NC STATE UNIVERSITY

3D Systems for Machine Learning

Paul Franzon

Cirrus Logic Distinguished Professor
Department of Electrical and Computer Engineering
NC State University
919.515.7351, paulf@ncsu.edu

Machine Learning Activities

Architectures

- **DARPA Chips Program**
  - ASIP PnP chiplets for machine learning
- **Cortical processor**
  - One-shot learning algorithm accelerator
  - Second part of this work
- **Modified DRAM for ML**
  - Customized 3D DRAM

ML for EDA

Center for Advanced Electronics through Machine Learning

- Applying Machine learning to EDA problems
- Back end design; DRC; Design Reuse

Outline

- **Machine Learning and Machine Intelligence**
- **Scale matters**
- **Memory-centric accelerators**
  - DNN and customized DRAM
  - Customizable 2.5D Processor
- **Conclusions**

Machine Learning to Machine Intelligence

Problem Types:

Image/Pattern Recognition

Labelled Data

"cat"

Improvement, Sequences

Inline One-time Learning

Unlabelled Data

Algorithms:

Deep / Convolutional Neural Networks

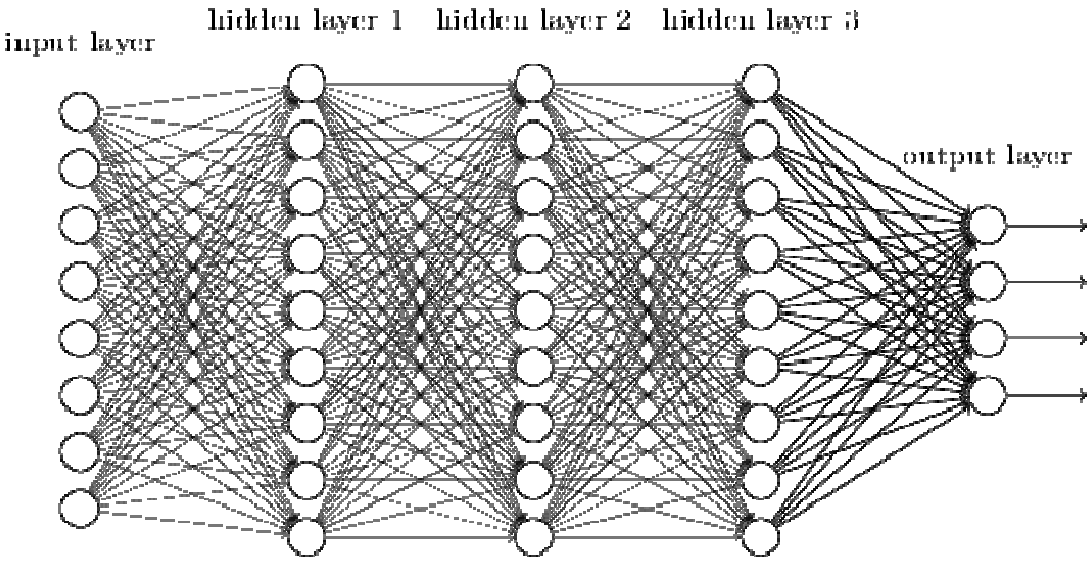
Reinforcement Learning, LSTM

Spatial Temporal Memories

Deep Networks

> **Multiple hidden layers to create needed degrees of freedom**

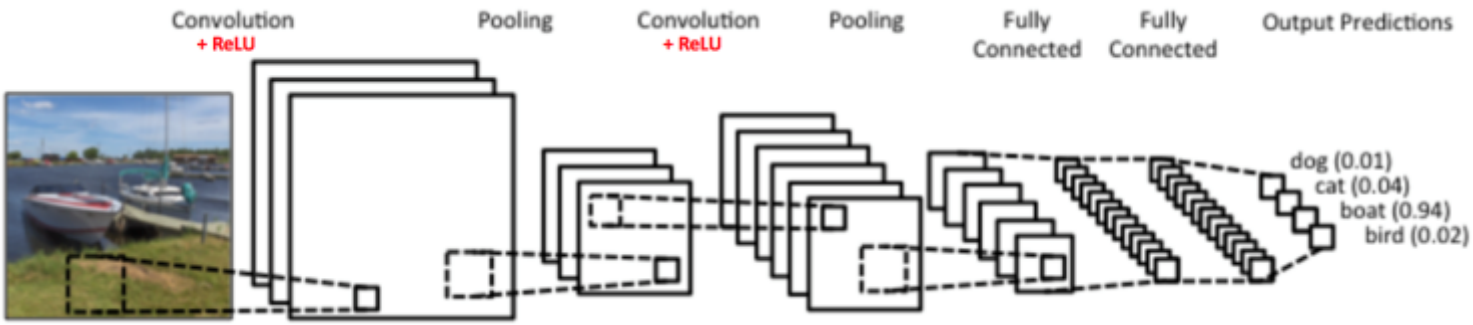
- Feed forward networks
- Fully connected network shown



Convolutional Networks

> **Space invariant pattern matching allows "step and repeat" with a single small network**

- Ie. Not all layers fully connected

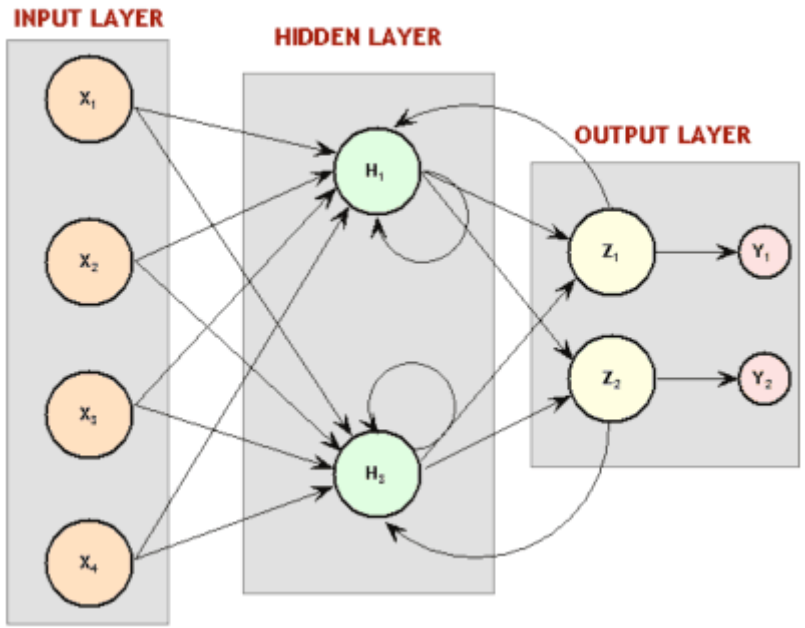


- Mix of partially connected and fully connected layers

Recurrent Networks

> **Feedback paths added in**

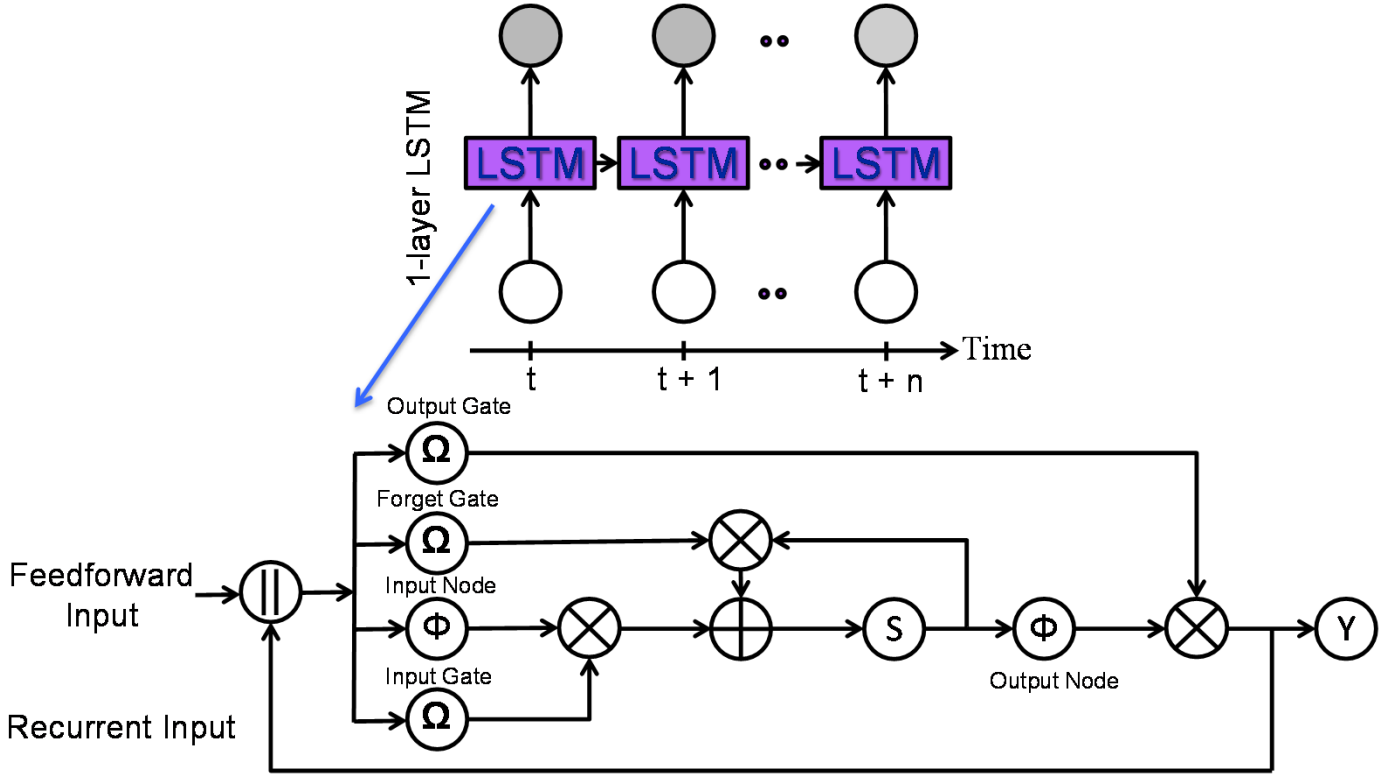
- Harder to train but works well on sequential data



Long Short Term Memory

> **Recurrent network that includes short term memory that can persist for long times**

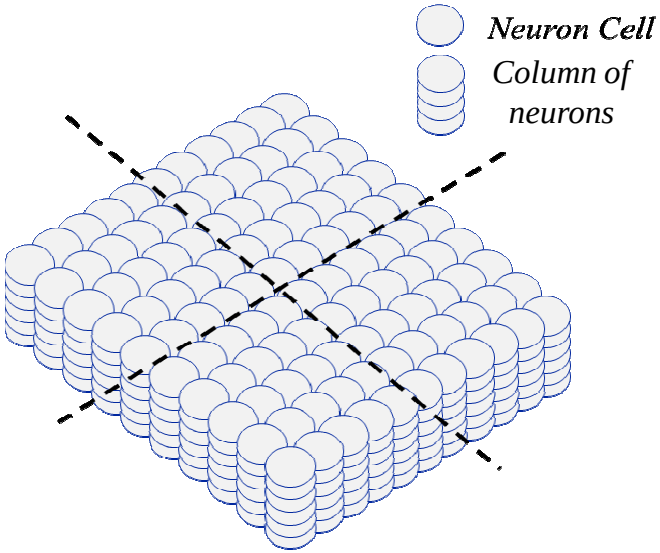
- Improves ability to deal with sequential data



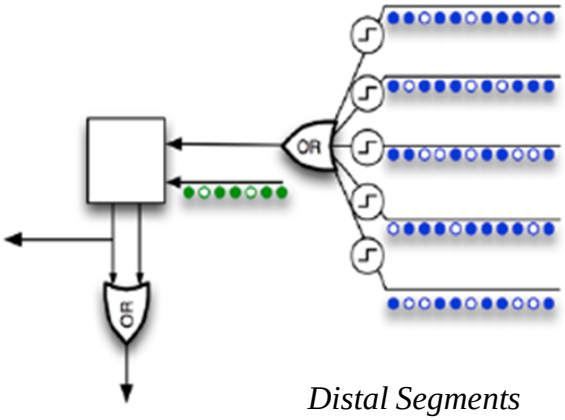
Single Cell

Numenta HTM

- Hierarchical
- Recurrent
- Sparse distributed codes
- Hetero-associative memory
- “One hot” learning
- Statically connected synapses for spatial learning and inference
- Dynamically connected synapses for temporal learning, inference, and predictions
- Predictions lead to stability
- **Highly divergent binary and integer operations**



Neuron Network



Distal Segments

D. George and B. Jaros, “The HTM Learning Algorithms,” Numenta, 2007.

Outline

- **Machine Learning and Machine Intelligence**
- **Scale matters**
- **Memory-centric accelerators**
  - DNN and customized DRAM
  - Customizable 2.5D Processor
- **Conclusions**

Typical Scales

> **Larger data sets lead to larger networks**

Problem	Complexity	Source
Single camera for autonomous vehicle	250,000 parameters	nVidia
Object identification in Video	> 1B parameters	Google
Speech Recognition (CNN)	4.5M parameters 11.7M ops	Microsoft
Speech Recognition (DNN)	8.9M parameters 8.9M ops	Microsoft
Speech Translation	151M parameters to 4B parameters	GoogleMind
Face Recognition	140M parameters	Google Facenet

Scaling for Embedded Systems

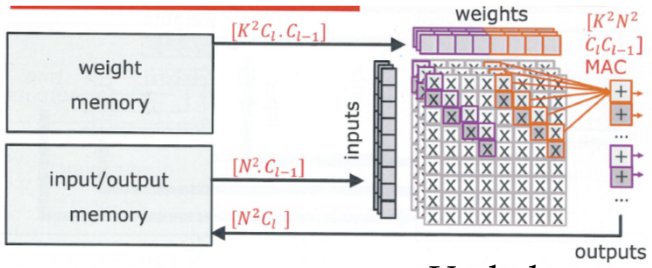
Support for multiple ANNs

Eight parallel DNN machine for self-driving car image classification[1]

Future implementations

- Support for disparate ANNs
- ~16 Gb of memory
- ~5% for weights
- ~25 Tbps for real time classification (60 fps) of multiple disparate ANNs
- ~80% weight memory traffic

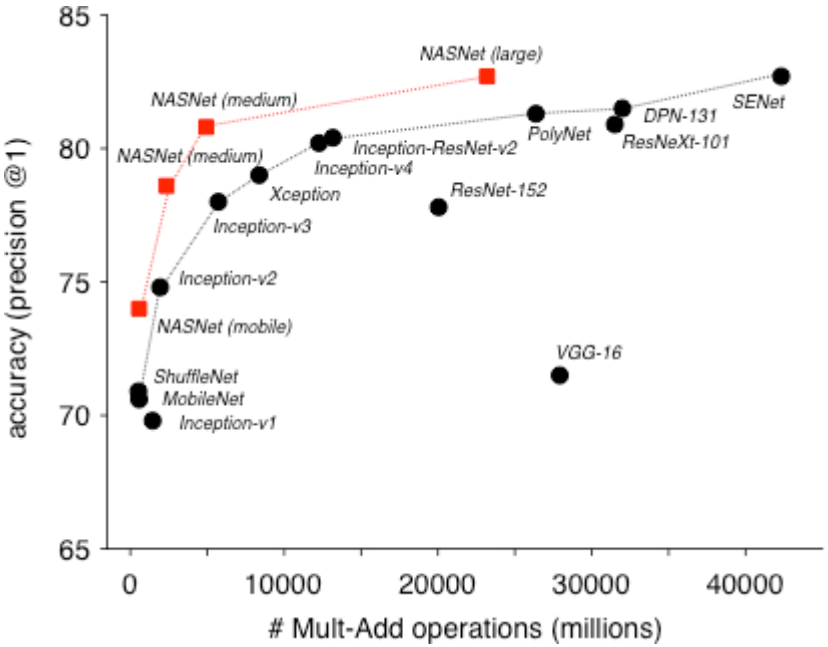
http://www.hackerboards.com

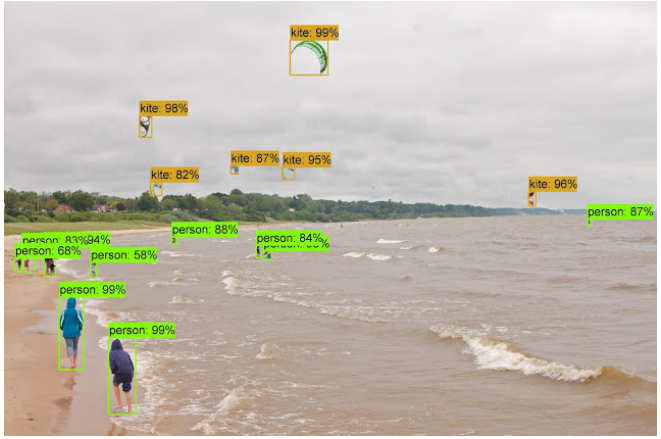


Verhelst

1 Bojarski, M. et al. "End to End Learning for Self-Driving Cars". *arXiv preprint arXiv:1604.07316* (2016).

Scale helps accuracy





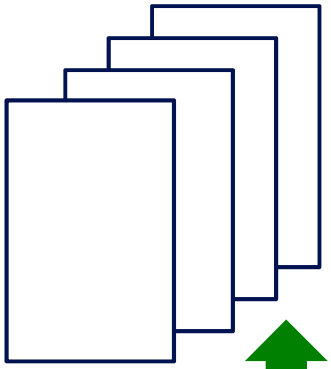
https://research.googleblog.com/2017/11/automl-for-large-scale-image.html

Outline

- **Machine Learning and Machine Intelligence**
- **Scale matters**
- **Memory-centric accelerators**
  - DNN and customized DRAM
  - Customizable 2.5D Processor
- **Conclusions**

Memory Hierarchy

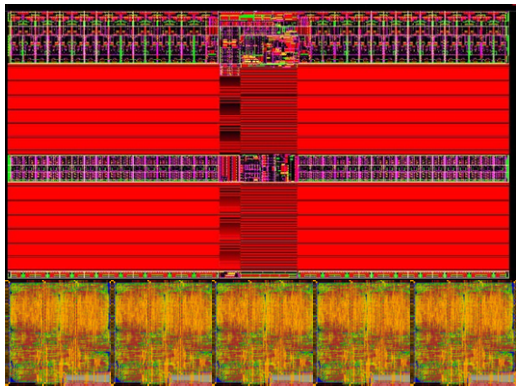
- > **Need capacity and bandwidth:**
- > **Traditional Solution: Use a memory hierarchy**



DDR DRAM
25.6 GBps/chip

A HKMG 28nm 1GHz Fully-Pipelined Tile-able 1MB Embedded SRAM IP with 1.39mm^2 per MB

Ming-Zhang Kuo, Osamu Takahashi, Ping-Lin Yang, Cheng-Chung Lin, Min-Jer Wang, Ping-Wei Wang, Sang-Hoo Dhong
Taiwan Semiconductor Manufacturing Company, Design Technology Platform, R&D, Hsinchu, Taiwan

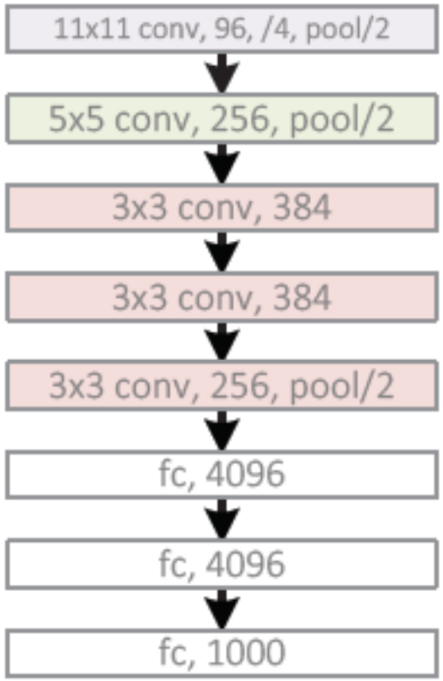


On-chip SRAM, SOA:
1.56 sq.mm/Mb
46 mW; 600 MHz; 1 MB; 144b,
76 pJ/access; 500 fJ/bit;
Inference Engine.

Machine Learning & Locality

> **Traditional Solution relies on locality**

- Does not work for Recurrent Network
- Works best for batch processing not real time edge processing

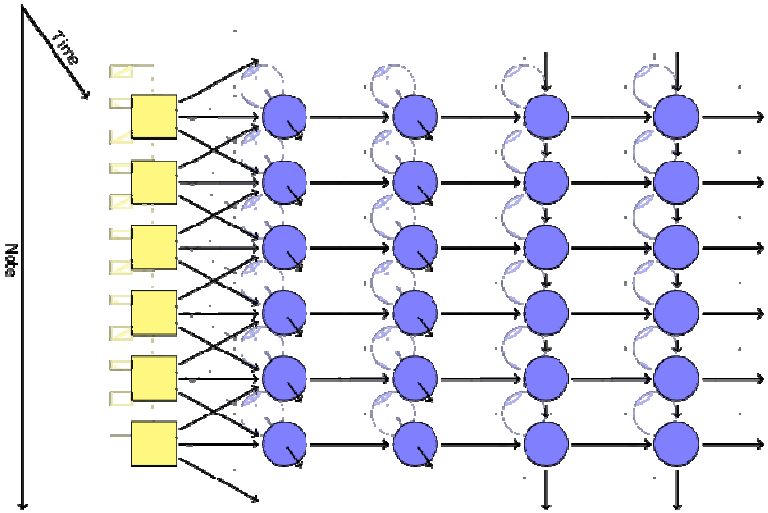


Cache weights for image (not for fc layers)

Cache weights to batch process multiple images (not for edge aps)

Machine Learning

> **Traditional Solution is useful but does not deal well with scale**

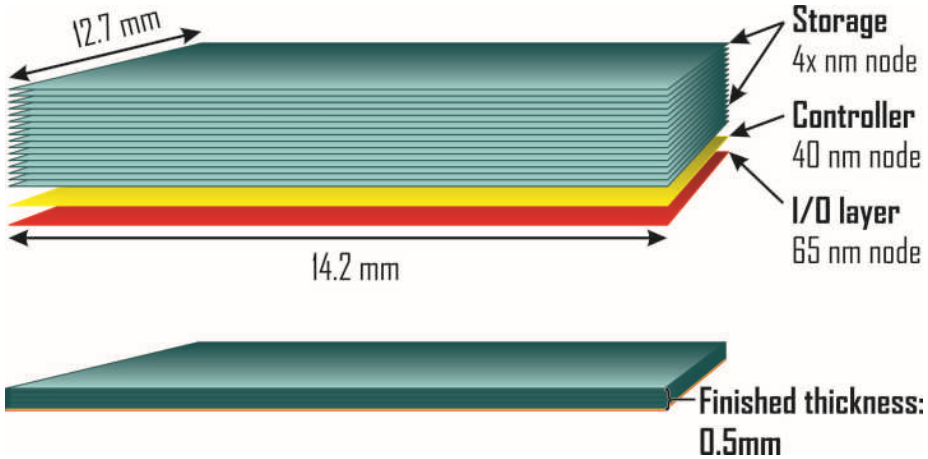


NC STATE UNIVERSITY

Tezzaron DiRAM4

> **"Dis-integrated" RAM**

- DRAM only tiers
- Sense amps, etc. on logic only tiers
- 32 bit DQ / port



Family Name	Ports	Banks per Port	Interface	Density	Data Bandwidth	Latency
DiRAM4-64C64™	64	64	0.6 – 1.3V CMOS I/O	64 Gb	4/4 Tb/s (R/W)	9 ns
DiRAM4-64C32™	64	64	0.6 – 1.3V CMOS I/O	32 Gb	4/4 Tb/s (R/W)	9 ns
DiRAM4-64C16™	64	64	0.6 – 1.3V CMOS I/O	16 Gb	4/4 Tb/s (R/W)	9 ns

Feature	Value	Comment
Capacity	64 Gbit	1 Gbit/port
Number of ports	64	Each port provides access to an individual memory
Maximum clock frequency	1 GHz	
Number of banks per port	64	
Pages per bank	4096	
Bits per page	4096	
Cycles per read	2	Burst of 2
Cycles per write	2	Burst of 2
Timing		
Name	Value	Comment
t_{POPO}	15 ns	Page open to page open
t_{POCRA}	3 ns	Page open to cache read, aligned to word address
t_{POCR}	9 ns	Page open to cache read
t_{POCW}	9 ns	Page open to cache write
t_{POPC}	9 ns	Page open to page close
t_{PCPO}	10 ns	Page close to page open
t_{CRL}	5 ns	cache read latency
t_{CWL}	−1 ns	cache write latency
Power		
Name	Value	Comment
Page open	100 pJ	
Page close	320 pJ	
Page refresh	320 pJ	
Cache read	64 pJ	
Cache write	64 pJ	
NOP	20 pJ	

Table C.1 DiRAM4 characteristics [14]

Modifications to DiRAM4

- **Standard DiRAM4**
  - 64 disjoint 1Gb memory ports
  - Standard DiRAM4 port is 32bits @ 1GHz
  - System has 64 sub-systems each operating on one memory port
  - 4 Tbs in each direction
- **Proposed Customizations for a 3D-DRAM**
  - We suggest widen to 2048 bits and use high-density TSVs
  - Entire page in one access using burst-of-2
  - Raw bandwidth ~2Tbps per port
  - Write mask since only partial writes needed

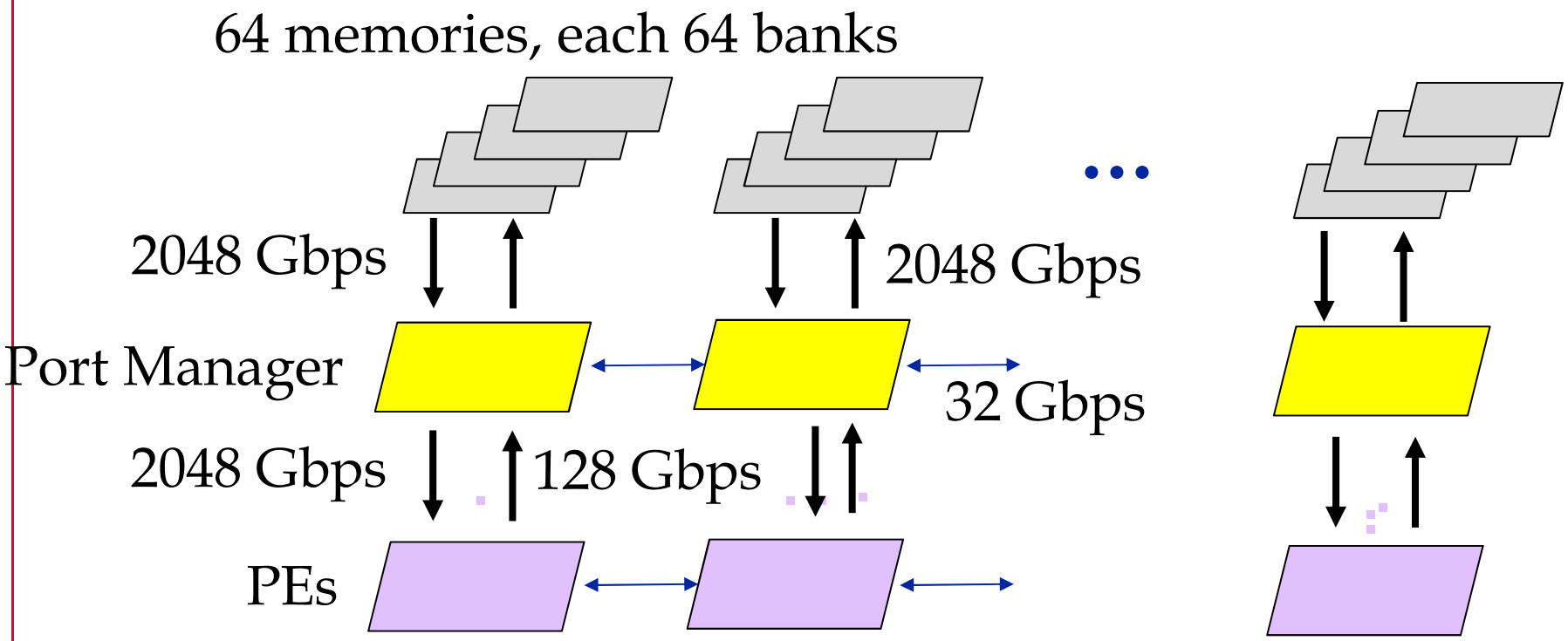
133 fJ/bit
131 Tbps bandwidth

SRAM: 500 fJ/bit
86 Gbps per port

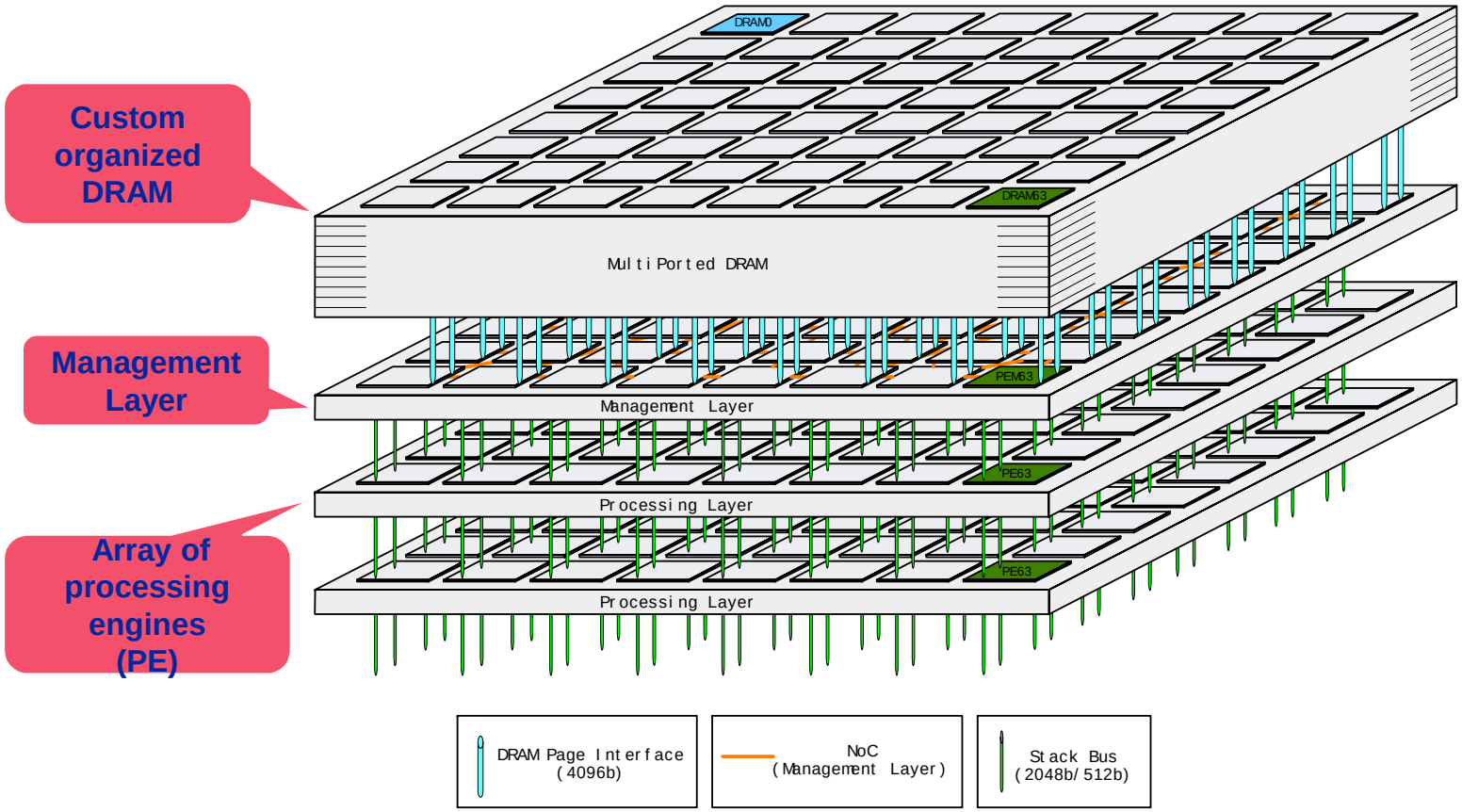
Modified DiRAM

> Collaboration with Tezzaron

- Match external bandwidth to internal bandwidth
  - Two cycle complete row access



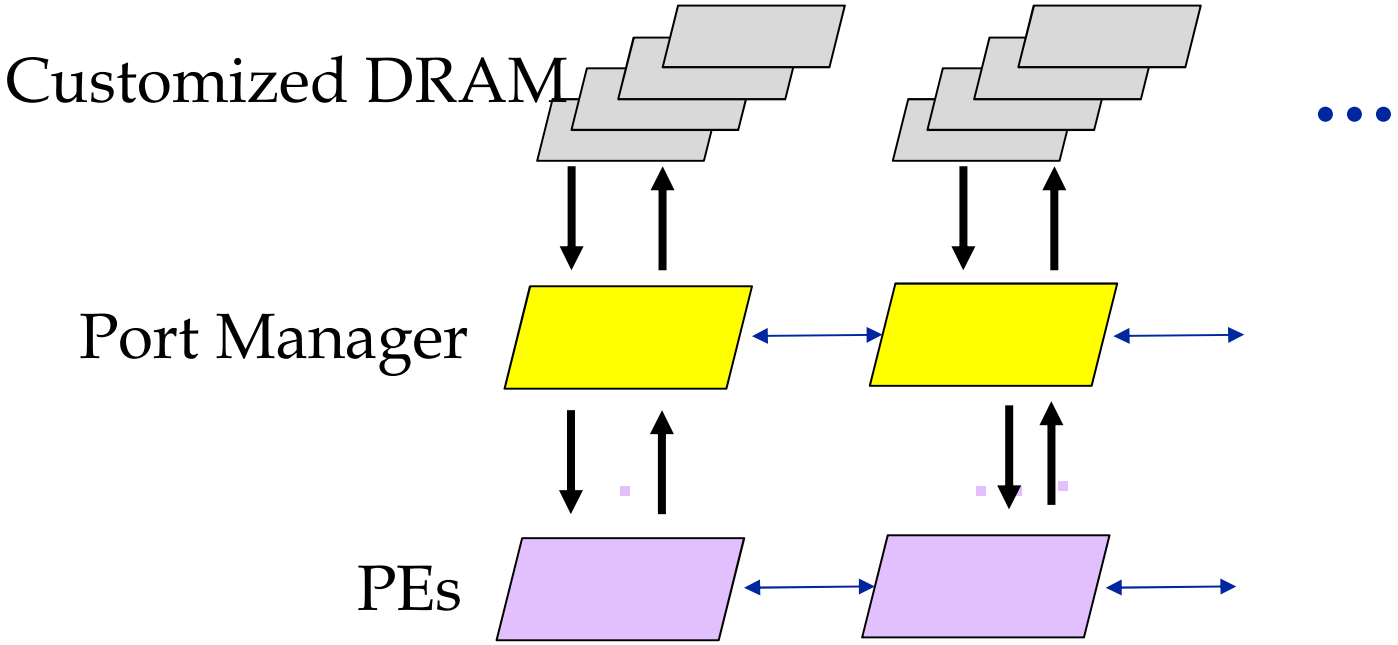
3D Configuration



Streaming Operations

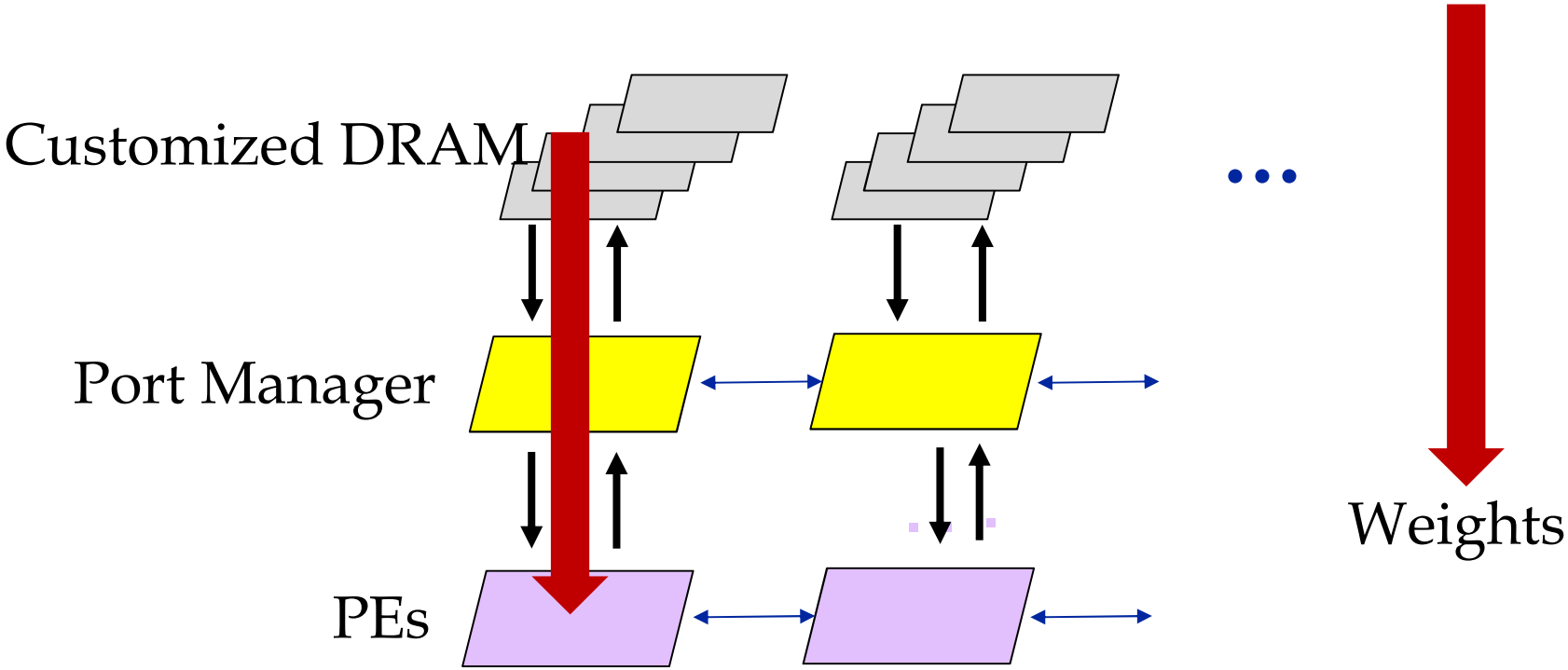
> **Weights and data streamed in predictive fashion**

- Avoids need for SRAM buffers, except for match-rating FIFOs
- Hides DRAM latency



Streaming Operations

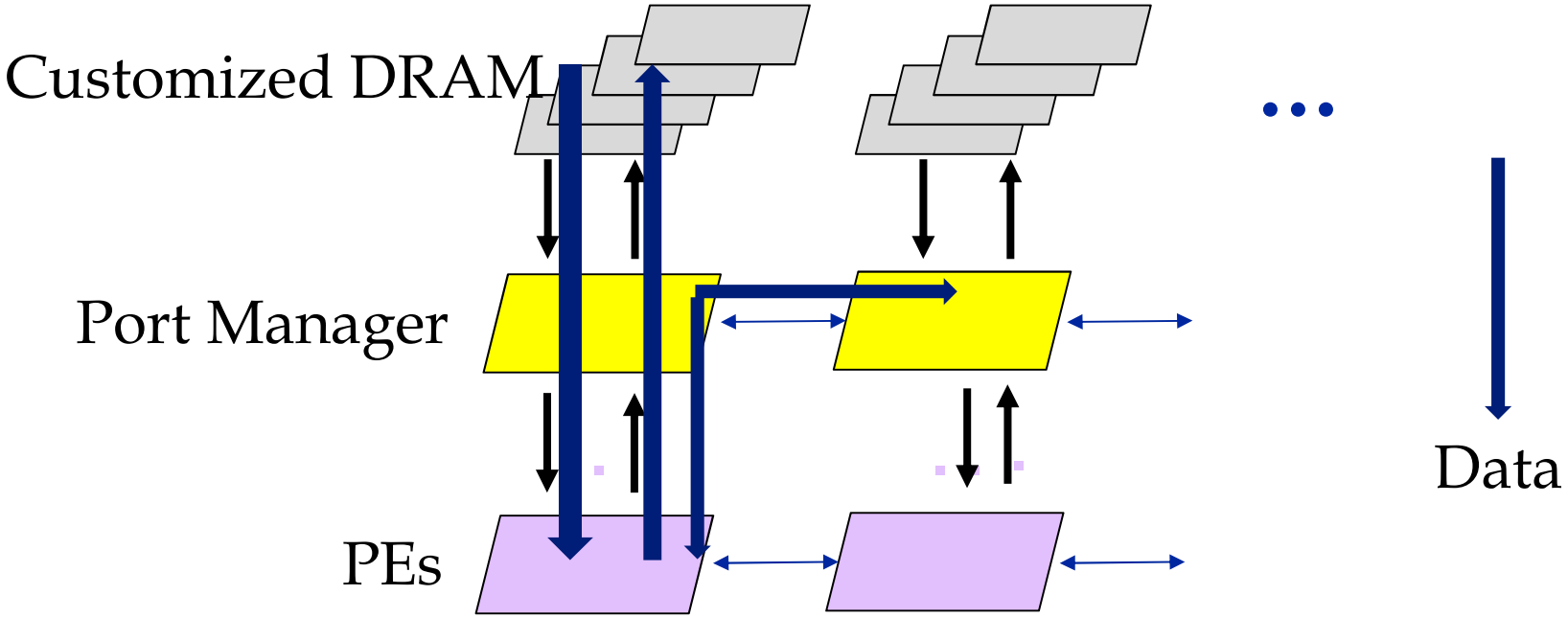
> **Weights and data streamed in predictive fashion**
- Avoids need for SRAM buffers, except for match-rating FIFOs
- Hides DRAM latency



Streaming Operations

> **Weights and data streamed in predictive fashion**

- Avoids need for SRAM buffers, except for match-rating FIFOs
- Hides DRAM latency



Design Details

- 12 x 14 mm footprint (dictated by DiRAM)
- Completely designed, synthesized and extracted
  - Assumed 5 um pitch TSVs
  - 2500 to/from PEs (0.06 sq.mm)
  - 4200 to/from DRAM (0.105 sq.mm)

Block	Power (W)
Manager	42.55
PE	26.50
DRAM	4.51
DRAM TSVs	1.14
Stack Bus TSVs	0.74
Total	**75.44**

Comparison with State of Art

> **Power and area needed to match inference throughput capability of this solution**

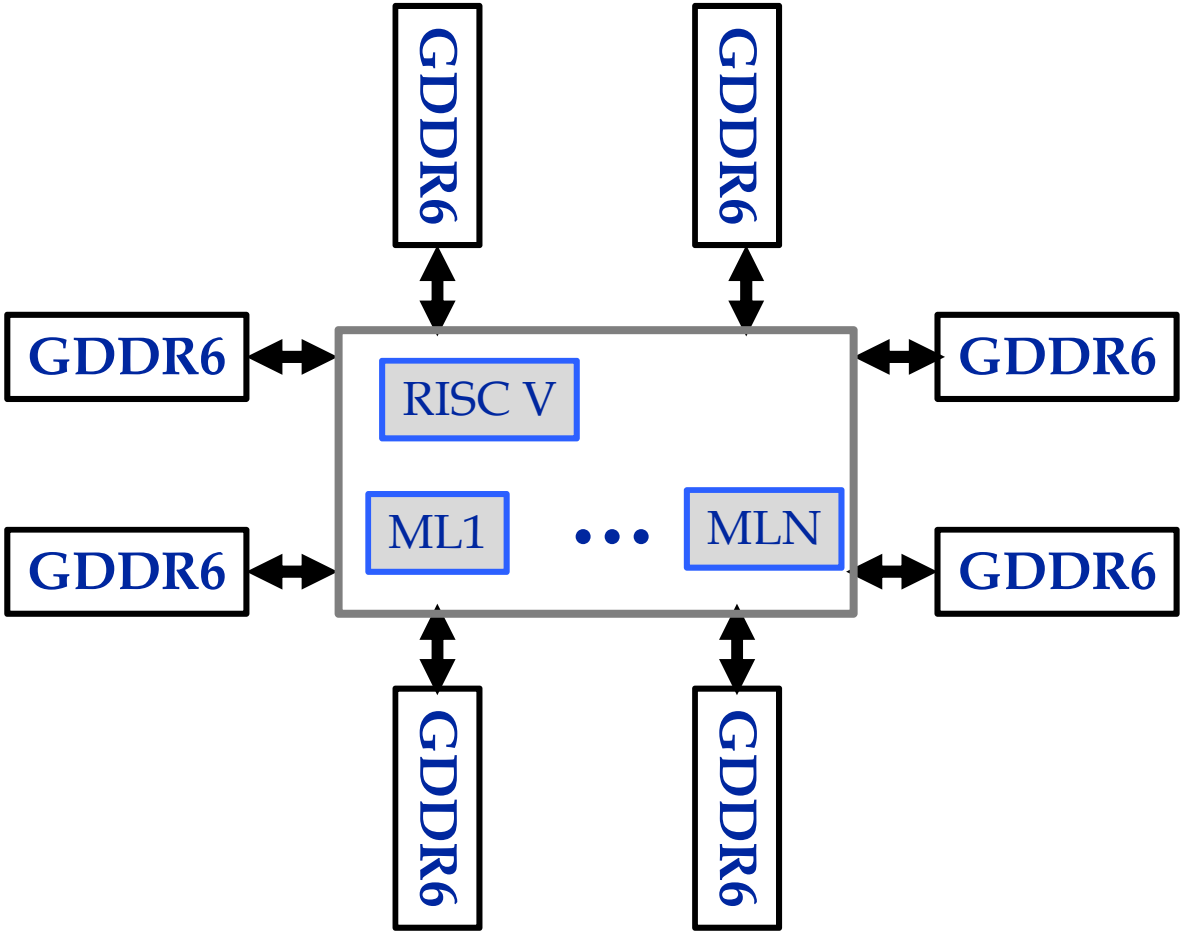
	Power (W)	Area (sq.mm)
This work	75	175
Research SOA	224	1,096

Outline

- **Machine Learning and Machine Intelligence**
- **Scale matters**
- **Memory-centric accelerators**
  - DNN and customized DRAM
  - Customizable 2.5D Processor
- **Conclusions**

Overall Approach

> **Customizable Interposer for Scalable Tasks**

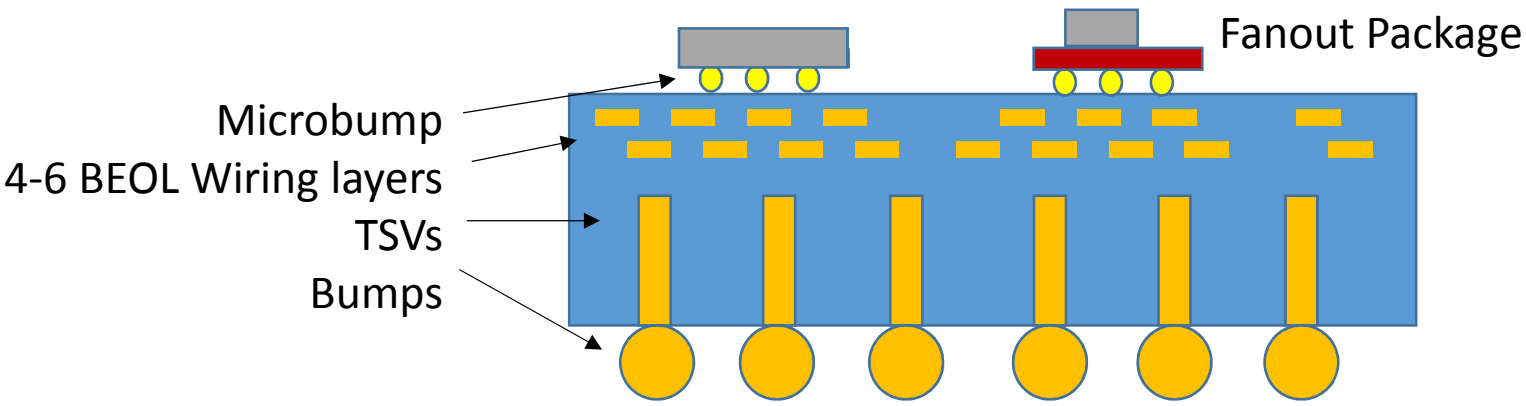


8 Gb
32 x 12 Gbps

Total:
8 GB
384 GBps

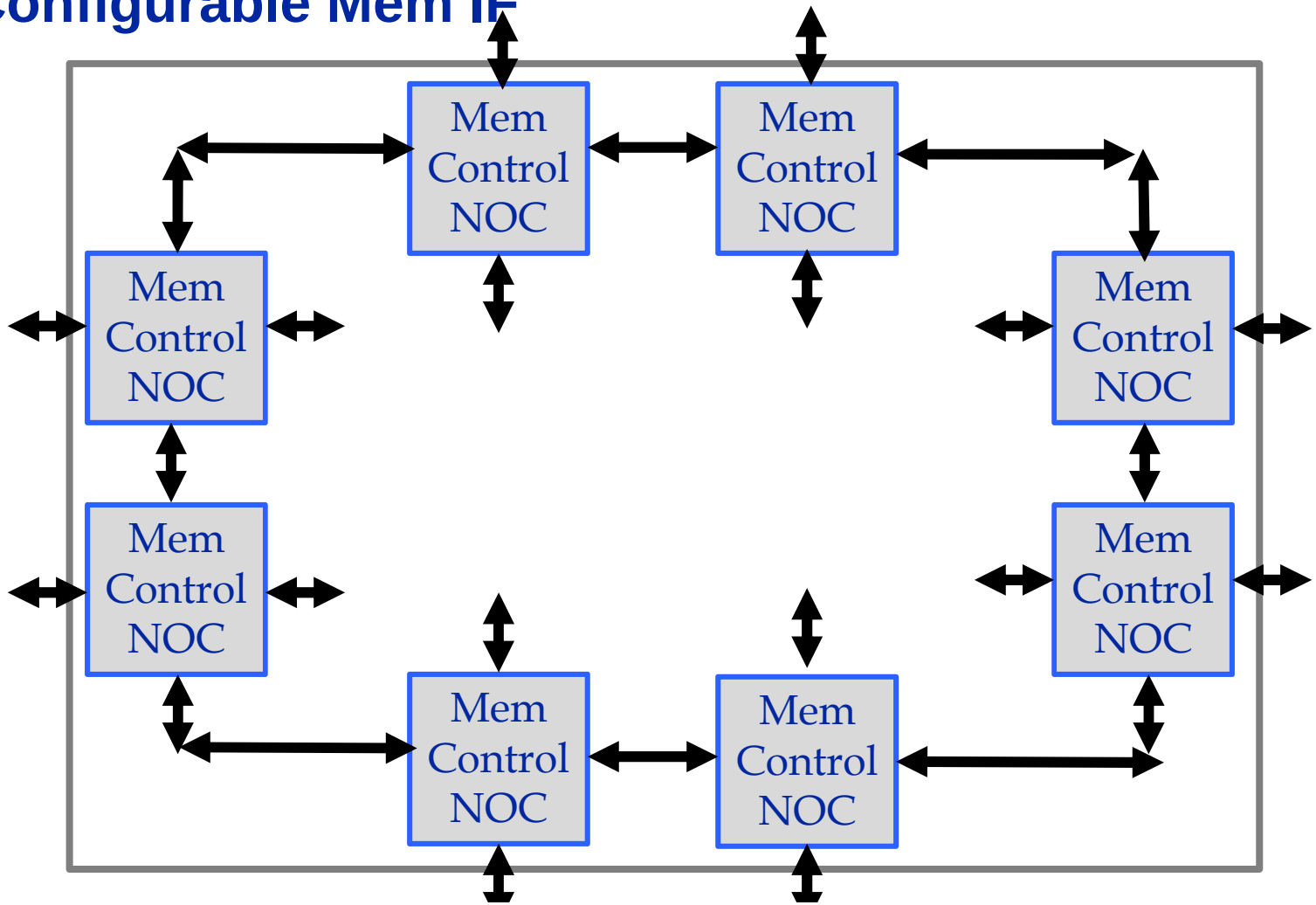
Interposer

> **Fanout package used to support high pin count small ICs**



Interposer Design – Memory Interfaces

> **Configurable Mem IF**



ML Scale to Task

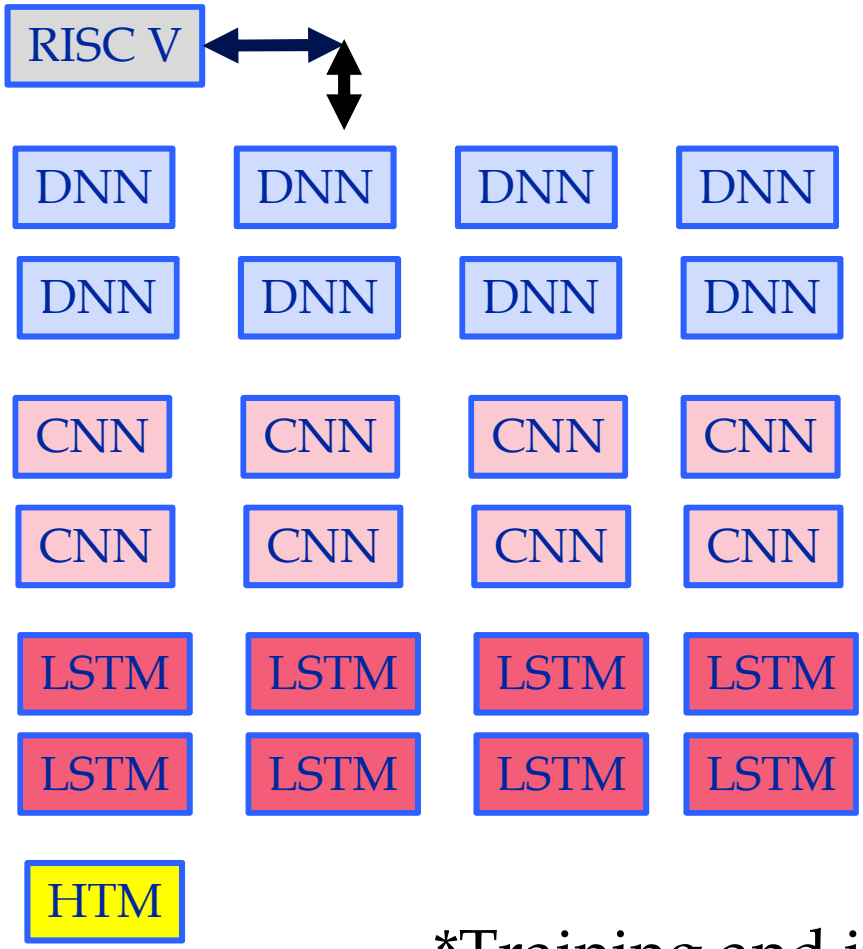
CPU

Up to 8 FC DNN accelerators

Up to 8 CNN accelerators

Up to 8 LSTM accelerators*

HTM accelerator*



+
8 GB
384 GBps
In
8 channels

*Training and inference

Application

> **Video description**

- HTM for anomaly detection

Long-term Recurrent Convolutional Networks for Visual Recognition and Description

Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, Trevor Darrell

arXiv:1411.4389v4 [cs.CV] 31 May 2016

Abstract— Models based on deep convolutional networks have dominated recent image interpretation tasks; we investigate whether models which are also recurrent are effective for tasks involving sequences, visual and otherwise. We describe a class of recurrent convolutional architectures which is end-to-end trainable and suitable for large-scale visual understanding tasks, and demonstrate the value of these models for activity recognition, image captioning, and video description. In contrast to previous models which assume a fixed visual representation or perform simple temporal averaging for sequential processing, recurrent convolutional models are "doubly deep" in that they learn compositional representations in space and time. Learning long-term dependencies is possible when nonlinearities are incorporated into the network state updates. Differentiable recurrent models are appealing in that they can directly map variable-length inputs (e.g., videos) to variable-length outputs (e.g., natural language text) and can model complex temporal dynamics; yet they can be optimized with backpropagation. Our recurrent sequence models are directly connected to modern visual convolutional network models and can be jointly trained to learn temporal dynamics and convolutional perceptual representations. Our results show that such models have distinct advantages over state-of-the-art models for recognition or generation which are separately defined or optimized.

1 Introduction

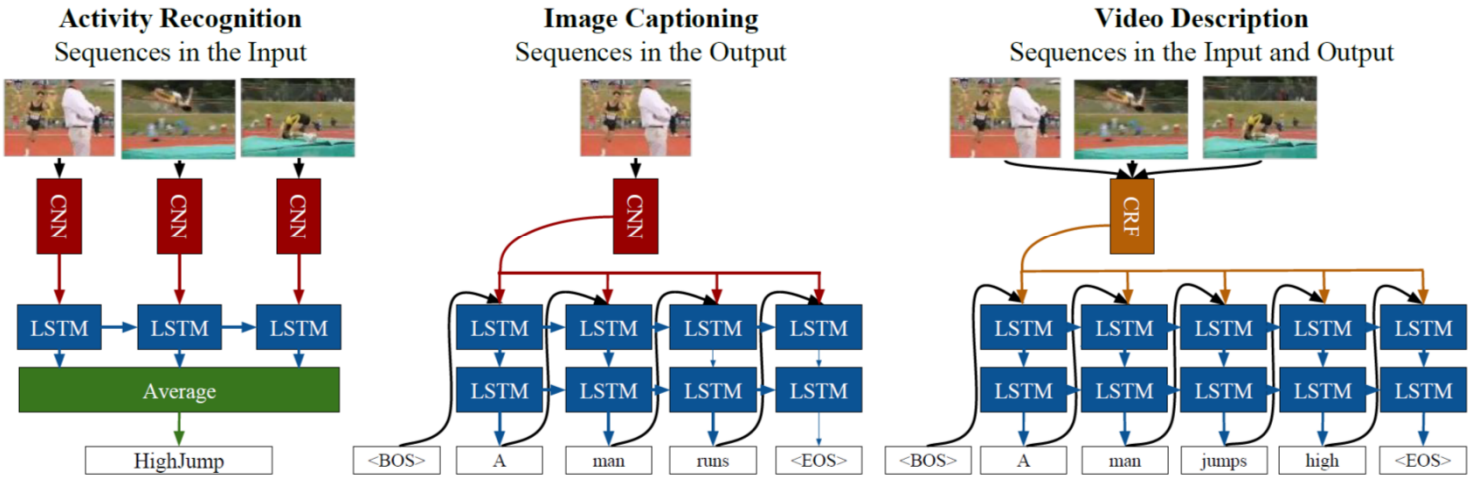
Recognition and description of images and videos is a fundamental challenge of computer vision. Dramatic progress has been achieved by supervised convolutional neural network (CNN) models on image recognition tasks, and a number of extensions to process video have been recently proposed. Ideally, a video model should allow processing of variable length input sequences, and also provide for variable length outputs, including generation of full-length sentence descriptions that go beyond conventional one-versus-all prediction tasks. In this paper we propose *Long-term Recurrent Convolutional Networks* (LRCNs), a class of architectures for visual recognition and description which combines convolutional layers and long-range temporal recursion and is end-to-end trainable (Figure 1). We instantiate our architecture for specific video activity recognition, image caption generation, and video description tasks as described below.

Research on CNN models for video processing has considered learning 3D spatio-temporal filters over raw sequence data [1], [2], and learning of frame-to-frame representations which incorporate instantaneous optic flow or trajectory-based models aggregated over fixed windows or video shot segments [3], [4]. Such models explore two extrema of perceptual time-series representation learning: either learn a fully general time-varying weighting, or apply

• J. Donahue, L. A. Hendricks, M. Rohrbach, S. Guadarrama, and T. Darrell

Fig. 1. We propose *Long-term Recurrent Convolutional Networks* (LRCNs), a class of architectures leveraging the strengths of rapid progress in CNNs for visual recognition problems, and the growing desire to apply such models to time-varying inputs and outputs. LRCN processes the (possibly) variable-length visual input (left) with a CNN (middle-left), whose outputs are fed into a stack of recurrent sequence models (*LSTMs*, middle-right), which finally produce a variable-length prediction (right). Both the CNN and LSTM weights are shared across time, resulting in a representation that scales to arbitrarily long sequences.

Fig. 6. Image description: images with corresponding captions generated by our finetuned LRCN model. These are images 1-12 of our randomly chosen validation set from COCO 2014 [33]. We used beam search with a beam size of 5 to generate the sentences, and display the top (highest likelihood) result above.



Conclusions

Deep Neural Networks

- Real applications results in large networks
  - 100M+ weights
- SRAM is useful but DRAM backing is needed
- Better solution: Modified 3D DRAM that rivals SRAM power efficiency and bandwidth
- Can hide DRAM latency

	Bandwidth	Capacity	Energy/bit	latency
SOA SRAM	N * 86 Gbps	N * 1 Mb	500 fJ/bit	1.2 ns
Modified DiRAM4	130 Tbps	64 Gb	133 fJ/bit	15 ns

Configurable 2.5D Accelerator

Acknowledgements

Team members: Lee Baker, Sumon Dey,
Weifu Li, Steve Lipa, Josh Schabel, Josh Stevens

Funding: DARPA Google AIR FORCE RESEARCH LABORATORY