Neuromorphic Tracking using Event-Based Cameras

WOJCIECH ROMASZKAN

NANOCAD LABORATORY

What are Event Driven Cameras?

- Instead of using a frame clock to capture whole images, only detect changing pixels and update them using asynchronous events.
  - Temporal contrast sensors sensitive to relative illuminance change.
  - Gradient-based sensors sensitive to static edges.
  - Edge-orientation sensitive devices.
  - Optical-flow sensors.
- Why?
  - High temporal resolution (~us).
    - Monitoring brightness changes is faster than measuring exposure.
  - High dynamic range (140dB vs 60dB).
    - Photoreceptors operate on log scale, each pixel is independent (no global shutter)
  - Low power consumption.
    - No redundant data transmission/processing.
  - High pixel bandwidth (~kHz).
    - No need to wait for global exposure latency, each pixel change can be transmitted immediately.

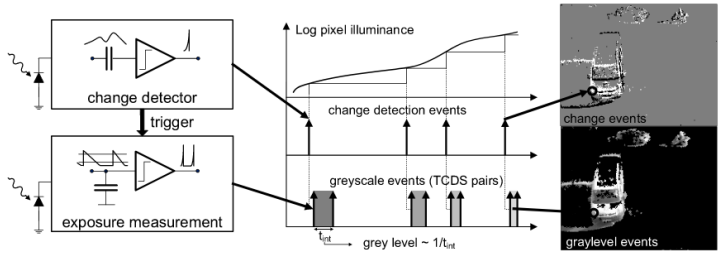
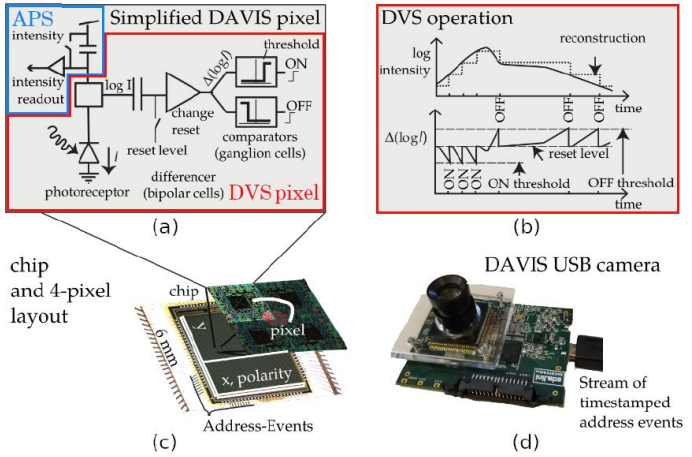


Fig. 1. Functional diagram of an ATIS pixel [42]. Two types of asynchronous events, encoding change and brightness information, are generated and transmitted individually by each pixel in the imaging array.

D. Reverter Valeiras, X. Lagorce, X. Clady, C. Bartolozzi, S. Ieng and R. Benosman, "An Asynchronous Neuromorphic Event-Driven Visual Part-Based Shape Tracking,"



Gallego, Guillermo, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger et al. "Event-based vision: A survey."

Commercial/Prototype Cameras

Table 1
Comparison of commercial or prototype event cameras. Values are approximate since there is no standard measurement testbed.

	Supplier	iniVation			Prophesee				Samsung			CelePixel		Insightness
	Camera model	DVS128	DAVIS240	DAVIS346	ATIS	Gen3 CD	Gen3 ATIS	Gen 4 CD	DVS-Gen2	DVS-Gen3	DVS-Gen4	CeleX-IV	CeleX-V	Rino 3
Sensor specifications	Year, Reference	2008 [2]	2014 [4]	2017	2011 [3]	2017 [66]	2017 [66]	2020 [67]	2017 [5]	2018 [68]	2020 [39]	2017 [69]	2019 [70]	2018 [71]
	Resolution (pixels)	128 × 128	240 × 180	346 × 260	304 × 240	640 × 480	480 × 360	1280 × 720	640 × 480	640 × 480	1280 × 960	768 × 640	1280 × 800	320 × 262
	Latency (µs)	12μs @ 1klux	12μs @ 1klux	20	3	40 - 200	40 - 200	20 - 150	65 - 410	50	150	10	8	125μs @ 10lux
	Dynamic range (dB)	120	120	120	143	> 120	> 120	> 124	90	90	100	90	120	> 100
	Min. contrast sensitivity (%)	17	11	14.3 - 22.5	13	12	12	11	9	15	20	30	10	15
	Power consumption (mW)	23	5 - 14	10 - 170	50 - 175	36 - 95	25 - 87	32 - 73	27 - 50	40	130	-	400	20-70
	Chip size (mm2)	6.3 × 6	5 × 5	8 × 6	9.9 × 8.2	9.6 × 7.2	9.6 × 7.2	6.22 × 3.5	8 × 5.8	8 × 5.8	8.4 × 7.6	15.5 × 15.8	14.3 × 11.6	5.3 × 5.3
	Pixel size (µm2)	40 × 40	18.5 × 18.5	18.5 × 18.5	30 × 30	15 × 15	20 × 20	4.86 × 4.86	9 × 9	9 × 9	4.95 × 4.95	18 × 18	9.8 × 9.8	13 × 13
	Fill factor (%)	8.1	22	22	20	25	20	> 77	11	12	22	8.5	8	22
	Supply voltage (V)	3.3	1.8 & 3.3	1.8 & 3.3	1.8 & 3.3	1.8	1.8	1.1 & 2.5	1.2 & 2.8	1.2 & 2.8		1.8 & 3.3	1.2 & 2.5	1.8 & 3.3
	Stationary noise (ev/pix/s) at 25C	0.05	0.1	0.1	-	0.1	0.1	0.1	0.03	0.03		0.15	0.2	0.1
	CMOS technology (nm)	350	180	180	180	180	180	90	90	90	65/28	180	65	180
		2P4M	1P6M MIM	1P6M MIM	1P6M	1P6M CIS	1P6M CIS	BI CIS	1P5M BSI			1P6M CIS	CIS	1P6M CIS
	Grayscale output	no	yes	yes	yes	no	yes	no	no	no	no	yes	yes	yes
	Grayscale dynamic range (dB)	NA	55	56.7	130	NA	> 100	NA	NA	NA	NA	90	120	50
	Max. frame rate (fps)	NA	35	40	NA	NA	NA	NA	NA	NA	NA	50	100	30
Camera	Max. Bandwidth (Meps)	1	12	12	-	66	66	1066	300	600	1200	200	140	20
	Interface	USB 2	USB 2	USB 3		USB 3	USB 3	USB 3	USB 2	USB 3	USB 3			USB 2
	IMU output	no	1 kHz	1 kHz	no	1 kHz	1 kHz	no	no	1 kHz	no	no	no	1 kHz

Gallego, Guillermo, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger et al. "Event-based vision: A survey."

- Trends:
  - Higher spatial resolution.
  - Higher readout speed.
  - Gray level output.
  - Intertial measurement unit.
  - Multi-camera time-stamps.
- Recent Trends:
  - Smaller pixel size.

"Event camera pixel size has shrunk pretty closely following feature size scaling, which is remarkable considering that a DVS pixel is a mixed-signal circuit, which generally do not scale following technology. However, achieving even smaller pixels is difficult and may require abandoning the strictly asynchronous circuit design philosophy that the cameras started with. Camera cost is constrained by die size (since silicon costs about \$5-\$10=cm_2 in mass production), and optics (designing new mass production miniaturized optics to fit a different sensor format can cost tens of millions of dollars)."

Challenges

- Fundamentally different output format than conventional cameras.
  - Asynchronous.
  - Sparse.
  - Requires new algorithms to deal with that.
- Different photometric sensing.
  - Binary information (increase/decrease) instead of grayscale values.
- Noise and dynamic effects.
  - Photon + transistor circuit noise is a problem for all cameras.
  - For event driven ones, quantizing temporal contrast is complex and not fully characterized.

Event Processing

- Representation
  - Individual events
    - Probabilistic filters, spiking neural networks.
    - Require retaining past knowledge.
  - Event packet
  - Event frame/2D histogram/edge maps
    - Familiar representation.
    - Looses sparsity and timestamp information.
  - Also: time surface, voxel grid, 3D point set, motion compensated images, reconstructed images

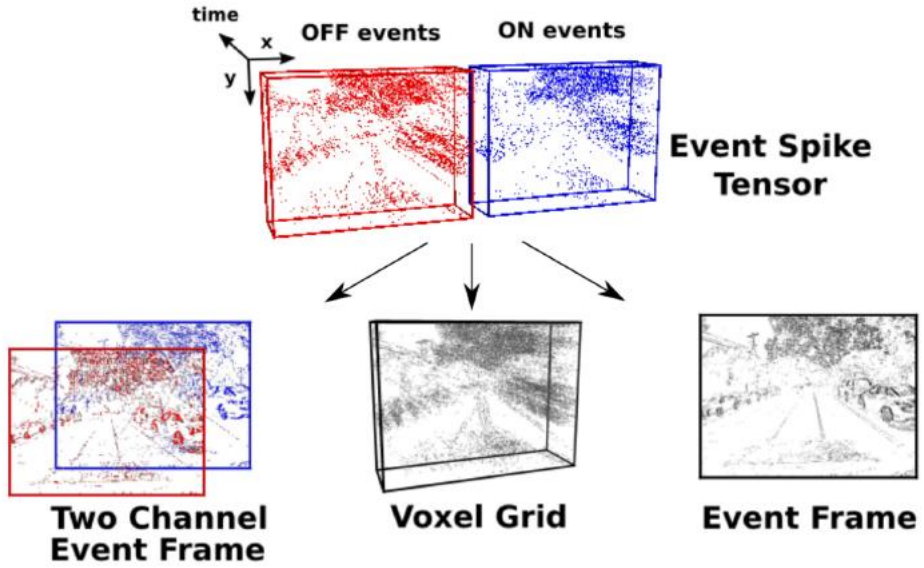


Figure 3. Different ways to convert events into more familiar representations, suitable for processing with modern learning architectures [111].

Event Processing

- Event-by-event
  - Filters
    - Deterministic filters (e.g. convolutions) for noise reduction, feature extraction, image reconstruction, brightness filtering.
    - Probabilistic filters (Bayesian methods) for tracking.
    - Require additional information (past events or grayscale map).
  - ANNs
    - Unsupervised learning is used to design feature extractors.
    - If there's enough labeled data, supervised learning can be used.
      - Training is generally done on packets of events.
      - Trained network can be converted to a SNN.
    - Used for object/action classification.
- Groups of events
  - Mainly pre-processing for classical computer vision tools.

How to Track Objects Based on Events?

- Gaussian blob trackers
  - Defined by mean (x,y) and covariance matrix.
- Detected events are assigned to a tracker based on highest probability:

$$p^i(\boldsymbol{u}) = \frac{1}{2\pi}|\Sigma^i|^{-\frac{1}{2}} e^{-\frac{1}{2}(\boldsymbol{u}-\boldsymbol{\mu}^i)^T(\Sigma^i)^{-1}(\boldsymbol{u}-\boldsymbol{\mu}^i)}$$

- Tracker is then updated:

$$\mu(t) = \alpha_1 \mu(t-\Delta t) + (1-\alpha_1)\mathbf{u}$$
$$\Sigma(t) = \alpha_2 \Sigma(t-\Delta t) + (1-\alpha_2)\Delta\Sigma$$

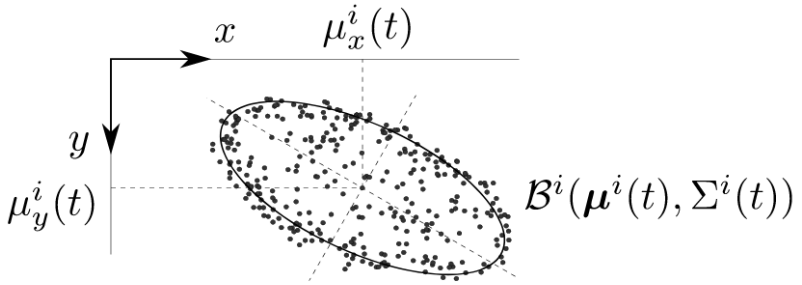


Fig. 2. Gaussian tracker $\mathcal{B}^i$ following a cloud of events is defined by its location $\mu^i(t) = [\mu_x^i(t), \mu_y^i(t)]^T$ and covariance matrix $\Sigma^i(t)$.

D. Reverter Valeiras, X. Lagorce, X. Clady, C. Bartolozzi, S. Ieng and R. Benosman, "An Asynchronous Neuromorphic Event-Driven Visual Part-Based Shape Tracking,"

Grouping Trackers Together

- More complex objects can be tracked by modeling a system of trackers connected by springs.
- Displacement, energy etc can be calculated from the Hooke's law and Newton's second law.
- This system can deal with occlusions and distortions.

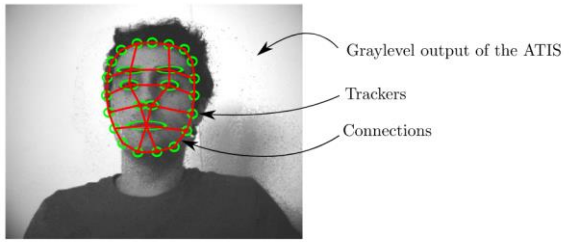


Fig. 18. Set of trackers and the structure of their connections used to follow a face from incoming events. Ellipses: position of the trackers. Lines: connections set between the trackers. Each connection is a combination of a Euclidean connection and a torsional connection, with $\alpha = 0.02$ for both the connections.

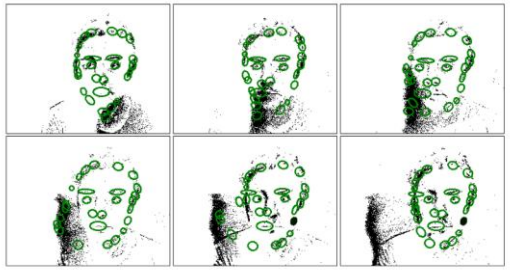
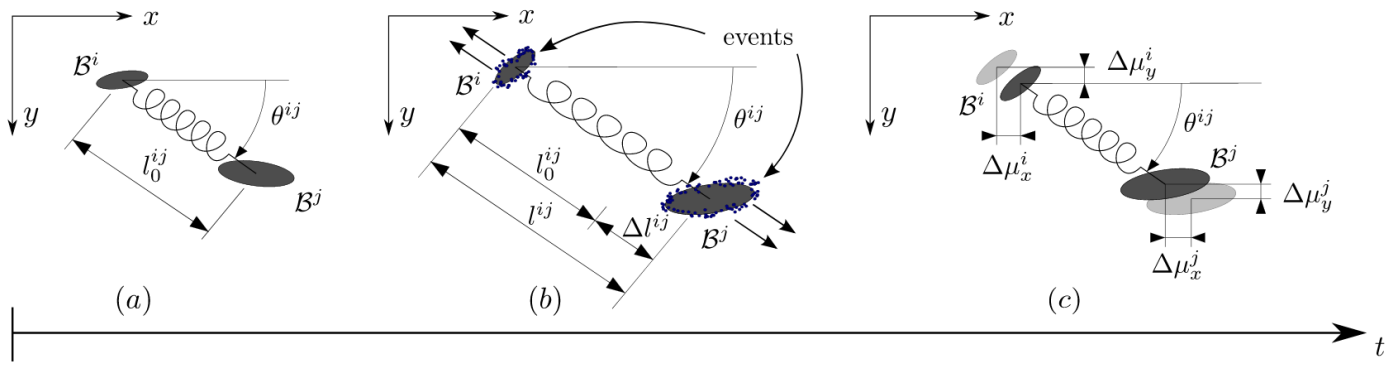


Fig. 19. Set of connected trackers is disturbed by a dynamic occlusion introduced by waving a hand in front of the face. As the hand passes in front of the face, it first attracts the trackers, displacing them from their right position. However, the system is sufficiently robust to compensate by attracting the trackers to the right position again, without losing track of the face.



D. Reverter Valeiras, X. Lagorce, X. Clady, C. Bartolozzi, S. Ieng and R. Benosman, "An Asynchronous Neuromorphic Event-Driven Visual Part-Based Shape Tracking,"

Convolutional Trackers - CAVIAR

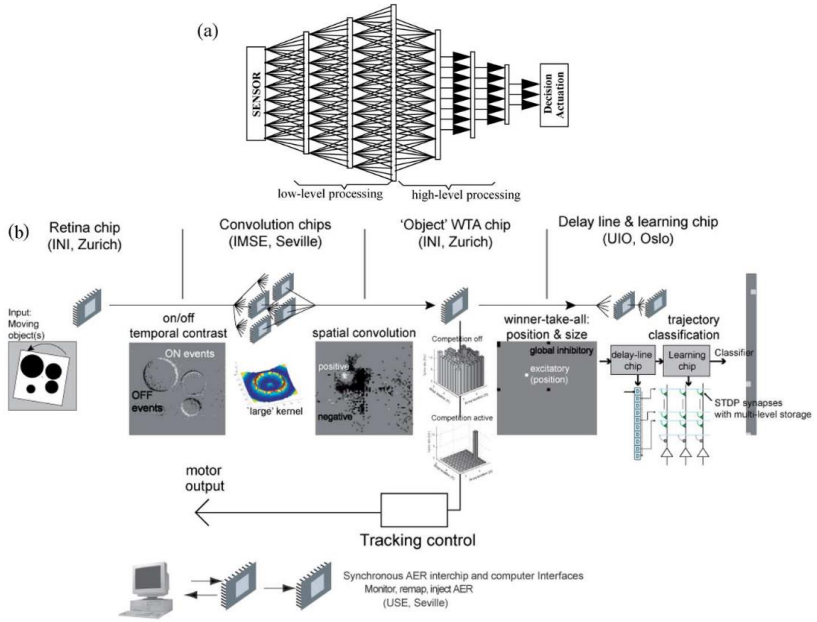


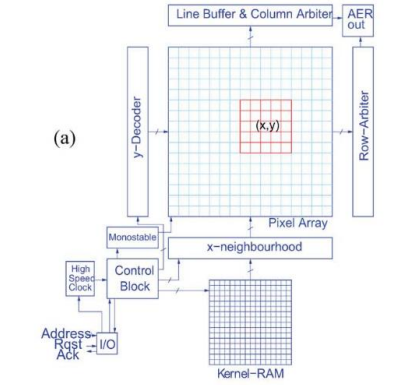
Fig. 1. CAVIAR system overview. (a) A bioinspired system architecture performing feedforward sensing + processing + actuation tends to have the following conceptual hierarchical structure: 1) a sensing layer; 2) a set of low-level processing layers usually implemented through projection fields (convolutions) for feature extraction and combination; 3) a set of high level processing layers that operate on "abstractions" and progressively compress information through, for example, dimension reduction, competition, and learning; 4) once a reduced set of signals/decisions is obtained they are conveyed to (usually mechanical) actuators. (b) The CAVIAR system components and multilayer architecture; an example output of each component is shown in response to the rotating stimulus and the basic functionality is illustrated below each chip component.

R. Serrano-Gotarredona *et al.*, "CAVIAR: A 45k Neuron, 5M Synapse, 12G Connects/s AER Hardware Sensory–Processing– Learning–Actuating System for High-Speed Visual Object Recognition and Tracking,"

TABLE I
TEMPORAL CONTRAST VISION SENSOR PROPERTIES ADAPTED FROM [20]

Functionality	Asynchronous temporal contrast
Pixel size (μm)	40x40
Chip size (mm)	6x6
Fill factor	9.4%
Fabrication process	4M 2P 0.35μm CMOS
Pixel complexity	26 transistors (14 analog), 3 capacitors
Array size	128x128
Interface	15-bit non-greedy AER
Power consumption	24mW
Dynamic range	120dB, 2 lux to >100klux scene illumination with f/1.2 lens. Moonlight capable with high contrast scene
Response latency	15μs @ 700mW/m^2
Max events/sec	~2 Meps
Standard deviation σ of temporal contrast threshold	2.1% scene contrast

CAVIAR ctd.



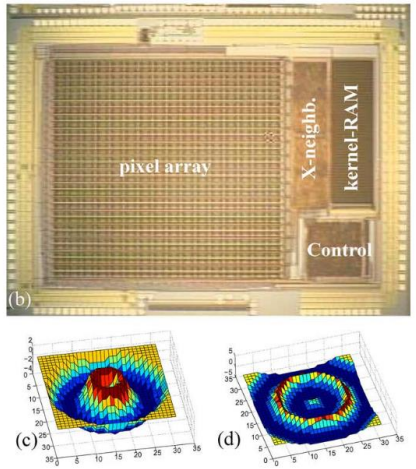


Fig. 3. Convolution chip. (a) Architecture of the convolution chip. (b) Microphotograph of fabricated chip. (c) Kernel for detecting circumferences of radius close to four pixels and (d) close to nine pixels.

R. Serrano-Gotarredona *et al.*, "CAVIAR: A 45k Neuron, 5M Synapse, 12G Connects/s AER Hardware Sensory–Processing– Learning–Actuating System for High-Speed Visual Object Recognition and Tracking,"

TABLE II
CONVOLUTION CHIP PROPERTIES

Functionality	Dynamic 2D Convolution with programmable kernel
Pixel size (µm)	90x90
Chip size (mm)	4.0x5.4
Fabrication process	4M 2P 0.35µm CMOS
Pixel complexity	364 transistors, 1 capacitor
Array size	32x32
max kernel size	31x31
kernel weight resolution	4 bit
calibration resolution	5 bit
Interface	15-bit word parallel non-greedy AER
Power consumption	66-150mW, depending on kernel size
forgetting rate	adjustable
Max out events/sec	25 Meps

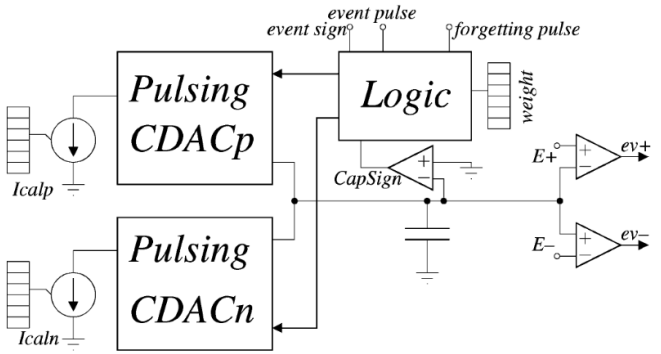


Fig. 4. Simplified block diagram of convolution chip pixel.

Convolutional Trackers ctd.

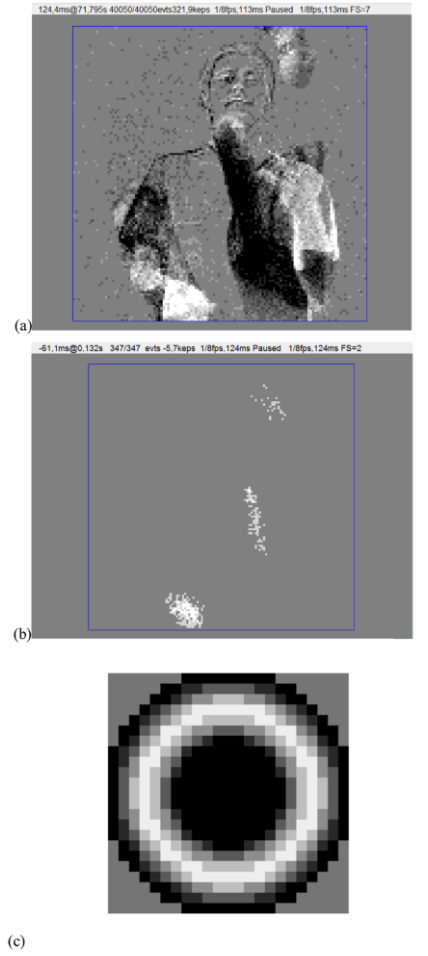
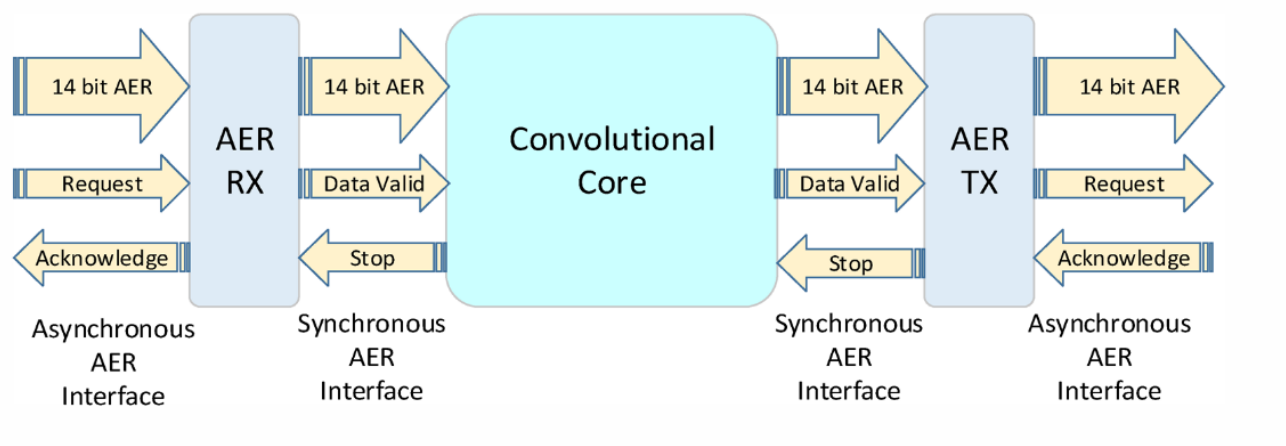


Fig. 2. Event-Driven 2D Convolution Processing. (a) 124ms histogram from a DVS output. (b) Ouput of reported 2D convolution processing core programmed with the kernel shown in (c).



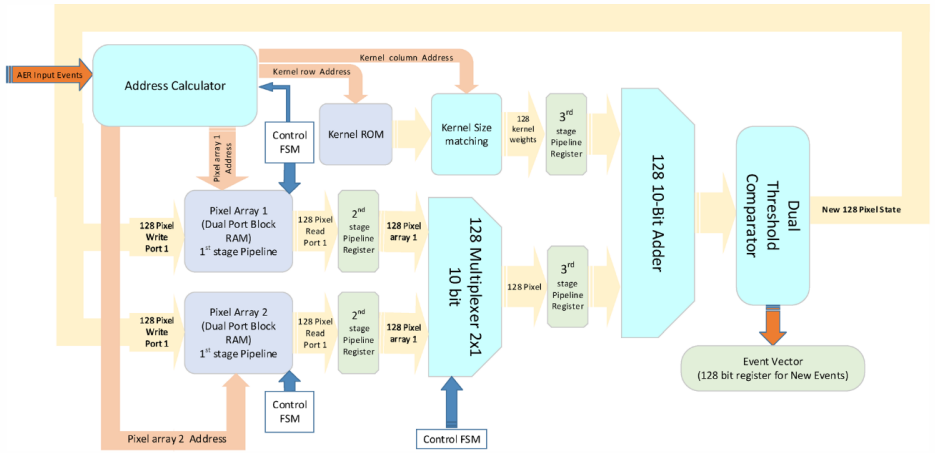


Fig. 5. Data flow diagram of the input event processing block

Yousefzadeh, Amirreza & Serrano-Gotarredona, Teresa & Linares-Barranco, Bernabé. (2015). Fast Pipeline 128×128 pixel spiking convolution core for event-driven vision processing in FPGAs.

Other Tracking Methods

- Iterative Closest Point (ICP)
- Gradient Descent
- Mean-shift
- Monte-Carlo methods
- Particle filtering