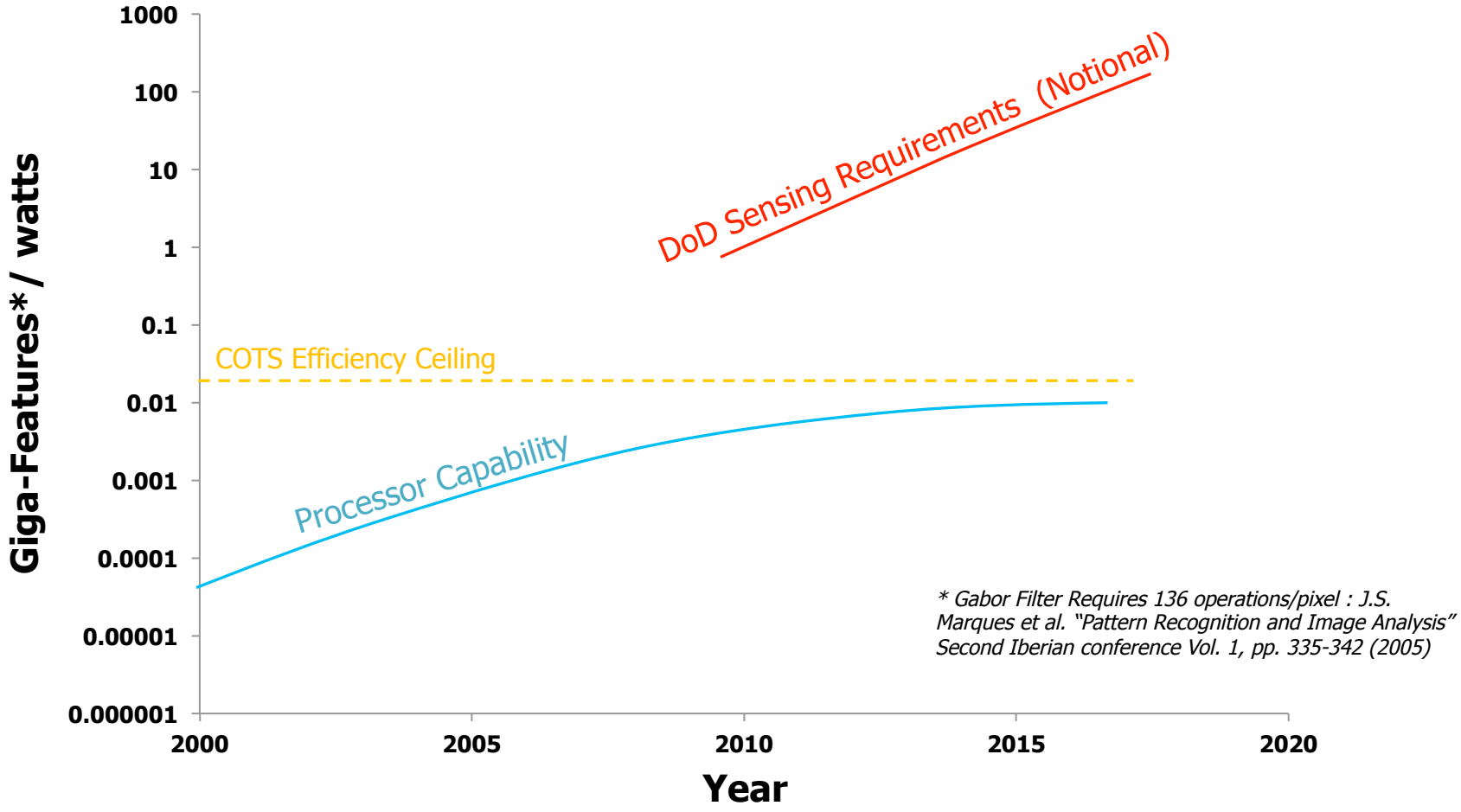
UPSIDE

Unconventional Processing of Signals
for Intelligent Data Exploitation

Dr. Dan Hammerstrom
Program Manager / MTO

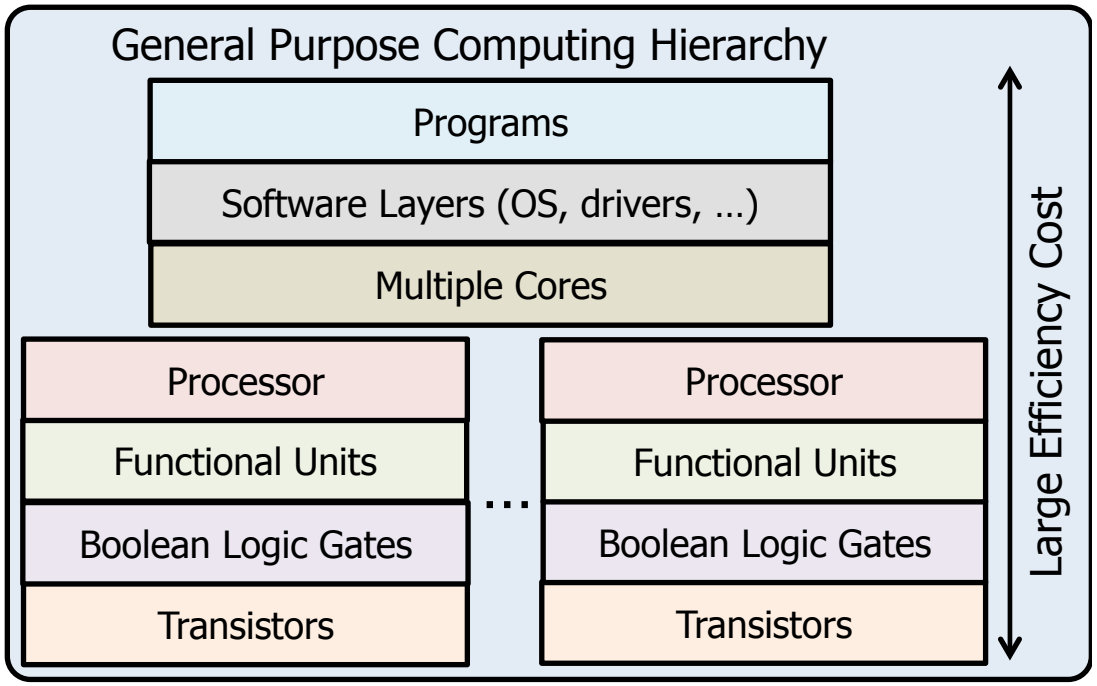
Problem: The Computer Efficiency Gap Is Increasing

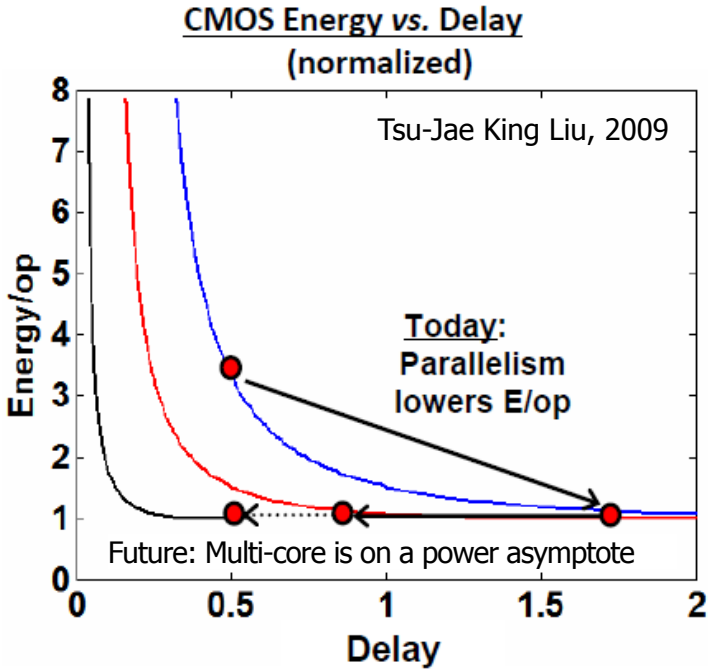


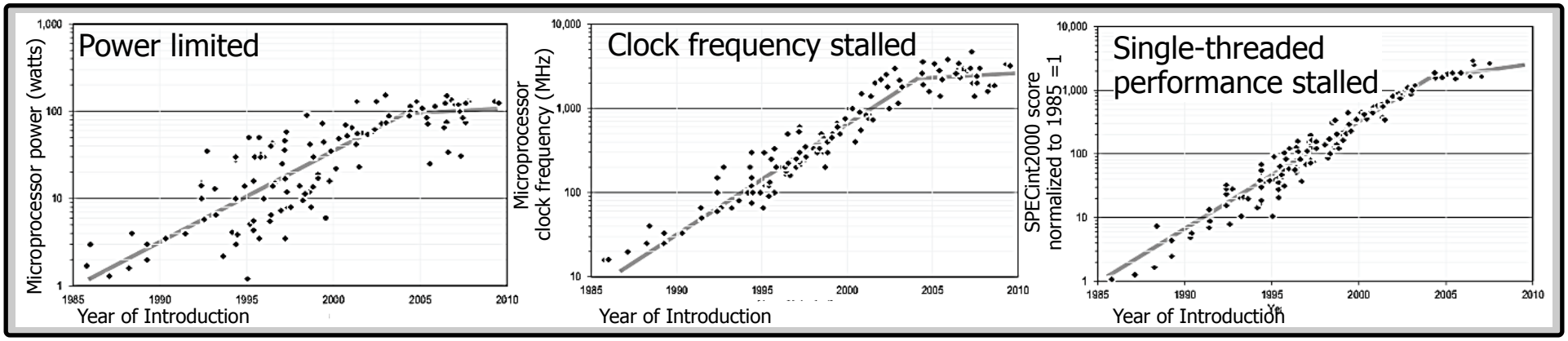
** Gabor Filter Requires 136 operations/pixel : J.S. Marques et al. "Pattern Recognition and Image Analysis" Second Iberian conference Vol. 1, pp. 335-342 (2005)*

Data increasing, analysis and compression requirements increasing …

Problem: Performance Advances Of General Purpose Computing Stalled

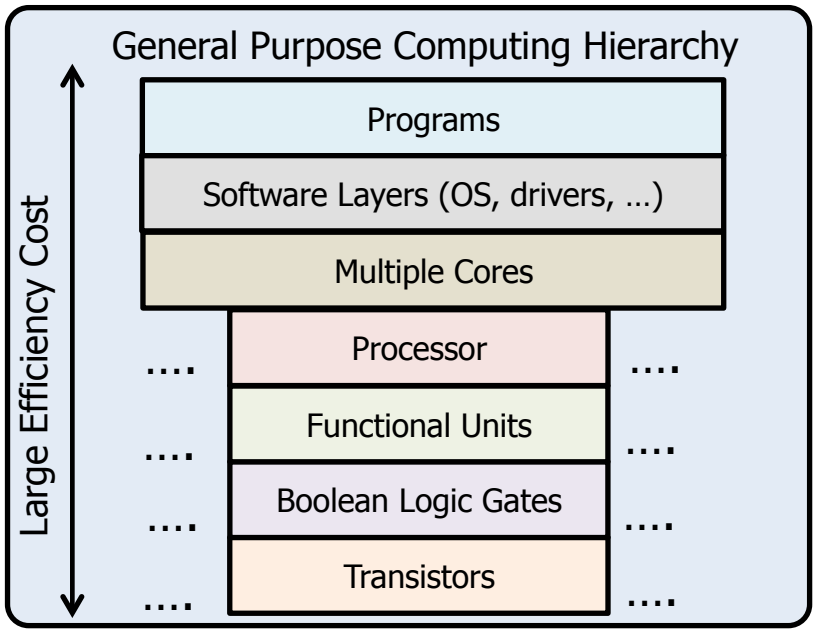






Approved for public release; distribution is unlimited.

Solution: Non-Digital, Probabilistic Computing Reduces Hierarchy

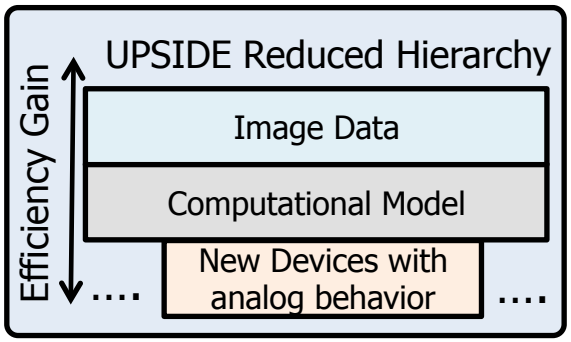


Digital architectures are not well matched to feature extraction from sensor images

- Images are inherently analog
- Digital algorithms are created to search for image structures based on existing digital number crunching architectures
- Digital abstractions limit data analysis
  - 100s to 1,000s of digital operations per image activity
  - Wasted energy in excess operations, data movement and precision

Need new computing approaches matched to image processing

- Use the physics of new emerging devices to extract features.
- Data naturally represented in sparse form are more suitable to devices and efficient for data transfer

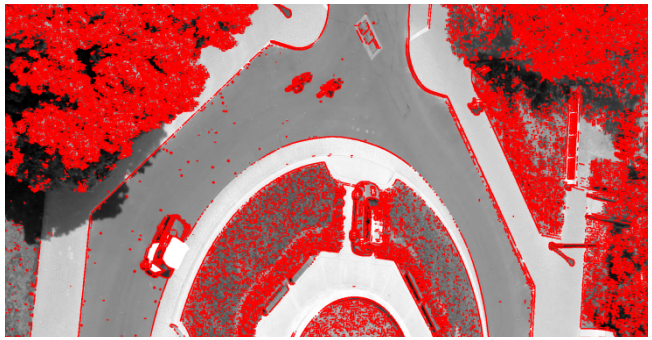


Computing directly with devices eliminates multiple layers of hierarchy/inefficiency

Solution: A New Computational Model and Implementation

New Paradigm – Non-Boolean, Probabilistic Computing

1. Computing occurs by the physics of the devices (highly parallel)
2. Devices perform the computational equivalent of hundreds of discrete digital operations
3. The model can be configured into hierarchies that accomplish most of the computational work required by the application



Sensor Data: Active Edges located (in red)

Example: Find Features in Sensor Data (7x7 Gabor Edge Finding, 10 Giga-pixel Array)

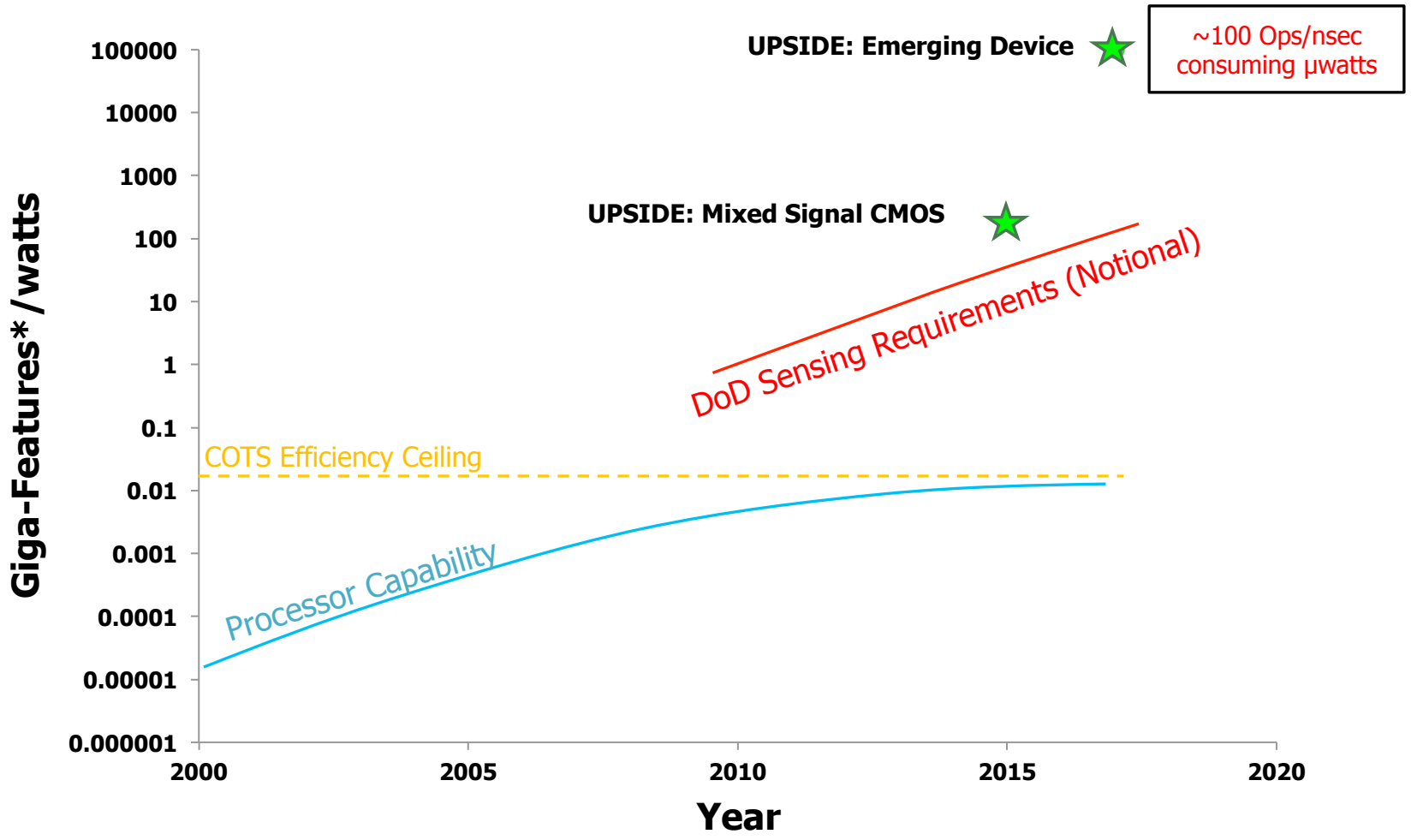
Boolean Computation

- Processor: Intel 6 Core i7, GOPS: 6.7
- 1 inference is 140 operations/kernel, 24 kernels are compared / pixel
- GOPs/watt: 0.1
- Compute time = 7,700 sec
- **460 kilo-joules** (60 watts for 7700 seconds)

Analog Direct Device Computation

- Processor: 10 X 10 Array of coupled oscillators Giga-Inferences/sec = 400 (56k GOPS equivalent)
- Compute time = 0.04 sec
- **430 milli-joules**

UPSIDE: Performance Goals



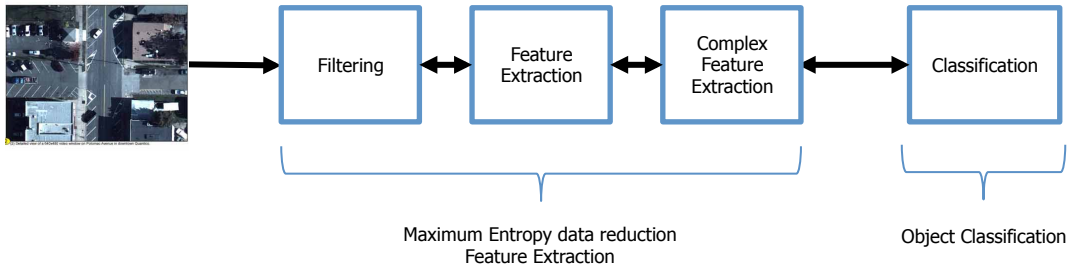
UPSIDE Goals: 3 orders of magnitude in throughput, 4 orders of magnitude in power efficiency, no loss in accuracy

UPSIDE Program Tasks

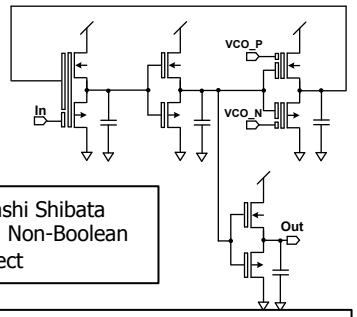
Image Processing Pipeline
An application driver

- Recreate the traditional image processing pipeline (IPP) hierarchies of Inference Modules (IM)

UPSIDE Task 1
IM Development & IPP Implementation



UPSIDE Task 2
Mixed signal CMOS implementation of computational model and system test bed

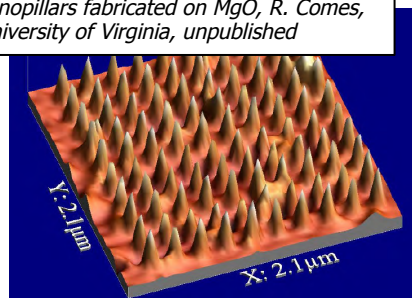


Tadashi Shibata
Intel Non-Boolean Project

- Design and Fabricate a mixed signal representation of the Inference Module in state of the art MS CMOS
- Implement a test bed system using MS CMOS
- Validate against IPP simulation

UPSIDE Task 3
Image processing demonstration combining next-generation devices with new computation model

CoFe2O4 epitaxial nanopillars fabricated on MgO, R. Comes, J. Lu, and S. Wolf, University of Virginia, unpublished



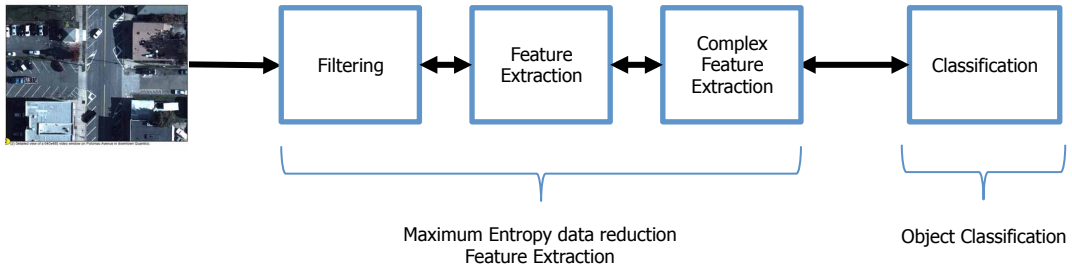
- Simulate the mapping of the Inference Module to specialized devices
- Determine systems level performance-price
- Show simple circuit operation

UPSIDE Program Tasks

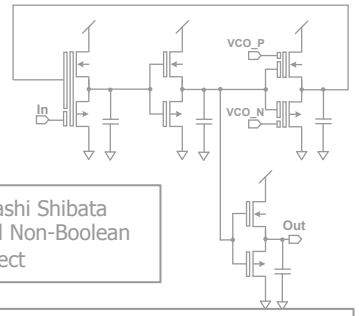
Image Processing Pipeline
An application driver

- Recreate the traditional image processing pipeline (IPP) hierarchies of Inference Modules (IM)

UPSIDE Task 1
IM Development & IPP Implementation



UPSIDE Task 2
Mixed signal CMOS implementation of computational model and system test bed

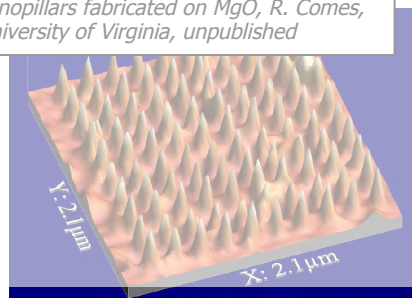


Tadashi Shibata
Intel Non-Boolean Project

- Design and Fabricate a mixed signal representation of the Inference Module in state of the art MS CMOS
- Implement a test bed system using MS CMOS
- Validate against IPP simulation

UPSIDE Task 3
Image processing demonstration combining next-generation devices with new computation model

CoFe2O4 epitaxial nanopillars fabricated on MgO, R. Comes, J. Lu, and S. Wolf, University of Virginia, unpublished



- Simulate the mapping of the Inference Module to specialized devices
- Determine systems level performance-price
- Show simple circuit operation

Task 1 – Inference Module and Image Processing Application

Task 1a: Develop Inference Module *(Phase 1)*

- Definition and simulation of Inference Module (IM).

Task 1b: Image Processing Pipeline (IPP) using Inference Module

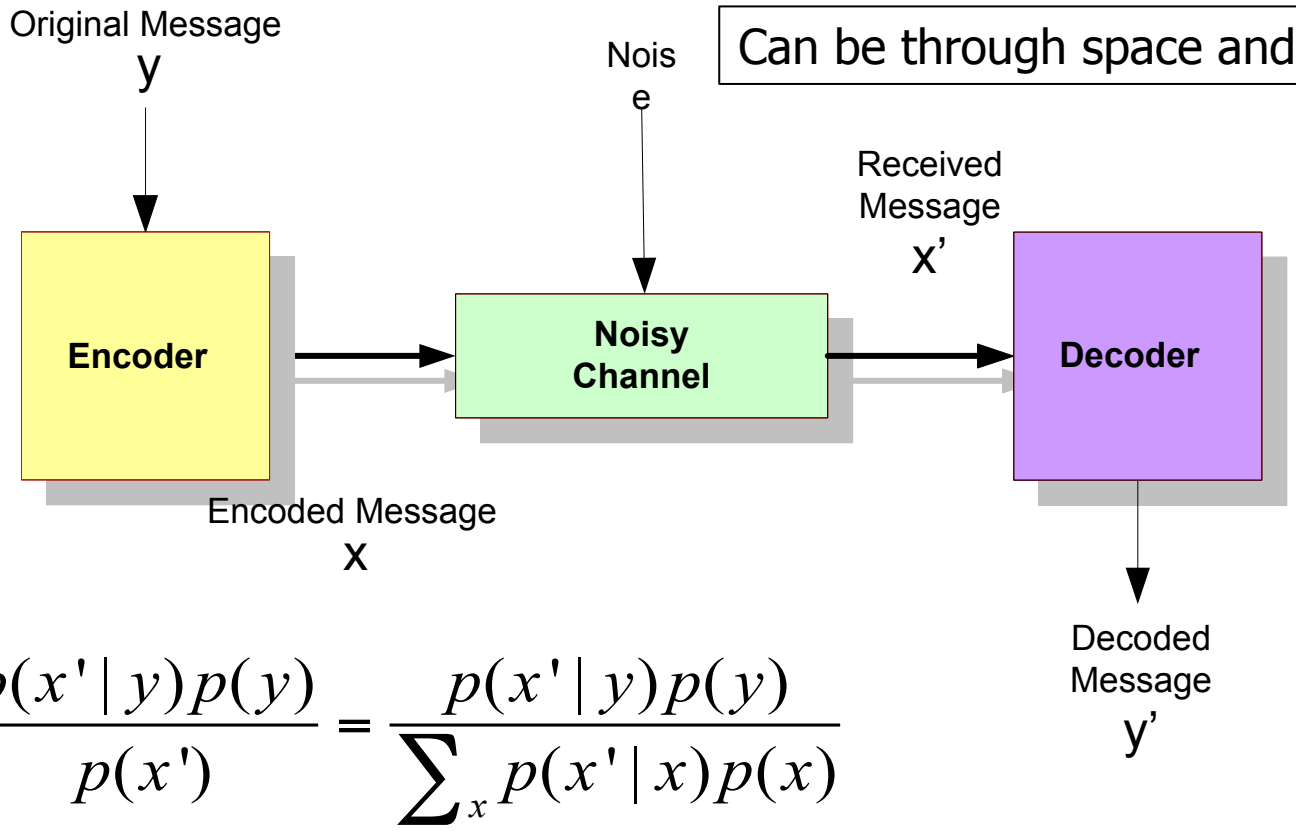
- IPP demonstrating a DoD relevant image application involving target recognition and tracking.
  - Conventional baseline implementation. *(Phase 1)*
  - "Gold" implementation of IPP using the IM as the fundamental building block. *(Phase 1)*
- Benchmark the Gold IPP against a conventional implementation of the original IPP using still images and short sequences of video data. *(Phase 1)*
- Port the IPP simulator to a high performance computing system. *(Phase 2)*
- Benchmark IPP simulations using longer, more complex object tracking sequences of higher definition video data. *(Phase 2)*

Task 1c: Additional application- Tactical Remote Sensor (TRS). *(Phase 2 Option)*

- Definition DoD relevant TRS processing pipeline, or other sensor system of interest.
- Simulation of TRS system, or other sensor system of interest.
- Analyze performance, power, and accuracy of the IM based TRS system.

Inference

One way to think about A IM is as a decoder



$$p(y \mid x') = \frac{p(x' \mid y)p(y)}{p(x')} = \frac{p(x' \mid y)p(y)}{\sum_x p(x' \mid x)p(x)}$$

The Inference Problem: Choose the most likely *y, using P[y|x']*

We need to "infer" the most likely original message given what we received and our knowledge of the statistics of channel errors and the messages being generated

Example IM, A Simple Associative Network

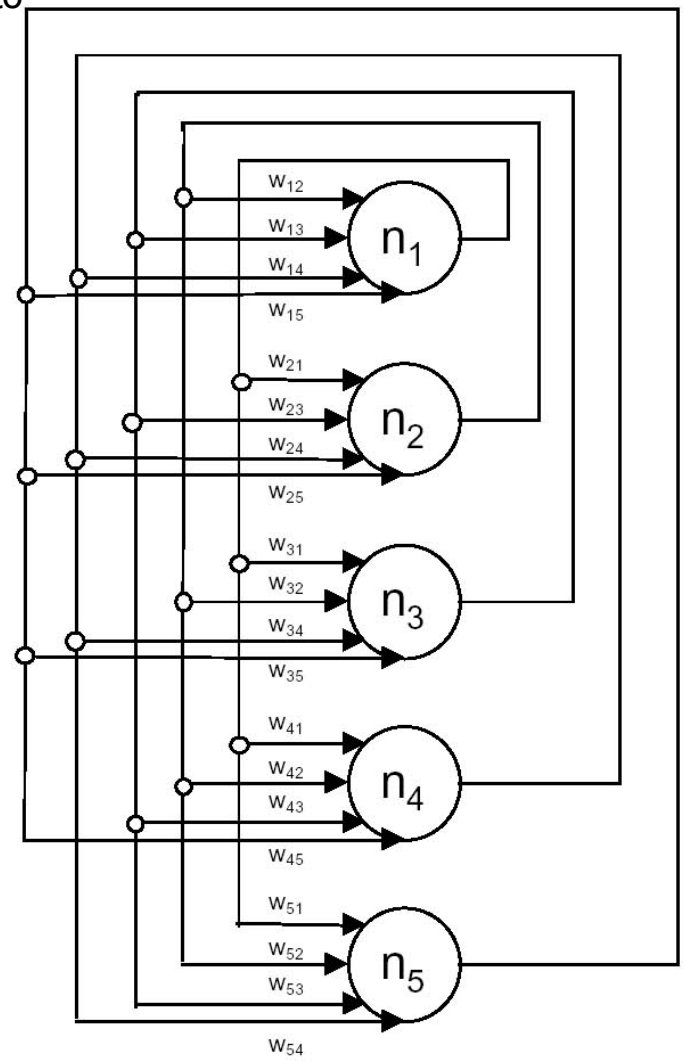
Recurrent associative memories create energy surfaces, the shape of the energy surface can be configured by the coupling coefficients, allowing the AM to approximate Bayesian inference

$$\begin{pmatrix} 0 & w_{12} & w_{13} & w_{14} & w_{15} \\ w_{21} & 0 & w_{23} & w_{24} & w_{25} \\ w_{31} & w_{32} & 0 & w_{34} & w_{35} \\ w_{41} & w_{42} & w_{43} & 0 & w_{45} \\ w_{51} & w_{52} & w_{53} & w_{54} & 0 \end{pmatrix}$$

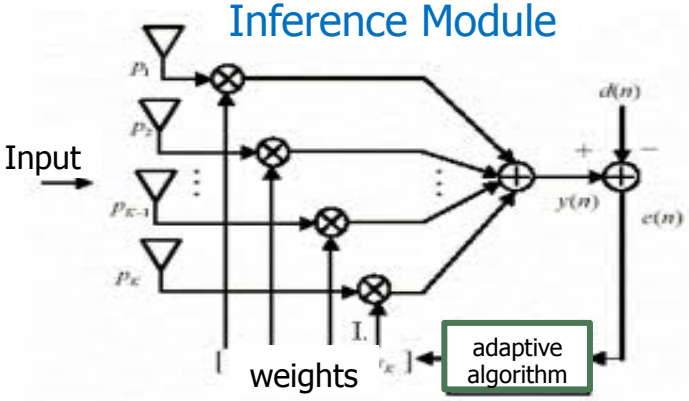
$$w_{ij} = w_{ji}$$

Constraints are “soft” and can be represented as an energy field

$$E = -\frac{1}{2}\sum w(i,j)y(i)y(j)$$



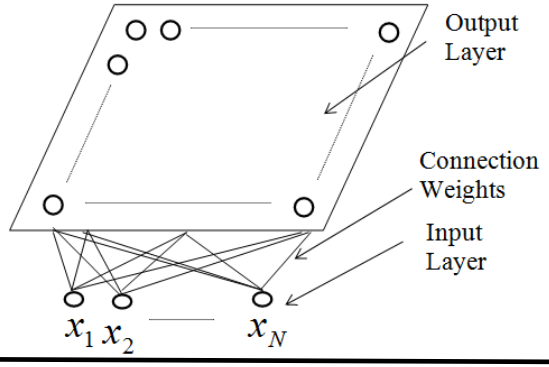
Key Concept: Self-Organization, Programming Via Adaptation

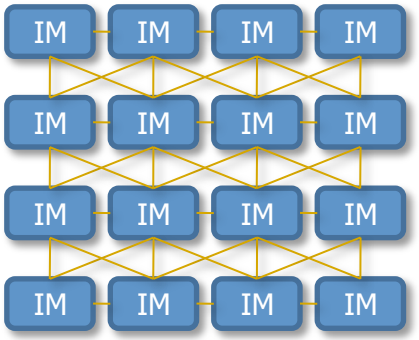


Each Inference Module (IM) adapts to the statistics of its input by altering a set of parameters, "weights," in Self-Organizing (SO), unsupervised models, "coupling coefficients" in oscillatory models

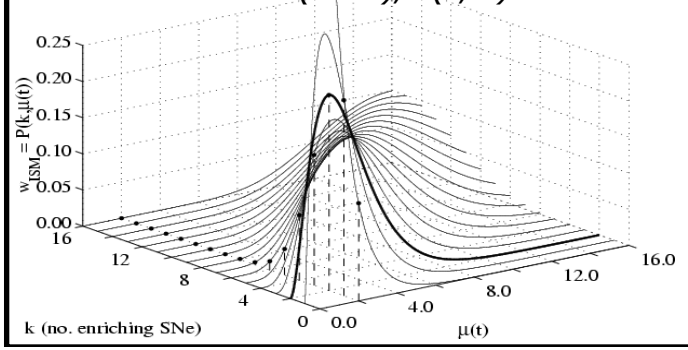
IMs can then be layered, each presenting a reduced data stream to the next layer, which, in turn, adapts to its input, etc.

Note, that such adaptation can also help reduce the effects of device variability and device yield





Probability density function of two state variables (0..16), P(k, u) = 0..1.0



By constraining the representation space, which Sparse Data Representations (SDRs) do naturally, each IM performs a "maximum entropy" (minimum information loss) data reduction.

Key Concept: Sparse Data Representations

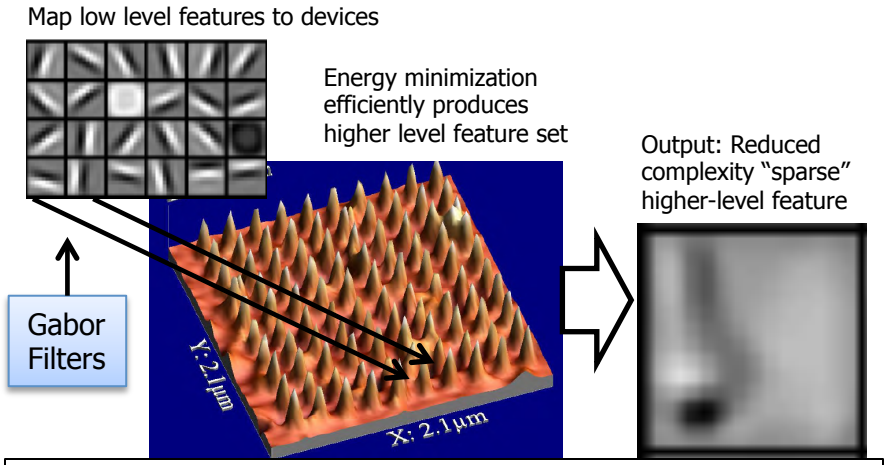
SDR - Natural Data Representation for Devices

- SDR is essential to approach (device oriented)
- Each code uses a sparse number of Representation Units (RU)

SDR In Computation

- Each device represents a RU in hardware
- Maps naturally and efficiently to devices
- Constant access time, independent of the number of stored vectors

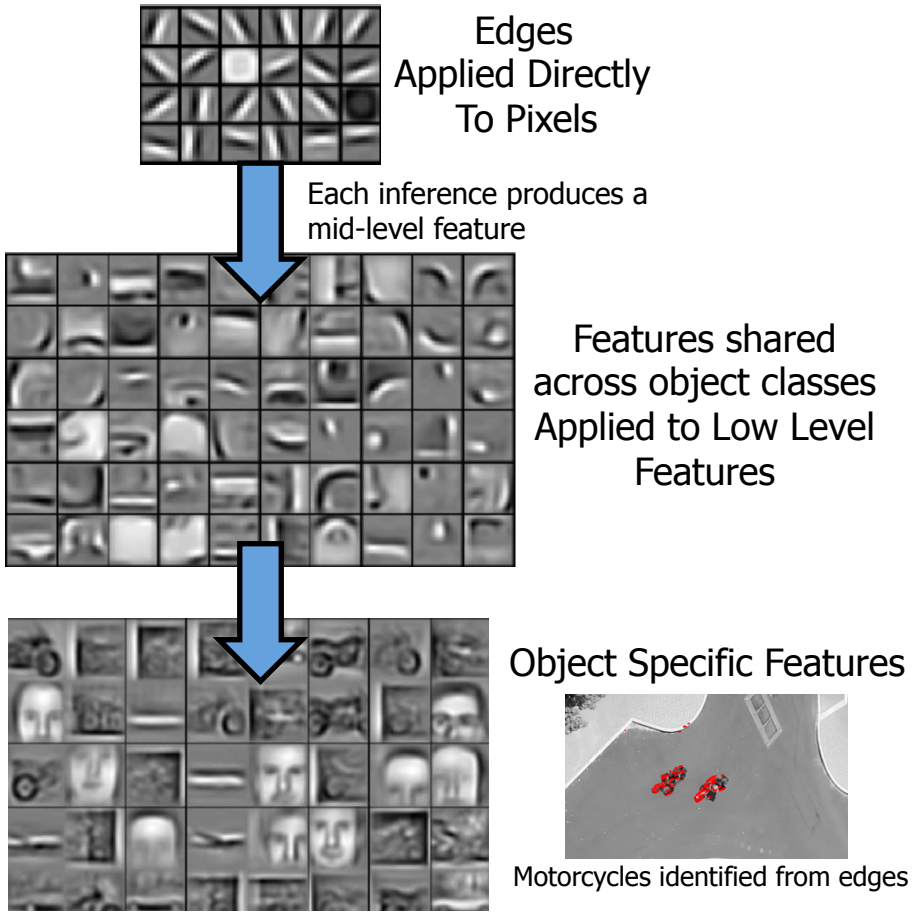
Sparse Data Probabilistic Inference



Scalable Probabilistic Computing via Sparse Distributed Representations, Gerard (Rod) Rinkus, Neurithmic Systems

Identification of 4 distinct objects using SDR

- Objects: Cars, Faces, Motorbikes, and Airplanes



Ref: Lee, Grosse, Ranganath and Ng. In *Proceedings of the Twenty-Sixth International Conference on Machine Learning*, 2009.

SDR casts the computation into a form that maps directly to emerging devices

Task 1 – Inference Module and Image Processing Application

Task 1a: Develop Inference Module *(Phase 1)*

- Definition and simulation of Inference Module (IM).

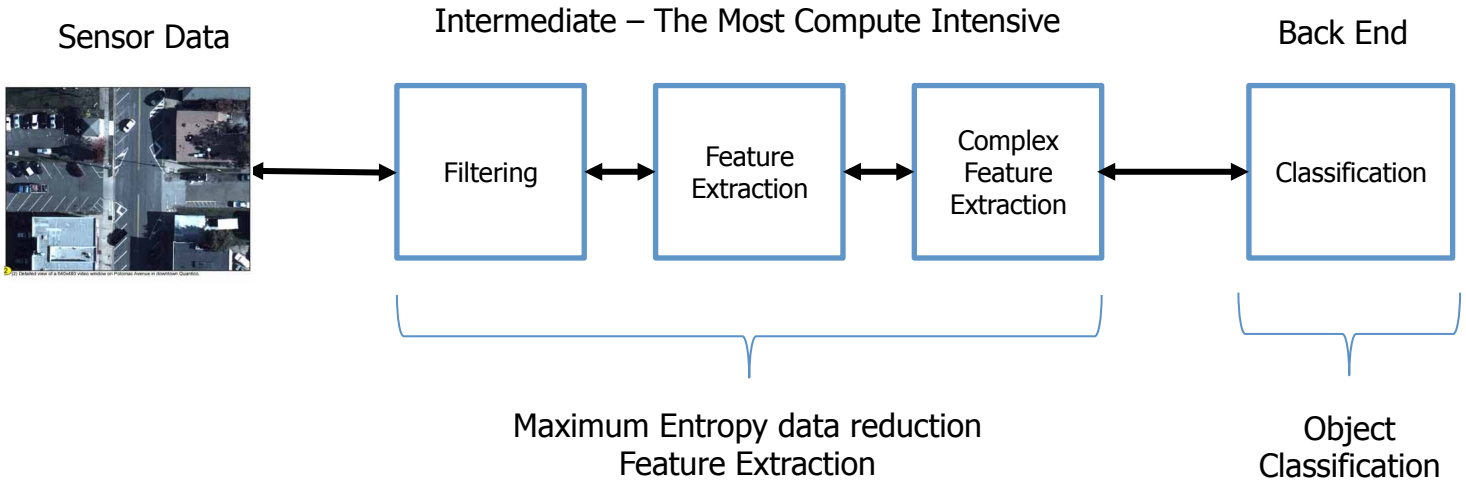
Task 1b: Image Processing Pipeline (IPP) using Inference Module

- IPP demonstrating a DoD relevant image application involving target recognition and tracking.
  - Conventional baseline implementation. *(Phase 1)*
  - "Gold" implementation of IPP using the IM as the fundamental building block. *(Phase 1)*
- Benchmark the Gold IPP against a conventional implementation of the original IPP using still images and short sequences of video data. *(Phase 1)*
- Port the IPP simulator to a high performance computing system. *(Phase 2)*
- Benchmark IPP simulations using longer, more complex object tracking sequences of higher definition video data. *(Phase 2)*

Task 1c: Additional application- Tactical Remote Sensor (TRS). *(Phase 2 Option)*

- Definition DoD relevant TRS processing pipeline or other sensor system of interest.
- Simulation of TRS system (or other sensor system of interest).
- Analyze performance, power, and accuracy of the IM based TRS system.

Task 1b Application Driver: Image Processing Pipeline



- Each proposer team will supply their own image processing application.
  - Application will do **object identification and tracking**.
  - Conventional, Boolean, application will serve as baseline for comparison.
- Performers will also be required to provide necessary data to exercise the pipeline.

Insert the IM into as many spots as possible including backend processing.

"Gold IPP" = New IPP using IM approach developed under Task 1

Task 1 – Inference Module and Image Processing Application

Task 1a: Develop Inference Module *(Phase 1)*

- Definition and simulation of Inference Module (IM).

Task 1b: Image Processing Pipeline (IPP) using Inference Module

- IPP demonstrating a DoD relevant image application involving target recognition and tracking.
  - Conventional baseline implementation. *(Phase 1)*
  - "Gold" implementation of IPP using the IM as the fundamental building block. *(Phase 1)*
- Benchmark the Gold IPP against a conventional implementation of the original IPP using still images and short sequences of video data. *(Phase 1)*
- Port the IPP simulator to a high performance computing system. *(Phase 2)*
- Benchmark IPP simulations using longer, more complex object tracking sequences of higher definition video data. *(Phase 2)*

Task 1c: Additional application- Tactical Remote Sensor (TRS). *(Phase 2 Option)*

- Definition DoD relevant TRS processing pipeline or other sensor system of interest.
- Simulation of TRS system (or other sensor system of interest).
- Analyze performance, power, and accuracy of the IM based TRS system.

Optional Task 1c Additional Application Driver

Multi-Sensor Data Fusion Demonstration Application

Proposers are *encouraged* to show the general applicability of the IM with a second application performing sensor fusion and data analysis, or in some other relevant sensor data analysis system of interest to the DoD.

Suggested example: Tactical Remote Sensor (TRS) system.

- TRS systems collect data from large numbers of geographically dispersed sensors for detection and recognition, providing real-time situational awareness.
- The sensor modalities investigated may include seismic, acoustic, magnetic, imaging (thermal and electro-optical), radio frequency, ultra-wide band, and electromagnetic.

Approach similar to Task 1b.

- Performer will provide conventional implementation of the sensor application and the necessary data to adequately exercise the pipelines.
- Demonstrate the improvement in speed and power efficiency, with comparable accuracy, based on the UPSIDE approach.

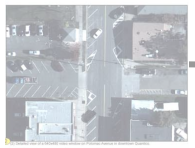
Note: This task will be performed during Phase 2 and need not be a part of the hardware demonstrations in Tasks 2 and 3.

UPSIDE Program Tasks

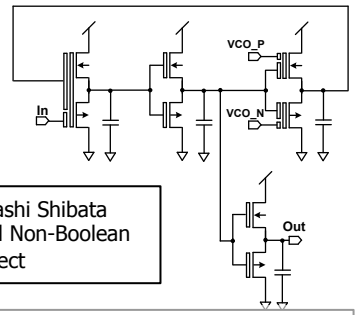
Image Processing Pipeline
An application driver

- Recreate the traditional image processing pipeline (IPP) hierarchies of Inference Modules (IM)

UPSIDE Task 1
IM Development & IPP Implementation



UPSIDE Task 2
Mixed signal CMOS implementation of computational model and system test bed

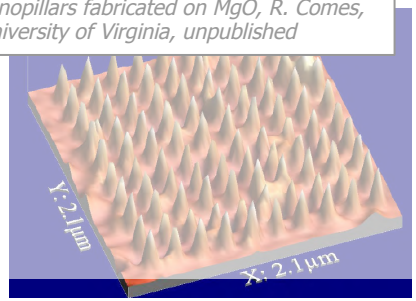


Tadashi Shibata
Intel Non-Boolean Project

- Design and Fabricate a mixed signal representation of the Inference Module in state of the art MS CMOS
- Implement a test bed system using MS CMOS
- Validate against IPP simulation

UPSIDE Task 3
Image processing demonstration combining next-generation devices with new computation model

CoFe2O4 epitaxial nanopillars fabricated on MgO, R. Comes, J. Lu, and S. Wolf, University of Virginia, unpublished



- Simulate the mapping of the Inference Module to specialized devices
- Determine systems level performance-price
- Show simple circuit operation

Task 2 – Mixed Signal CMOS Implementation

Mixed Signal (MS) CMOS Inference Processor

- Implementation of IPP using a simulation of the MS CMOS IM chip. (*Phase 1)*
  - Simulations implemented in software.
  - Obtain estimates of system performance, power requirements, data precision, and accuracy.
- Design of a MS CMOS IM chip. (*Phase 1)*
  - Develop IM digital control and communication circuitry.
- Fabricate, package and test the MS CMOS IM chip. (*Phase 1)*
  - Verify performance, power requirements, and accuracy.
- Develop a hardware IPP testbed system based on the MS CMOS IM chip fabricated in Phase 1. (*Phase 2)*
- Validate and benchmark system testbed. (*Phase 2)*
- Report on requirements for commercialization of MS CMOS IM chip. (*Phase 2)*

Mixed Signal CMOS & System Testbed

Goal: Demonstrate that the computational model developed under Task 1 has the expected advantages, when implemented in traditional hardware.

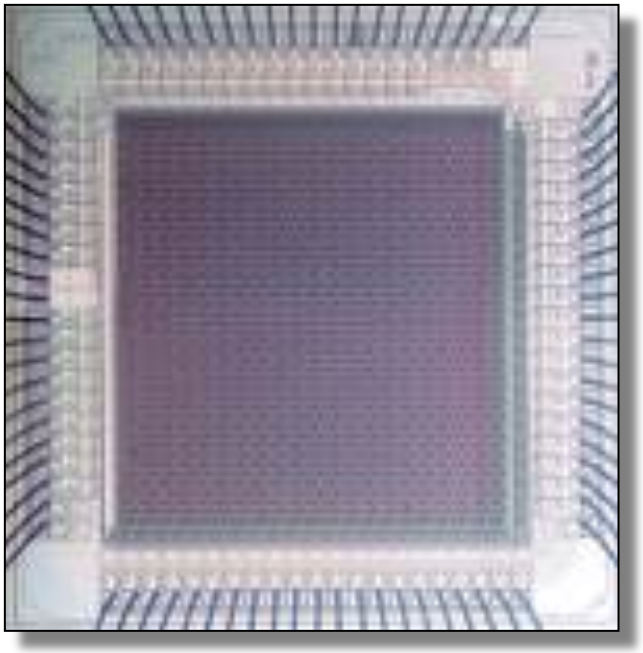
MS CMOS Inference Module

- Multiple IMs on a single chip implementing the required inference characteristics modeled in Task 1.
- Should include any necessary circuitry for I/O and control.
- Outputs:
  - Hardware level description and simulation of the design
  - Fabricated, packaged, and tested chip produced in a SOA MS CMOS process.

CMOS Test Bed

- Hardware Testbed: validated against the Gold IPP.
- Processing of image sensor data in real-time.
- Measure accuracy, performance and power efficiency.

Commercial Chip: FABA



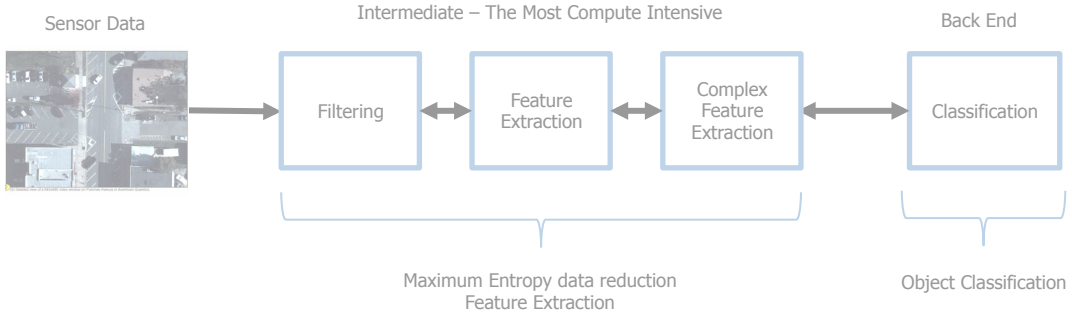
- One potential outcome of the analog CMOS development could be a commercial product, a "Field Adaptable Bayesian Array"
- A useful building block for intelligent systems
- It should far exceed, in performance, capacity, and low power, implementations of the same functionality in maximally scaled Digital analog (sub-threshold) CMOS
  - Early projections show roughly 1000x increase in operations per Watt
- Such a technology can be applied where-ever computer systems interact with the real world, not just computer vision

UPSIDE Program Tasks

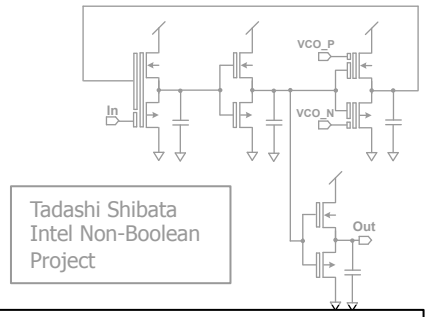
Image Processing Pipeline
An application driver

- Recreate the traditional image processing pipeline (IPP) hierarchies of Inference Modules (IM)

UPSIDE Task 1
IM Development & IPP Implementation



UPSIDE Task 2
Mixed signal CMOS implementation of computational model and system test bed

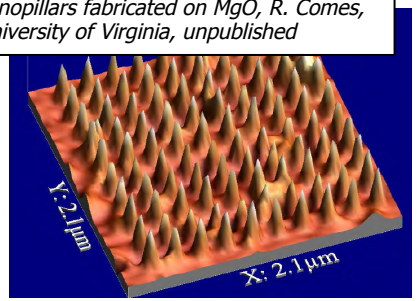


Tadashi Shibata
Intel Non-Boolean Project

- Design and Fabricate a mixed signal representation of the Inference Module in state of the art MS CMOS
- Implement a test bed system using MS CMOS
- Validate against IPP simulation

UPSIDE Task 3
Image processing demonstration combining next-generation devices with new computation model

CoFe2O4 epitaxial nanopillars fabricated on MgO, R. Comes, J. Lu, and S. Wolf, University of Virginia, unpublished



- Simulate the mapping of the Inference Module to specialized devices
- Determine systems level performance-price
- Show simple circuit operation

Task 3 – Non-CMOS, Nanoscale, Emerging Devices

Emerging Nanoscale Devices Running UPSIDE IM

- Develop emerging device demonstration of simple circuits. (*Phase 1)*
  - A circuit with two or more instantiations of the selected emerging device.
- Develop simulation of the ED IM and insert into the IPP. (*Phase 1)*
- Develop test, I/O and control circuitry for the ED IM. (*Phase 1)*
  - Devices fabricated on CMOS desired, but not required.
  - Close integration between emerging devices and circuitry on CMOS if possible.
- Benchmark the software ED IM IPP against the Gold simulation. (*Phase 2)*
- Introduce fault simulation capability to the ED IM IPP.
  - Demonstrate operation over a reasonable set of yield assumptions. (*Phase 2)*
- Fabrication of ED IM, and integration with the test, I/O and control circuitry. (*Phase 2)*
- Benchmark the physical realization of ED IM against the simulation. (*Phase 2)*
- Report describing technical challenges for commercialization of an ED IM based IPP system. (*Phase 2)*

What Is An Emerging Device?

UPSIDE will drive the development of a synergistic approach to computing based on probabilistic computational models and new device technologies that will emerge *("emerging devices")* to become alternatives to today's commercial CMOS FETs.

Emerging Device Guidelines:

- Nano-scale
- Non-CMOS (device is non-CMOS, but can be in a hybrid circuit with CMOS)
- Not readily available commercially
- Teams are encouraged to investigate multiple candidate emerging devices
  - But performers are not expected to take all the devices through Phase 2.

The guidelines are flexible: Main criterion is to get the job done!

Example: A Simple Associative Network

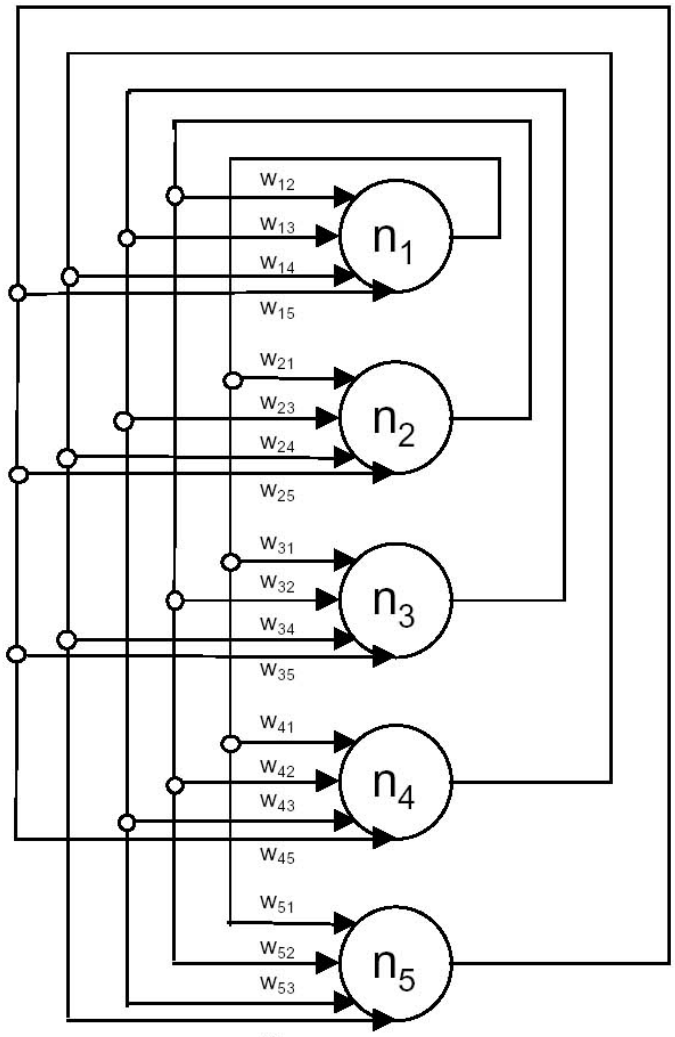
Recurrent associative memories create energy surfaces, the shape of the energy surface can be configured by the coupling coefficients, allowing the AM to approximate Bayesian inference

$$\begin{pmatrix} 0 & w_{12} & w_{13} & w_{14} & w_{15} \\ w_{21} & 0 & w_{23} & w_{24} & w_{25} \\ w_{31} & w_{32} & 0 & w_{34} & w_{35} \\ w_{41} & w_{42} & w_{43} & 0 & w_{45} \\ w_{51} & w_{52} & w_{53} & w_{54} & 0 \end{pmatrix}$$

$$w_{ij} = w_{ji}$$

Constraints are “soft” and can be represented as an energy field

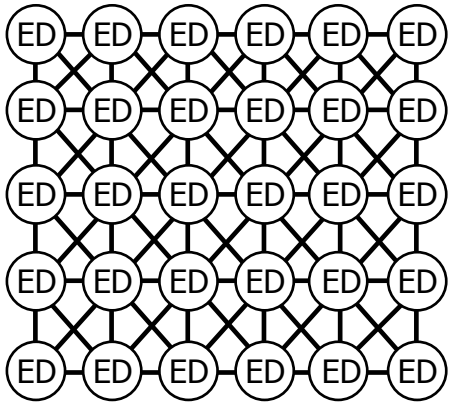
$$E = -\frac{1}{2}\sum w(i,j)y(i)y(j)$$

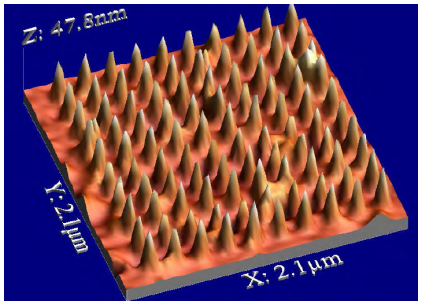


Key Concept: Association Via Emerging Devices – The Hardware is the Algorithm

Device Physics ⇨ Interacting Nonlinear Systems ⇨ Bayesian Inference

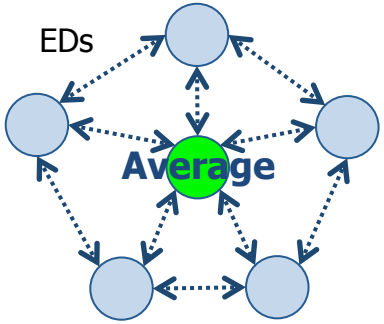
- Coupled devices operating at very low power levels



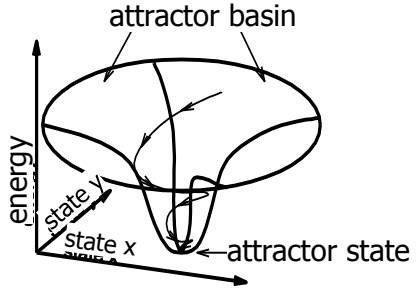


Nano-device Array

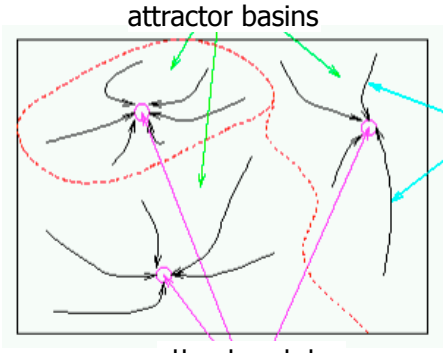
- Spontaneous synchronization corresponds to a stable minimum in an energy space



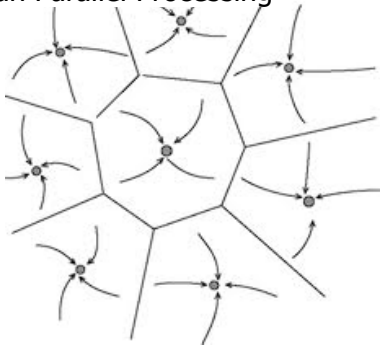
- The coupling coefficients create a controlled energy surface with specific local minima that models a probability distribution



- The system converges to the "closest" energy minimum, which is generally the most likely vector in a Bayesian sense

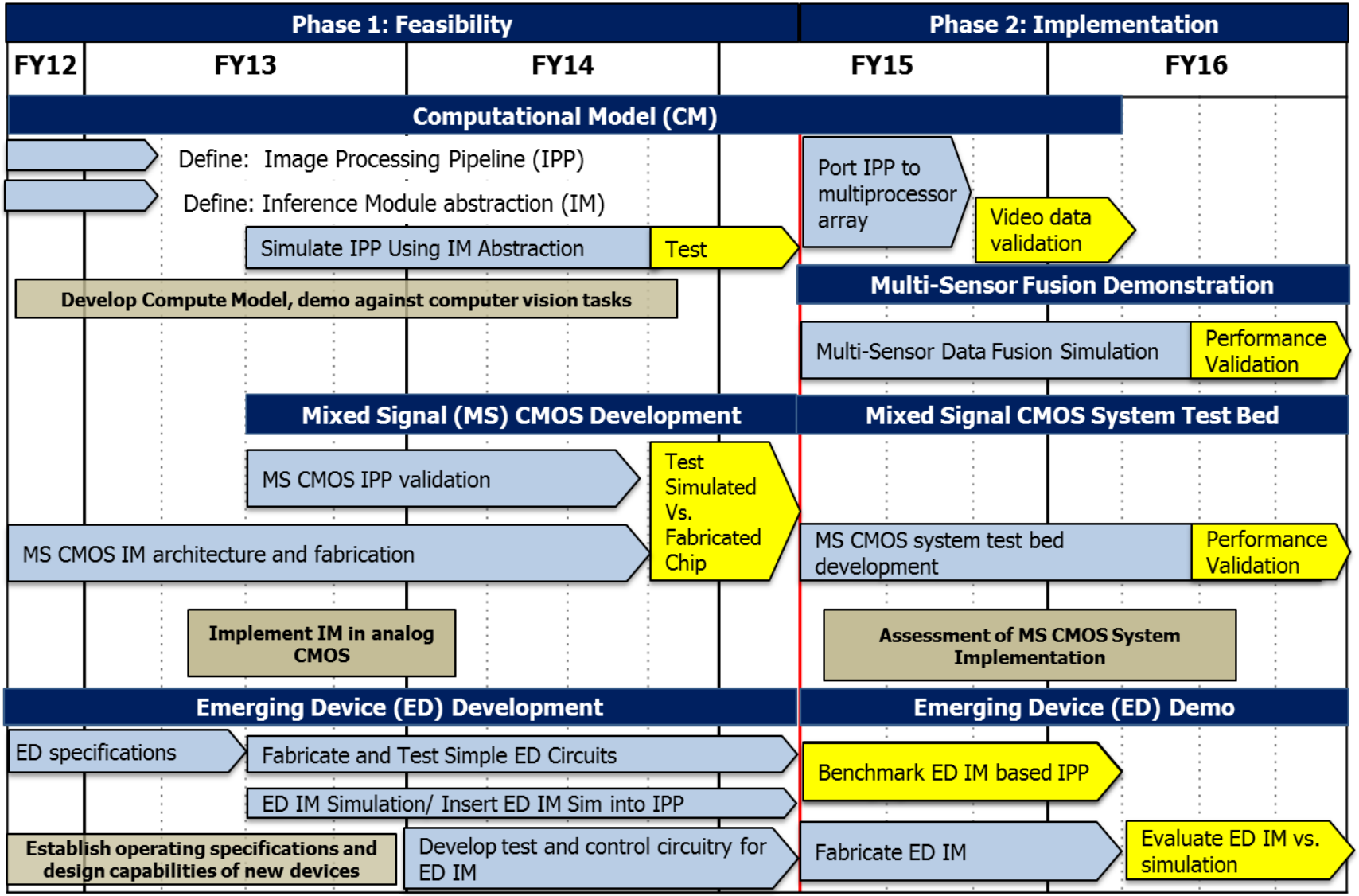


Bayesian Parallel Processing



Associative memories approximate probabilistic inference, at Giga-ops Performance levels using micro-watts of power

UPSIDE Schedule



Why Now?

- There has been significant progress in the fields of Machine Learning, Bayesian techniques, and neuroscience
- There are increasing numbers of exotic and "emerging" devices that have been developed (and in most cases fabricated) with potentially useful behavior ("physics")
- A systems perspective has been lacking – however there is a general recognition that there is sufficient critical mass in the relevant technologies to begin considering "systems" approaches
- Extreme power limitations are forcing us to re-evaluate the entire computing paradigm
- Investment is being made by DARPA to stimulate sufficient critical mass to demonstrate the promise of computing systems based on unconventional techniques and hardware