Mixed-Signal Techniques for Embedded Machine Learning Systems

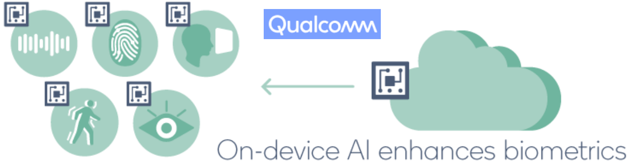
Boris Murmann
June 16, 2019

SystemX Alliance
Stanford University

Stanford ENGINEERING
Electrical Engineering

murmann mixed|signal

Edge ⟵ Applications ⟶ Cloud



Conversational Interfaces
…natural?
…seamless?
…real-time?

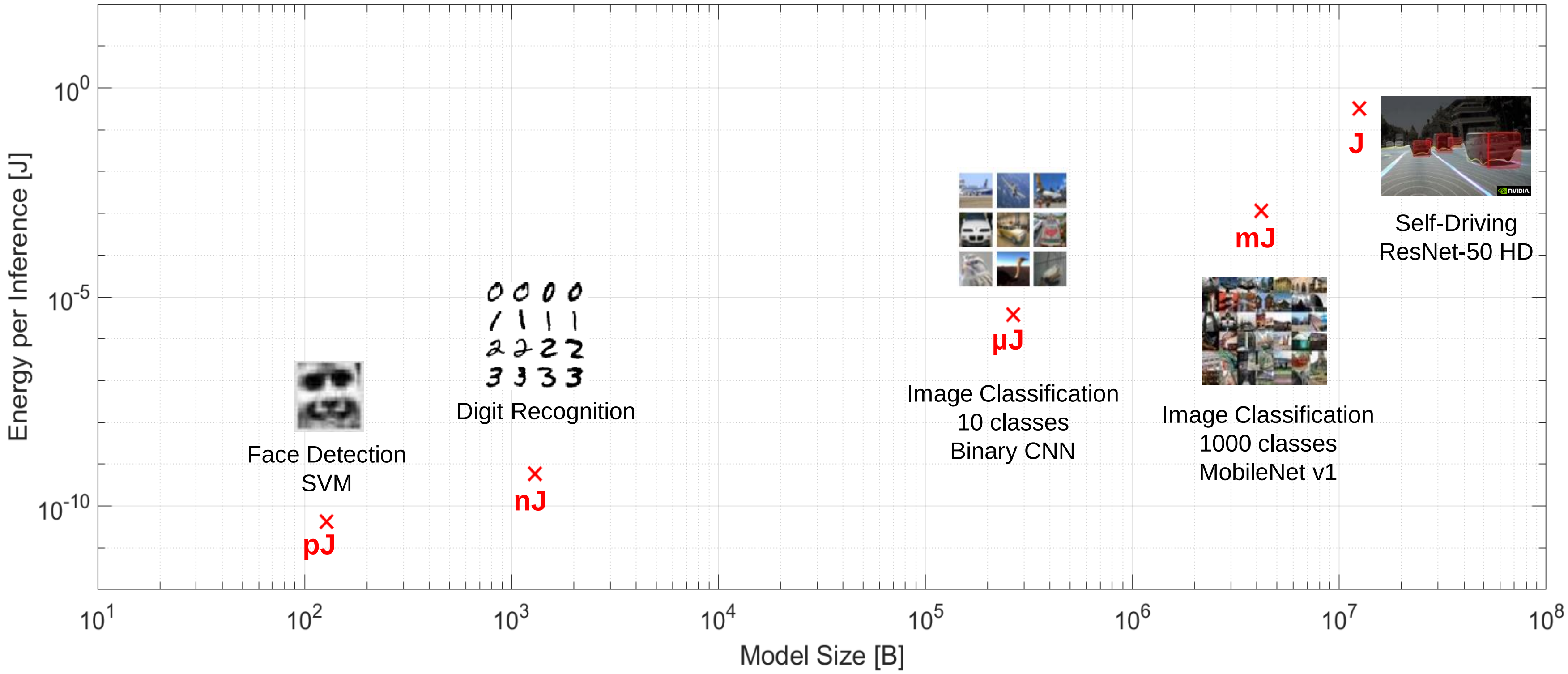
Speed of response

Bandwidth utilized

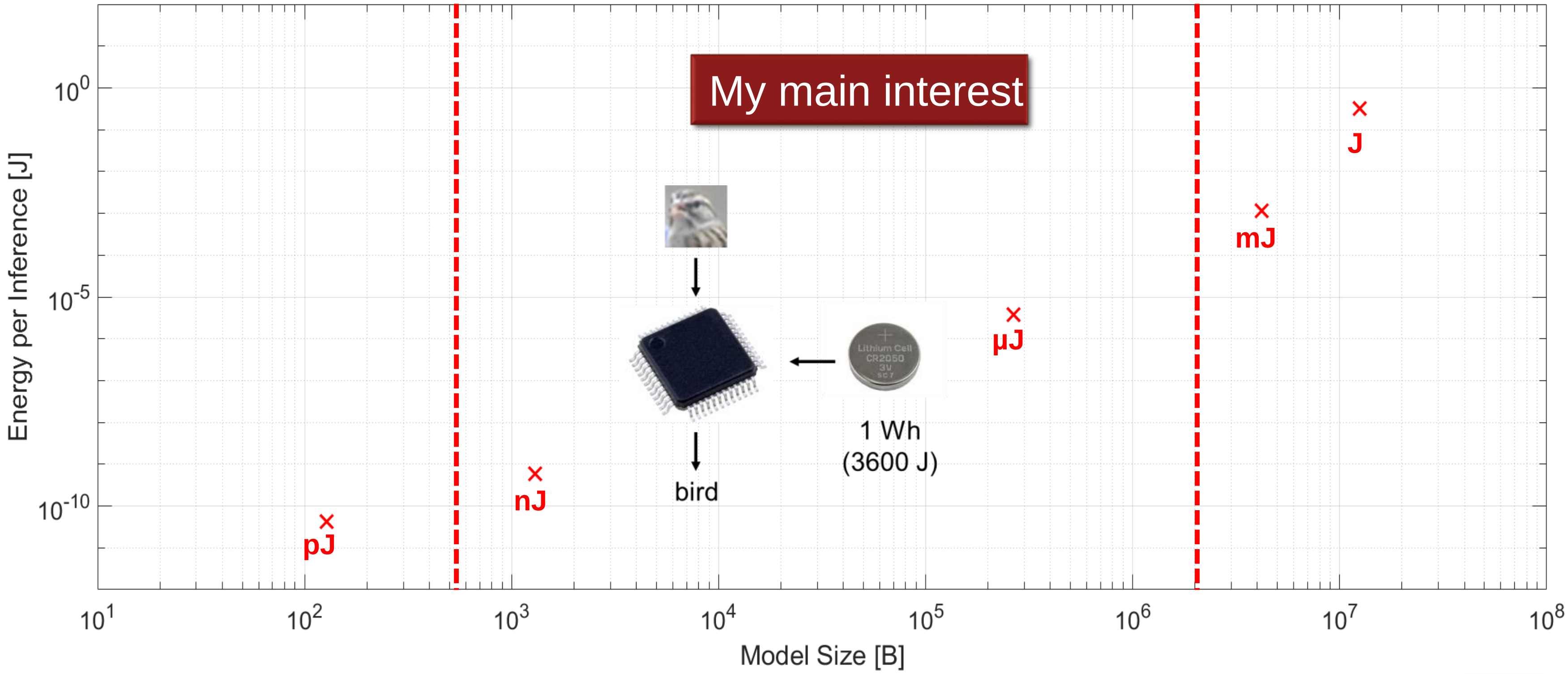
Privacy

Power consumed

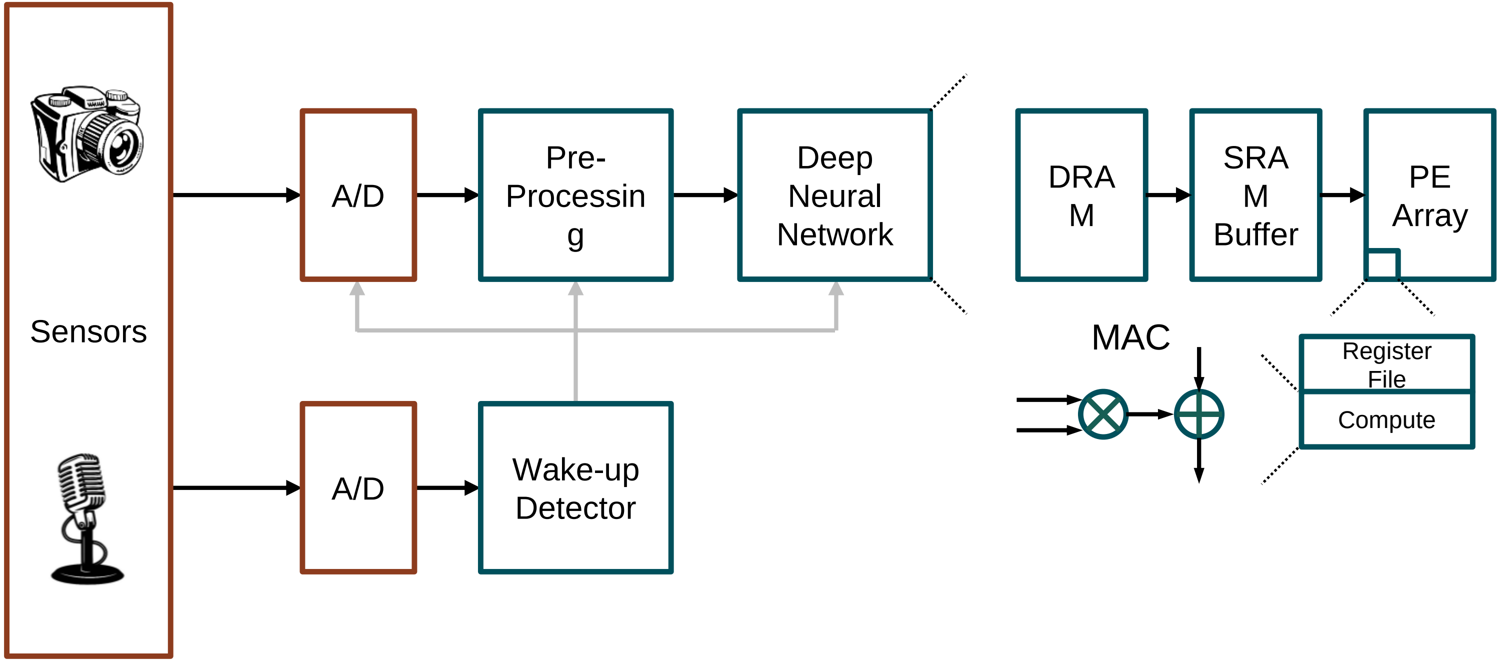
Task Complexity, Memory and Classification Energy



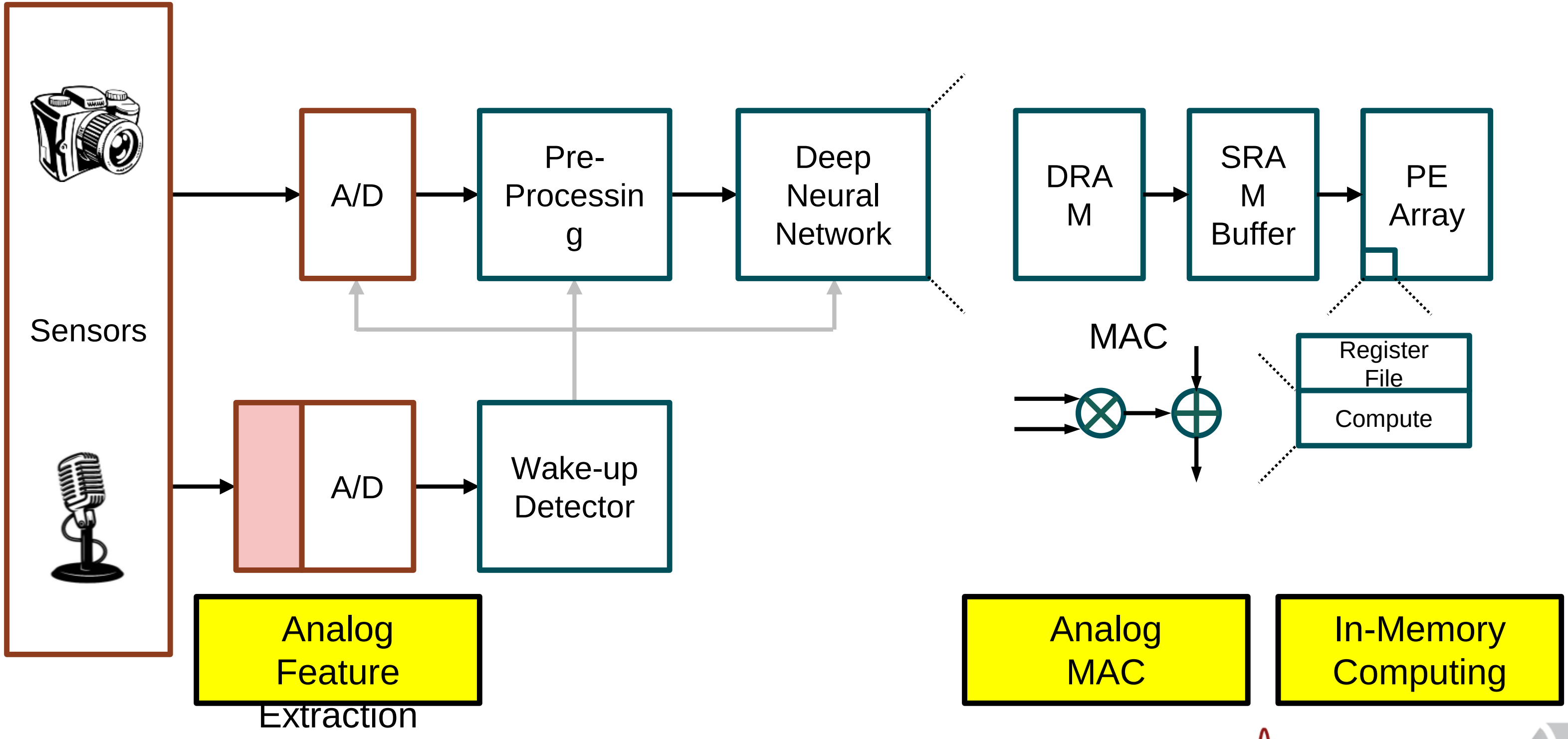
Task Complexity, Memory and Classification Energy



Edge Inference System



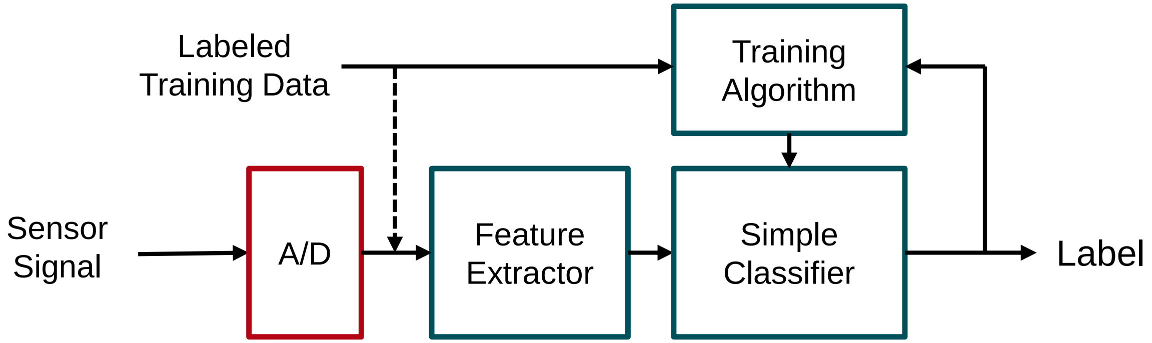
Opportunities for Analog/Mixed-Signal Design



Outline

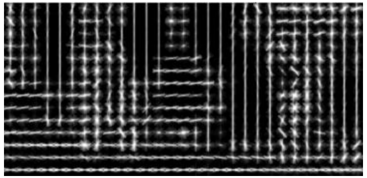
- **Data-Compressive Imager for Object Detection**
  - Omid-Zohoor & Young, TCSVT 2018 & ISSCC 2019
- **Mixed-Signal ConvNet**
  - Bankman, ISSCC 2018 & JSSC 2019
- **RRAM-based ConvNet with In-Memory Compute**
  - Ongoing work

Wake-Up Detector with Hand-Crafted Features



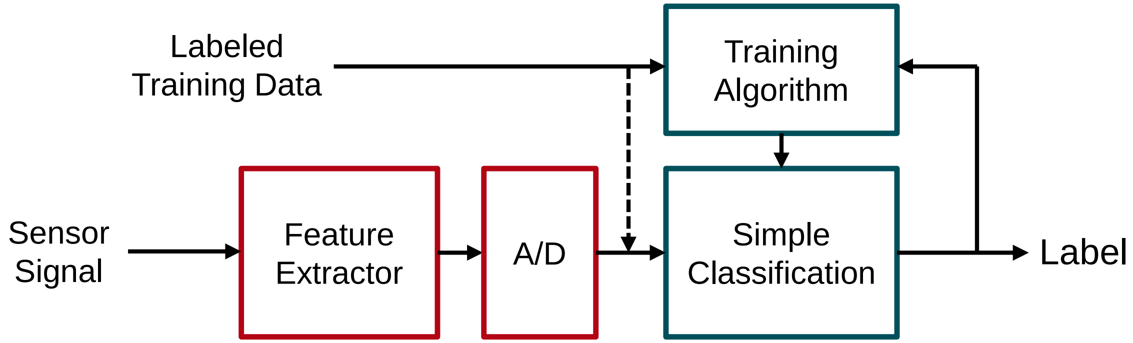
Data Deluge

High-dimensional
data

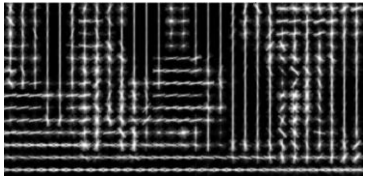


Low-dimensional
representation

Analog Feature Extractor

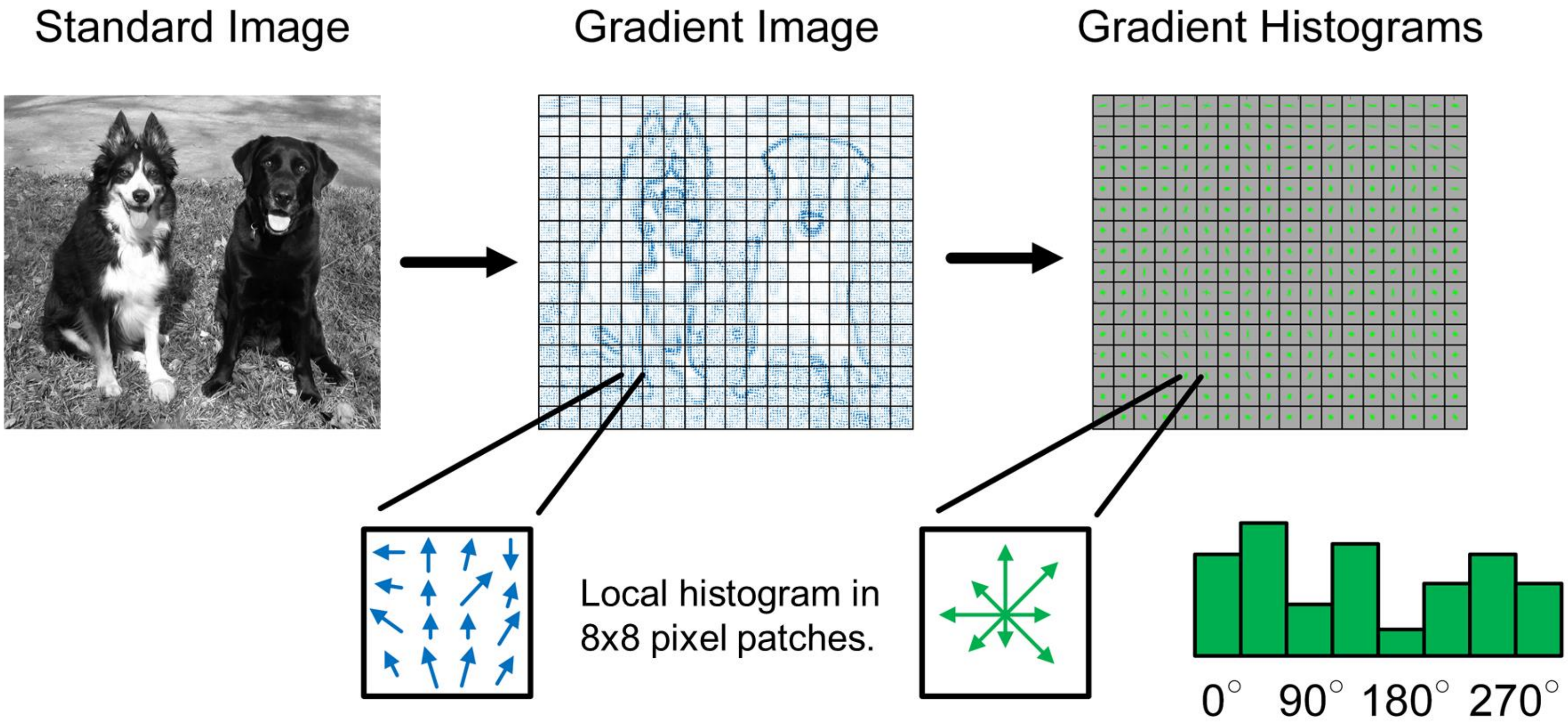


- Low-rate and/or low–resolution ADC
- Low data rate digital I/O
- Reduced memory requirements

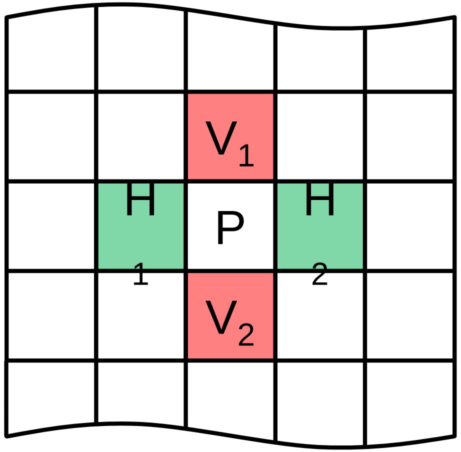


Low-dimensional
representation

Histogram of Oriented Gradients



Analog Gradient Computation



$$G_H = \boxed{H_1} - \boxed{H_2}$$ horizontal

$$G_V = \boxed{V_1} - \boxed{V_2}$$ vertical

Bright patch

$$G_H = 400mV - 100mV = \mathbf{300mV}$$

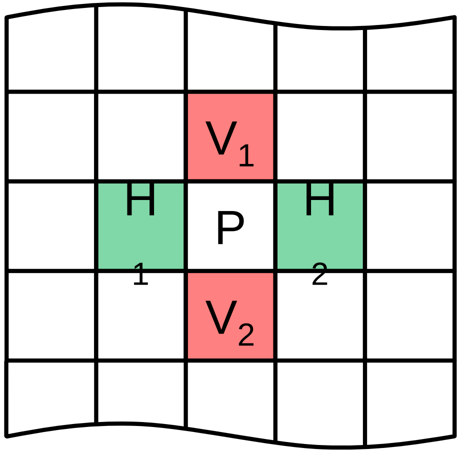
Dark patch

$$G_H = \left(\frac{1}{4}\right)400mV - \left(\frac{1}{4}\right)100mV = \mathbf{75mV}$$

High Dynamic Range Images

- Gradient magnitude varies significantly across image
- Would require high-resolution ADCs (≥ 9b) to digitize computed gradients
  - But, we want to produce as little data as possible

Ratio-Based ("Log") Gradients

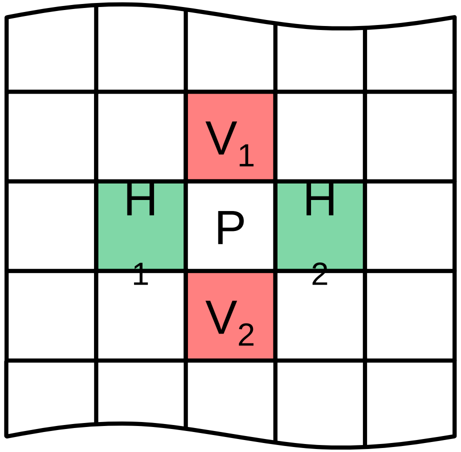


$$G_H = \frac{\boxed{H_1}}{\boxed{H_2}} \qquad G_V = \frac{\boxed{V_1}}{\boxed{V_2}}$$

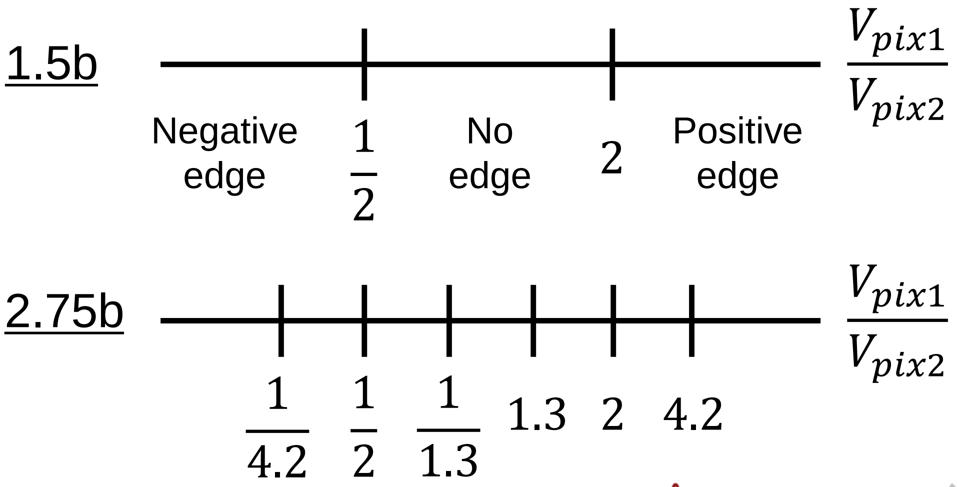
$$G_H = \frac{\alpha \times \boxed{H_1}}{\alpha \times \boxed{H_2}} = \frac{\boxed{H_1}}{\boxed{H_2}}$$

Illumination Invariant

Ratio Quantization

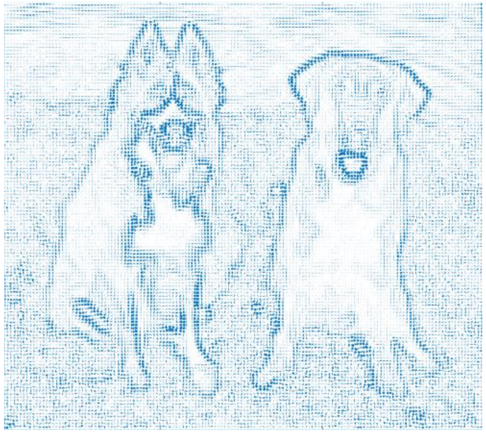


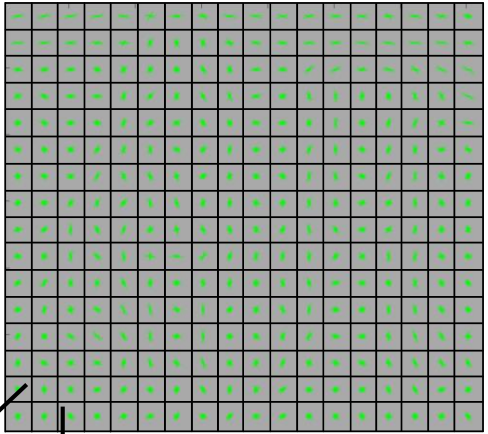
$$G_H = Q\left(\frac{\boxed{H_1}}{\boxed{H_2}}\right) \qquad G_V = Q\left(\frac{\boxed{V_1}}{\boxed{V_2}}\right)$$

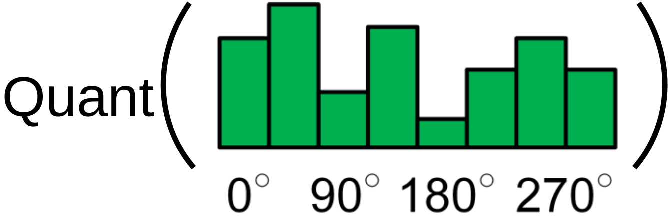


*Empirically determined thresholds

HOG Feature Compression with 1.5b Gradients

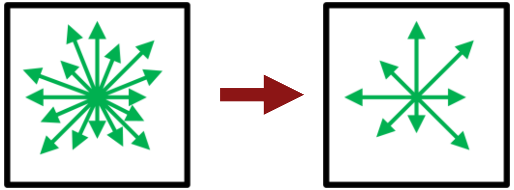






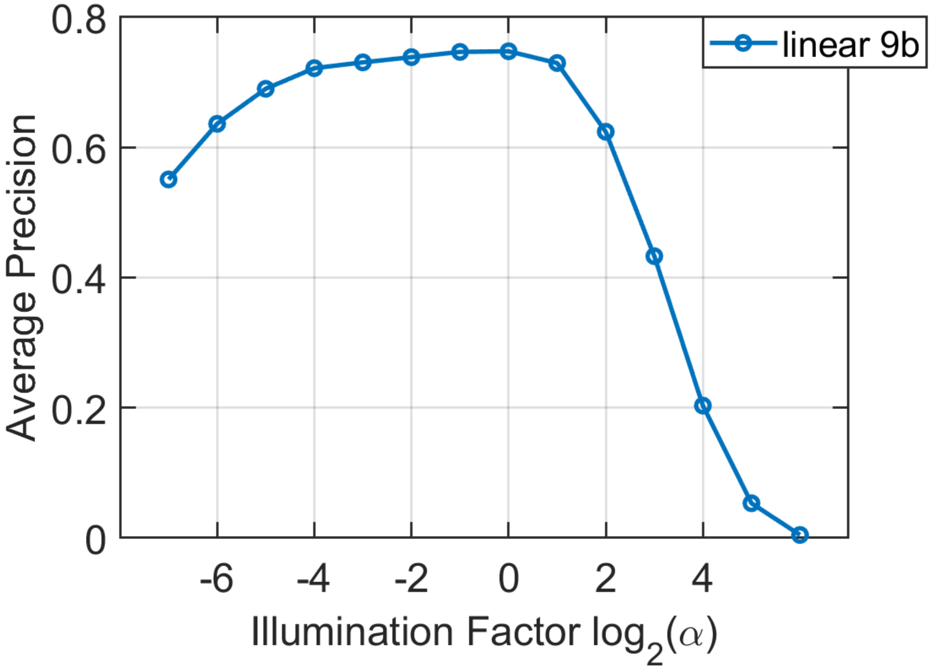
Quantizing histogram magnitudes

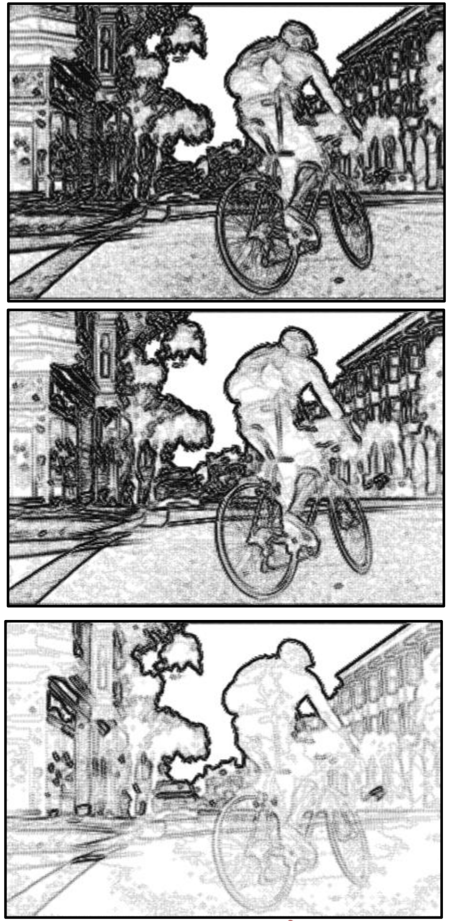
Fewer angle bins



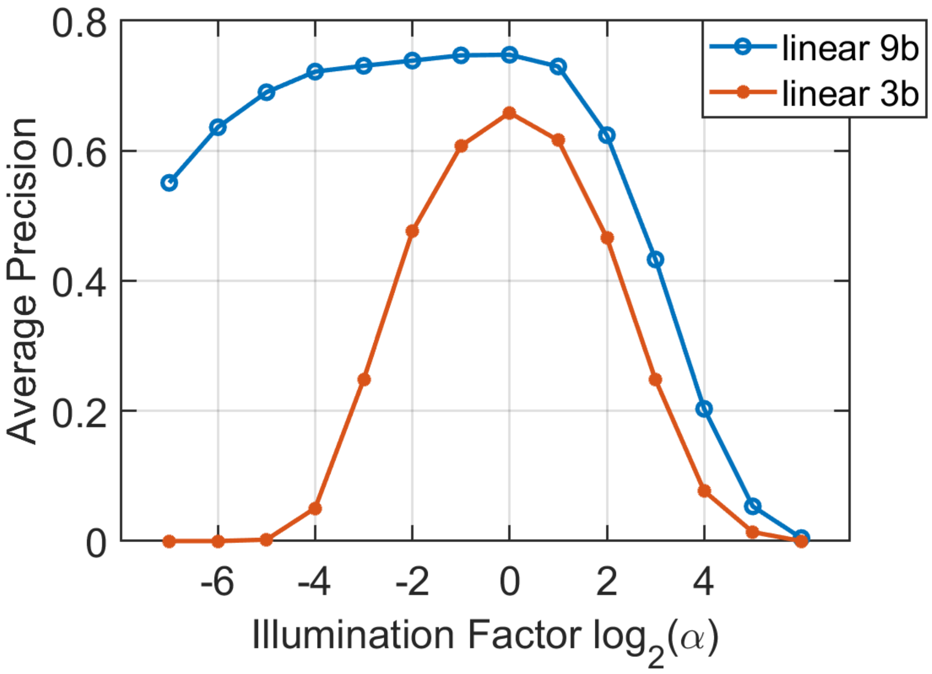
25× less data in HOG features compared to 8-bit image

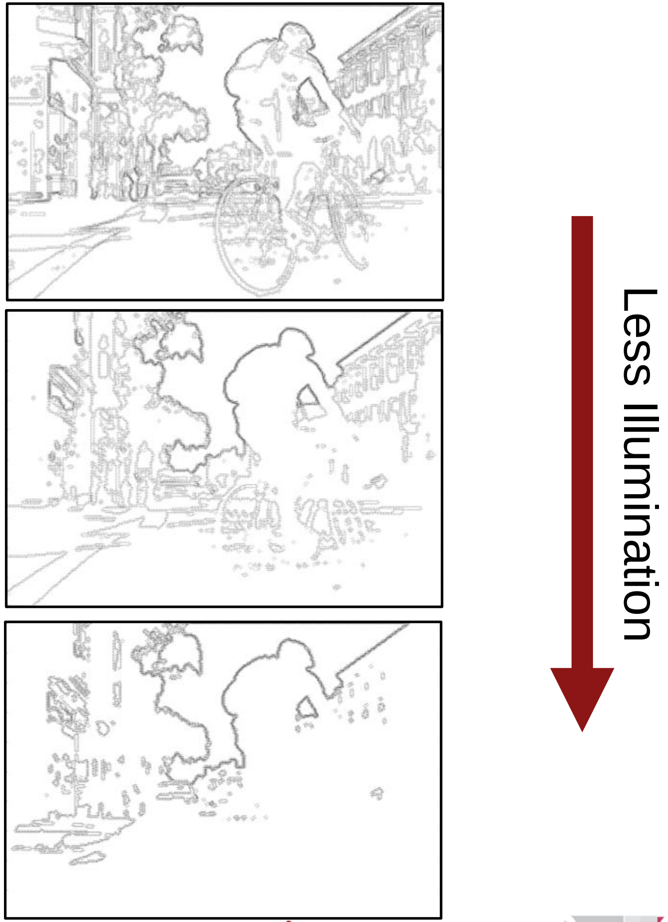
Log vs. Linear Gradients



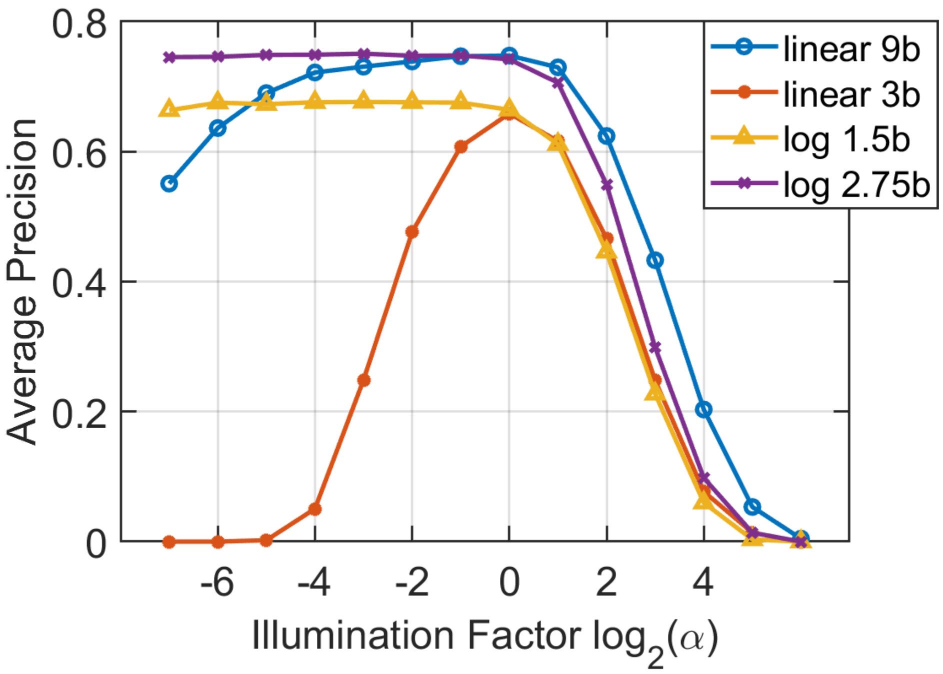


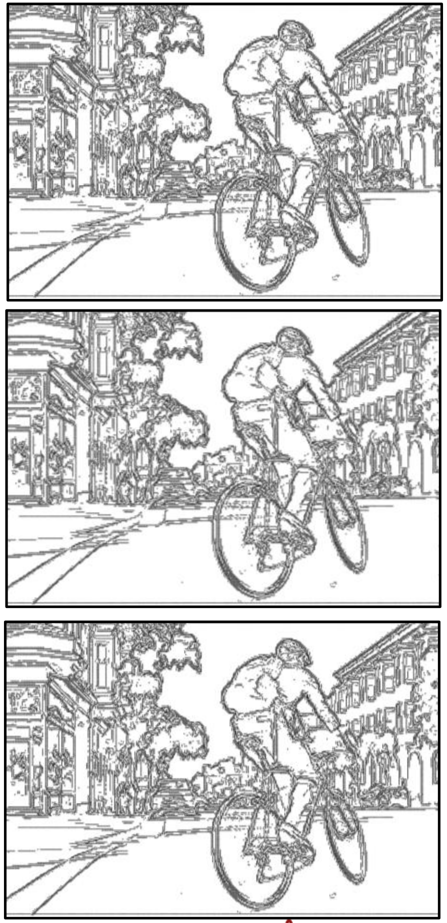
Log vs. Linear Gradients





Log vs. Linear Gradients





Less Illumination

Prototype Chip

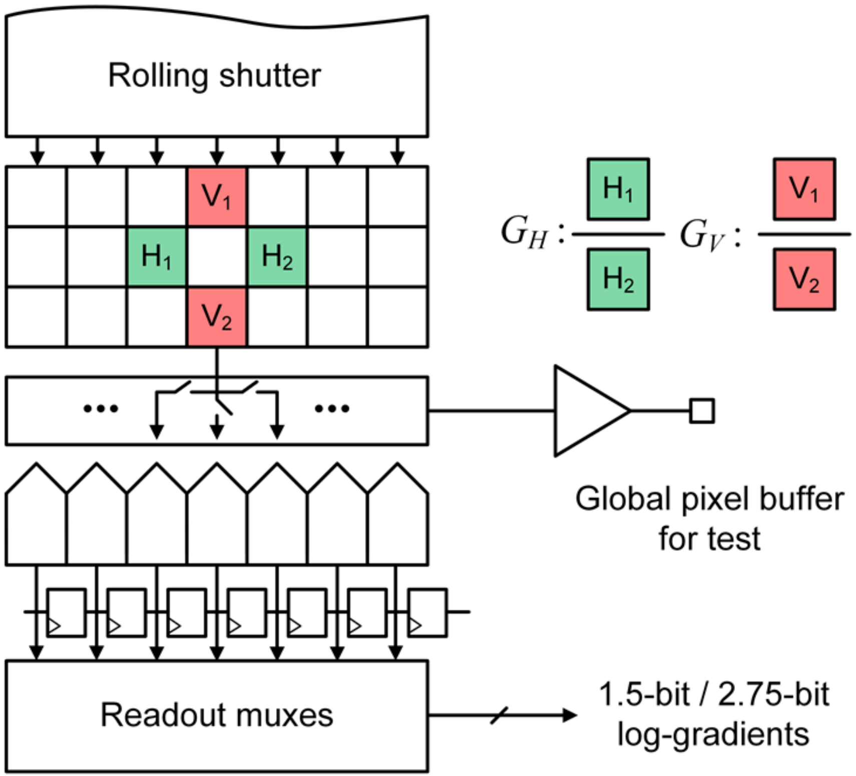
4-T Pixel Array
320V × 240H

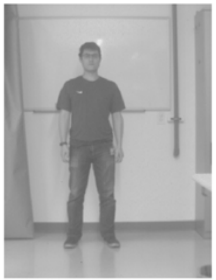
3-Row Cyclic Buffer

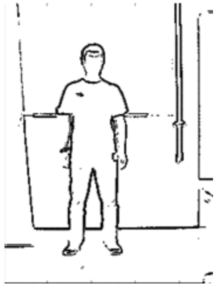
3-Column Muxes

240 Column RDCs

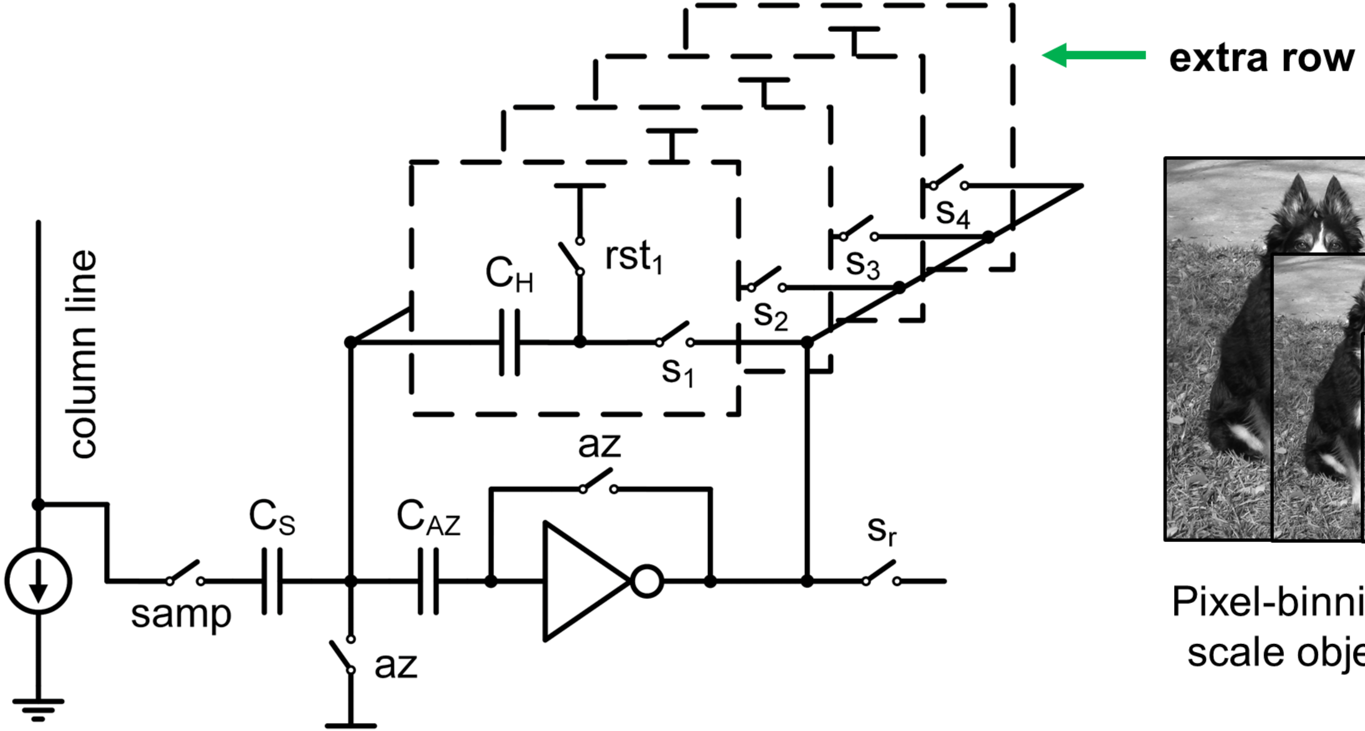
Pixel-Binning
Registers

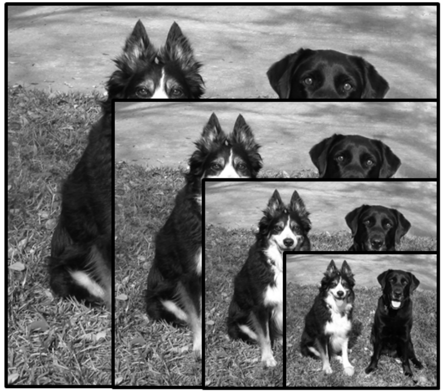






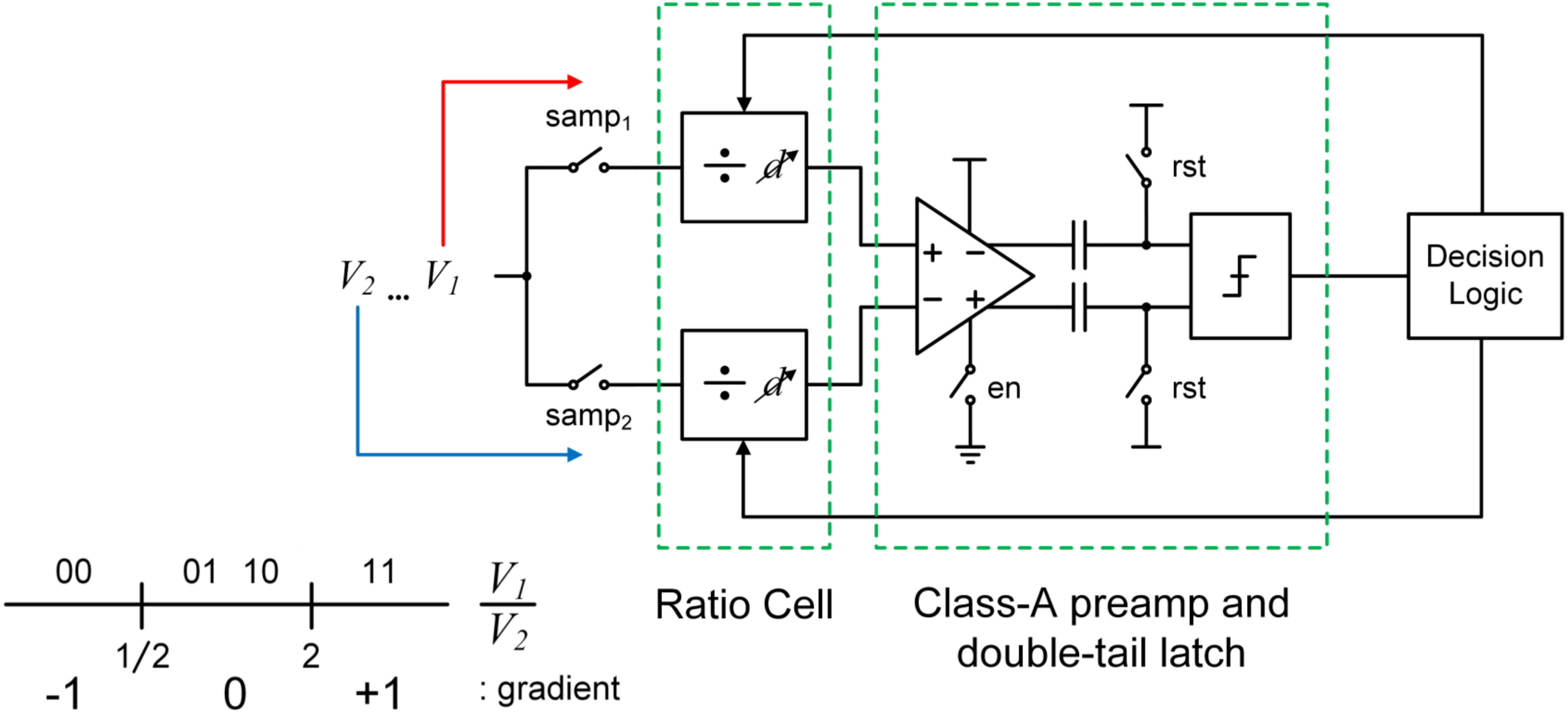
Row Buffers with Pixel Binning (Image Pyramid)





Pixel-binning and multi-scale object detection

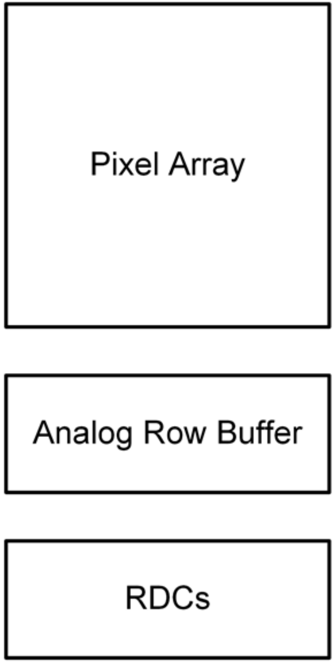
Ratio-to-Digital Converter (RDC)



Data-Driven Spec Derivation

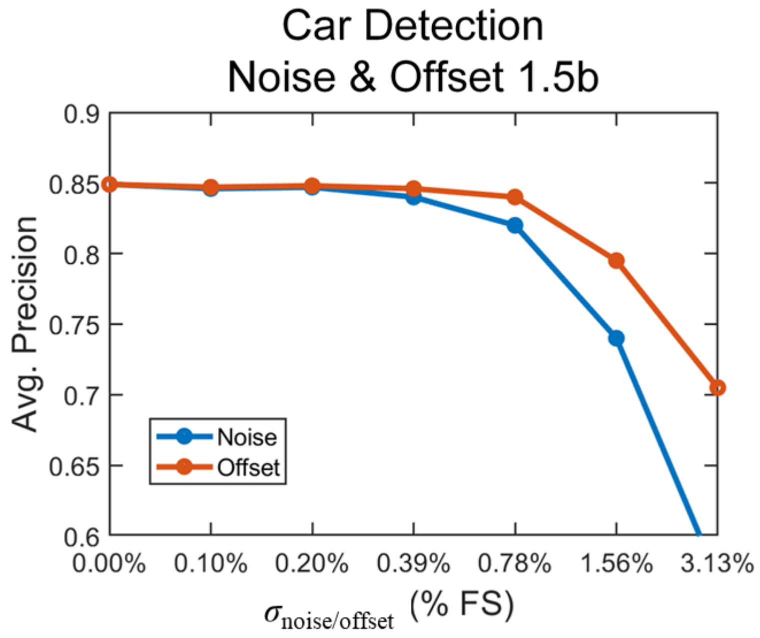
$$\frac{H_1 + \boldsymbol{\epsilon_1}}{H_2 + \boldsymbol{\epsilon_2}}$$

referred
thermal noise
& offset



Sources

- Pixels
- Row buffers
- Pre-Amps
- Comparators

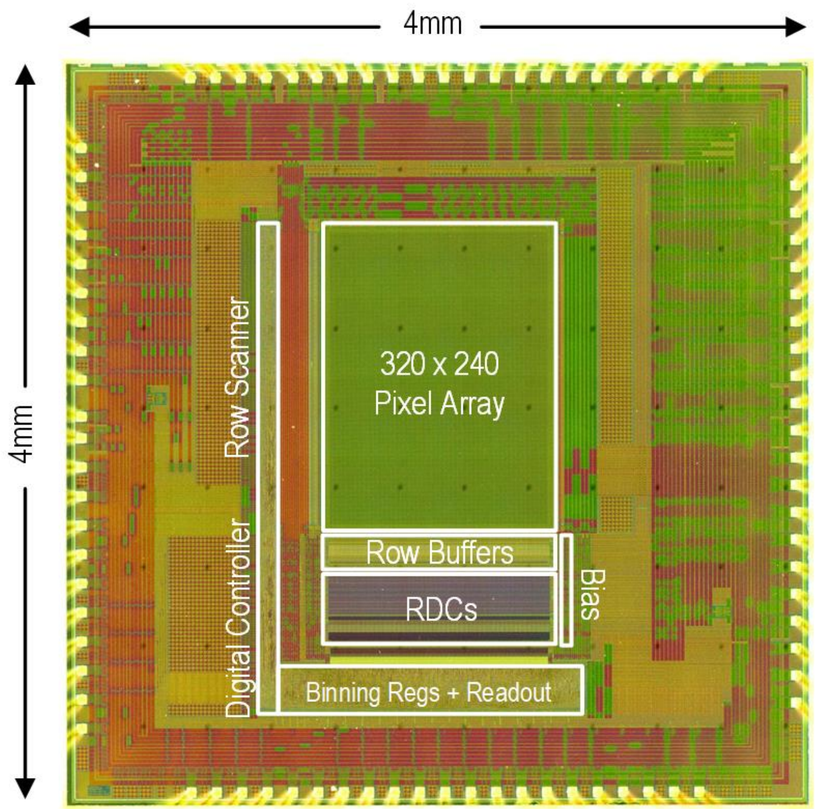


≈ 9 ENOB

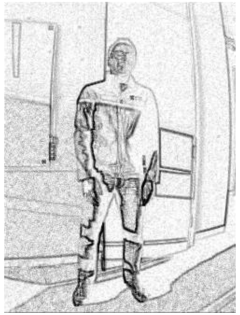
Chip Summary

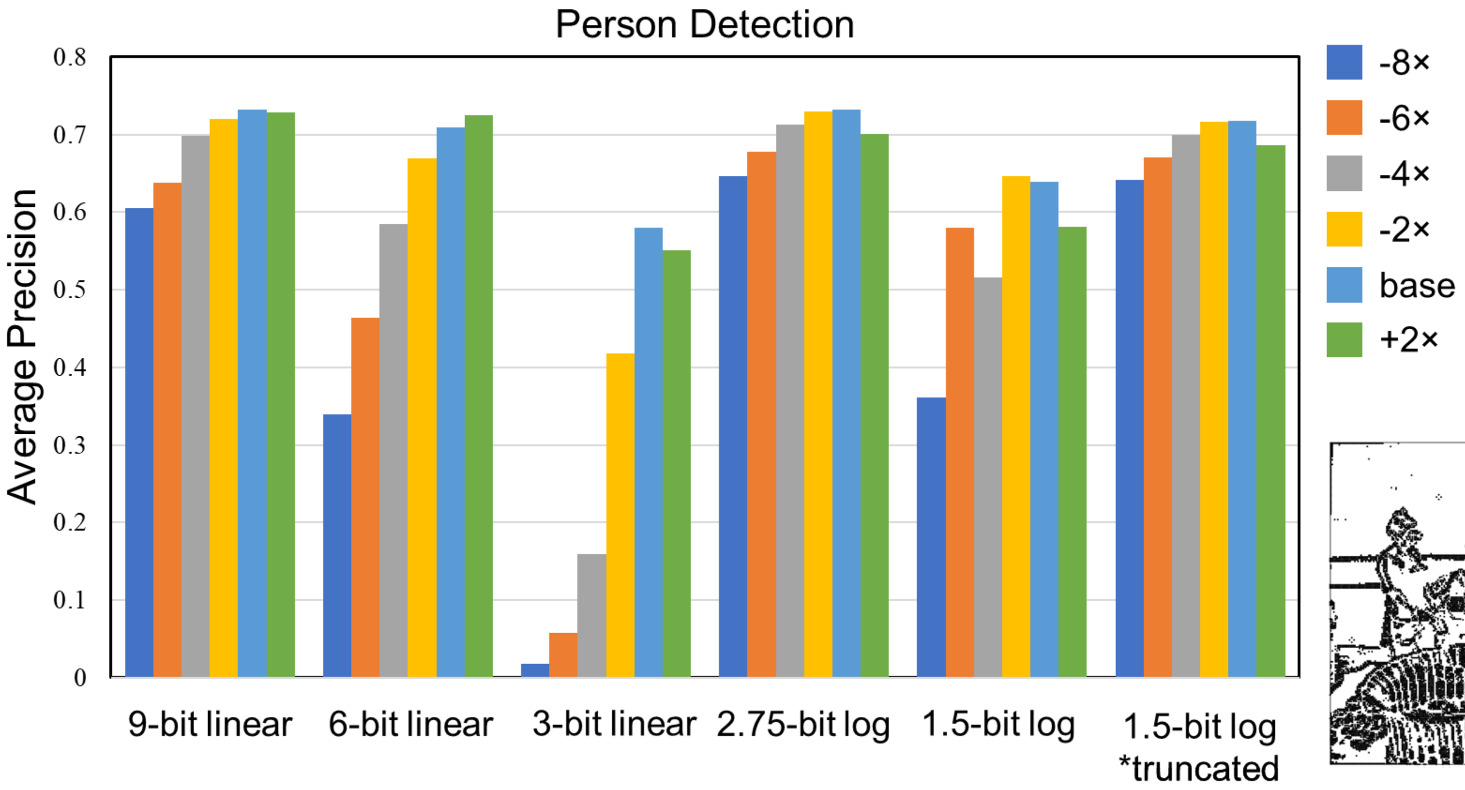
- 0.13 µm CIS 1P4M
- 5µm 4T pixels
- QVGA 320(V) x 240(H)
- 229 µW @ 30 FPS

<u>Supply Voltages</u>
Pixel: 2.5V
Analog: 1.5V, 2.5V
Digital: 0.9V



Sample Images

	Raw, 8b Pixels	9b Linear Gradients	2.75b Log Gradients	1.5b Log Gradients	1.5b Log Gradients *Truncated
Nominal Exposure					
4x less light					



Results using Deformable Parts Model detection & custom database (PascalRAW)

Comparison to State of the Art

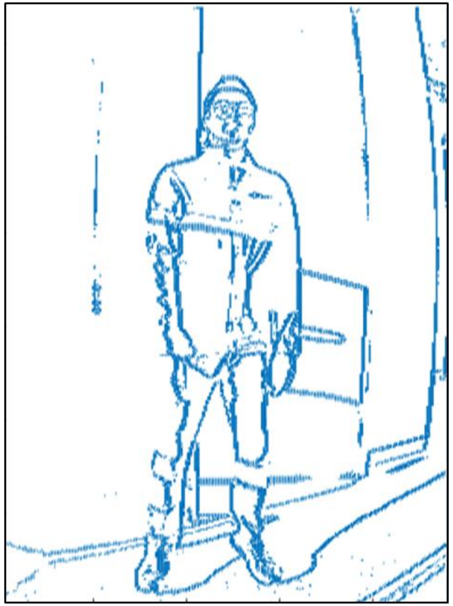
	This Work	[Choi, ISSCC'13]	[Katic, Sens.J.'15]	[Bong, ISSCC'17]
Technology	**0.13 μm 1P4M**	0.18 μm 1P4M	0.18 μm	65 nm 1P8M
Resolution	**320x240**	256x256	32x32	320x240
Pixel Size	**5 μm x 5 μm**	5.9 μm x 5.9 μm	31 μm x 26 μm	7 μm x 7 μm
Fill Factor	**60.4%**	30%	24%	-
Feature Type	**log-gradients**	linear HOGs	relative ratios between pixels	linear Haar-like w/ face-detector
Frame Rate	**30 fps nom. 207 fps max**	15 fps - reported	9756 fps nom. 24000 fps max	1 fps - reported
Dynamic Range	**59.3 dB**[1]	54.8 dB	43 dB[2]	-
Power Consumption	**229 μW @ 30fps**	51 μW @ 15 fps	4 mW @ 9765 fps	24-96 μW @ 1fps
Energy Efficiency	**1.5-bit: 99 pJ/pixel 2.75-bit: 114 pJ/pixel**	52 pJ/pixel	404 pJ/pixel	312 - 1250 pJ/pixel
Multi-Scale	**Yes - arbitrary square bins**	No	No	Yes - three scales

1. At output of cyclic row buffer, without RDC 2. Pixel-to-pixel dynamic range

Information Preservation

Raw Pixels

1.5-bit Log Gradients

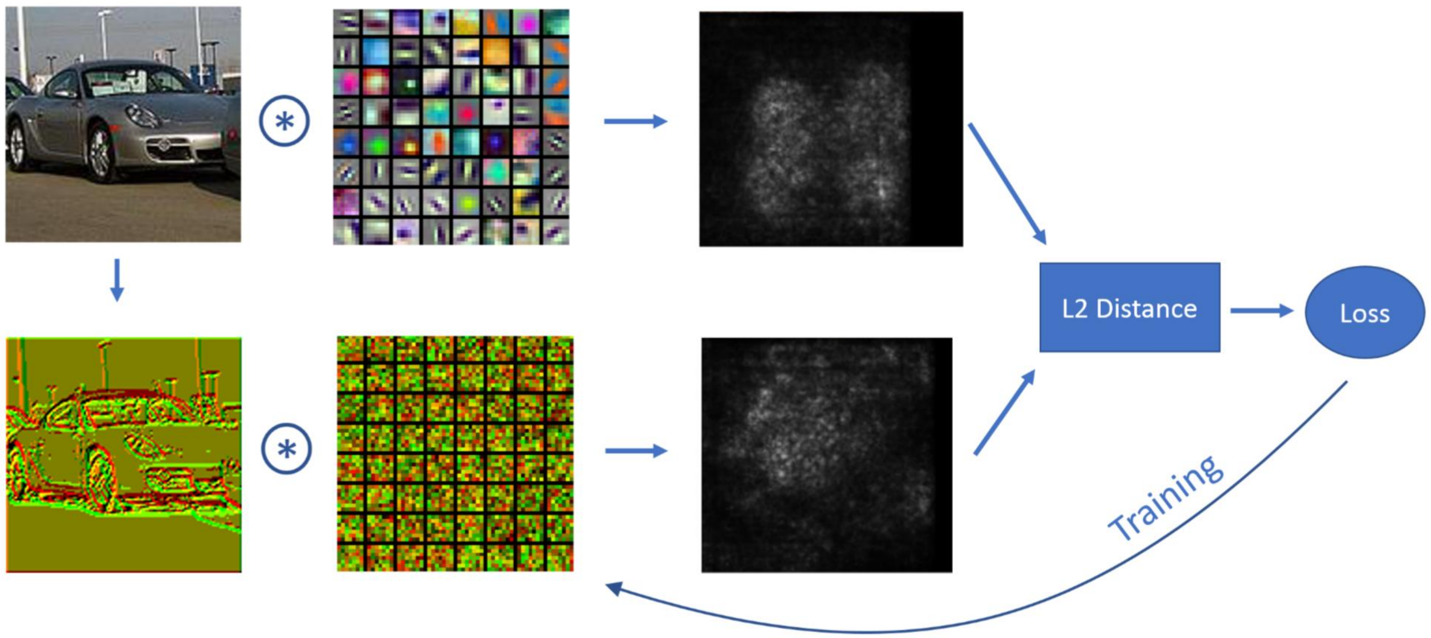


*truncated from 2.75-bit

Reconstruction

*courtesy Julien Martel

Use Log Gradients as ConvNet Input?



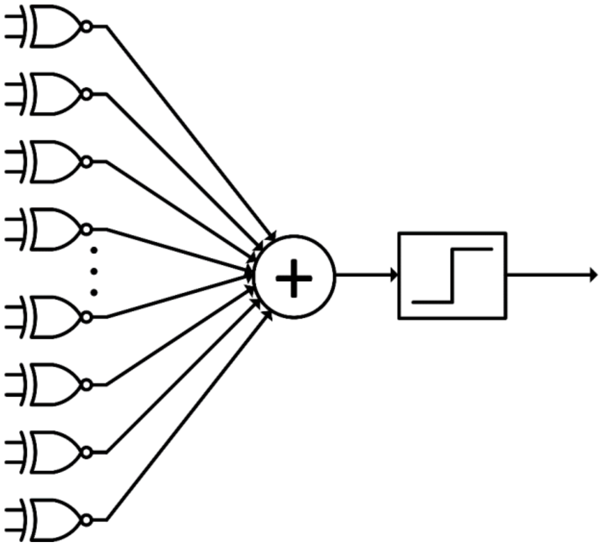
- Ongoing work; comparable performance using ResNet-10 (PascalRaw dataset)

Can We Play Mixed-Signal Tricks in a ConvNet?

Digital ConvNet Fabric

BinaryNet

- Courbariaux et al., NIPS 2016
- Weights and activations constrained to +1 and -1, multiplication becomes XNOR
- Minimizes D/A and A/D overhead
- Nice option for small/medium-size problems and mixed-signal exploration

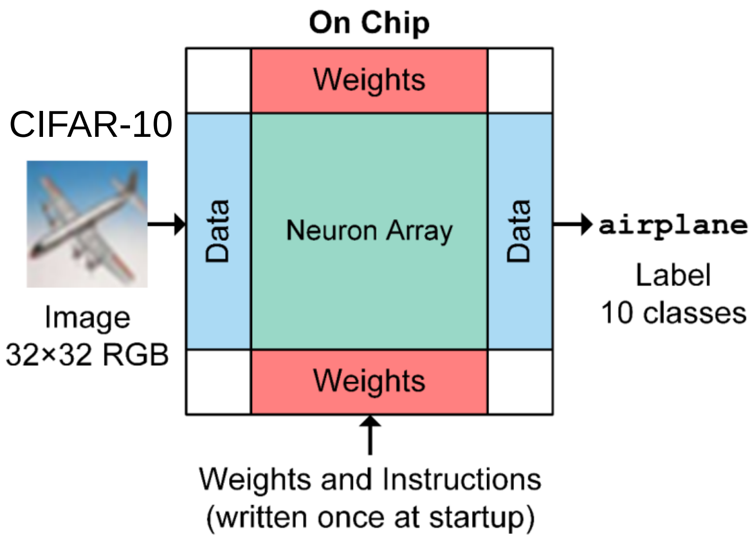


Mixed-Signal Binary CNN Processor

1. Binary CNN with "CMOS-inspired" topology, engineered for minimal circuit-level path loading

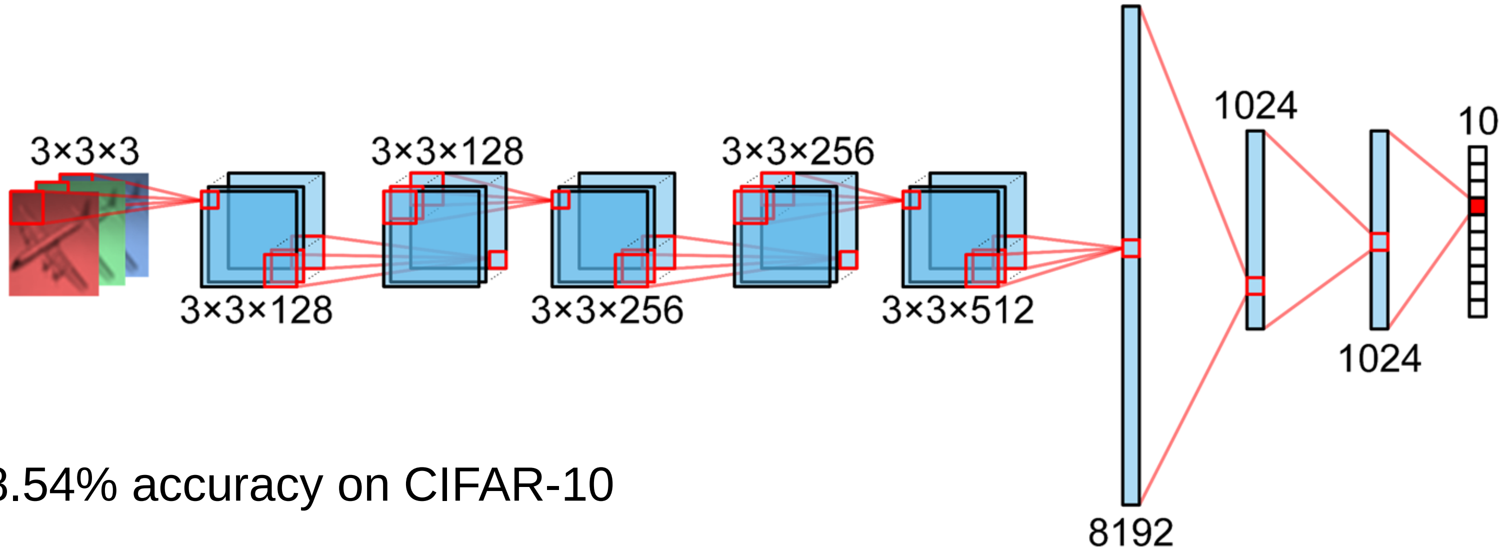
2. Hardware architecture amortizes memory access across many computations, with all memory on chip (328 KB)

3. Energy-efficient switched-capacitor neuron for wide vector summation, replacing digital adder tree



Bankman et al., ISSCC 2018

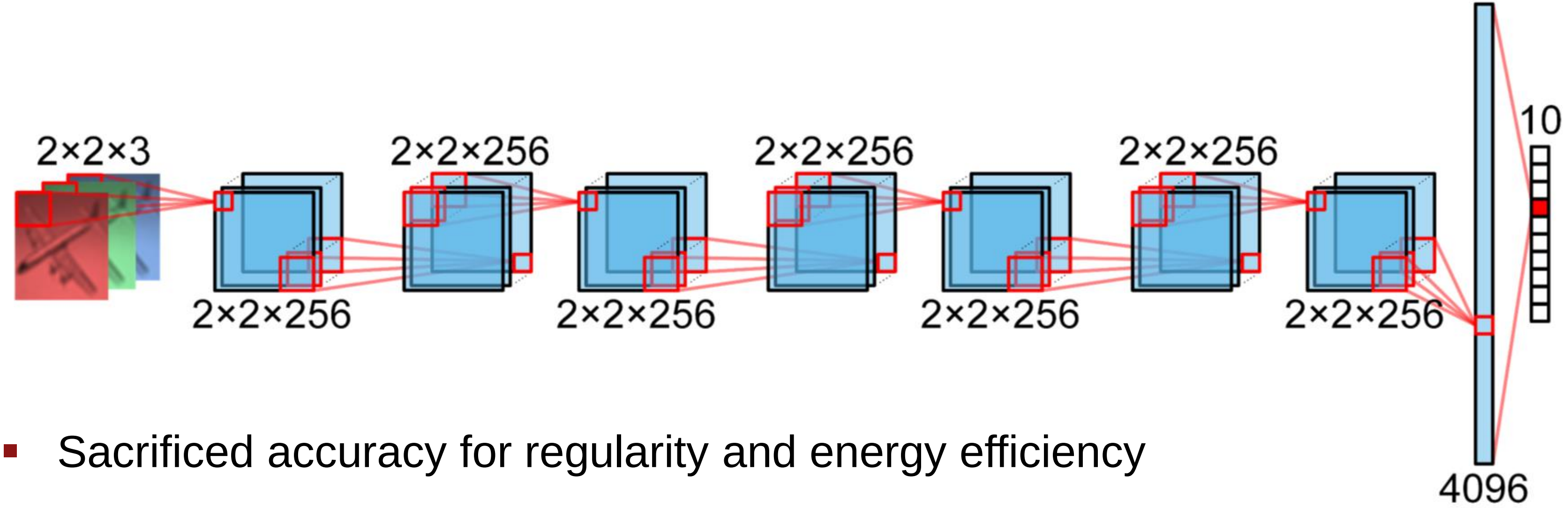
Original BinaryNet Topology



- 88.54% accuracy on CIFAR-10
- 1.67 MB weight memory (68% FC layers)
- 27.9 mJ/classification with FPGA

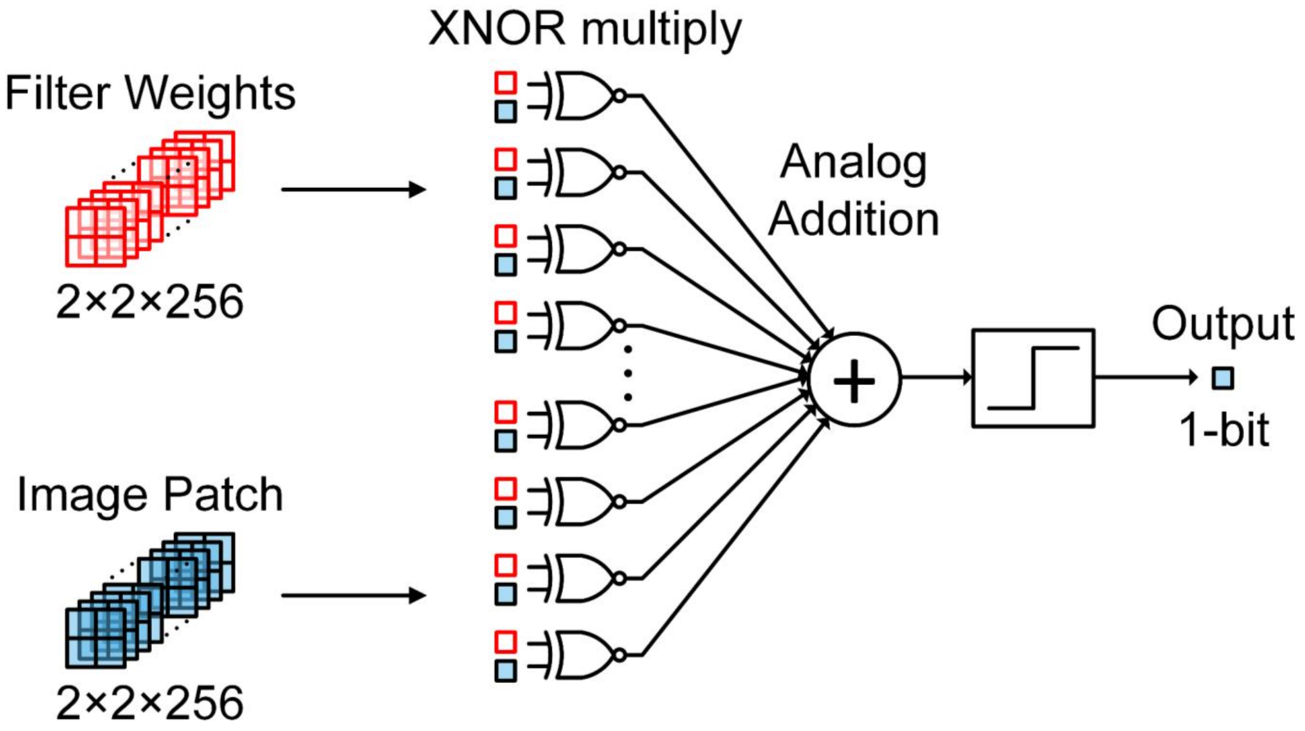
Zhao et al., FPGA 2017

Mixed-Signal BinaryNet Topology



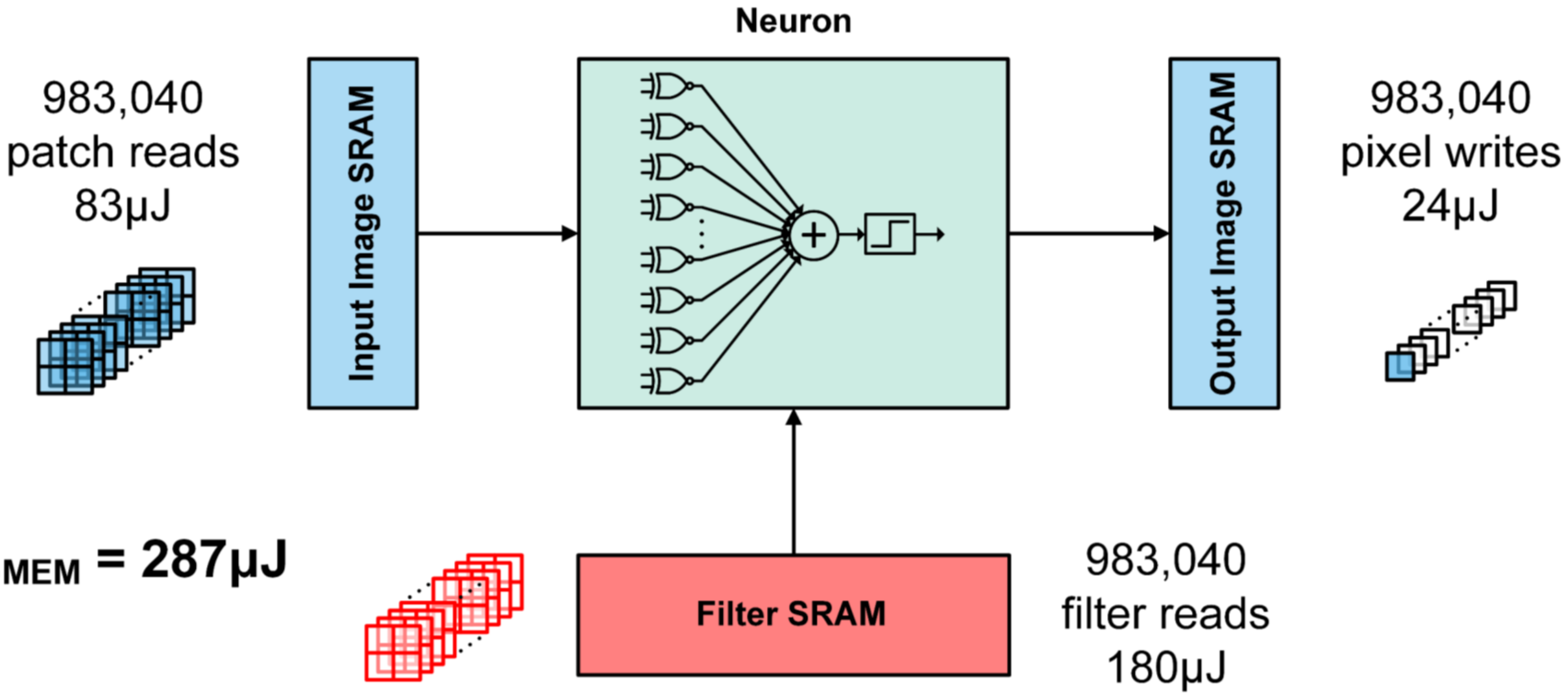
- Sacrificed accuracy for regularity and energy efficiency
- 86.05% accuracy on CIFAR-10
- 328 KB weight memory
- 3.8 μJ per classification

Neuron



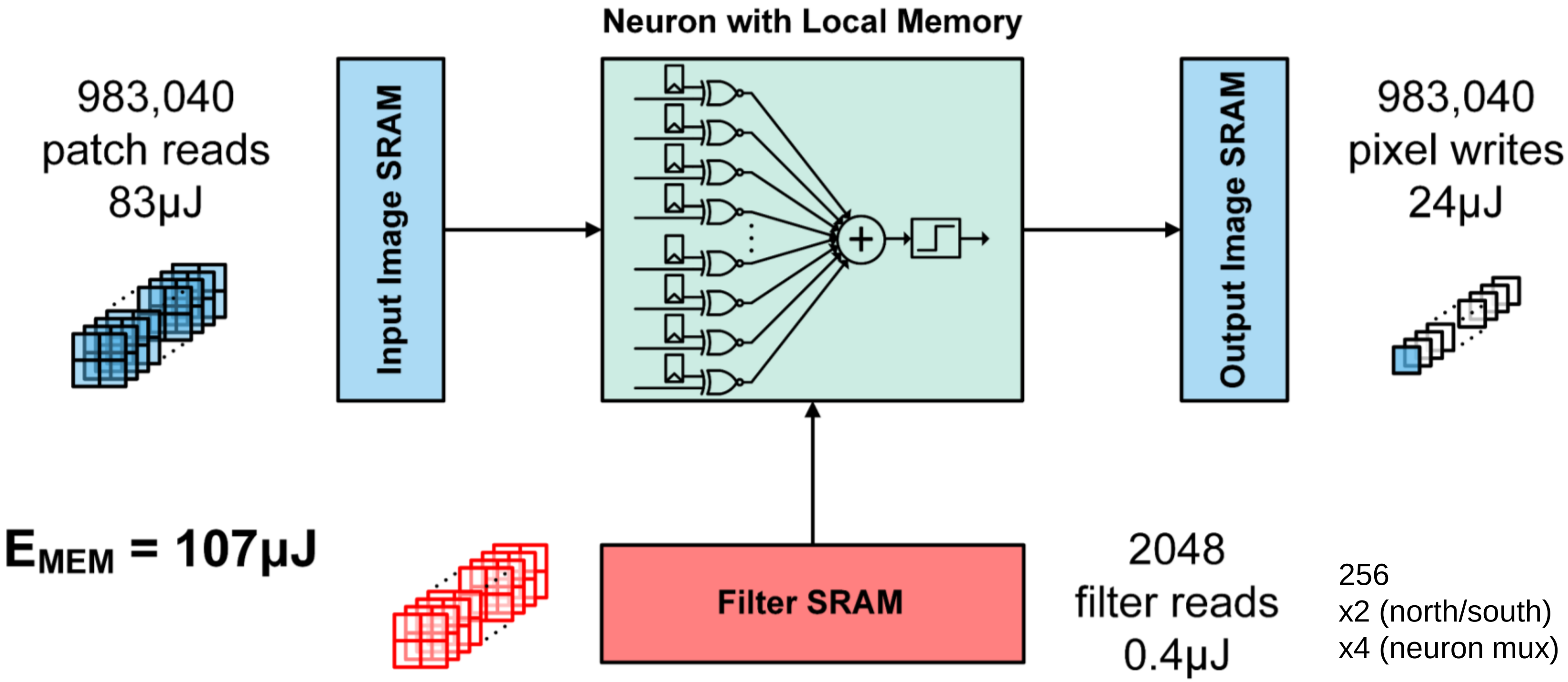
XNOR multiply
Filter Weights
2×2×256
Analog
Addition
Output
1-bit
Image Patch
2×2×256

Naïve Sequential Computation

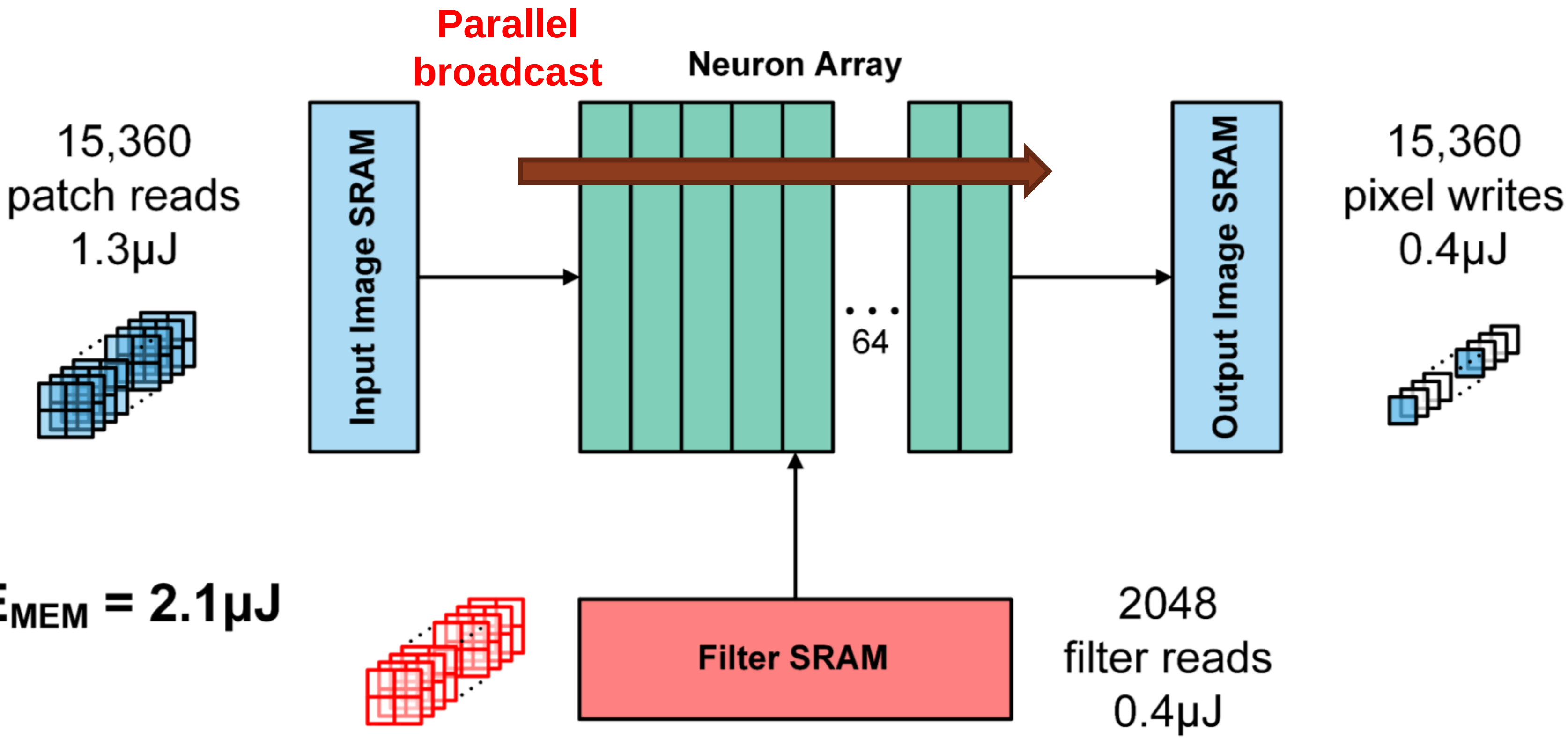


E_{MEM} = 287μJ

Weight-Stationary

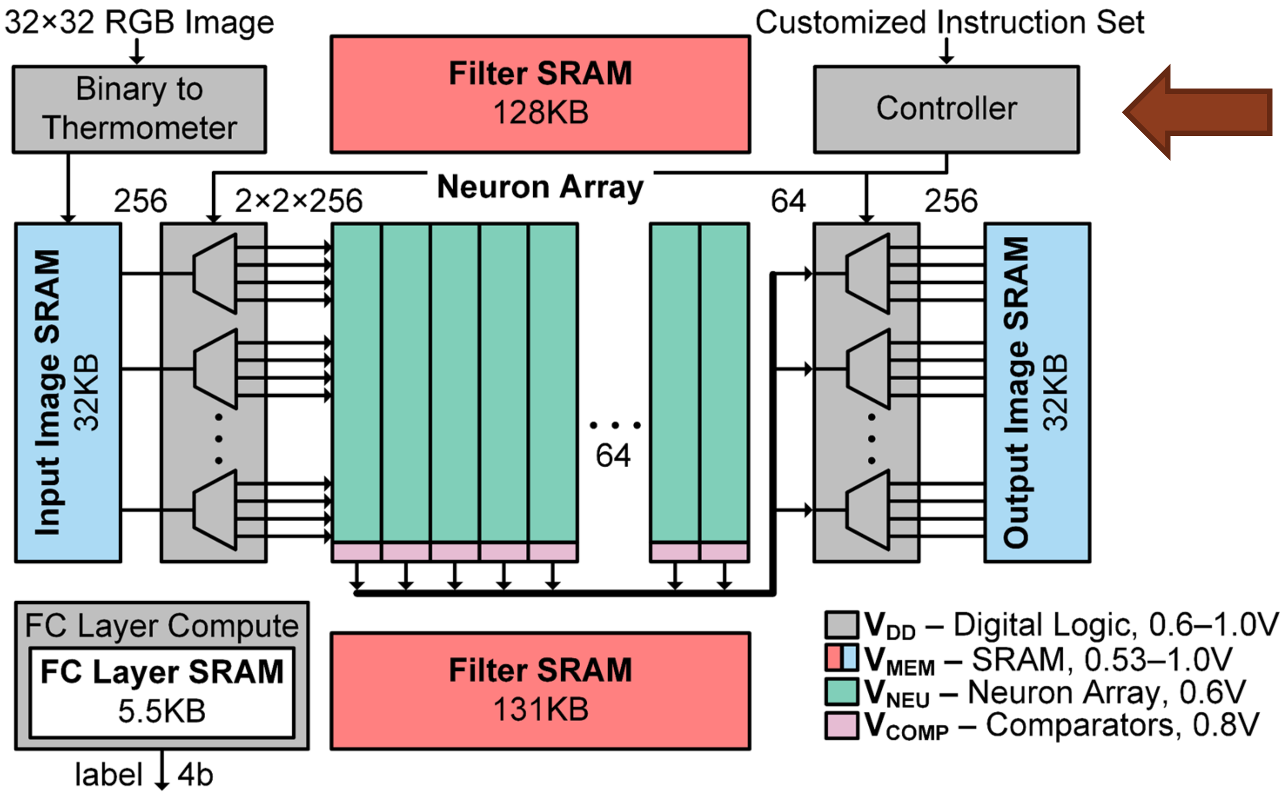


Weight-Stationary and Data-Parallel



$E_{MEM} = 2.1\mu J$

Complete Architecture

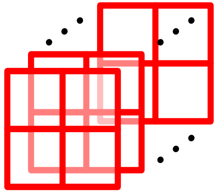


Neuron Function

$$z = \operatorname{sgn}\left(\sum_{i=0}^{1023} w_i x_i + b\right)$$

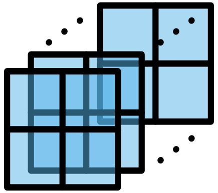
z
Output

w
Filter Weights



$2 \times 2 \times 256$

x
Image Patch



$2 \times 2 \times 256$

b
Filter Bias
9b sign-magnitude

$$b = (-1)^s \sum_{i=0}^{7} 2^i m_i$$

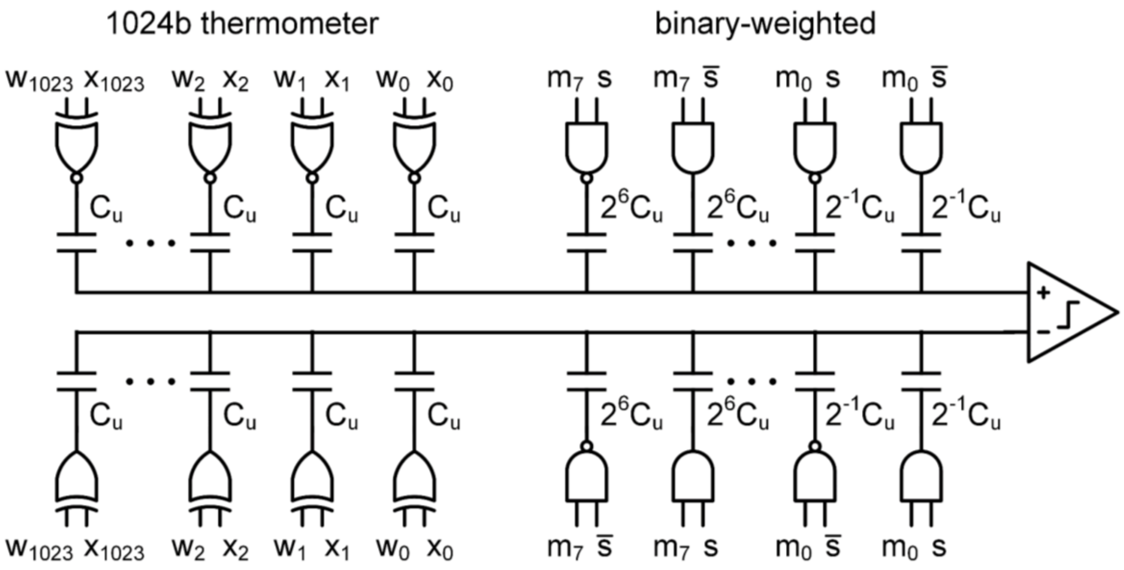
Switched-Capacitor Implementation

$$\frac{v_{\text{diff}}}{V_{DD}} = \frac{C_u}{C_{tot}}\left(\sum_{i=0}^{1023} w_i x_i + b\right)$$

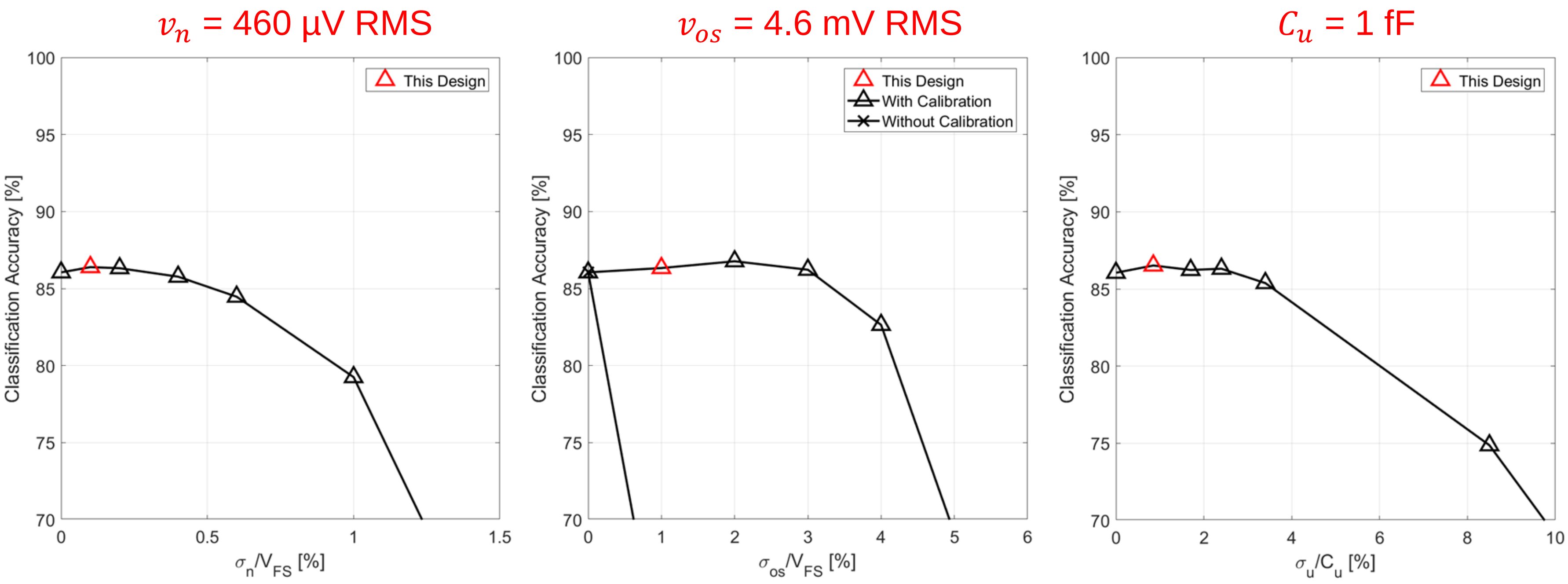
$$b = (-1)^s \sum_{i=0}^{7} 2^i m_i$$

- Batch normalization folded into weight signs and bias

Weights x inputs **Bias & offset cal.**

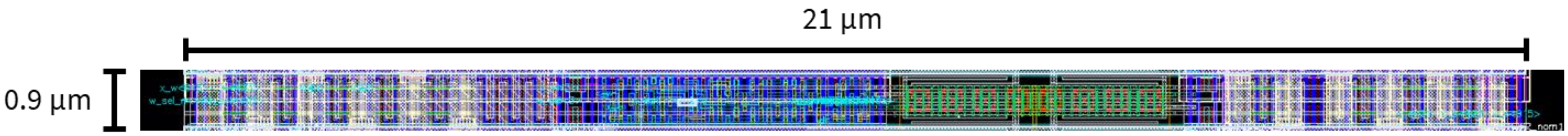


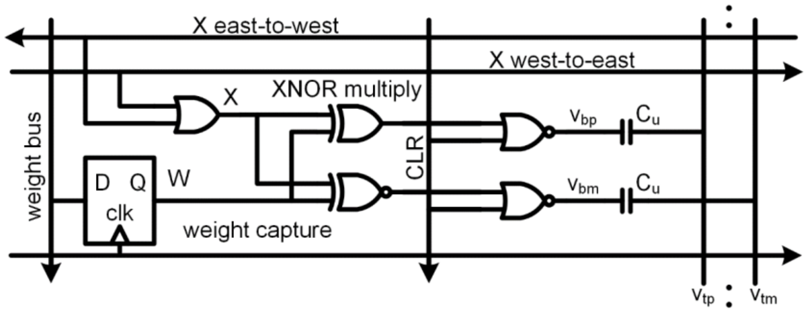
Behavioral Simulations

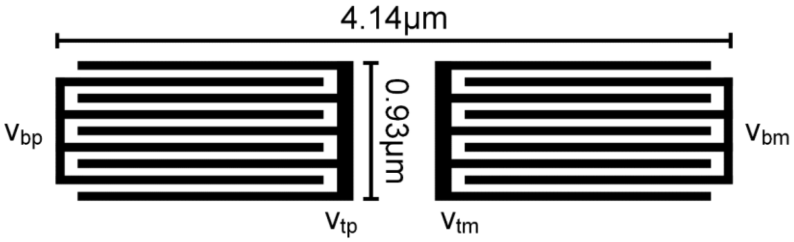


- Significant margin in noise, offset, and mismatch (V_{FS} = 460 mV)

“Memory-Cell-Like” Processing Element







Standard-cell-based
42 transistors
24107 F^2

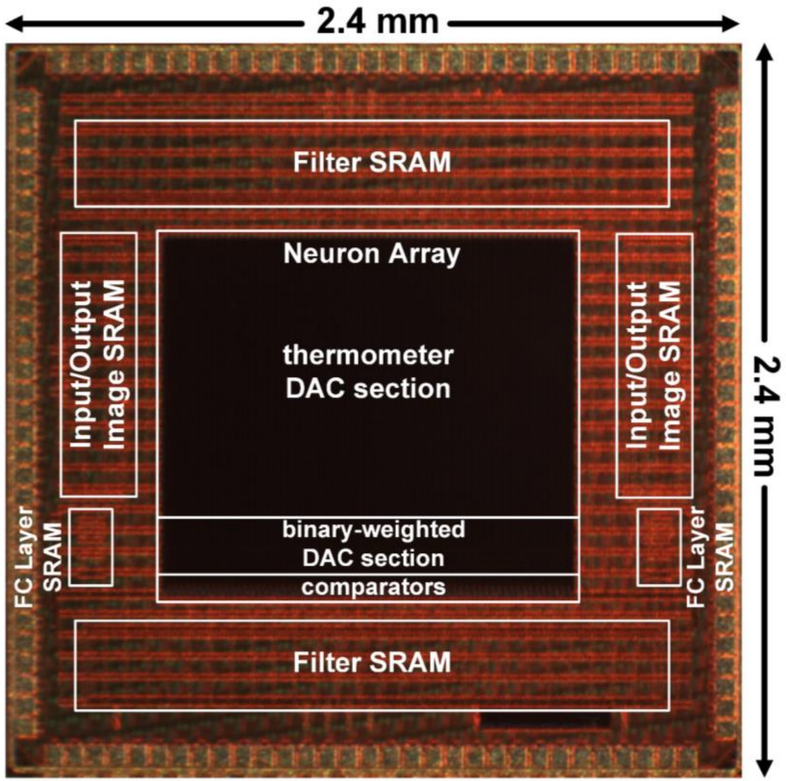
1 fF metal-oxide-metal fringe capacitor

Die Photo

- TSMC 28nm HPL 1P8M
- 6 mm^2 area
- 328 KB SRAM
- 10 MHz clock

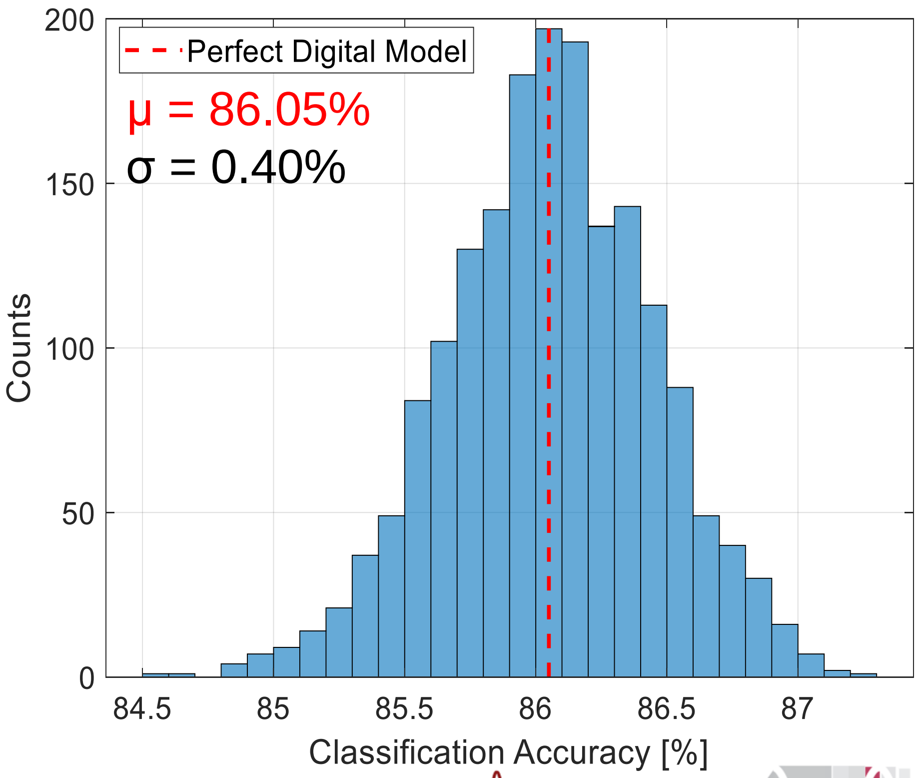
<u>Supply Voltages</u>

V_{DD} – Digital Logic, 0.6V – 1.0V
V_{MEM} – SRAM, 0.53V – 1.0V
V_{NEU} – Neuron Array, 0.6V
V_{COMP} – Comparators, 0.8V



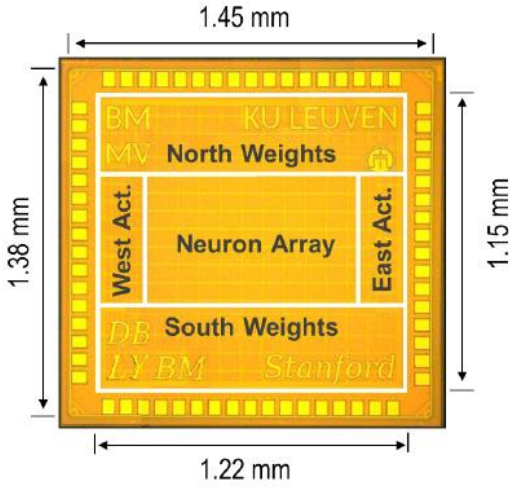
Measured Classification Accuracy

- 10 chips, 180 runs each through 10,000 CIFAR-10 test images
- V_{DD} = 0.8V, V_{MEM} = 0.8V
- 3.8 μJ/classification
- 237 FPS, 899 μW
- 0.43 μJ in 1.8V I/O
- Mean accuracy μ = 86.05% same as perfect digital model



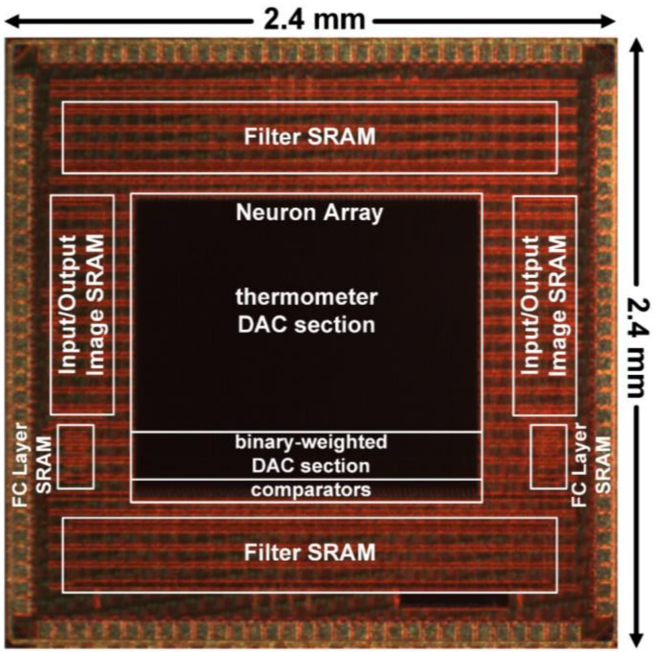
Comparison to Synthesized Digital

Synthesized Digital

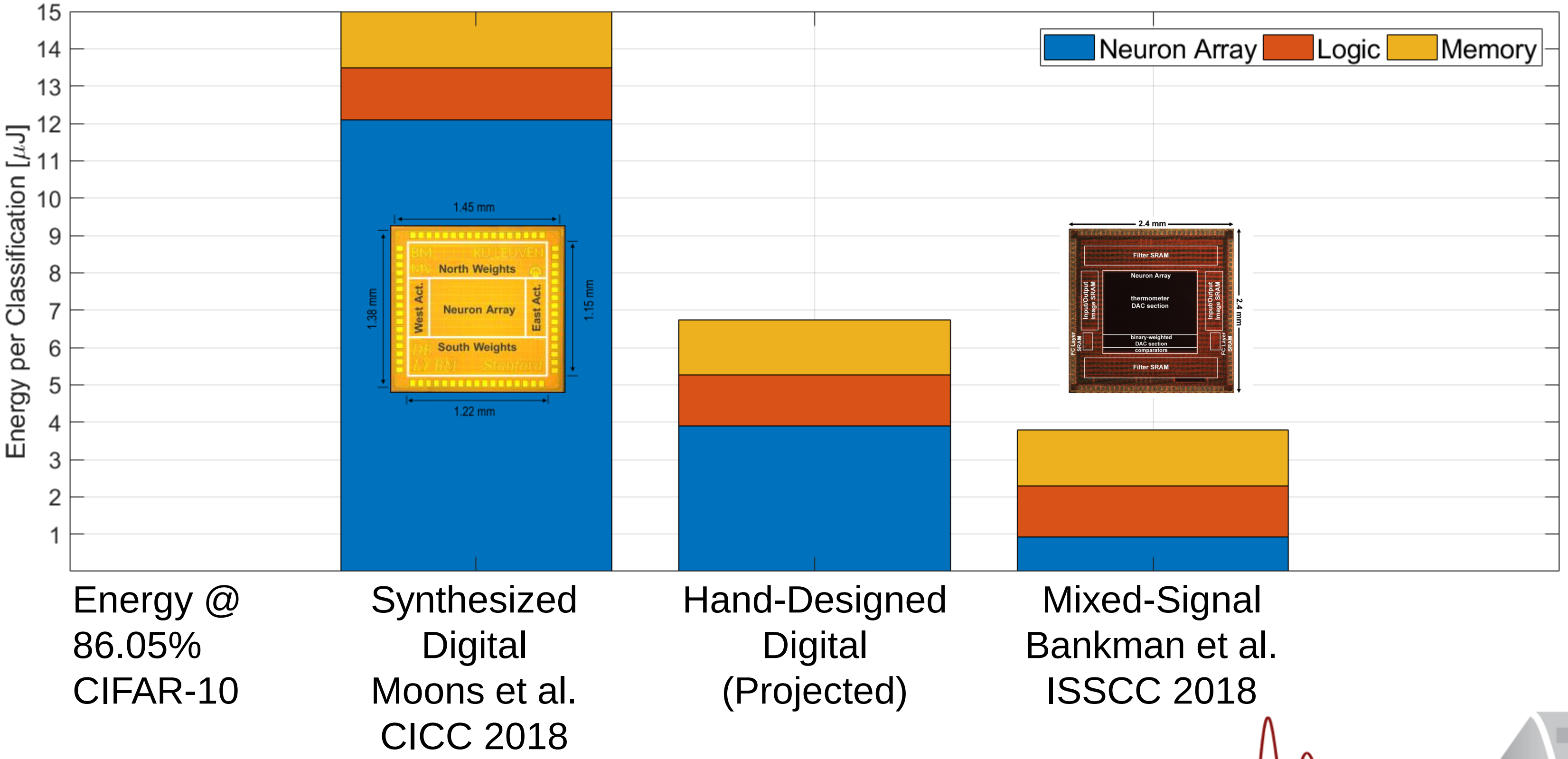


BinarEye
(Moons et al., CICC 2018)

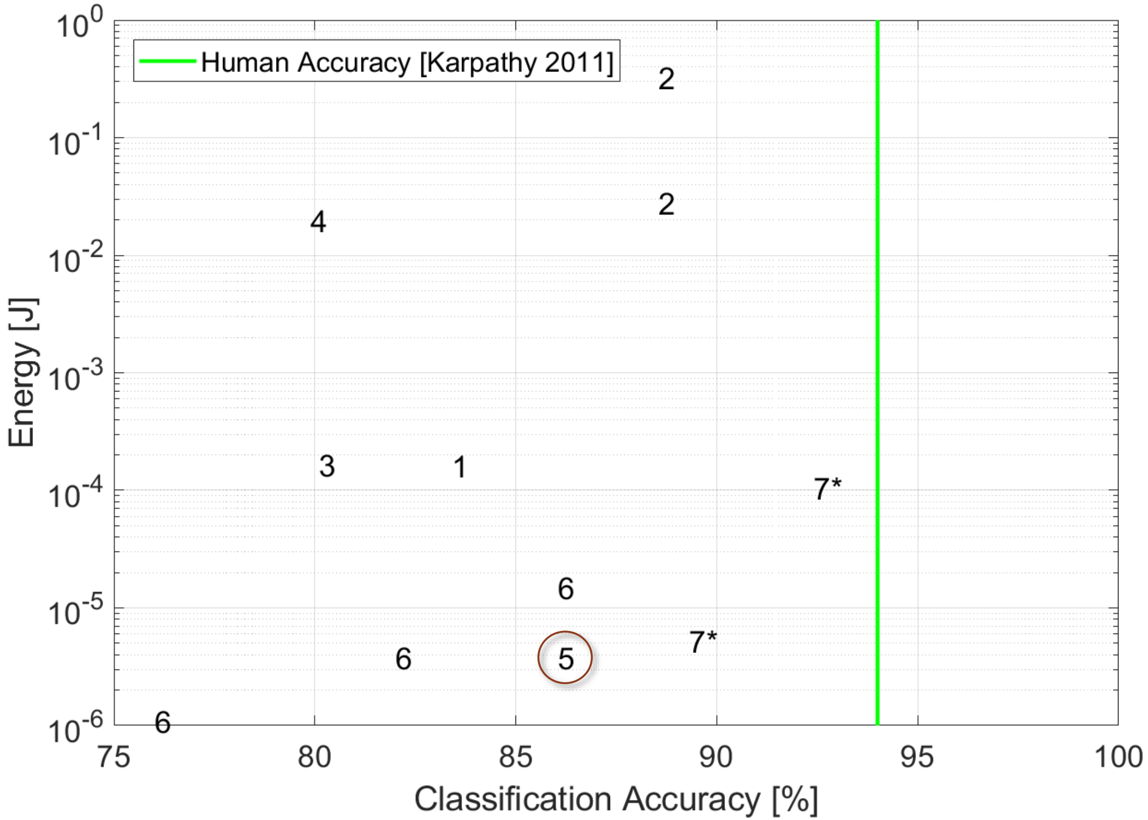
Mixed-Signal



Digital vs. Mixed-Signal Binary CNN Processor



CIFAR-10 Energy vs. Accuracy



- Neuromorphic
  - [1] TrueNorth, Esser PNAS 2016
- GPU
  - [2] Zhao FPGA 2017
- FPGA
  - [2] Zhao FPGA 2017
  - [3] Umuroglu FPGA 2017
- MCU
  - [4] CMSIS-NN, Lai arXiv 2018
- **Memory-like, mixed-signal**
  - **[5] Bankman ISSCC 2018**
- BinarEye, digital
  - [6] Moons CICC 2018
- In-memory, mixed-signal
  - [7] Jia arXiv 2018

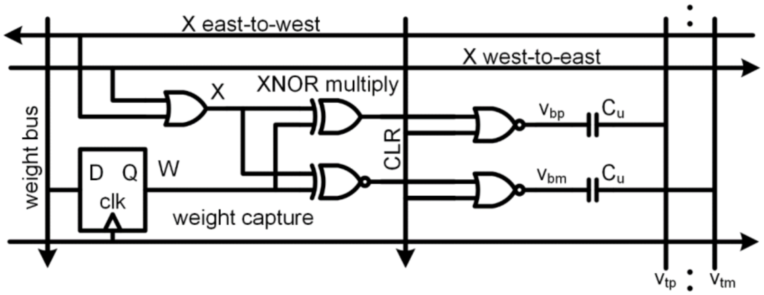
  *energy excludes off-chip DRAM

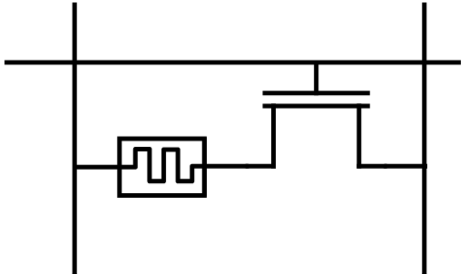
Limitations of Mixed-Signal BinaryNet

- Poor programmability
- Relatively limited accuracy (even on CIFAR-10) due to 1b arithmetic
- Energy advantage over customized digital is not revolutionary
  - Same SRAM, essentially same dataflow

- **Need a more "analog" memory system to unleash larger gains**
  - **In-memory computing**

BinaryNet Synapse versus Resistive RAM

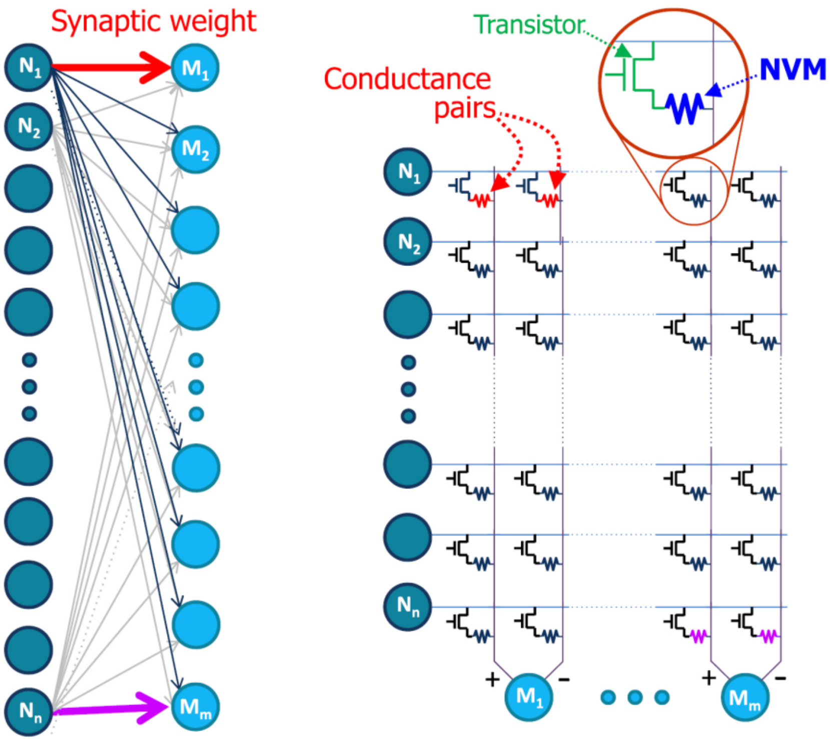




- 0.93 fJ per 1b-MAC in 28 nm
- **24107 F^2**
- Single-bit

- TBD
- **25 F^2**
- Multi-bit (?)

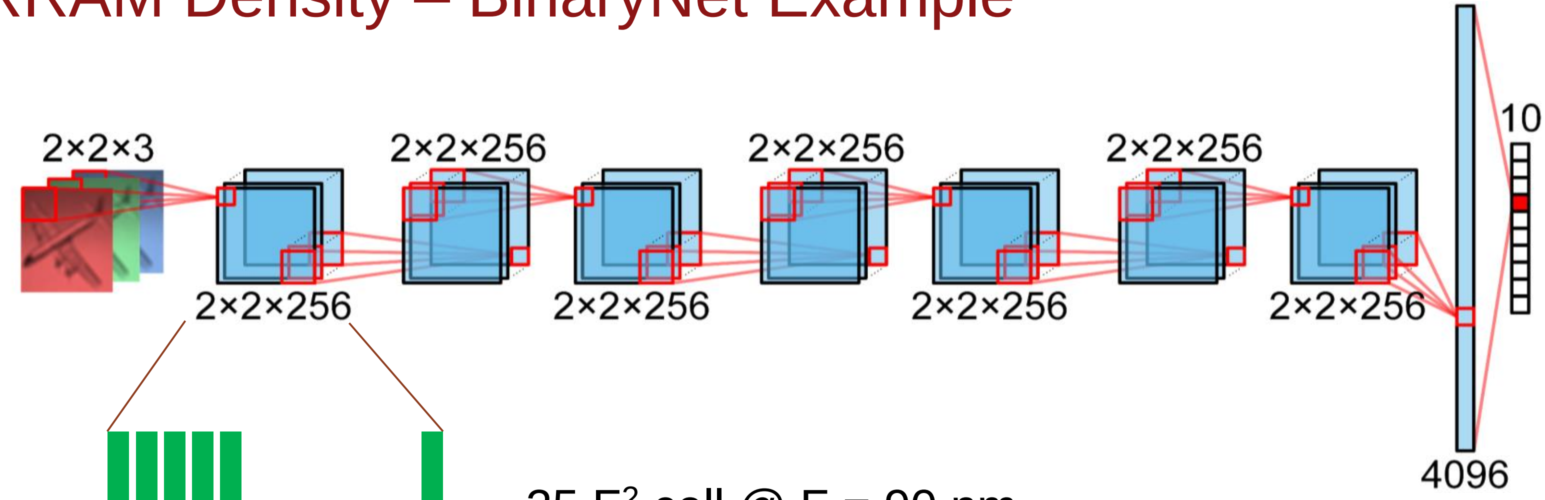
Matrix-Vector Multiplication with Resistive Cells

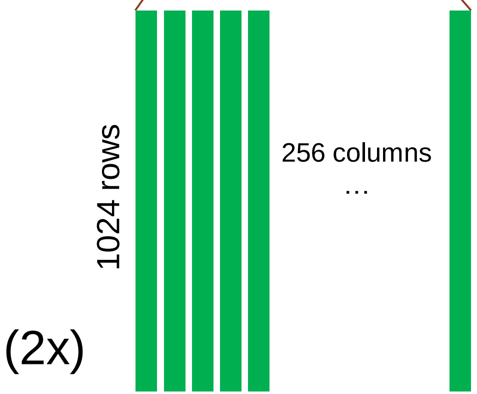


Typically use two cells to achieve pos/neg weights (other schemes possible)

Tsai, 2018

RRAM Density – BinaryNet Example



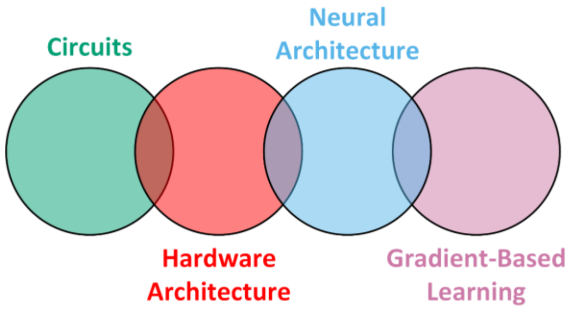


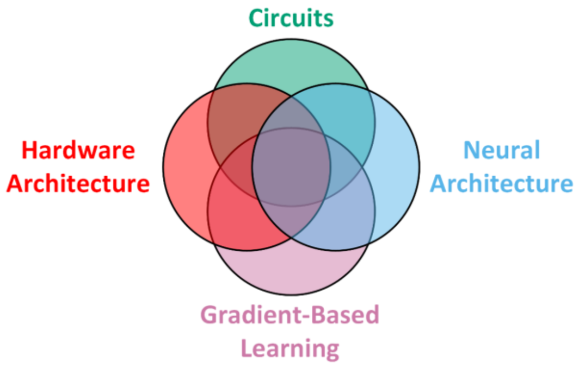
25 F^2 cell @ F = 90 nm
Side length s = 0.45 μm
2 x 1024 x 0.45 um x 256 x 0.45 um = 0.106 mm^2
per layer

0.84 mm^2 for the complete 8-layer network

Ongoing Research

- What is the best architecture?
- How many levels can be stored in each cell?
- What is the most efficient readout?
- Can we cope with nonidealities using training techniques?





(Content deleted for online distribution)

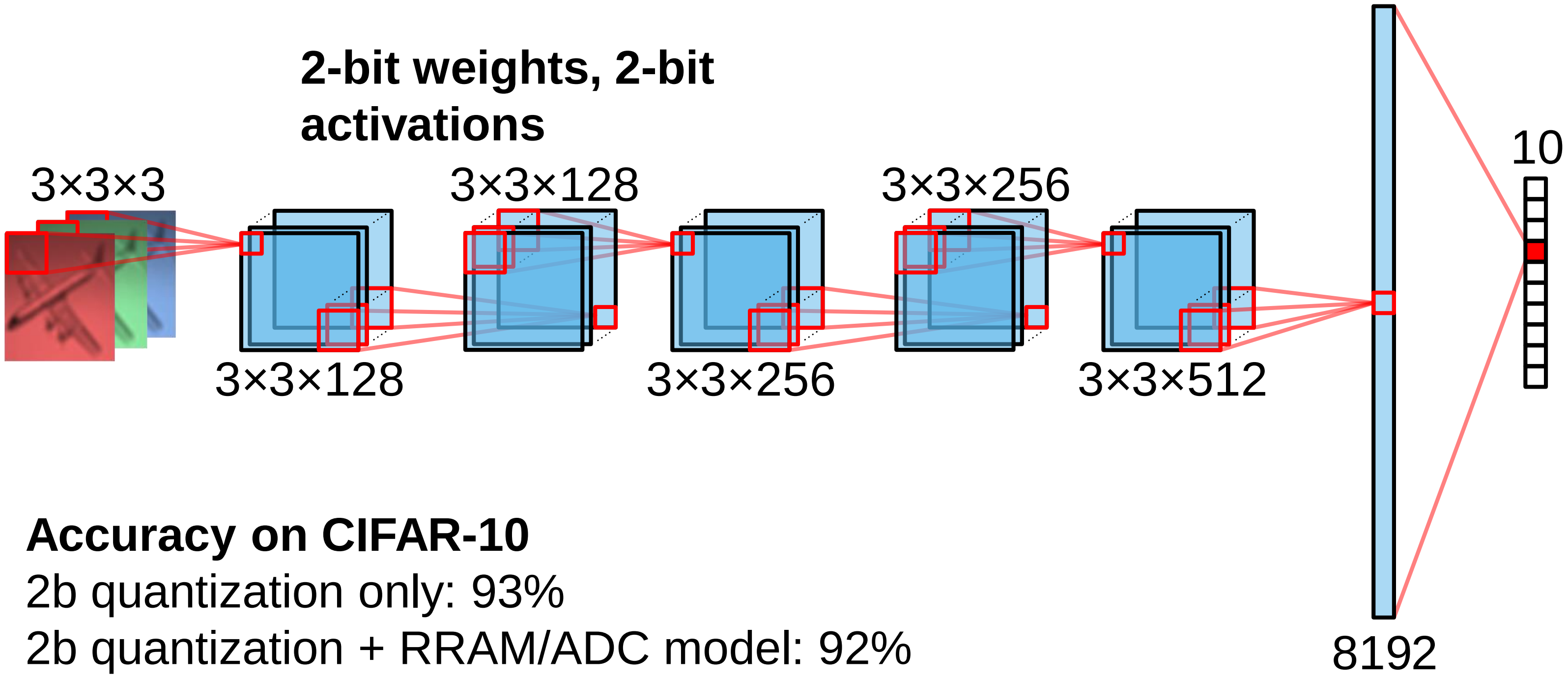
(Content deleted for online distribution)

(Content deleted for online distribution)

(Content deleted for online distribution)

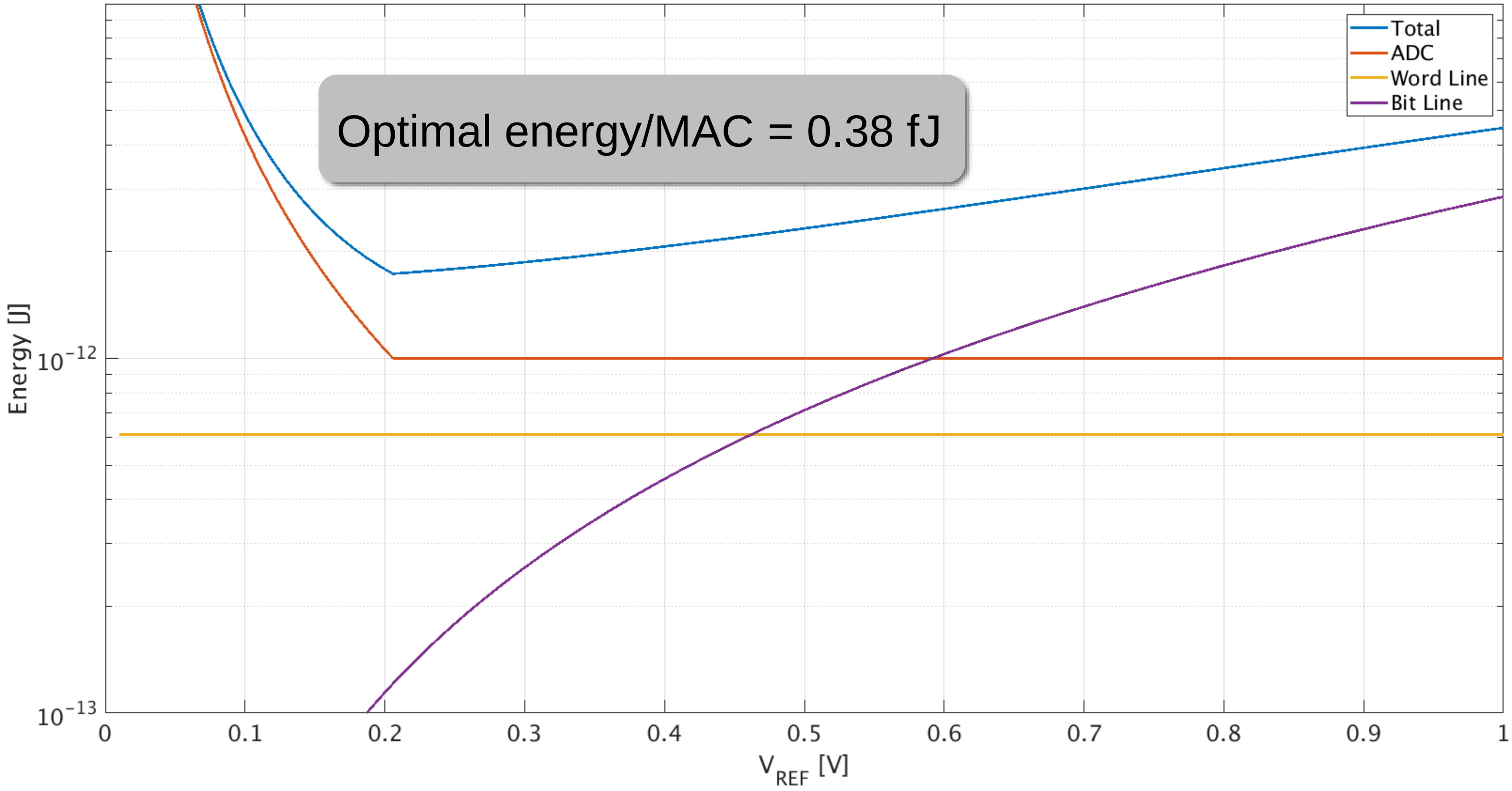
VGG-7 Experiment (**4.8 Million Parameters**)

2-bit weights, 2-bit activations



Accuracy on CIFAR-10
2b quantization only: 93%
2b quantization + RRAM/ADC model: 92%
Work in progress!

Energy Model for Column in Conv6 Layer



Summary

- Analog feature extraction is attractive for wake-up detectors
- Adding analog compute in ConvNets can be beneficial when it simultaneously lets us reduce data movement
  - In-memory analog compute looks most promising
  - Can consider SRAM or emerging memories (e.g. RRAM)
- Expect significant progress as more application drivers for “machine learning at the edge” emerge