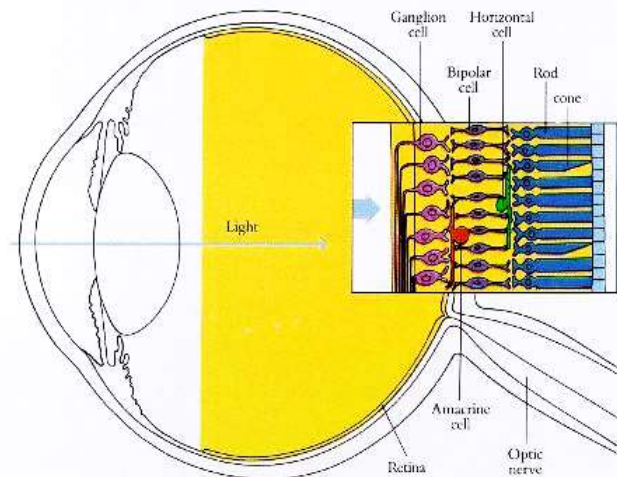


The development of the DVS and DAVIS sensors

Tobi Delbruck

Inst. of Neuroinformatics, University of Zurich and ETH Zurich

Sensors Group sensors.ini.uzh.ch



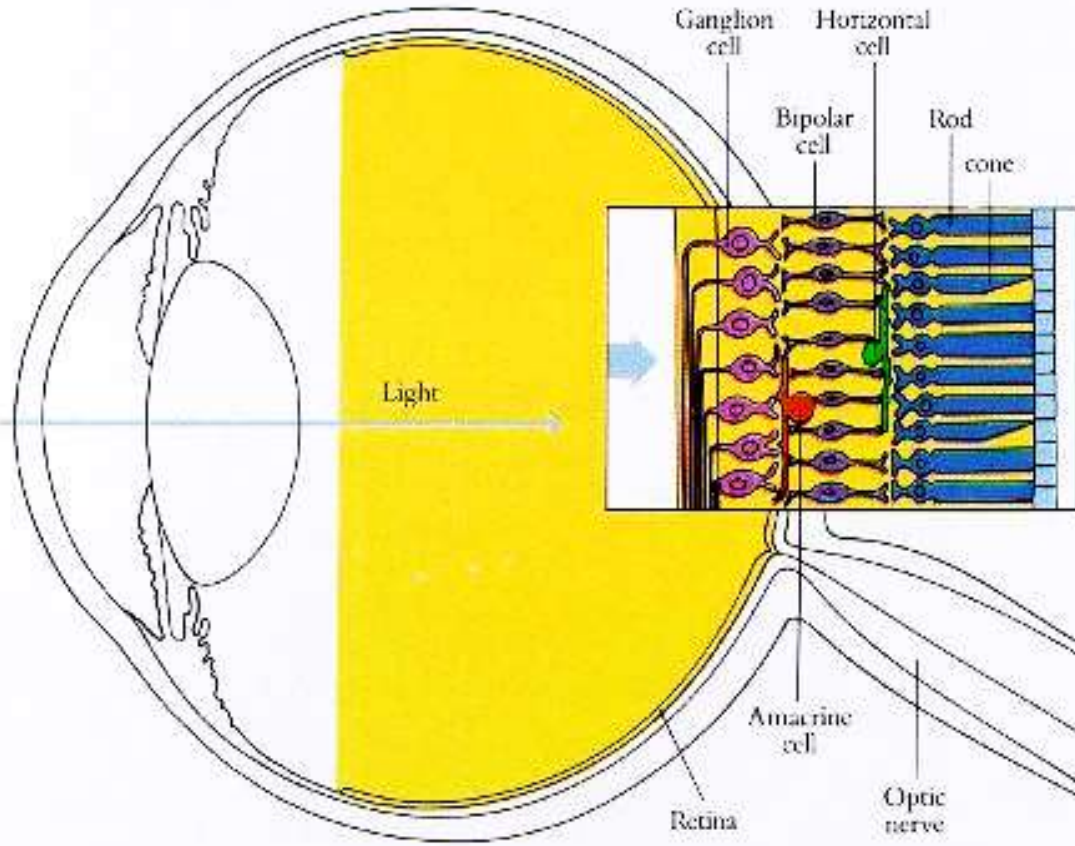
Sponsors: Swiss National Science Foundation **NCCR Robotics** project, EU projects **SEEBETTER** and **VISUALISE**, Samsung, DARPA



sensors.ini.uzh.ch
inilabs.com

Sponsors: Swiss National Science Foundation **NCCR Robotics**, EU projects **CAVIAR**, **SEEBETTER**, **VISUALISE**, **Samsung**, **DARPA**, **University of Zurich** and **ETH Zurich**

The Human Eye as a digital camera



100M photoreceptors

1M output fibers carrying max
100Hz spike rates

180dB (10^9) operating range

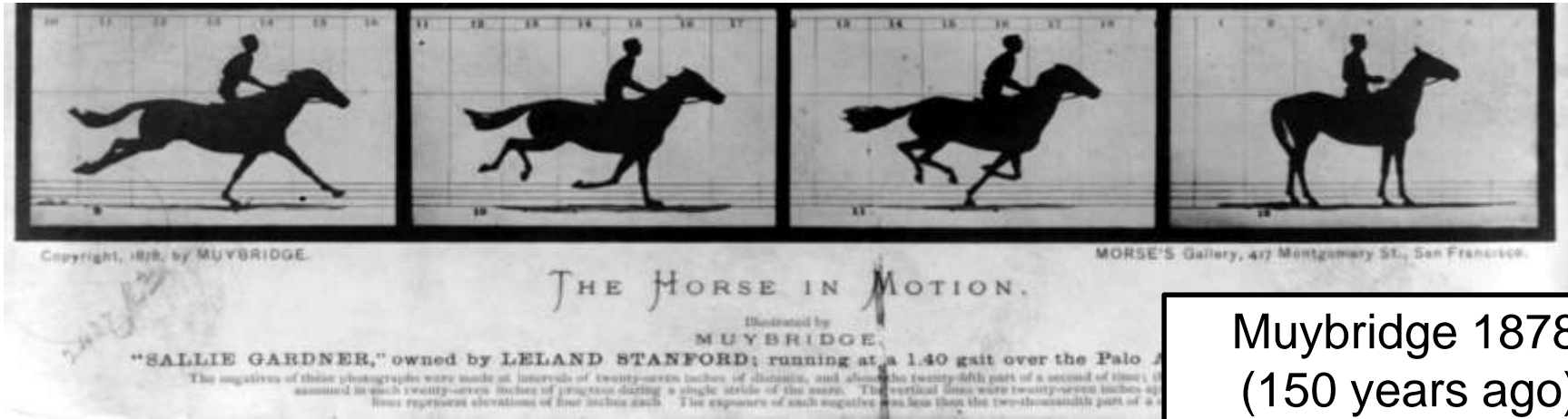
>20 different “eyes”

Many GOPs computing

3mW power consumption

Output is sparse,
asynchronous stream of digital
spike events

Conventional cameras (**Static vision sensors**) output a stroboscopic sequence of frames



Muybridge 1878
(150 years ago)

Good

Compatible with 50+years of machine vision
Allows small pixels (1um for consumer, 3-5um for machine vision)

Bad

Redundant output
Temporal aliasing
Limited dynamic range (60dB)

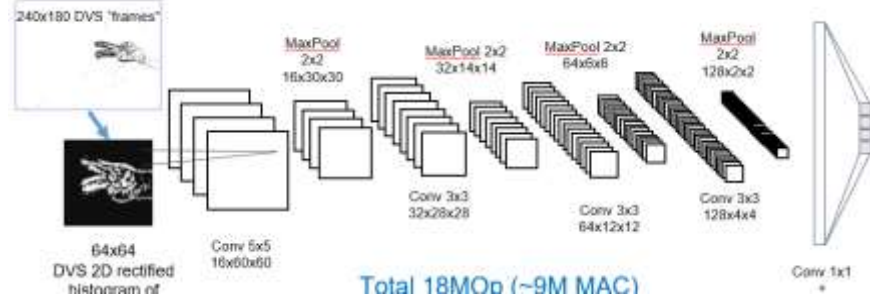
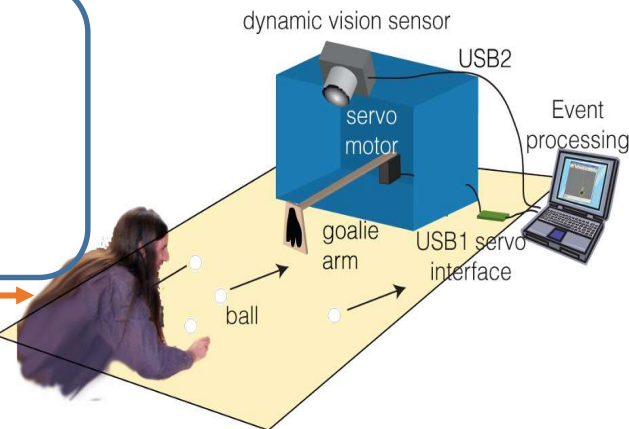
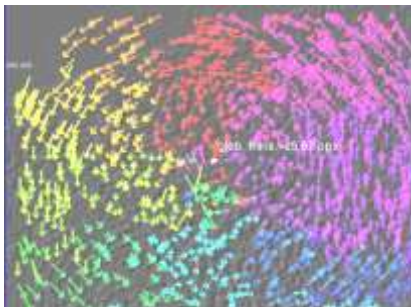
Fundamental “latency vs. power” trade-off

This talk has 4 parts (but I will only show first part)

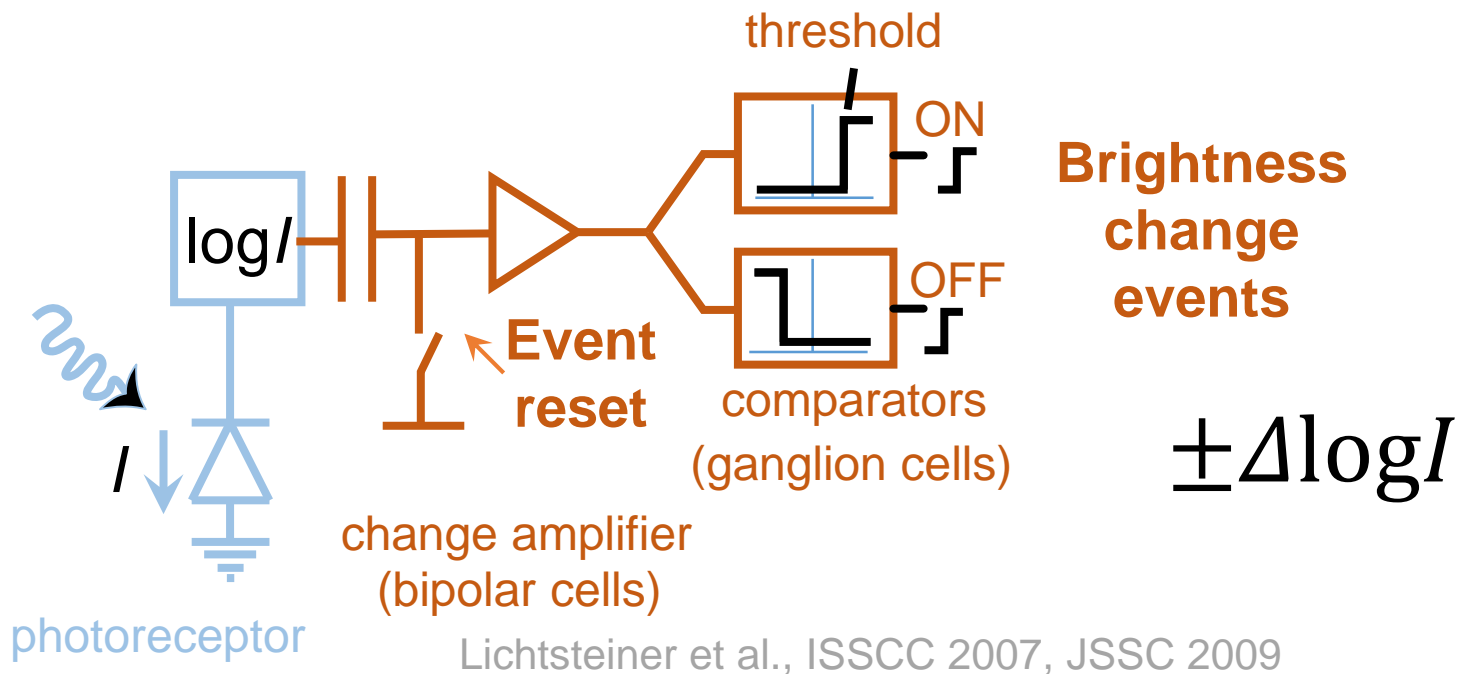
- **Dynamic Vision Sensor Silicon Retinas**



- Simple object tracking by algorithmic processing of events
- DVS Optical Flow architecture
- “Data-driven” deep inference with CNNs

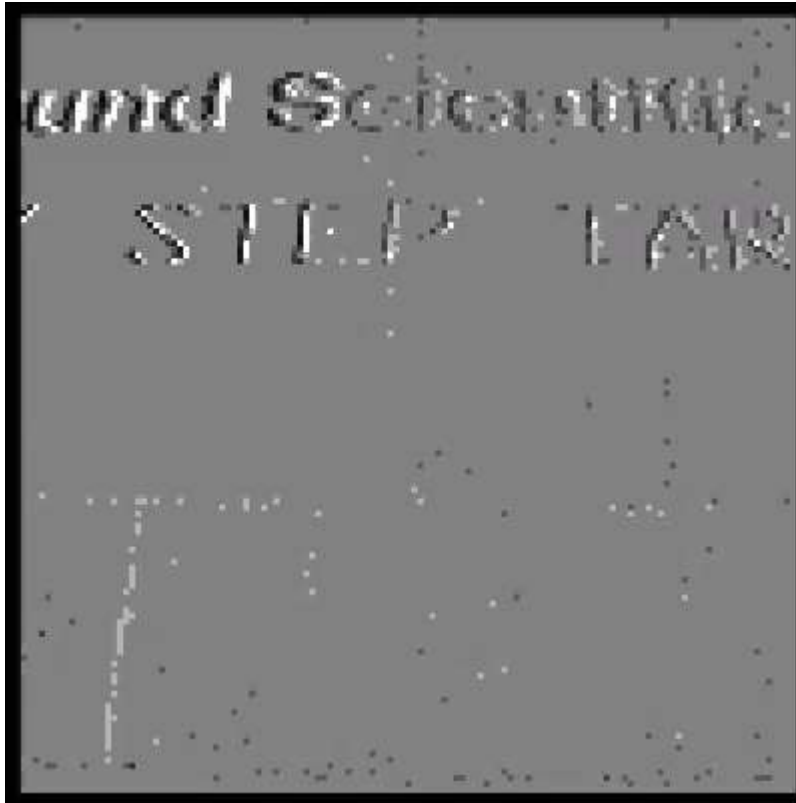


DVS (Dynamic Vision Sensor) Pixel



From Rodieck 1998

DVS pixel has wide dynamic range

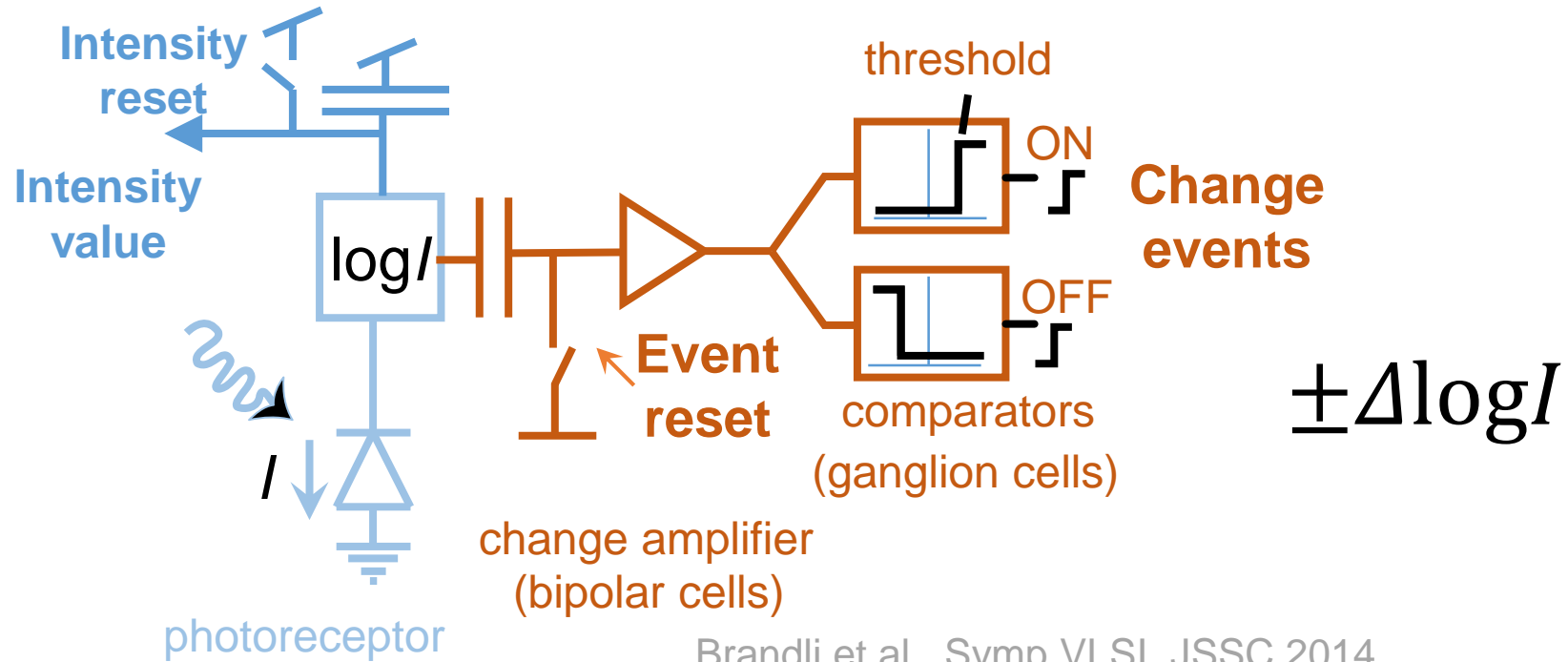


780 lux : 5.8 lux



Edmund 0.1 density chart
Illumination ratio=135:1

DAVIS (Dynamic and Active Pixel Vision Sensor) Pixel

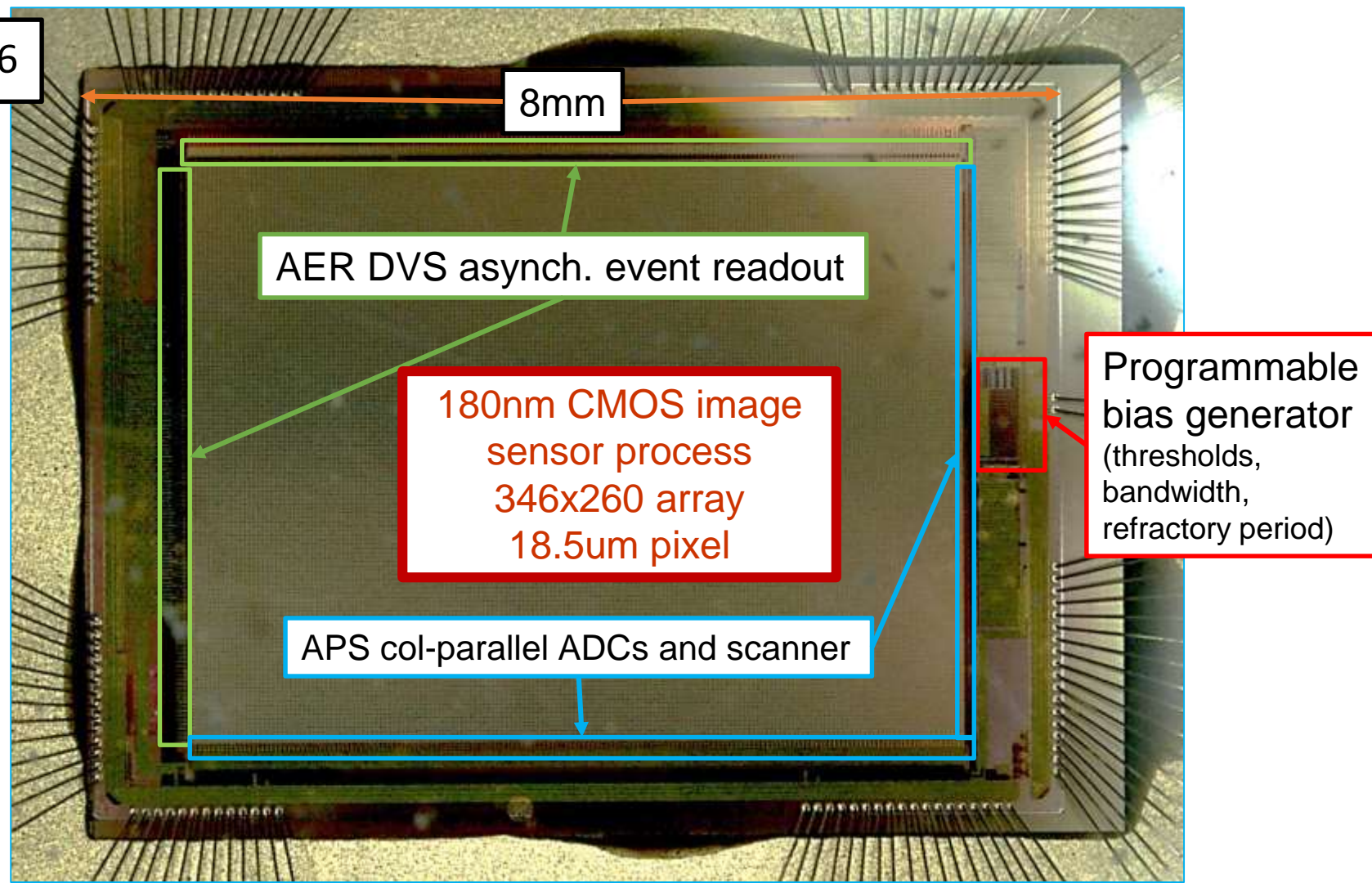


Brandli et al., Symp VLSI, JSSC 2014



From Rodieck 1998

DAVIS346



Resolution: 1, 0.4 ms. Frame period: 98.07 ms (11.227122 Hz)



DVS/DAVIS +IMU demo



Start DAVIS
Demo

DVS sensor specifications

	TEMPDIFF128	R
Functionality	Asynchronous temporal contrast	F
Pixel size μm (λ)	40x40 (200nm)	
Fill factor (%)	8.1% (PD area 1)	
Fabrication process	4M 2P 0.35	
Pixel complexity	26 transistors (analog), 3	
Array size	128x128	
Die size mm^2	6x6.3	
Interface	15-bit word-parallel AER	
Power consumption	24mW @ 3.3V 1.5mA core 0.3mA logic 5.5mA biases	
Dynamic range	120dB 2 lux to > 100 klux scene illumination with f/1.2 lens	
Photodiode dark current at room temperature	4fA ($\sim 10\text{nA}/\text{cm}^2$) Nwell photodiode	2
Response latency Frames/se or bandwidth	15 μs @ 1 klux chip illumination $\sim 1\text{M}$ events/sec	< 6
FPN, matching	2.1% contrast	2

Power consumption	24mW @ 3.3V	3
-------------------	-------------	---

Only at die level.

USB DVS cameras burn $\sim 500\text{mW}$

Dynamic range	120dB	1
	2 lux to > 100 klux	

Read test conditions

Response latency	15 μs @ 1 klux chip illumination	< 6
------------------	---	-----

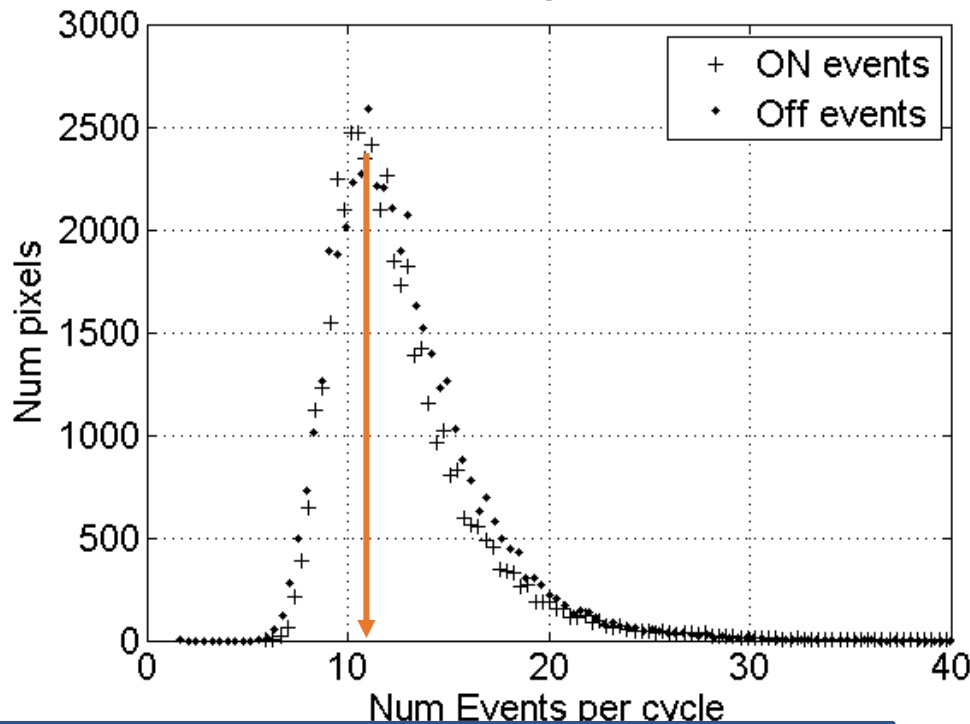
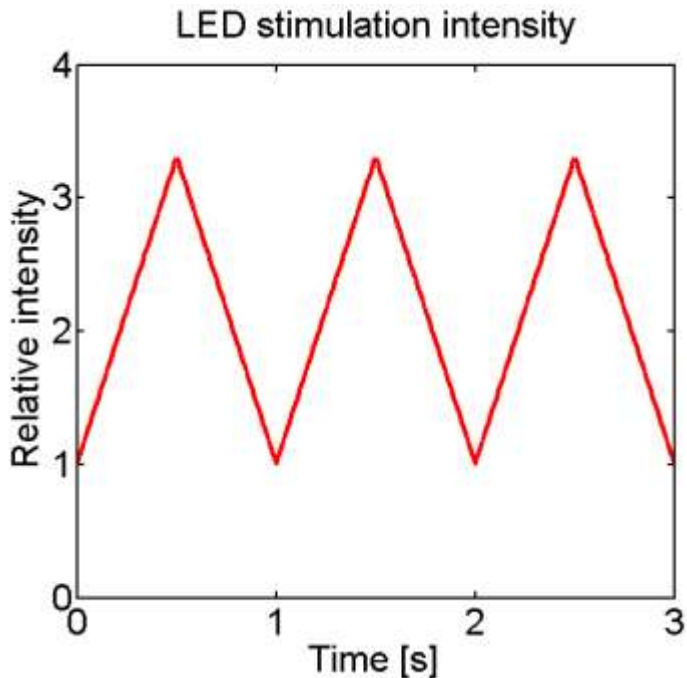
Artificial conditions. Real world: 100us-10ms

FPN, matching	2.1% contrast	2
---------------	---------------	---

Pixel-to-pixel event threshold matching. Important for modeling DVS, e.g. Bayes

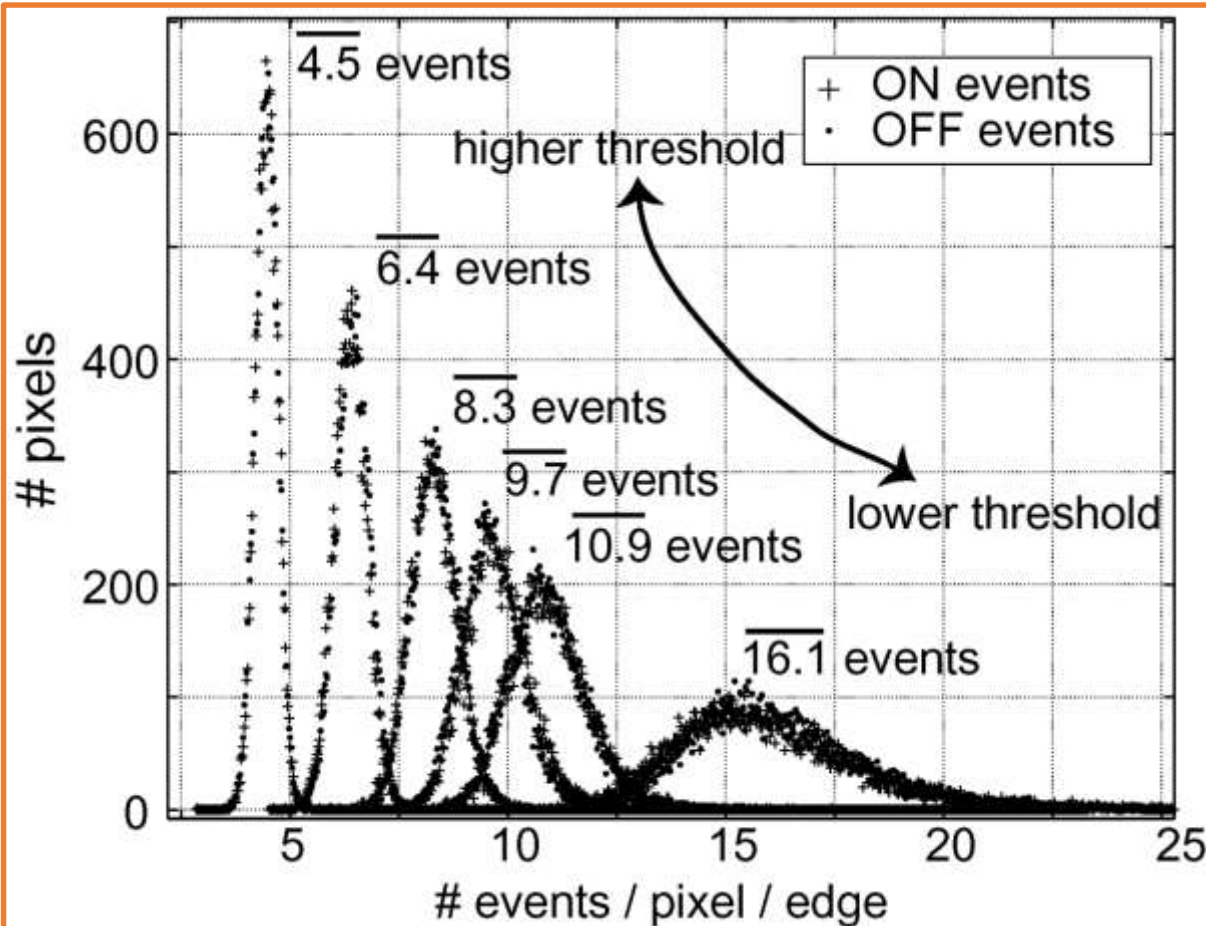
Event threshold matching measurement

Experiment: Apply slow triangle wave LED stimulus to entire array, measure number of events that pixels generate

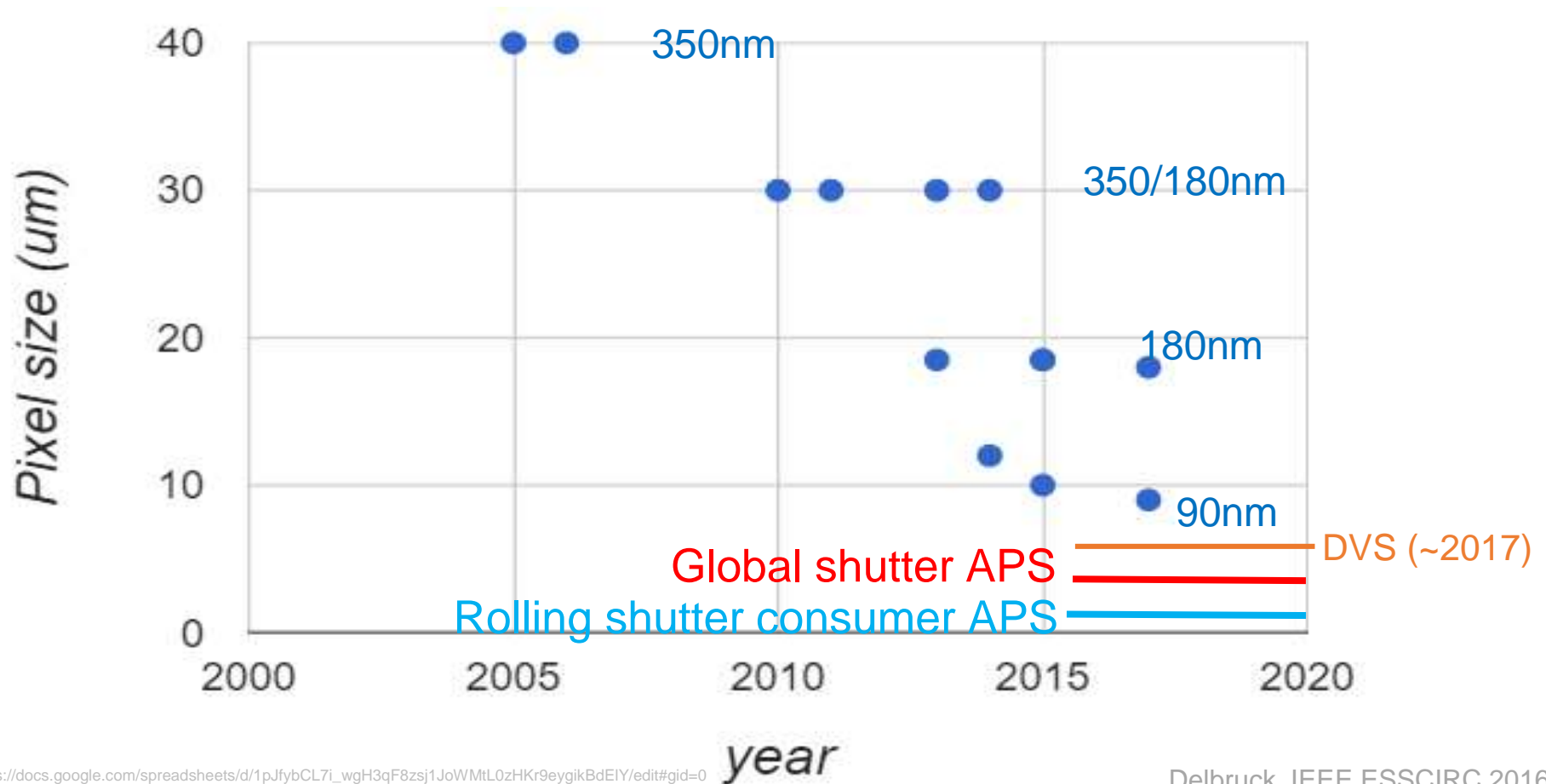


Conclusion: Pixels generate 11 ± 3 events per factor 3.3 contrast.
Since $\ln(3.3) = 1.19$ and $1.19/11 = 0.11$, contrast threshold = $11\% \pm 4\%$

Event threshold matching measurement



DVS pixel size trend



Event camera silicon retina developments





www.iniLabs.com

Founded 2009 Run as not-for-profit

Neuromorphic sensor R&D prototypes

[Open source software, user guides,
app notes, sample data](#)

**Shipped devices based on EU funded silicon to
>200 organizations**

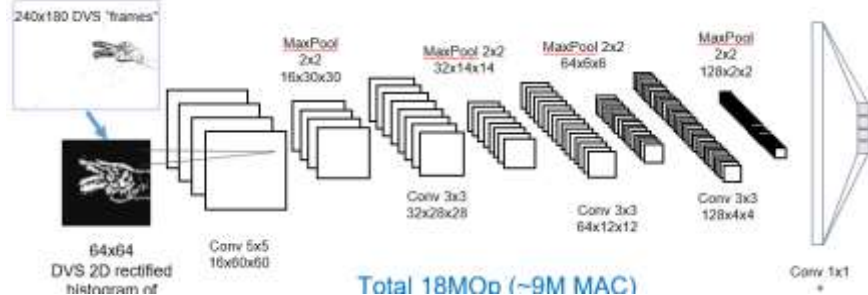
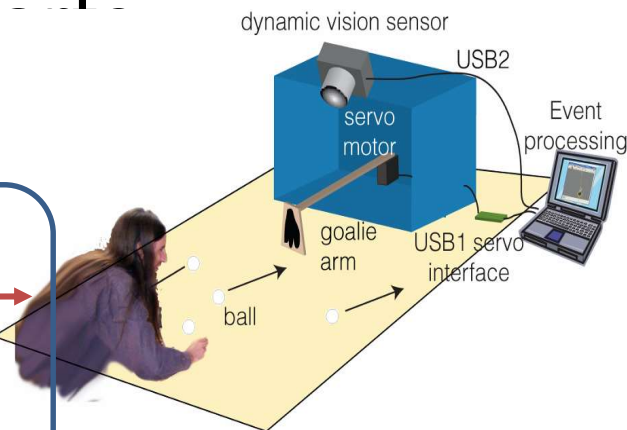


This talk 4 p

- Dynamic Vision Sensor
Silicon Retinas

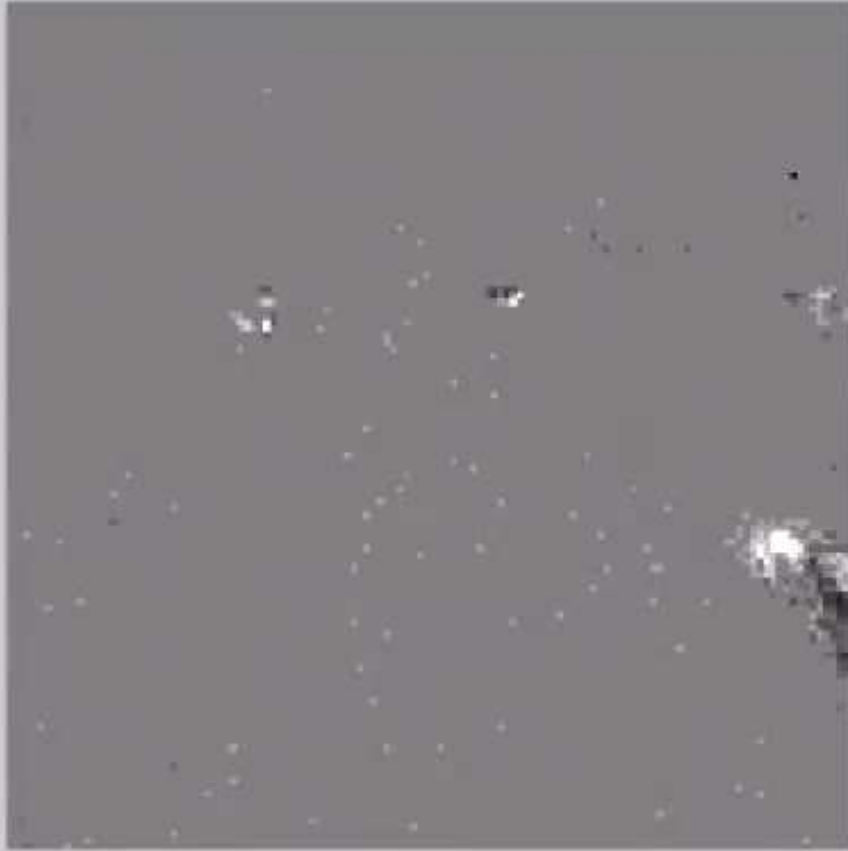
- Simple object tracking
by algorithmic
processing of events

- DVS Optical Flow
architecture
- “Data-driven” deep
inference with CNNs



Tracking objects from DVS events using spatio-temporal coherence

40ms@3.746/78.5s, 1133-evts, 0keps, FS=3 evts, Fwd

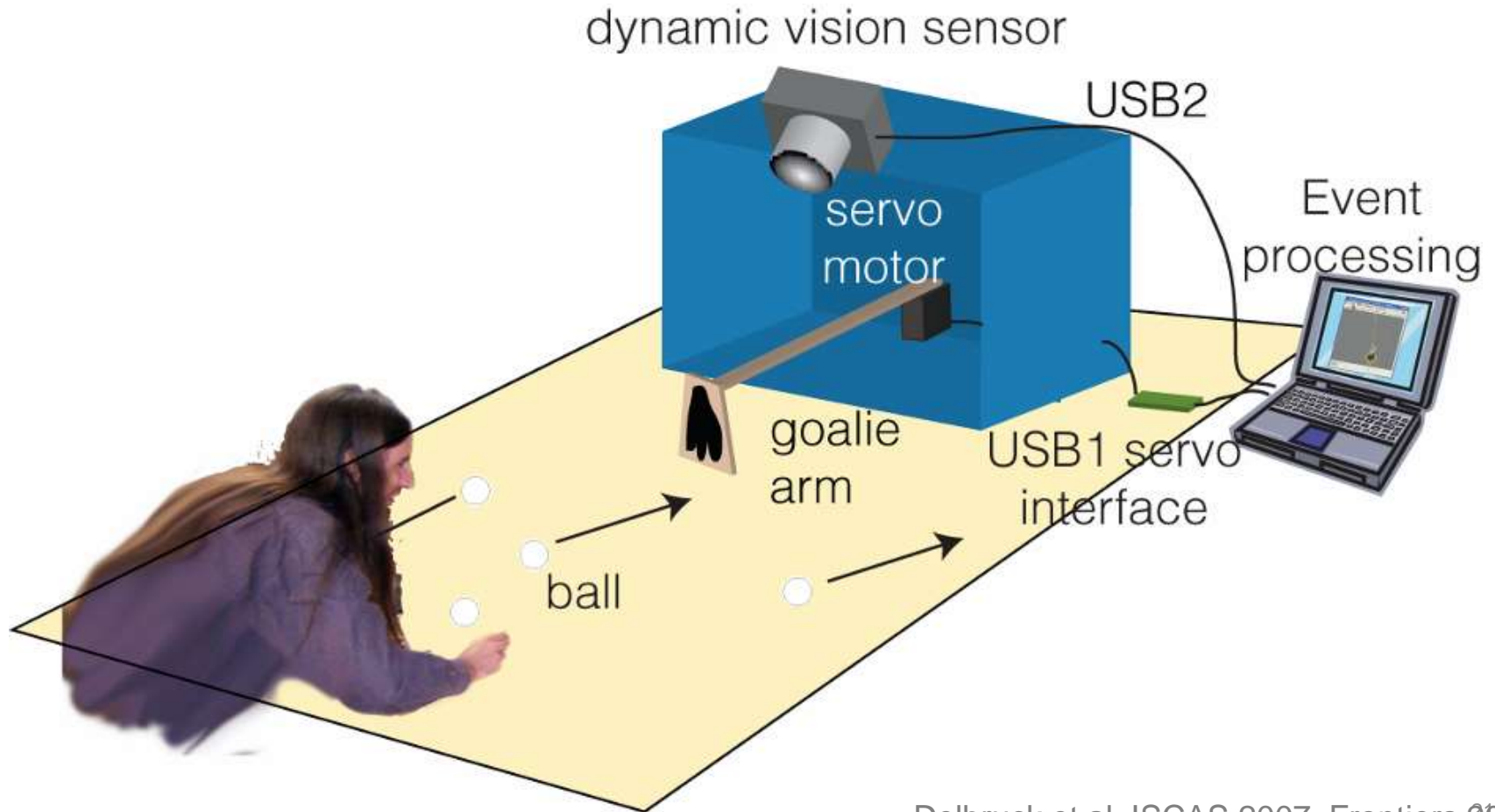


1. For each event, find nearest cluster
 - If event within a cluster, move cluster
 - If event not within cluster, seed new cluster
2. Periodically prune starved clusters, merge clusters, etc (lifetime mgmt)


Advantages

1. Low computational cost (e.g. <5% CPU)
2. No frame memory (~100 bytes/object).
3. No frame correspondence problem

Robo Goalie



Using DVS allows 2 ms reaction time at 4% processor load with USB bus connections

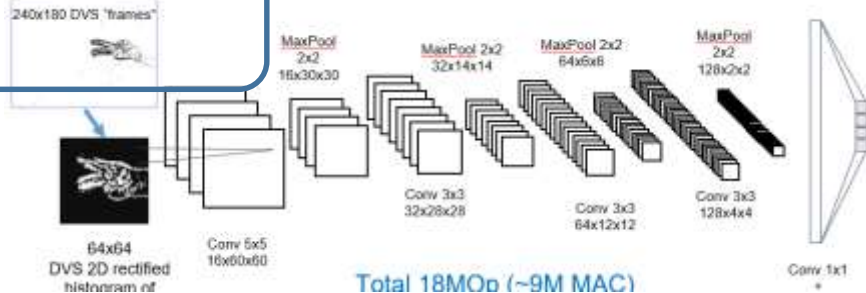
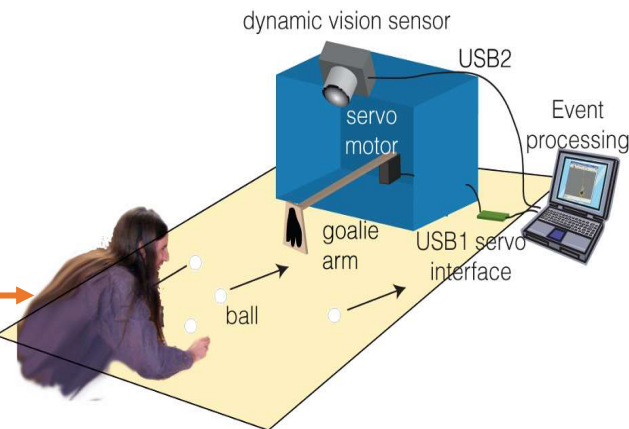


RoboGoalie
www.ini.uzh.ch
2007

<3% laptop CPU load
<3ms control latency

This talk has 4 parts

- Dynamic Vision Sensor Silicon Retinas
- Simple object tracking by algorithmic processing of events
- **DVS Optical Flow architecture**
- “Data-driven” deep inference with CNNs



Presented last Monday at ISCAS 2015 in Baltimore

Block-Matching Optical Flow for Dynamic Vision Sensor: Algorithm and FPGA Implementation



Min Liu

Liu, Min, and Tobi Delbruck. 2017. "Block-Matching Optical Flow for Dynamic Vision Sensor: Algorithm and FPGA Implementation." In *2017 IEEE Symposium on Circuits and Systems (ISCAS 2017)*, in press. Baltimore, MD, USA.

Why do we need rapid and low power optic flow?

Human first person view drone racing



Child runs in front of car



<https://youtu.be/nLUmW6OfEy0>

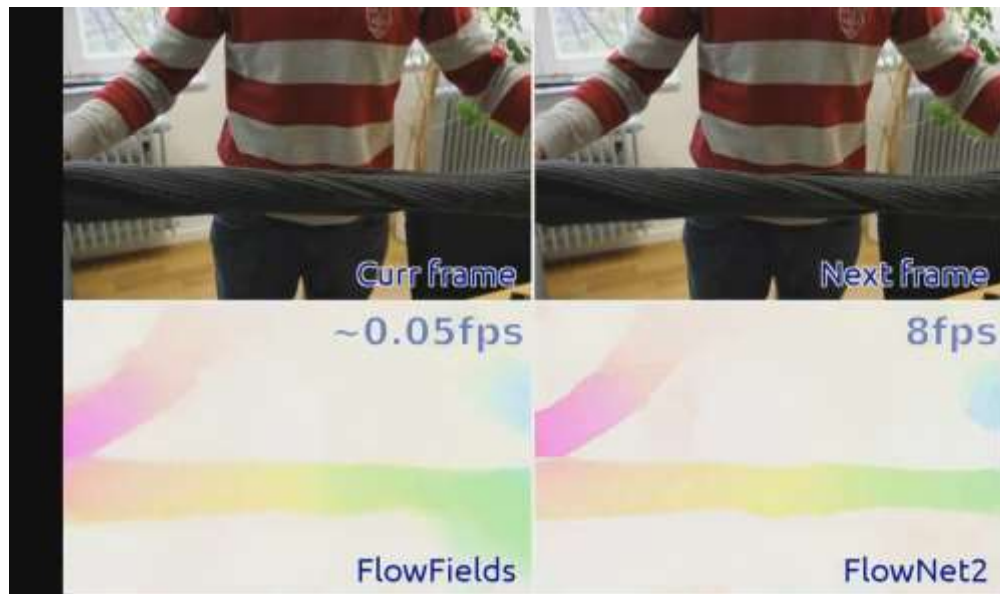
<https://youtu.be/UsoxrsrsdgA>

Optical flow could be key part of enabling solutions to these problems

Example of State of the Art Deep Learning Optic Flow

“FlowNet 2.0”

1. Computes dense OF with amazing accuracy
2. Uses a complex stack of multiple CNNs that process image pairs
3. Requires labeled training data that is difficult to obtain and the CNN is difficult to train
4. Requires powerful PC plus GPU with **~400W power consumption** to run HD video at 8 frames/second



Ilg, Eddy, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. 2016. “FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks.” arXiv:1612.01925 [Cs], December.

<http://arxiv.org/abs/1612.01925>.

<https://www.youtube.com/watch?v=JSzUdVBmQP4>

Prior work for DVS optical flow

Delbruck, 2007, jAER project
Benosman et al., Neural Networks 2012
Benosman et al., IEEE TNNLS 2013
Orchard et al., BioCAS 2013
Barranco et al., Proc IEEE, 2014
Brosch et al., Frontiers 2015
Mueggler et al, ICRA 2015
Conradt, ROBIO 2015
Rueckauer and Delbruck, Frontiers 2016
Bardow et al., CVPR 2016

Existing methods are serial algorithms that robustly solve linear or nonlinear constraints. They require several us/event on fast PC or PC + GPU.

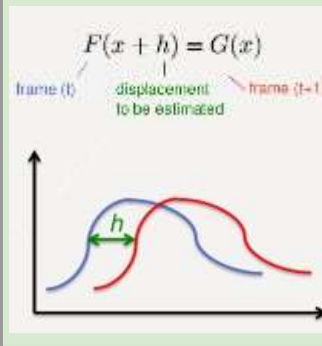
Methods for DVS flow

Spatio-Temporal Correlation or BioInspired



Brosch et al., Frontiers 2015

Adapted CV Method, e.g. Lucas-Kanade



Variational + Probabilistic

$$\min_{u,v} \int_0^1 \int \left(\lambda_1 \|u_x\|_1 + \lambda_2 \|u_y\|_1 + \lambda_3 \|L_x\|_1 \right) dx dy + \int_0^1 \sum_{n=1}^{(P-1)} \left(\lambda_4 \|L_n - L_{n-1}\|_1 + \lambda_5 \|L_n - L(t_p)\|_1 \right) dt$$

Bardow et al., CVPR 2016

Our work is inspired by MPEG motion estimation hardware. We seek **semi dense OF** architecture for **event-based vision sensors** that is **easily & cheaply** implemented in digital logic

Block matching DVS flow

Search finds best matching block in slice $t-2d$

1 bitmaps

d = sample interval
= slice duration

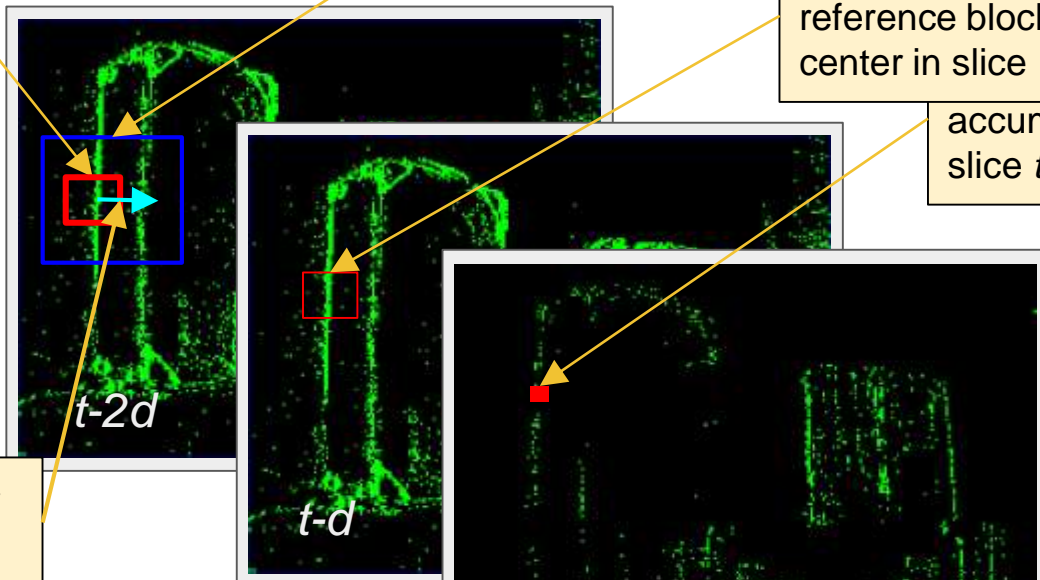
$d=100\text{ms}$ here

Resulting flow vector
 $v_x, v_y = d_x, d_y / d$

And center of
search region in
slice $t-2d$

Event specifies
reference block
center in slice $t-d$

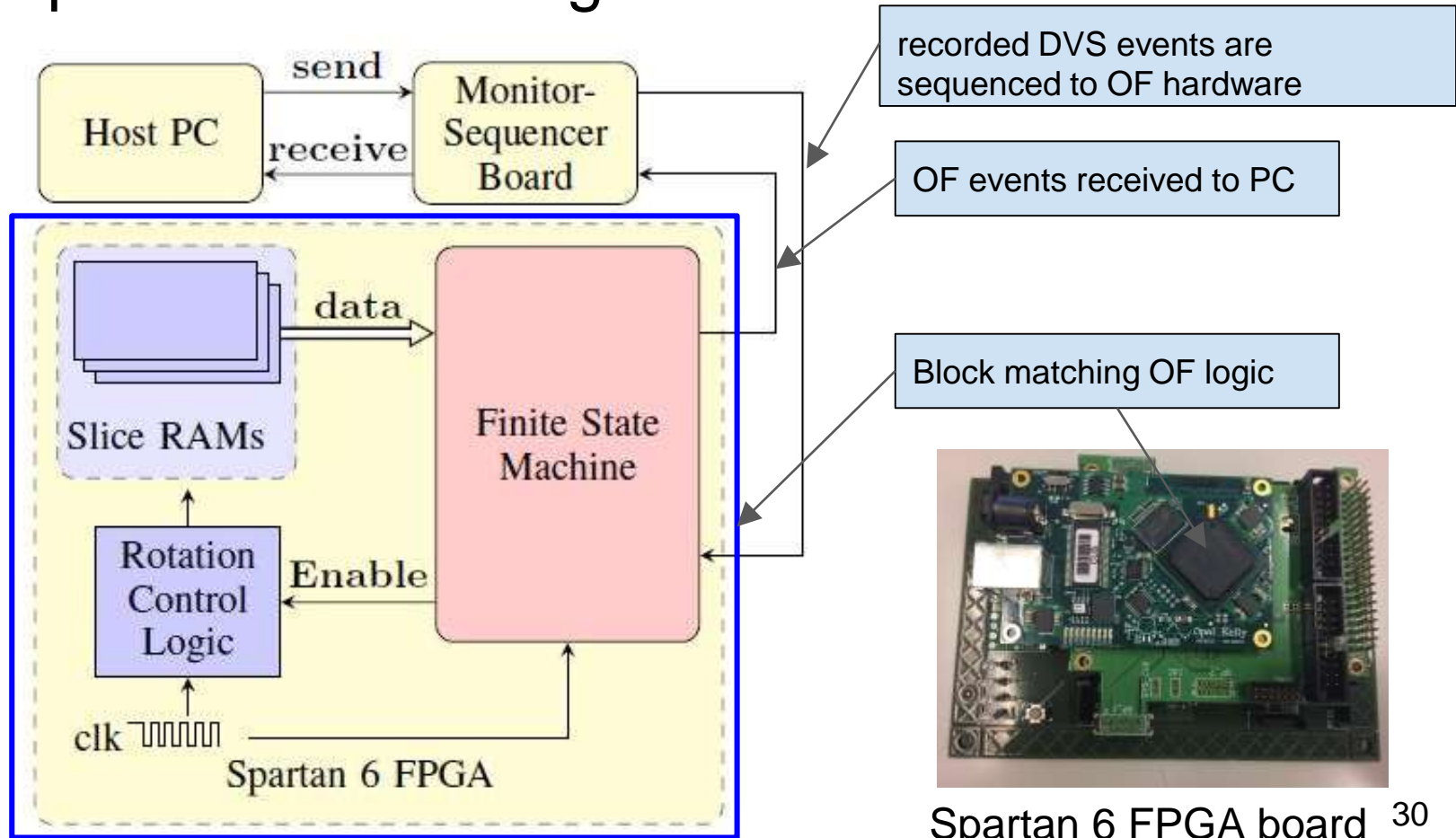
accumulated to
slice t



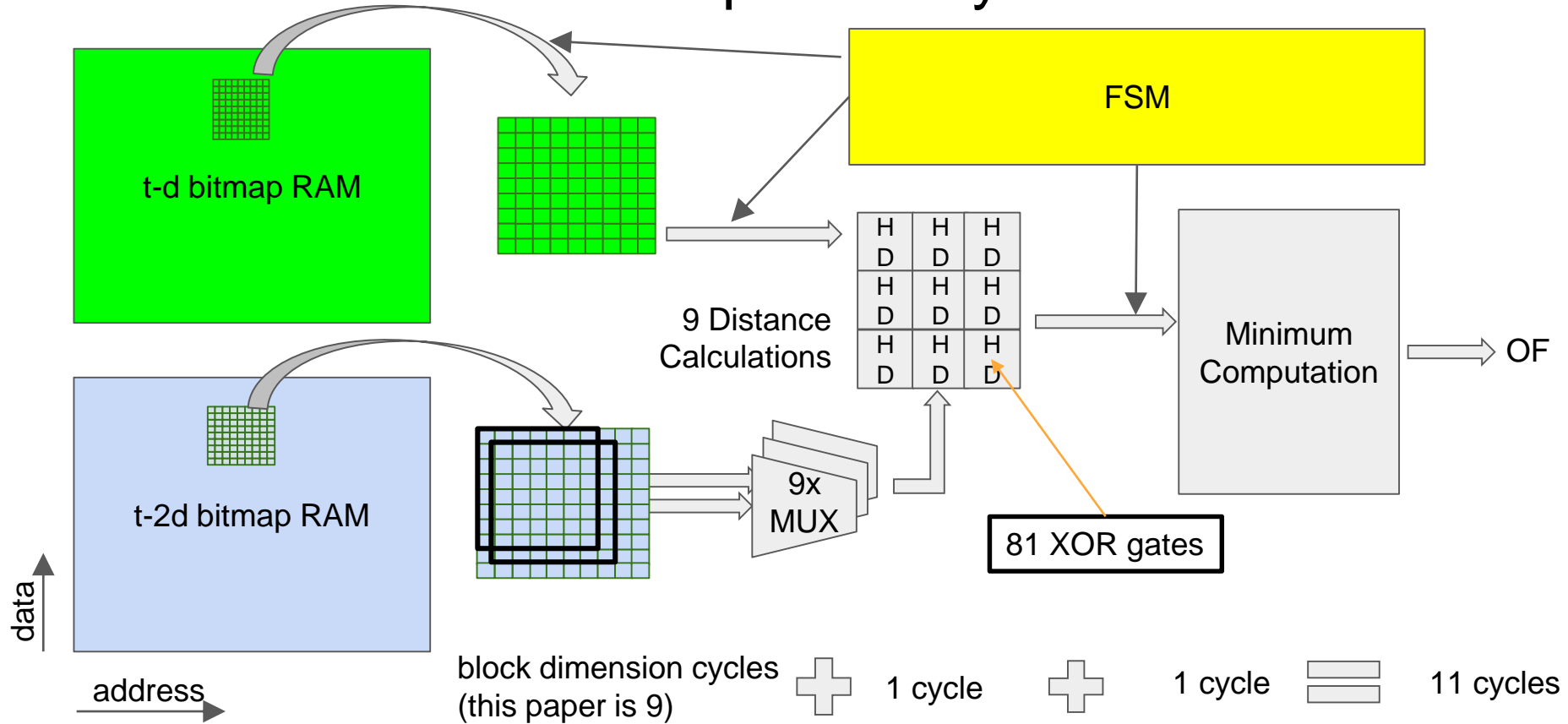
Advantages

1. Decouples the sample rate from incoming event rate
2. Incoming events drive (**optional!**) flow computation
3. Parallel block matching hardware quickly computes block distances to find best match

Development and Testing Architecture



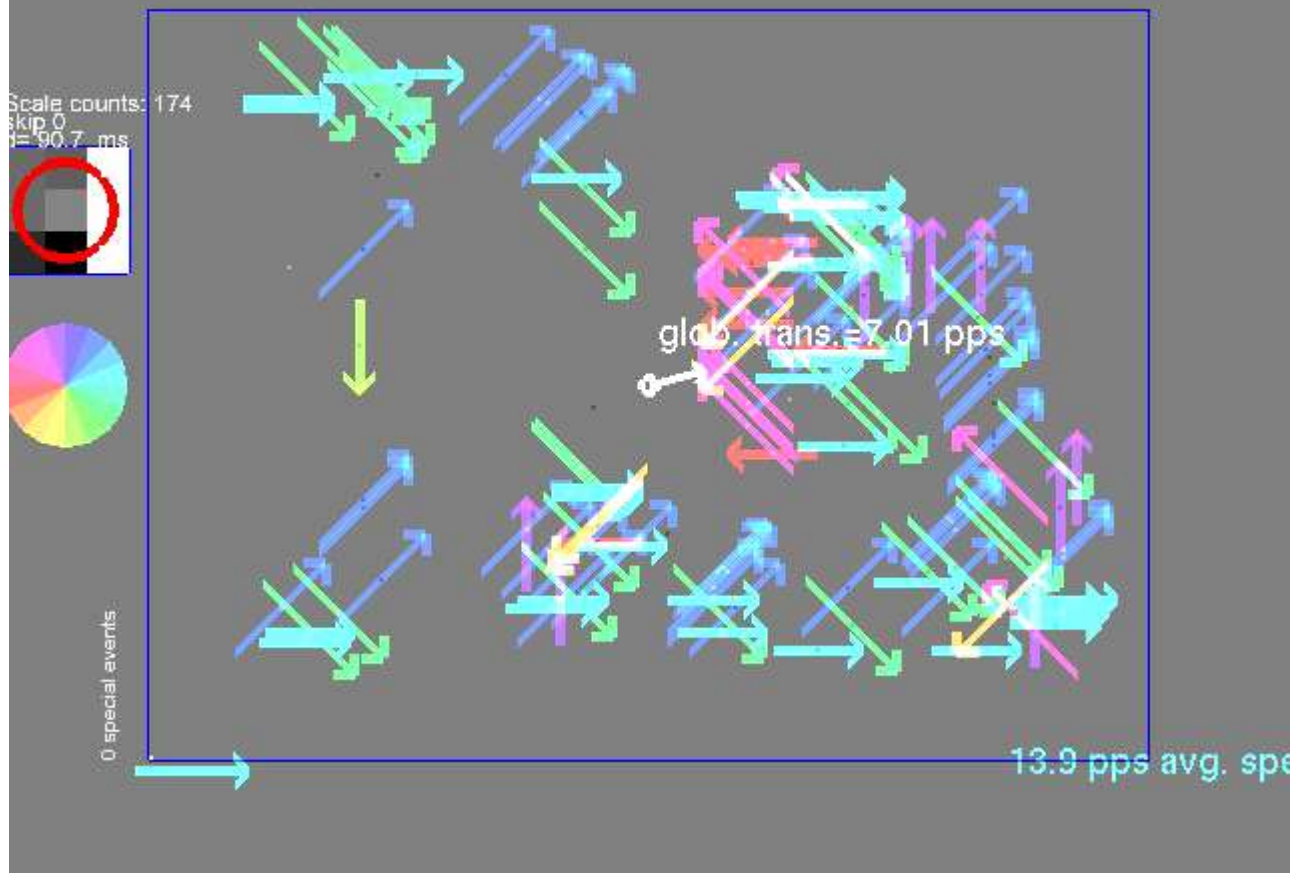
State machine and bitmap memory access



Uses <5% LUT and 200 kb RAM of Spartan 6 FPGA resources for 240x180 pixel sensor 38

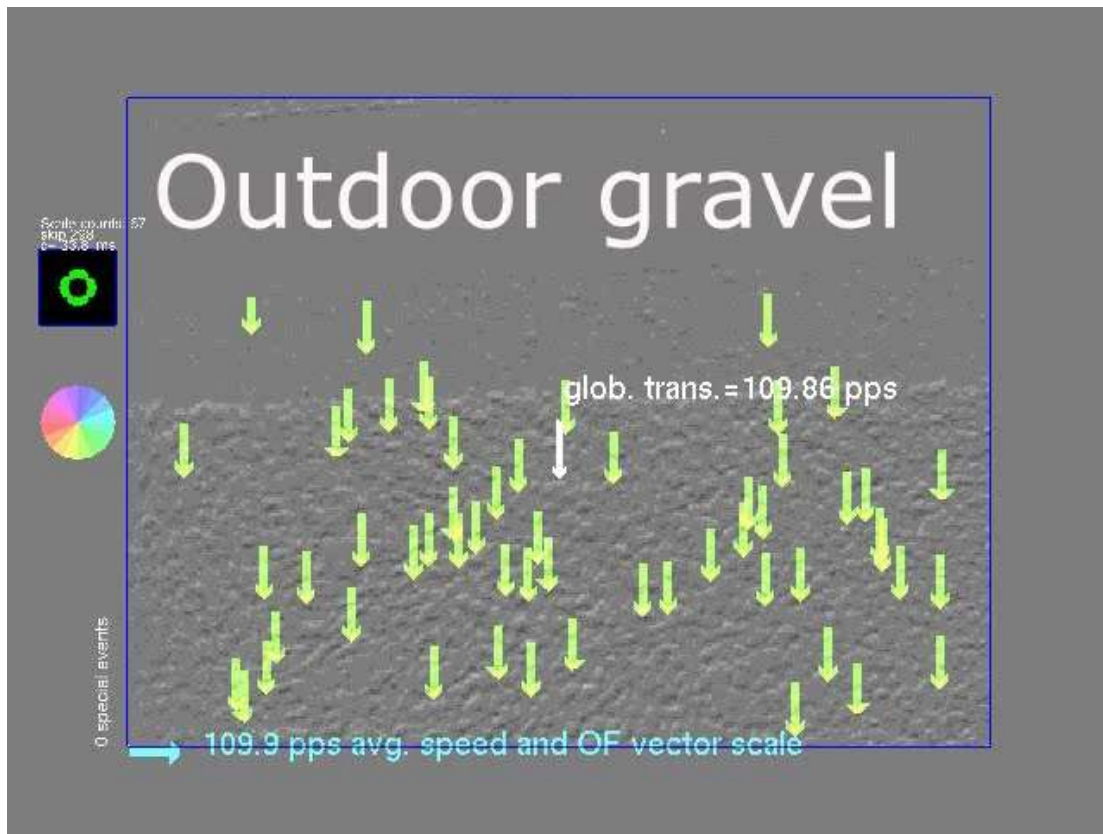
OF Results

1. Using 9x9 block size and search distance of 1 pixel
2. “Translating boxes” data from Rueckauer & Delbruck, 2016
3. Ground truth is 10 pps to right
4. $d = 100\text{ms}$
5. Average measured pps is correct
6. Many vectors not “normal flow”
7. Solved by improved search



Latest algorithm improvements (not in paper)

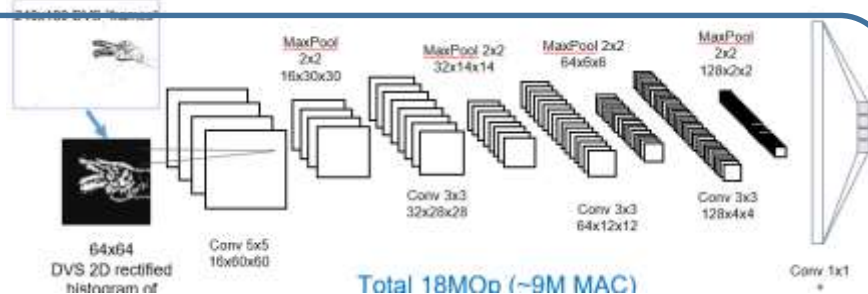
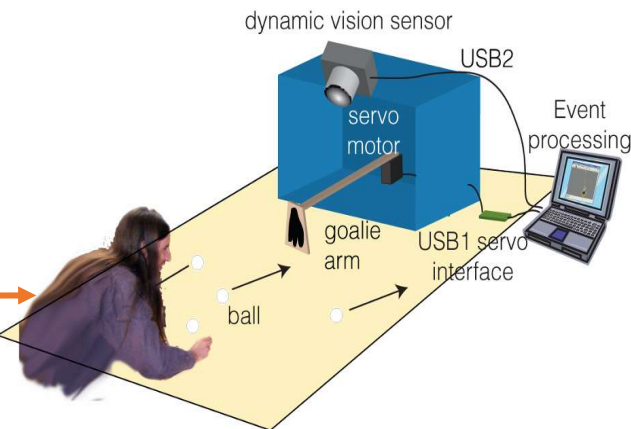
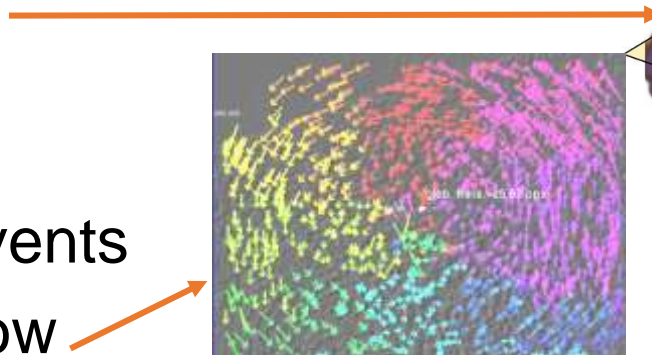
1. Using **multiscale bitmaps** cheaply matches longer distances and lower spatial frequencies.
2. Using **multiple bits/pixel** avoids bitmap saturation, e.g. 3 bits/pixel holds 8 events.
3. Using **diamond search** improves search efficiency by >20X for search distances of 12 pixels.
4. **Adapting slice duration** under feedback control achieves a target average match distance, increasing speed range and usability.



2nd generation OF algorithm

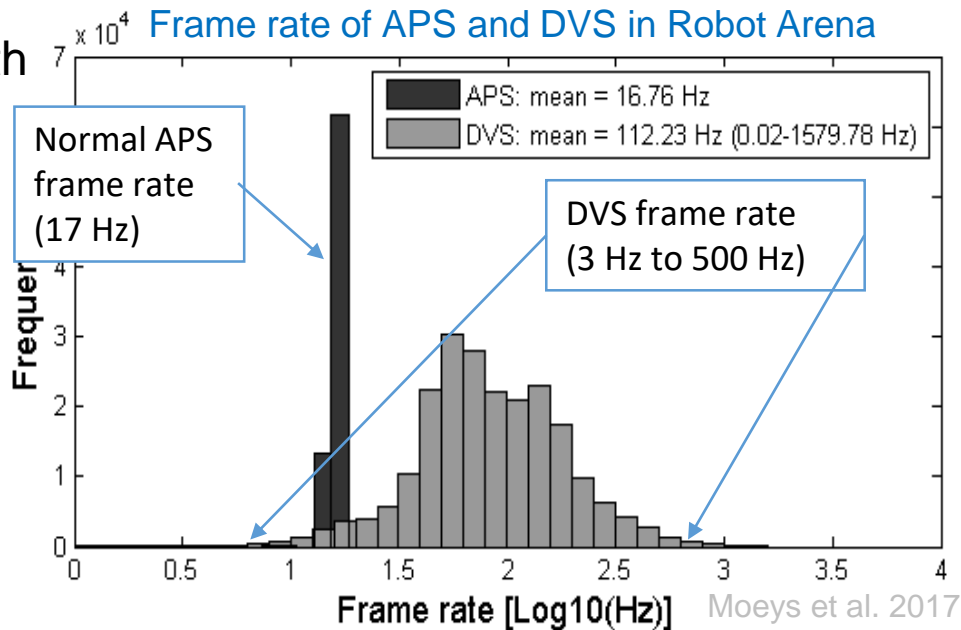
This talk has 4 parts

- Dynamic Vision Sensor Silicon Retinas
- Simple object tracking by algorithmic processing of events
- DVS Optical Flow architecture
- “Data-driven” deep inference with CNNs



Driving Convolutional Neural Networks with DVS

- Use DVS to drive conventional CNN with **constant event count** DVS frames.
- This way, **the frame rate is proportional to the rate of change of the scene.**
- The DVS local gain control and sparse output makes good input features for the CNN
- The CNNs can be trained using conventional DNN toolchains, e.g. Caffe.
- The sparse DVS frames benefit CNN accelerators that take advantage of sparsity (e.g. our NullHop).



Aimar, et al. 2016. “Nullhop: Flexibly Efficient FPGA CNN Accelerator Driven by DAVIS Neuromorphic Vision Sensor.” *NIPS 2016 Live Demonstration*.

Lungu, et al. 2017. “Live Demonstration: Convolutional Neural Network Driven by Dynamic Vision Sensor Playing RoShamBo.” In *2017 IEEE Symposium on Circuits and Systems (ISCAS 2017)*.

Moeys, D. P., et al. 2016. “Steering a Predator Robot Using a Mixed Frame/Event-Driven Convolutional Neural Network.” In *2016 IEEE Second International Conference on Event-Based Control, Communication, and Signal Processing (EBCCSP)*.

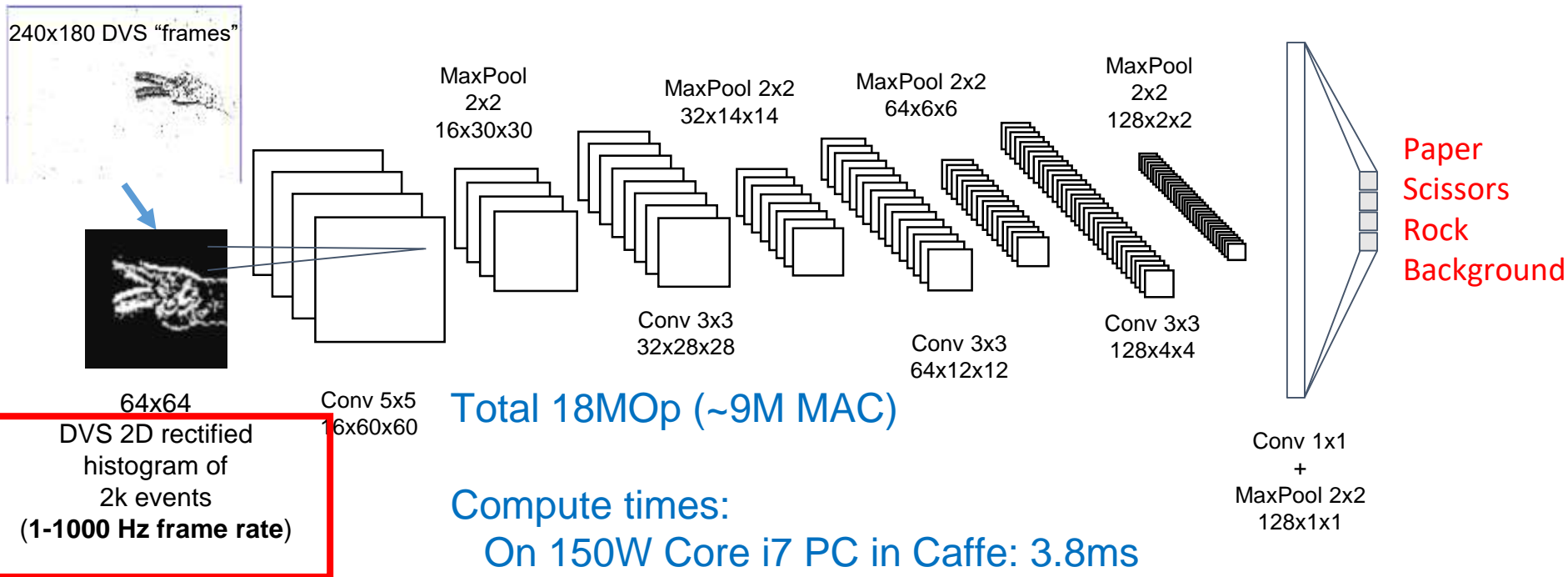
What are these people doing?

They are playing RoShambo
aka Rock-Scissors-Paper aka Janken

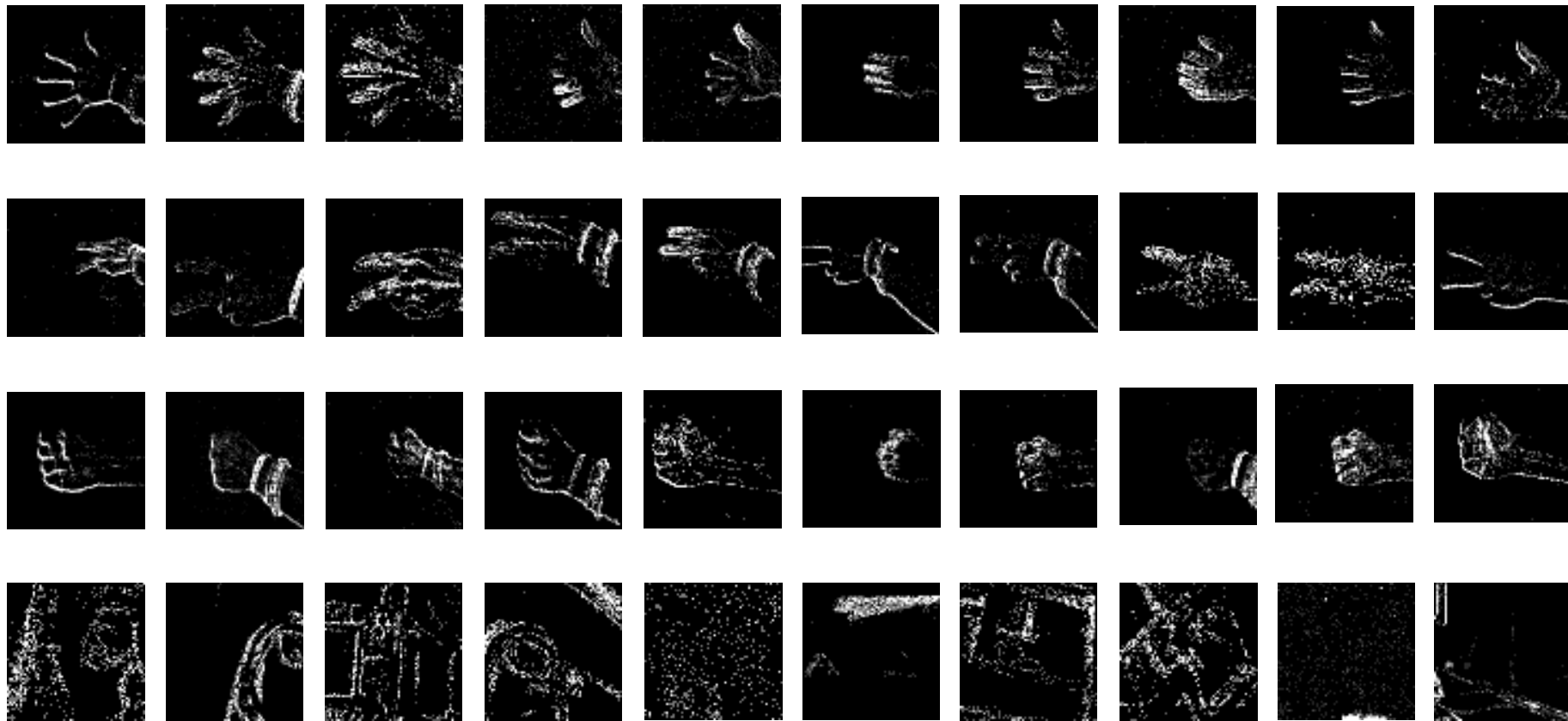


RoShamBo CNN architecture

Conventional 5-layer LeNet with ReLU/MaxPool and 1 FC layer before output.



RoShamBo training images



I.-A. Lungu, F. Corradi, and T. Delbruck, "Live Demonstration: Convolutional Neural Network Driven by Dynamic Vision Sensor Playing RoShamBo," in *2017 IEEE Symposium on Circuits and Systems (ISCAS 2017)*, Baltimore, MD, USA, 2017.



PAPER
SCISSORS

Start RoShamBo Demo

A. Aimar, E. Calabrese, H. Mostafa, A. Rios-Navarro, R. Tapiador, I.-A. Lungu, A. Jimenez-Fernandez, F. Corradi, S.-C. Liu, A. Linares-Barranco, and T. Delbruck, "Nullhop: Flexibly efficient FPGA CNN accelerator driven by DAVIS neuromorphic vision sensor," in *NIPS 2016*, Barcelona, 2016.

Conclusions

1. The DVS was developed by following a neuromorphic approach of emulating **key properties of biological retinas**
2. **Wide dynamic range** and **sparse, quick output** make these sensors useful in real time uncontrolled conditions
3. Applications: vision prosthetics, surveillance, robotics and consumer electronics
4. Precise event timing could improve learning and inference
5. Main challenges are to **reduce pixel size** and to **develop effective algorithms**. Only industry can do the first but academia has plenty of room to play for the second
6. **Event sensors can nicely drive deep inference**. There is a lot of room for improvement of deep inference power efficiency at the system level

