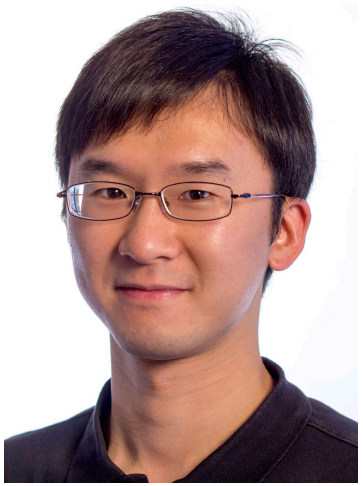
Building Energy-Efficient Accelerators for Deep Learning

Vivienne Sze & Yu-Hsin Chen

Massachusetts
Institute of
Technology

Eyeriss Project



Yu-Hsin Chen

PhD Candidate

MIT

Vivienne Sze

Principal Investigator

MIT

Joel Emer

Principal Investigator

MIT

Senior Distinguished Research Scientist

NVIDIA

Tushar Krishna

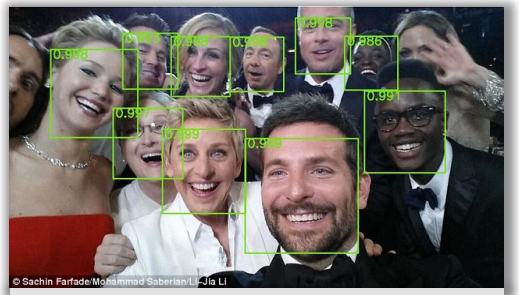
Postdoc

MIT

(currently)
Assistant Professor

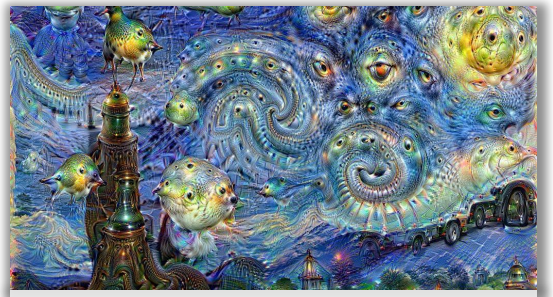
Georgia Tech.

Future of Deep Learning



Recognition

Self-Driving Cars



AI

Computation

- **Deep CNN**
  15k – 300k OP/Px
- **Hand-Crafted Method***
  0.1k – 0.5k OP/Px

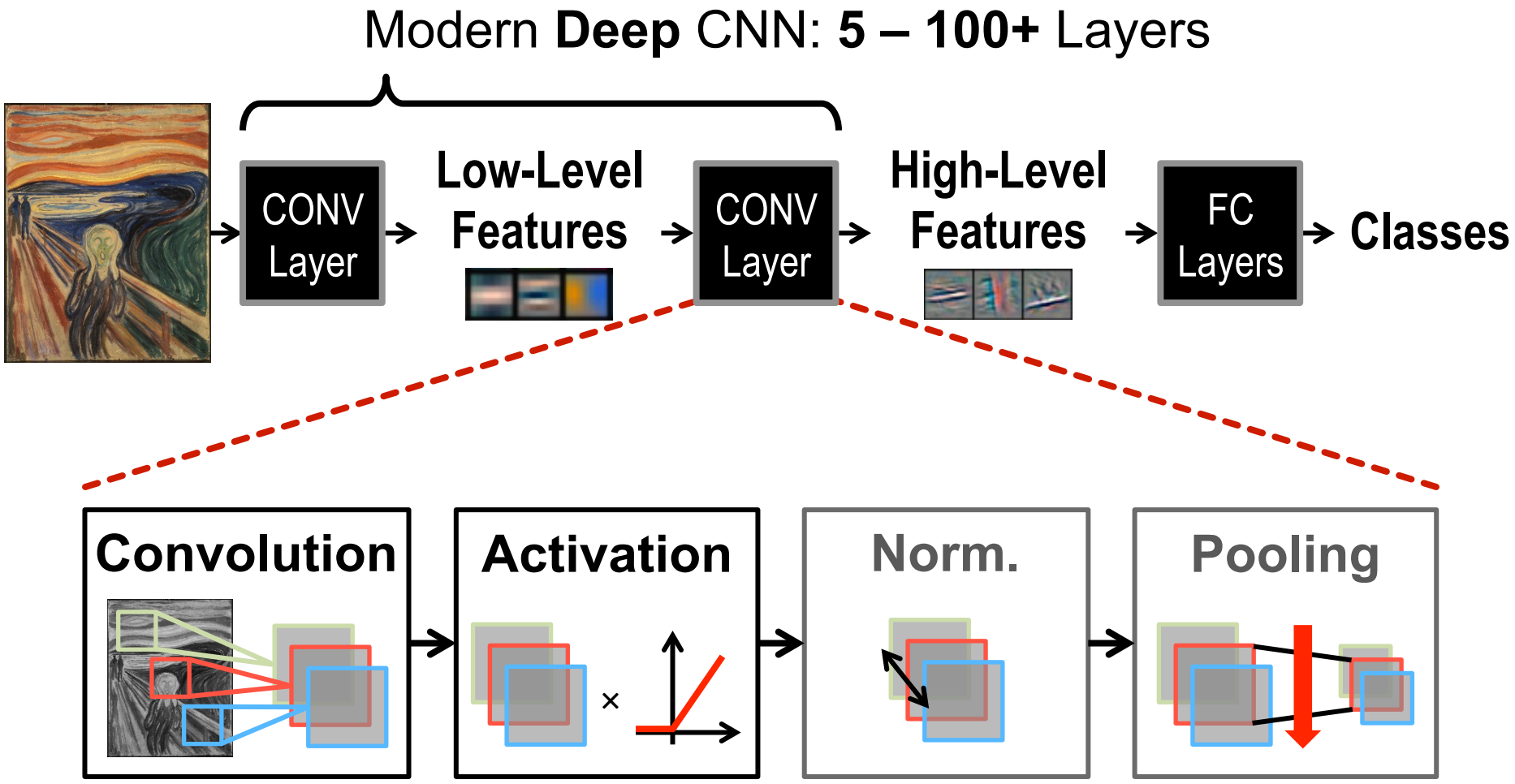
Accuracy

DCNN Accelerator is Crucial

- **High Throughput** for Real-time Processing
- **Sub-watt Power/Energy** Consumption

* Deformable Part Model

Deep CNN Explained

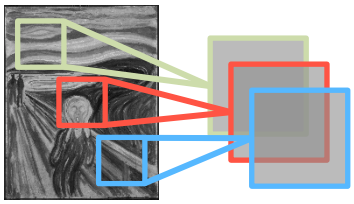


Convolution is the Most Important

Modern **Deep** CNN: **5 – 100+** Layers

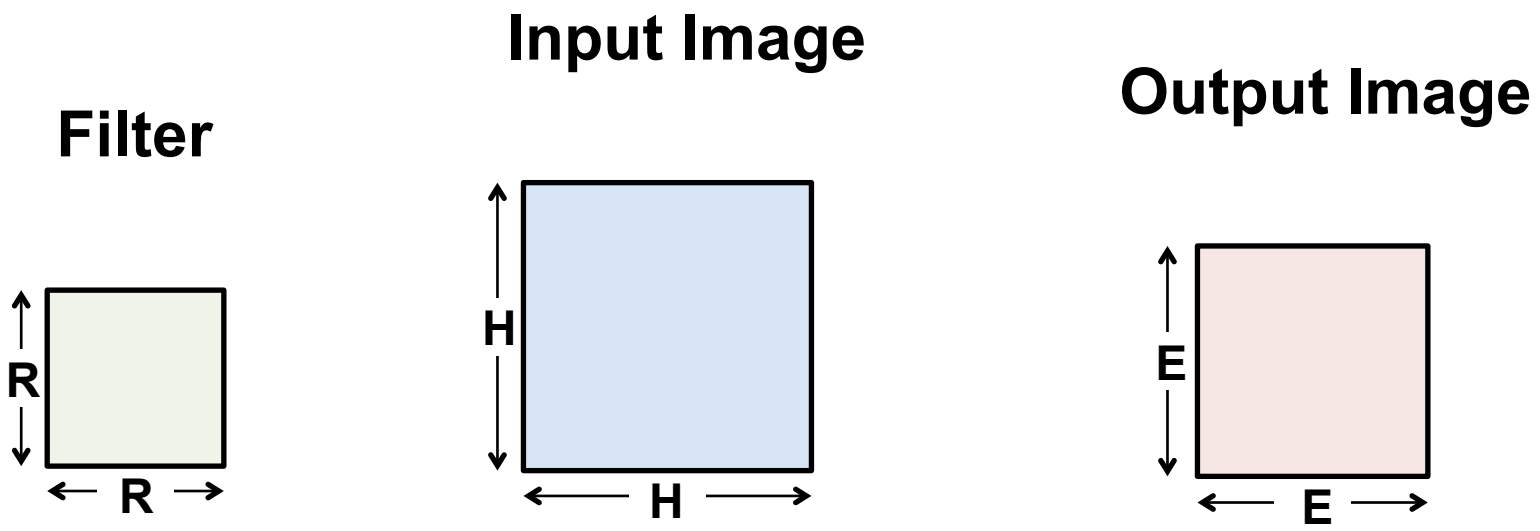
CONV Layer → **Low-Level Features** → CONV Layer → **High-Level Features** → FC Layers → **Classes**

Convolution

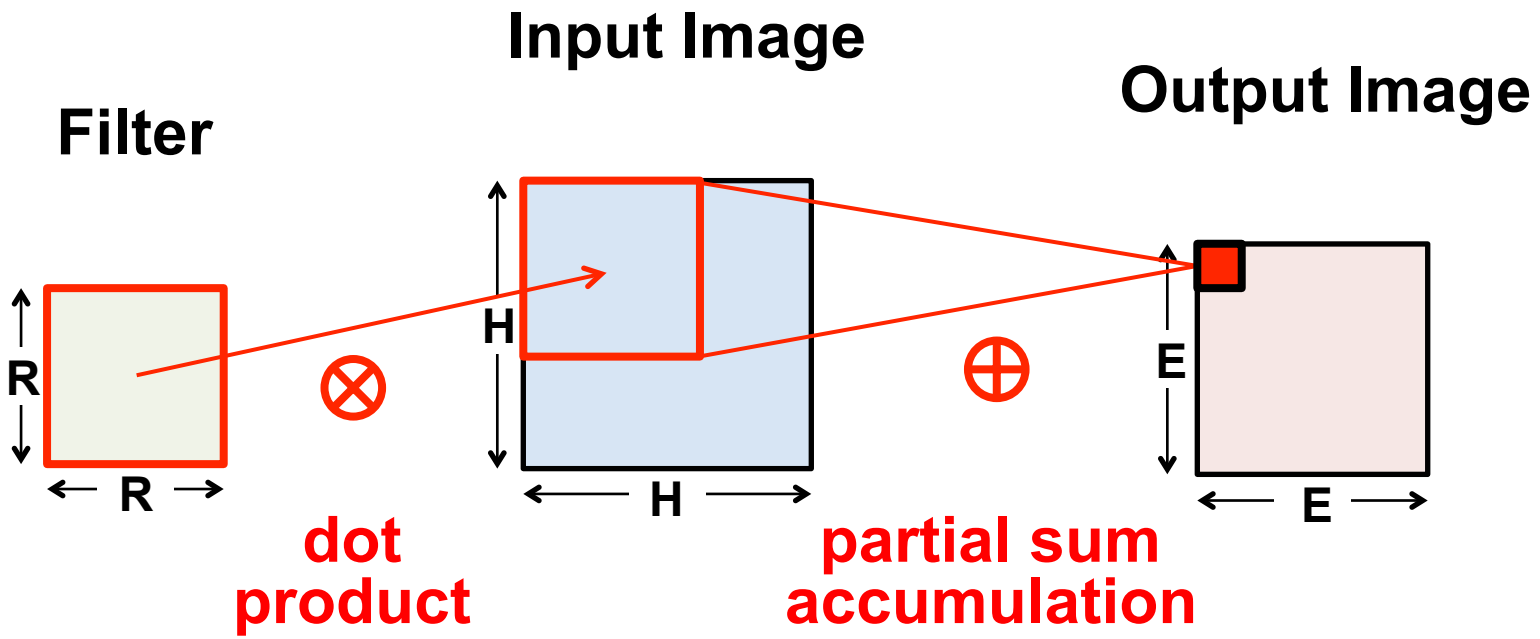


Takes **90% – 99%** of **Computation** and **Runtime**

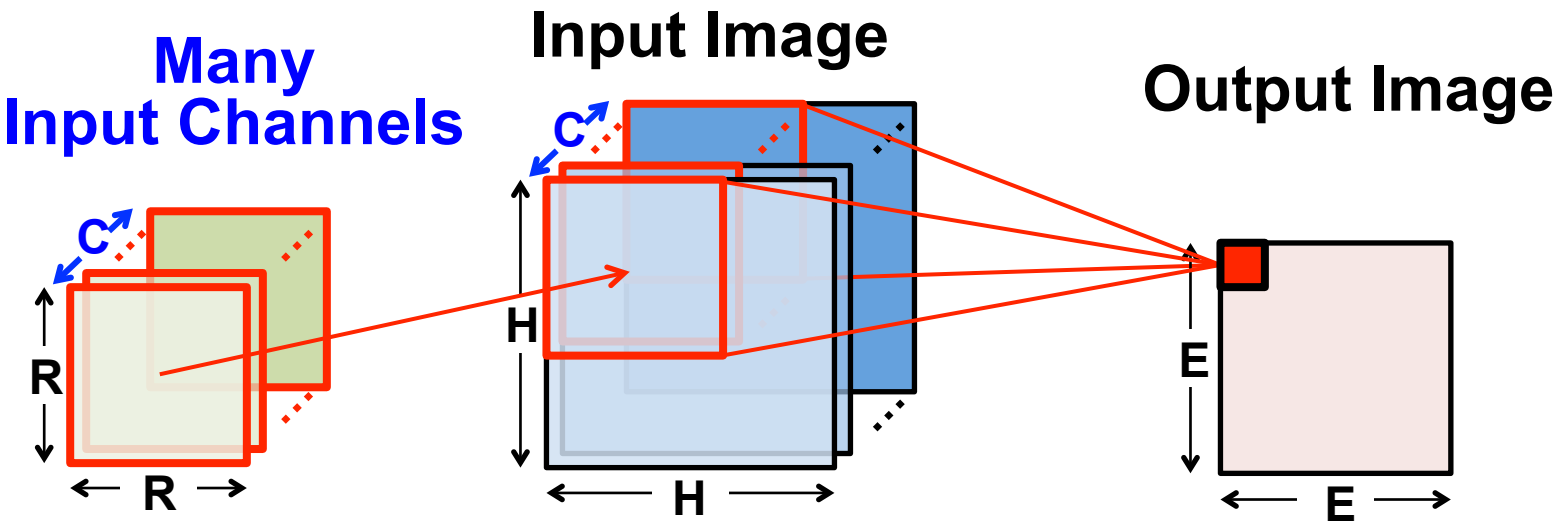
Convolution in CNN



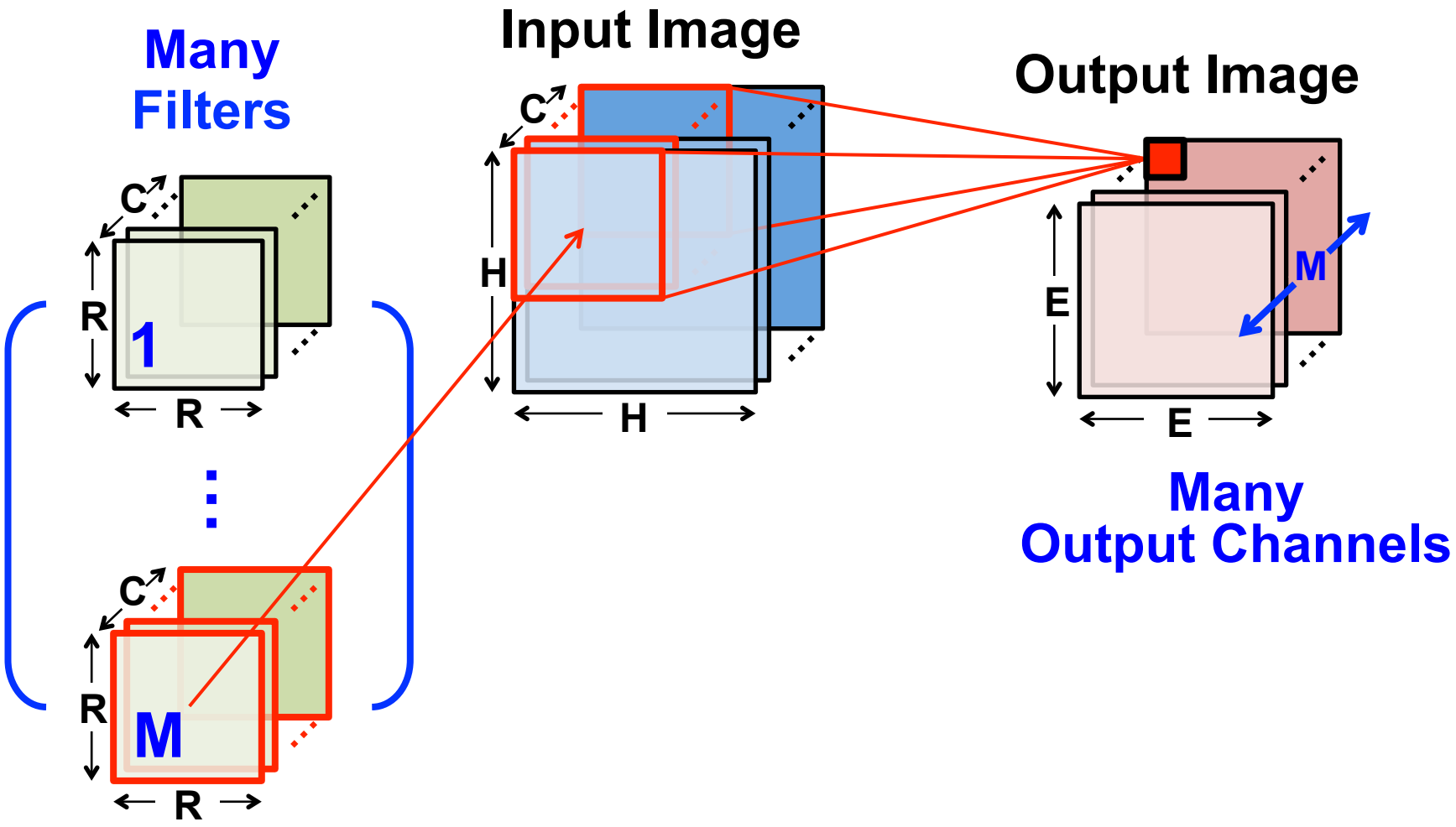
Convolution in CNN



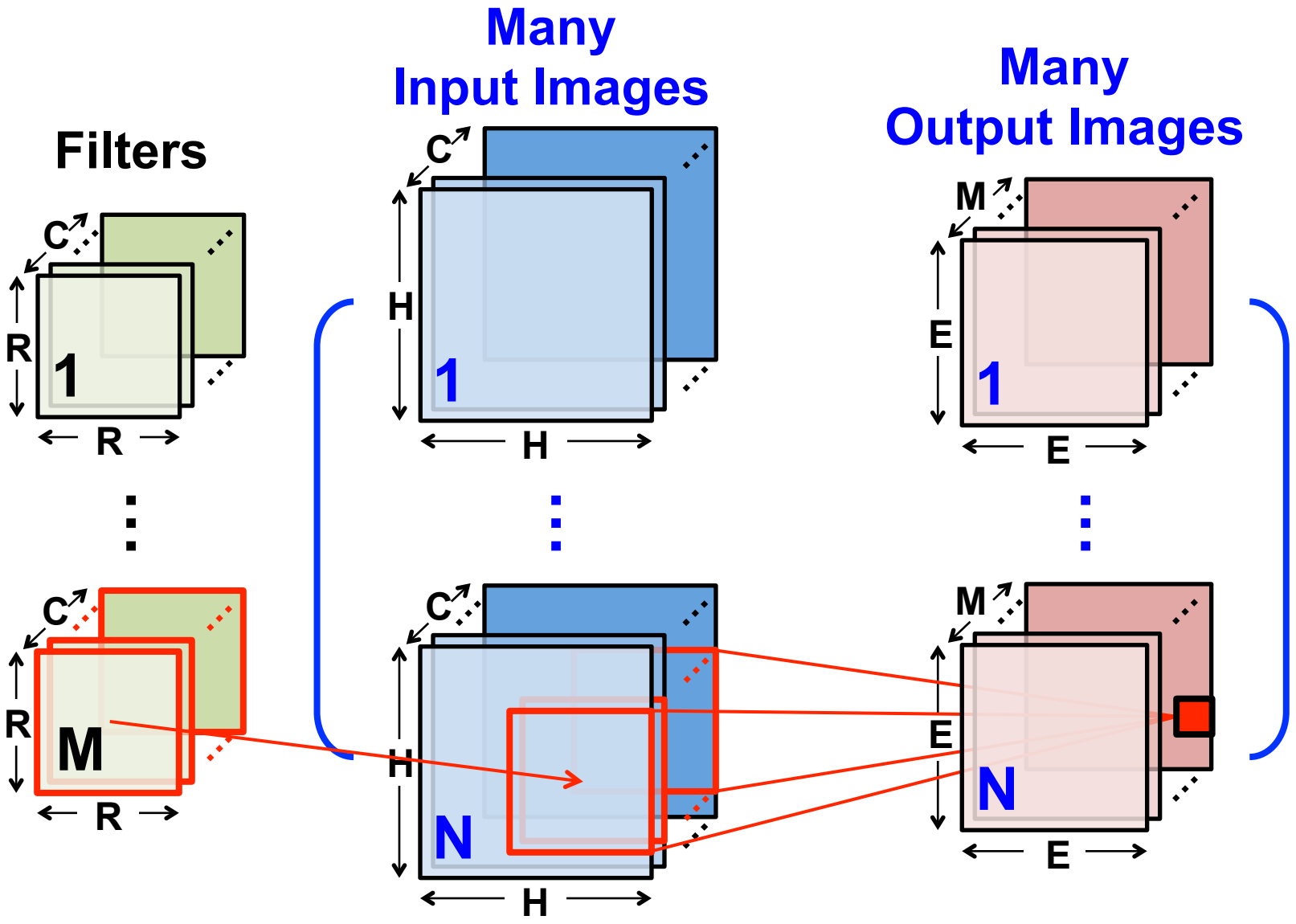
Convolution in CNN



Convolution in CNN



Convolution in CNN

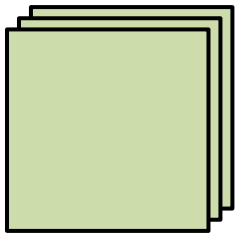


Large Sizes with Varying Shapes

AlexNet[1] Convolutional Layer Configurations

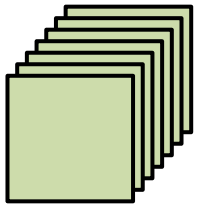
Layer	Filter Size (R)	# Filters (M)	# Channels (C)	Stride
1	11x11	96	3	4
2	5x5	256	48	1
3	3x3	384	256	1
4	3x3	384	192	1
5	3x3	256	192	1

Layer 1



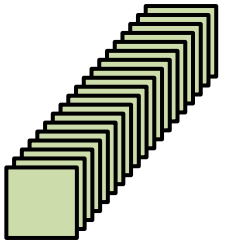
105M MACs*

Layer 2



224M MACs*

Layer 3



150M MACs*

1. Krizhevsky, NIPS 2012

* per frame. MAC = Multiply and Accumulate

Large Sizes with Varying Shapes

AlexNet[1] Convolutional Layer Configurations

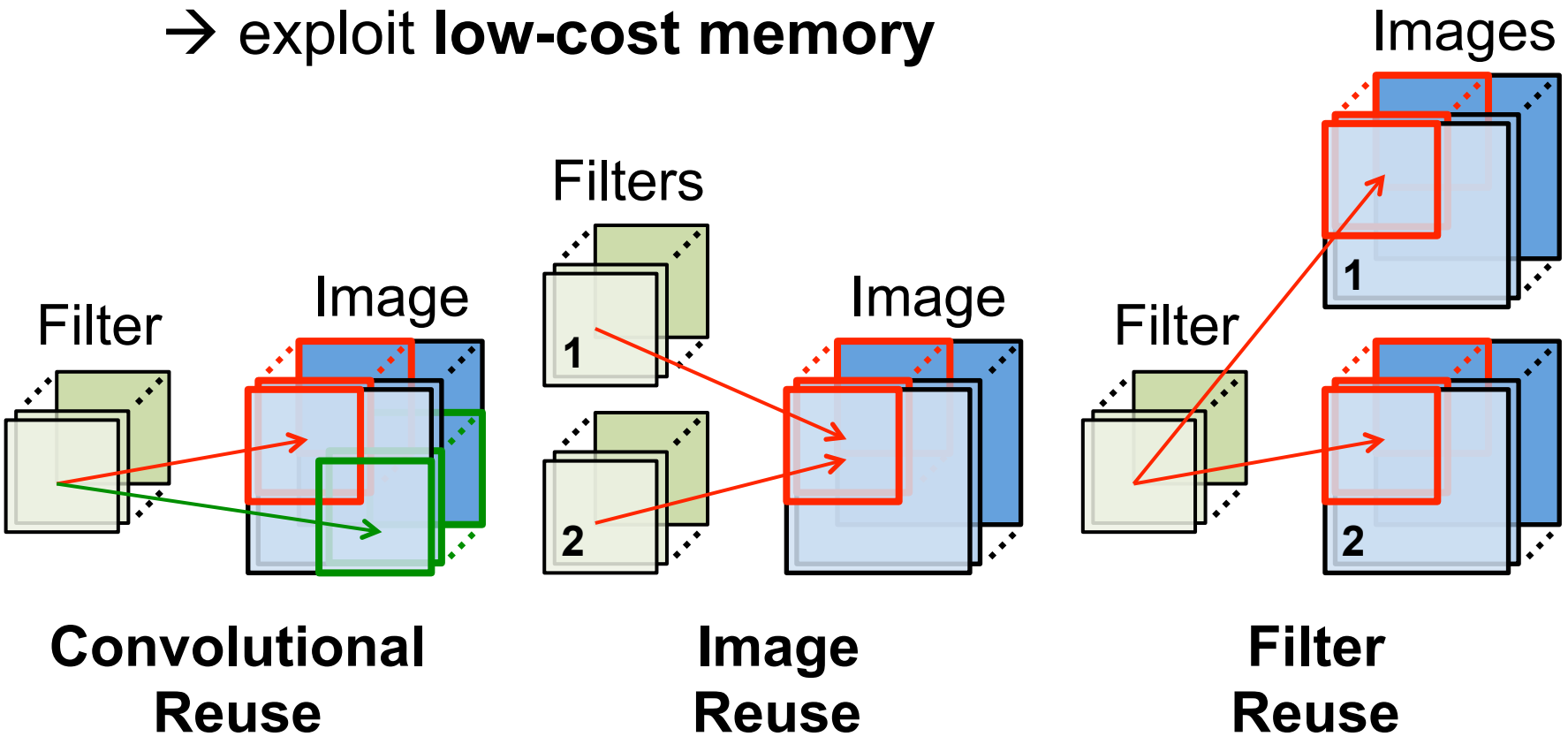
Layer	Filter Size (R)	# Filters (M)	# Channels (C)	Stride
1	11x11	96	3	4
2	5x5	256	48	1
3	3x3	384	256	1
4	3x3	384	192	1
5	3x3	256	192	1

- A large amount of **computation** in each layer
- A large amount of **data accesses to memory**
- **Adaptive processing** required for different shapes

1. Krizhevsky, NIPS 2012

Properties We Can Leverage

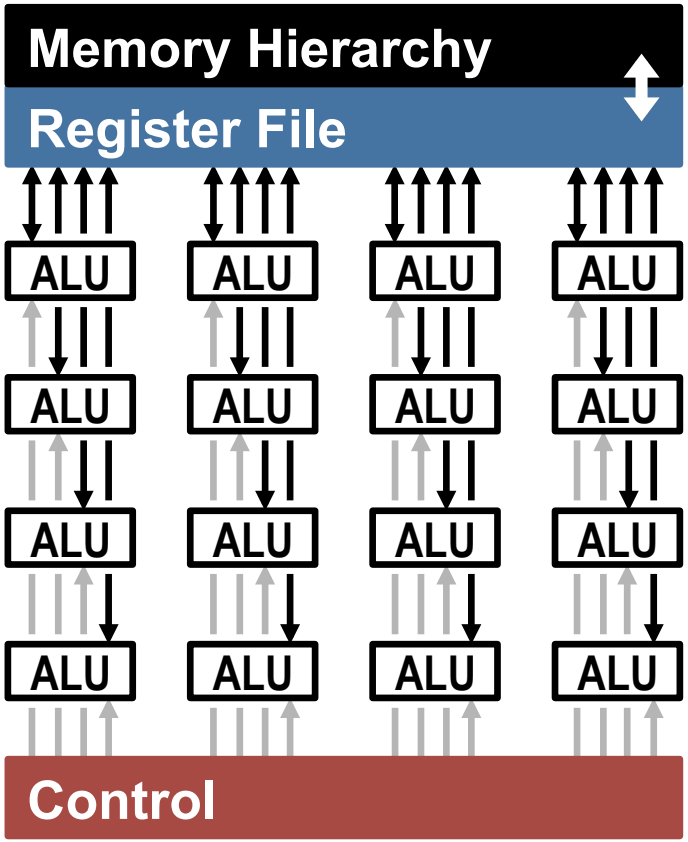
- Operations exhibit **high parallelism**
  → **high throughput** possible
- **Data reuse** opportunities
  → exploit **low-cost memory**



Architecture

Highly Parallel Compute Paradigms

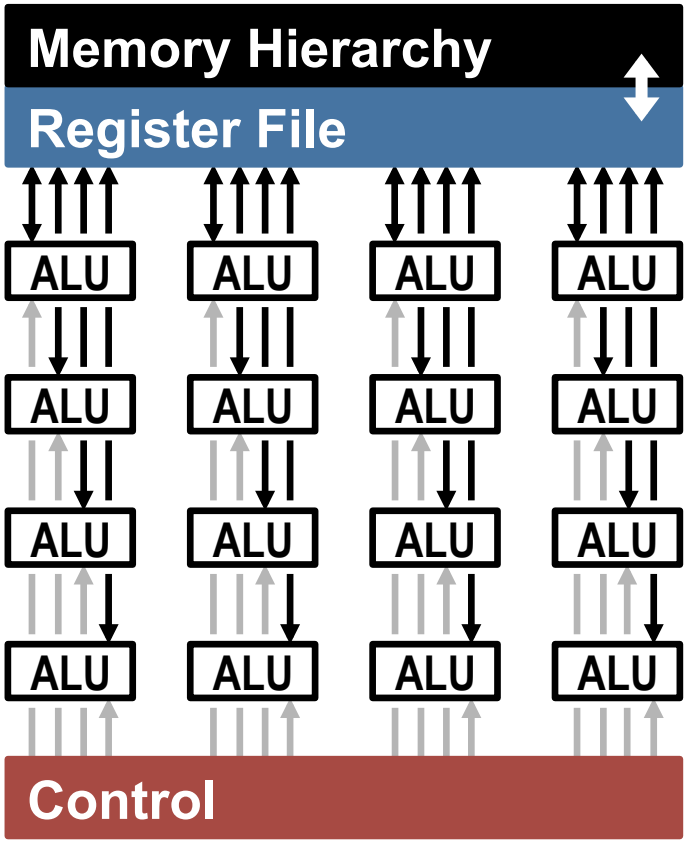
Temporal Architecture (SIMD/SIMT)



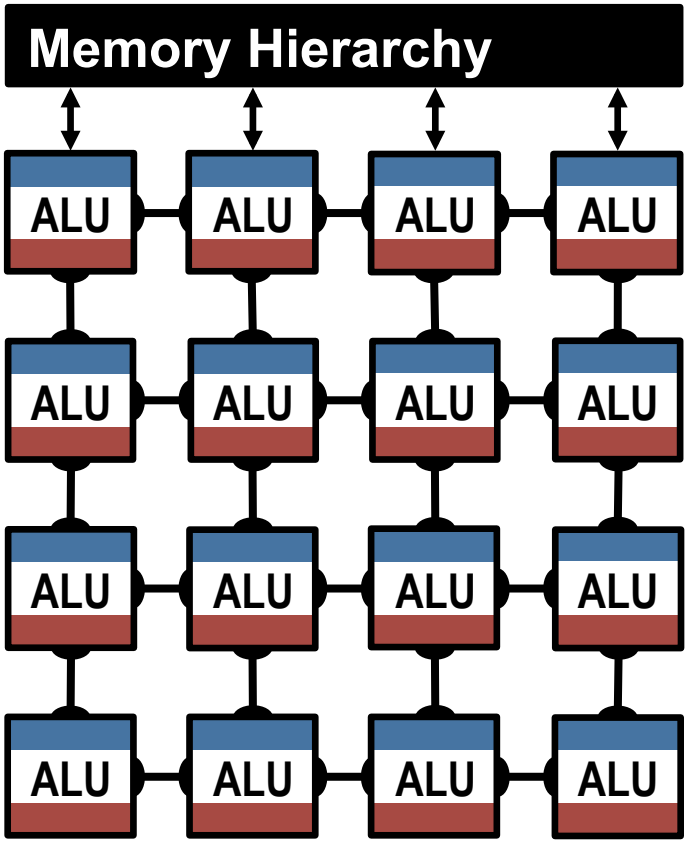
Spatial Architecture (Dataflow Processing)

Highly Parallel Compute Paradigms

Temporal Architecture (SIMD/SIMT)



Spatial Architecture (Dataflow Processing)



Highly Parallel Compute Paradigms

Temporal Architecture
(SIMD/SIMT)

Memory Hierarchy

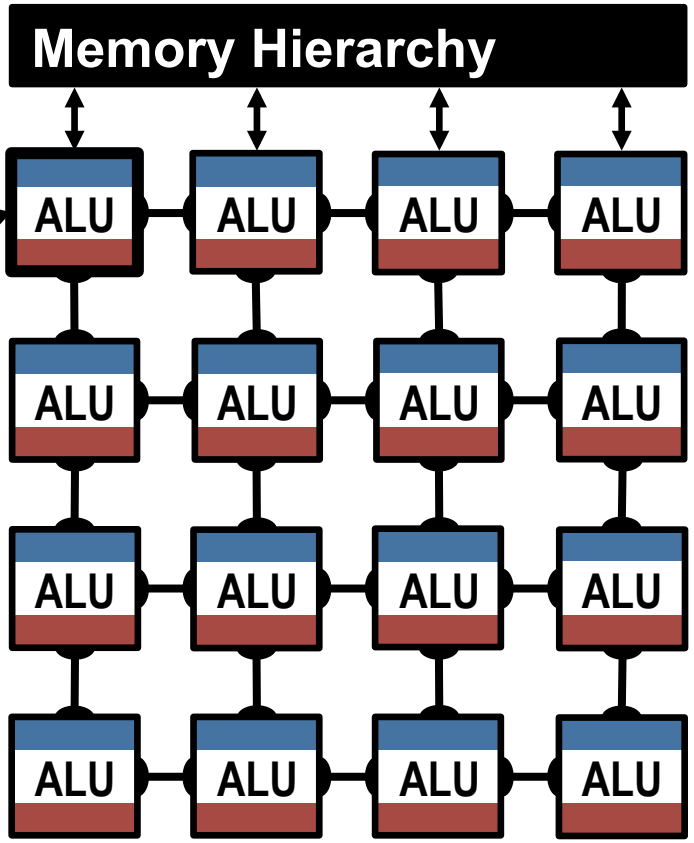
Register File

ALU ALU ALU ALU
ALU ALU ALU ALU
ALU ALU ALU ALU
ALU ALU ALU ALU

Control

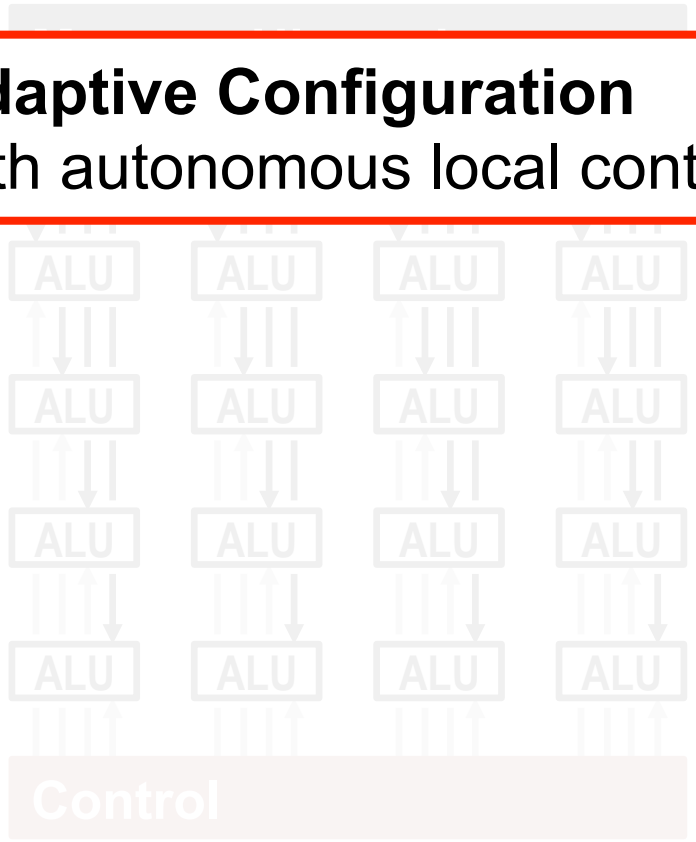
Processing Engine
(PE)

Spatial Architecture
(Dataflow Processing)



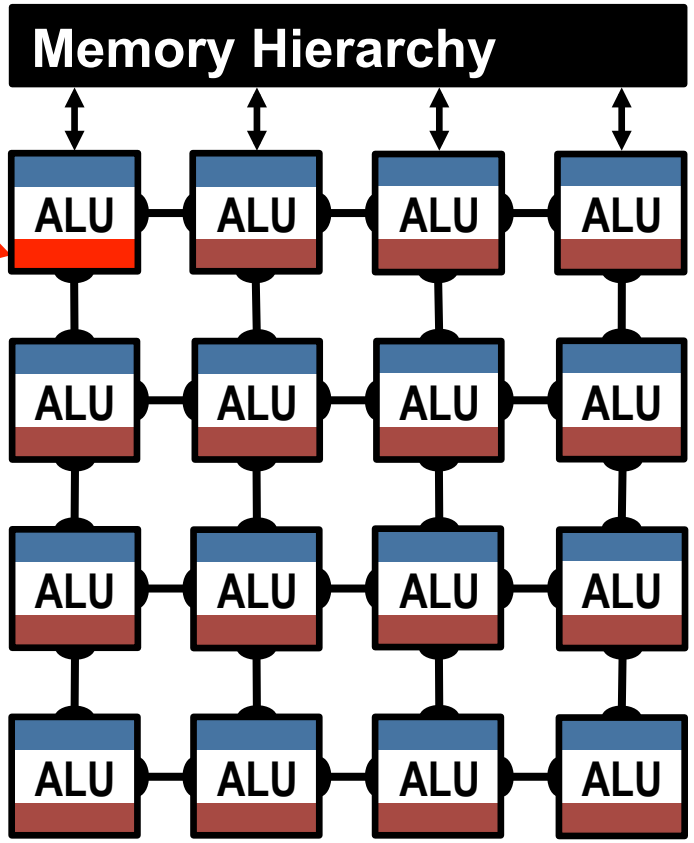
Highly Parallel Compute Paradigms

Temporal Architecture
(SIMD/SIMT)



Adaptive Configuration
with autonomous local control

Spatial Architecture
(Dataflow Processing)

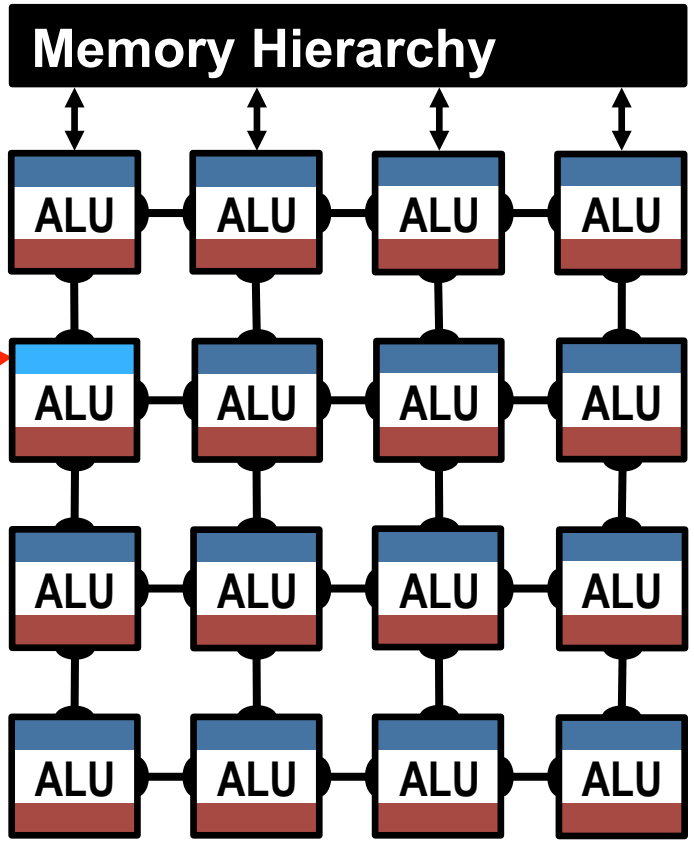


Highly Parallel Compute Paradigms

Adaptive Configuration
with autonomous local control

Efficient Data Reuse
thru. distributed local storage

Spatial Architecture (Dataflow Processing)



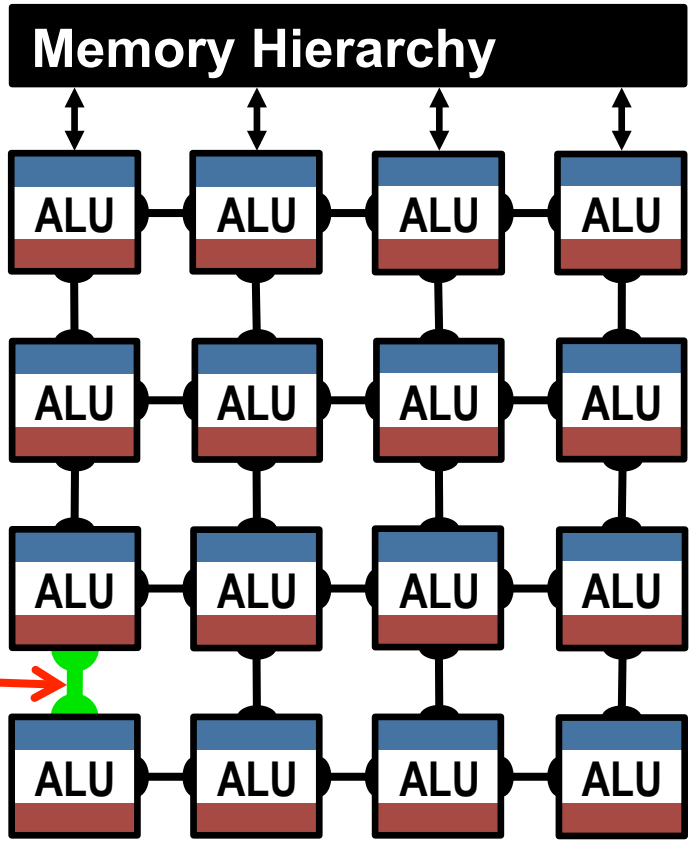
Highly Parallel Compute Paradigms

Adaptive Configuration
with autonomous local control

Efficient Data Reuse
thru. distributed local storage

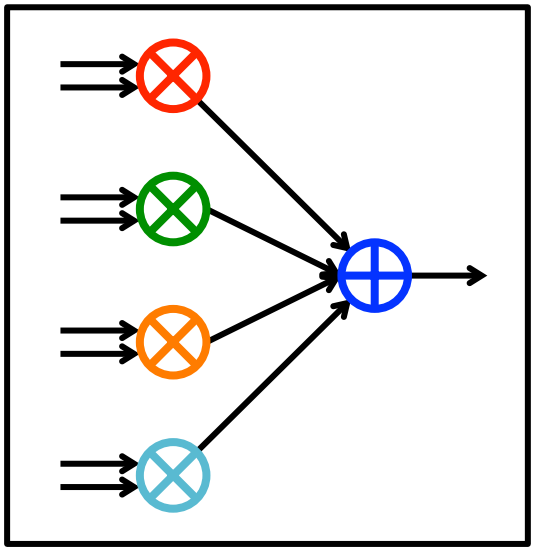
Natural Dataflow Mapping
in-place data consumption

Spatial Architecture (Dataflow Processing)



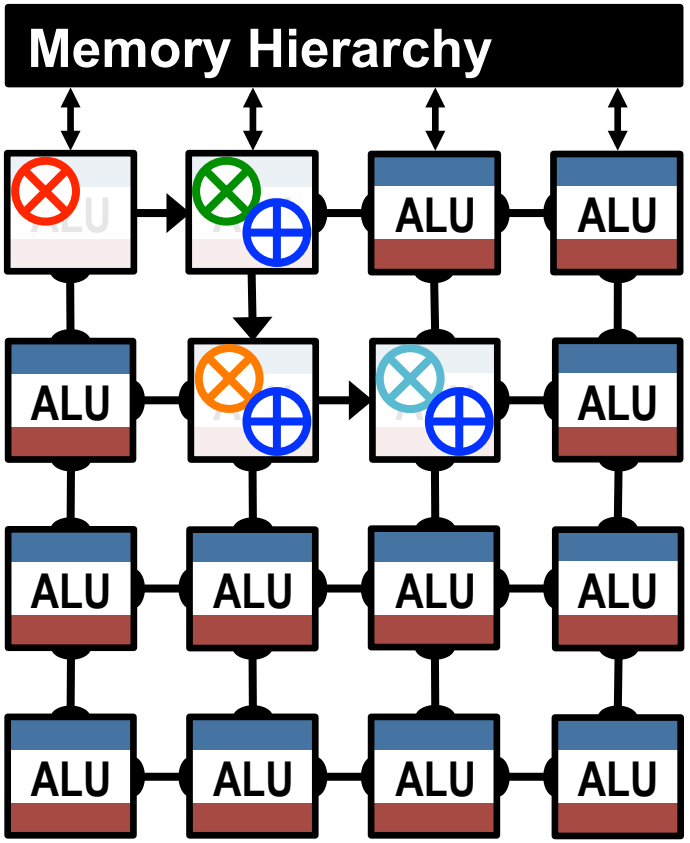
How to Map the Dataflow?

CNN Convolution



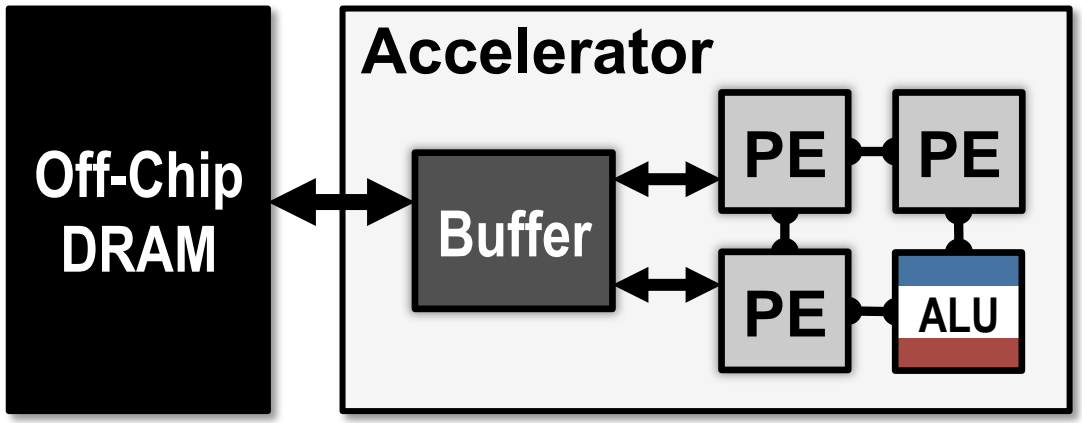
?

Spatial Architecture (Dataflow Processing)

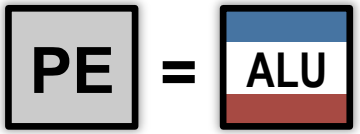


Dataflow Mapping

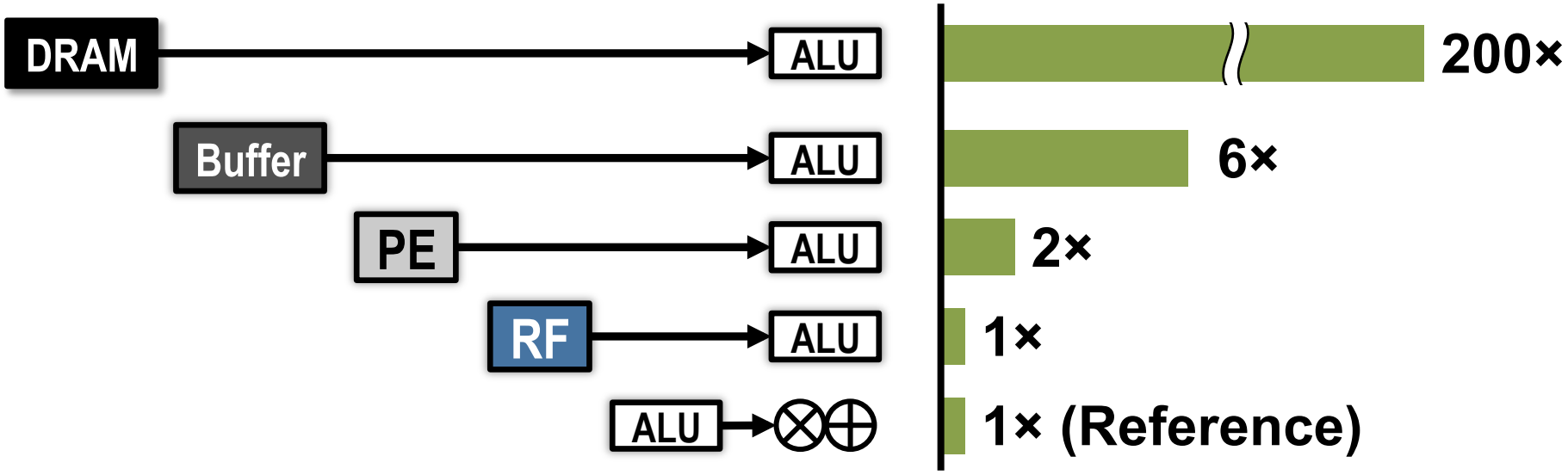
Moving Data is Expensive



Processing Engine

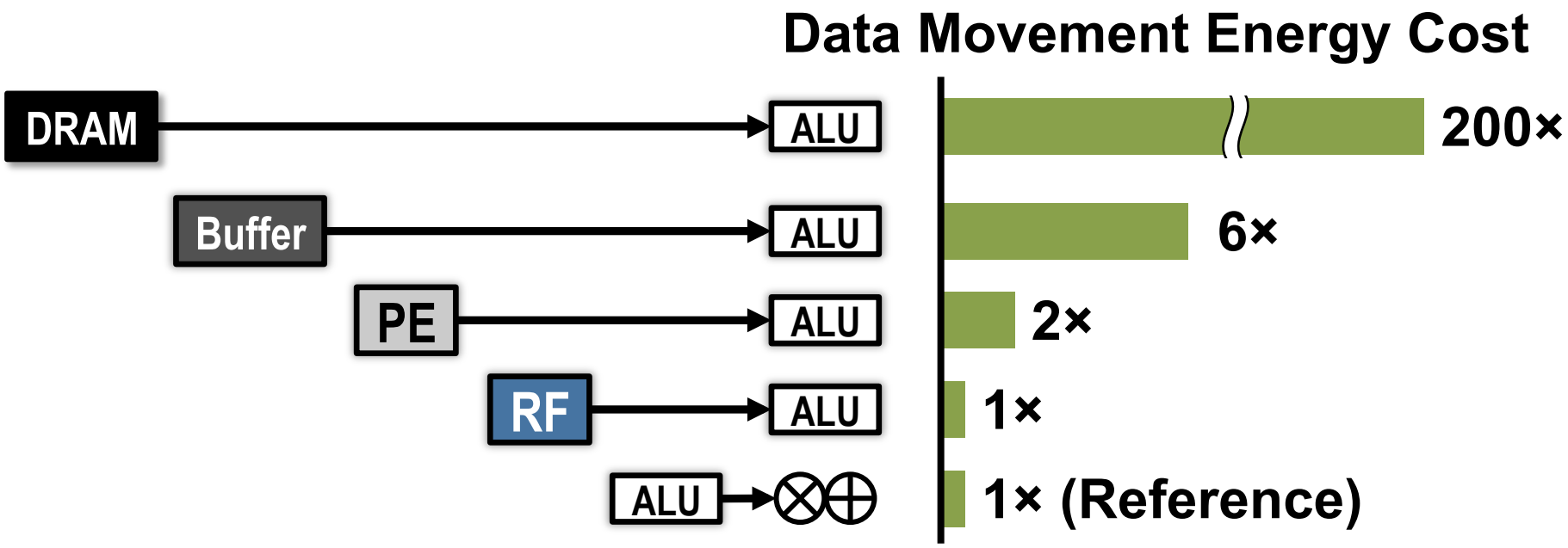


Data Movement Energy Cost

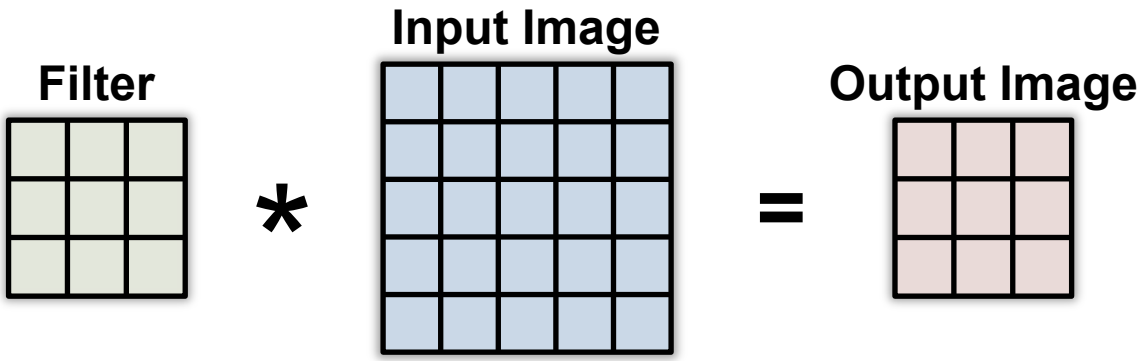


Moving Data is Expensive

- **Reuse input data** (Filter/Image) in local memories
- Keep **partial sum accumulation** local too

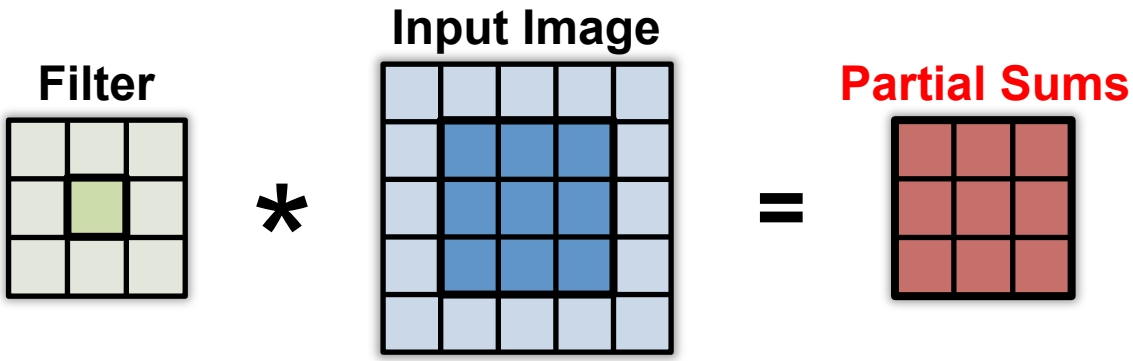


Energy-Efficient CNN Dataflow



Processing **9 MACs** within the same PE

Energy-Efficient CNN Dataflow

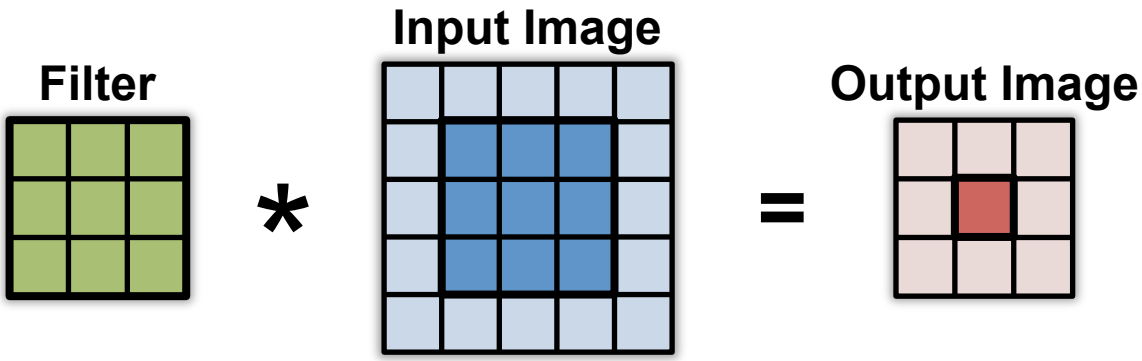


Processing **9 MACs** within the same PE

- **Weight Stationary**: Max **filter weight** reuse

  # Data Touches: 1 + 9 + 9 = **19**

Energy-Efficient CNN Dataflow

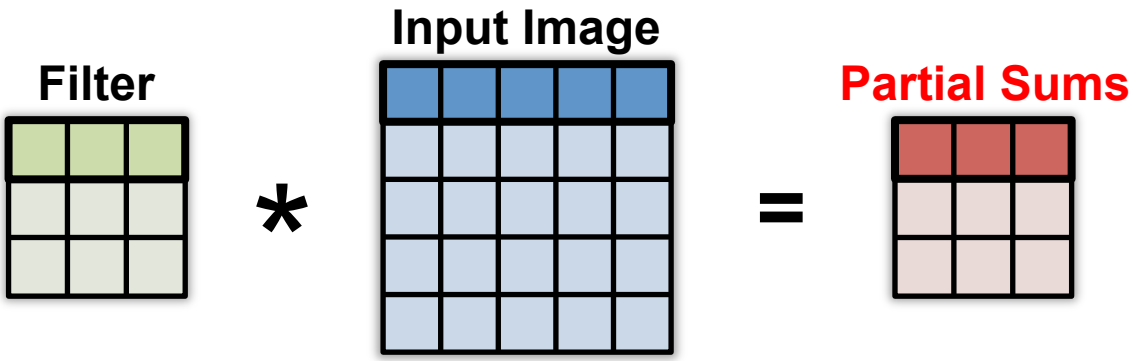


Processing **9 MACs** within the same PE

- **Weight Stationary**: # Data Touches = **19**
- **Output Stationary**: Max **partial sum** accumulation

  # Data Touches: 9 + 9 + 1 = **19**

Energy-Efficient CNN Dataflow

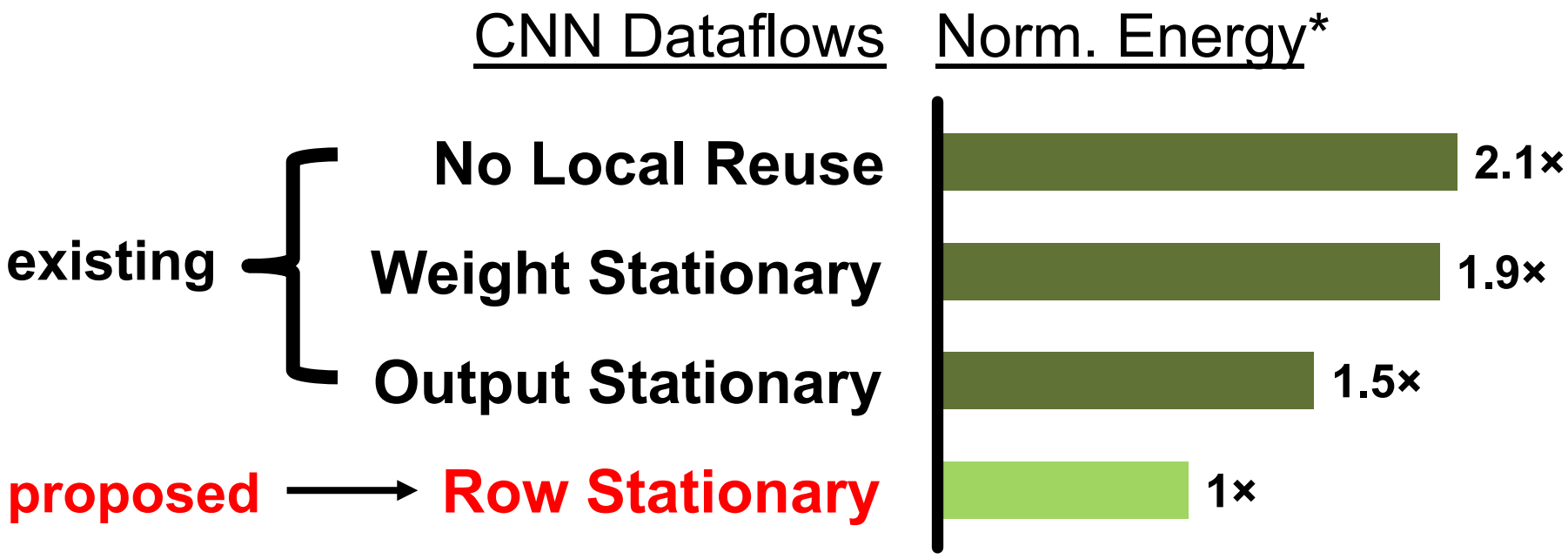


Processing **9 MACs** within the same PE

- **Weight Stationary**: # Data Touches = **19**
- **Output Stationary**: # Data Touches = **19**
- **Row Stationary**: **balance** reuse of **all data types**

  # Data Touches: 3 + 5 + 3 = **11**

CNN Dataflows Comparison[1]



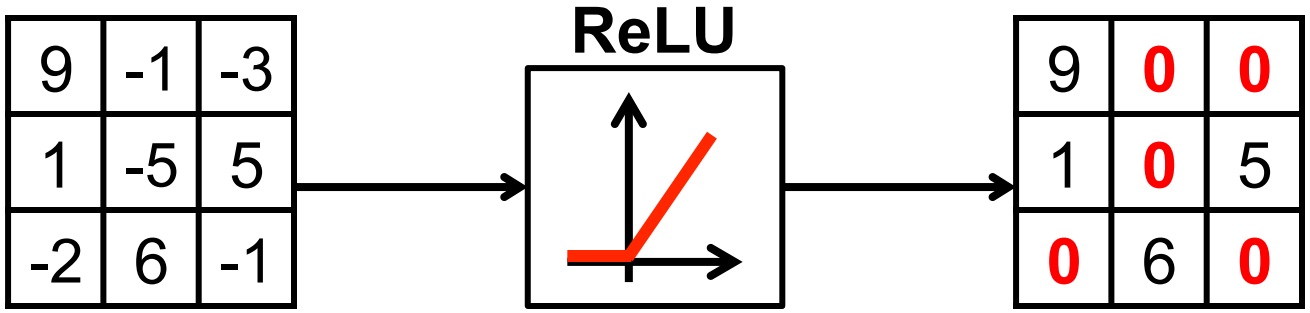
* 256 PEs with the same total memory area

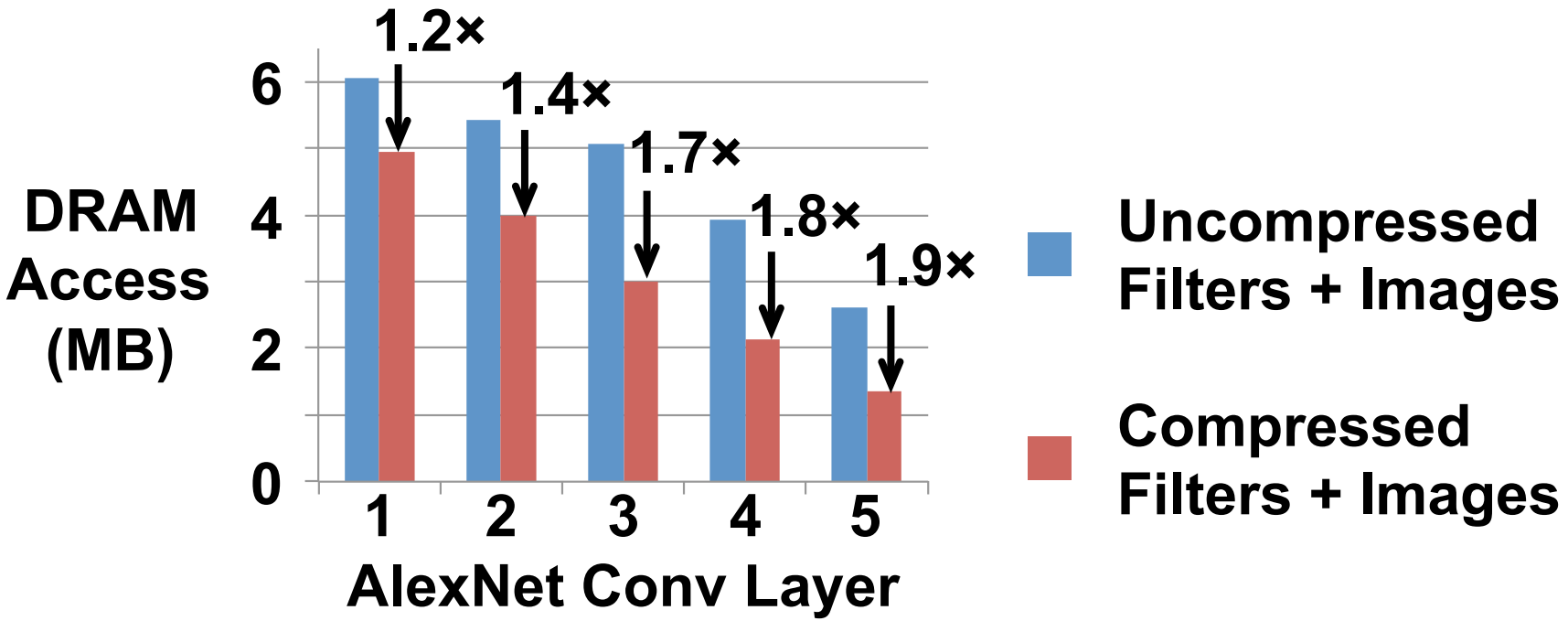
1. Yu-Hsin Chen, Joel Emer and Vivienne Sze, "Eyeriss: A Spatial Architecture for Energy-Efficient Dataflow for Convolutional Neural Networks," *ISCA 2016*

Exploit Data Statistics

Zero Compression Saves DRAM BW

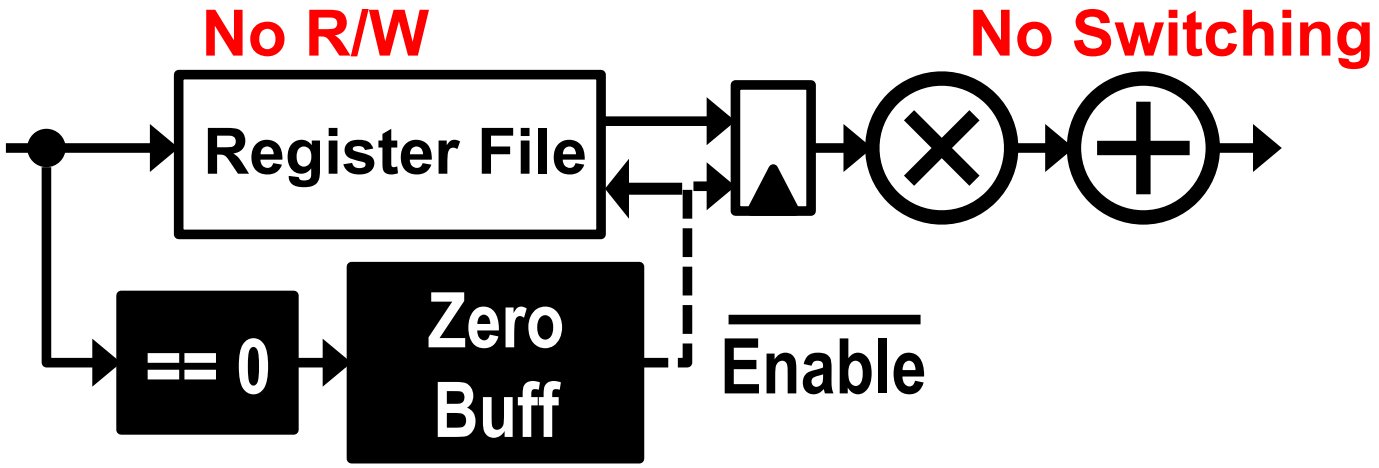
Apply Activation (**ReLU**) on Filtered Image Data





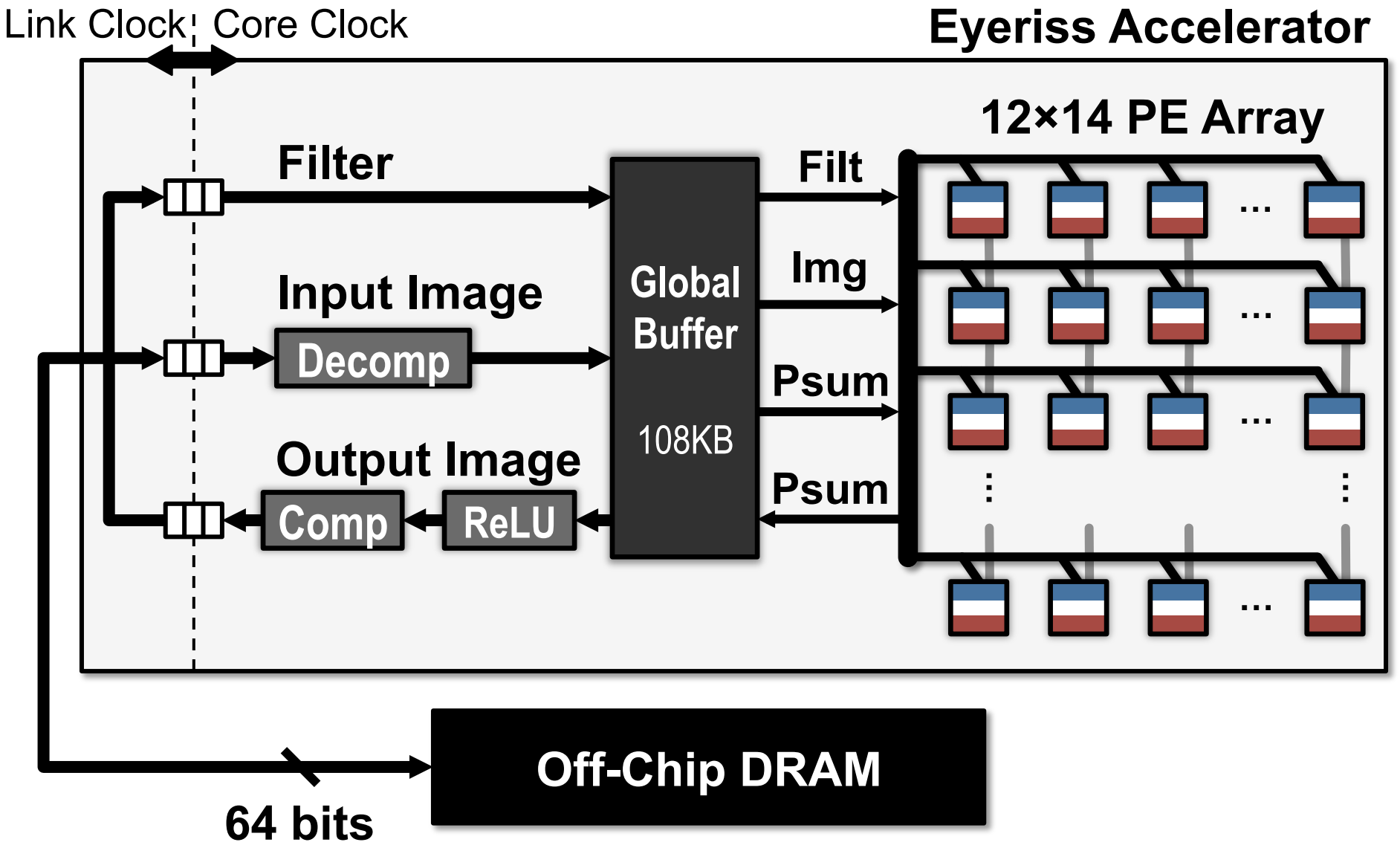
Zero Data Processing Gating

- Skip PE local **memory access**
- Skip MAC **computation**
- Save PE processing power by 45%



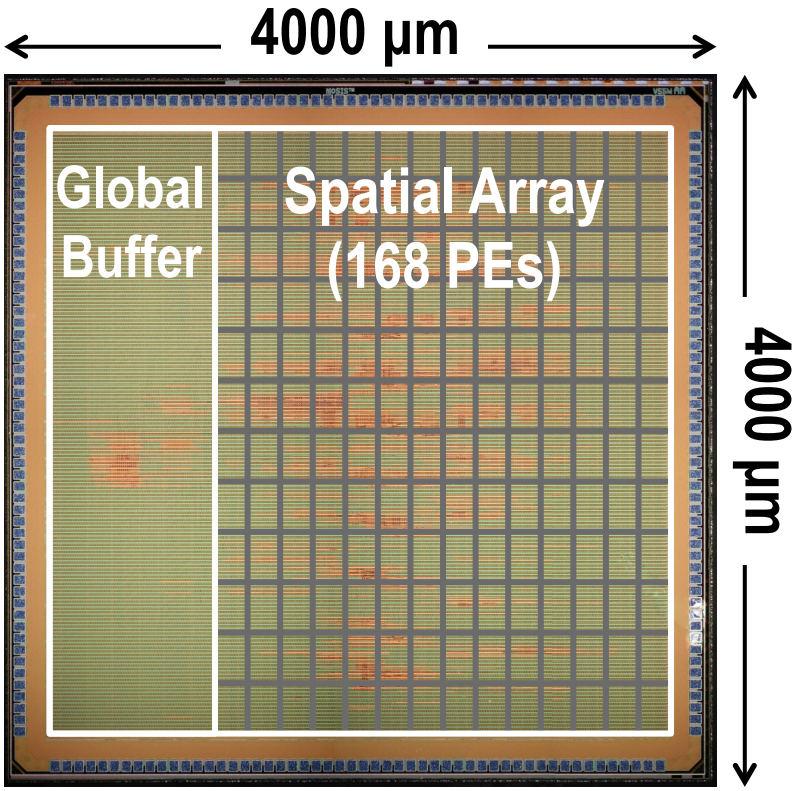
Eyeriss Accelerator

Eyeriss DCNN Accelerator System



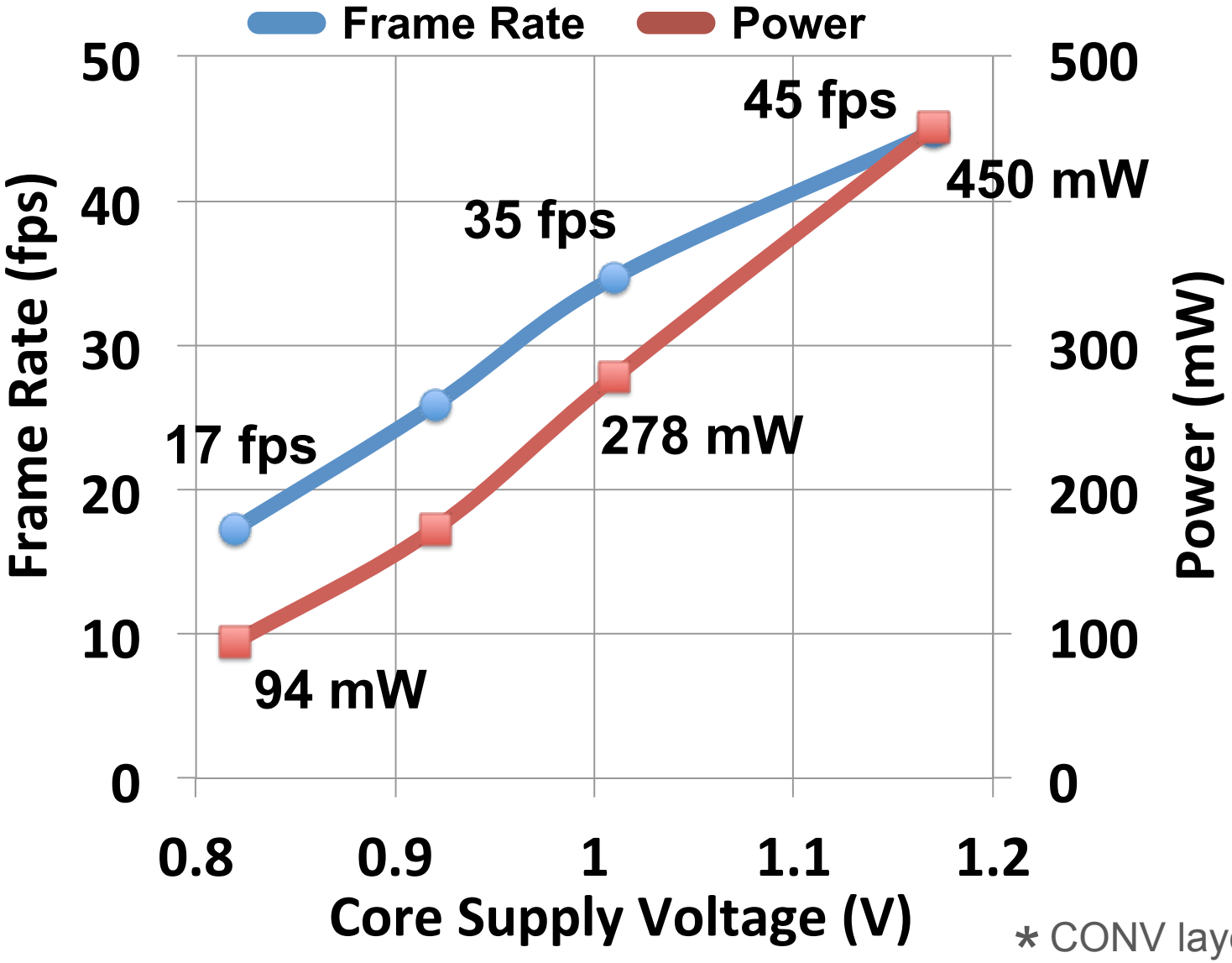
Chip Spec & Measurement Results[1]

Technology	TSMC 65nm LP 1P9M
On-Chip Buffer	108 KB
# of PEs	168
Scratch Pad / PE	0.5 KB
Core Frequency	100 – 250 MHz
Peak Performance	33.6 – 84.0 GOPS
Word Bit-width	16-bit Fixed-Point
Natively Supported CNN Shapes	Filter Width: 1 – 32 Filter Height: 1 – 12 Num. Filters: 1 – 1024 Num. Channels: 1 – 1024 Horz. Stride: 1–12 Vert. Stride: 1, 2, 4



1. Yu-Hsin Chen, Tushar Krishna, Joel Emer and Vivienne Sze, "Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks," *ISSCC 2016*

AlexNet* Throughput vs. Power



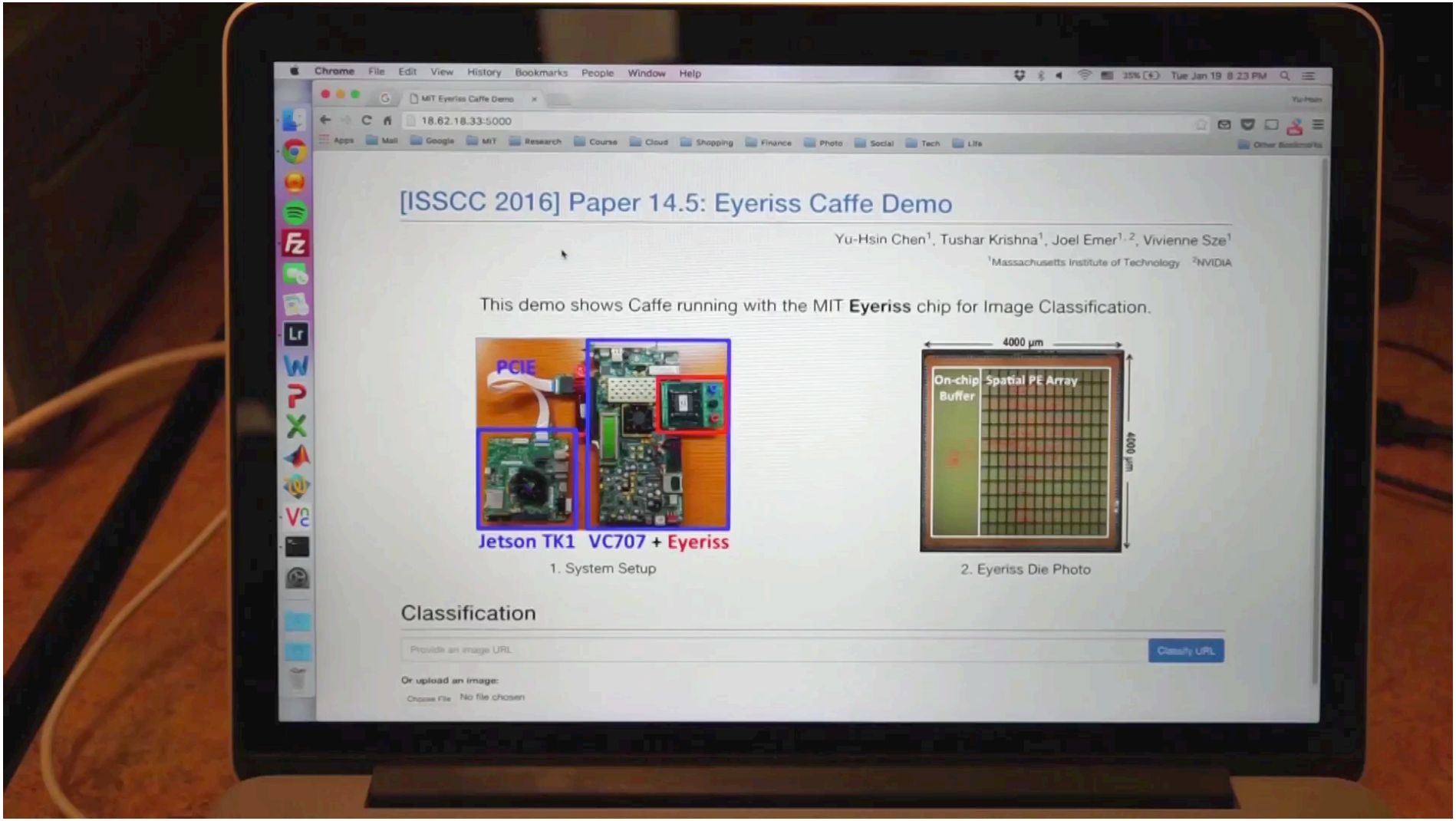
* CONV layers only

Comparison with GPU

	Eyeriss	NVIDIA TK1 (Jetson Kit)
Technology	65nm	28nm
Clock Rate	200MHz	852MHz
# Multipliers	168	192
On-Chip Storage	Buffer: 108KB Spad: 75.3KB	Shared Mem: 64KB Reg File: 256KB
Word Bit-Width	16b Fixed	32b Float
Throughput[1]	34.7 fps	68 fps
Measured Power	278 mW	Idle/Active[2]: 3.7W/10.2W

1. AlexNet Convolutional Layers
2. Board Power

Image Classification on Eyeriss



AlexNet: Krizhevsky, NIPS 2012

Summary

- **Eyeriss**: a **reconfigurable** accelerator for state-of-the-art deep CNNs at **below 300mW**
- **Reduce data movement** & **exploit data statistics** to achieve high energy efficiency
- **Integrated** with the **Caffe DL framework** and demonstrated an image classification system

Want to know more?
Visit our project page:

Acknowledgement: funded by DARPA YFA, MIT CICS and a gift from Intel