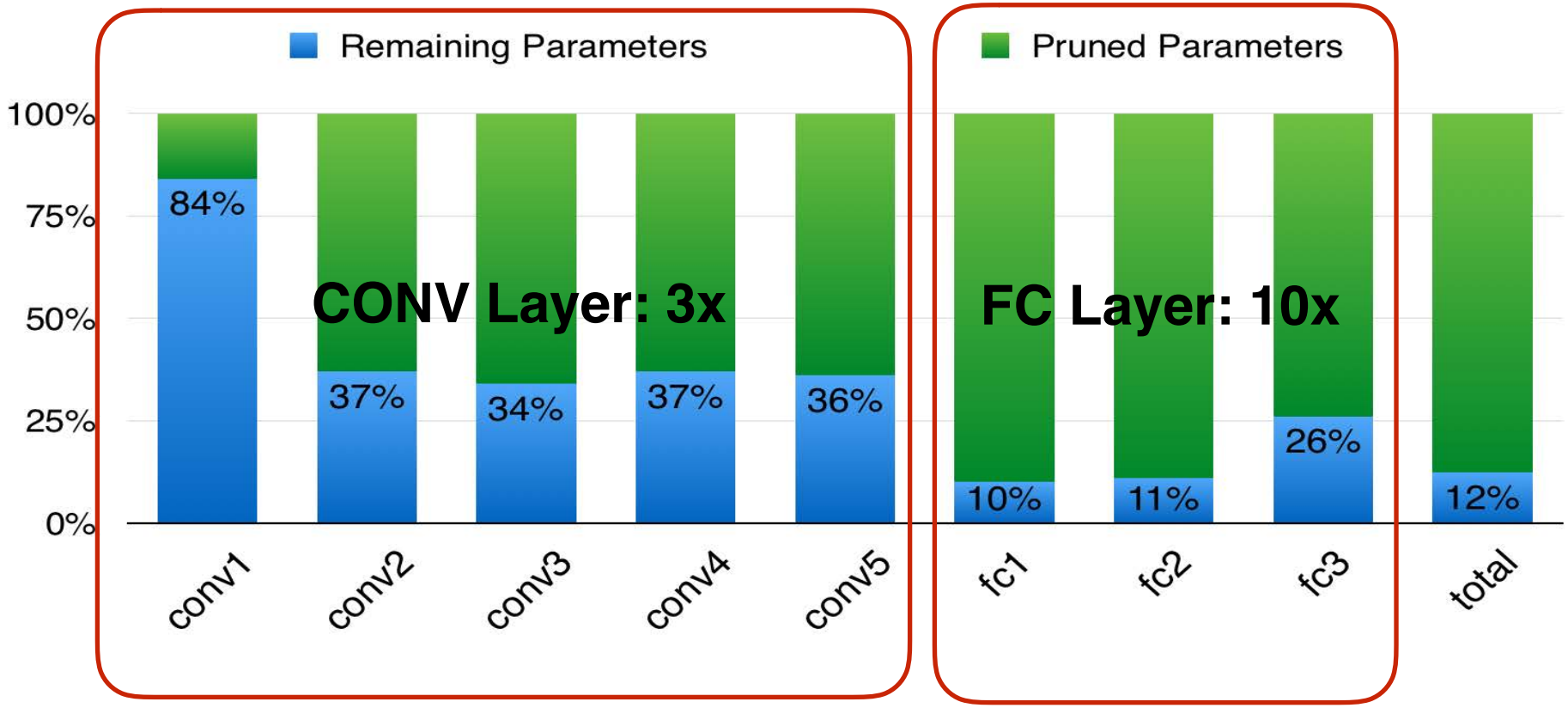
In 1979, an analysis of synapse numbers in the frontal lobe at different ages revealed **a dramatic decrease in synapses during adolescence**. Sleep brainwave amplitudes follow a similar pattern. Furthermore, symptoms of schizophrenia typically emerge in adolescence. In 1983, these facts together led the psychiatrist Irwin Feinberg, who was studying sleep, to propose that **defects in adolescent pruning of synapses might be a cause of schizophrenia**.

/skitsəˈfrēnēə/

Schizophrenia: a long-term **mental disorder** of a type involving a breakdown in the relation between thought, emotion, and behavior, leading to **faulty perception** and inappropriate actions.

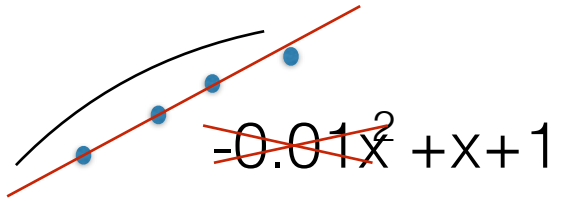
https://www.nature.com/magazine-assets/d41586-018-02053-7/d41586-018-02053-7.pdf

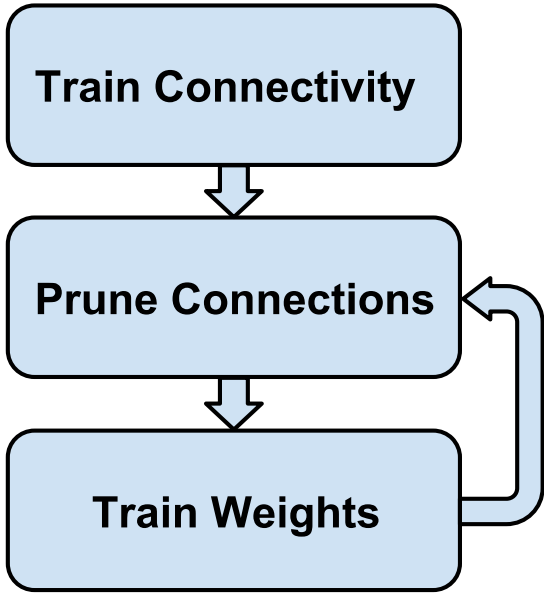
Pruning AlexNet



Song Han, Jeff Pool, John Tran, Bill Dally, "Learning both Weights and Connections for Efficient Neural Networks" NIPS'15

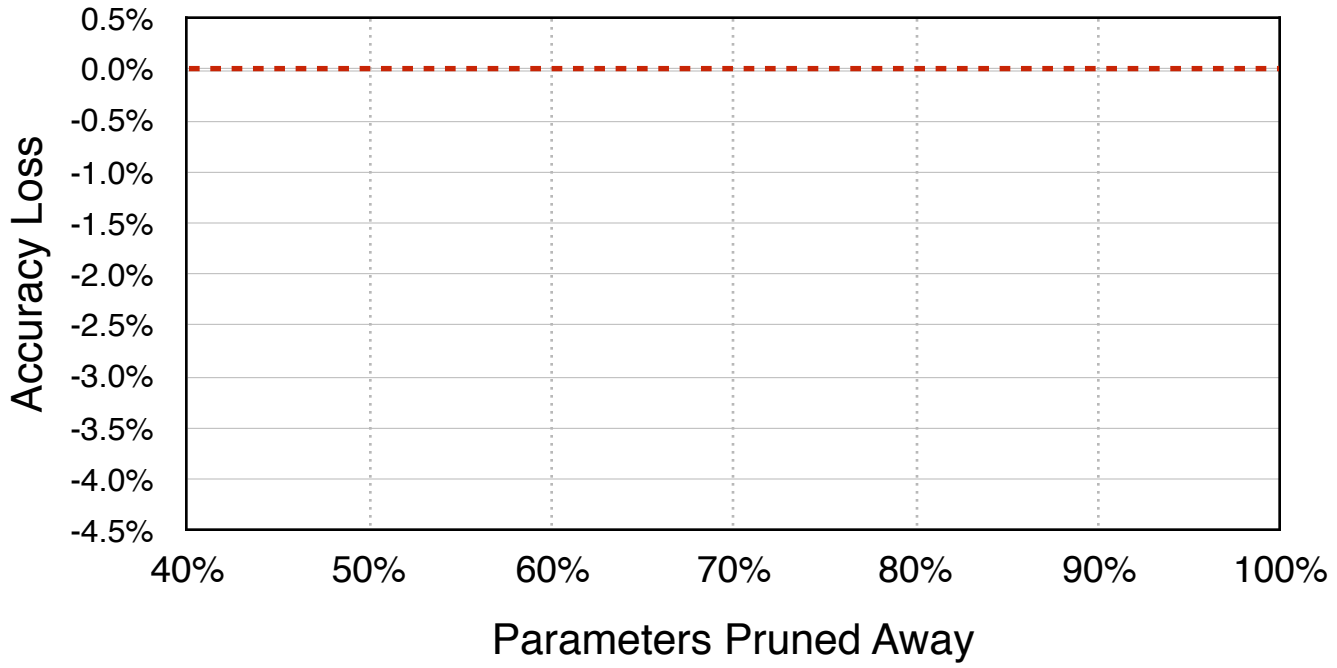
Pruning Neural Networks



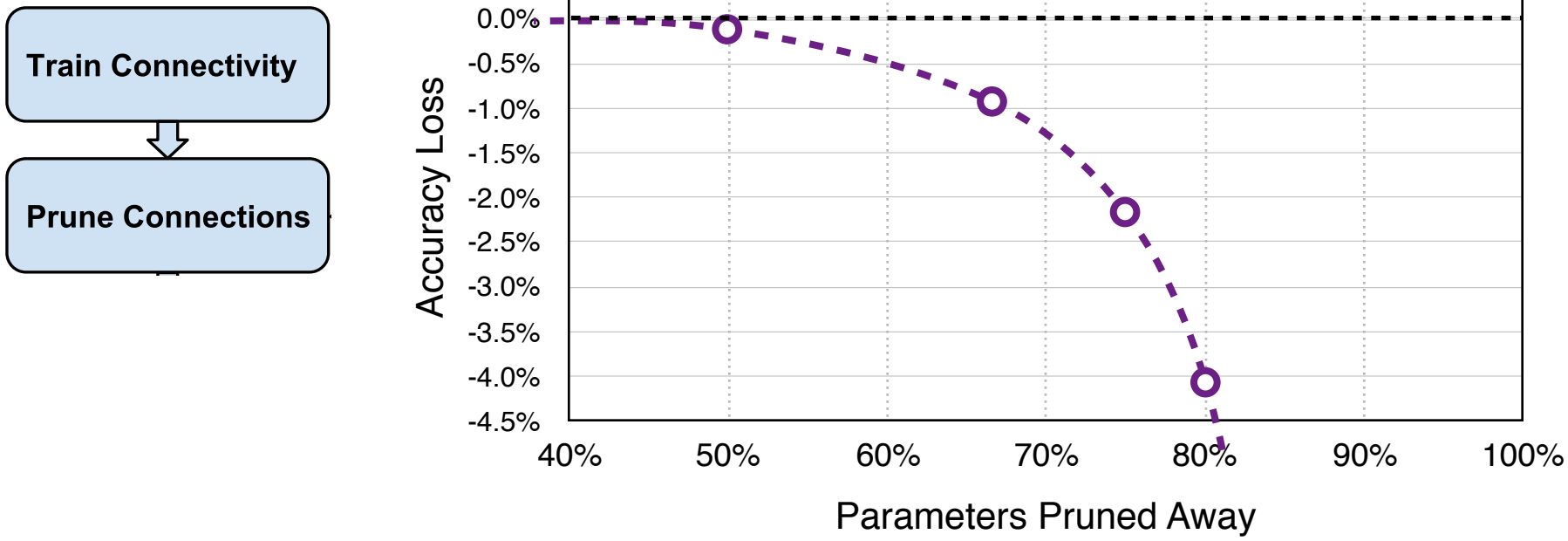


Pruning Neural Networks

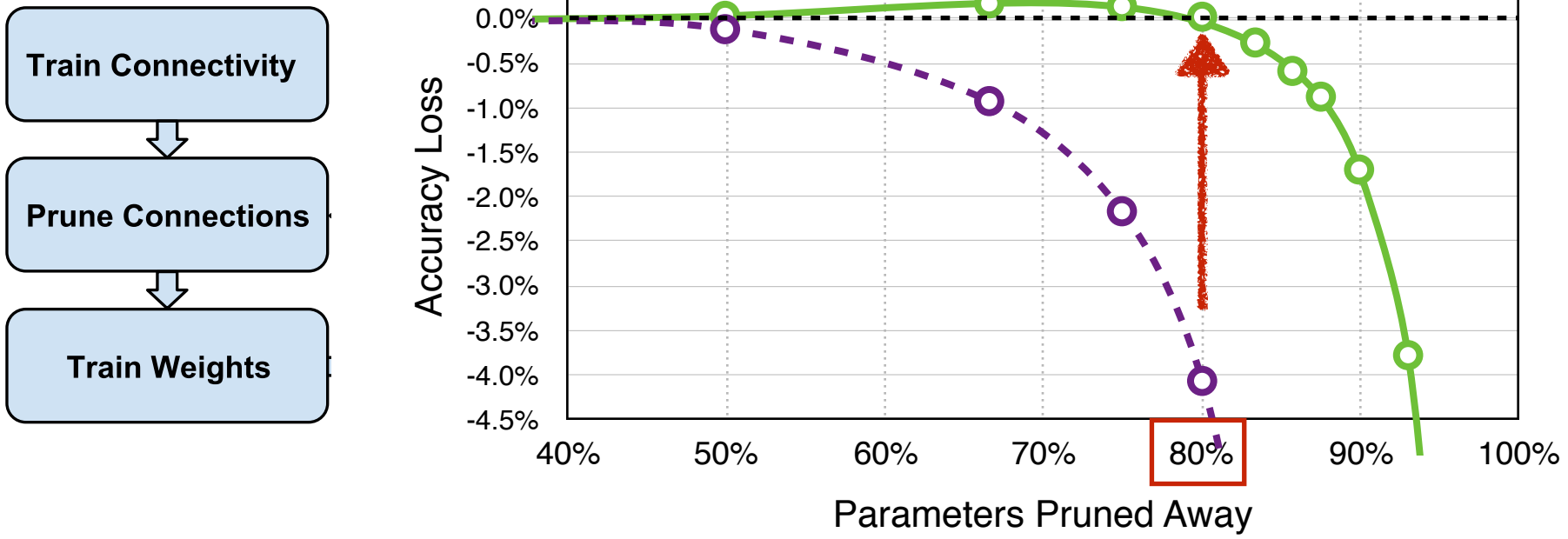
Train Connectivity



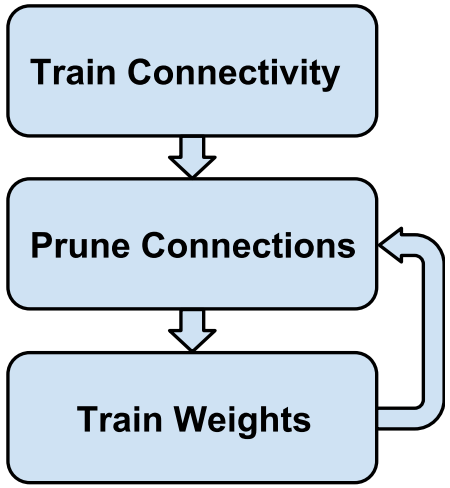
Pruning Neural Networks

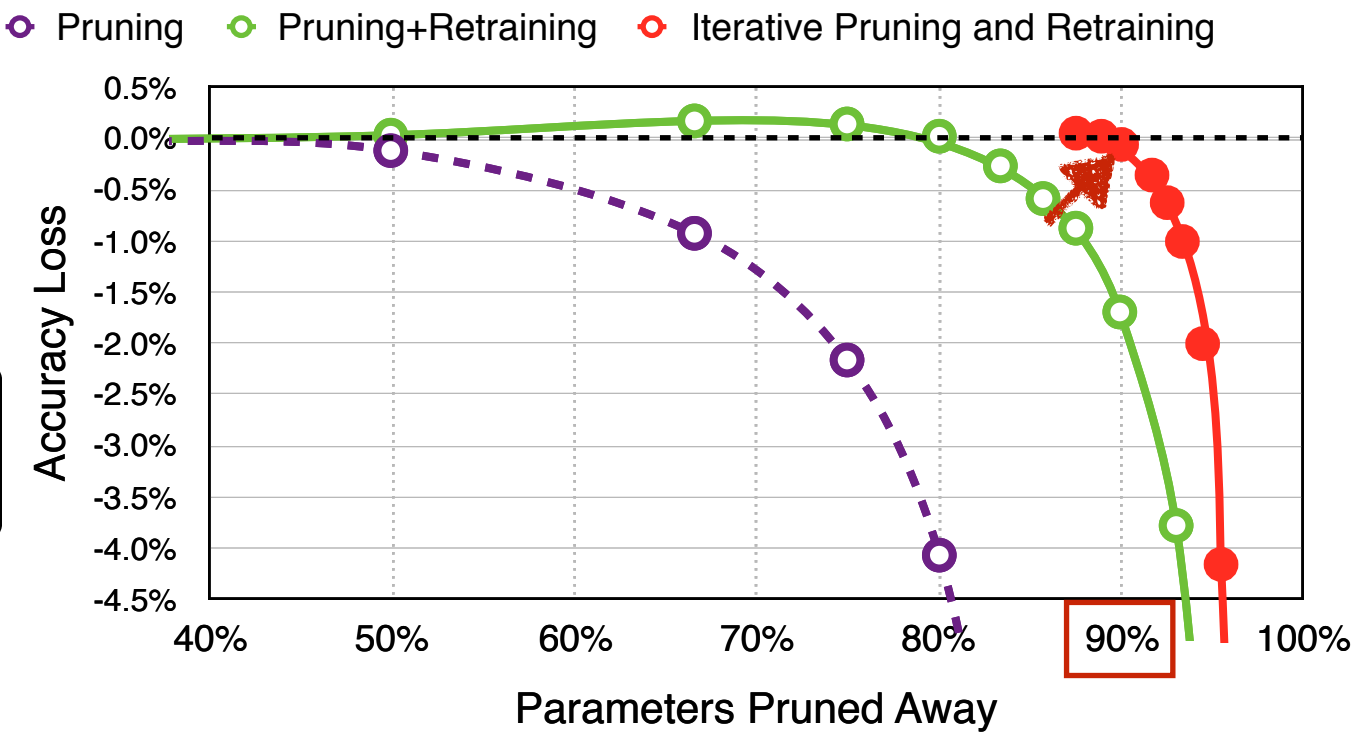


Retrain to Recover Accuracy

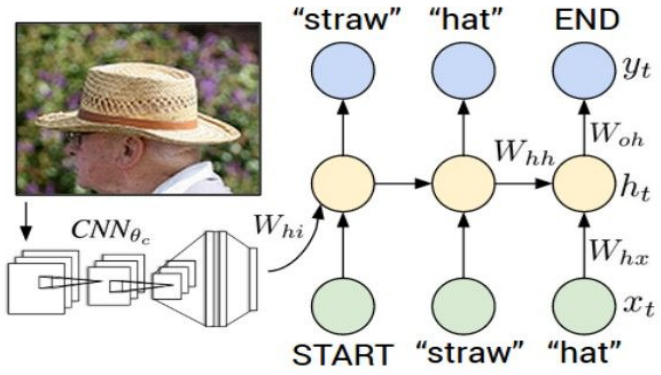


Iteratively Retrain to Recover Accuracy

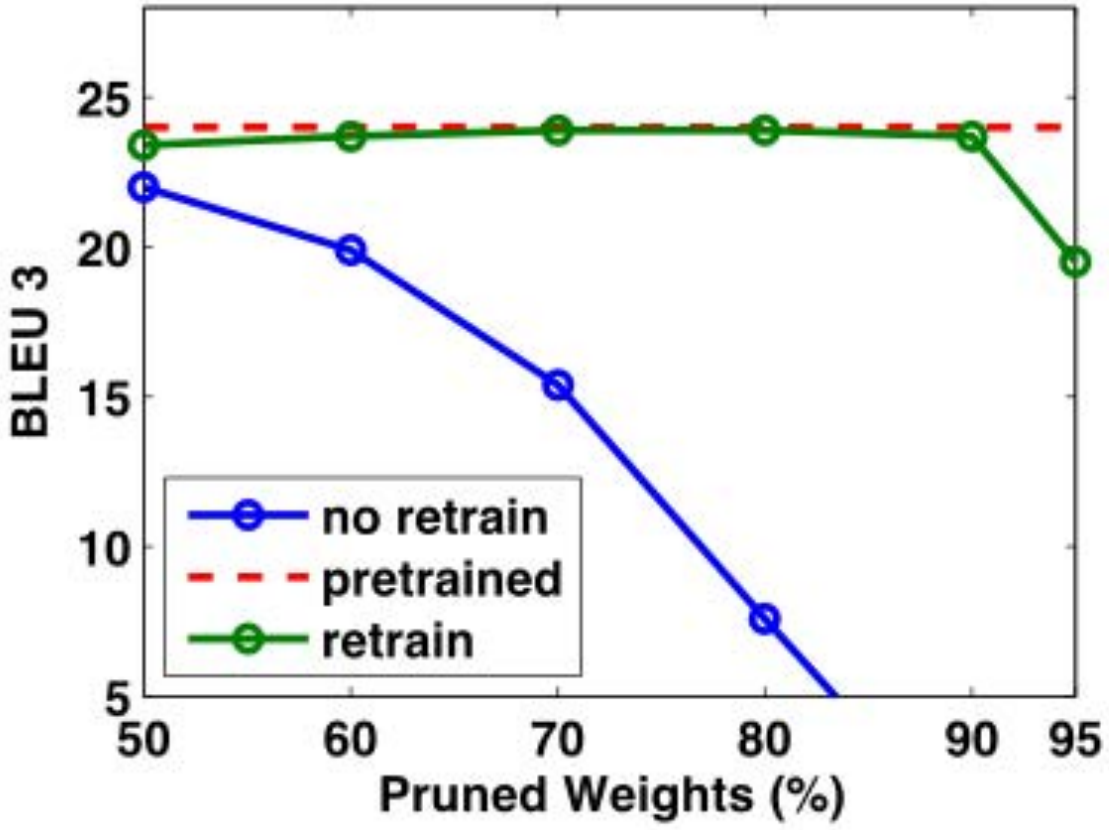




Pruning RNN and LSTM



*Karpathy et al "Deep Visual-Semantic Alignments for Generating Image Descriptions"



Pruning Tool is Available at DeePhi

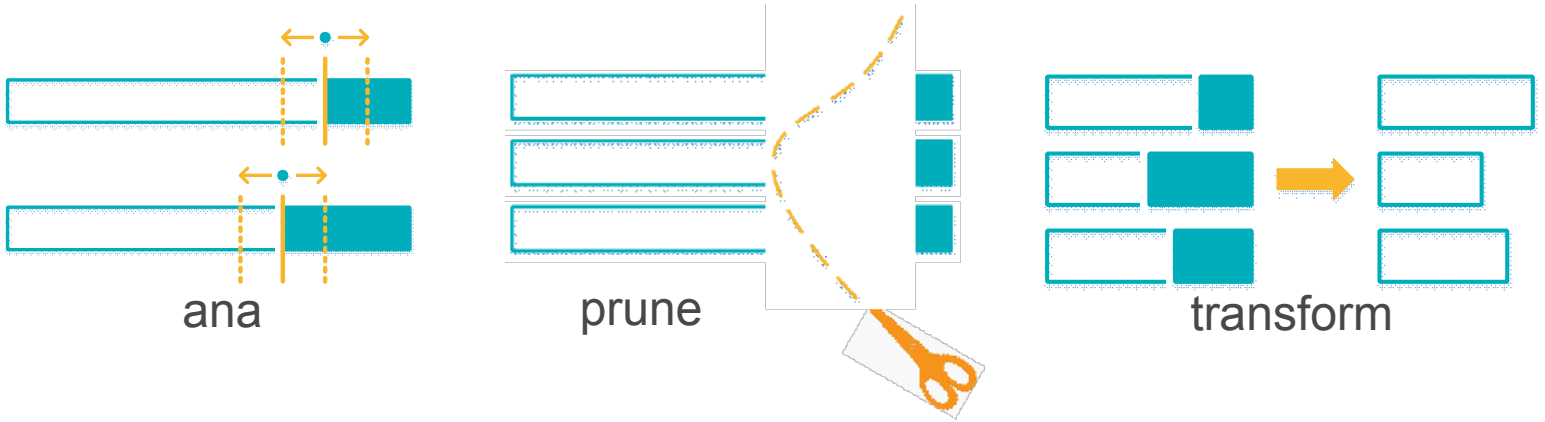
Deep Neural Network Development Kit (DNNDK)

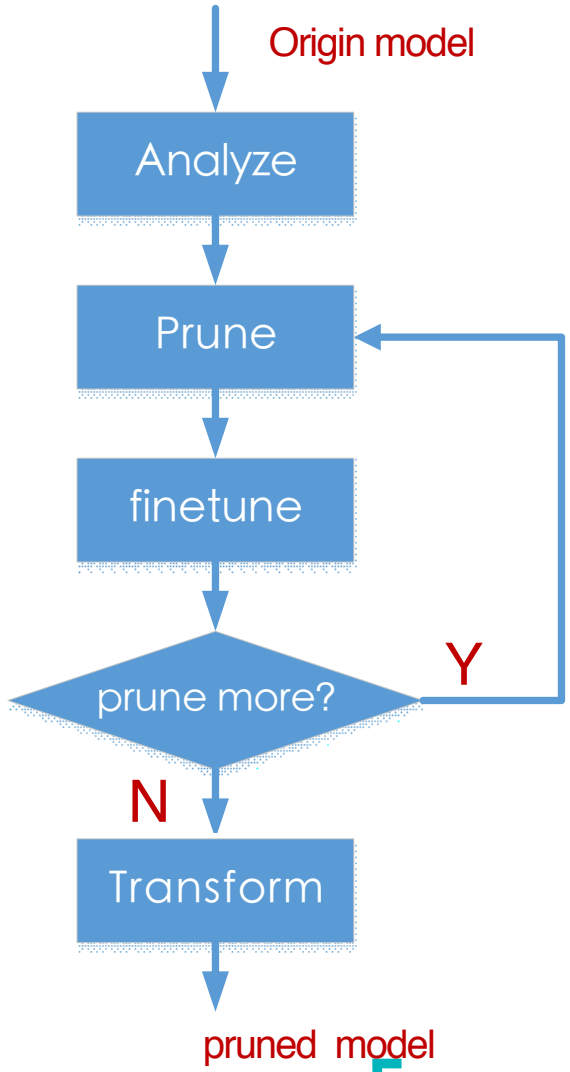
http://www.deephi.com/dnndk.html

DEEPHi 深鉴科技 Now Part of Xilinx

DeePhi's Pruning Tool

- 4 commands in decent_p
  - ana – analyze the network
  - prune – prune the network according to config
  - finetune – finetune the network to recovery accuracy
  - transform – transform the pruned model to regular model





DeePhi's Pruning Tool

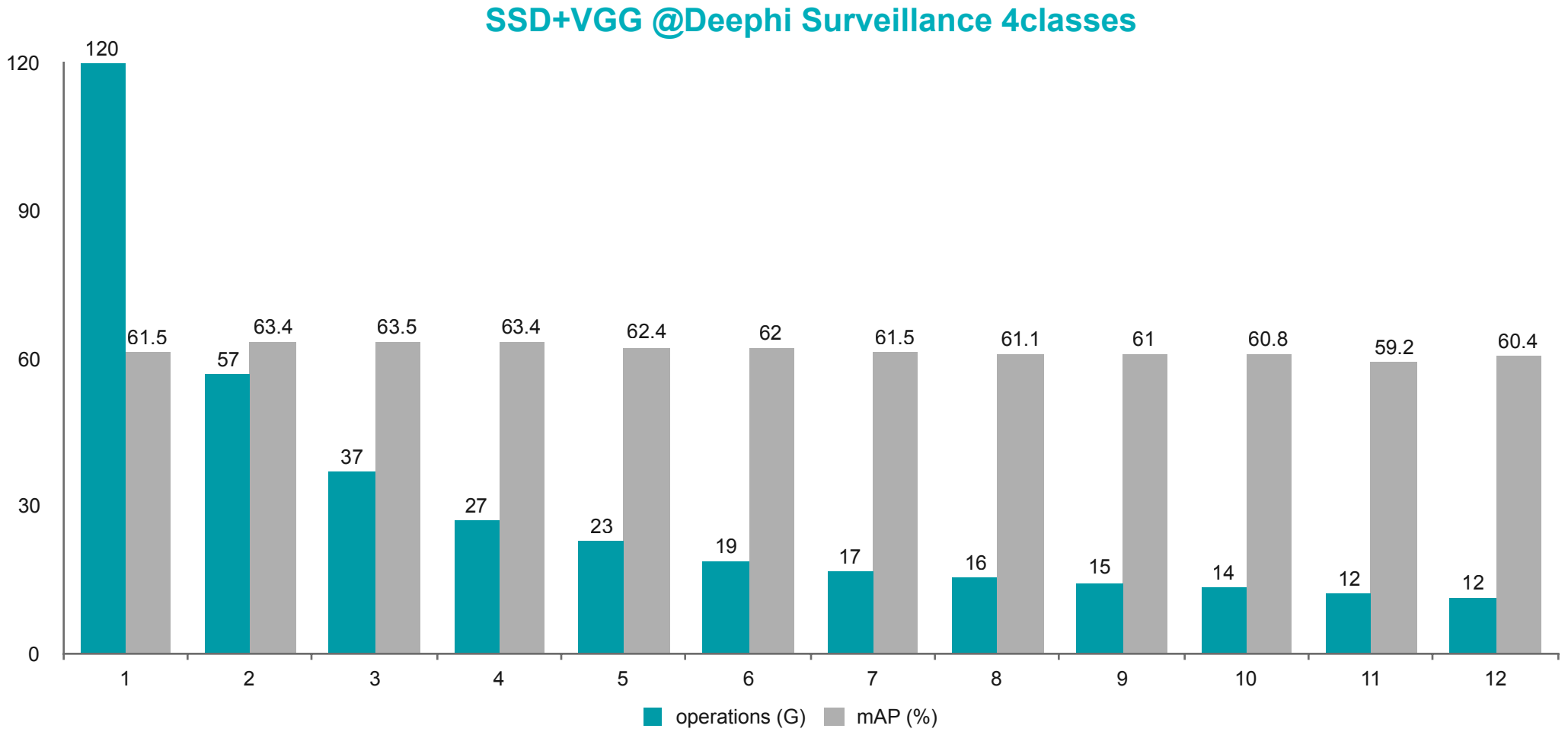
Classification Networks	Baseline	Pruning Result 1			Pruning Result 2		
	Top-5	Top-5	ΔTop5	ratio	Top-5	ΔTop5	ratio
Resnet50 [7.7G]	91.65%	91.23%	-0.42%	40%	90.79%	-0.86%	32%
Inception_v2 [4.0G]	91.07%	90.37%	-0.70%	60%	90.07%	-1.00%	55%
SqueezeNet [778M]	83.19%	82.46%	-0.73%	89%	81.57%	-1.62%	75%

Detection Networks	Baseline mAP	Pruning Result 1			Pruning Result 2		
		mAP	ΔmAP	ratio	mAP	ΔmAP	ratio
DetectNet [17.5G]	44.46	45.7	+1.24	63%	45.12	+0.66	50%
SSD+VGG [117G]	61.5	62.0	+0.5	16%	60.4	-1.1	10%
[A] SSD+VGG [173G]	57.1	58.7	+1.6	40%	56.6	-0.5	12%
[B] Yolov2 [198G]	80.4	81.9	+1.5	28%	79.2	-1.2	7%

Deep Compression Tool (DECENT)

DEEPHI 深鉴科技 Now Part of Xilinx

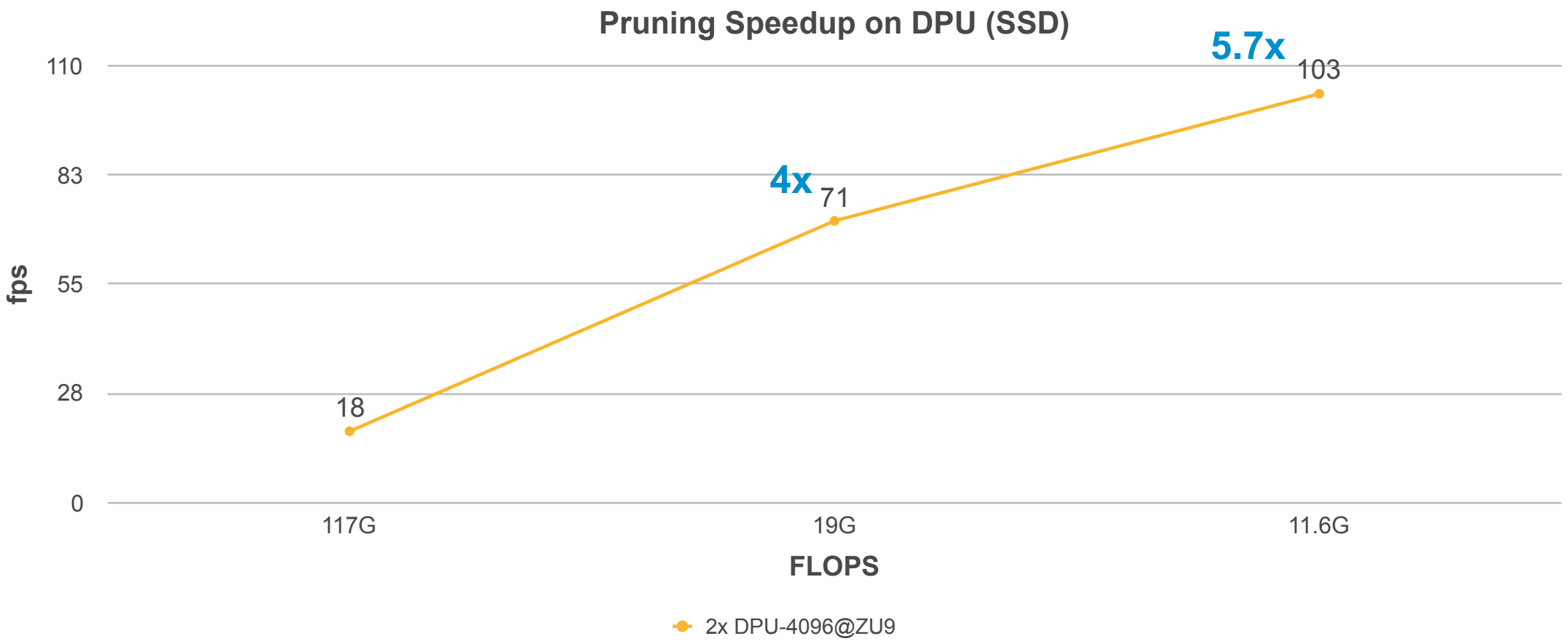
Pruning for Object Detection



Deep Compression Tool (DECENT)

DEEPHI 深鉴科技 Now Part of Xilinx

DECENT Speedup



Deep Compression Tool (DECENT)

DEEPHI 深鉴科技 Now Part of Xilinx

Pruning Accelerates Image Classification on FPGA

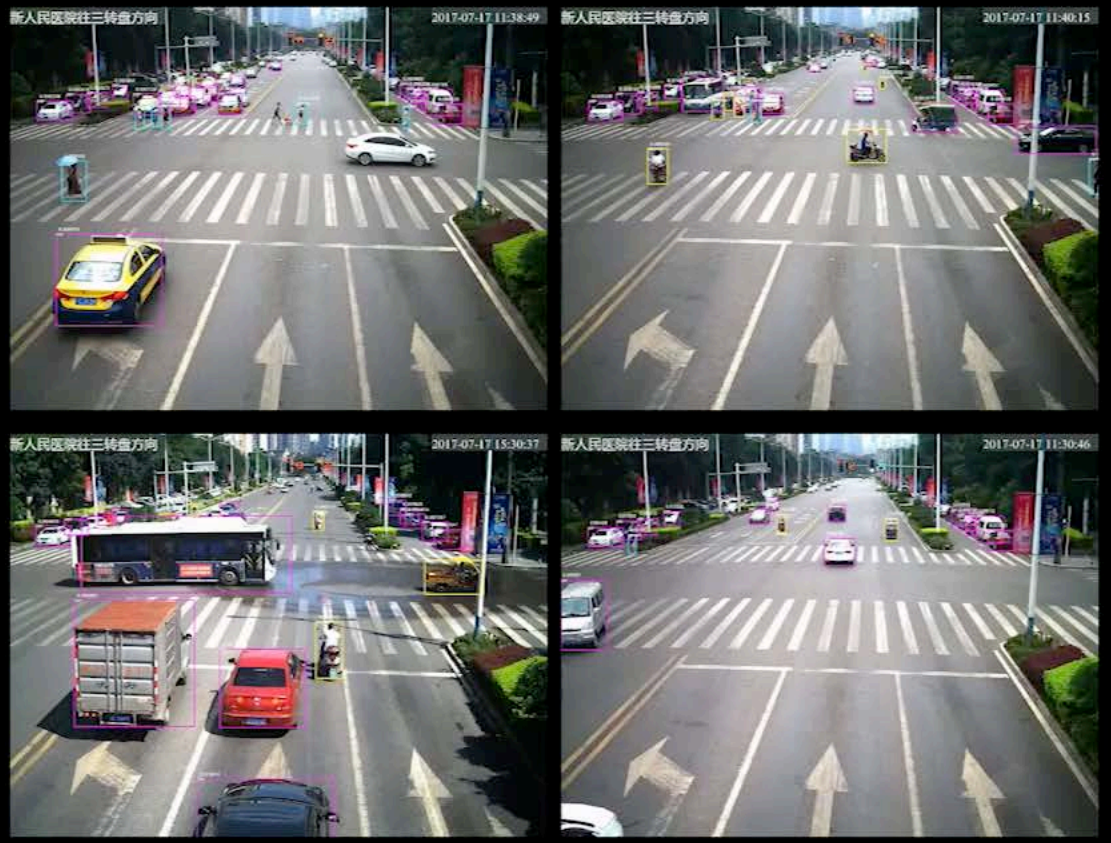
Before Compression
30FPS

After 2.5X Compression
62FPS

DEEPHI
深 鉴 科 技

Pruning Accelerates Object Detection on FPGA

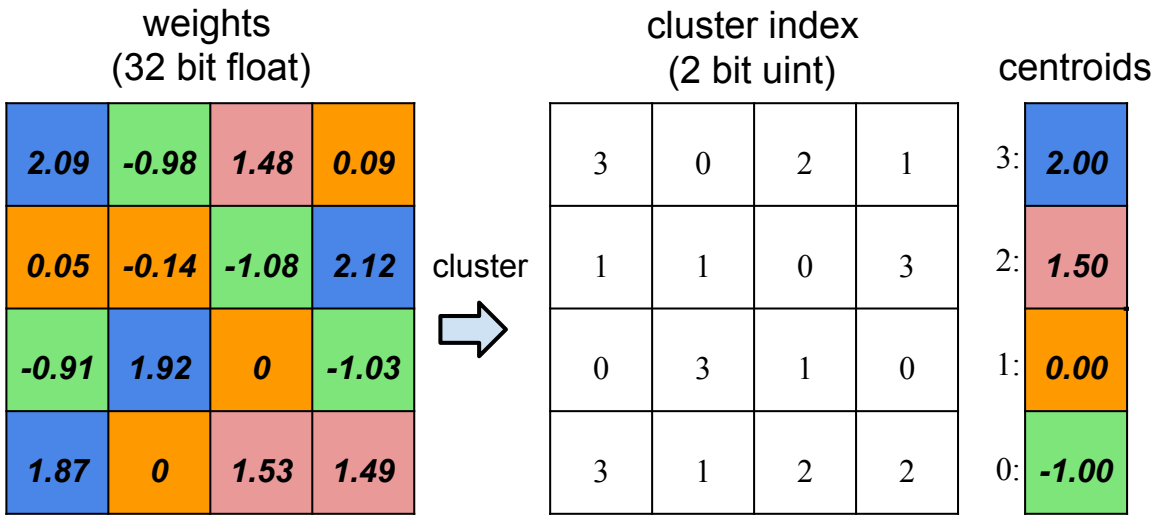
Before Compression
16FPS@120GOP



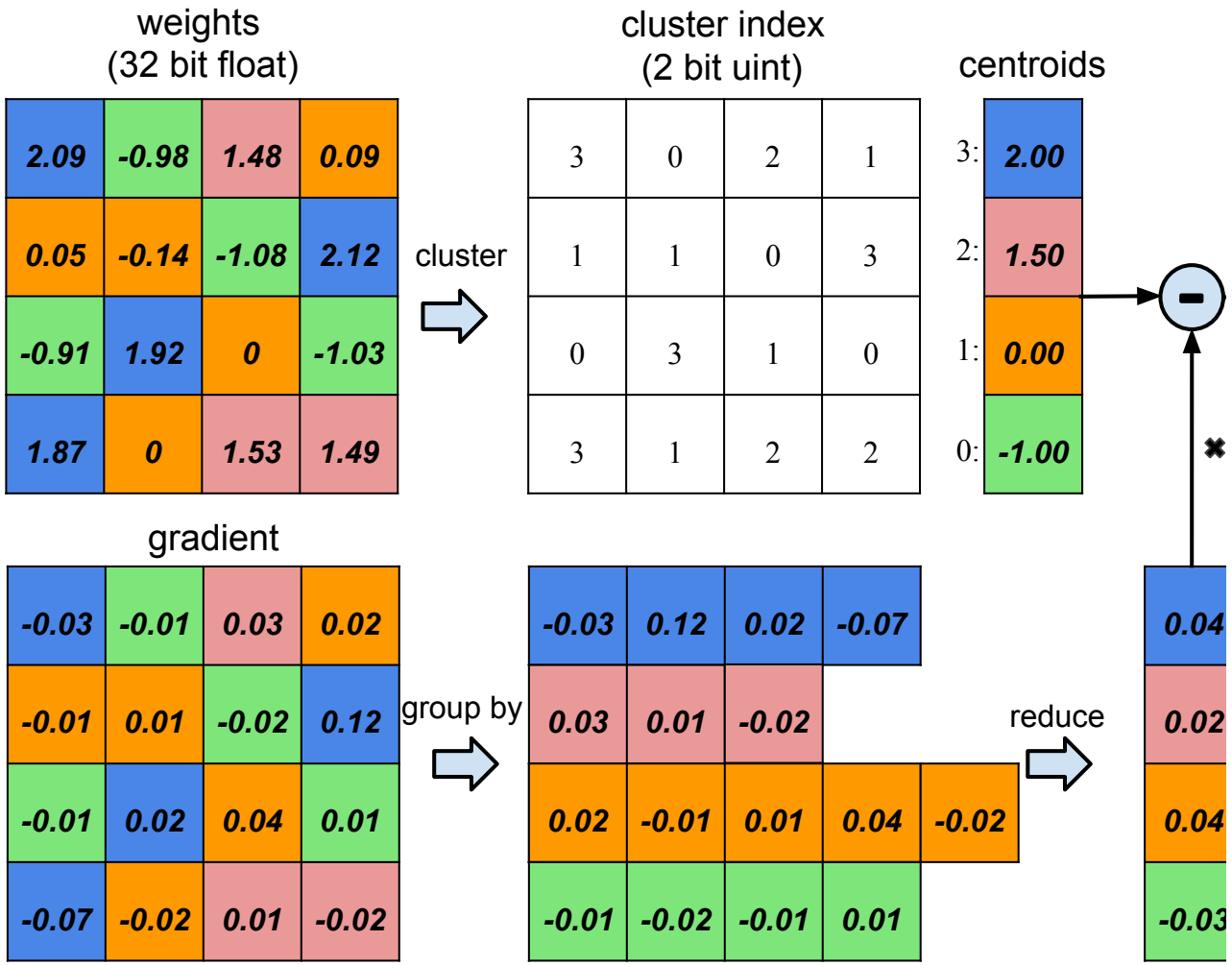
After 10X Compression
125FPS@11.5GOP

DEEPHI
深 鉴 科 技

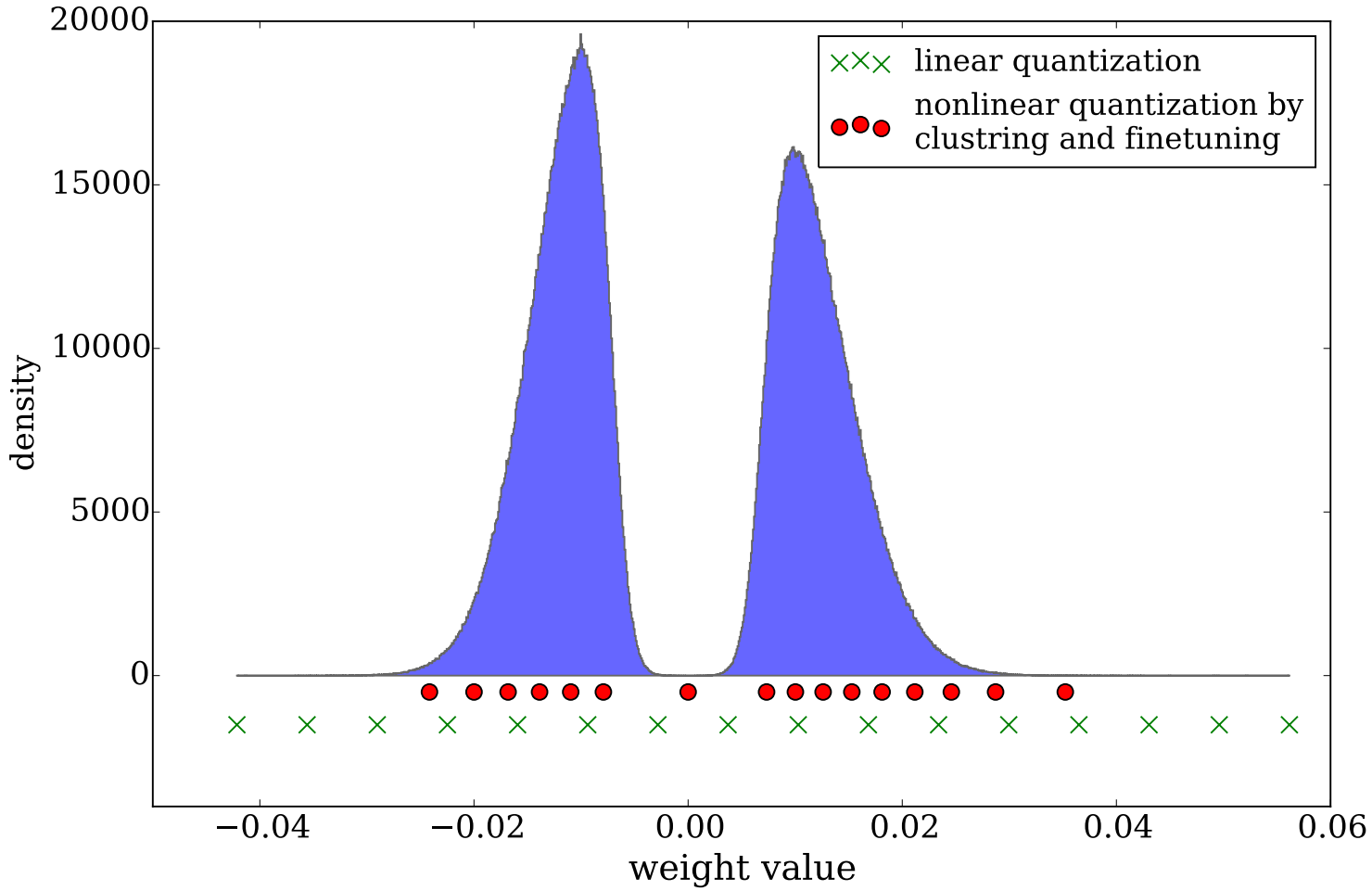
Deep Compression Part2: Trained Quantization



Deep Compression Part2: Trained Quantization

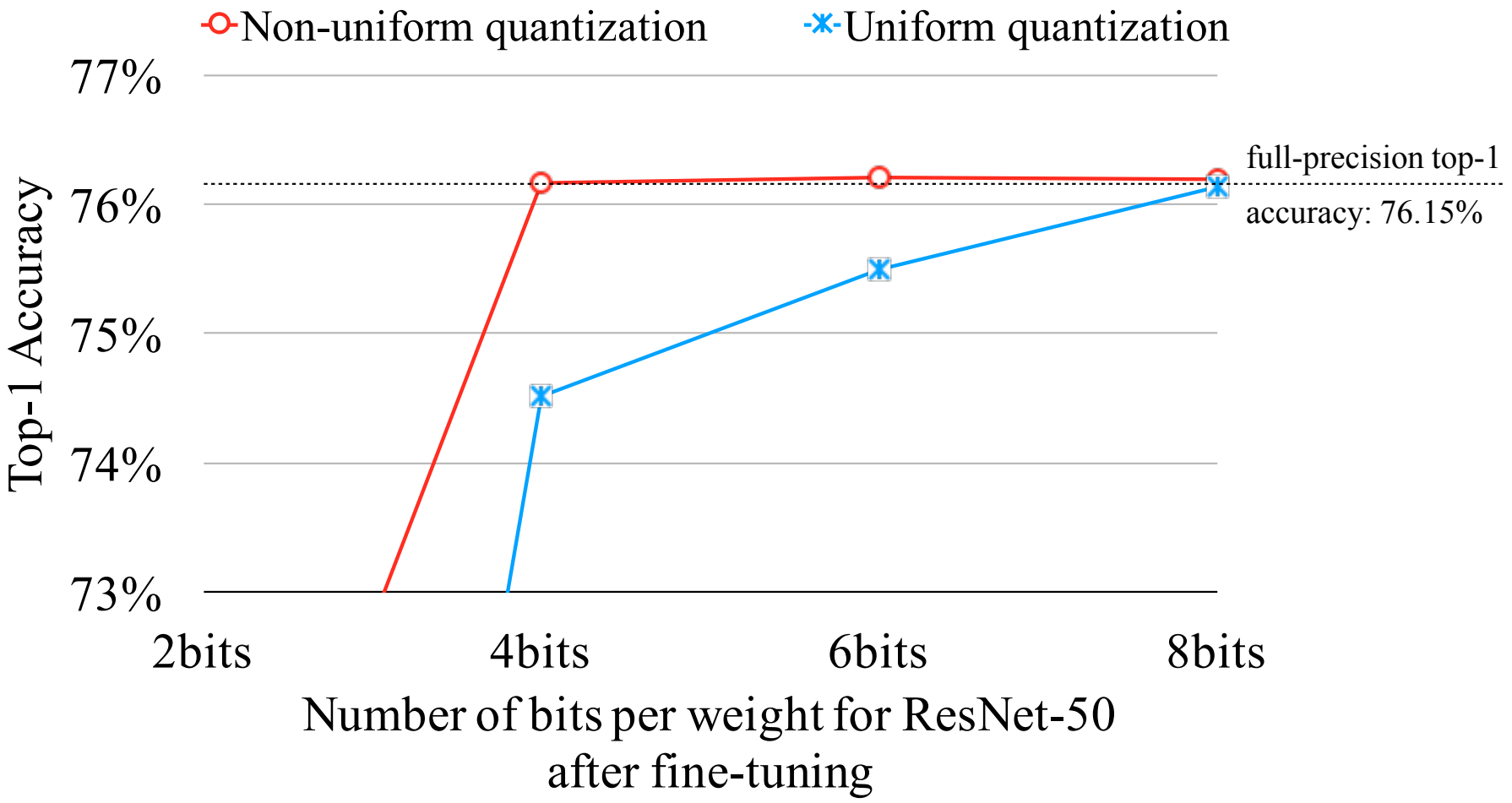


Uniform vs non-Uniform Quantization



Song Han, Huizi Mao, Bill Dally, "Deep Compression", ICLR'16

How Many Bits do We Need?

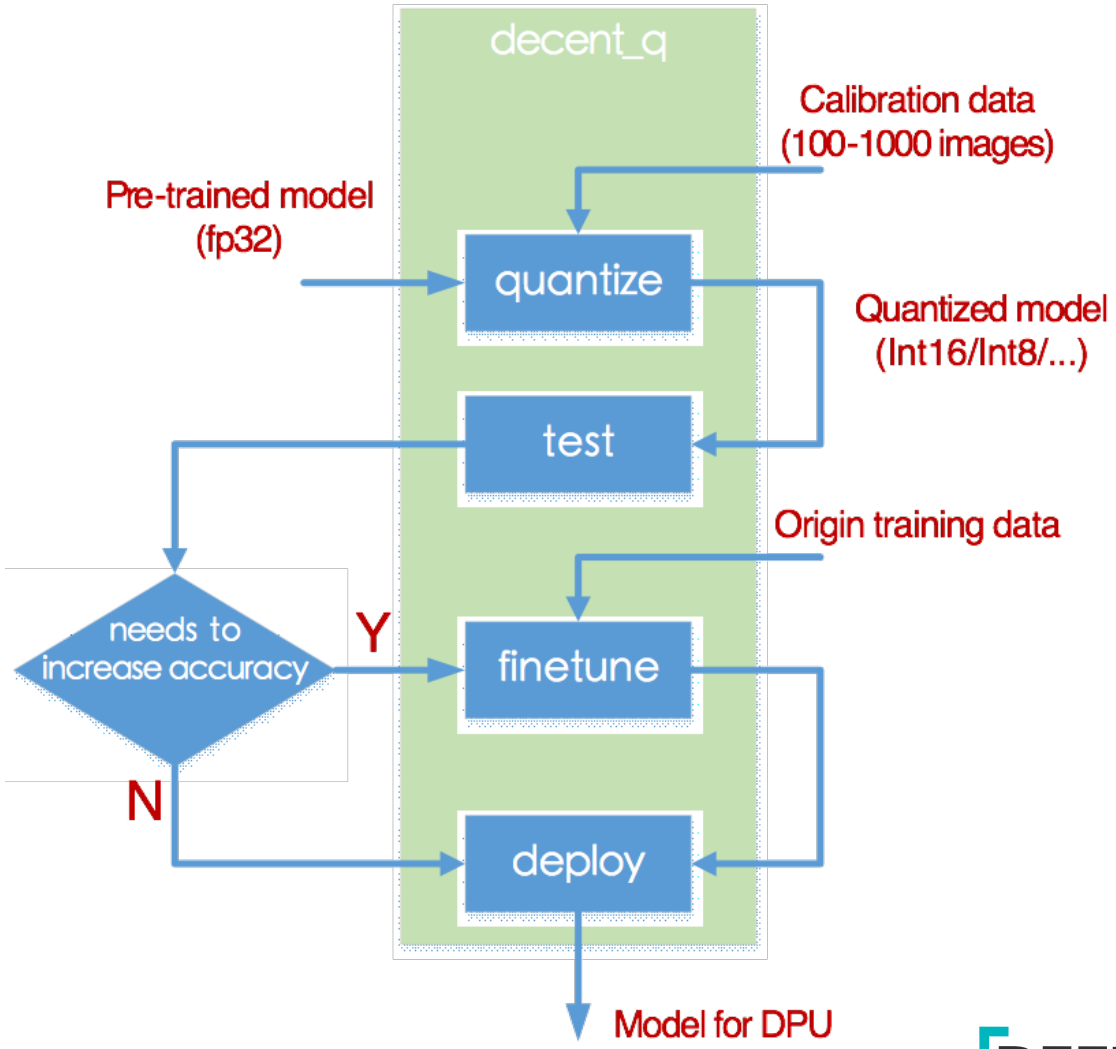


Song Han, Huizi Mao, Bill Dally, "Deep Compression", ICLR'16

DeePhi's Quantization Tool

- 4 commands in decent_q
  - quantize – quantize network
  - test – test network accuracy/mAP
  - finetune – finetune quantized network
  - deploy – generate model for DPU

- Data
  - Calibration data – quantize activation
  - Training data – further increase accuracy

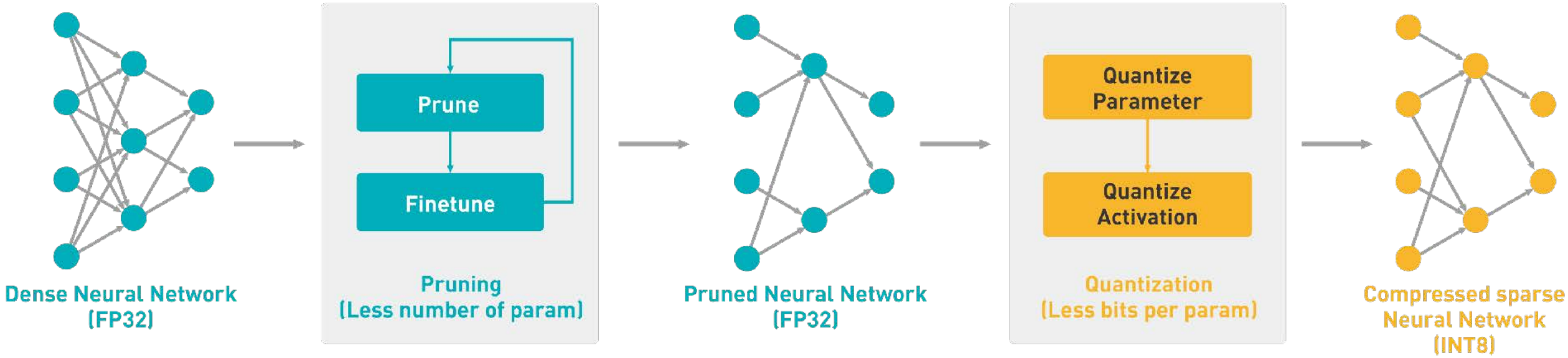


DECENT Quantization Result

- Uniform quantization
  - 8bit for both weights and activation
  - A small set of images for calibration

Networks	float32 baseline		8-bit Quantization			
	Top1	Top5	Top1	ΔTop1	Top5	ΔTop5
Inception_v1	66.90%	87.68%	66.62%	-0.28%	87.58%	-0.10%
Inception_v2	72.78%	91.04%	72.40%	-0.38%	90.82%	-0.23%
Inception_v3	77.01%	93.29%	76.56%	-0.45%	93.00%	-0.29%
Inception_v4	79.74%	94.80%	79.42%	-0.32%	94.64%	-0.16%
ResNet-50	74.76%	92.09%	74.59%	-0.17%	91.95%	-0.14%
VGG16	70.97%	89.85%	70.77%	-0.20%	89.76%	-0.09%
Inception-ResNet-v2	79.95%	95.13%	79.45%	-0.51%	94.97%	-0.16%

DeePhi Deep Compression Tool (DECENT)

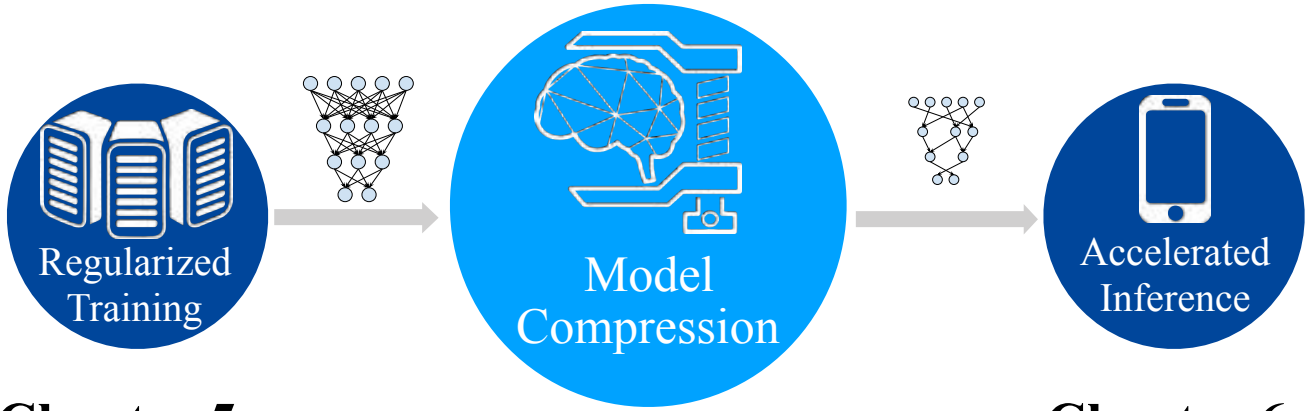


Workflow of Deep Compression Tool (DECENT)

Results: Compression Ratio

Network	Original Size	Compressed Size	Compression Ratio	Original Accuracy	Compressed Accuracy
LeNet-300	1070KB	27KB	**40x**	98.36%	98.42%
LeNet-5	1720KB	44KB	**39x**	99.20%	99.26%
AlexNet	240MB	6.9MB	**35x**	80.27%	80.30%
VGGNet	550MB	11.3MB	**49x**	88.68%	89.09%
Inception-V3	91MB	4.2MB	**22x**	93.56%	93.67%
ResNet-50	97MB	5.8MB	**17x**	92.87%	93.04%

Now we can come to the 3rd generation of NN accelerator: Accelerating Compressed Models



Chapter 5

Han et al. ICLR'17

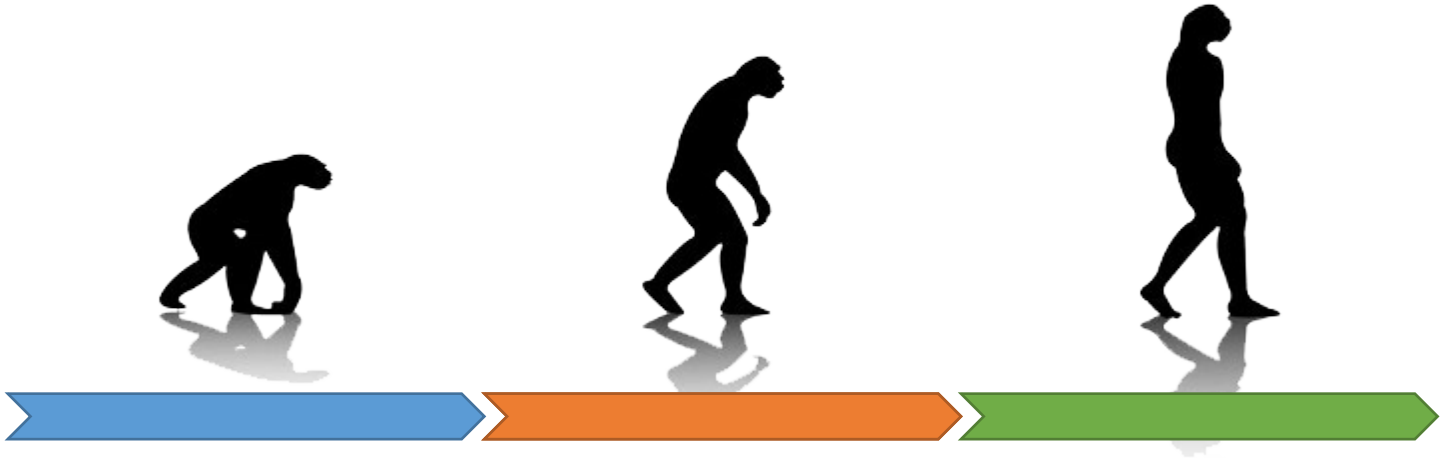
Chapter 3, 4

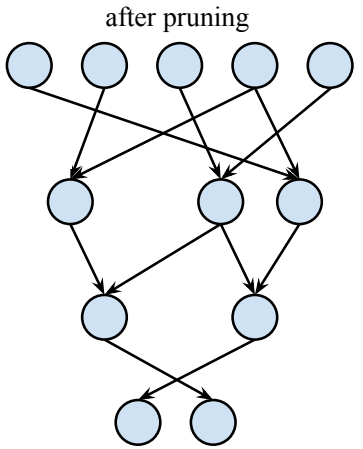
Han et al. NIPS'15
Han et al. ICLR'16

Chapter 6

Han et al. ISCA'16
Han et al. FPGA'17

Part 3/4: Accelerating Compressed Model



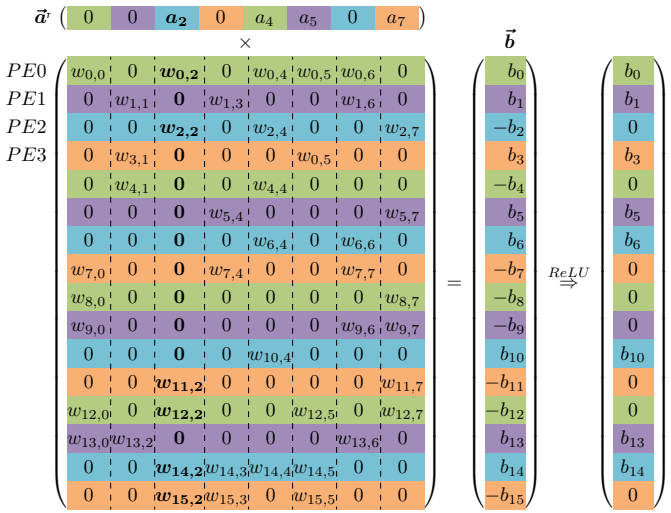


Compression before Acceleration

- EIE [Han, 2016]
- Cnvlutin [Albericio, 2016]
- Cambricon-X [Zhang, 2016]
- ESE [Han, 2017]
- SCNN [Parashar, 2017]

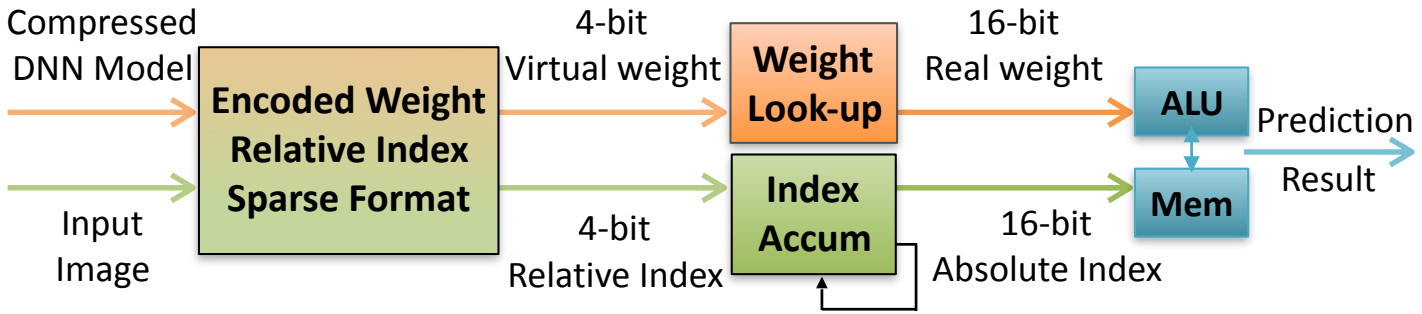
Part 3/4: Accelerating Compressed Model

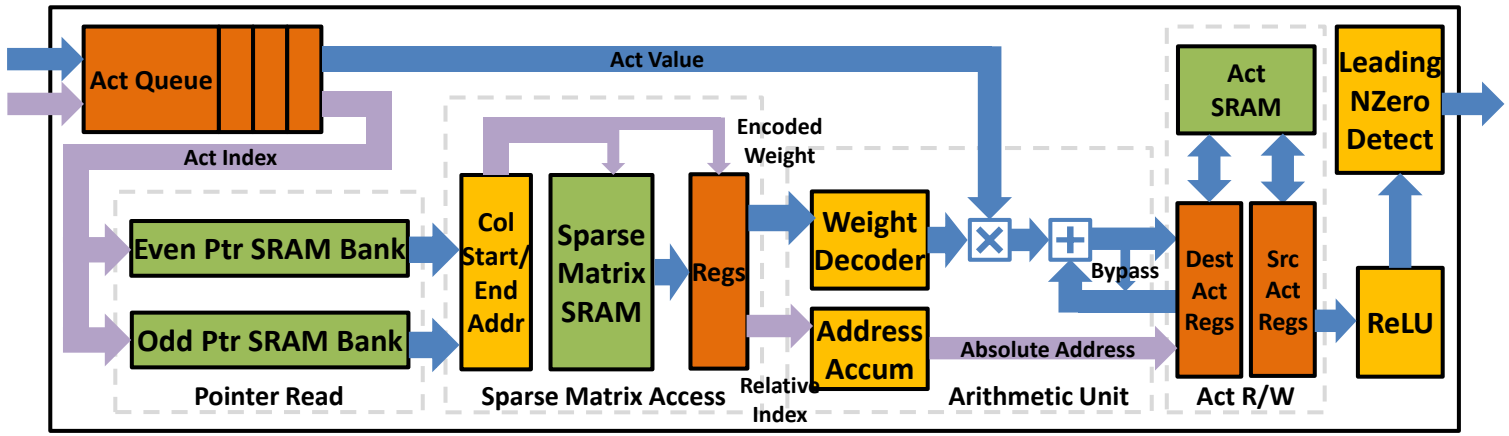
EIE [Han, 2016]



Save Memory by Deep Compression

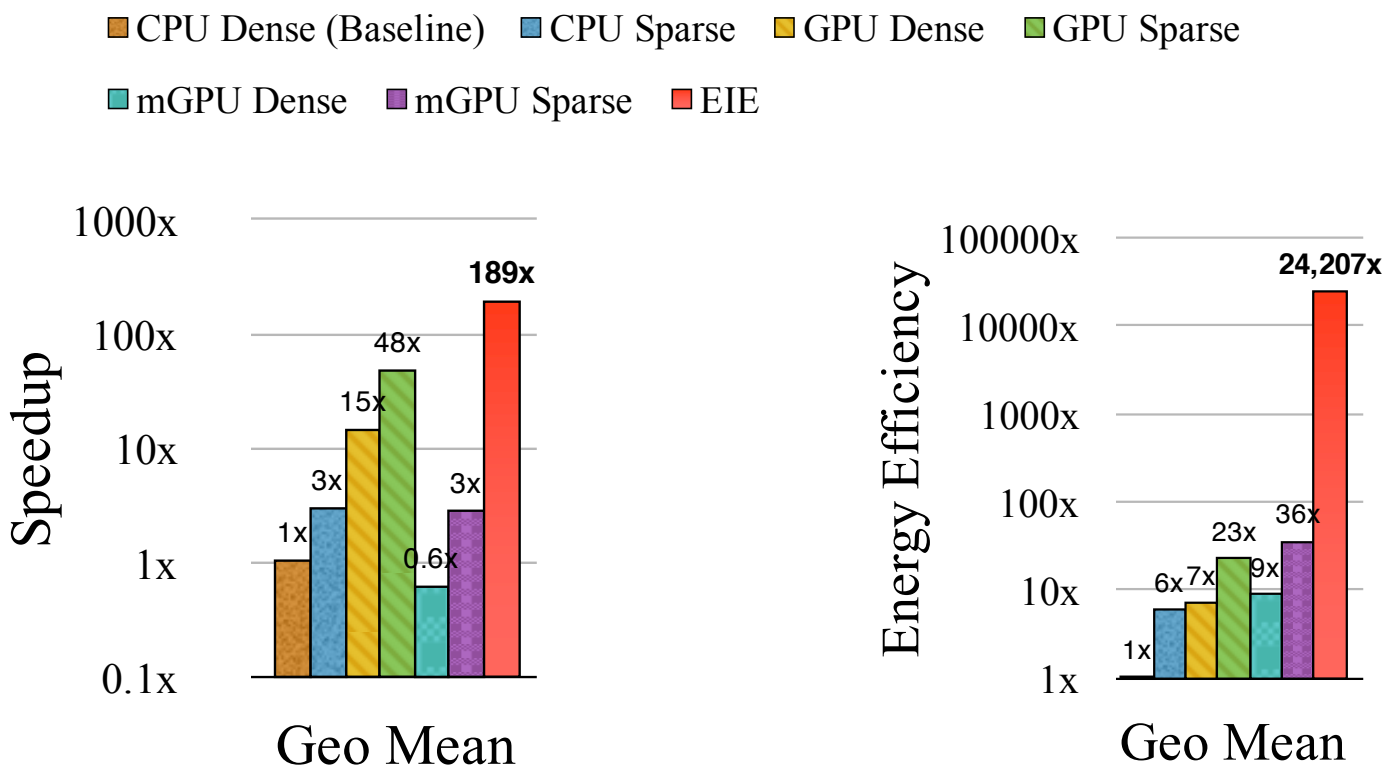
Virtual Weight	$W_{0,0}$	$W_{8,0}$	$W_{12,0}$	$W_{4,1}$	$W_{0,2}$	$W_{12,2}$	$W_{0,4}$	$W_{4,4}$	$W_{0,5}$	$W_{12,5}$	$W_{0,6}$	$W_{8,7}$	$W_{12,7}$
Relative Row Index	0	1	0	1	0	2	0	0	0	2	0	2	0
Column Pointer	0	3	4	6	6	8	10	11	13				





Part 3/4: Accelerating Compressed Model

EIE [Han, 2016]

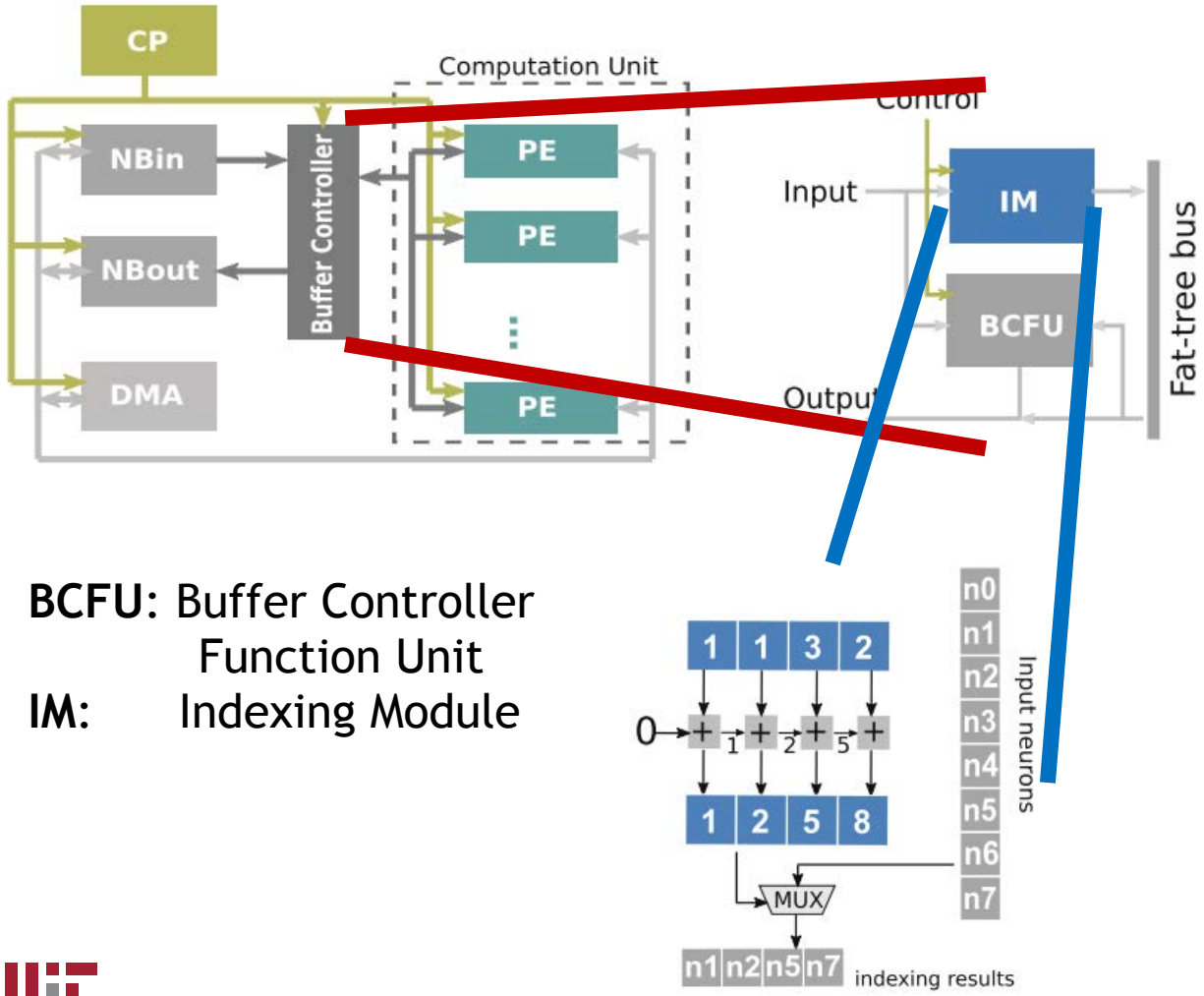


Geo Mean Over AlexNet, VGG, NeuralTalk

Song Han, Xingyu Liu, Huizi Mao, Jing Pu, Ardavan Pedram, Mark A. Horowitz, Bill Dally
"EIE: Efficient Inference Engine on Compressed Deep Neural Network", ISCA'16

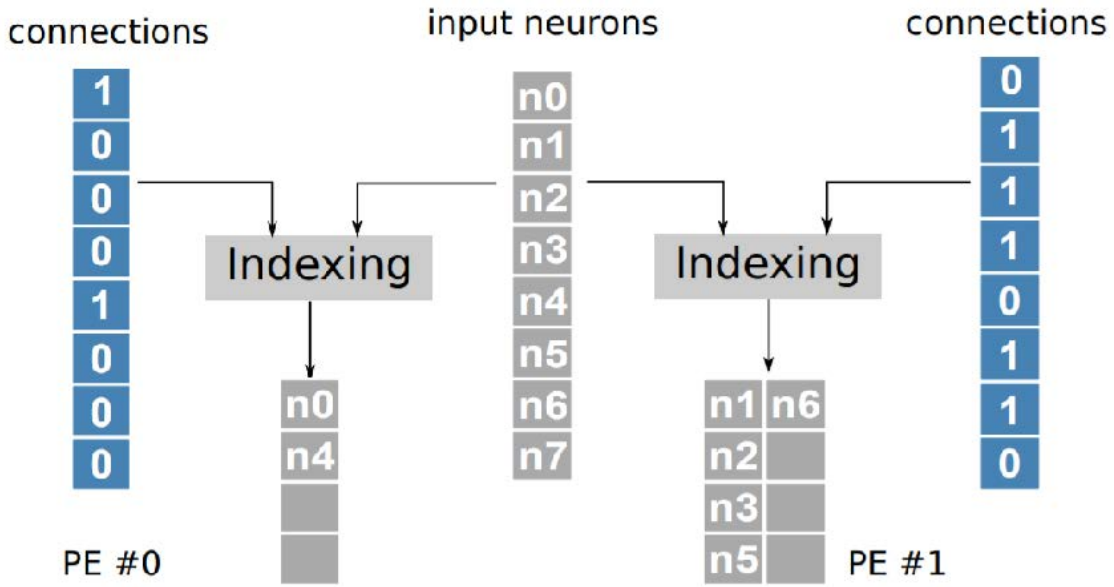
Part 3/4: Accelerating Compressed Model

Cambricon-X [Zhang, 2016]



BCFU: Buffer Controller Function Unit

IM: Indexing Module

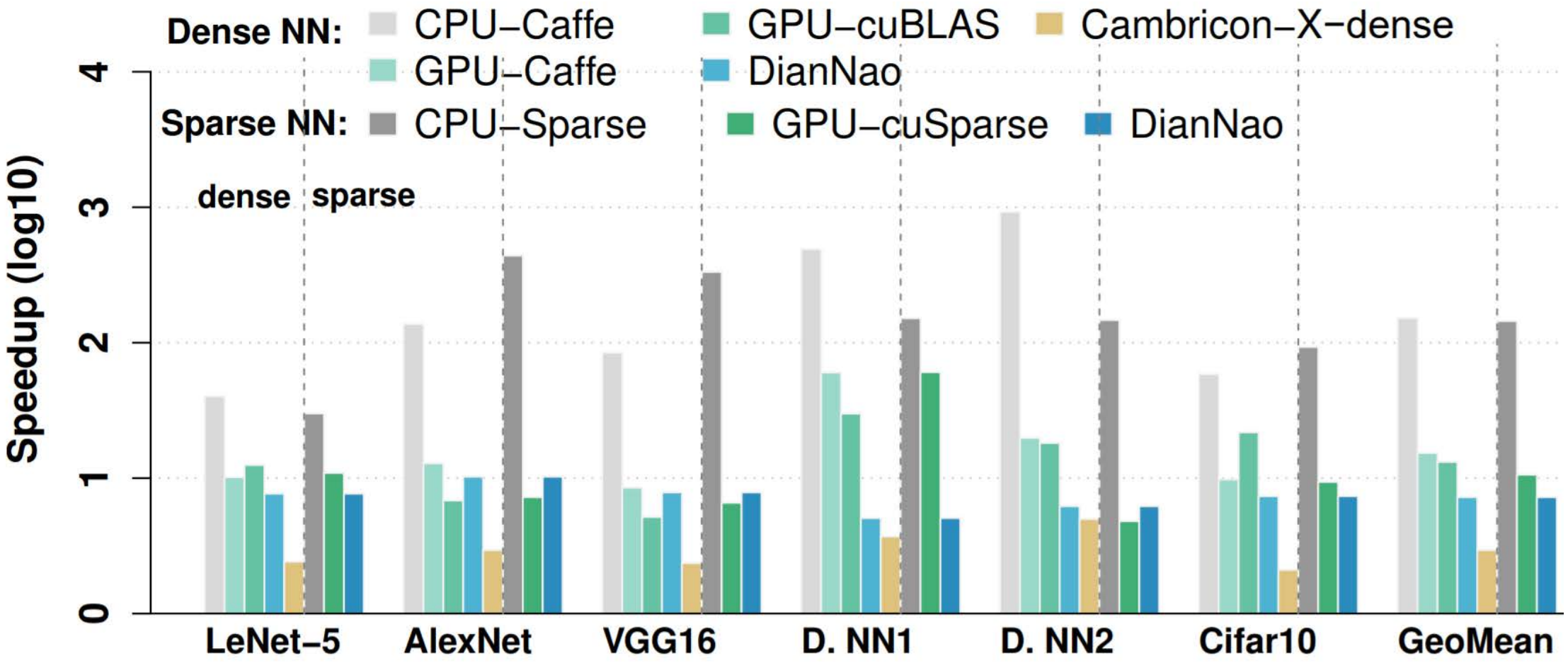


"…the inputs are directed fed into the BCFU…"

"…*only transfer the indexed neurons to PEs*…"

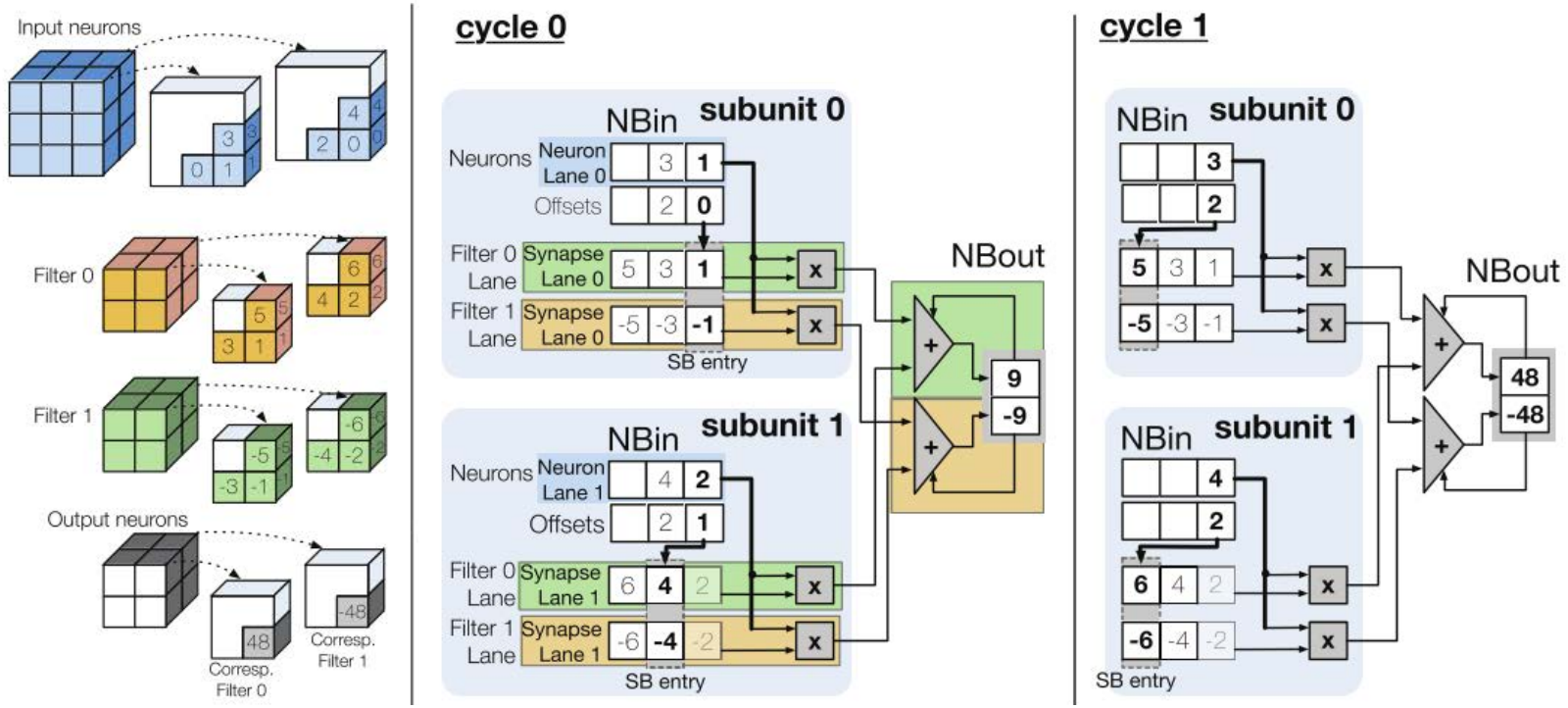
Part 3/4: Accelerating Compressed Model

Cambricon-X [Zhang, 2016]



Part 3/4: Accelerating Compressed Model

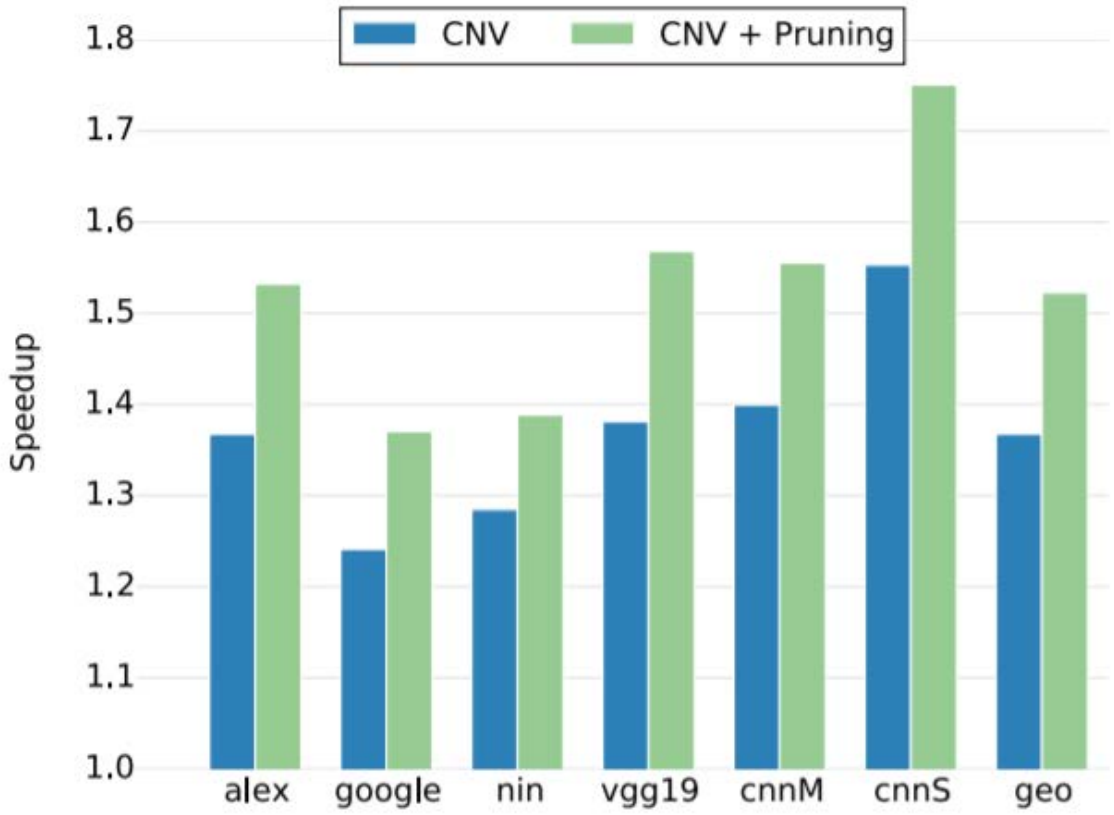
Cnvlutin [Albericio, 2016]



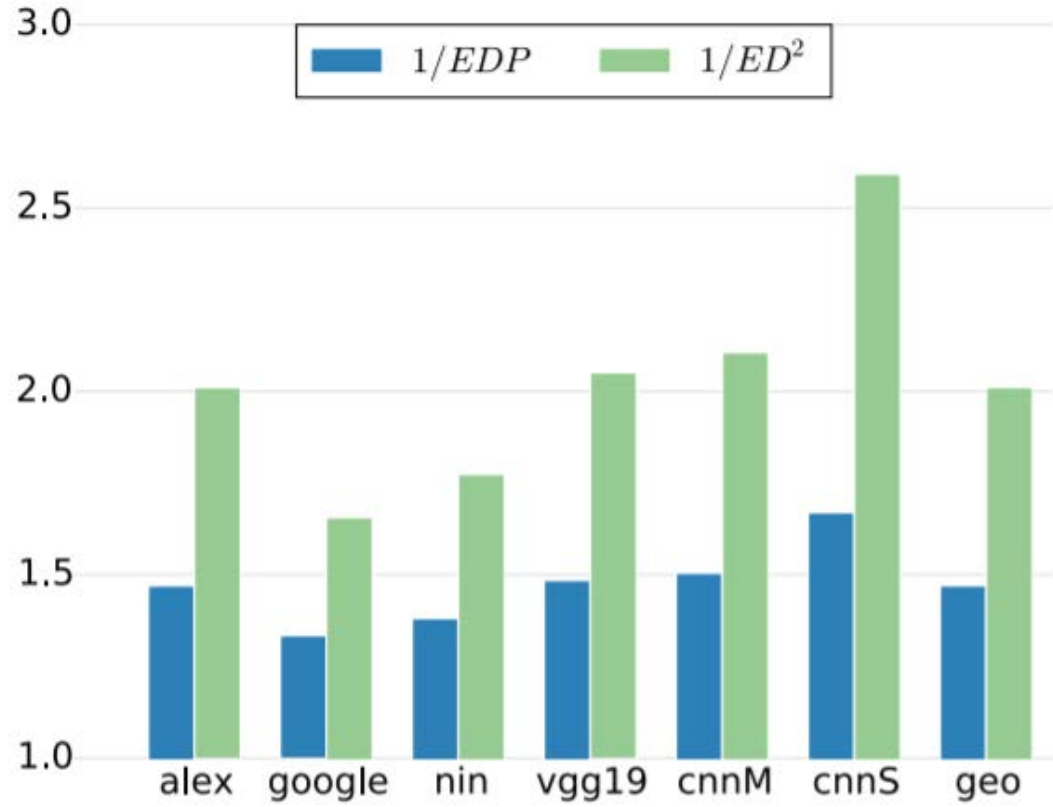
Index weights by Nonzero Activations

Part 3/4: Accelerating Compressed Model

Cnvlutin [Albericio, 2016]



Speedup Compared to the Baseline

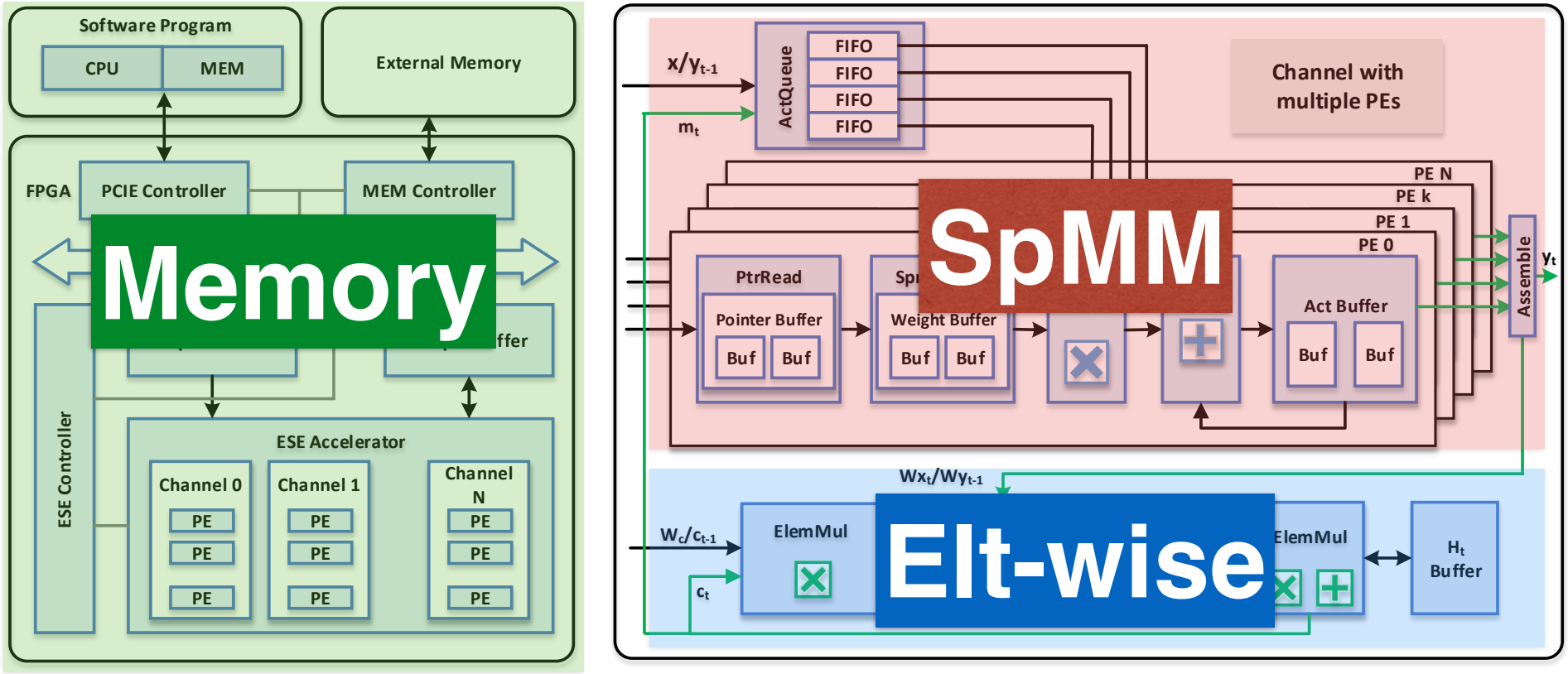


Energy Delay Product
Compared to DaDianNao

Jorge Albericio Patrick Judd Tayler Hetherington Tor Aamodt Natalie Enright Jerger Andreas Moshovos
"Cnvlutin: Ineffectual-Neuron-Free Deep Neural Network Computing", ISCA'16

Part 3/4: Accelerating Compressed Model

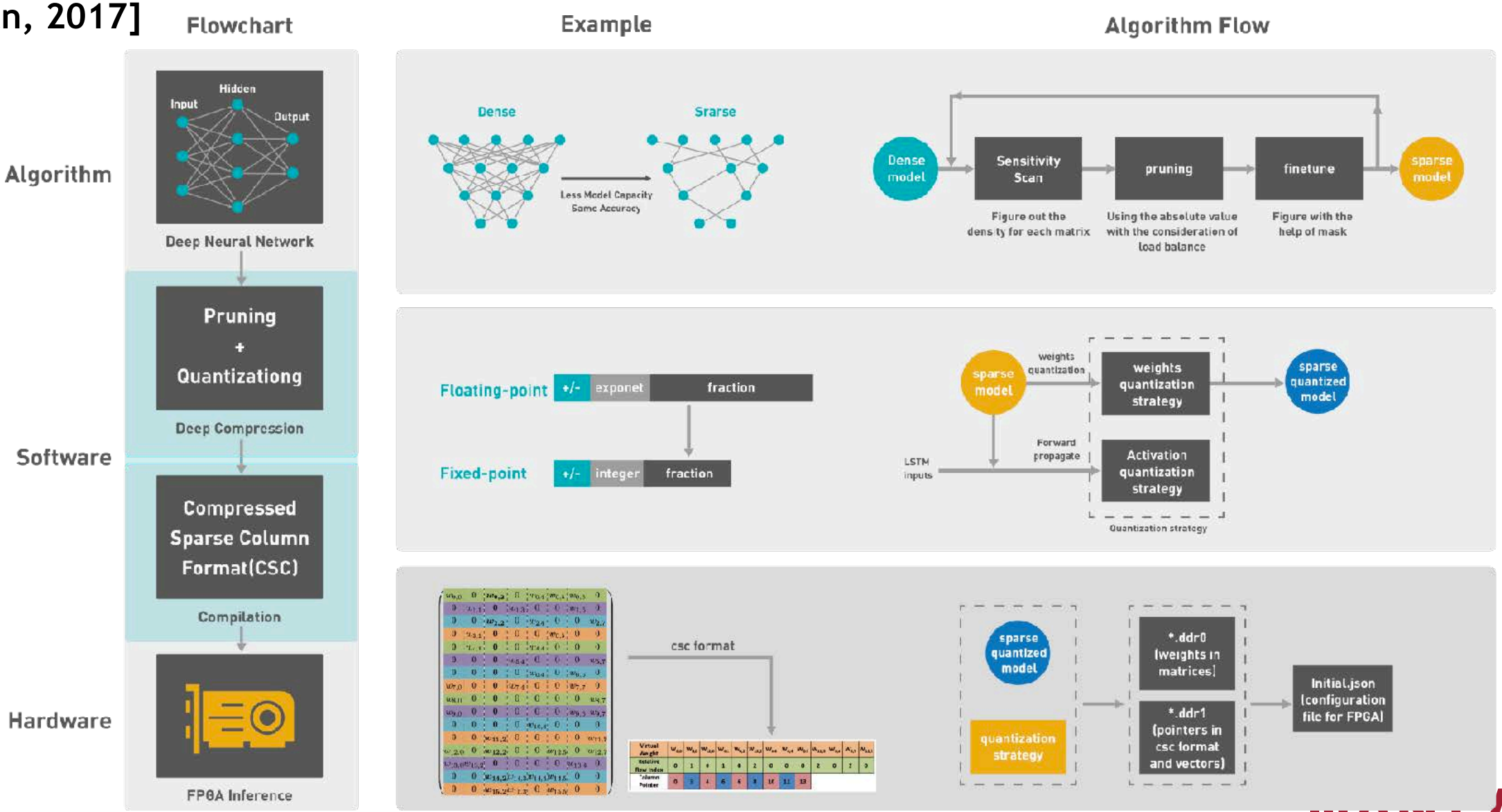
ESE [Han, 2017]



Song Han, Junlong Kang, Huizi Mao, Yiming Hu, Xin Li, Yubin Li, Dongliang Xie, Hong Luo, Song Yao, Yu Wang, Huazhong Yang, Bill Dally,
ESE: Efficient Speech Recognition Engine with Sparse LSTM on FPGA, FPGA'17
https://aws.amazon.com/marketplace/pp/B079N2J42R

Part 3/4: Accelerating Compressed Model

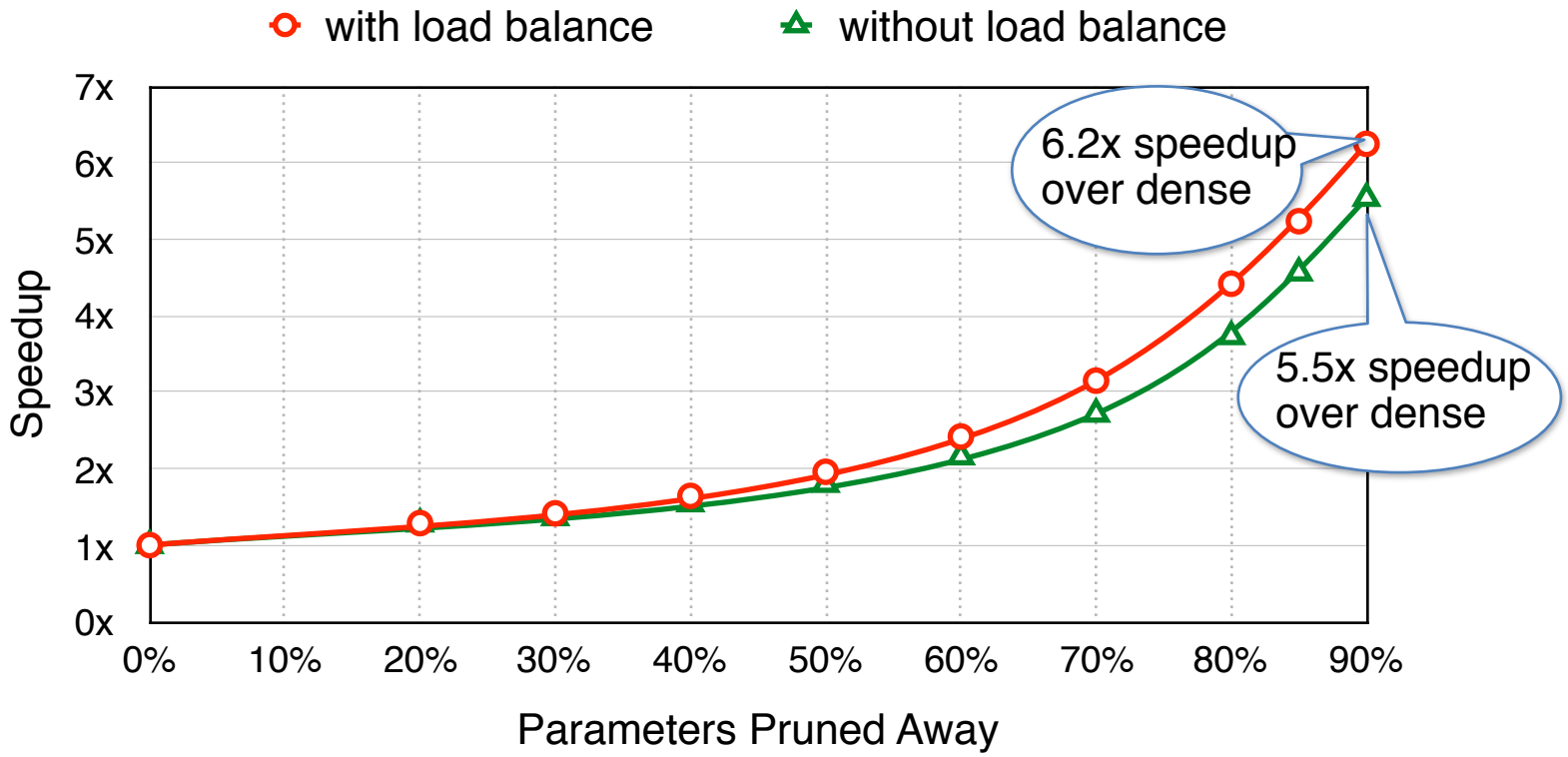
ESE [Han, 2017]



https://aws.amazon.com/marketplace/pp/B079N2J42R

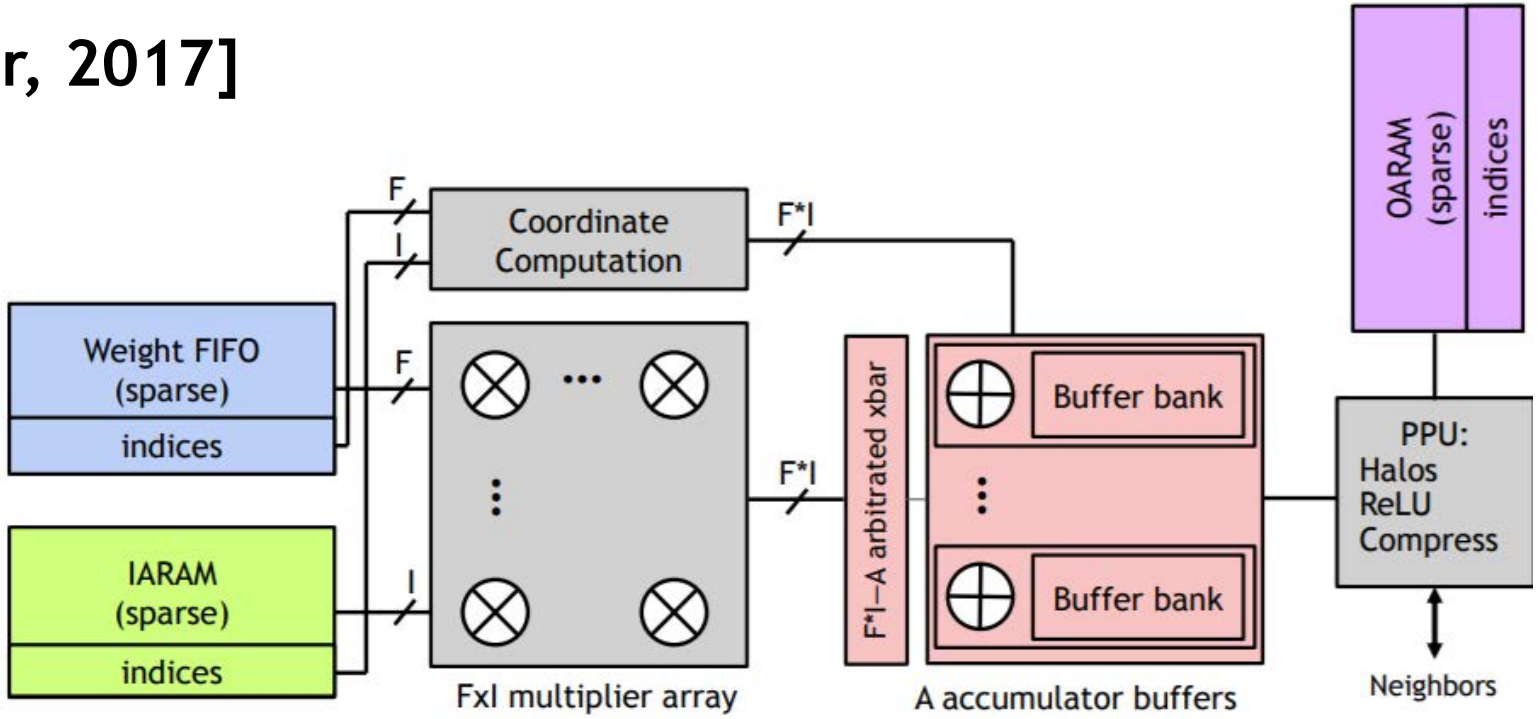
Part 3/4: Accelerating Compressed Model

ESE [Han, 2017]



Part 3/4: Accelerating Compressed Model

SCNN [Parashar, 2017]

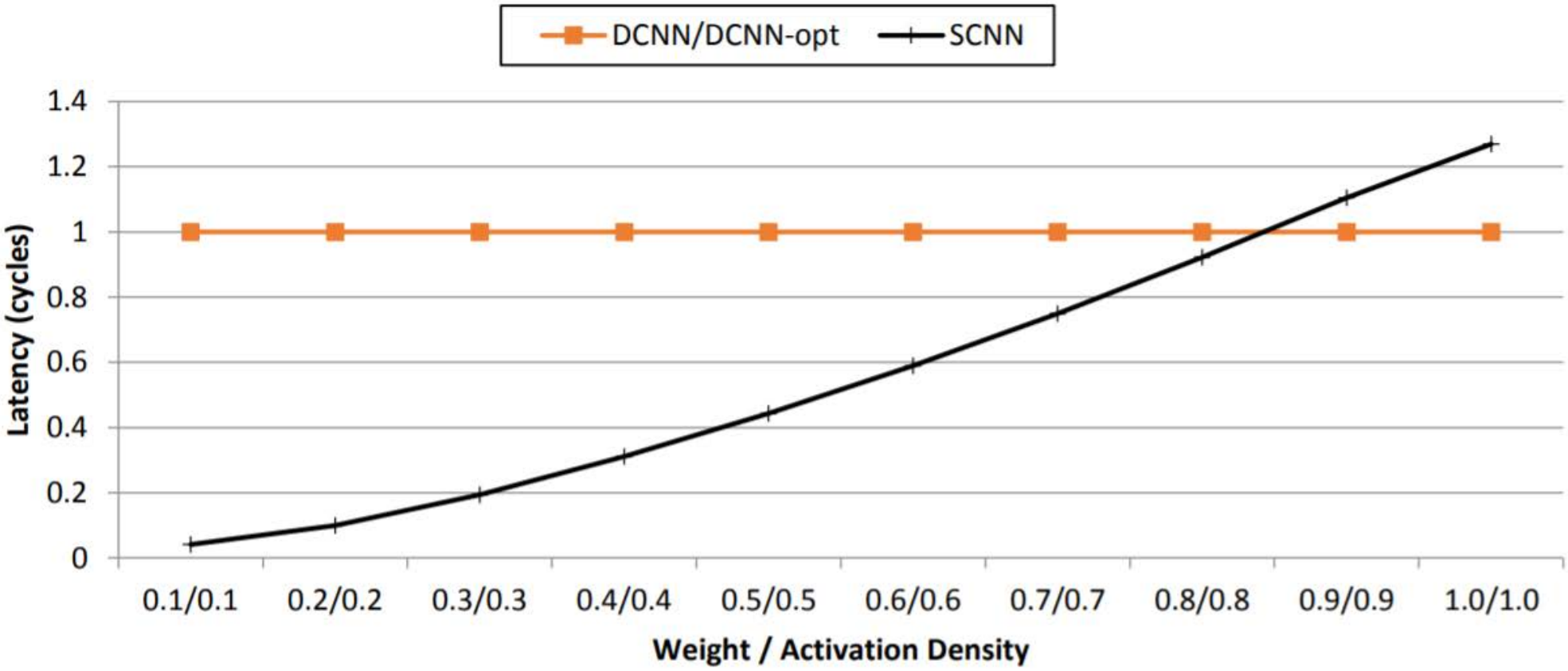


"we employ a novel ***Cartesian product*** dataflow that … performs ***an all-to-all multiply of non-zero weight and activation*** vector elements that …"

Angshuman Parashar, Minsoo Rhu, Anurag Mukkara, Antonio Puglielli, Rangharajan Venkatesan, Brucek Khailany, Joel Emer, Stephen W. Keckler, Bill Dally "Scnn: An accelerator for compressed-sparse convolutional neural networks." *ISCA*'2017

Part 3/4: Accelerating Compressed Model

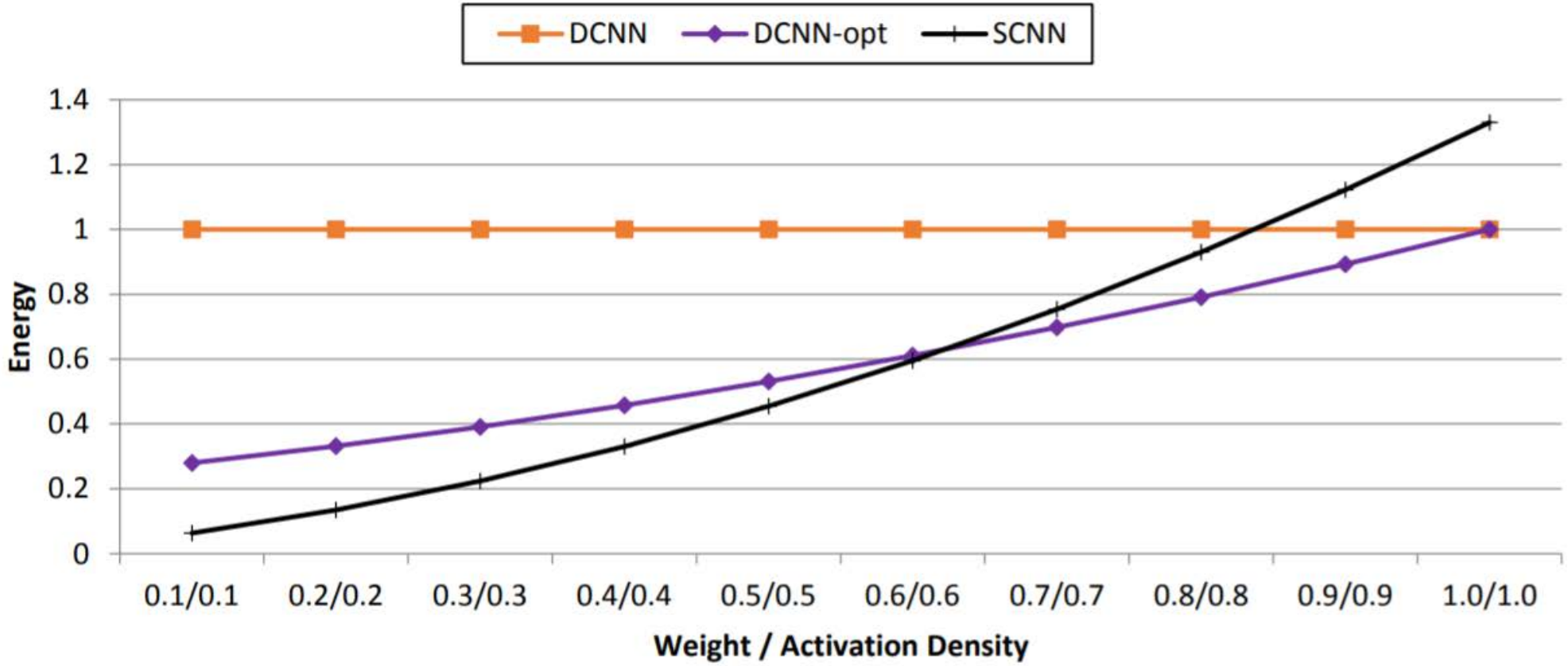
SCNN [Parashar, 2017]



Angshuman Parashar, Minsoo Rhu, Anurag Mukkara, Antonio Puglielli, Rangharajan Venkatesan, Brucek Khailany, Joel Emer, Stephen W. Keckler, Bill Dally "Scnn: An accelerator for compressed-sparse convolutional neural networks." *ISCA*'2017

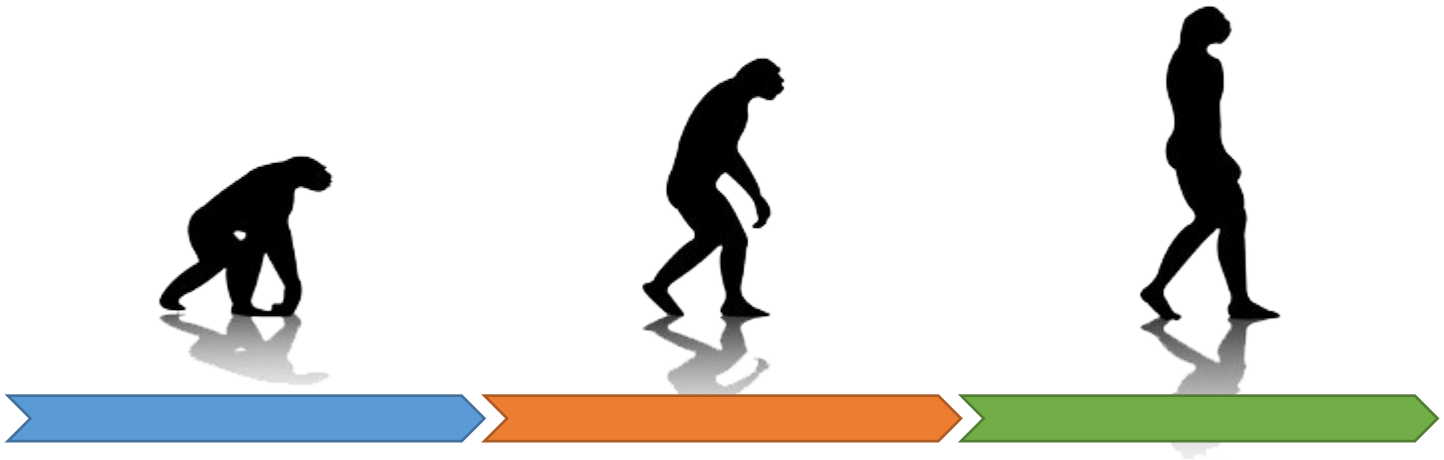
Part 3/4: Accelerating Compressed Model

SCNN [Parashar, 2017]



Angshuman Parashar, Minsoo Rhu, Anurag Mukkara, Antonio Puglielli, Rangharajan Venkatesan, Brucek Khailany, Joel Emer, Stephen W. Keckler, Bill Dally "Scnn: An accelerator for compressed-sparse convolutional neural networks." *ISCA*'2017

Evolution of NN Accelerators

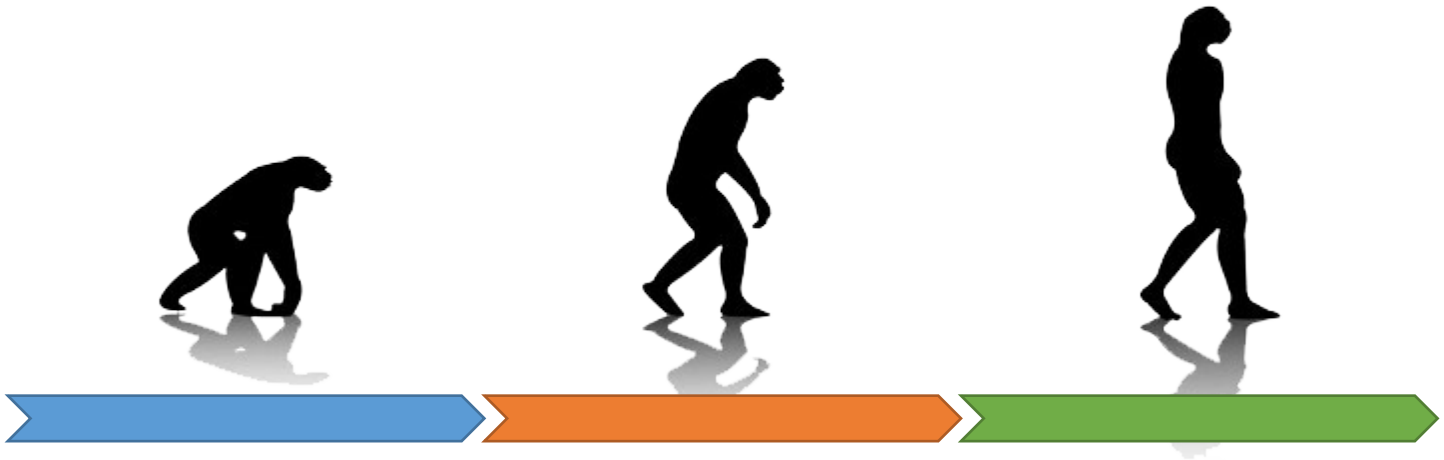


Compression before Acceleration

Memory-centric Specialization

Computation Specialization

Evolution of NN Accelerators



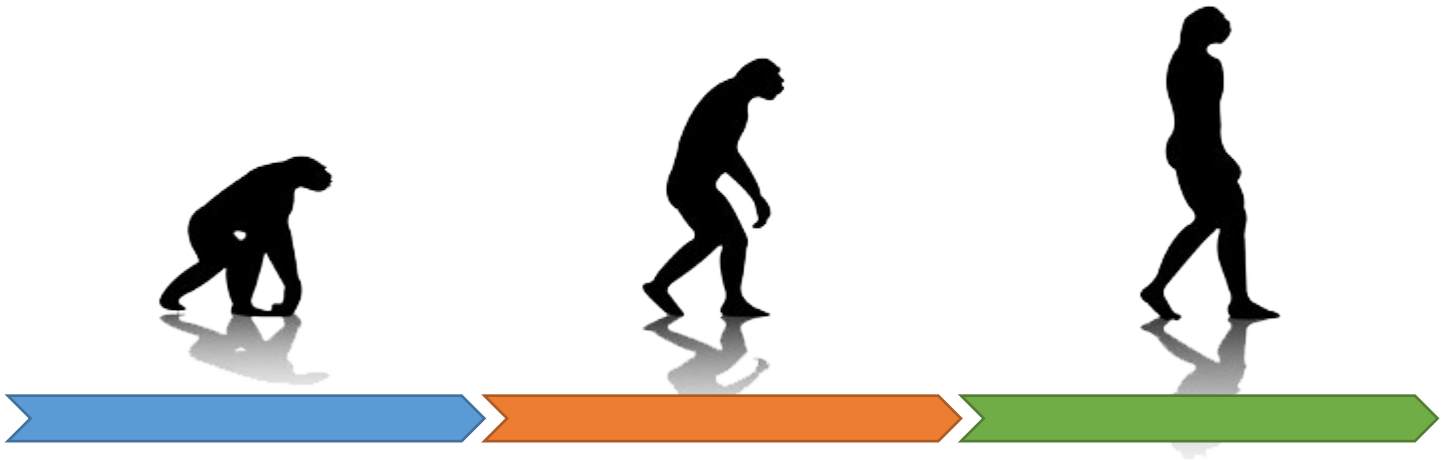
Compression before Acceleration

Memory-centric Specialization

Computation Specialization

AlexNet, VGG

Evolution of NN Accelerators



Compression before Acceleration

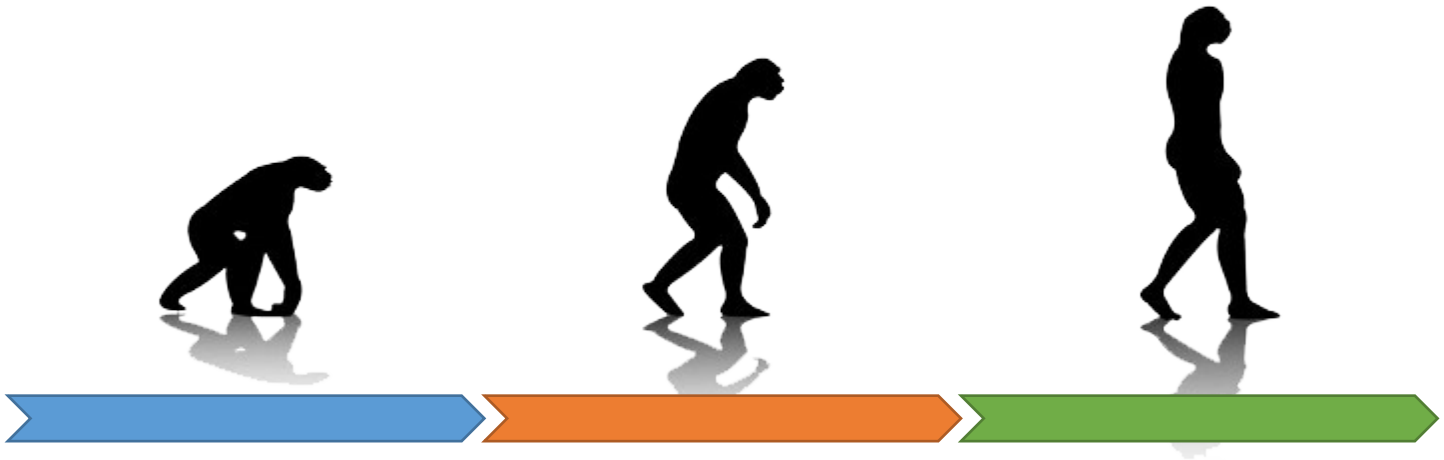
Memory-centric Specialization

Computation Specialization

GoogleNet, ResNet

AlexNet, VGG

Evolution of NN Accelerators



The End ?

Compression before Acceleration

Memory-centric Specialization

Sparse NN

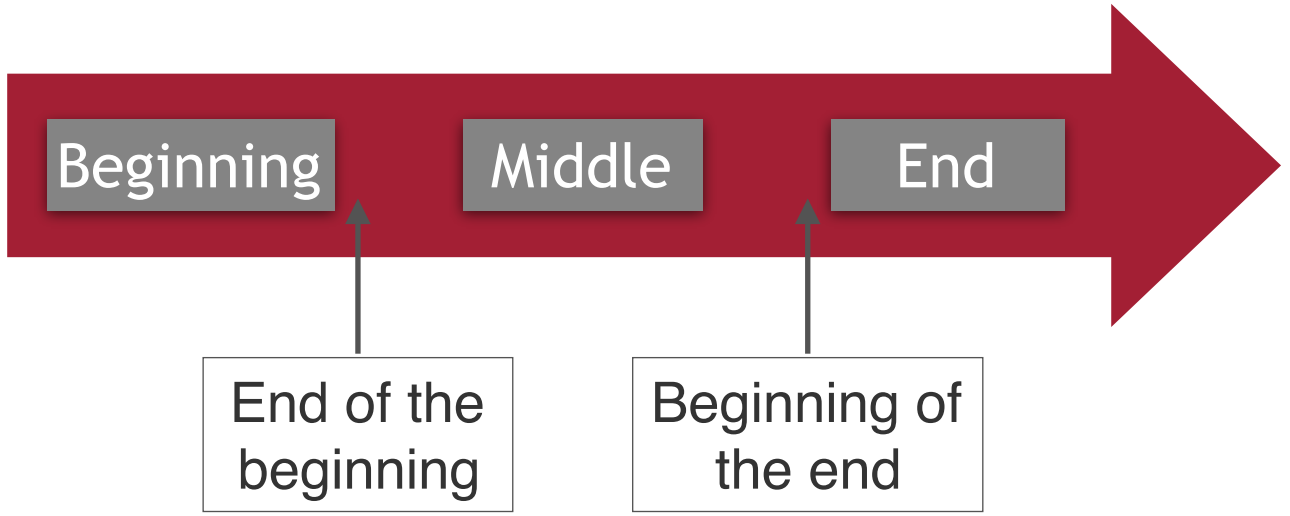
Computation Specialization

GoogleNet, ResNet

AlexNet, VGG

"Now this is not the end.
It is not even the beginning of the end.
But it is, perhaps, the end of the beginning."

—— Winston Churchill, 1942



Not the end. There's plenty of room at the top

given the computation pattern of deep learning and achieved higher efficiency compared with CPUs and GPUs. The first wave of accelerators efficiently implemented the computational primitives for neural networks [16, 18, 24]. Researchers then realized that memory access is more expensive and critically needs optimization, so the second wave of accelerators efficiently optimized memory transfer and data movement [19–23]. These two generations of accelerators have made promising progress in improving the speed and energy efficiency of running DNNs.

However, both generations of deep learning accelerators treated the algorithm as a black box and focused on only optimizing the hardware architecture. In fact, there is plenty of room at the top by optimizing the algorithm. We found that DNN models can be significantly compressed and simplified before touching the hardware; if we treat these DNN models merely as a black box and hand them directly to hardware, there is massive redundancy in the workload. However, existing hardware accelerators are optimized for uncompressed DNN models, resulting in huge wastes of computation cycles and memory bandwidth compared with running on compressed DNN models. We therefore need to co-design the algorithm and the hardware.

In this dissertation, we co-designed the algorithm and hardware for deep learning to make it run faster and more energy-efficiently. We developed techniques to make the deep learning workload more efficient and compact to begin with and then designed the hardware architecture specialized for the optimized DNN workload. Figure 1.1 illustrates the design methodology of this thesis. Breaking

Song Han, "EFFICIENT METHODS AND HARDWARE FOR DEEP LEARNING". Stanford PhD Thesis advised by Prof. Bill Dally

Evolution of NN Accelerators

Computation Specialization

AlexNet, VGG

Memory-centric Specialization

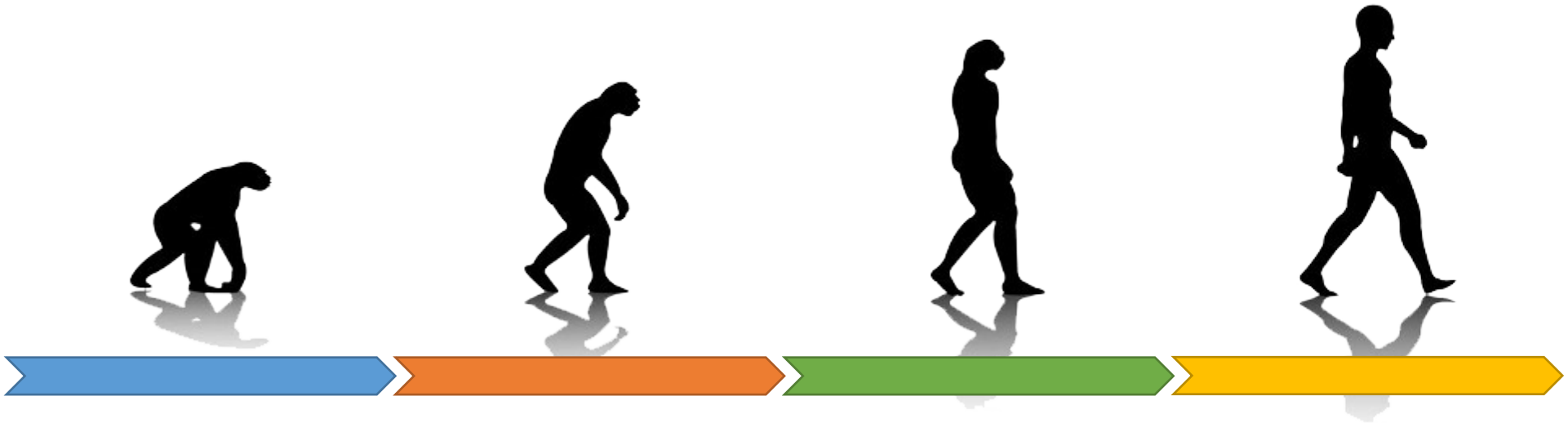
GoogleNet, ResNet

Compression before Acceleration

Sparse NN

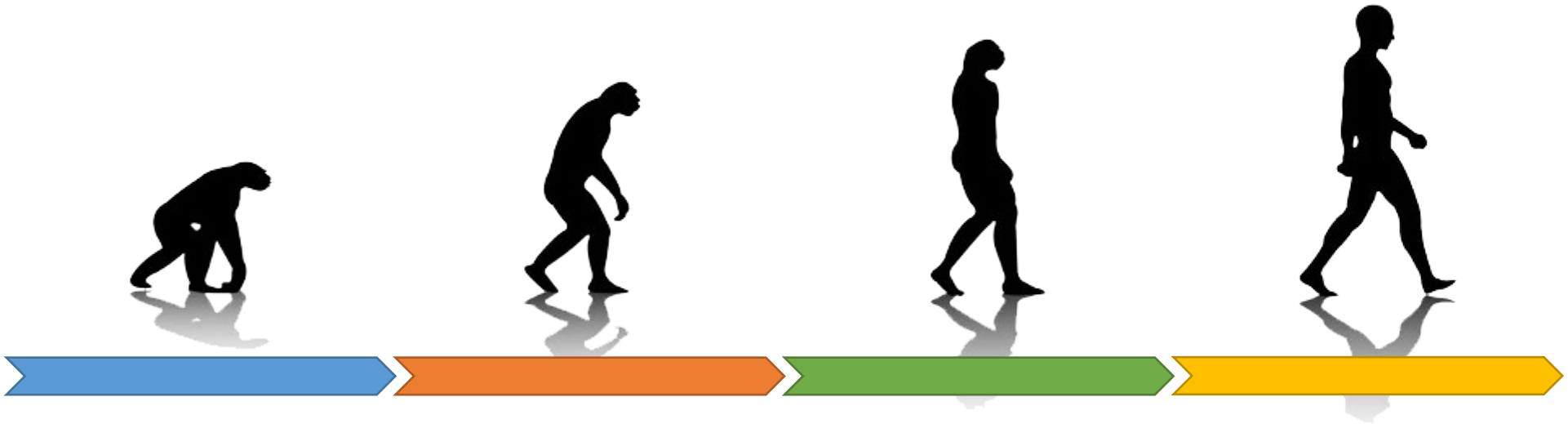
MobileNet,
ShiftNet,
and More

Evolution of NN Accelerators



Hardware & NN
Architectures
Co-Design

Part 4/4: HW Architecture/NN Architecture Co-design



Hardware & NN Architectures Co-Design

SqueezeNext ←→ • Squeezelerator [Kwon, 2018]

Depth-wise Convolution → • DeePhi's DPU-v2 [DeePhi, 2018]

ShiftNet ←→ • ShiftNet Accelerator [Wu, 2018]

Part 4/4: HW Architecture/NN Architecture Co-design

SqueezeNext, Squeezelerator [Gholami, Kwon, 2018]

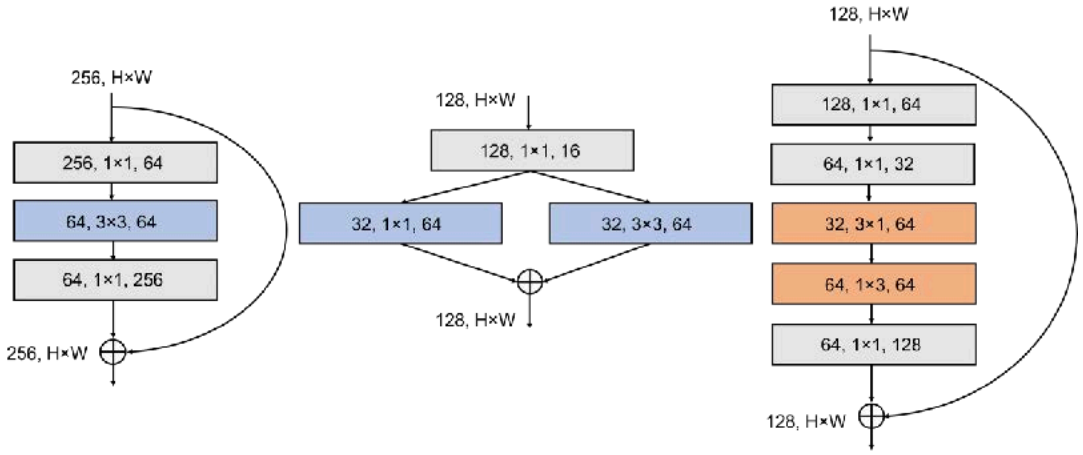
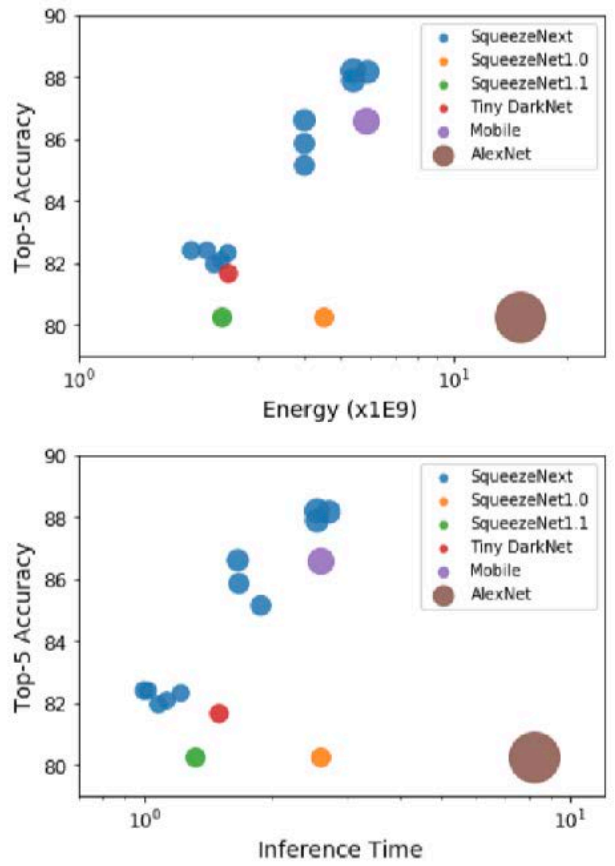


Illustration of a ResNet block on the left, a SqueezeNet block in the middle, and a SqueezeNext (SqNxt) block on the right. SqueezeNext uses a two-stage bottleneck module to reduce the number of input channels to the 3 × 3 convolution. The latter is further decomposed into separable convolutions to further reduce the number of parameters (orange parts), followed by a 1 × 1 expansion module.

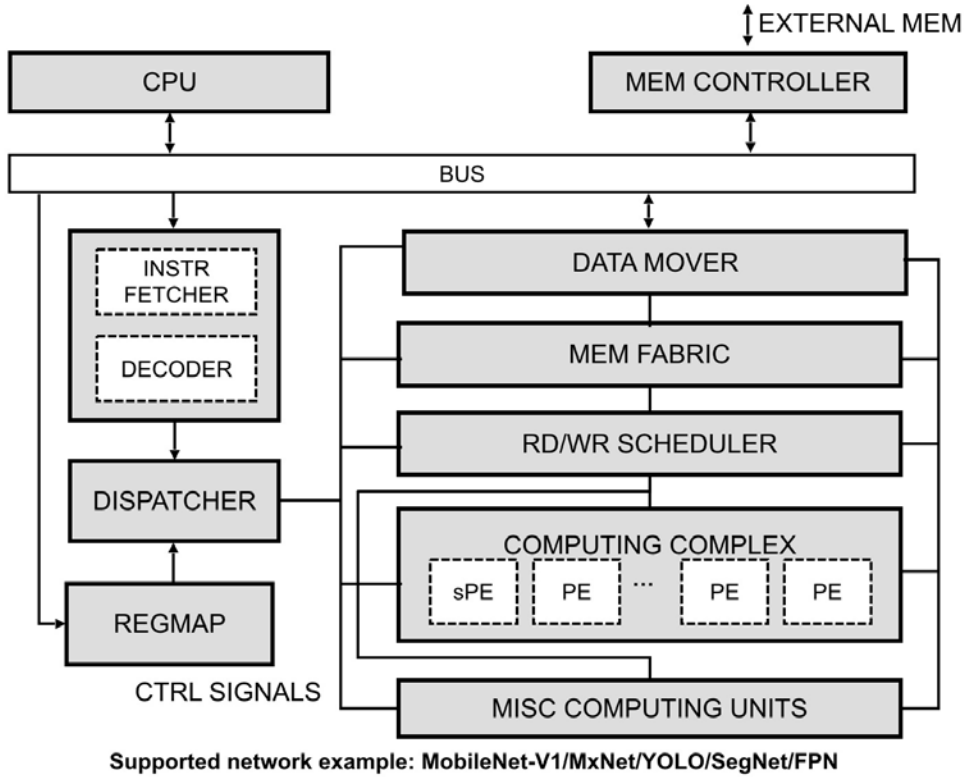


The spectrum of accuracy versus energy and inference speed for SqueezeNext, SqueezeNet (v1.0 and v1.1), Tiny DarkNet, and MobileNet. SqueezeNext shows superior performance (in both plots higher and to the left is better).

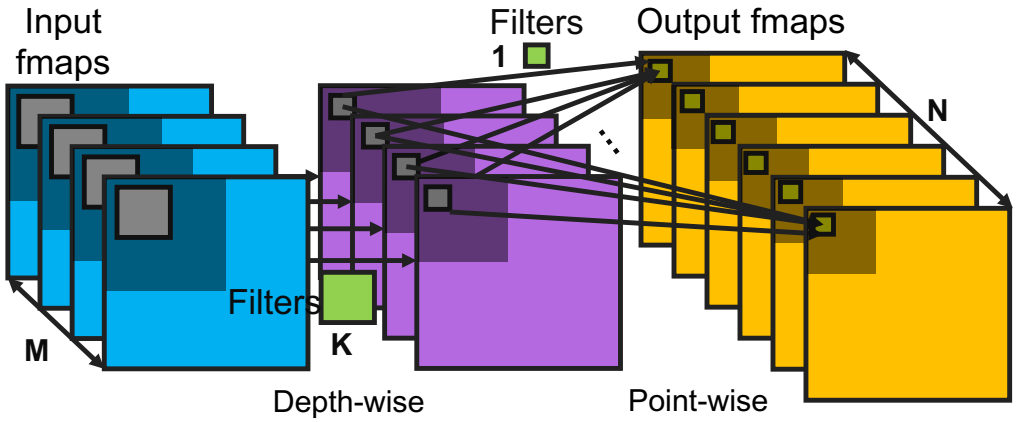
Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, Kurt Keutzer
SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size
Amir Gholami, Kiseok Kwon, Bichen Wu, Zizheng Tai, Xiangyu Yue, Peter Jin, Sicheng Zhao, Kurt Keutzer
SqueezeNext: Hardware-Aware Neural Network Design
Kiseok Kwon, Alon Amid, Amir Gholami, Bichen Wu, Krste Asanovic, Kurt Keutzer
Co-Design of Deep Neural Nets and Neural Net Accelerators for Embedded Vision Applications, DAC'18

Part 4/4: HW Architecture/NN Architecture Co-design

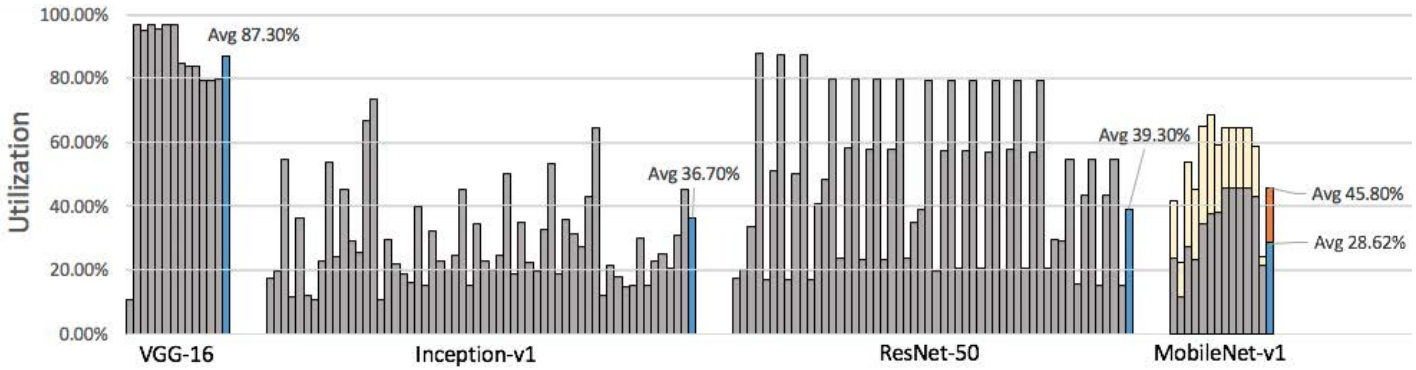
DPU [DeePhi, 2018]



(a) DeePhi's DPU Architecture for MobileNet



(b) Fused DP-convolution and PW-convolution



(c) Result on Xilinx ZU9 FPGA: increased the utilization from 28.6% to 45.8%

Part 4/4: HW Architecture/NN Architecture Co-design

ShiftNet [Wu, 2018]

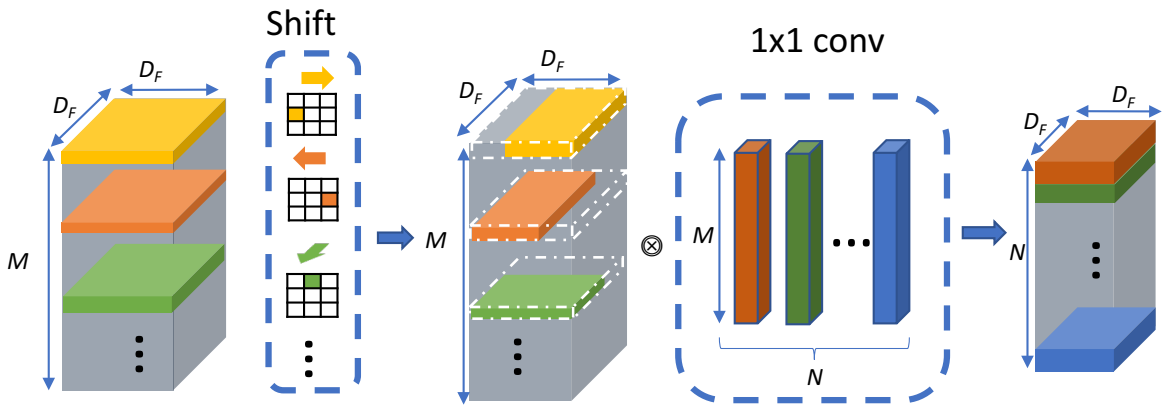
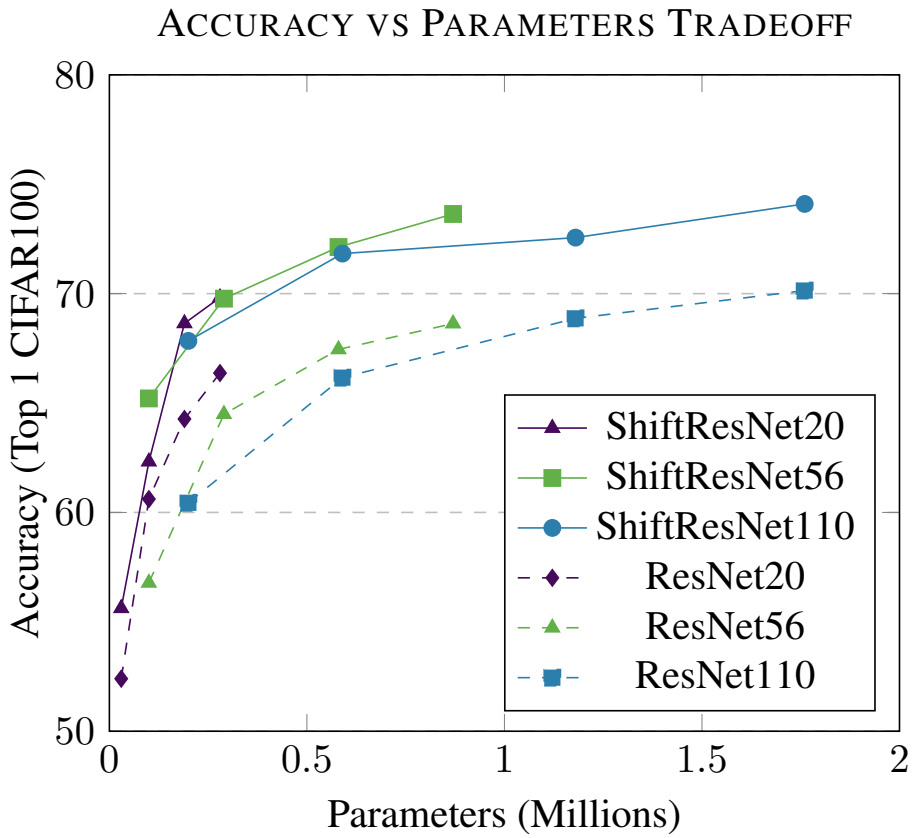


Figure 1: Illustration of a shift operation followed by a 1x1 convolution. The shift operation adjusts data spatially and the 1x1 convolution mixes information across channels.

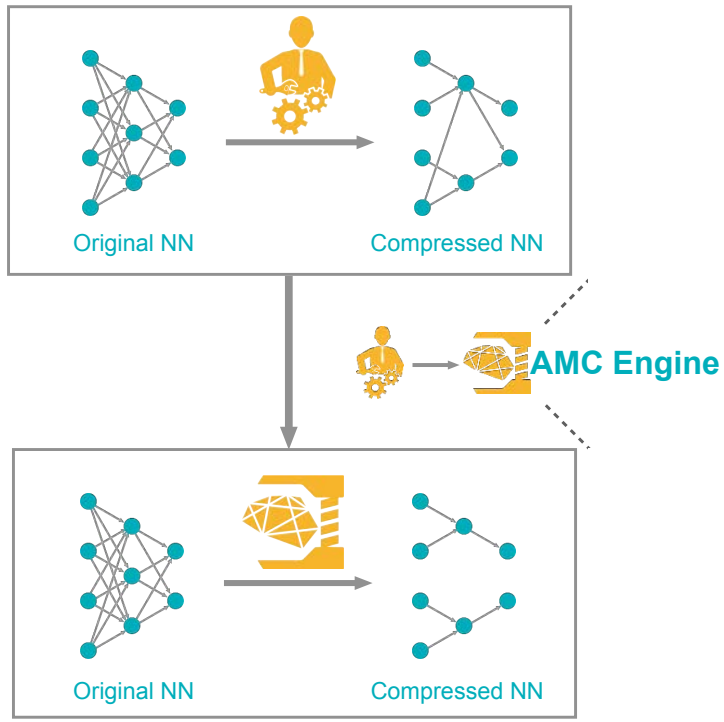


Challenge:
Zero FLOP for 3x3, but ***increased*** #param and #FLOP for 1x1
=> there's no free lunch, but ShiftNet opened up a much larger design space

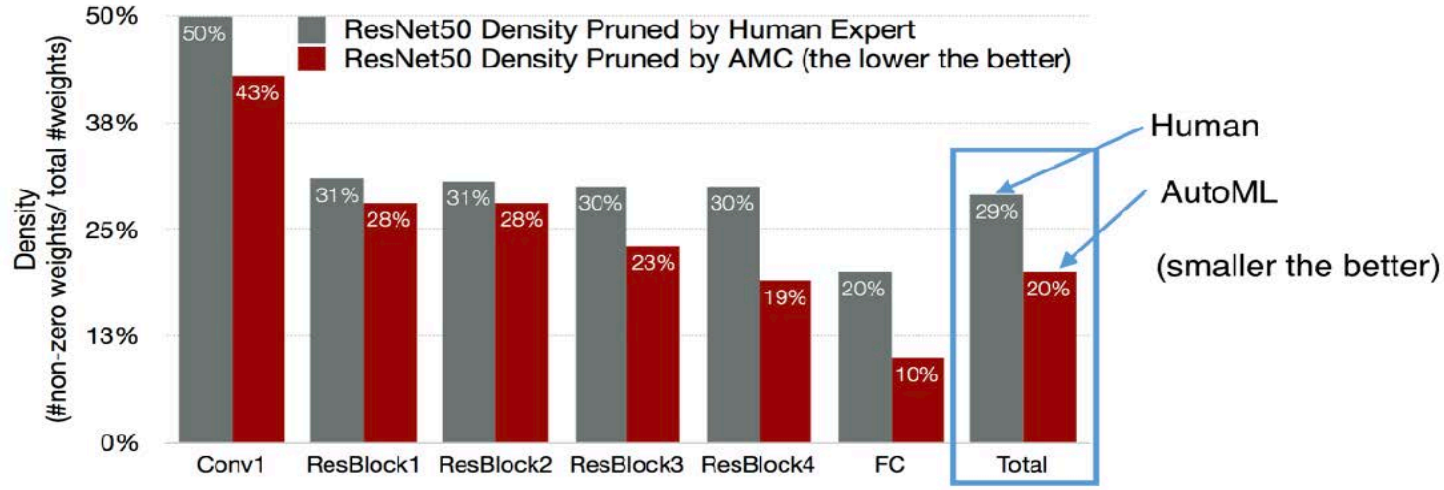
Part 4/4: HW Architecture/NN Architecture Co-design

AMC [He, 2018]

Model Compression by Human:
Labor Consuming, Sub-optimal



Model Compression by AI:
Automated, Higher Compression Rate, Faster

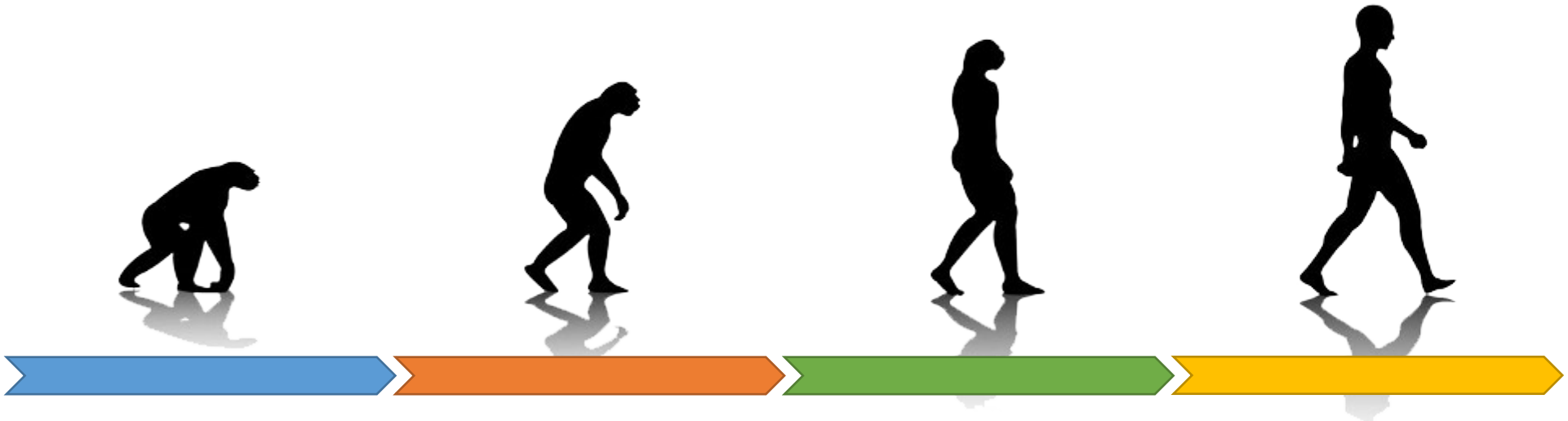


Model	MAC	Top-1	Top-5	Latency	Speed	Memory
1.0 MobileNet	569M	70.6%	89.5%	**119.0ms**	8.4 fps	20.1MB
AMC (50% MAC)	285M	70.5%	89.3%	**64.4ms**	15.5 fps (1.8x)	14.3MB
AMC (50% Time)	272M	70.2%	89.2%	**59.7ms**	16.8 fps (2.0x)	13.2MB
0.75 MobileNet	325M	68.4%	88.2%	**69.5ms**	14.4 fps (1.7x)	14.8MB

AutoML outperforms human

Yihui He, Ji Lin, Zhijian Liu, Hanrui Wang, Jia Li, Song Han
AMC: AutoML for Model Compression and Acceleration on Mobile Devices, ECCV'18

Evolution of NN Accelerators



Computation Specialization

Memory-centric Specialization

Compression before Acceleration

Hardware & NN Architectures Co-Design

Thank you!

Hardware for **A**I and **N**eural-nets