Sensory Level Processing

Great Possibilities,

Full Scale Practical Implementations Illusive

Dr. Vladimir Brajovic

Director, Computational Sensor Laboratory

The Robotics Institute, Carnegie Mellon University

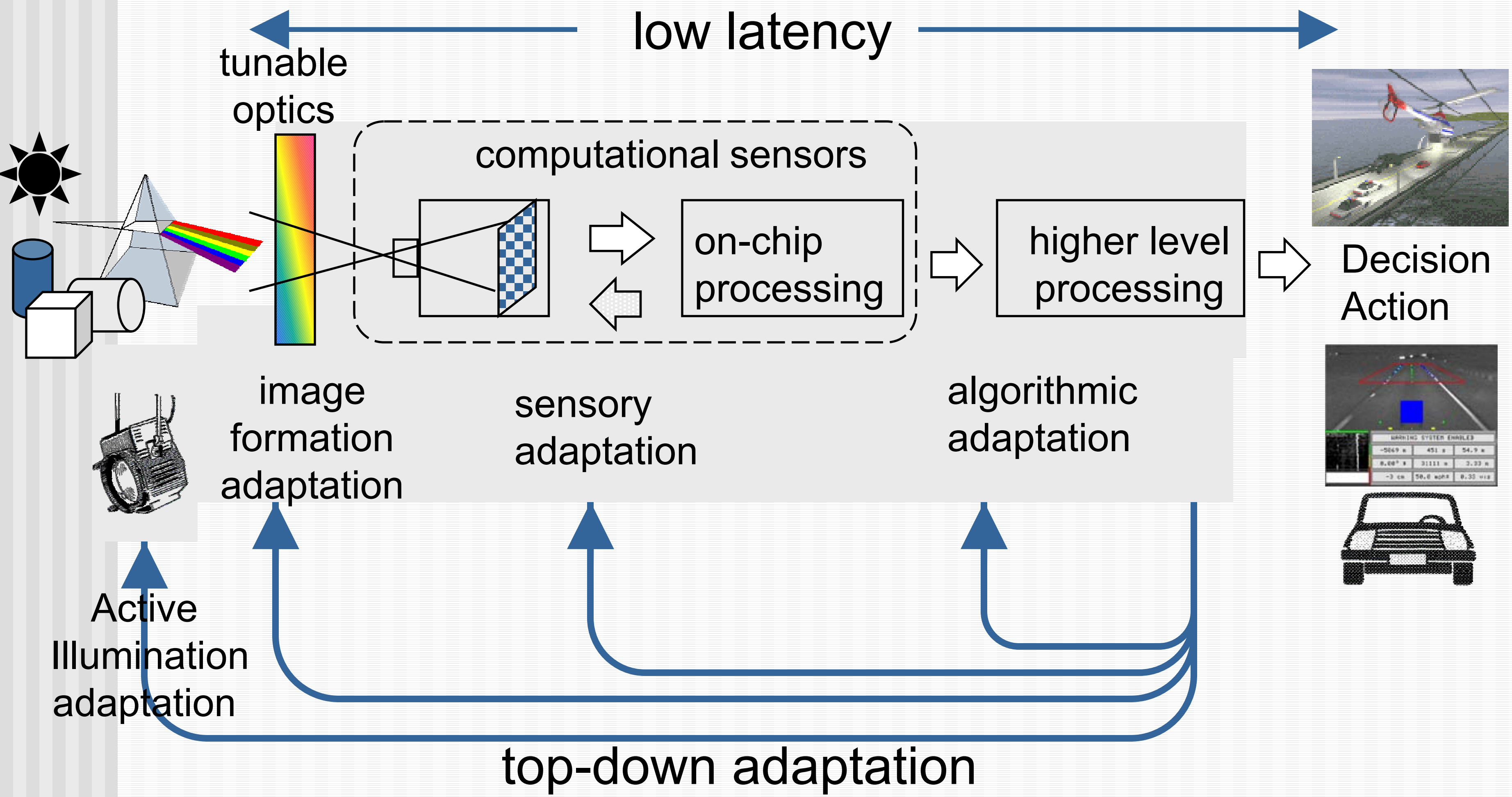
brajovic@cs.cmu.edu

This talk only scratches a surface

- Introduction/background for our later brainstorming.
- Illustrate what has been/could be done on a chip (incomplete)
- Elucidate some new opportunities (incomplete, personal opinions)
- Hint what optics, active illumination and algorithms could do to make life easier on-chip. (incomplete)

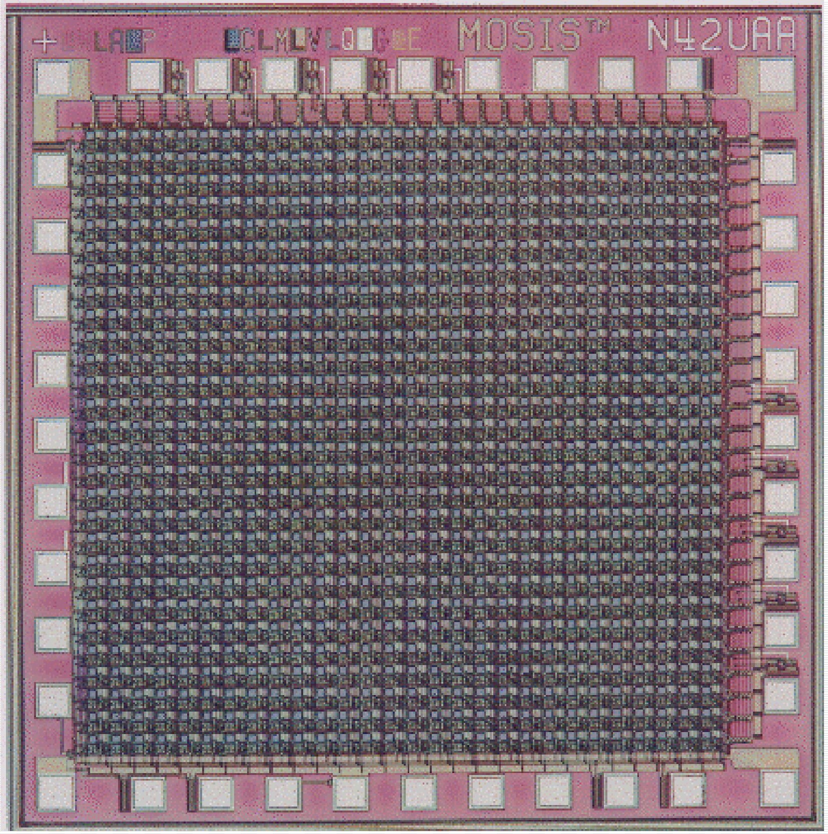
Back-to-Back Approach to Vision Systems

Reliable, Fast, Real



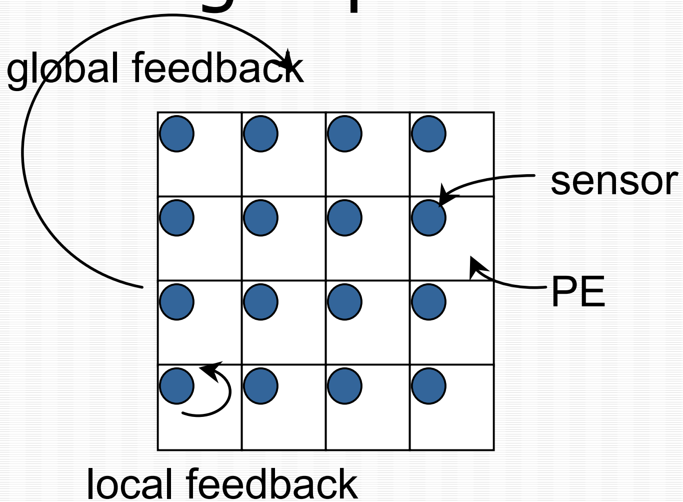
Computational Sensors are...

VLSI chips that integrate sensors with processors.



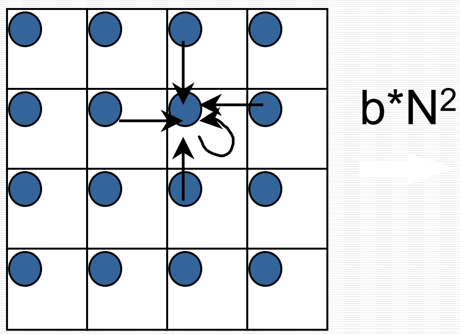
Computational Sensors Offer New Opportunity

- **Massive Parallelism**: Fine-grain, one data item per processor.
- **High data transfer**: wire each input signal to a processor.
- **Top-down adaptation:** tight processor-sensor feedback.



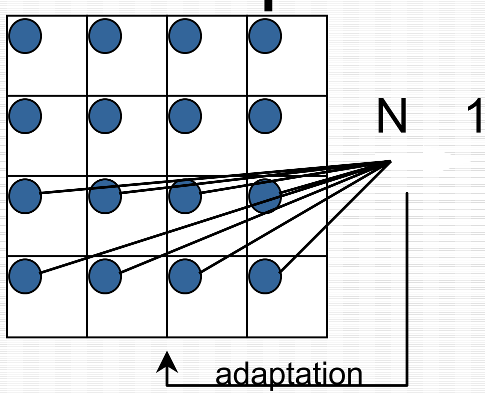
Operation Classes

- Local Operations



Difficulty: must read out large amount of data

- Global Operations



Advantages
- a few data to read out
- a few data to use for global feedback.

Disadvantages:
- requires global data aggregation

Sensors: Photodetectors

- CCD well
- photodiodes
- phototransistors
- MEMS (IR, …)
- Bio-sensors (e.g., bacteriorhodopsin)
- Other

Processor

- The processor can be optical or electronic.
- Functions:
  - as simple as gain/offset adjust, or
  - as complex as regularization, motion computation or sorting;
  - usually hardwired (no programmability)
- Digital vs. Analog

Digital Circuits

- Advantages
  - standard cells available
  - advanced design tools
  - shorter design and test
- Disadvantages
  - requires A/D and D/A conversion
  - low throughput
  - more real-estate and power than analog for same functionality

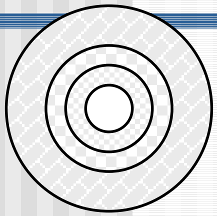
Analog Circuits

- Advantages:
  - Speed - high throughput/low-latency
  - area-efficient charge-domain processing
  - low power
- Disadvantages
  - limited precision (typically 6-10 bits)
  - no long-term storage (< 1second)
  - no programmability
  - long design, test, debug cycles

Candidate Algorithms...

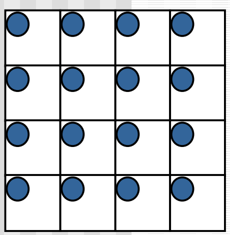
- Algorithms that map naturally to physical processes encountered in semiconductors.
- Algorithms that require massively parallel integration of sensory data and/or cue.
- Algorithms that can be done on array periphery (global projections, global data aggregation).
- Algorithms that need access to instantaneous sensory signals (e.g., Lidar)
- Minituarize conventional image processing – probably NOT!

System-on-chip: Architectures



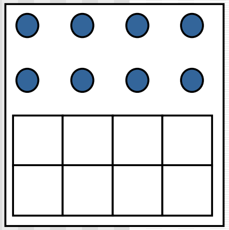
- **Spatio-geometric architecture**

  Computation takes place in its inherent geometrical structure and/or optical properties (e.g., optical masks, photodetector shape/arrangement/signal-wiring).



- **Focal plane architecture**

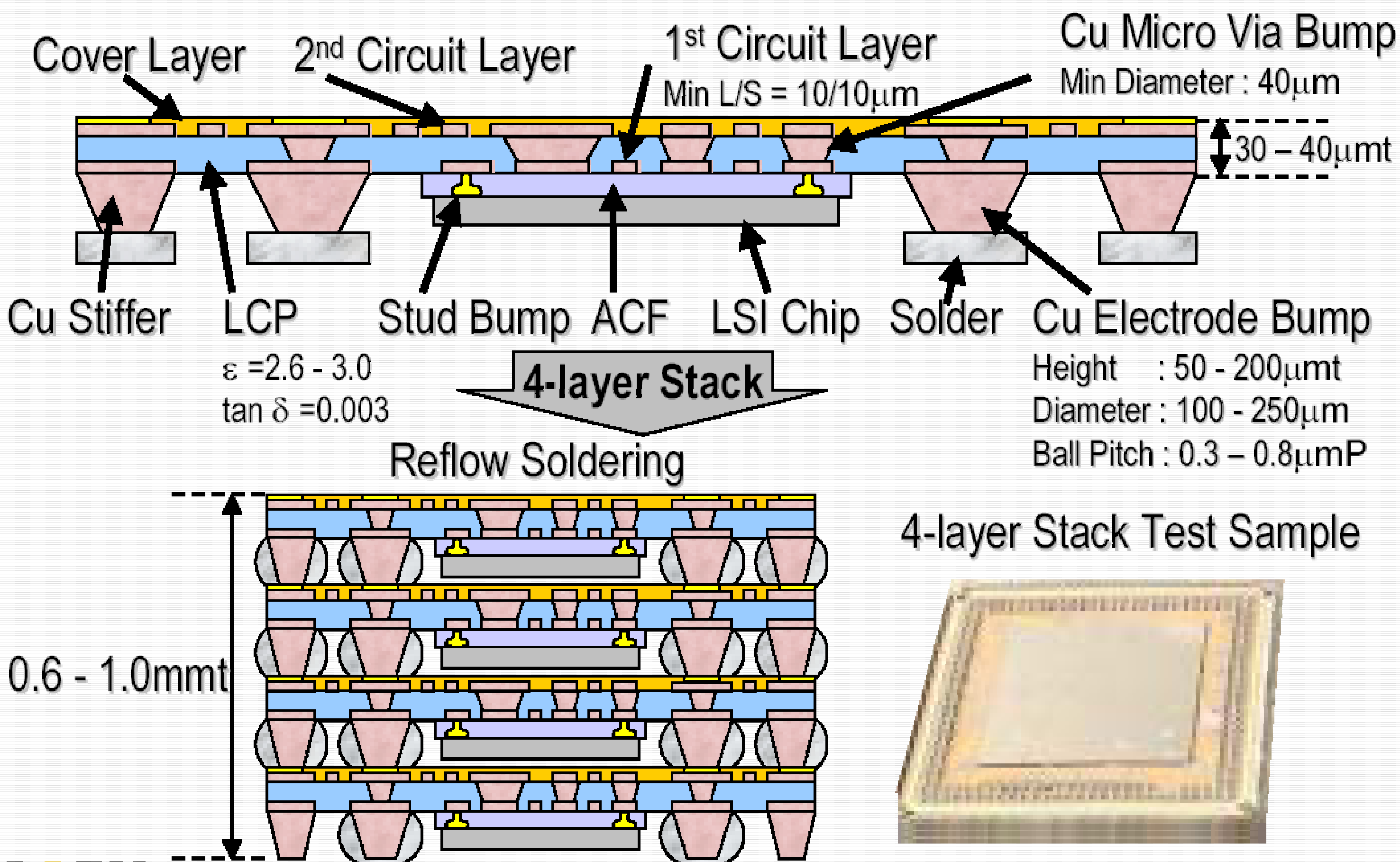
  Sensors and processing elements are tightly coupled (e.g., SIMD, distributed global processors).



- **Computational Module**

  Sensor and processing elements are not tightly coupled, but desired processing and speed is only achievable in VLSI (e.g., processing at the sensor array periphery)

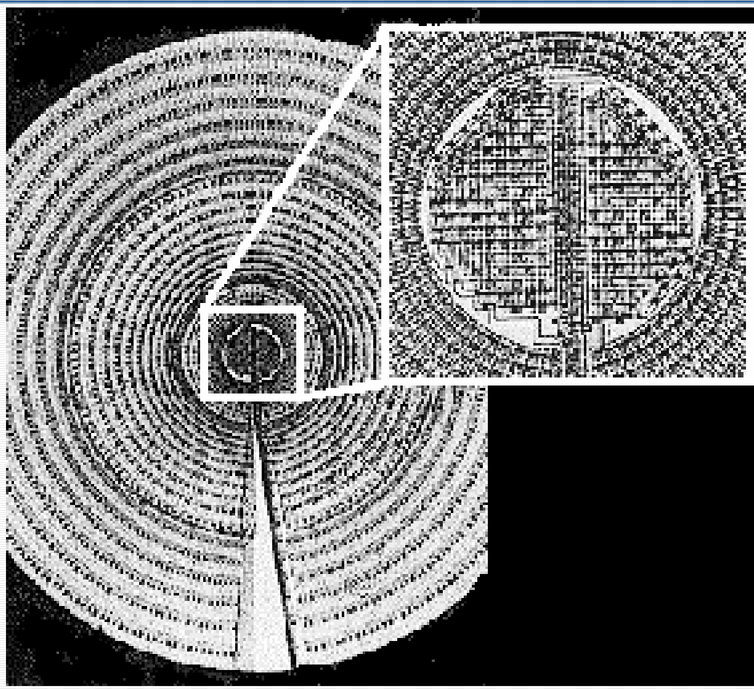
System-on-chip vs. System-in-package: 3D Packaging



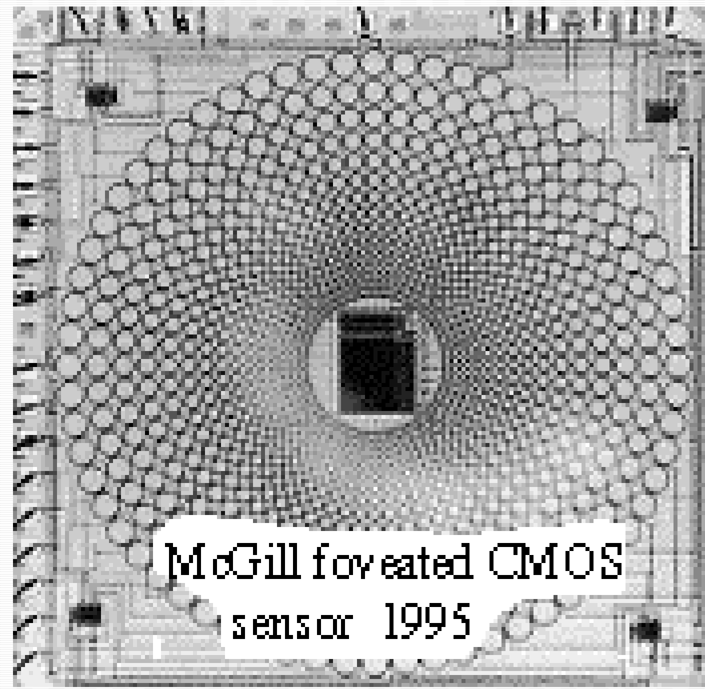
NORTH

NMTI - NORTH CORPORATION 2001/10/01

Geometric Arrangement Example: Foveating Sensor

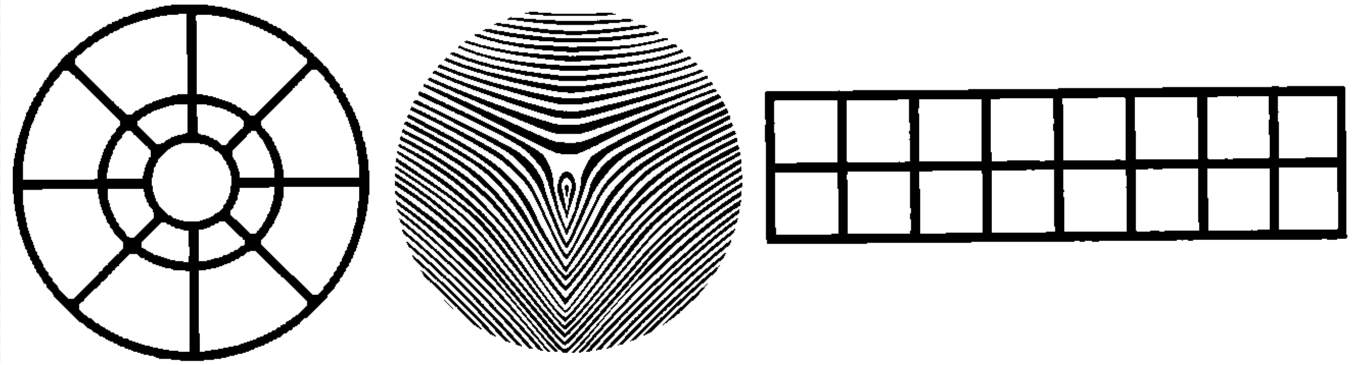


Van der Spiegel, et al. 1991



Geometric Arrangement Example: Optically Achieved Fovea

Binary Optics

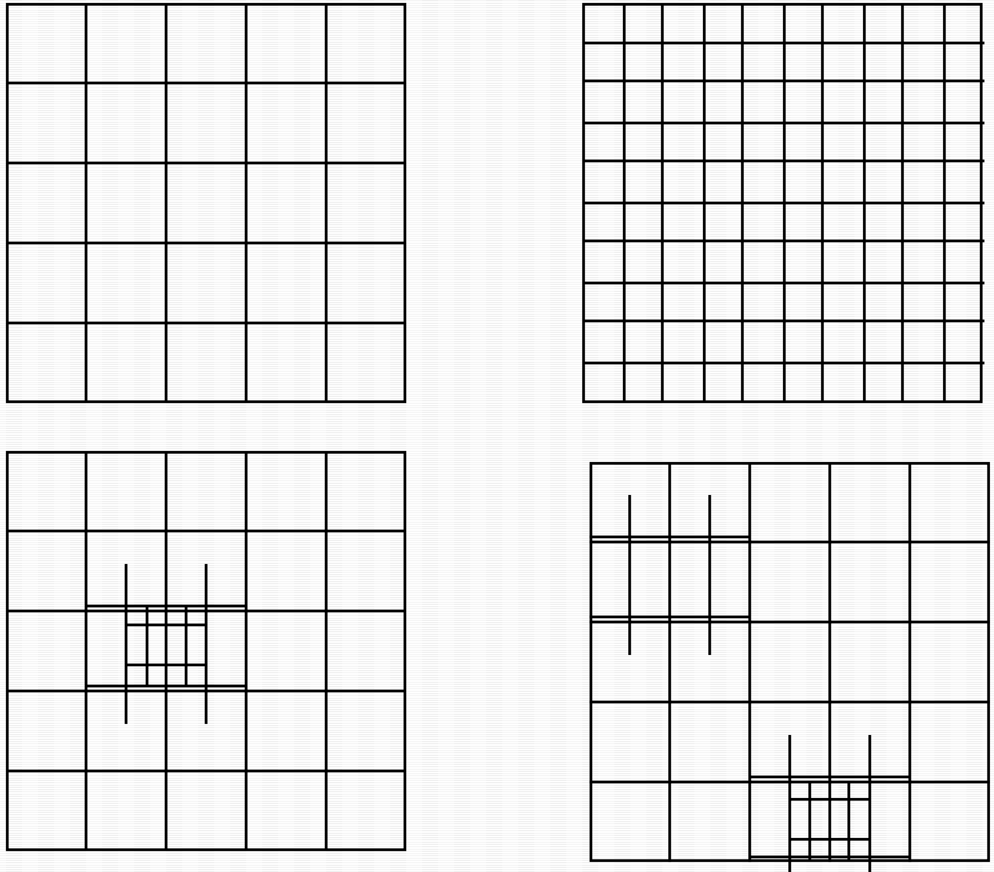


object diffractive optics image

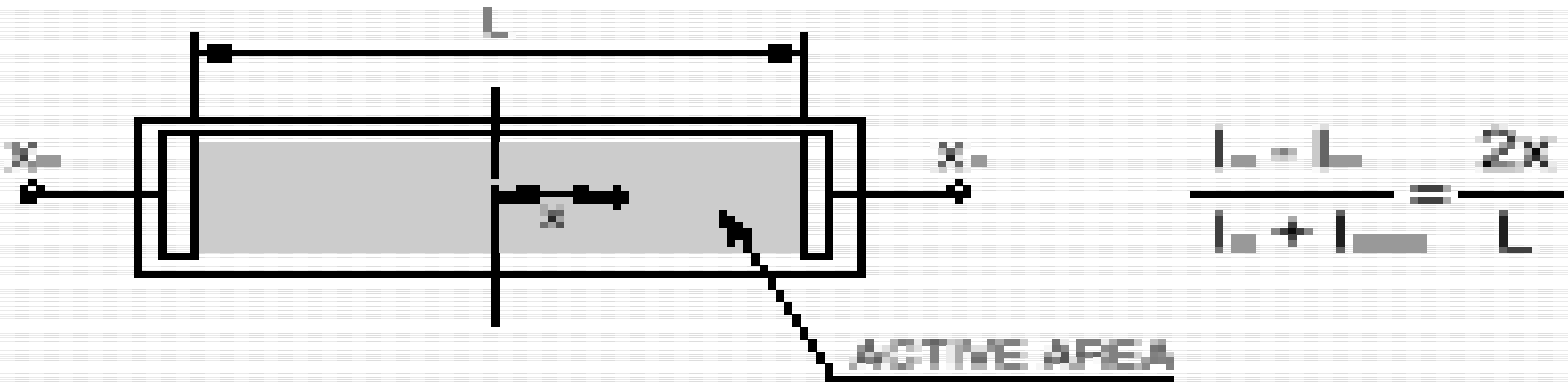
Geometric Arrangement Example: Electrically programmable fovea

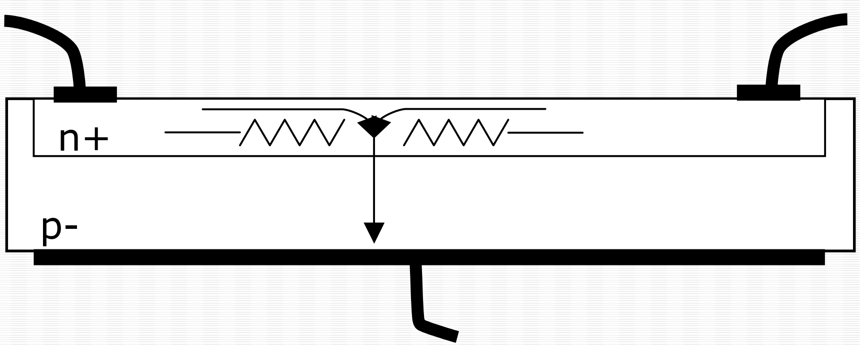
Kosonocky and Wilder 1993: Multi-resolution Sensor

Pixel binning

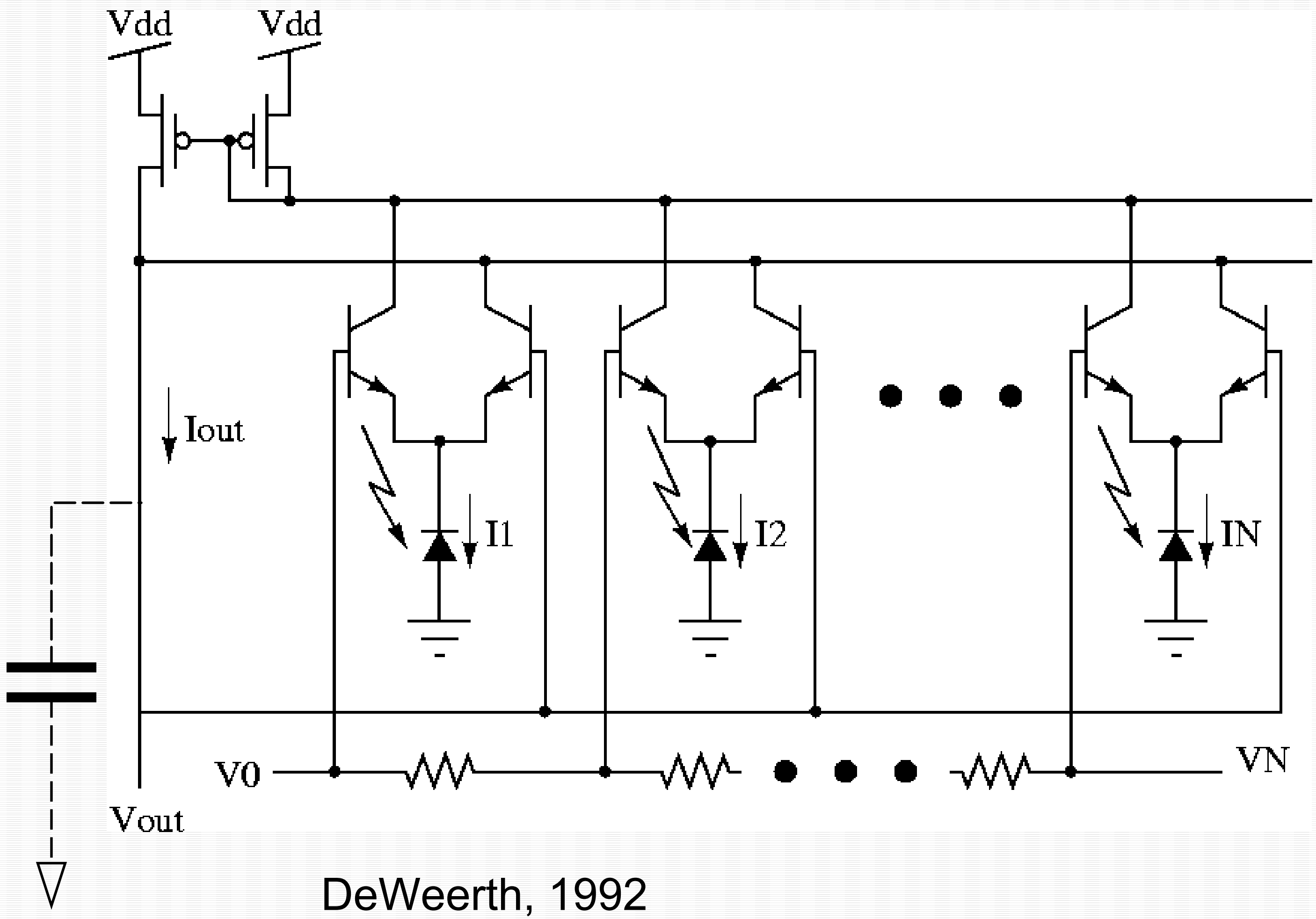


Global Data Aggregation Example: Position Sensitive Detector (PSD)





Global Data Aggregation Example: Discretized Position Sensitive Detector



DeWeerth, 1992

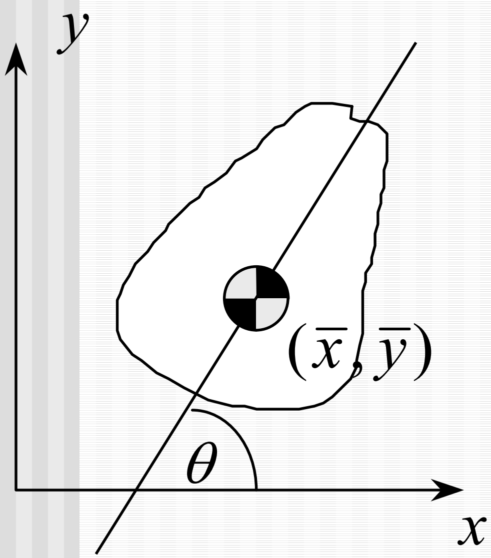
Global Data Aggregation Example: Adding Orientation Sensitivity

- For position of the object centroid $(\bar{x}, \bar{y})$ these moments are required:

$$A = \iint_R I(x,y)\,dxdy$$

$$A\bar{x} = \iint_R xI(x,y)\,dxdy$$

$$A\bar{y} = \iint_R yI(x,y)\,dxdy$$



... and for orientation:

$$a' - c' = \iint_R (x^2 - y^2) I(x,y)\,dxdy$$

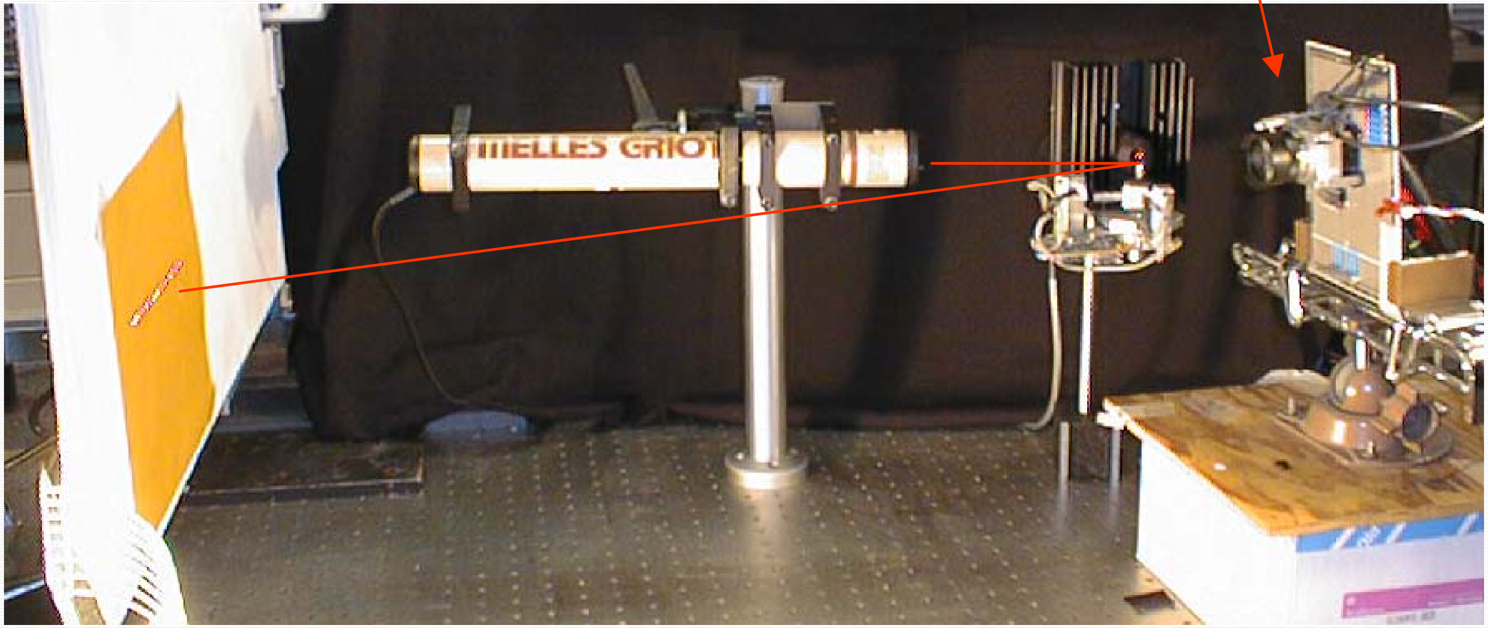
$$b' = \iint_R xyI(x,y)\,dxdy$$

$$\tan 2\theta = \frac{2b'}{a' - c'}$$

Standley and Horn, 1994

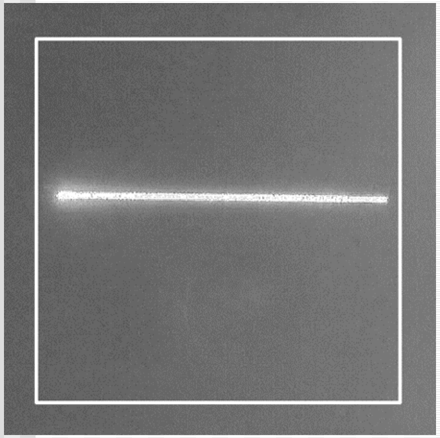
What is PSD good for? Tracking

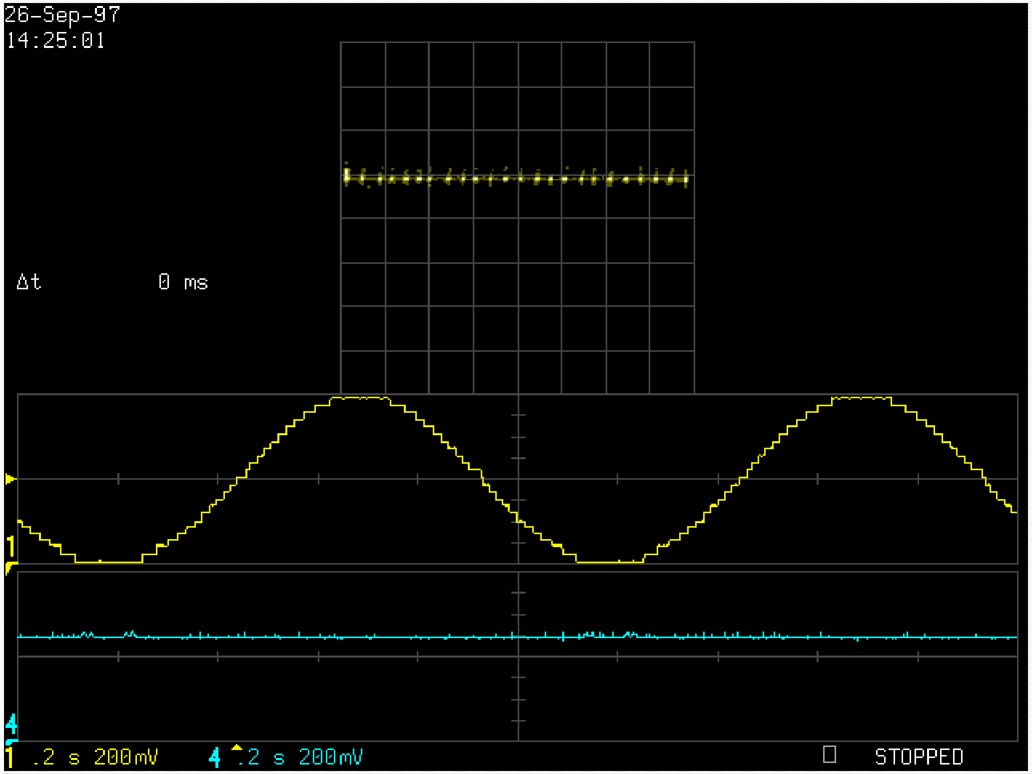
tracking sensor



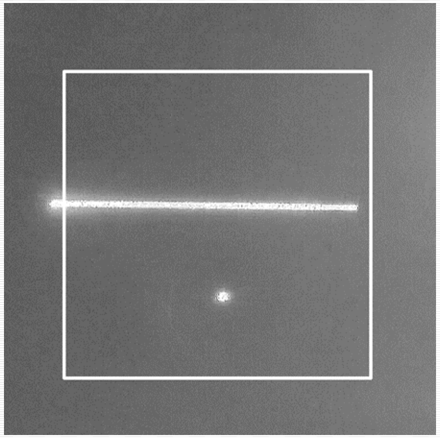
Tracking a Target

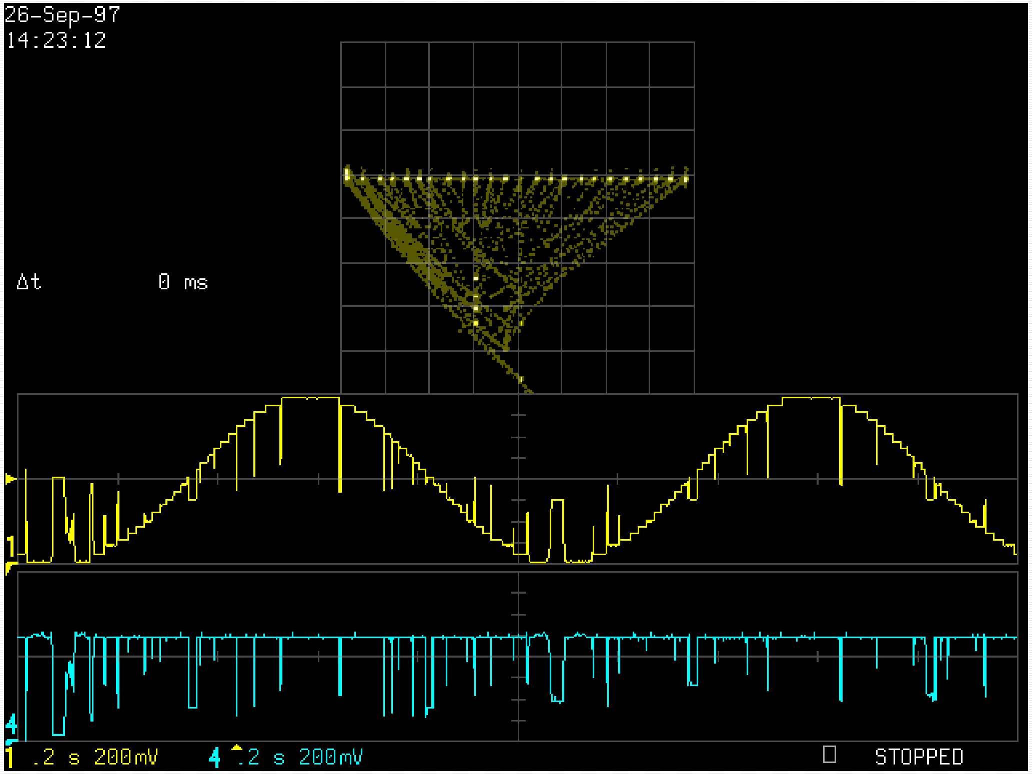
Speed= 7000 pix/sec =
2000 ccd_pix/ccd_frame



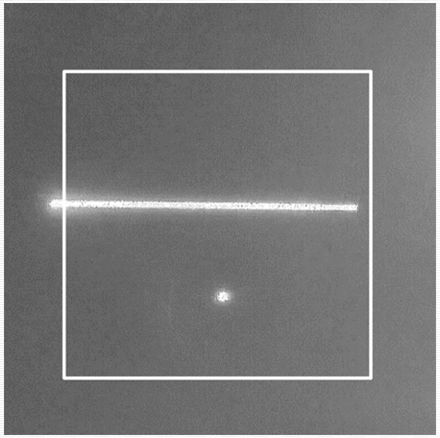


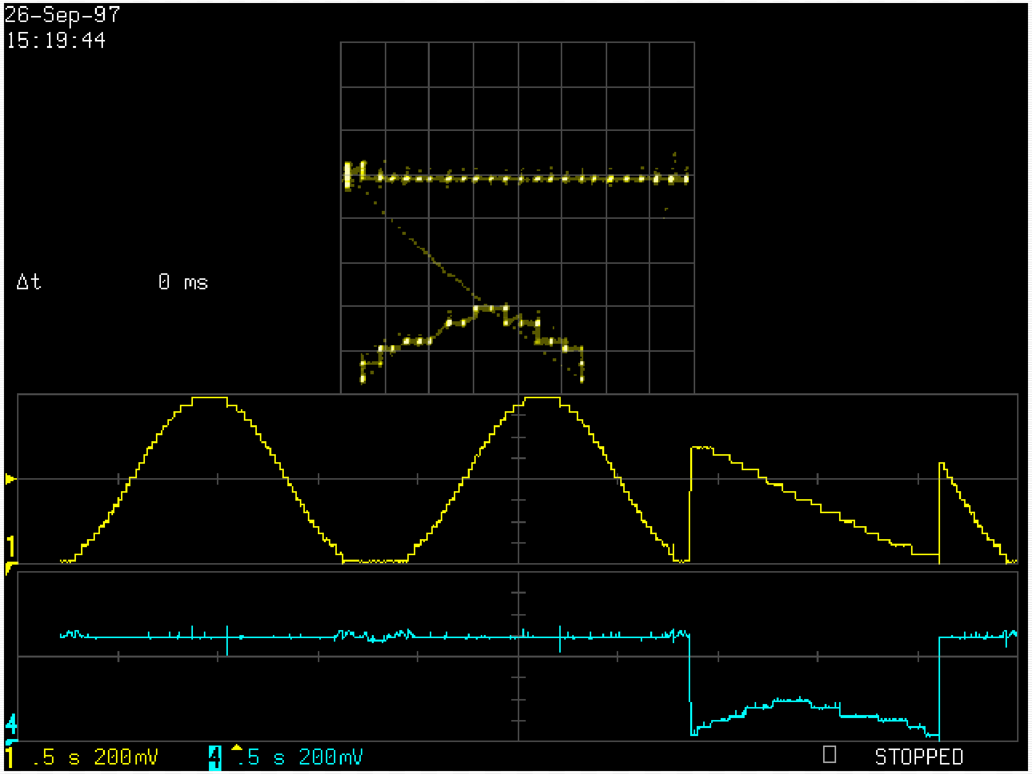
Two Target Interference





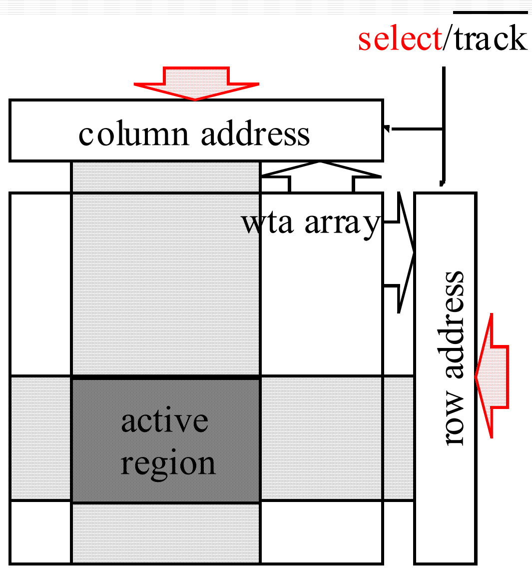
Robust Tracking with Attention





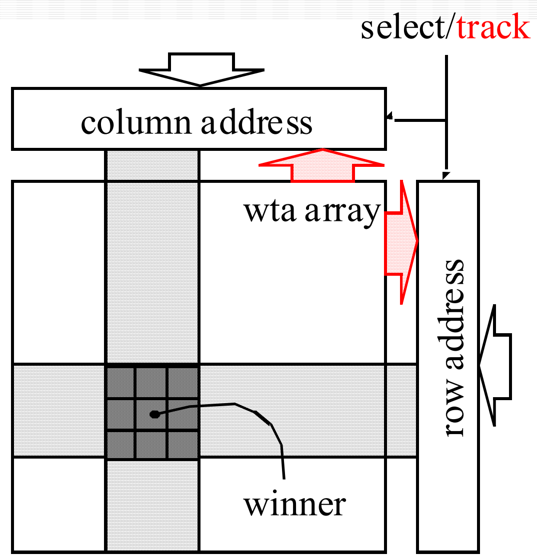
Exploitation of time-space interaction: Tracking with Attention

Winner-take-all selects the salient target
Adaptive attention shifts via top-down sensory feedback locks on target



Select Mode

arbitrary rectangle



Track Mode

hardwired 3x3 region

Sensory Computing for Efficiency

- The target task is known and “manageable”!
- If possible, (simplify and) hardwire the algorithm.
- Optical image IN – A few “measurements” OUT
- Advantages: very efficient (e.g, fast, cheap, low-power…)
- Disadvantages:
  - no flexibility,
  - could be fooled because only so much adaptation (feedback) can be accounted and wired at the design stage.
  - Probably cannot accommodate sophisticated signal processing algorithms (as we traditionally know them).

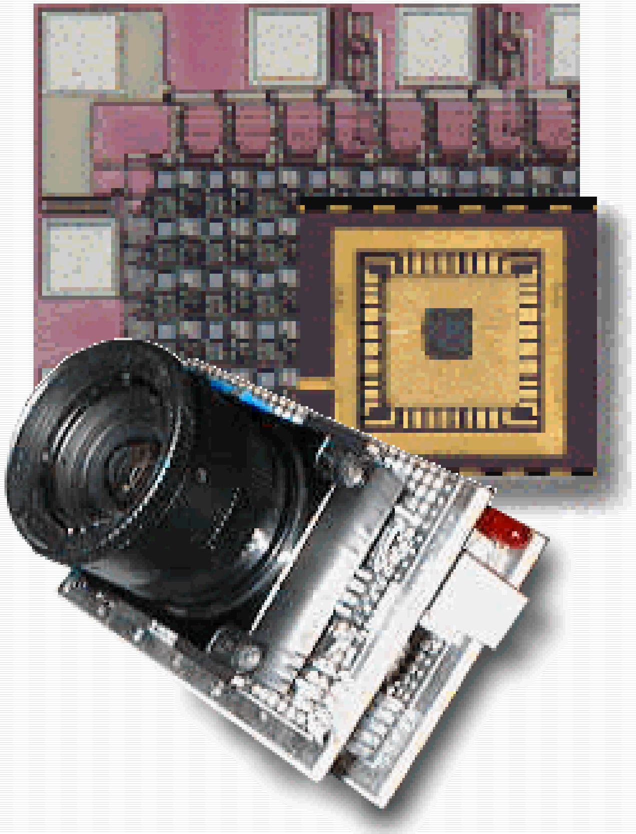
The assumption is that we have good image sensors! Actually we don't!

Conventional Sensors Fail!

(Trial Evidence)

Sensory Computing for Flexibility

- Computation at the sensory level that delivers information not obtainable otherwise (with conventional systems)
- Sensory computing extracts sensory information
- Delivers better images.



Things we could be improving by on-chip computation (for Flexibility)

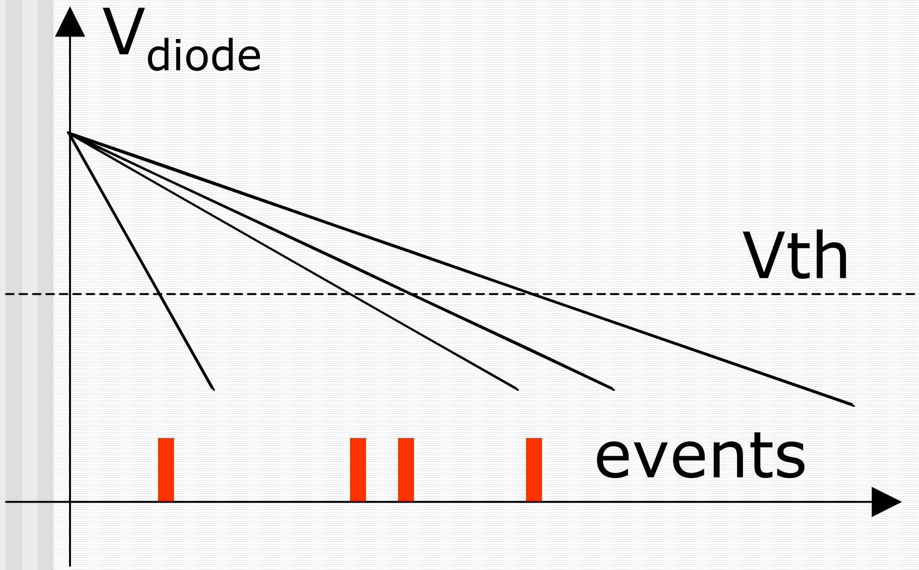
- Dynamic Range
- SNR (e.g., pixel binning)
- Adaptation to compensate for illumination-induced variations in optical images.
- Pixel uniformity
- Novel (information-theoretic) readout strategies
- …

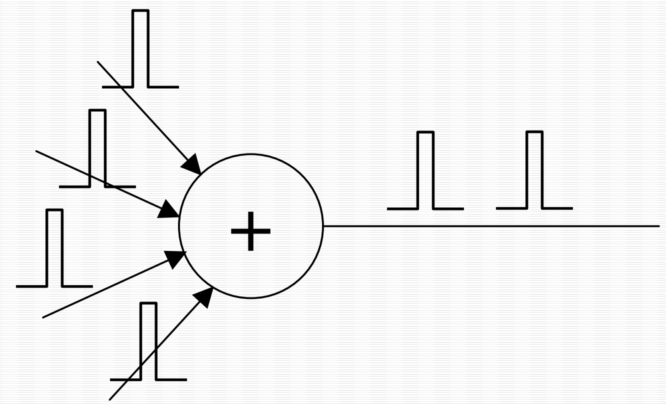
Some things can only be achieved on a chip: Dynamic Range Adaptation

Sensory Computing is probably not warranted for:

- Miniaturization of conventional image processing systems
  - One Test: If it can be done with CCD+DSP then it should be done so (use system-in-a-package)
- Speed, miniaturization, low power and cost are natural benefits, but not the primary reason for going on-chip.

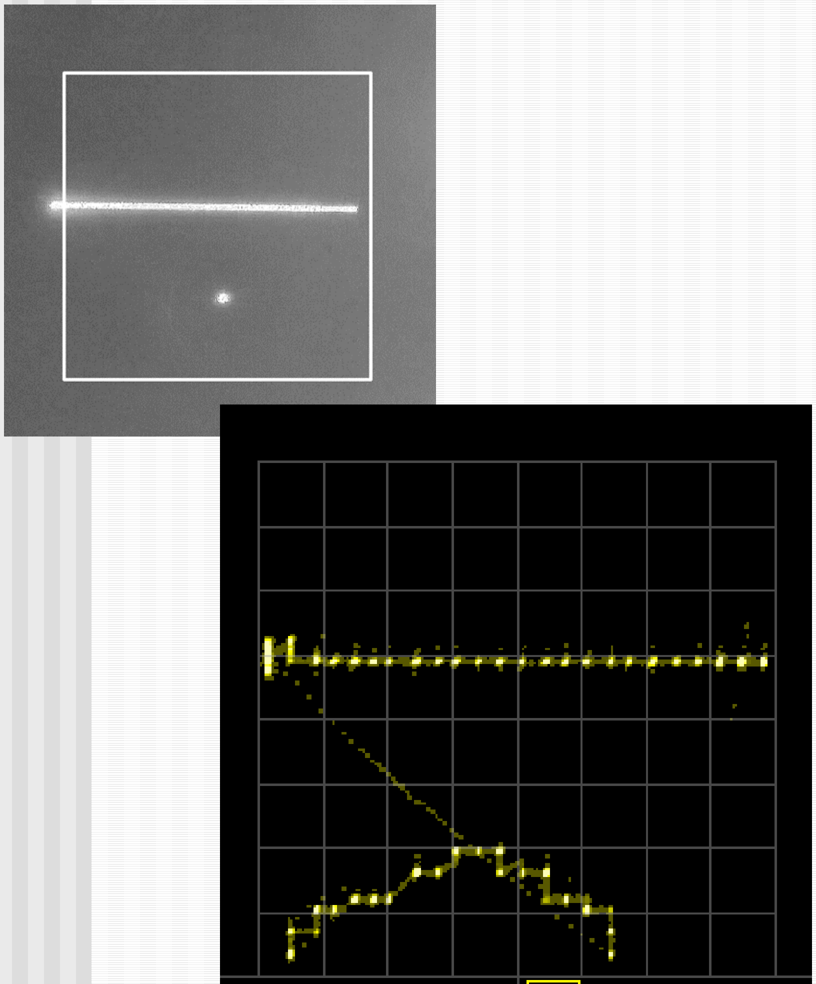
How to process on-chip(some ideas): Exploiting temporal dimension of sensing

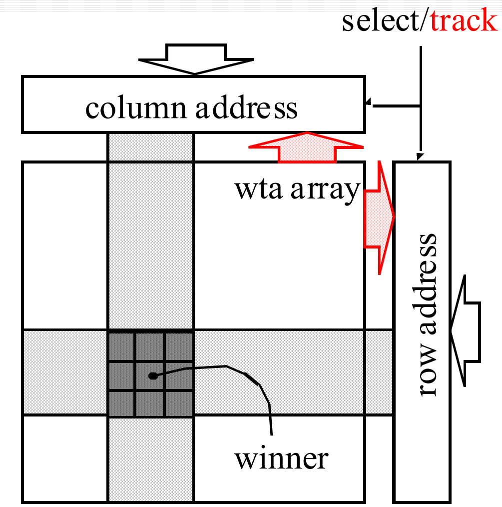




- Encoding contained in time intervals
- Large Number of Inputs
- Precision vs. Parallelism
- Asynchronous Signal Transmission
- Cooperative Parallelism

Exploiting spatio-temporal interaction (2D space+time)



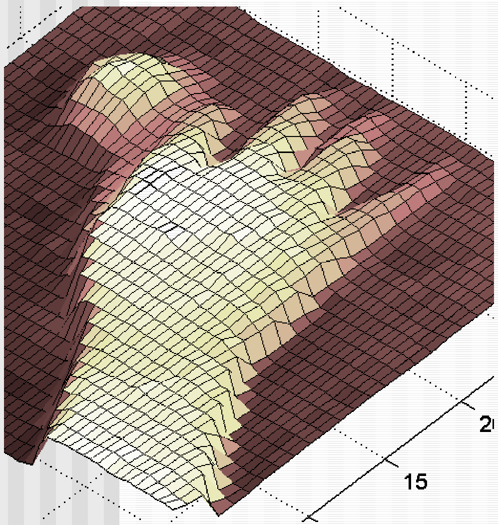


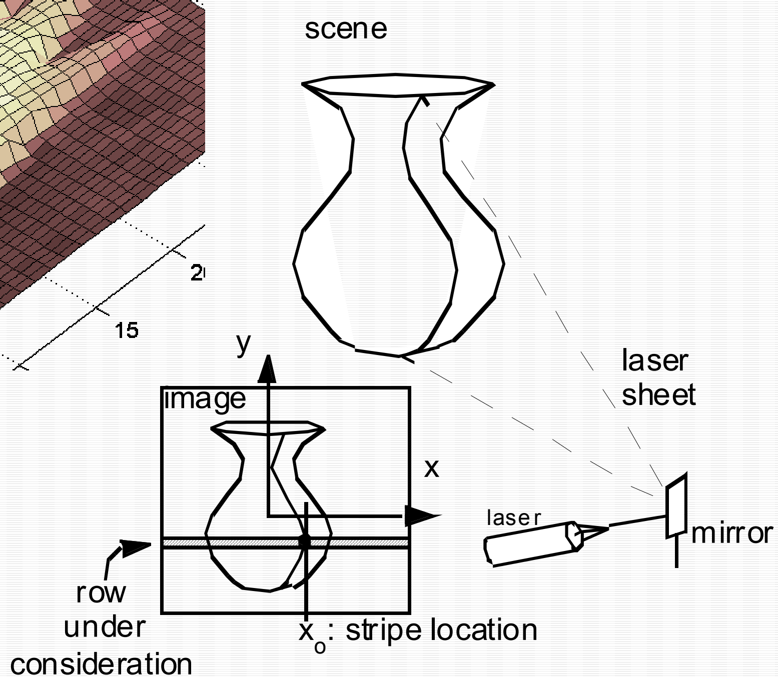
Track Mode

hardwired 3x3 region

Exploiting spatio-temporal interaction in image formation (3D space + time)

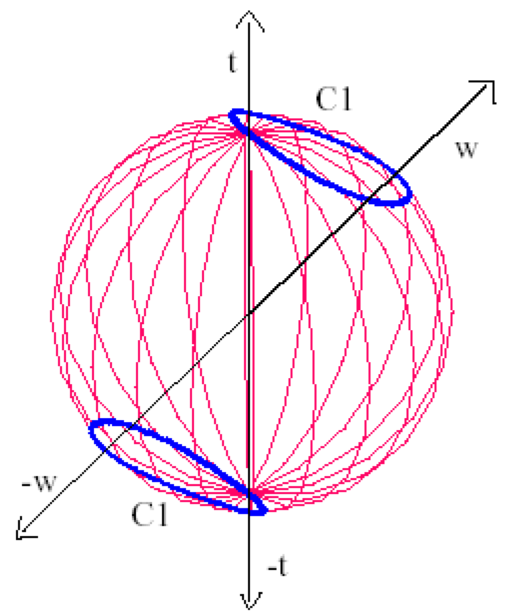
- Active Illumination example

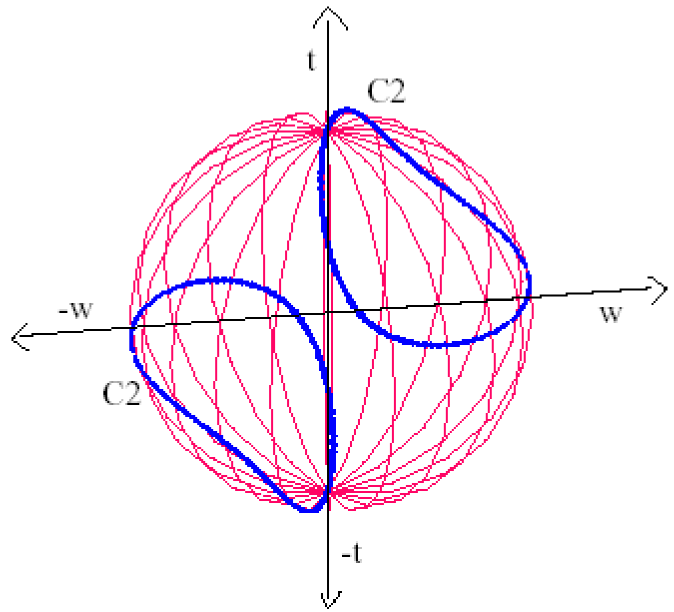




Exploiting spatio-temporal interaction in image formation (3D space + time)

- Unconventional projective geometries



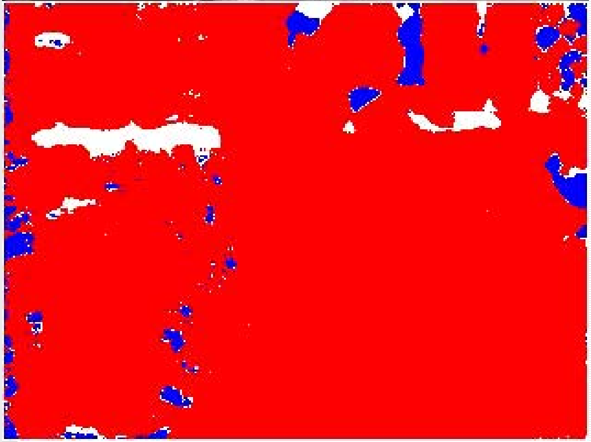


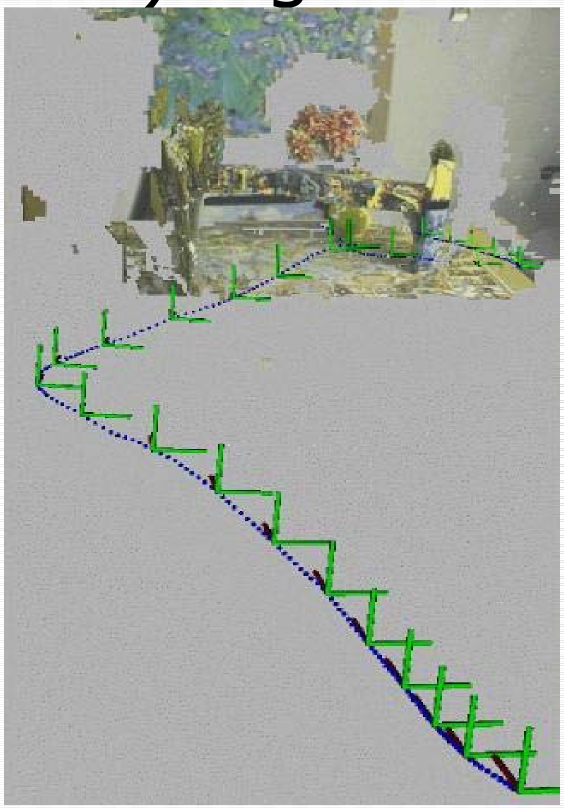
Motion computation on spherical retinae

Fermuller and Aloimonos, UMD

Exploiting spatio-temporal interaction in image formation (3D space + time)

- "Simplified" (unconventional) algorithms.

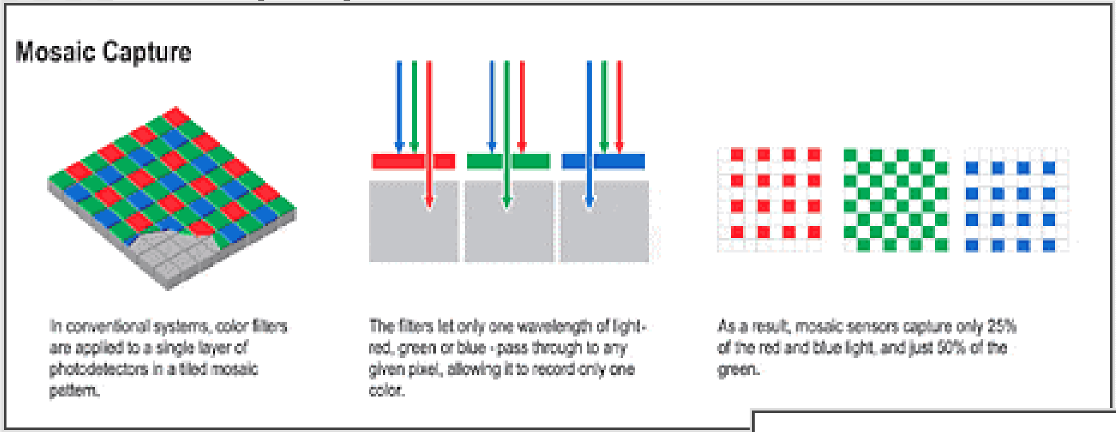


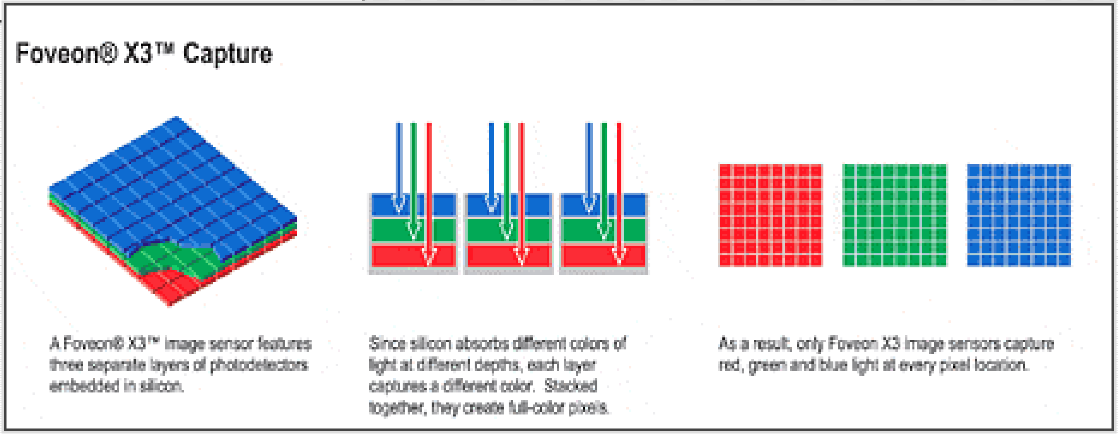


Fermuller and Aloimonos, UMD

How to “process” signals efficiently on-chip: Some Ideas

- Use optical/electronic processing available in the physics of the semiconductor.

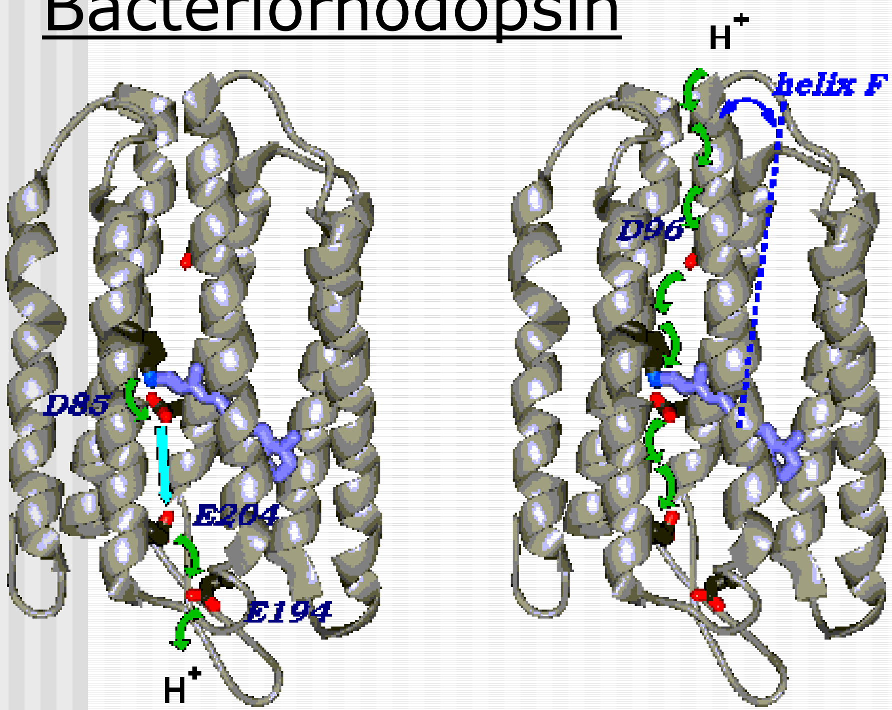


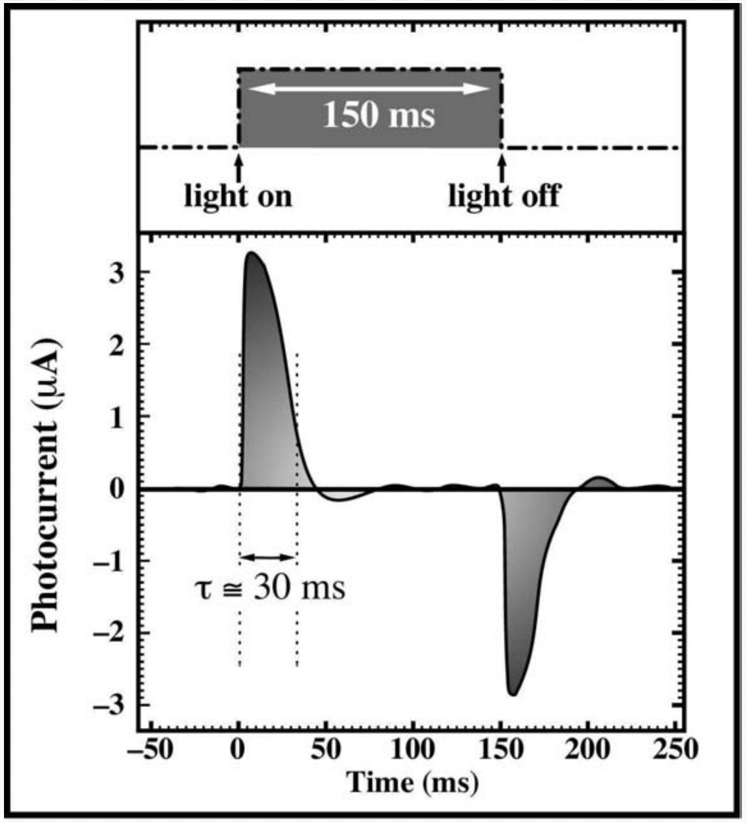


How to do it efficiently on-chip: Some Ideas

- New materials that exhibit some form of image processing.

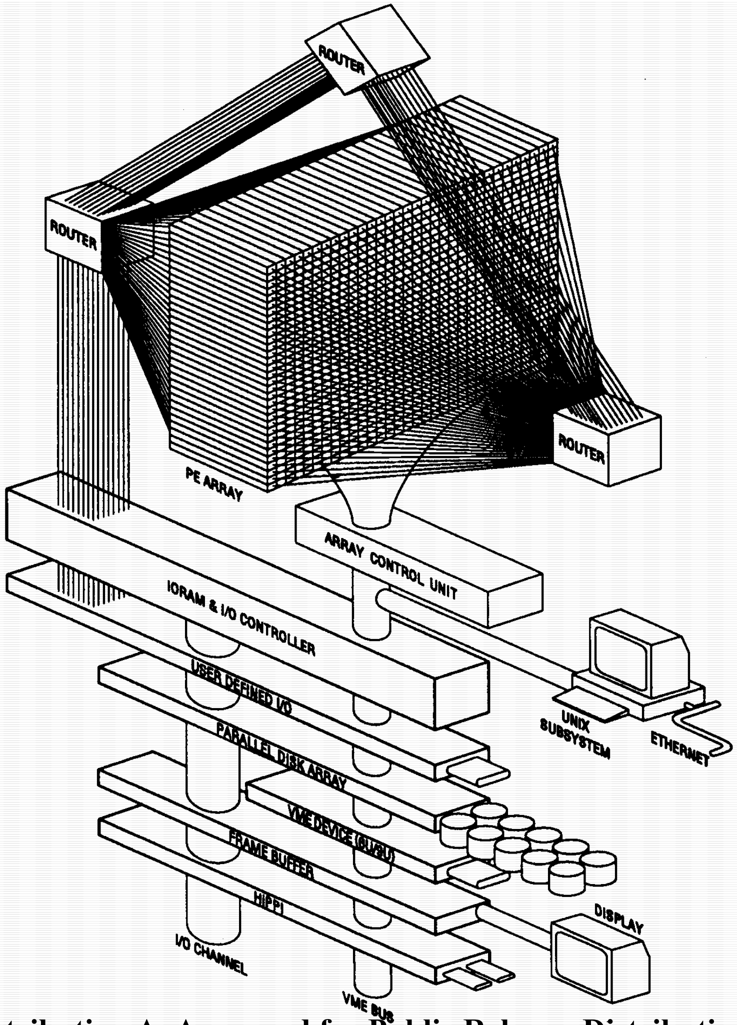
Bacteriorhodopsin



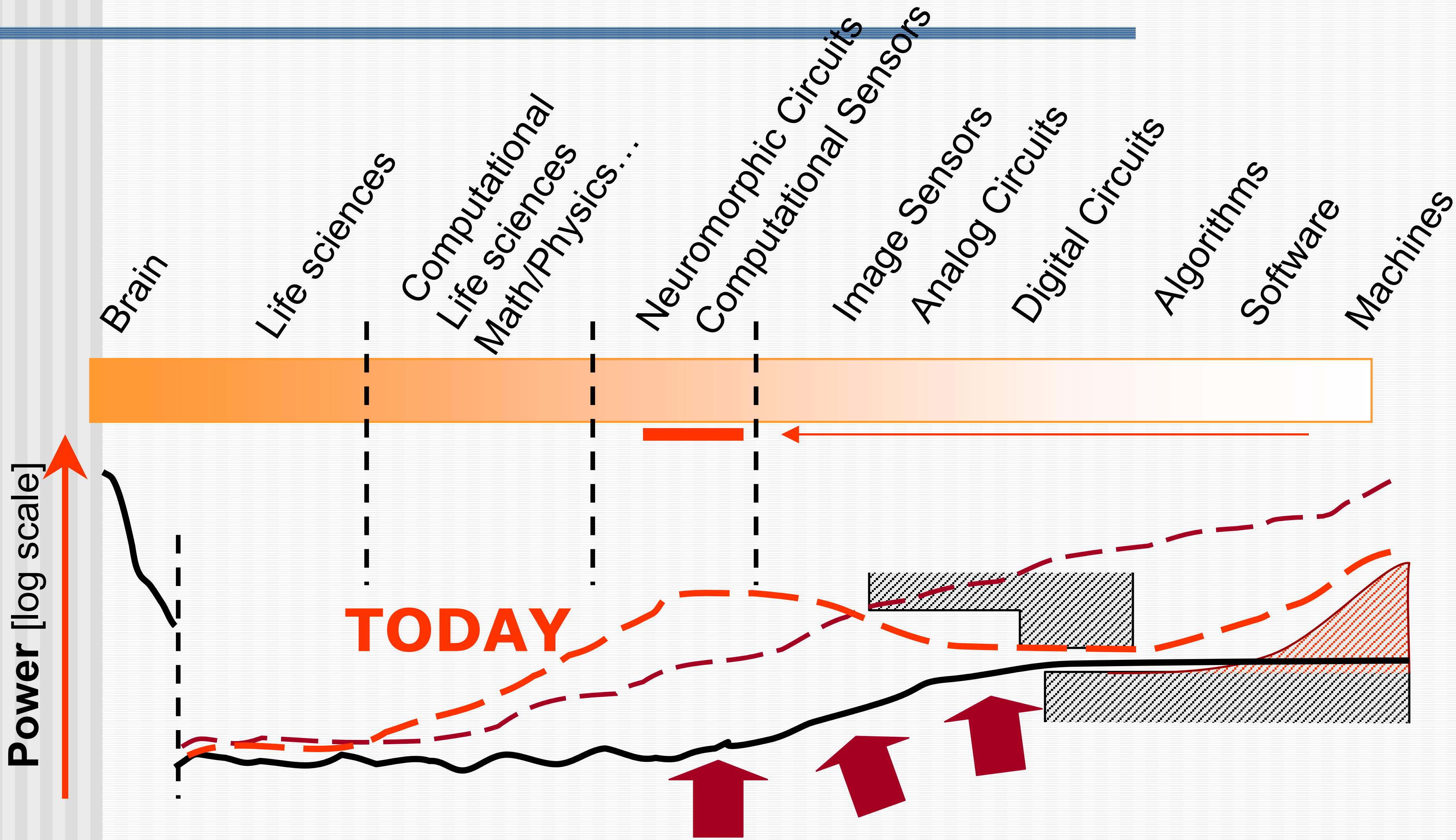


Vision Bottlenecks in Machines

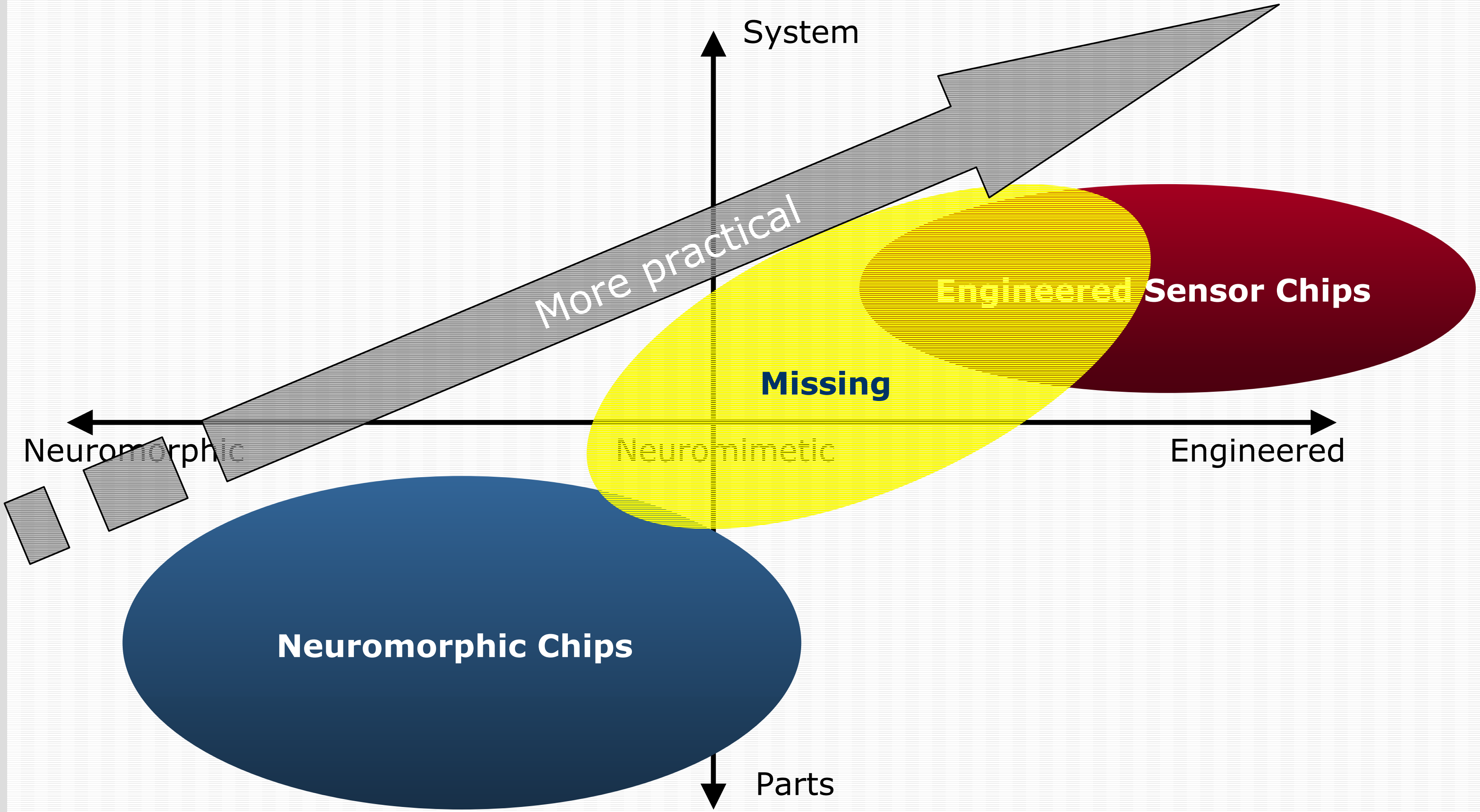
Old thinking



From Brains to Machines: Think Multidisciplinary

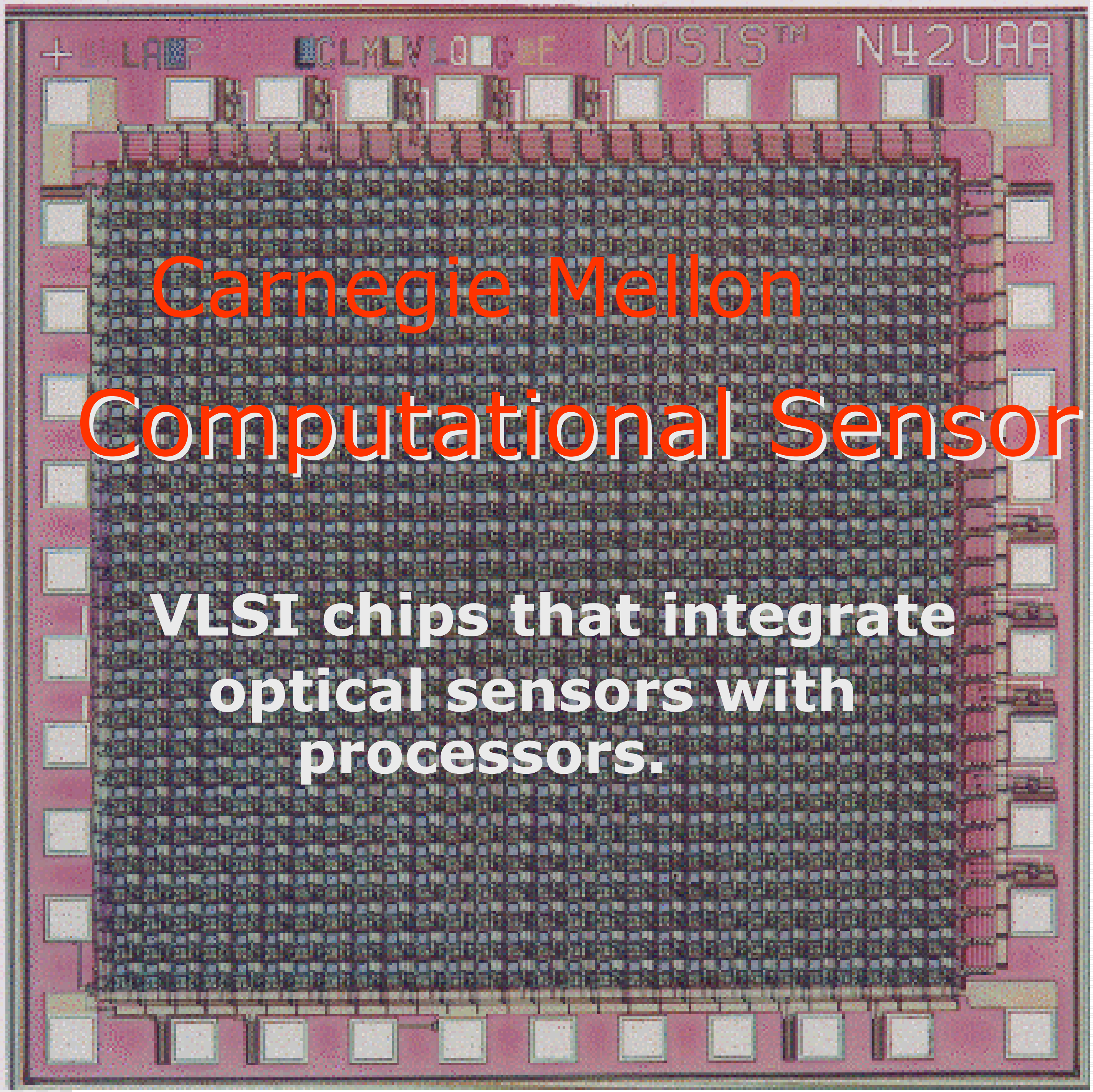


Smart Imaging Chips Space



Sensory Computing: Be Creative

- **When**:Use Sensory Computing to obtain useful environmental information from ordinary "imperfect" detectors
- No conventional system substitute as to:
  - Signal utilization, manipulation and information delivery
- Secondary benefits are: Speed (low latency), low-power, low system complexity, low cost
- Don't forget practical aspects: resolution, image/signal quality, speed, etc.



Carnegie Mellon

Computational Sensor Lab

VLSI chips that integrate optical sensors with processors.

Q&A