Literature Review on Event Cameras

Jianhao JIAO

The Hong Kong University of Science and Technology

香港科技大學
THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Ri 香港科大機器人研究所
ROBOTICS INSTITUTE | HKUST

November 20, 2020

Outline

1. Challenges on Traditional Computer Vision
2. Introduction to Event-based Cameras
3. Some Important Tasks , Solutions, and Opportunities
4. Recap and Resources

Open Challenges in Computer Vision

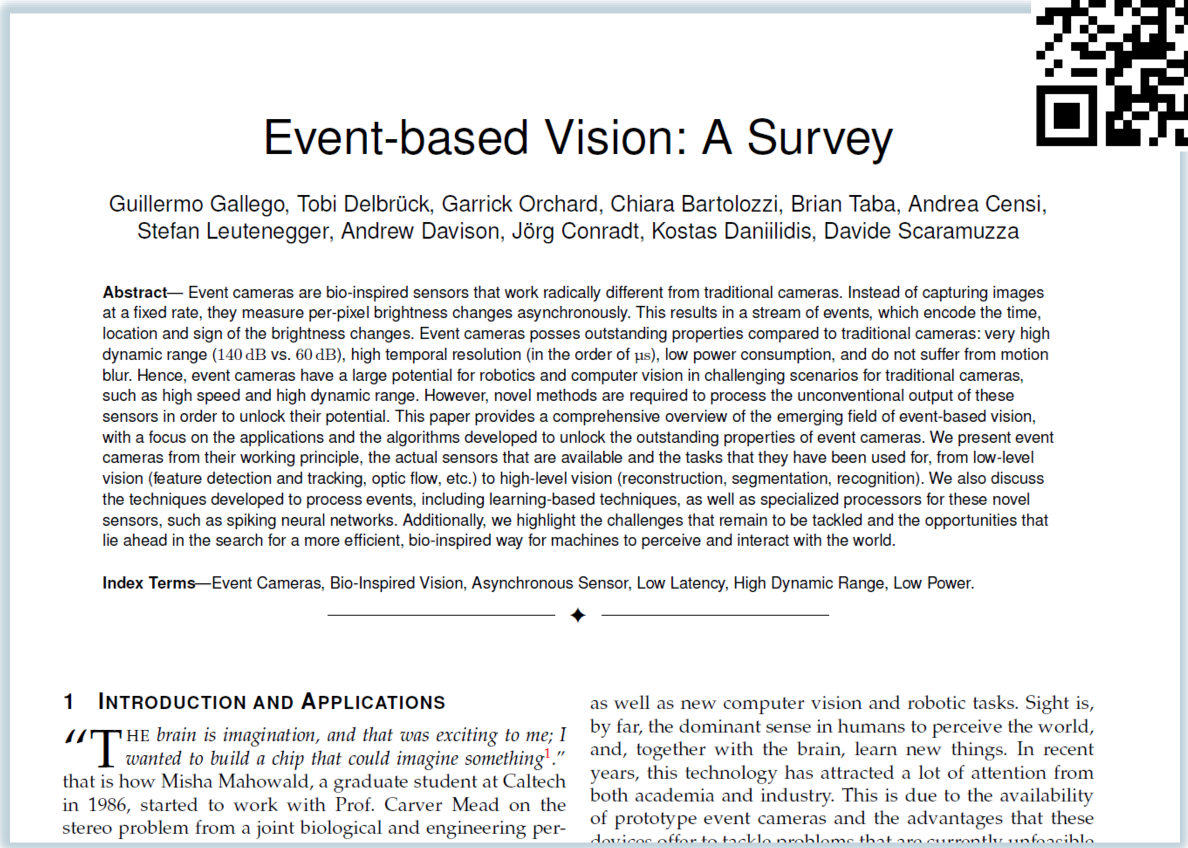
The past 60 years of research have been devoted to frame-based cameras …but they are not good enough!

Latency & Motion blur

Dynamic Range

Event cameras do not suffer from these problems!

Reference: T-PAMI 2020 paper



Event-based Vision: A Survey

Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew Davison, Jörg Conradt, Kostas Daniilidis, Davide Scaramuzza

Abstract— Event cameras are bio-inspired sensors that work radically different from traditional cameras. Instead of capturing images at a fixed rate, they measure per-pixel brightness changes asynchronously. This results in a stream of events, which encode the time, location and sign of the brightness changes. Event cameras posses outstanding properties compared to traditional cameras: very high dynamic range (140 dB vs. 60 dB), high temporal resolution (in the order of µs), low power consumption, and do not suffer from motion blur. Hence, event cameras have a large potential for robotics and computer vision in challenging scenarios for traditional cameras, such as high speed and high dynamic range. However, novel methods are required to process the unconventional output of these sensors in order to unlock their potential. This paper provides a comprehensive overview of the emerging field of event-based vision, with a focus on the applications and the algorithms developed to unlock the outstanding properties of event cameras. We present event cameras from their working principle, the actual sensors that are available and the tasks that they have been used for, from low-level vision (feature detection and tracking, optic flow, etc.) to high-level vision (reconstruction, segmentation, recognition). We also discuss the techniques developed to process events, including learning-based techniques, as well as specialized processors for these novel sensors, such as spiking neural networks. Additionally, we highlight the challenges that remain to be tackled and the opportunities that lie ahead in the search for a more efficient, bio-inspired way for machines to perceive and interact with the world.

Index Terms—Event Cameras, Bio-Inspired Vision, Asynchronous Sensor, Low Latency, High Dynamic Range, Low Power.

1 Introduction and Applications

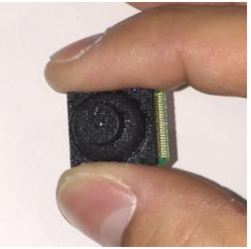
"*THE brain is imagination, and that was exciting to me; I wanted to build a chip that could imagine something*[1]." that is how Misha Mahowald, a graduate student at Caltech in 1986, started to work with Prof. Carver Mead on the stereo problem from a joint biological and engineering perspective, as well as new computer vision and robotic tasks. Sight is, by far, the dominant sense in humans to perceive the world, and, together with the brain, learn new things. In recent years, this technology has attracted a lot of attention from both academia and industry. This is due to the availability of prototype event cameras and the advantages that these devices offer to tackle problems that are currently unfeasible

http://rpg.ifi.uzh.ch/docs/EventVisionSurvey.pdf

What is an event camera?

Novel sensor that measures only **motion in the scene**

First commercialized in 2008 by T. Delbruck (UZH&ETH) under the name of Dynamic Vision Sensor (DVS)



Mini DVS sensor from IniVation.com

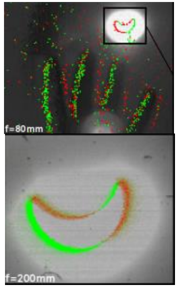


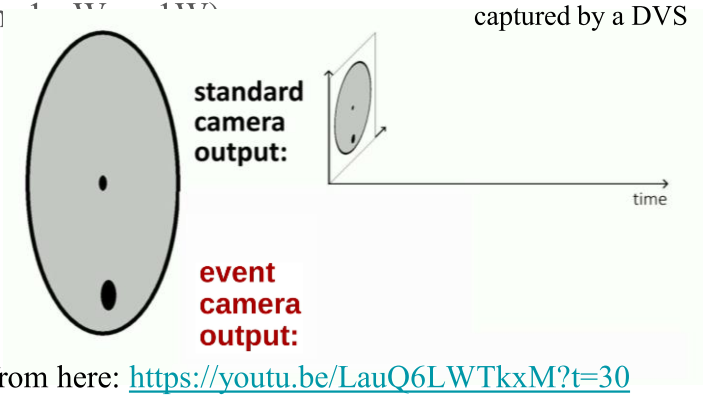
Image of the solar eclipse captured by a DVS

Advantages:

- High temporal resolution (1 MHz clock)
- Low-latency (~ 1 μs) -> No motion blur
- High dynamic range (140 dB instead of 60 dB)
- Ultra-low power (implicit feature extraction, mean

Traditional vision algorithms cannot be used because:

- **Asynchronous** pixels
- **No intensity information** (only binary intensity changes)

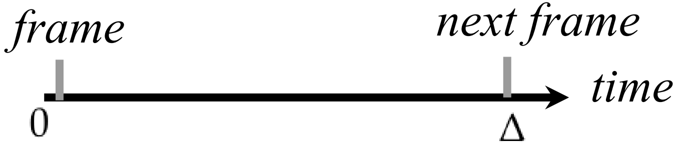


Video from here: https://youtu.be/LauQ6LWTkxM?t=30

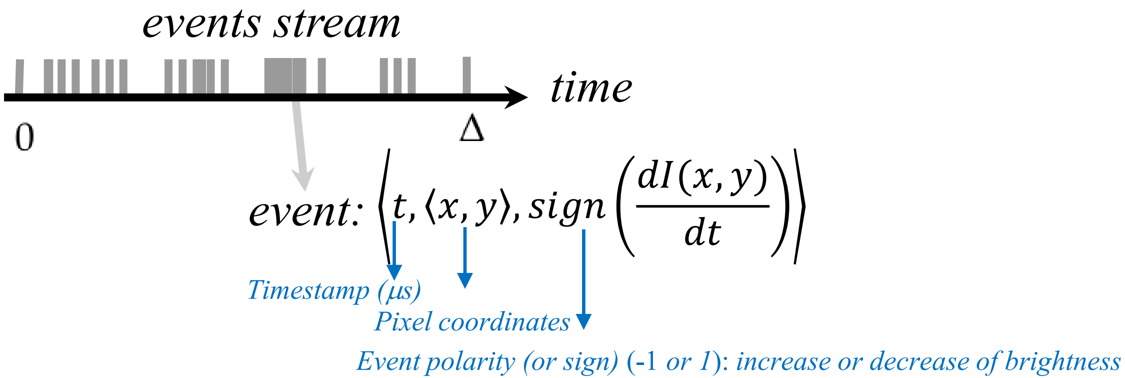
[Lichtsteiner, Posch, Delbruck, A 128x128 120 dB 15μs Latency Asynchronous Temporal Contrast Vision Sensor, 2008

Camera vs Event Camera

- A **traditional camera** outputs frames at **fixed time intervals**:



- By contrast, a **DVS** outputs **asynchronous events** at ***microsecond*** **resolution**. An event is generated each time a single pixel detects an intensity changes value



$$event\text{:}\ \left\langle t, \langle x, y\rangle, sign\left(\frac{dI(x,y)}{dt}\right)\right\rangle$$

Timestamp (μs)

Pixel coordinates

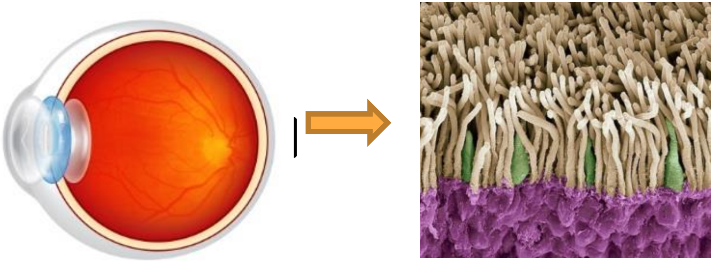
Event polarity (or sign) (-1 *or 1*): *increase or decrease of brightness*

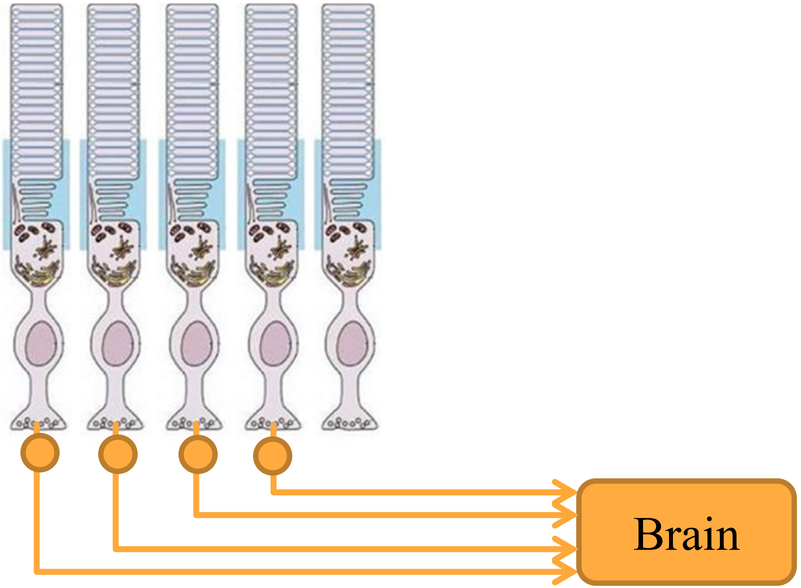
[Lichtsteiner, Posch, Delbruck, A 128x128 120 dB 15µs Latency Asynchronous Temporal Contrast Vision Sensor, 2008

Event cameras are inspired by the Human Eye

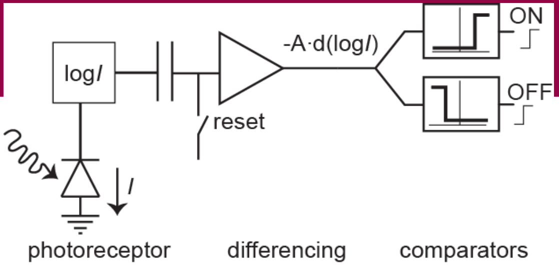
Human retina:

- 130 million **photoreceptors**
- But only 2 million **axons**!



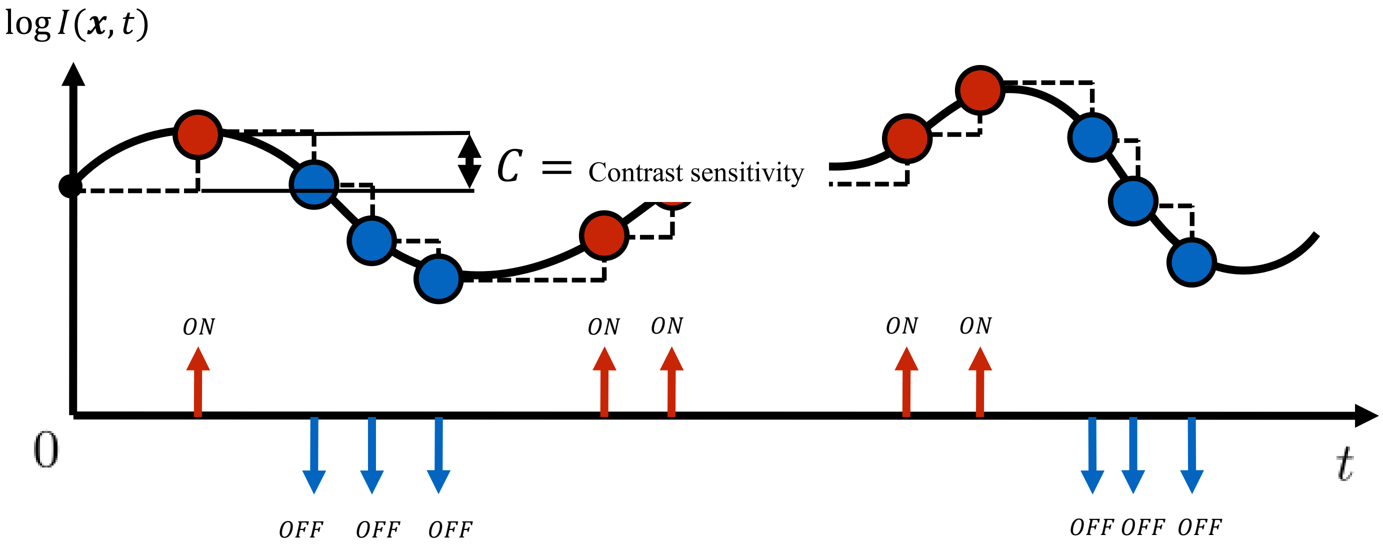


Generative Event Model



Consider the intensity at a **single pixel…**

$$\pm C = \log I(\boldsymbol{x}, t) - \log I(\boldsymbol{x}, t - \Delta t)$$



Events are triggered **asynchronously**

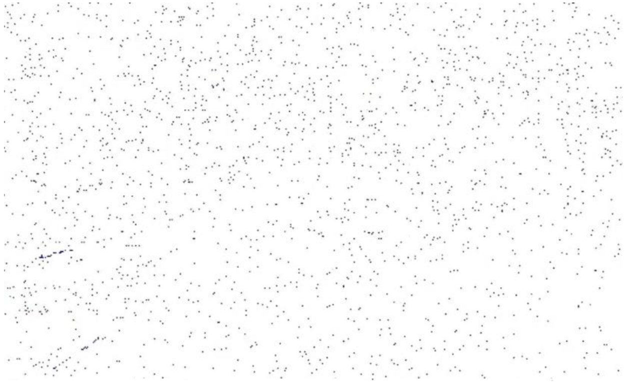
Lichtsteiner et al., A 128x128 120 dB 15μs Latency Asynchronous Temporal Contrast Vision Sensor, IEEE Journal of Solid-State Circuits, 2008

Event Camera Output with No Motion

Without motion, only background noise is output

Standard Camera

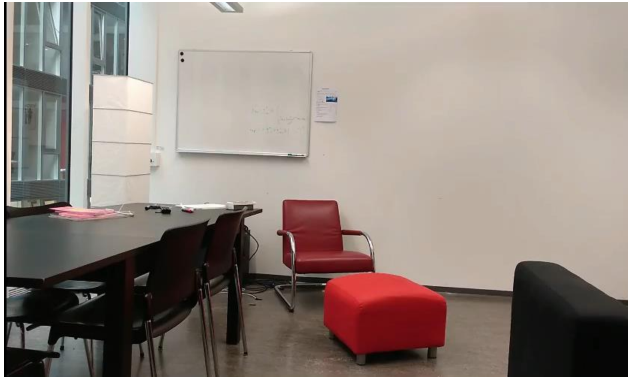
Ǝvent Camera (ON, OFF events)



ΔT = 40 ms

Event Camera Output with Relative Motion

Standard Camera

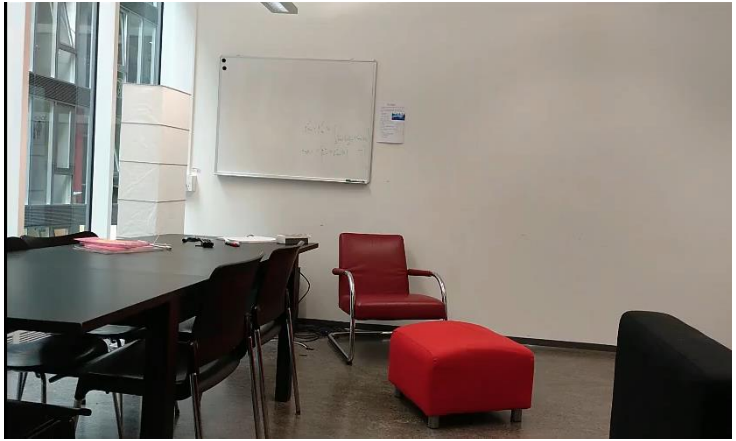


Event Camera (ON, OFF events)

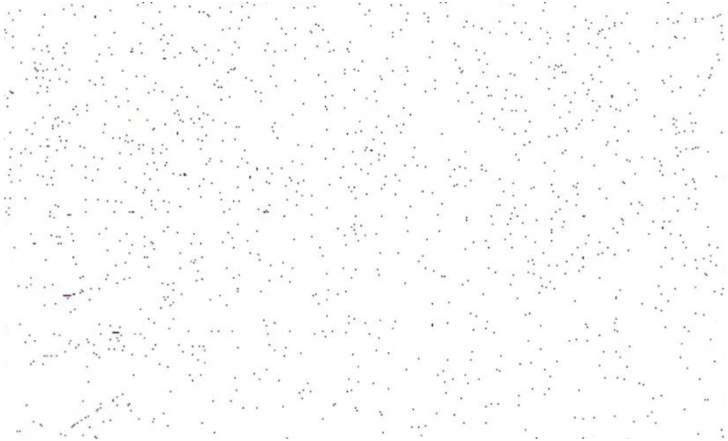
$\Delta T = 10$ ms

Event Camera Output with Relative Motion

Standard Camera

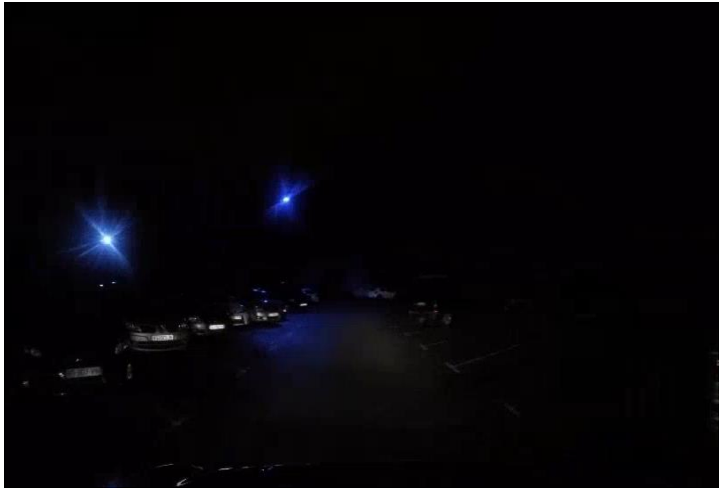


Event Camera (ON, OFF events)

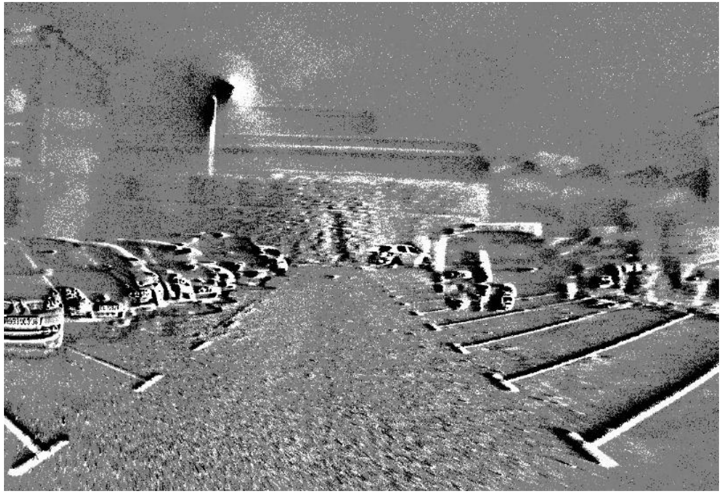


ΔT = 40 ms

An example: Low-light Sensitivity (night drive)



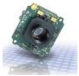
GoPro Hero 6

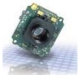


Event Camera by Prophesee
White = Positive events
Black = Negative events

Video courtesy of Prophesee: https://www.prophesee.ai

High-speed vs Event Cameras





	High speed camera	Standard camera	Event Camera
Max fps or measurement rate	**Up to 1MHz**	100-1,000 fps	**1MHz**
Resolution at max fps	64x16 pixels	**>1Mpxl**	**>1Mpxl**
Bits per pixels (event)	12 bits	8-10 per pixel	~**40 bits/event** {t,(x,y),p)}
Weight	6.2 Kg	**30 g**	**30 g**
Active cooling	yes	**No cooling**	**No cooling**
Data rate	1.5 GB/s	32MB/s	**~1MB/s on average (depends on dynamics)**
Mean power consumption	150 W + external light	1 W	**1 mW**
Dynamic range	n.a.	60 dB	**140 dB**

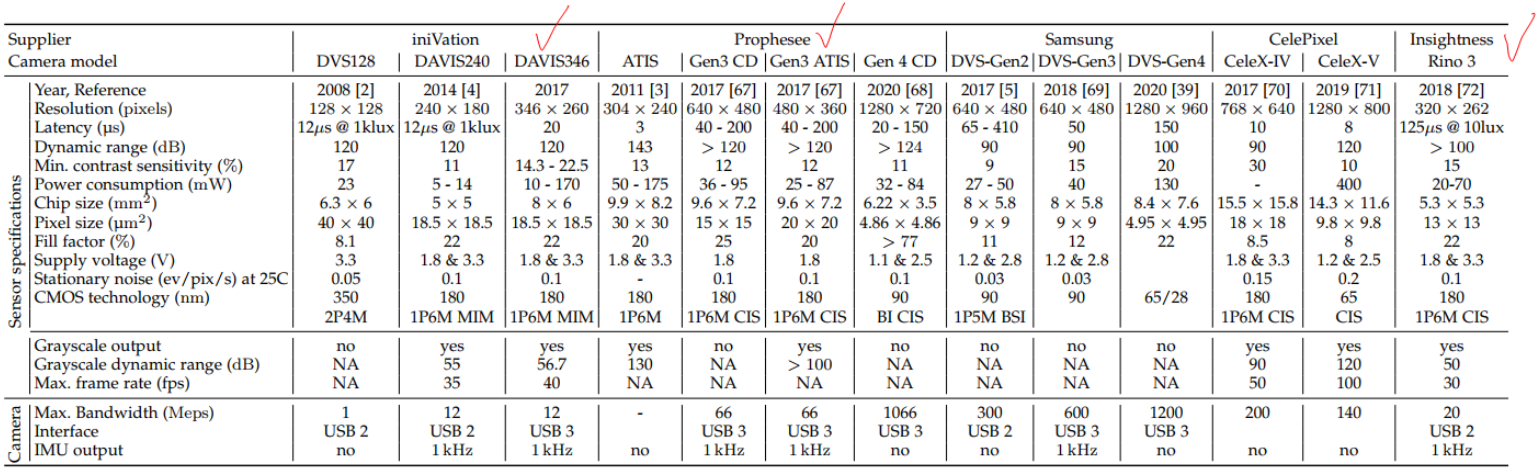
Current commercial applications of event cameras

- **Internet of Things (IoT)**
    1. Low-power, always-on devices for monitoring and surveillance
- **Automotive**:
    1. low-latency, high dynamic range (HDR) object detection
    2. low-power training & inference
    3. low-memory storage
- **AR/VR**
    1. low-latency, low-power tracking
- **Industrial automation**
    1. Fast pick and place

Who sells event cameras and how much are they?

- Inivation:
  1. **DAVIS sensor**: **frames, events, IMU**.
  2. Resolution**:** ~QVGA (346x260 pixels)
  3. **Cost: 6,000 USD**
- Insightness:
  1. **RINO sensor**: frames, events, IMU.
  2. **Resolution:** ~QVGA (320x262 pixels)
  3. **Cost: 6,000 USD**
- Prophesee:
  1. **ATIS sensor:** events, IMU, absolute intensity at the event pixel
  2. Resolution: 1M pixels
  3. **Cost: 4,000 USD**
- CelexPixel Technology:
  1. **Celex One:** events, IMU, absolute intensity at the event pixel
  2. Resolution: 1M pixels
  3. **Cost: 1,000 USD**.
- **Samsung Electronics**
  1. Samsung DVS: events, IMU
  2. Resolution: up to 1Mpxl
  3. **Cost: not listed**

Comparison of current event cameras



	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 [67]	2017 [67]	2020 [68]	2017 [5]	2018 [69]	2020 [39]	2017 [70]	2019 [71]	2018 [72]
	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 - 84	27 - 50	40	130	-	400	20-70
	Chip size (mm^2)	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 (μm^2)	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

Table from [Guillermo et al., T-PAMI’20], Table 1

[95] P. Lichtsteiner, C. Posch, and T. Delbruck. “A 128×128 120 dB 15 μs latency asynchronous temporal contrast vision sensor”. In: *IEEE J. Solid-State Circuits,* 2008, http://dx.doi.org/10.1109/JSSC.2007.914337

[135] C. Posch, D. Matolin, and R. Wohlgenannt. “A QVGA 143 dB Dynamic Range Frame-Free PWM Image Sensor With Lossless Pixel-Level Video Compression and Time-Domain CDS”. In: *IEEE J. Solid-State Circuits* 46.1, 2011, http://dx.doi.org/10.1109/JSSC.2010.2085952

[16] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck. “A 240x180 130dB 3us Latency Global Shutter Spatiotemporal Vision Sensor”. In: *IEEE J. Solid-State Circuits* 49.10 (2014), pp. 2333–2341. http://dx.doi.org/10.1109/JSSC.2014.2342715

[31] https://inivation.com/wp-content/uploads/2019/07/2019-07-09-DVS-Specifications.pdf

[138] https://www.prophesee.ai/event-based-evk/

[166] http://rpg.ifi.uzh.ch/docs/CVPR19workshop/CVPRW19_Eric_Ryu_Samsung.pdf

[23] http://rpg.ifi.uzh.ch/docs/CVPR19workshop/CVPRW19_CelePixel.pdf

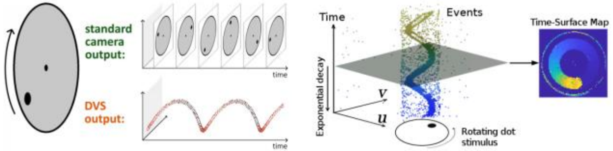
Event Representation: how to deal with events?

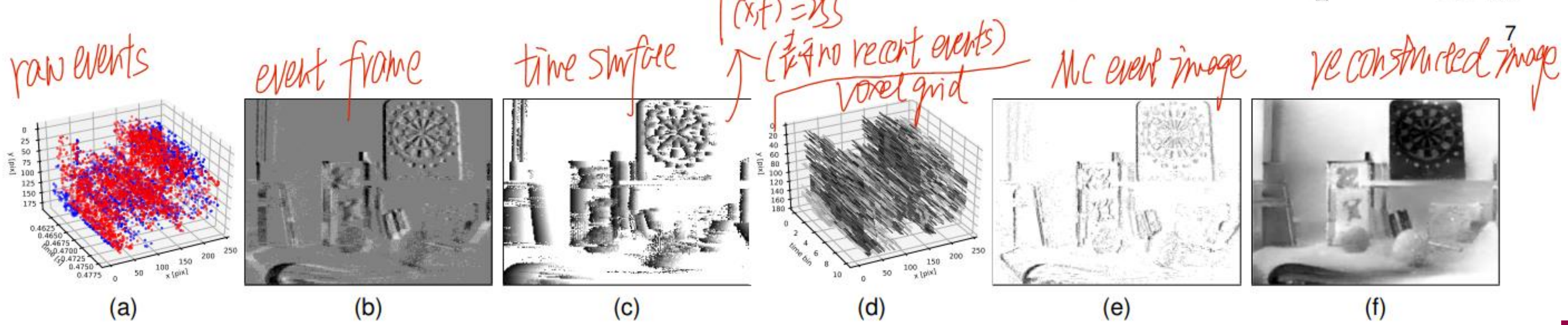
Event-by-event

1. Individual events -> probabilistic filters, Spiking Neural Networks (SNNs)

Groups or packets of events

1. Event frame/image or 2D histogram -> conventional cv algorithms can be applied
2. Motion-compensated event image
3. Time surface (TS): each pixel stores a time value -> objective of temporal difference can be used
4. Voxel grid
5. 3D point set: utilize the spatial-temporal information:(x_k, y_k, t_k)
6. Point sets on image plane -> mean-shift or ICP can be applied
7. Reconstructed frame-based images

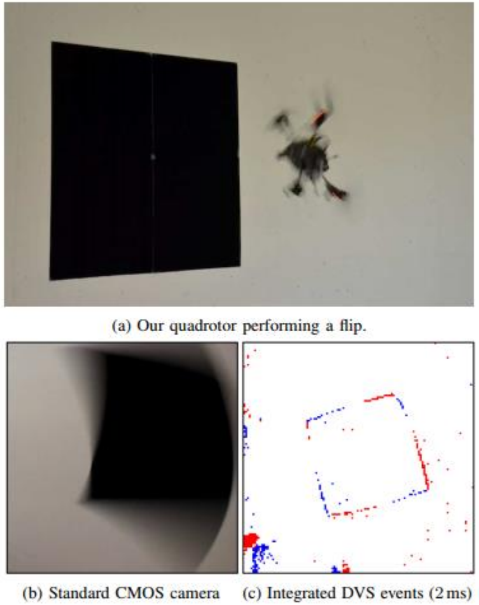




Task 1: Pose estimation and SLAM (monocular)

Challenges:

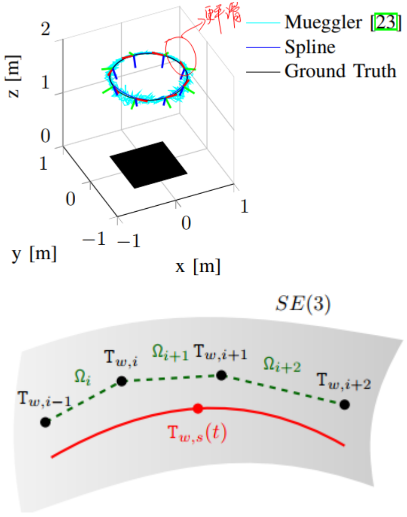
1. Dimensionality of the problem: pure rotation or 2D pose -> 6DoF pose tracking -> tracking and mapping
2. Type of motion: rotational or planar motion -> 6Dof motion; discrete-time trajectory -> continuous-time trajectory
3. Type of scene: artificial scenes -> natural scenes; indoor, small-scale -> outdoor, large-scale



(a) Our quadrotor performing a flip.

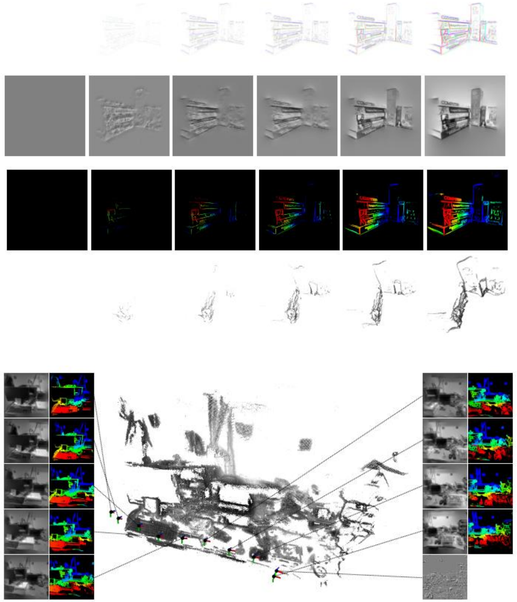
(b) Standard CMOS camera (c) Integrated DVS events (2 ms)

[mueggler2014event]
6-DoF pose tracking with artificial scenes

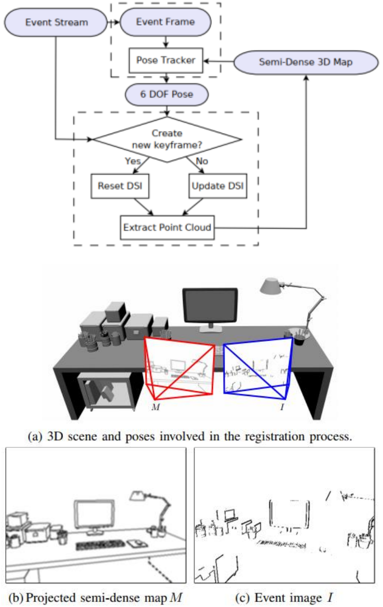


$$T_{w,s}(u(t)) = T_{w,i-1} \prod_{j=1}^{3} \exp\left(\tilde{\mathbf{B}}_j(u(t))\Omega_{i+j-1}\right)$$

[mueggler2015continuous]
Continuous-time trajectory



[kim2016real]
Three EKFs to estimate camera motion, log-intensity image, and inverse depth in real-time; but need GPU



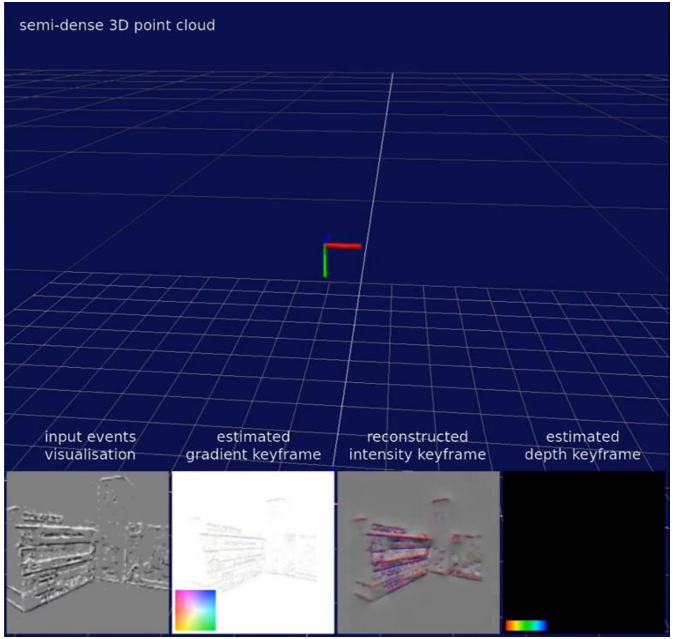
(a) 3D scene and poses involved in the registration process.

(b) Projected semi-dense map M (c) Event image I

[rebecq2016evo]
Semi-dense Event-based VO

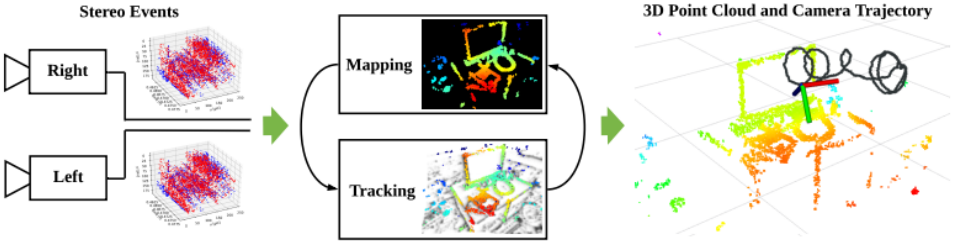
Example: [kim2016real])

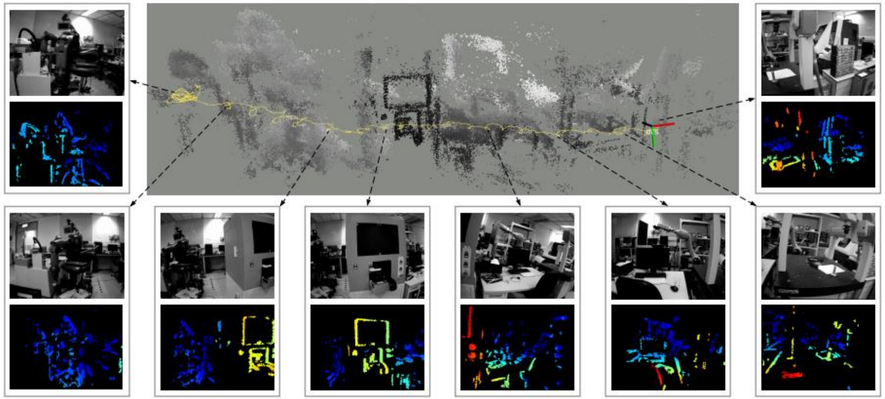
- Tracking: EKF in 6 DOF pose
  - Uses random walk model & inverse depth
  - Use 1st order approximation of generative event model to update pose
- Runs in real time on a GPU

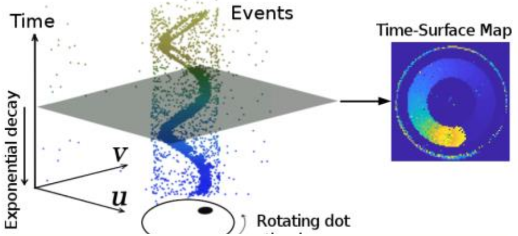

semi-dense 3D point cloud
input events visualisation
estimated gradient keyframe
reconstructed intensity keyframe
estimated depth keyframe

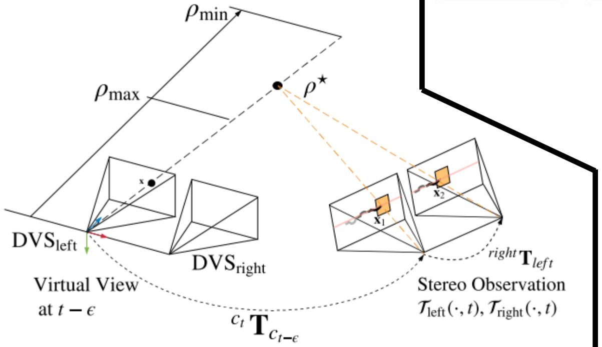
Kim et al., Real-Time 3D Reconstruction and 6-DoF Tracking with an Event Camera, ECCV'16

Task 1: Pose estimation and SLAM (stereo)



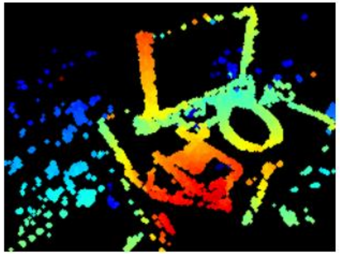


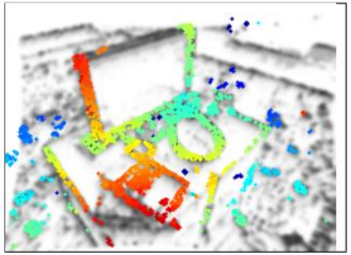



ρ_{min}
ρ_{max}
$\rho^\star$
x
DVS_{left}
$\text{DVS}_{\text{right}}$
Virtual View
at $t-\epsilon$
$\mathbf{x}_1$
$\mathbf{x}_2$
${}^{right}\mathbf{T}_{left}$
Stereo Observation
$\mathcal{T}_{\text{left}}(\cdot,t), \mathcal{T}_{\text{right}}(\cdot,t)$
${}^{c_t}\mathbf{T}_{c_{t-\epsilon}}$

$$\rho^\star = \arg\min_{\rho} C(\mathbf{x}, \rho, \mathcal{T}_{\text{left}}(\cdot,t), \mathcal{T}_{\text{right}}(\cdot,t), \mathbf{T}_{t-\delta t:t}) \quad (2)$$

$$C(\cdots) \doteq \sum_{\mathbf{x}_{1,i}\in W_1, \mathbf{x}_{2,i}\in W_2} \left\| \tau^t_{\text{left}}(\mathbf{x}_{1,i}) - \tau^t_{\text{right}}(\mathbf{x}_{2,i}) \right\|_2^2. \quad (3)$$





[zhou2018semi] Semi-Dense 3D Reconstruction with a stereo event camera -> [zhou2020event] Event-based stereo visual odometry

Task 1: Pose estimation and SLAM

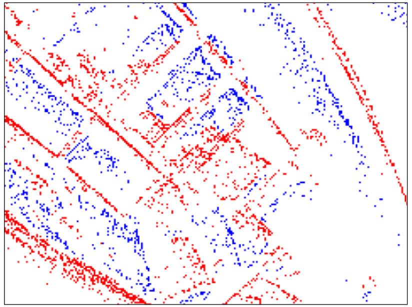
Opportunities:

1. Stereo event-based SLAM is seldom explored.
2. Most of existing works are direct-based, feature-based SLAM is not explored; bundle adjustment is also not explored;
3. Loop-closure capability to reduce drift.
4. The scales of event-based SLAM can be larger.
5. Robustness of SLAM systems can be improved by sensor fusion (IMUs, frame-based cameras, etc.).

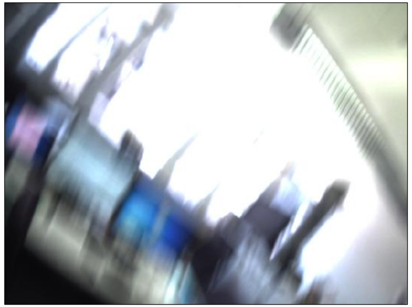
Task 2: Sensor fusion (event camera + frame camera + IMU)

How to find feasible applications?

< 10 years research



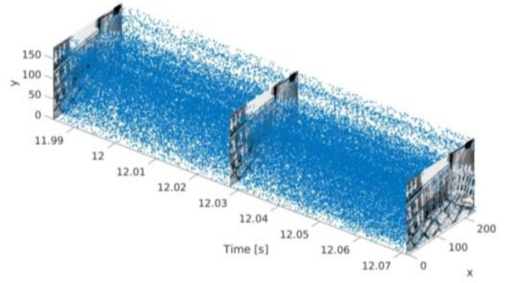
> 60 years of research!



	Event Camera	**Standard Camera**
Update rate	High (asynchronous): 1 MHz	Low (synchronous)
Dynamic Range	High (140 dB)	Low (60 dB)
Motion Blur	No	Yes
Static motion	No (event camera is a high pass filter)	Yes
Absolute intensity	No (reconstructable up to a constant)	Yes
Maturity	< 10 years of research	> 60 years of research!

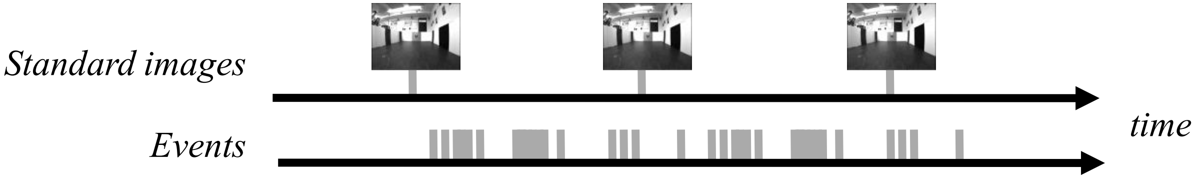
Task 2: Sensor fusion (event camera + frame camera + IMU)

- DAVIS: Combines **an event and a standard** camera in the same pixel array (→ the same pixel can both trigger events and integrate light intensity).
- **It also has an IMU**
- **It is useful for calibration**



Spatio-temporal visualization of the output of a DAVIS sensor

Temporal aggregation of events overlaid on a DAVIS frame



Standard images

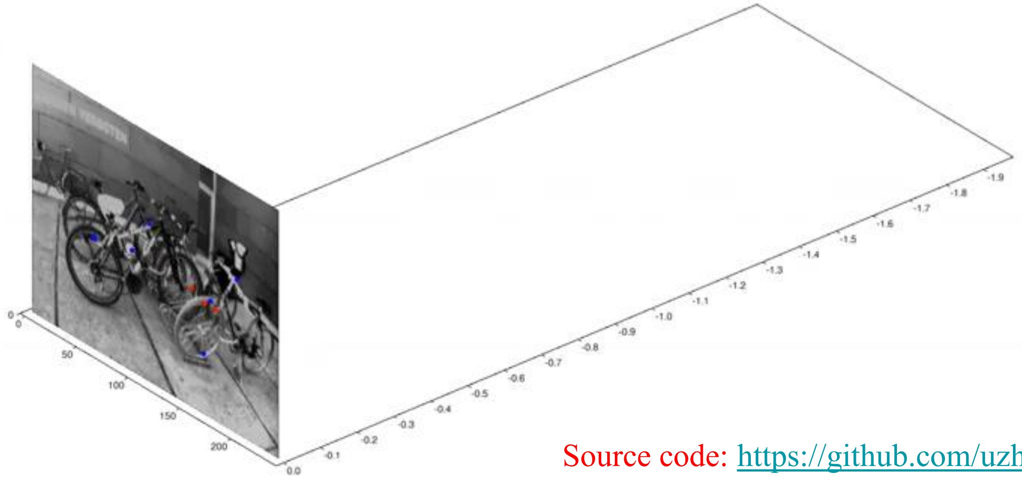
Events

time

Brandli et al. *A 240x180 130dB 3us latency global shutter spatiotemporal vision sensor*. IEEE JSSC, 2014

Task 2.1: Sensor fusion for feature tracking

- **Goal**: Extract features on **frames** and track them using only **events** in the **blind time** between two **frames**
- Uses the event generation model via **joint estimation of patch warping and optic flow**



Source code: https://github.com/uzh-rpg/rpg_eklt

Gehrig et al., EKLT: Asynchronous, Photometric Feature Tracking using Events and Frames, IJCV 2019.
PDF, YouTube, Evaluation Code, Tracking Code

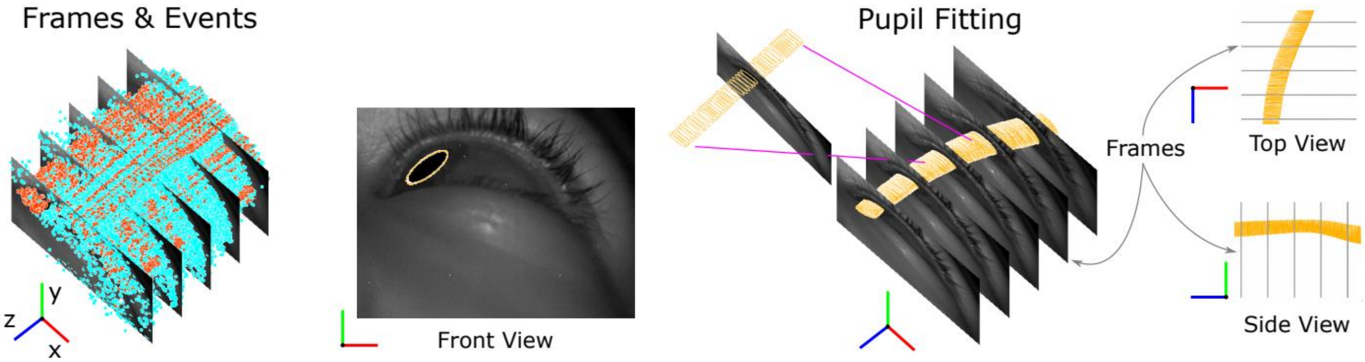
Task 2.1: Sensor fusion for High-Speed, Near-Eye Gaze Tracking

Task: Estimate gaze-vector of the eye at >1'000Hz in real-time on portable system

- Update Rate beyond 10 kHz
- Accuracy comparable to commercial product (EyeLink)
- Portable and low-power

Example Applications:

- AR/VR
- Driver drowsiness detection



Angelopoulos et al., "Event Based, Near-Eye Gaze Tracking Beyond 10,000Hz", Arxiv20. PDF

Task 2.1: Sensor fusion for High-Speed, Near-Eye Gaze Tracking

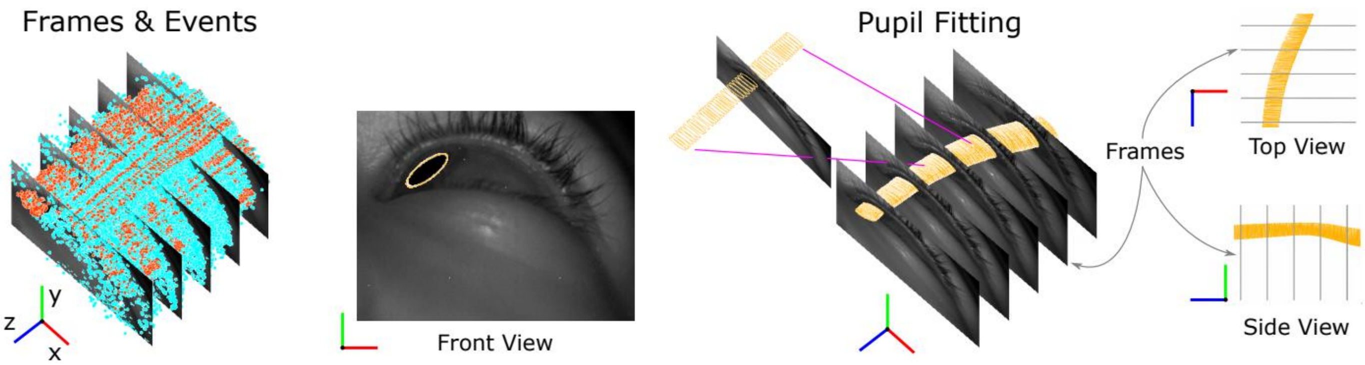
Task: Estimate gaze-vector of the eye at >1'000Hz in real-time on portable system

- Update Rate beyond 10 kHz
- Accuracy comparable to commercial product (EyeLink)
- Portable and low-power

system	update rate (Hz)	accuracy (°)	portable
Pupil Labs [1]	200	~ 1	✓
Tobii [3]	120	0.5–1.1	✓
EyeLink [2,14]	1,000	~ 0.5	
Ours	$> 10,000$	0.45–1.75	✓

Example Applications:

- AR/VR
- Driver drowsiness detection

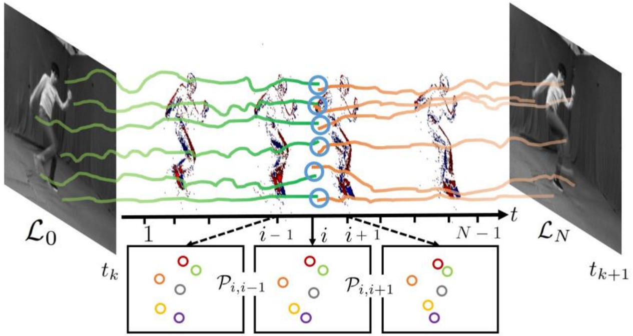


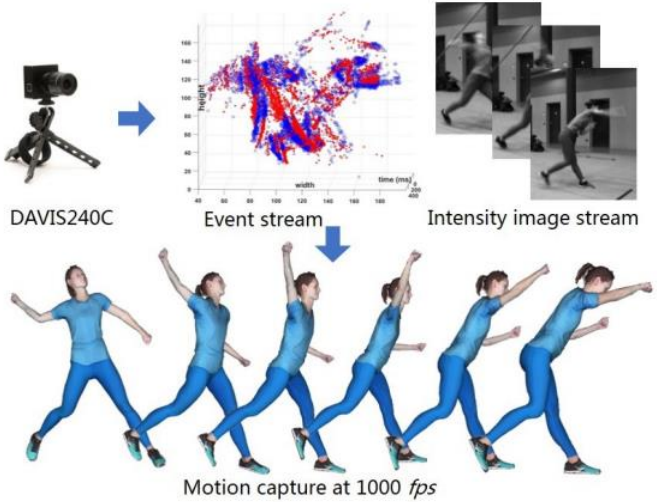
Angelopoulos et al., “Event Based, Near-Eye Gaze Tracking Beyond 10,000Hz”, Arxiv20. PDF

Task 2.1: Sensor fusion for High-Speed Human-Motion Tracking

Task: 3D human motion capture at 1'000 Hz

- 30x lower data-bandwidth than high-speed frame-based approach
- Works even in low-light
- Utilize frames (25 Hz) and events





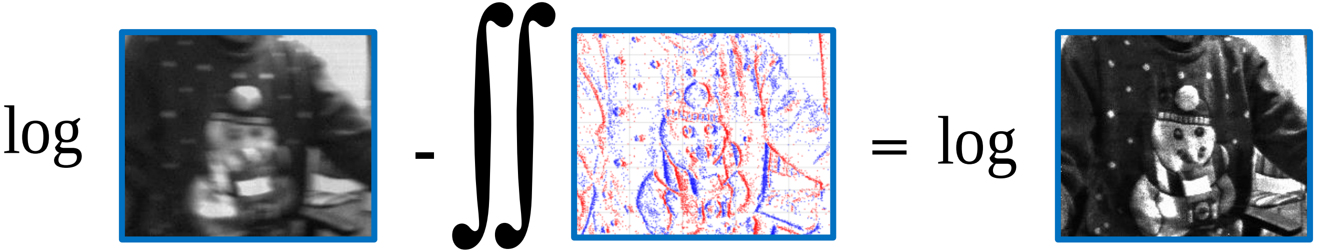
Xu et al., “EventCap: Monocular 3D Capture of High-Speed Human Motions using an Event Camera”, CVPR20. [PDF]

Calabreset al., "DHP19: Dynamic Vision Sensor 3D Human Pose Dataset", CVPRW19. [PDF]

Task 2.2: Sensor fusion for video deblurring

- A **blurry image** can be regarded as the **integral of a sequence of *latent images*** during the exposure time, while the **events** indicate the **changes between the latent images**.

- **Finding**: sharp image obtained by subtracting the double integral of event from input image



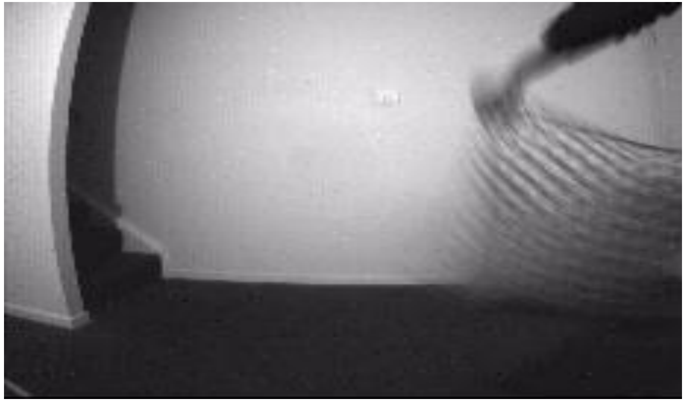
Input blur image **Input events** **Output sharp image**

Pan et al., Bringing a Blurry Frame Alive at High Frame-Rate with an Event Camera, CVPR’19

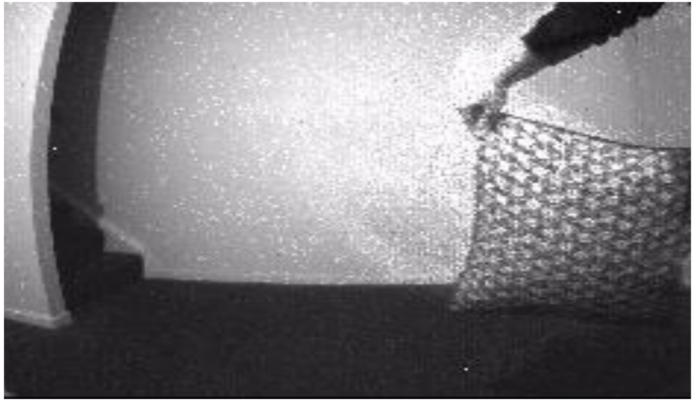
Task 2.2: Sensor fusion for video deblurring

- A **blurry image** can be regarded as the **integral of a sequence of *latent images*** during the exposure time, while the **events** indicate the **changes between the latent images**.

- **Finding**: sharp image obtained by subtracting the double integral of event from input image



Input blur image



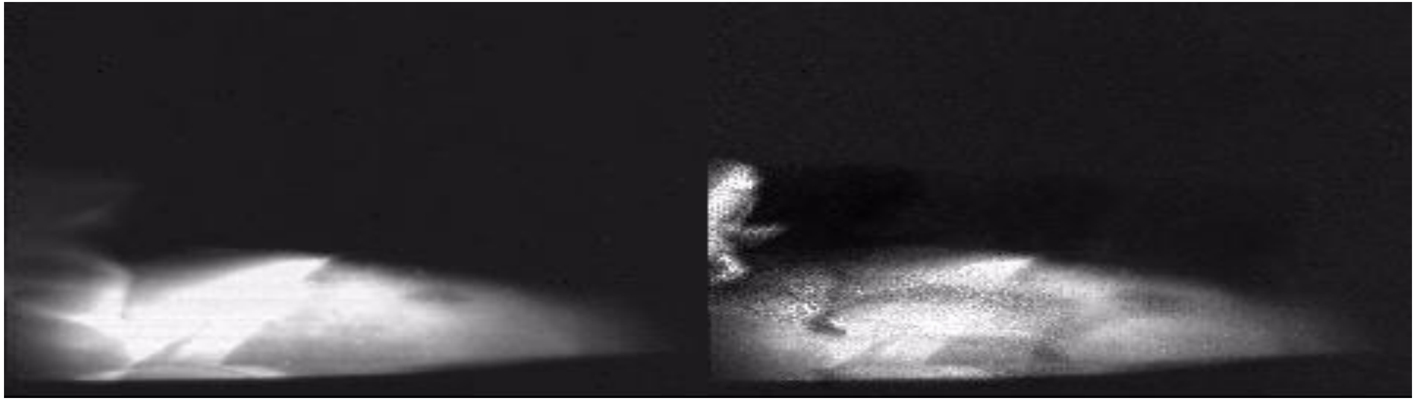
Output sharp video

Pan et al., Bringing a Blurry Frame Alive at High Frame-Rate with an Event Camera, CVPR’19

Task 2.2: Sensor fusion for video deblurring

- A **blurry image** can be regarded as the **integral of a sequence of *latent images*** during the exposure time, while the **events** indicate the **changes between the latent images**.

- **Finding**: sharp image obtained by subtracting the double integral of event from input image



Input blur image **Output sharp video**

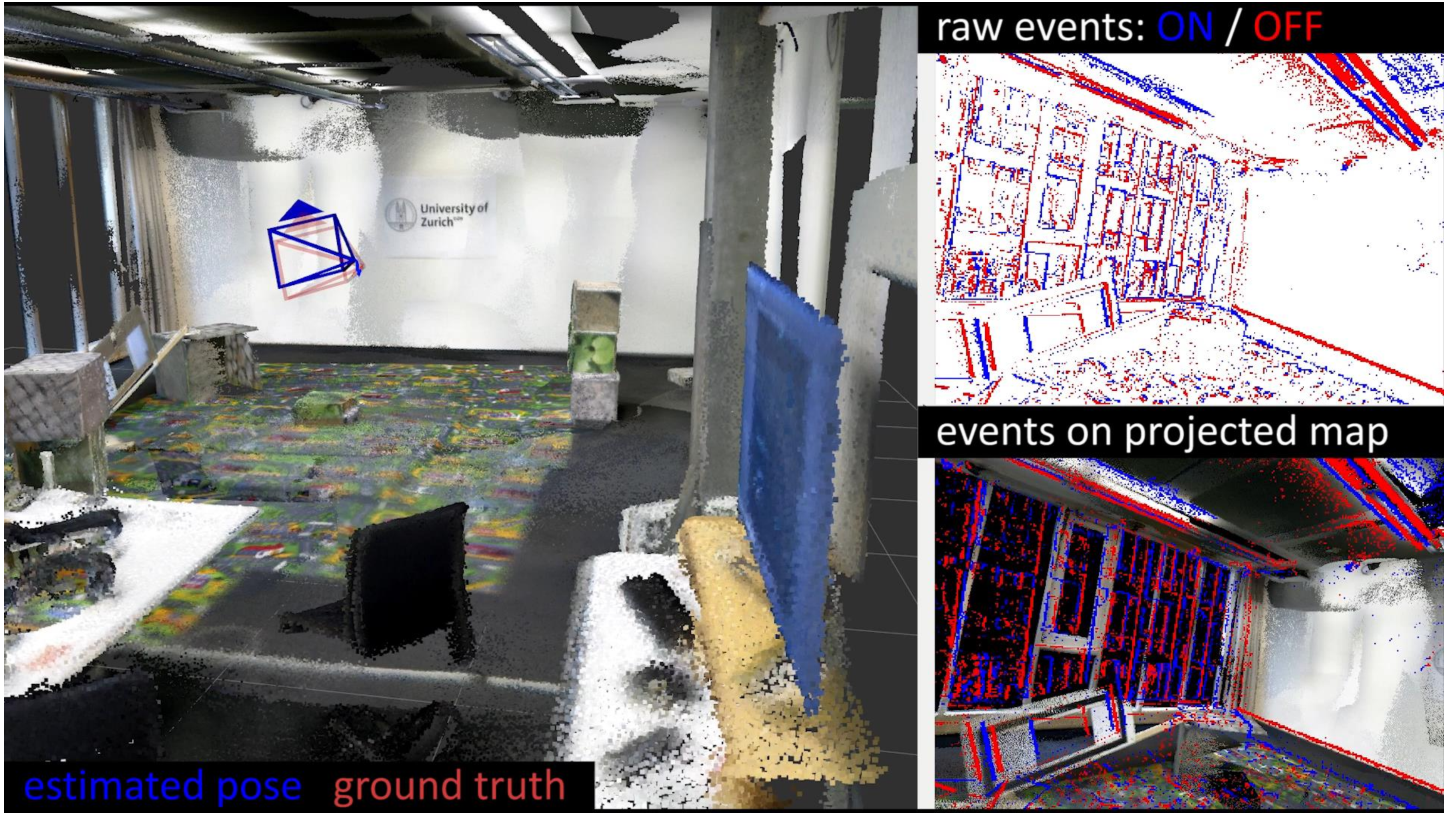
Pan et al., Bringing a Blurry Frame Alive at High Frame-Rate with an Event Camera, CVPR'19

Task 2.3: Sensor fusion for visual localization

- Probabilistic, simultaneous motion & contrast estimation from $C = -\nabla L \cdot \mathbf{u}$
- Assumes photometric map (x,y,z, grayscale Intensity) is given
- Useful for VR/AR applications (low-latency, HDR, no motion blur)
- Requires GPU to run in real time

Gallego et al., Event-based 6-DOF Camera Tracking from Photometric Depth Maps, **T-PAMI**, 2018. PDF Video

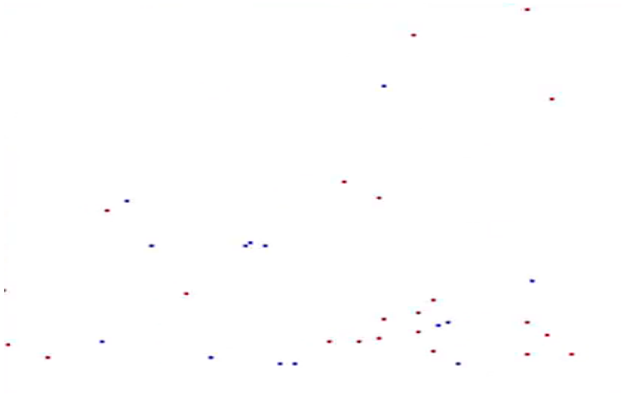
Task 2.3: Sensor fusion for visual localization



Bryner et al., Event-based, Direct Camera Tracking from a Photometric Depth Map, ICRA’19. PDF Video

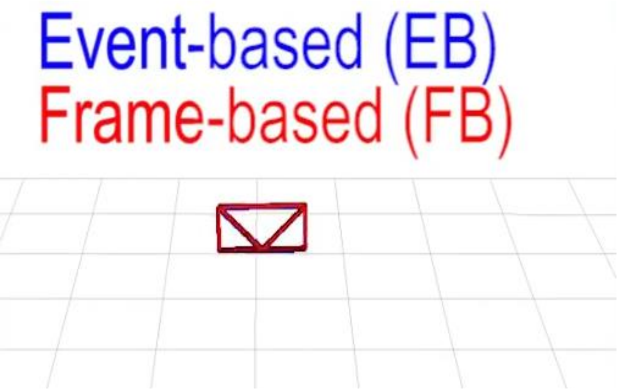
Task 2.3: Sensor fusion for visual localization

Event camera



Standard camera

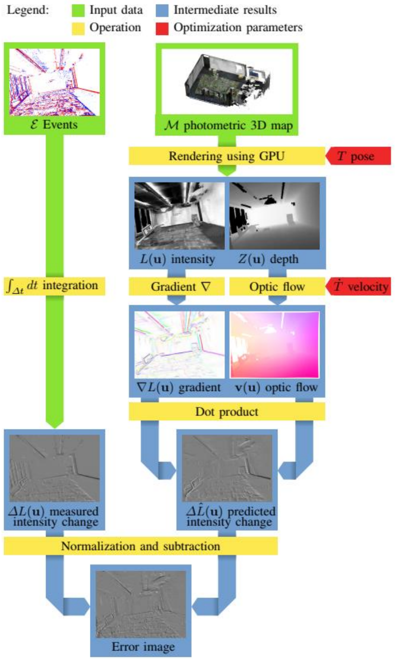
Motion estimation



Gallego et al., Event-based 6-DOF Camera Tracking from Photometric Depth Maps, **T-PAMI**'18. PDF Video

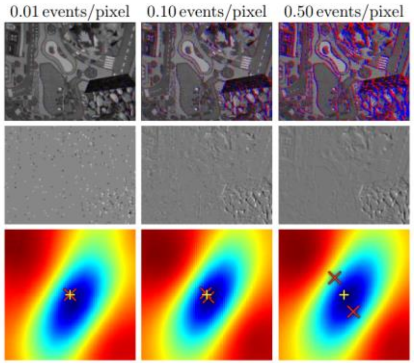
Task 2.3: Sensor fusion for visual localization

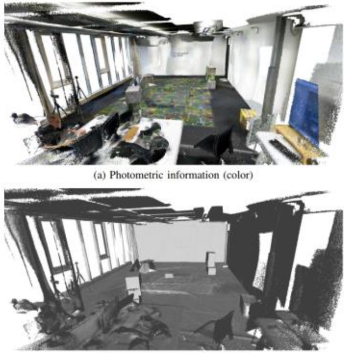
- Improve [Gallego2018Event] (filter-based) with the Nonlinear optimization
- Propose the metric the calculate difference between event images and frame-based images



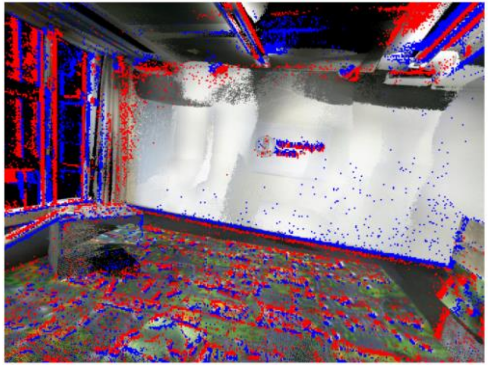
Legend:
Input data
Operation
Intermediate results
Optimization parameters
$\mathcal{E}$ Events
$\mathcal{M}$ photometric 3D map
Rendering using GPU
T pose
$L(\mathbf{u})$ intensity
$Z(\mathbf{u})$ depth
$\int_{\Delta t} dt$ integration
Gradient ∇
Optic flow
$\dot{T}$ velocity
$\nabla L(\mathbf{u})$ gradient
$\mathbf{v}(\mathbf{u})$ optic flow
Dot product
$\Delta L(\mathbf{u})$ measured intensity change
$\Delta \hat{L}(\mathbf{u})$ predicted intensity change
Normalization and subtraction
Error image

$$\min_{T,\dot{T}} \left\| \frac{\Delta L(\mathbf{u})}{\|\Delta L(\mathbf{u})\|_{\mathcal{L}^2(\Omega)}} - \frac{\Delta \hat{L}(\mathbf{u};T,\dot{T})}{\|\Delta \hat{L}(\mathbf{u};T,\dot{T})\|_{\mathcal{L}^2(\Omega)}} \right\|^2_{\mathcal{L}^2(\Omega)}$$




(a) Photometric information (color)

(b) Geometric information (shape)



Bryner et al., Event-based, Direct Camera Tracking from a Photometric 3D Map using Nonlinear Optimization, **ICRA**, 2019

Task 2.4: Sensor fusion for pose estimation

Challenges:

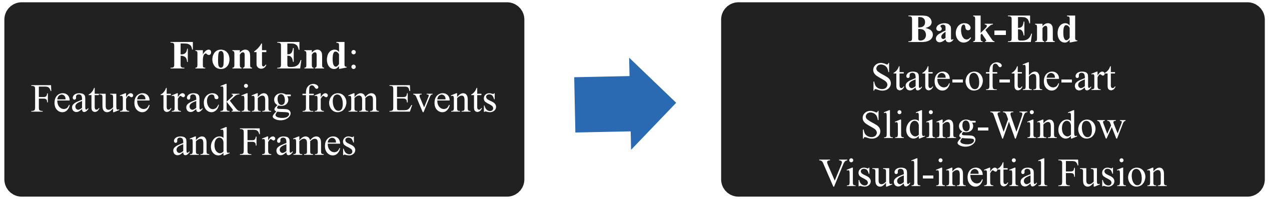
- A key issue in VIO is how to temporally fuse data from the synchronous, high-rate IMU (e.g., 1 kHz) and the asynchronous event camera.

Three current solutions:

1. Use an asynchronous probabilistic filter
2. Use pre-integration theory to convert IMU data into lower-rate pieces of information at desired times where events are collected
3. Consider a continuous-time framework so that both sensor measurements are referred to a common temporal axis that is also used to model the camera poses.

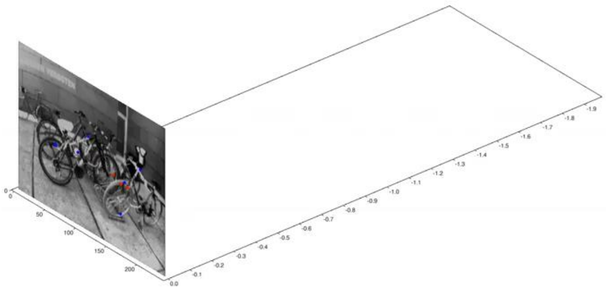
Task 2.4: Sensor fusion for pose estimation

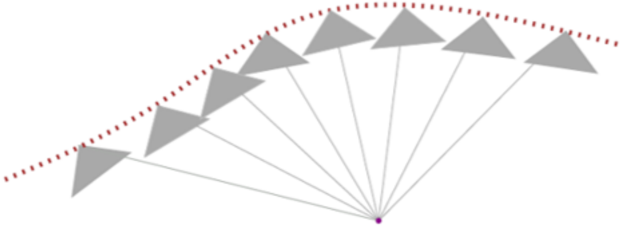
Ultimate SLAM: combining Events + Frames + IMU



Front End: Feature tracking from Events and Frames

Back-End State-of-the-art Sliding-Window Visual-inertial Fusion



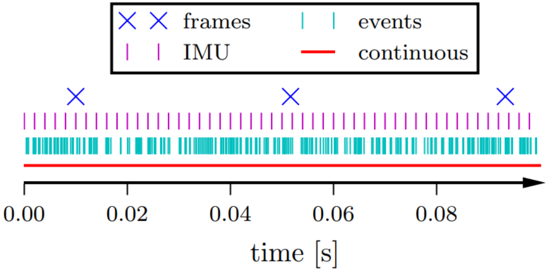


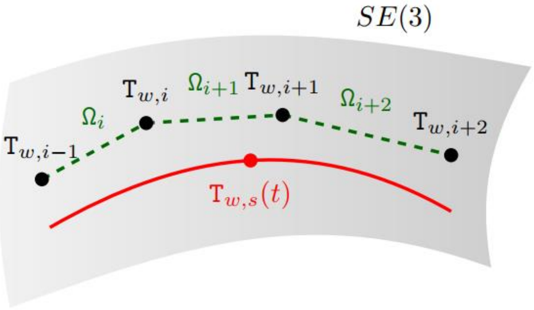
Rebecq et al., Real-time Visual-Inertial Odometry for Event Cameras using Keyframe-based Nonlinear Optimization, BMVC 2018.
Rosinol et al., Ultimate SLAM? **RAL'18 – Best RAL'18 Paper Award Honorable Mention** PDF. Video. IEEE Spectrum.
Mueggler et al., Continuous-Time Visual-Inertial Odometry for Event Cameras, **TRO'18**. PDF

Task 2.4: Sensor fusion for pose estimation

Events + IMU, continuous-time formulation

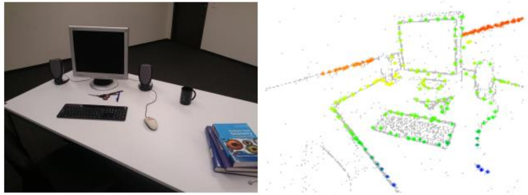
- ➢ Nonlinear Optimization
- ➢ Assumption: current data association is given





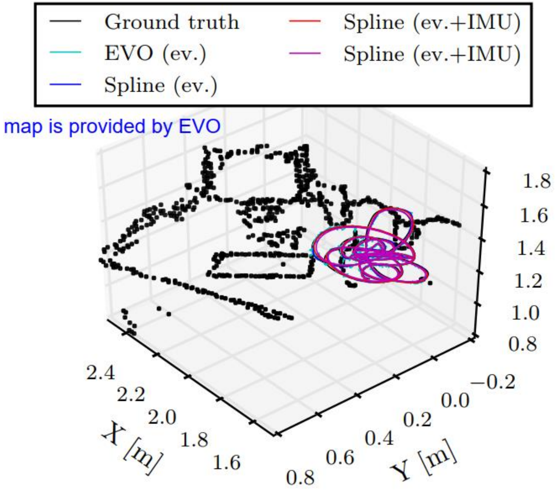
$$
\begin{aligned}
F \doteq\ & \frac{1}{N}\sum_{k=1}^{N}\frac{1}{\sigma_e^2}\|\mathbf{e}_k - \hat{\mathbf{e}}_k(\mathbf{x}(t_k),\mathcal{M})\|^2 \\
& + \frac{1}{M}\sum_{j=1}^{M}\frac{1}{\sigma_\omega^2}\|\boldsymbol{\omega}_j - \hat{\boldsymbol{\omega}}_j(\mathbf{x}(t_j))\|^2 \\
& + \frac{1}{M}\sum_{j=1}^{M}\frac{1}{\sigma_a^2}\|\mathbf{a}_j - \hat{\mathbf{a}}_j(\mathbf{x}(t_j))\|^2,
\end{aligned}
$$

$$
\{\mathrm{T}_{w,i}^*, \boldsymbol{\theta}^*\} = \arg\min_{\mathrm{T},\boldsymbol{\theta}} F.
$$



(a) Scene *desk*

(b) Event-based tracking. Events (gray) and reprojected map (colored using depth).



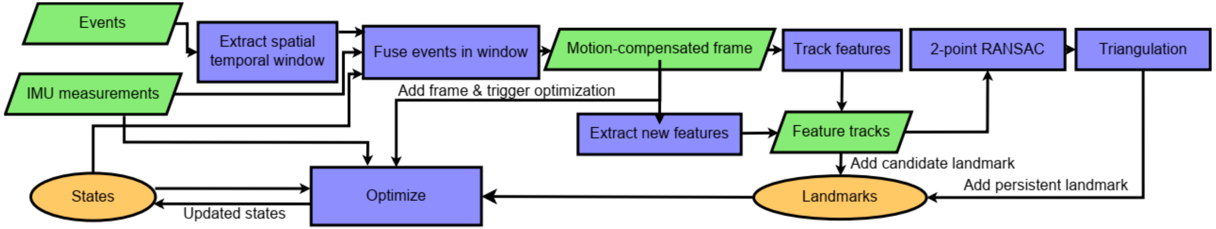
map is provided by EVO

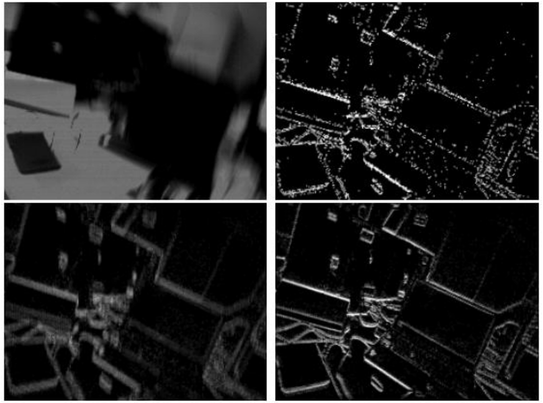
Mueggler et al., Continuous-Time Visual-Inertial Odometry for Event Cameras, **TRO'18**. PDF

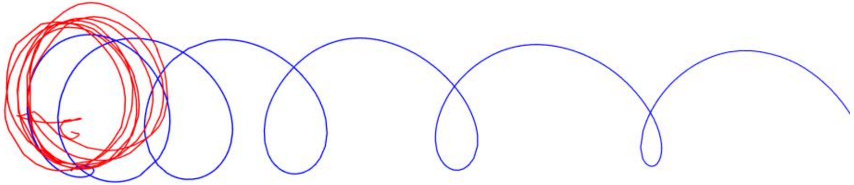
Task 2.4: Sensor fusion for pose estimation

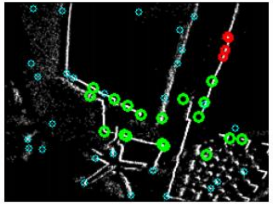
Events + IMU, discrete-time formulation

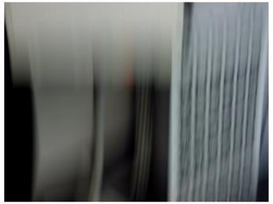
- ➢ Extend the *OKVIS* framework
- ➢ Use IMU measurement to compensate each event frame









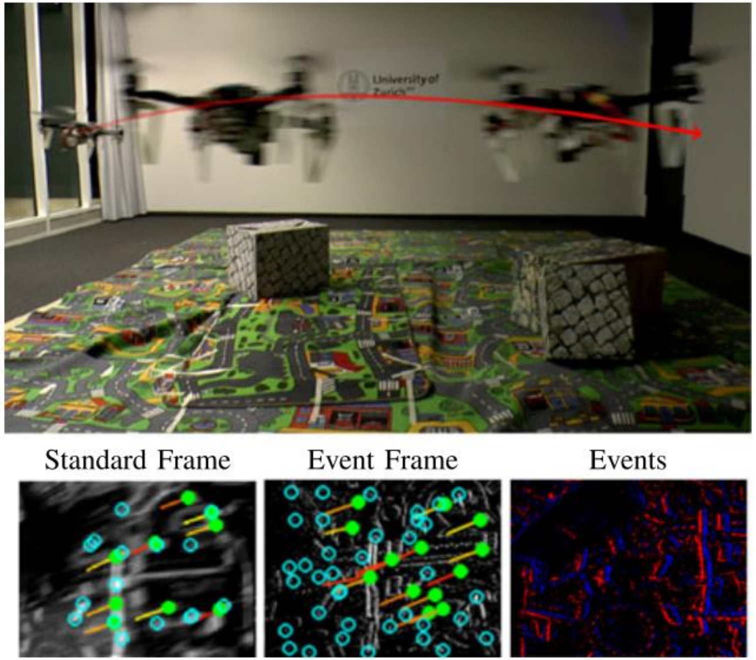


Rebecq et al., Real-time Visual-Inertial Odometry for Event Cameras using Keyframe-based Nonlinear Optimization, BMVC 2018.

Application: Autonomous Drone Navigation in Low Light

Extension: Ultimate SLAM running on board (CPU: Odroid XU4)

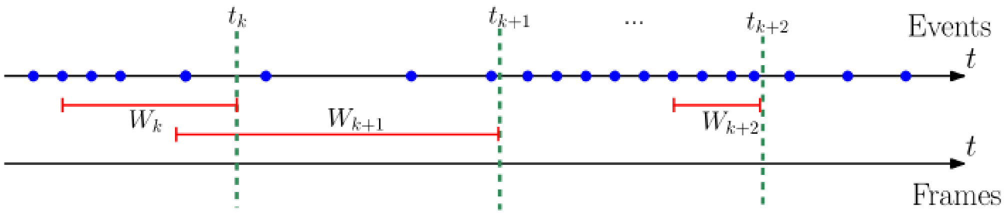
- ➢ Extract FAST corners from event frames and standard frames; KLT trackers; nonlinear optimization



Standard Frame

Event Frame

Events

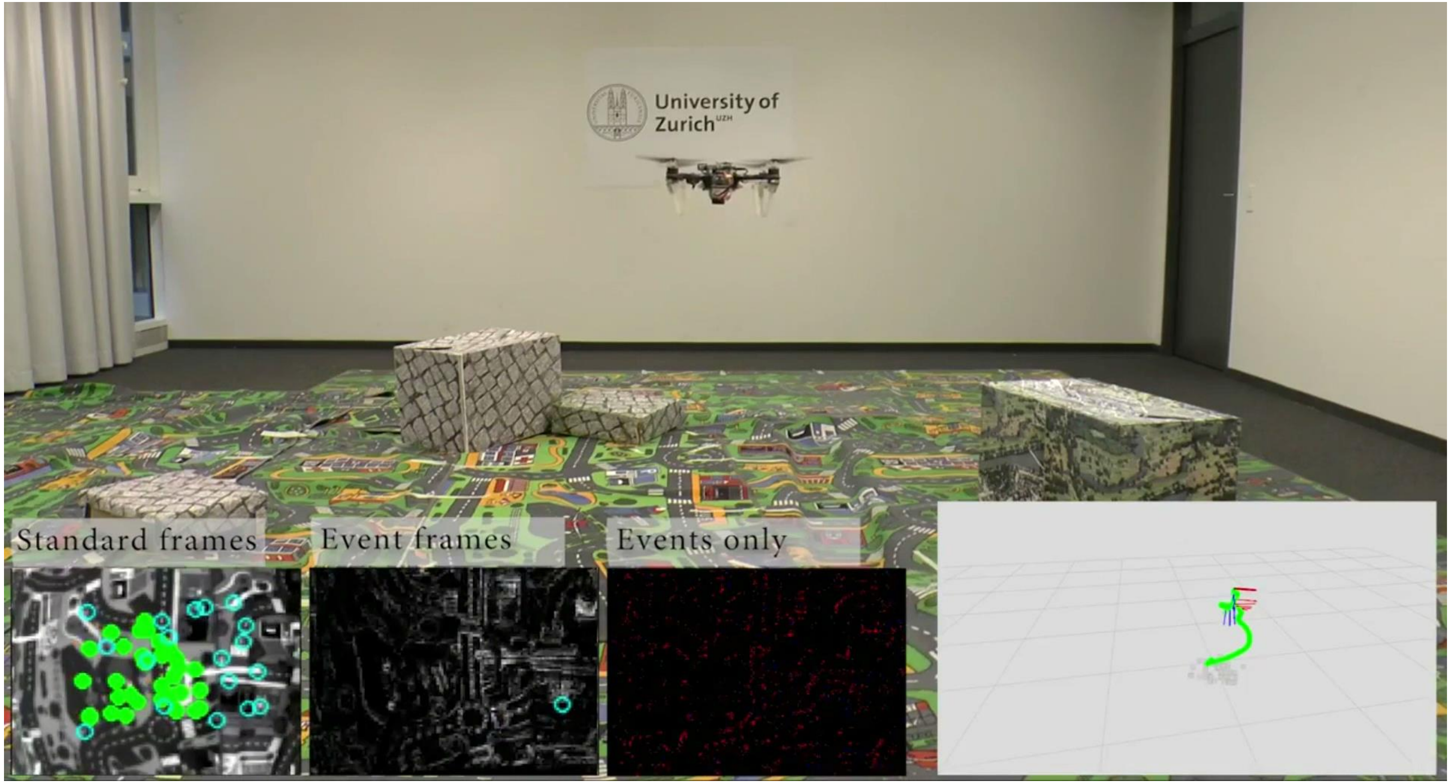


$$J = \sum_{i=0}^{1} \sum_{k=1}^{K} \sum_{j \in \mathcal{J}(i,k)} \mathbf{e}^{i,j,k^T} \mathbf{W}_r^{i,j,k} \mathbf{e}^{i,j,k} + \sum_{k=1}^{K-1} \mathbf{e}_s^{k^T} \mathbf{W}_s^k \mathbf{e}_s^k$$

Rosinol et al., Ultimate SLAM? **RAL'18 – Best RAL'18 Paper Award Honorable Mention** PDF. Video. IEEE Spectrum.

Application: Autonomous Drone Navigation in Low Light

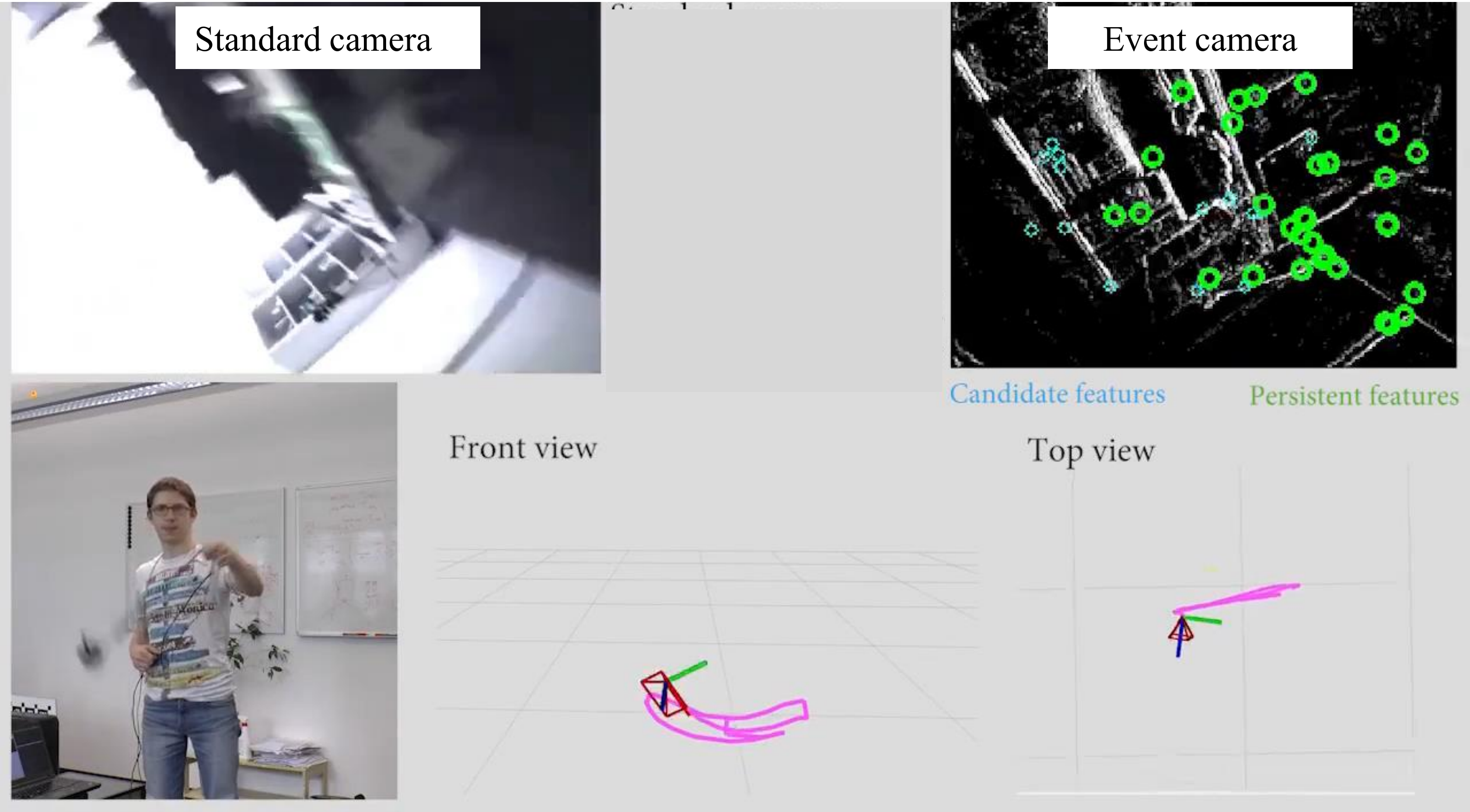
Extension: Ultimate SLAM running on board (CPU: Odroid XU4)



Rosinol et al., Ultimate SLAM? **RAL'18 – Best RAL'18 Paper Award Honorable Mention** PDF. Video. IEEE Spectrum.

Application: Large Motion Blur

85% accuracy gain over standard Visual-Inertial SLAM in HDR and high speed scenes



Rosinol et al., Ultimate SLAM? **RAL'18 – Best RAL'18 Paper Award Honorable Mention** PDF. Video. IEEE Spectrum.

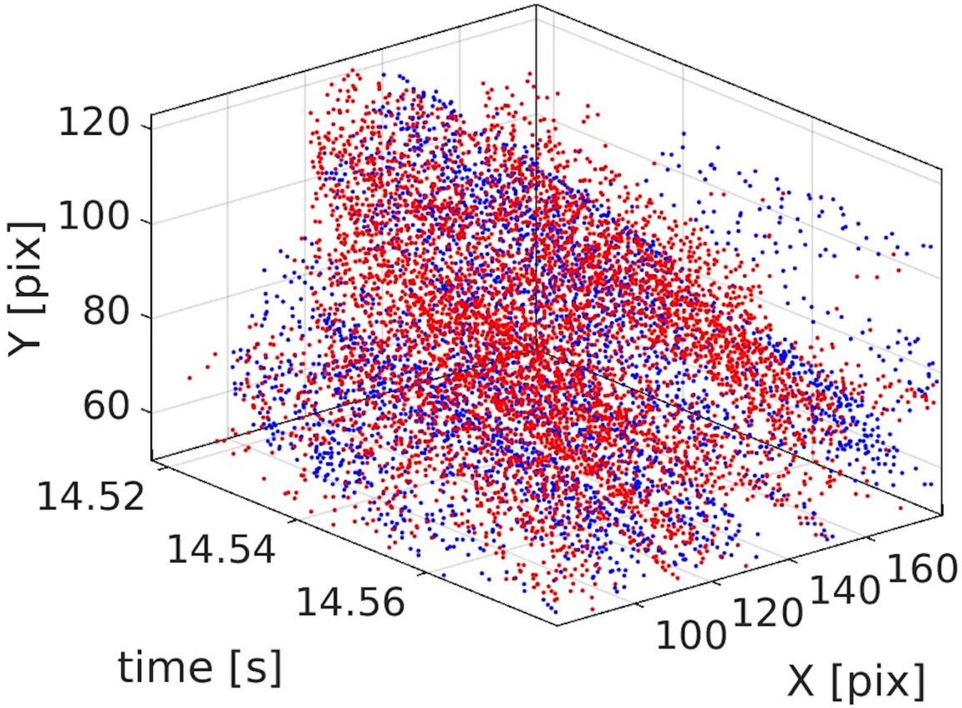
Task 3: Contrast Maximization

Focus Maximization for:

- Motion estimation
- 3D reconstruction
- SLAM
- Optical flow estimation
- Feature tracking
- Motion segmentation
- Unsupervised learning

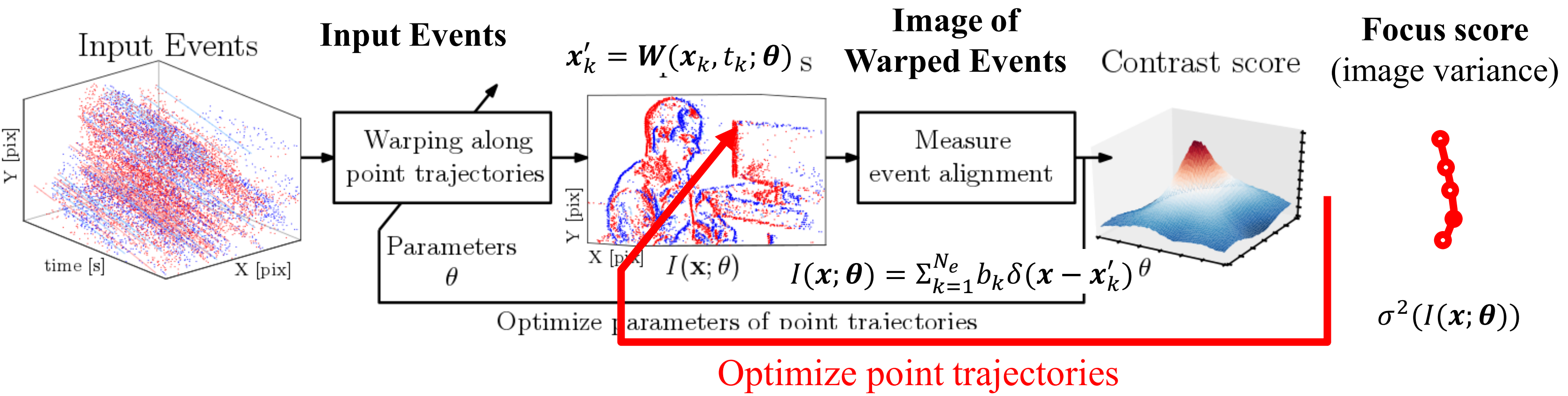
Gallego et al., A Unifying Contrast Maximization Framework for Event Cameras, CVPR18, PDF, YouTube
Gallego et al., Focus Is All You Need: Loss Functions for Event-based Vision, CVPR19, PDF.

Basic idea: Warp **spatio-temporal volume** of events to **maximize focus/contrast** (e.g., sharpness) of the resulting image



Gallego et al., A Unifying Contrast Maximization Framework for Event Cameras, CVPR18, PDF, YouTube
Gallego et al., Focus Is All You Need: Loss Functions for Event-based Vision, CVPR19, PDF.

Focus Maximization Framework



Input Events

$\boldsymbol{x}'_k = \boldsymbol{W}(\boldsymbol{x}_k, t_k; \boldsymbol{\theta})$

Image of Warped Events

$I(\boldsymbol{x}; \boldsymbol{\theta}) = \Sigma_{k=1}^{N_e} b_k \delta(\boldsymbol{x} - \boldsymbol{x}'_k)$

Focus score
(image variance)

$\sigma^2(I(\boldsymbol{x}; \boldsymbol{\theta}))$

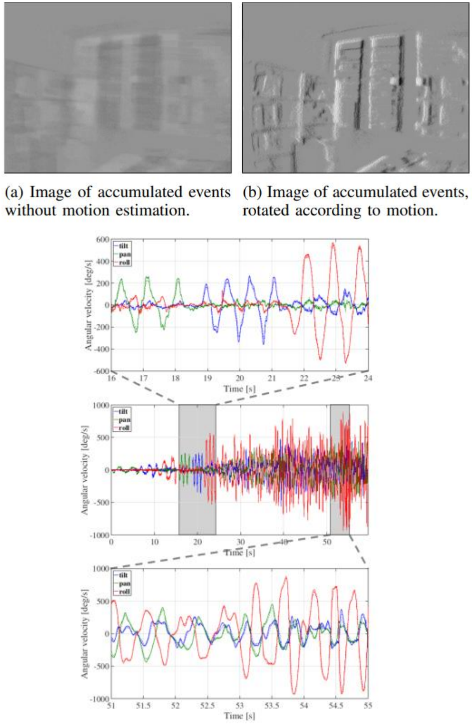
Optimize point trajectories

Can be implemented in a sliding-window fashion to enable per low-latency, per-event update rate

Runs in real time on a CPU

Gallego et al., Accurate Angular Velocity Estimation with an Event Camera, IEEE RAL’16. PDF. Video.
Gallego et al., A Unifying Contrast Maximization Framework for Event Cameras, CVPR18, PDF, YouTube
Gallego et al., Focus Is All You Need: Loss Functions for Event-based Vision, CVPR19, PDF.

Related Work

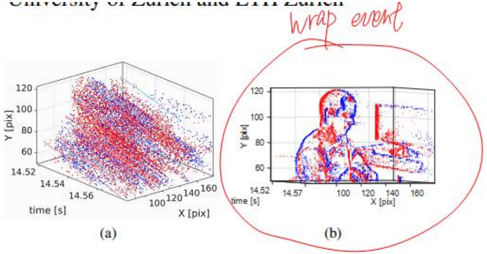


(a) Image of accumulated events without motion estimation.

(b) Image of accumulated events, rotated according to motion.

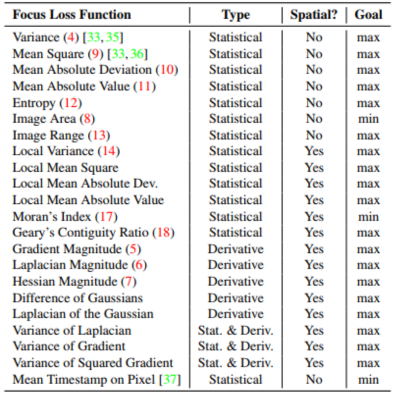
[gallego2016accurate]
Estimate pure rotation

University of Zurich and ETH Zurich



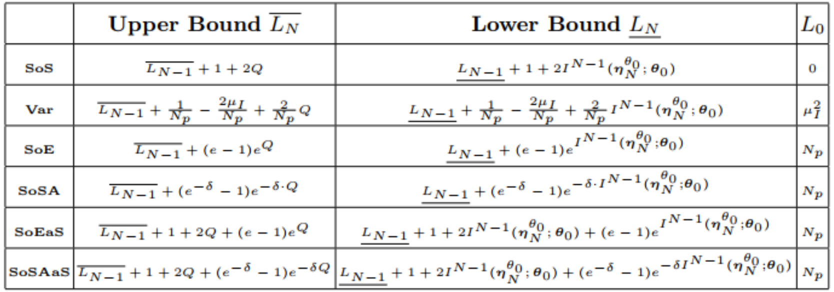
wrap event

[gallege2018a]
Define the contrast maximization and use apply to many tasks



Focus Loss Function	Type	Spatial?	Goal
Variance (4) [33, 35]	Statistical	No	max
Mean Square (9) [33, 36]	Statistical	No	max
Mean Absolute Deviation (10)	Statistical	No	max
Mean Absolute Value (11)	Statistical	No	max
Entropy (12)	Statistical	No	max
Image Area (8)	Statistical	No	min
Image Range (13)	Statistical	No	max
Local Variance (14)	Statistical	Yes	max
Local Mean Square	Statistical	Yes	max
Local Mean Absolute Dev.	Statistical	Yes	max
Local Mean Absolute Value	Statistical	Yes	max
Moran's Index (17)	Statistical	Yes	min
Geary's Contiguity Ratio (18)	Statistical	Yes	max
Gradient Magnitude (5)	Derivative	Yes	max
Laplacian Magnitude (6)	Derivative	Yes	max
Hessian Magnitude (7)	Derivative	Yes	max
Difference of Gaussians	Derivative	Yes	max
Laplacian of the Gaussian	Derivative	Yes	max
Variance of Laplacian	Stat. & Deriv.	Yes	max
Variance of Gradient	Stat. & Deriv.	Yes	max
Variance of Squared Gradient	Stat. & Deriv.	Yes	max
Mean Timestamp on Pixel [37]	Statistical	No	min

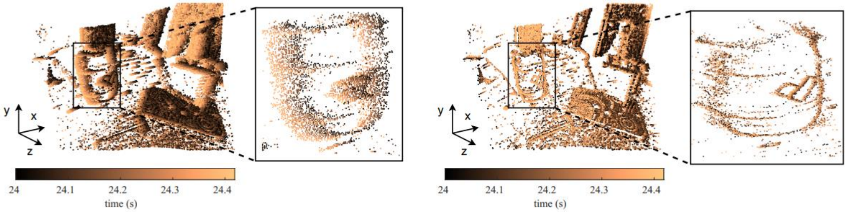
[gallege2019focus]
Explore different objective functions



	Upper Bound $\overline{L_N}$	Lower Bound $\underline{L_N}$	L_0
SoS	$\overline{L_{N-1}} + 1 + 2Q$	$\underline{L_{N-1}} + 1 + 2I^{N-1}(\eta_N^{\theta_0};\theta_0)$	0
Var	$\overline{L_{N-1}} + \frac{1}{N_p} - \frac{2\mu_I}{N_p} + \frac{2}{N_p}Q$	$\underline{L_{N-1}} + \frac{1}{N_p} - \frac{2\mu_I}{N_p} + \frac{2}{N_p}I^{N-1}(\eta_N^{\theta_0};\theta_0)$	μ_I^2
SoE	$\overline{L_{N-1}} + (e-1)e^{Q}$	$\underline{L_{N-1}} + (e-1)e^{I^{N-1}(\eta_N^{\theta_0};\theta_0)}$	N_p
SoSA	$\overline{L_{N-1}} + (e^{-\delta}-1)e^{-\delta\cdot Q}$	$\underline{L_{N-1}} + (e^{-\delta}-1)e^{-\delta\cdot I^{N-1}(\eta_N^{\theta_0};\theta_0)}$	N_p
SoEaS	$\overline{L_{N-1}} + 1 + 2Q + (e-1)e^{Q}$	$\underline{L_{N-1}} + 1 + 2I^{N-1}(\eta_N^{\theta_0};\theta_0) + (e-1)e^{I^{N-1}(\eta_N^{\theta_0};\theta_0)}$	N_p
SoSAaS	$\overline{L_{N-1}} + 1 + 2Q + (e^{-\delta}-1)e^{-\delta Q}$	$\underline{L_{N-1}} + 1 + 2I^{N-1}(\eta_N^{\theta_0};\theta_0) + (e^{-\delta}-1)e^{-\delta I^{N-1}(\eta_N^{\theta_0};\theta_0)}$	N_p

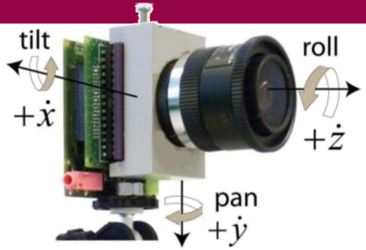
[peng2020globally]
Propose the global solution: BnB to solve 6 objective functions

[liu2020globally]
Propose the global solution: BnB

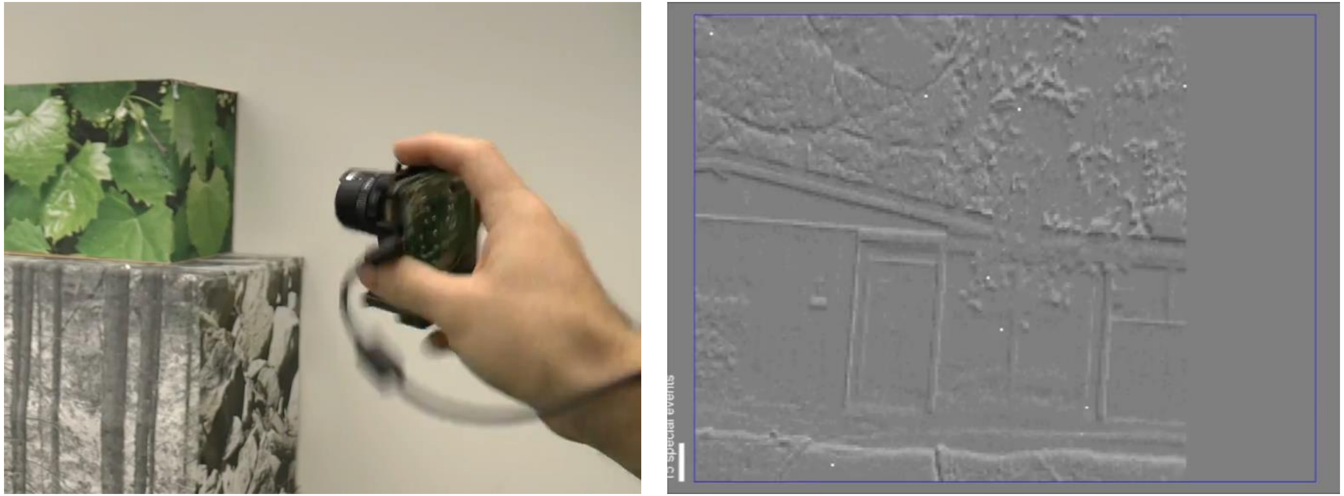


[nunes2020entropy]
Two contributions: propose new metrics: entropy minimization; new formulation to directly operate features in 3D

Application 1: Image Stabilization

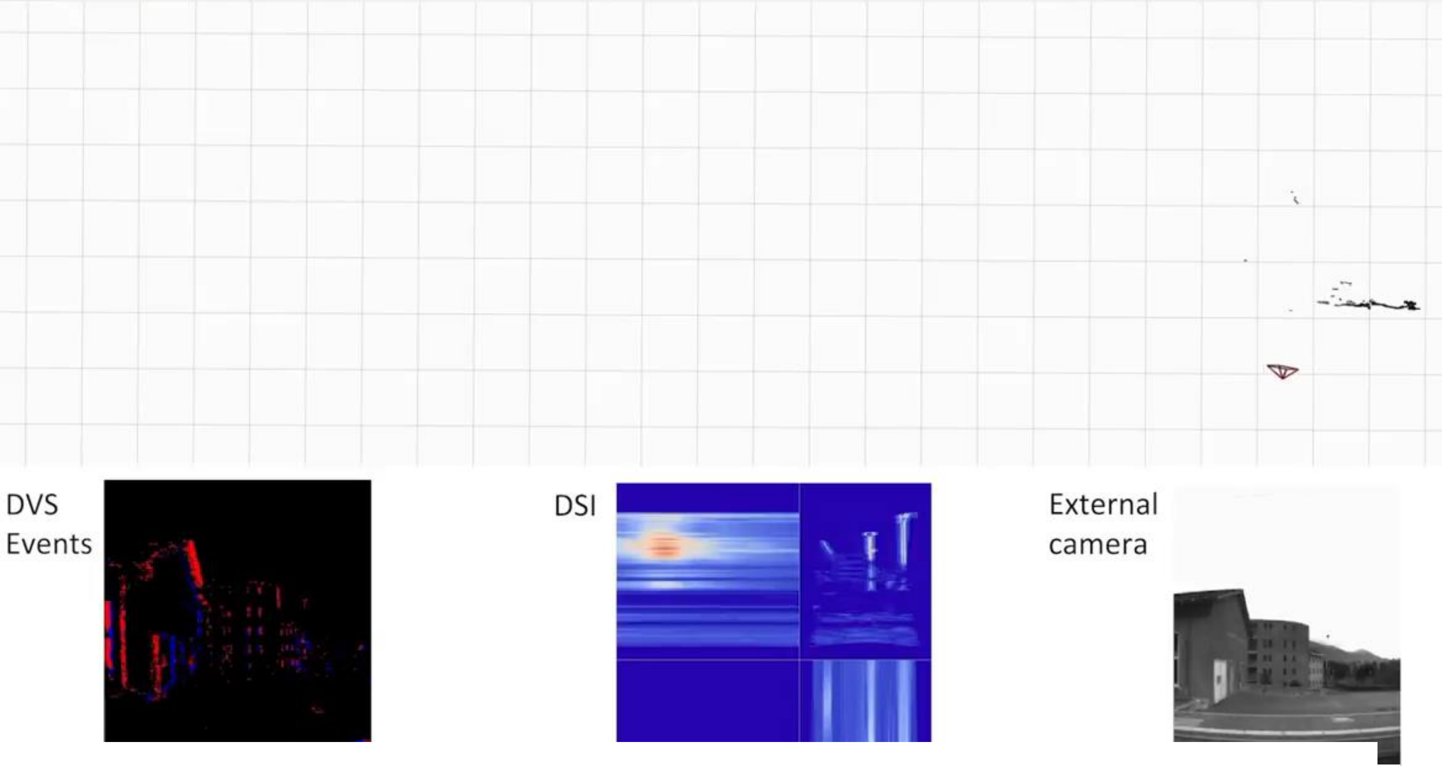


- Problem: Estimate rotational motion (3DoF) of an event camera
- Can process millions of events per second in real time on a smartphone CPU (OdroidXU4)
- Works up to over ~1,000 deg/s



Gallego et al., Accurate Angular Velocity Estimation with an Event Camera, IEEE RAL’16. PDF. Video.

Application 2: 3D Reconstruction from a Train at 200km/h



In collaboration with SIEMENS and the Swiss Railway company, SE SBB CFF FFS

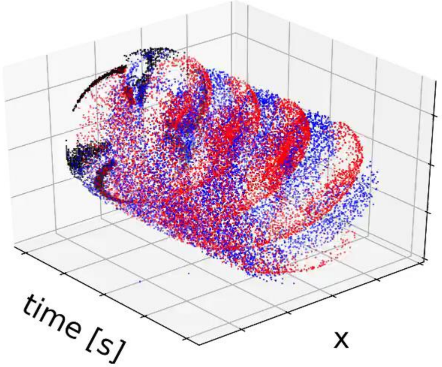
Rebecq et al., EVO: A Geometric Approach to Event-based 6-DOF Parallel Tracking and Mapping, RAL'17. PDF. Video

Rebecq et al., EMVS: Event-based Multi-View Stereo, IJCV, 2018. PDF Video Source Code

Application 3: Motion Segmentation

Conventional Frames

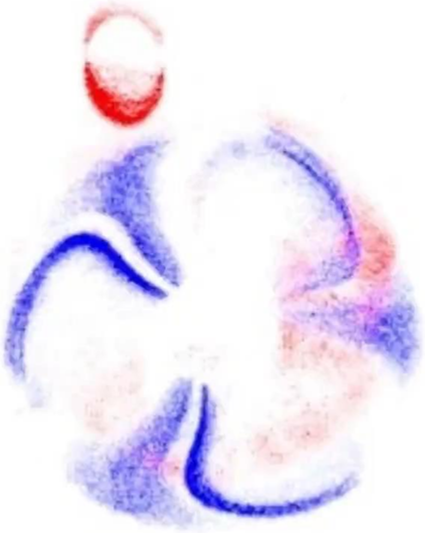
Events



Sequence: Fan and Coin

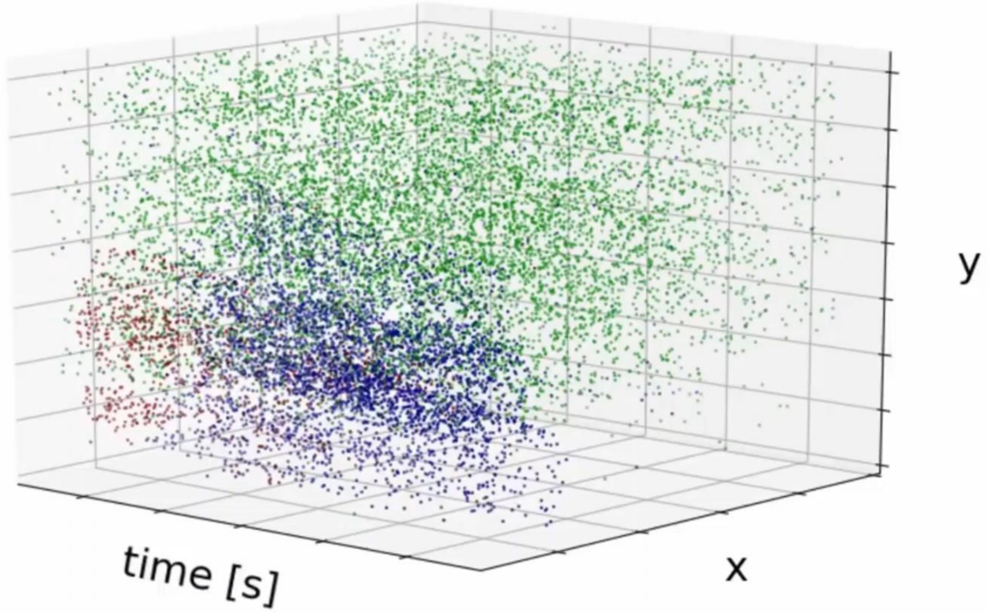
One motion model is used per cluster; one for the fan, modelling rotation, one for the coin, modelling optic flow

Motion-Compensated Segmented Events



Stoffregen et al., Motion Segmentation by Motion Compensation, ICCV'19. PDF. Video.

Application 3: Motion Segmentation

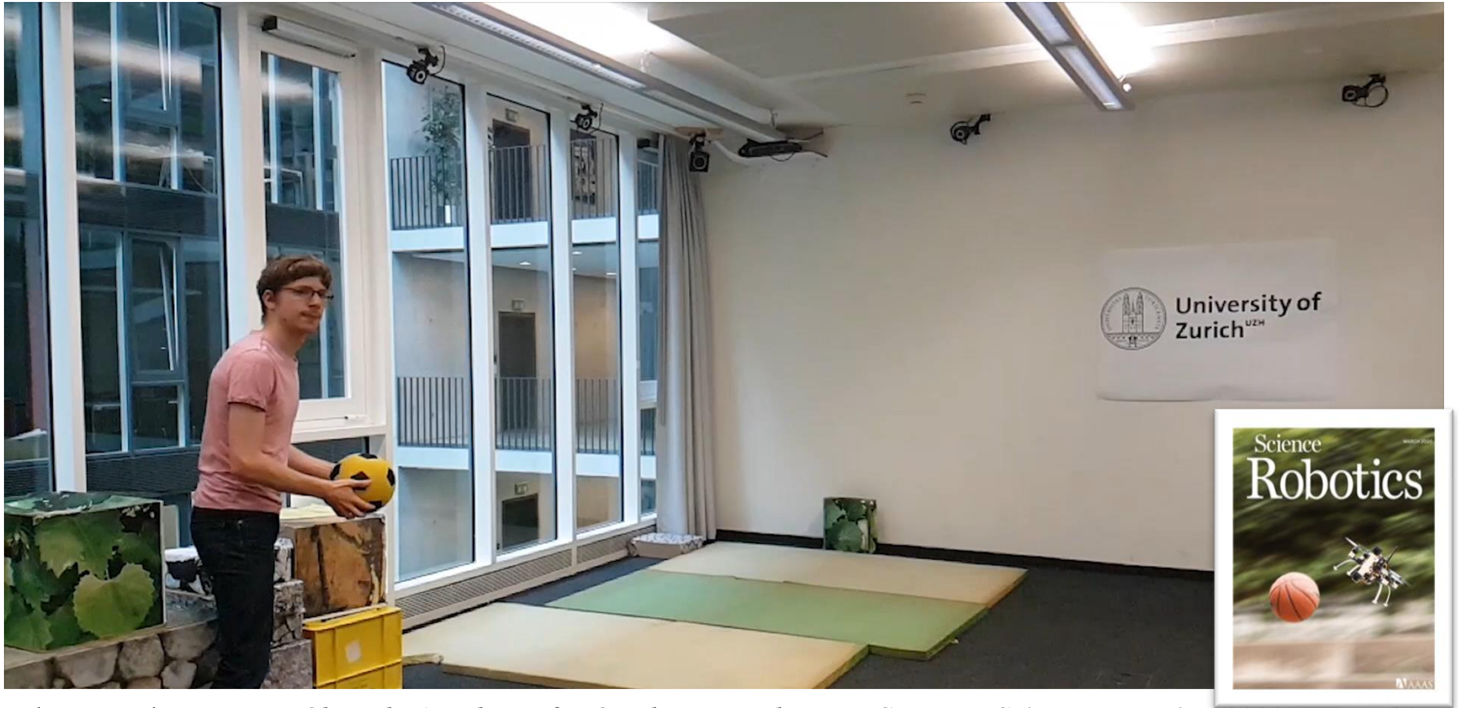


Stoffregen et al., Motion Segmentation by Motion Compensation, ICCV’19. PDF. Video.

Application 5:
Drone Dodging Dynamic Obstacles

Event-based Dynamic Obstacle Detection & Avoidance

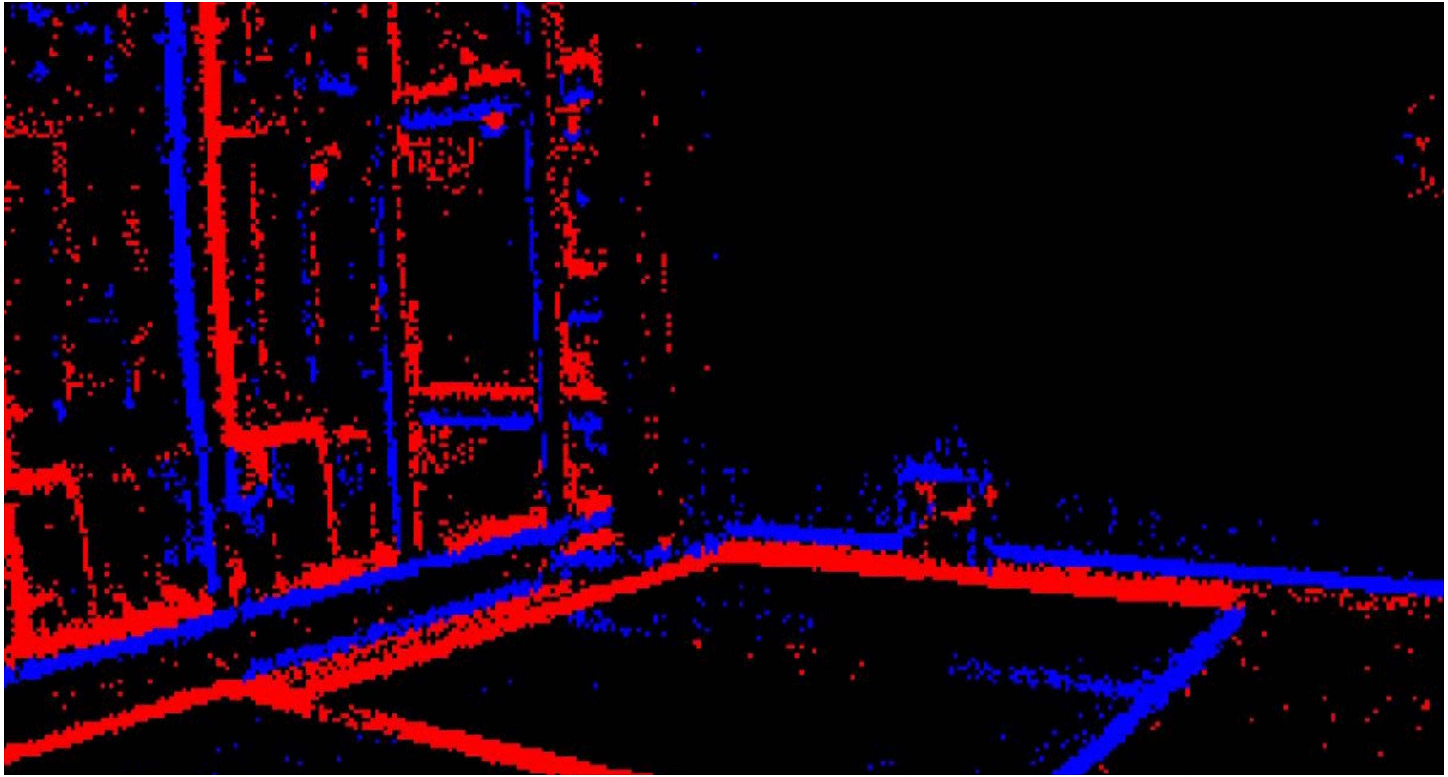
- Works with relative speeds of up to **10 m/s**
- Perception latency: **3.5 ms**



Falanga et al., *Dynamic Obstacle Avoidance for Quadrotors with Event Cameras*, **Science Robotics,** 2020. PDF. Video

Falanga et al. *How Fast is too fast? The role of perception latency in high speed sense and avoid*, RAL'19. PDF. Video

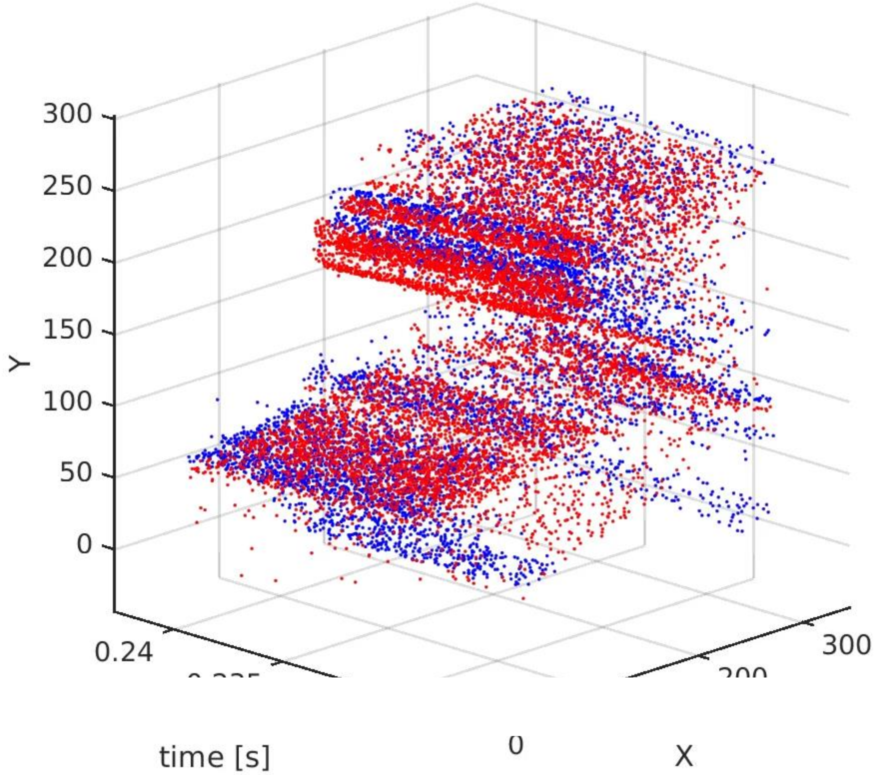
Another challenge: How can we separate events triggered by ego-motion from events triggered by the moving object?



Falanga et al., *Dynamic Obstacle Avoidance for Quadrotors with Event Cameras*, **Science Robotics,** 2020. PDF. Video

Falanga et al. *How Fast is too fast? The role of perception latency in high speed sense and avoid*, RAL'19. PDF

Idea: Warp spatio-temporal volume of events to maximize contrast of the resulting image: **Static objects will appear sharp, while moving ones will appeared blurred.**



Falanga et al., *Dynamic Obstacle Avoidance for Quadrotors with Event Cameras*, **Science Robotics,** 2020. PDF. Video

Falanga et al. *How Fast is too fast? The role of perception latency in high speed sense and avoid*, RAL'19. PDF

Uses Rino event camera from Insightness

Falanga et al., *Dynamic Obstacle Avoidance for Quadrotors with Event Cameras*, **Science Robotics,** 2020. PDF. Video

Falanga et al. *How Fast is too fast? The role of perception latency in high speed sense and avoid*, RAL'19. PDF

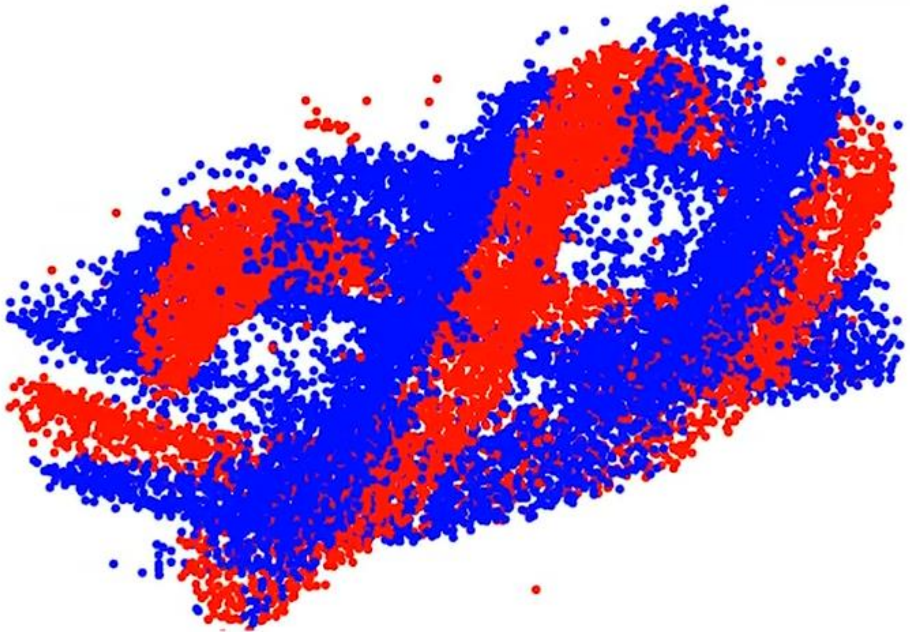
Task 4: Deep-learning based methods

Learning with Event Cameras

- **Synchronous**, **Dense**, Artificial Neural Networks (ANNs) designed for standard images
- **Asynchronous**, **Sparse** ANNs
- **Asynchronous**, **Spiking** Neural Networks (SNNs)

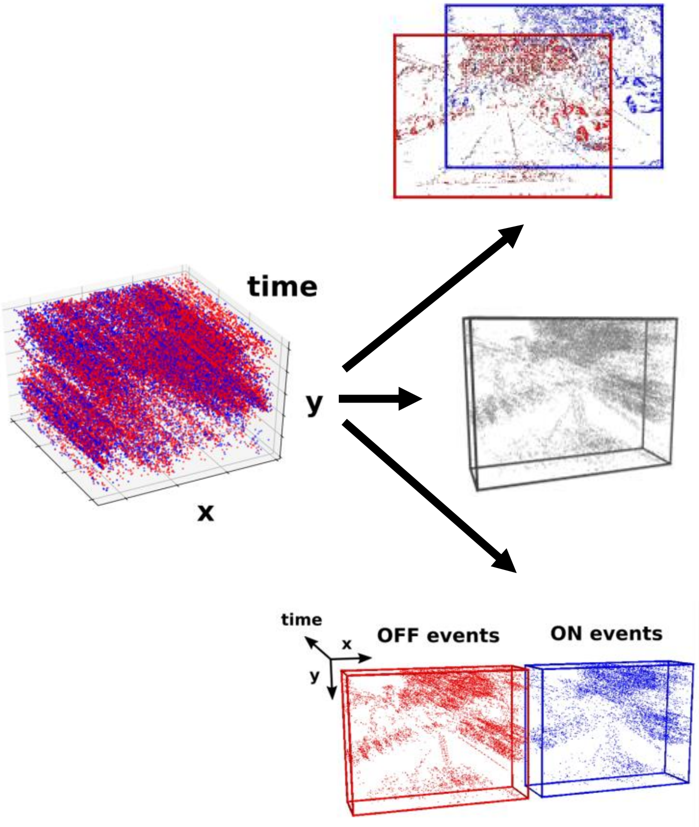
Task 4.1: Event Representation for network?

Challenges: How do we pass sparse events into a convolutional neural network designed for standard images?



Video from Zhu et al. (link)

Task 4.1: Event Representation for network?



[Maqueda CVPR’18], [Zhu’RSS’18]

- **Aggregate positive and negative events into separate channels**
- Discards temporal information

[Zhu ECCVW’18], [Rebecq, CVPR’19], [Zhu, CVPR’19]

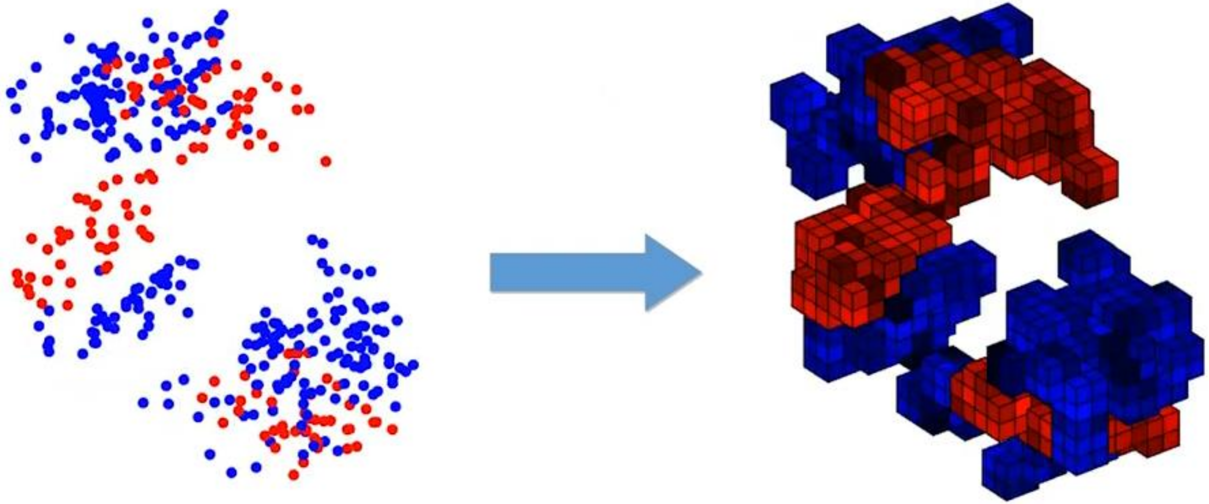
- **Represent events in space-time into a 3D voxel grid (x,y,t)**
- Each voxel contains sum of ON and OFF events falling within the voxel
- Preserves temporal information but discards polarity information

[Gehrig, ICCV’19]

- **Represent events in space-time as a 4D Event Spike Tensor (x,y,t,p)**
- Polarity information is preserved

Task 4.1: Event Representation for network?

Discretized 3D volume (x,y,t): events are inserted into the volume with trilinear interpolation, resulting in minimal loss in resolution

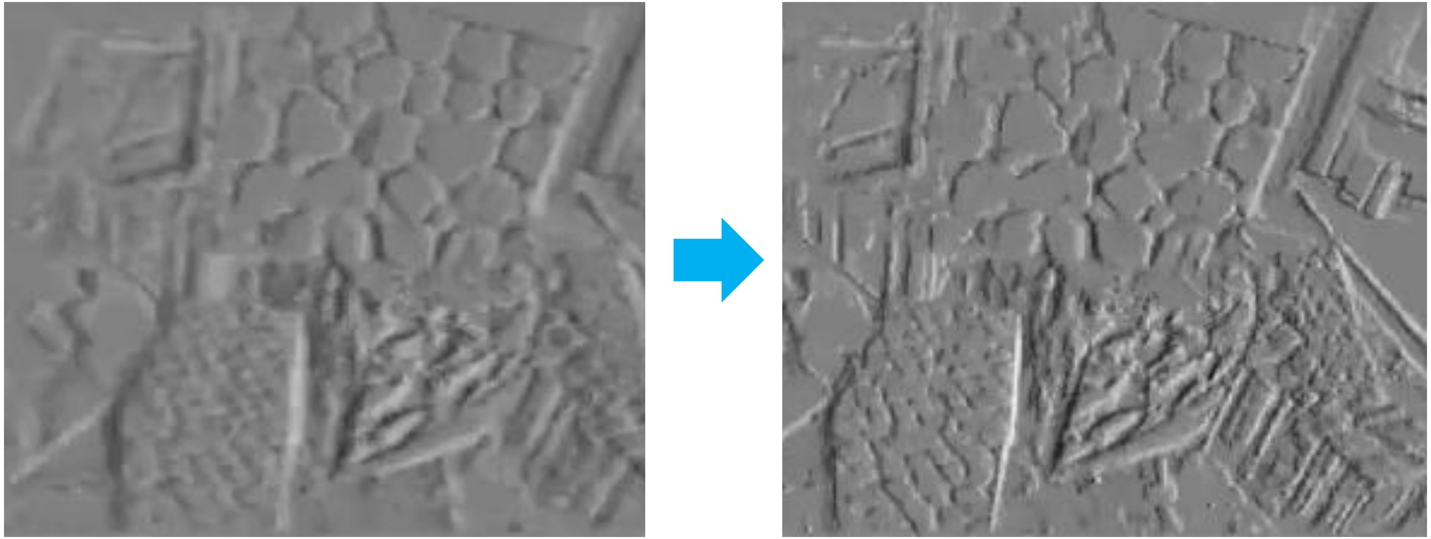


Video from [Zhu et all, CVPR'19]

[Zhu, ECCVW'18], [Zhu, CVPR'19], [Gehrig, ICCV'19], [Rebecq, CVPR'19]

Task 4.2: Focus as Loss Function for Unsupervised Learning

Focus used as loss: maximize sharpness of the aggregated event image.



Video from here

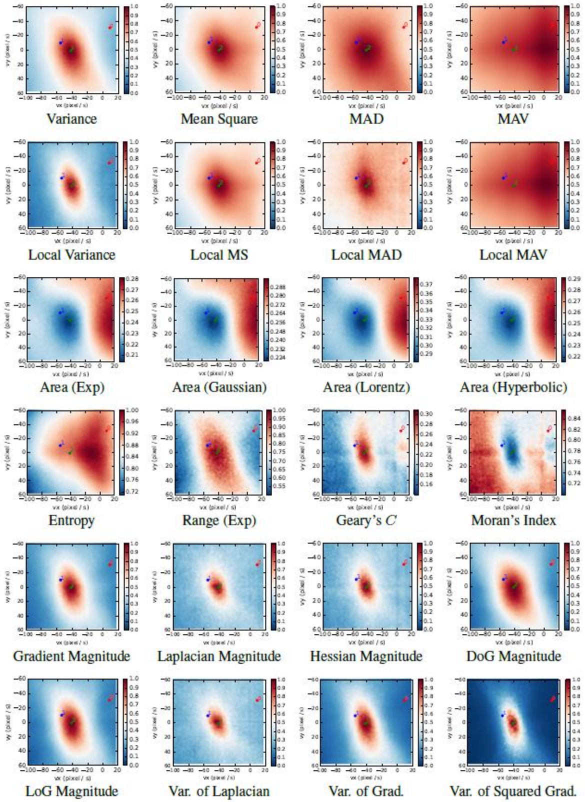
Zhu, Unsupervised Event-based Learning of Optical Flow, Depth and Egomotion, CVPR 19. PDF

Gallego et al., Focus Is All You Need: Loss Functions for Event-based Vision, CVPR19, PDF.

Focus as Loss Function for Unsupervised Learning

- We proposed and benchmarked **22 focus loss functions**

- **Focus is the “data fidelity” term**



Variance, Mean Square, MAD, MAV

Local Variance, Local MS, Local MAD, Local MAV

Area (Exp), Area (Gaussian), Area (Lorentz), Area (Hyperbolic)

Entropy, Range (Exp), Geary’s *C*, Moran’s Index

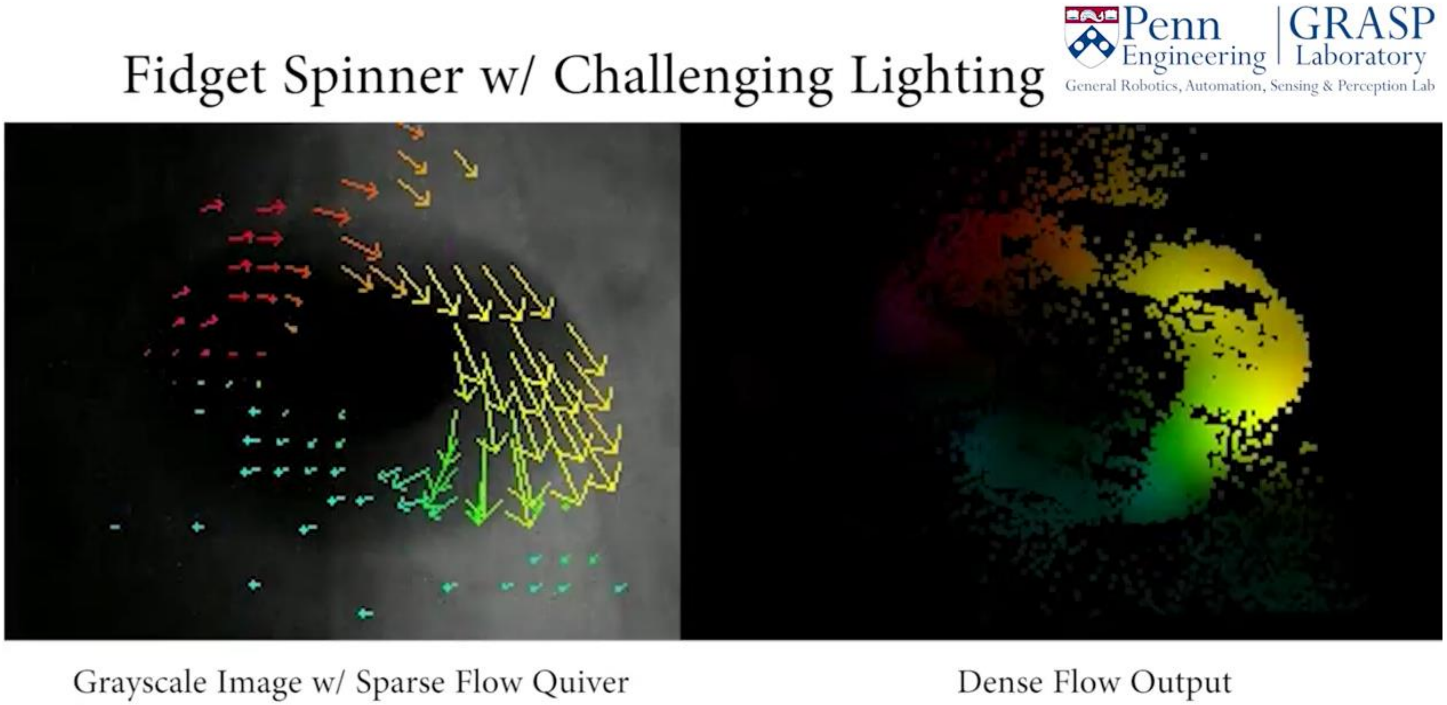
Gradient Magnitude, Laplacian Magnitude, Hessian Magnitude, DoG Magnitude

LoG Magnitude, Var. of Laplacian, Var. of Grad., Var. of Squared Grad.

Gallego et al., Focus Is All You Need: Loss Functions for Event-based Vision, CVPR19, PDF.

Application: Unsupervised Learning of Optical Flow, Depth and Ego Motion

Focus used as loss: maximize sharpness of the aggregated event image.



Fidget Spinner w/ Challenging Lighting

Grayscale Image w/ Sparse Flow Quiver

Dense Flow Output

Zhu et al., Unsupervised Learning of Optical Flow, Depth and Ego Motion, CVPR’19

Task 4.3: How to reconstruct frame-based images?

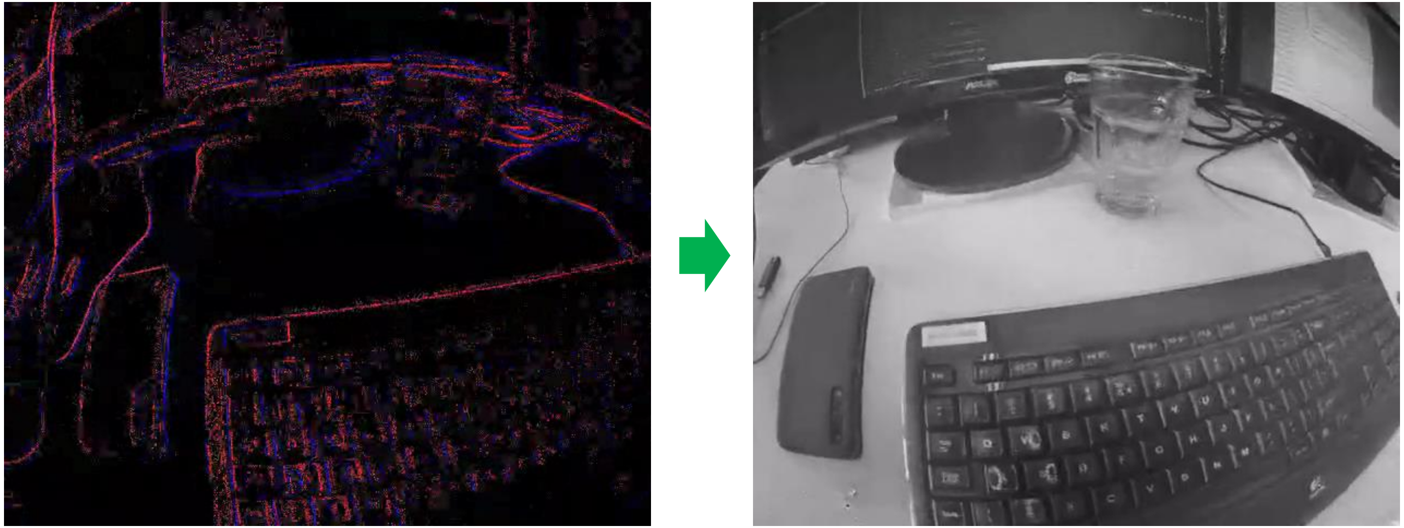
Application:
Learning High-speed and HDR Video Rendering from an Event Camera

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., "Events-to-Video: Bringing Modern Computer Vision to Event Cameras", CVPR19. PDF Video.
Rebecq et al., "High Speed and High Dynamic Range Video with an Event Camera", PAMI, 2019. PDF Video Code

Task 4.3: How to reconstruct frame-based images?



Events

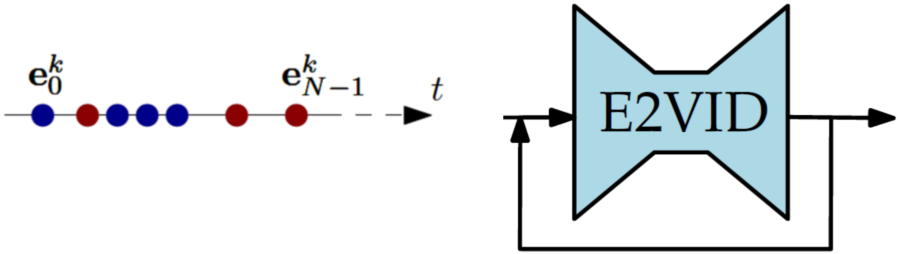
Reconstructed image from events (Samsung DVS)

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.
Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video
Code

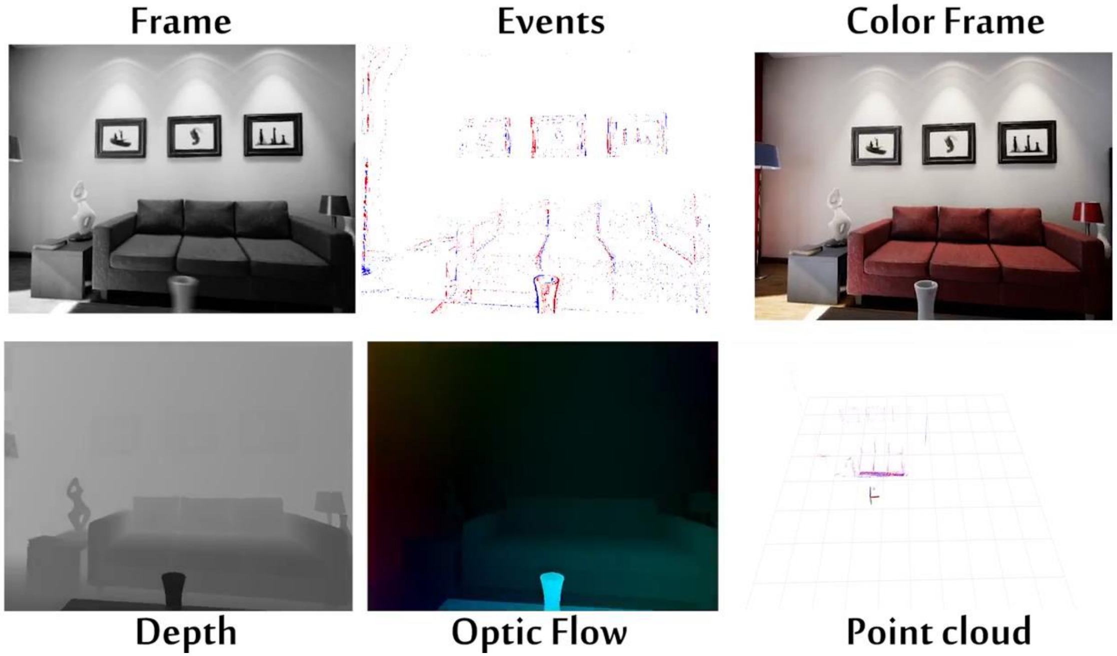
Overview

- **Recurrent neural network** (main module: Unet)
- Input: last reconstructed frame + **sequences of *event tensors*** (spatio-temporal 3D voxels grid: each voxel contains sum of ON and OFF events falling within the voxel)
- Network processes **last *N* events** (10,000)
- **Trained in simulation only** (without seeing a single real image) (we used our event camera simulator: http://rpg.ifi.uzh.ch/esim.html)



Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.
Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Acquire Training Data: From the Event Camera Simulator



Event Camera Simulator (ESIM): http://rpg.ifi.uzh.ch/esim.html

Rebecq, ESIM: an Open Event Camera Simulator, CORL’18. PDF, YouTube, Project page

High Speed Video Reconstruction Results

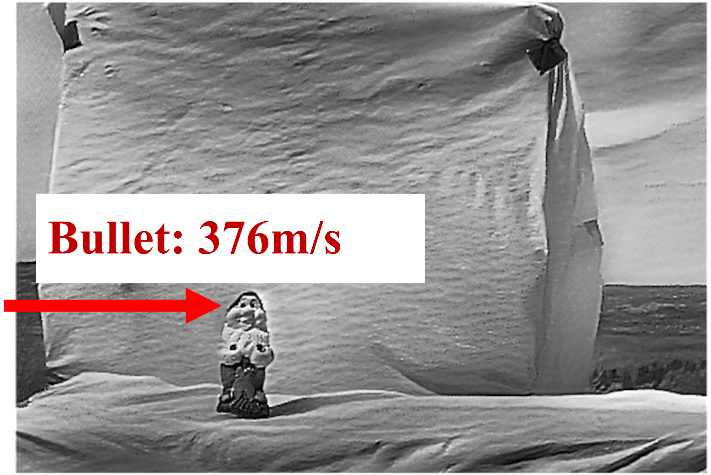
Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.
Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Bullet shot by a gun (376m/s (=1,354km/h)

Recall: trained in simulation only!

Huawei P20 Pro (240 FPS)



Our reconstruction (5400 FPS)

We used Samsung DVS

Real time

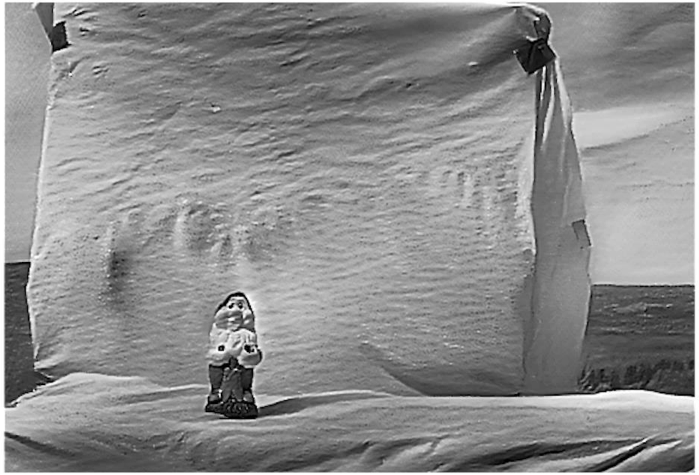
Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.
Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Bullet shot by a gun (1,300 km/h)

Recall: trained in simulation only!

Huawei P20 Pro (240 FPS)



Our reconstruction (5400 FPS)

We used Samsung DVS

100 x slow motion

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

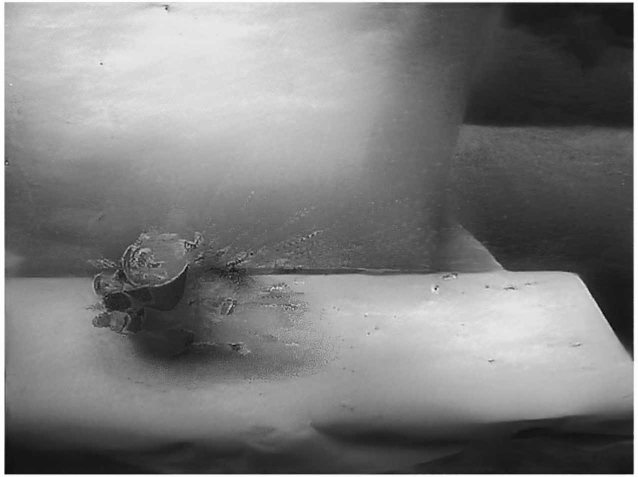
Rebecq et al., "Events-to-Video: Bringing Modern Computer Vision to Event Cameras", CVPR19. PDF Video.

Rebecq et al., "High Speed and High Dynamic Range Video with an Event Camera", PAMI, 2019. PDF Video Code

Bullet shot by a gun (1,300 km/h)

Recall: trained in simulation only!

Huawei P20 Pro (240 FPS)



Our reconstruction (4800 FPS)

We used Samsung DVS

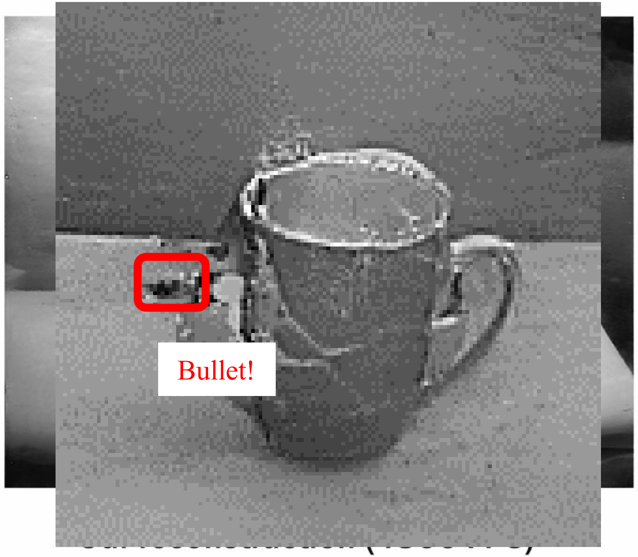
100 x slow motion

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.
Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Bullet shot by a gun (1,300 km/h)

Recall: trained in simulation only!



Bullet!

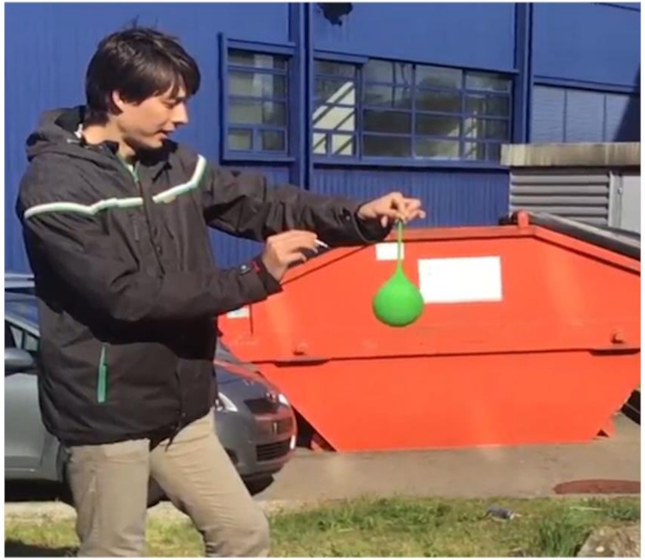
We used Samsung DVS

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

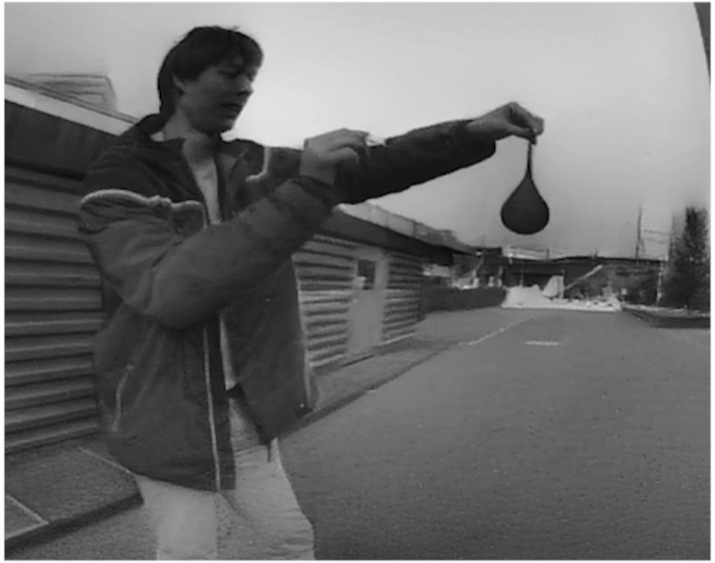
Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.
Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Popping a water balloon

Recall: trained in simulation only!



Apple iPad (120 FPS)



Our reconstruction (4800 FPS)

* different sequences, recorded in identical conditions

We used Samsung DVS

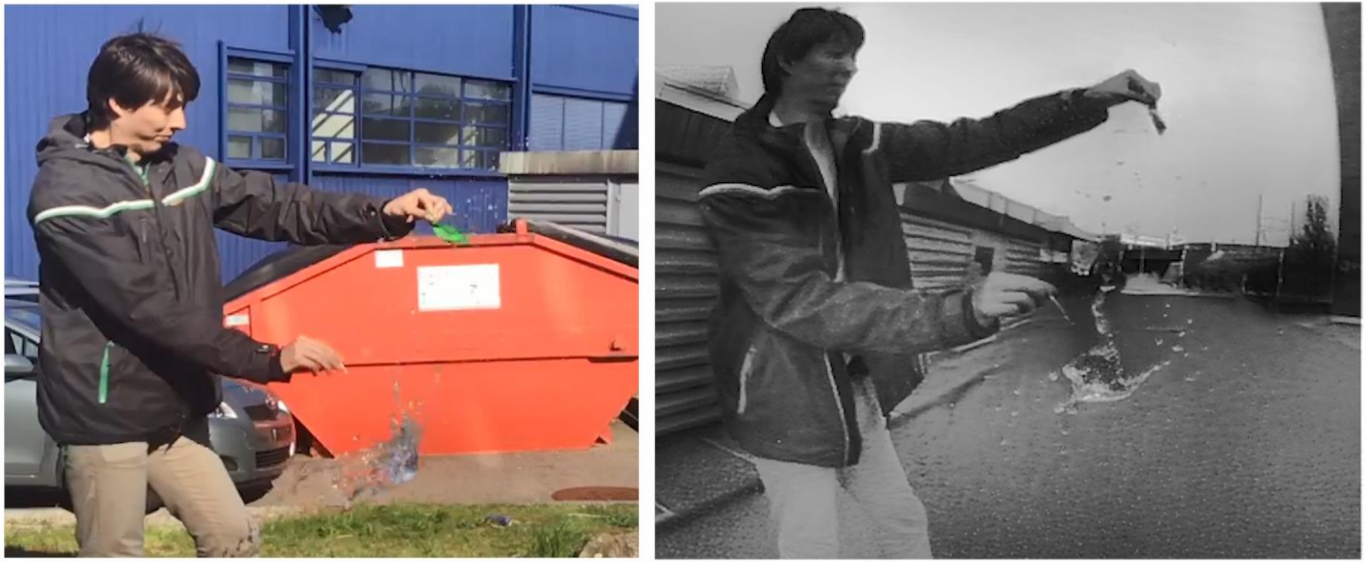
Real time

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., "Events-to-Video: Bringing Modern Computer Vision to Event Cameras", CVPR19. PDF Video.
Rebecq et al., "High Speed and High Dynamic Range Video with an Event Camera", PAMI, 2019. PDF Video Code

Popping a water balloon

Recall: trained in simulation only!



Apple iPad (120 FPS)

Our reconstruction (4800 FPS)

* different sequences, recorded in identical conditions

100 x slow motion

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.

Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

HDR Video Reconstruction Results

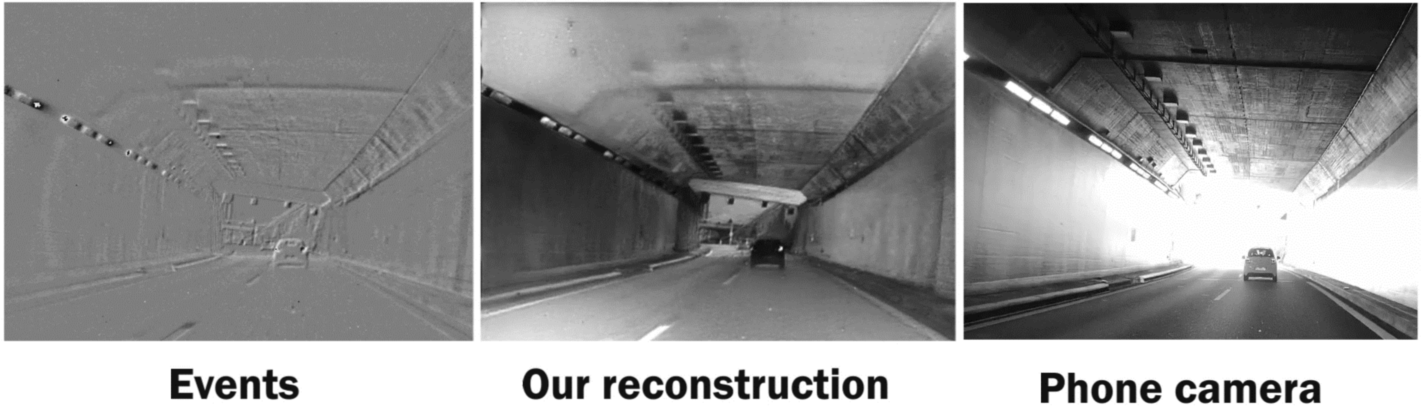
Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.

Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

HDR Video: Driving out of a tunnel

Recall: trained in simulation only!



Events

Our reconstruction

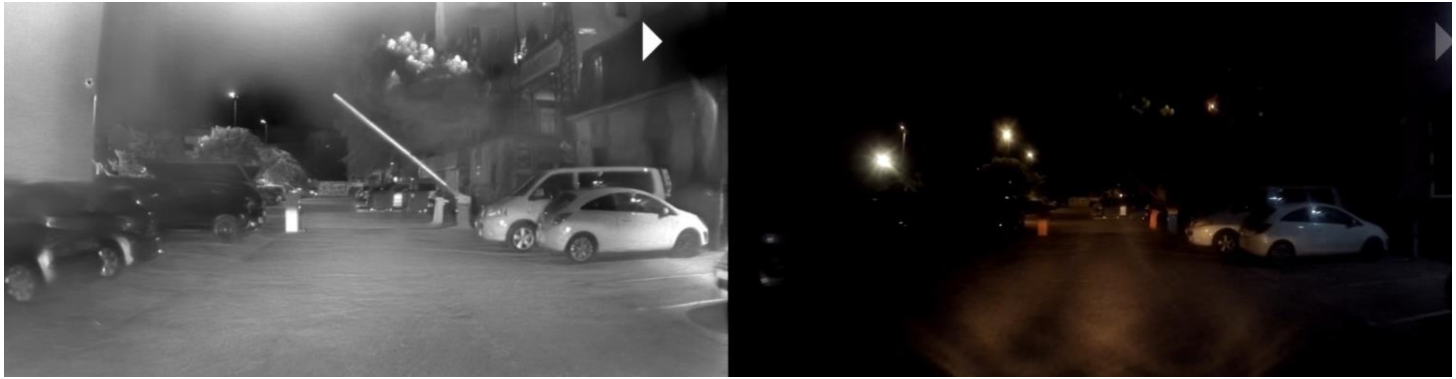
Phone camera

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.

Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Recall: trained in simulation only!



Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

PDF Video

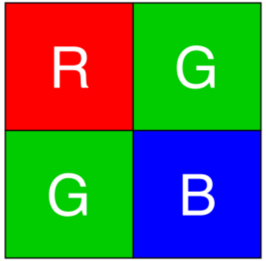
PDF Video

Code

Color video reconstruction

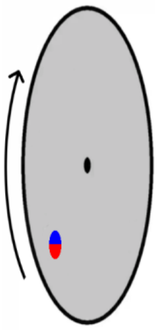
Color events

DAVIS346 Red Color

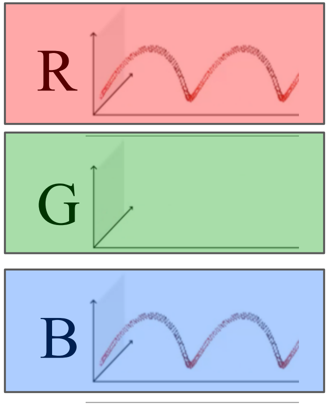


R	G
G	B

Bayer pattern



Input

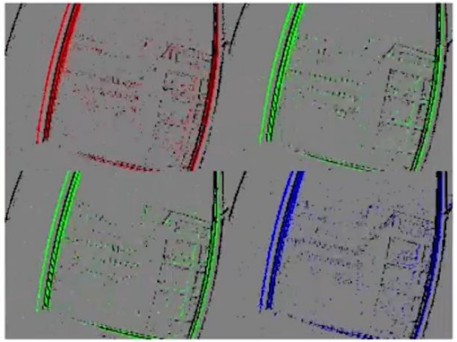


Output

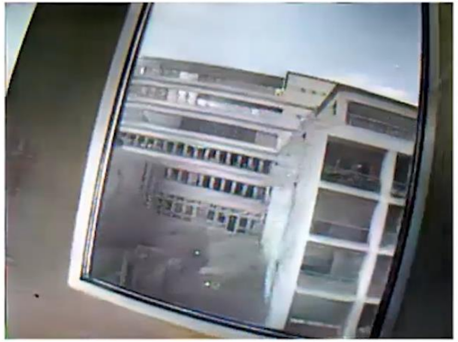
- Each pixel is sensitive to **red, green or blue** light.
- Transmits **brightness changes** in each color channel

Taverni et al., Front and back illuminated Dynamic and Active Pixel Vision Sensors comparison, TCS’18

Color Event Camera Reconstruction (HDR)



Color events



Our reconstruction

Color frame

Color Event Camera Datasets: http://rpg.ifi.uzh.ch/CED.html

Scheerlinck, Rebecq, Stoffregen, Barnes, Mahony, Scaramuzza
CED: Color Event Camera Dataset. CVPRW, 2019. PDF YouTube Dataset

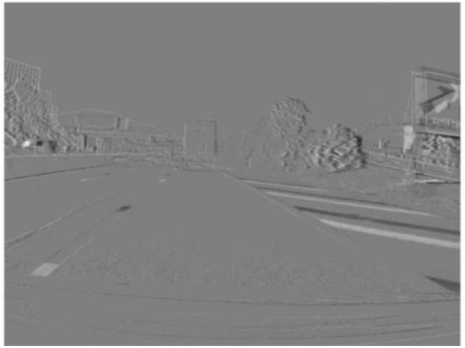
Downstream Applications:
What if we input the reconstructed frames to state of the art ML algorithms?

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

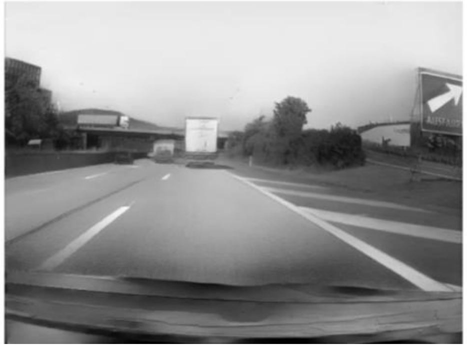
Rebecq et al., "Events-to-Video: Bringing Modern Computer Vision to Event Cameras", CVPR19. PDF Video.

Rebecq et al., "High Speed and High Dynamic Range Video with an Event Camera", PAMI, 2019. PDF Video Code

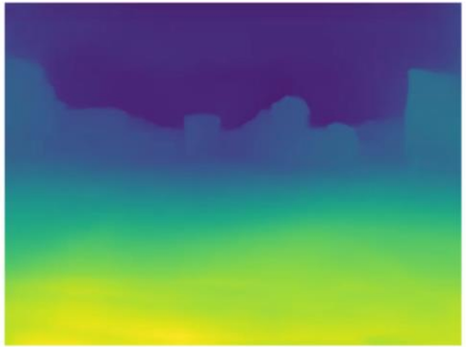
Monocular Depth Estimation



Events



Our reconstruction



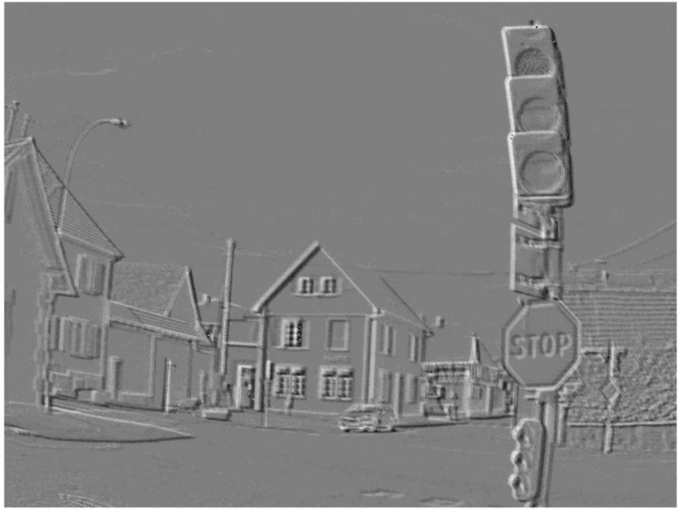
Monocular depth estimation (Megadepth) applied on the reconstructed frames

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

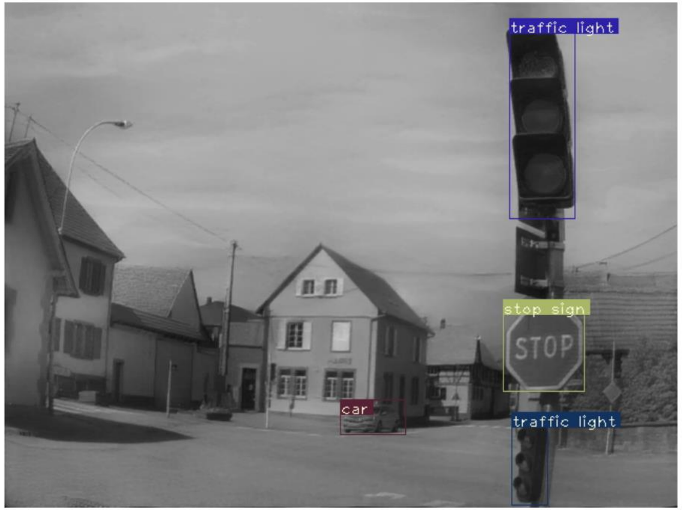
Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.

Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Object detection



Events



Our reconstruction + object detections (YOLOv3)

Code & datasets: https://github.com/uzh-rpg/rpg_e2vid

Rebecq et al., “Events-to-Video: Bringing Modern Computer Vision to Event Cameras”, CVPR19. PDF Video.

Rebecq et al., “High Speed and High Dynamic Range Video with an Event Camera”, PAMI, 2019. PDF Video Code

Does it mean that in order to use event cameras we must first reconstruct an image?

NO!

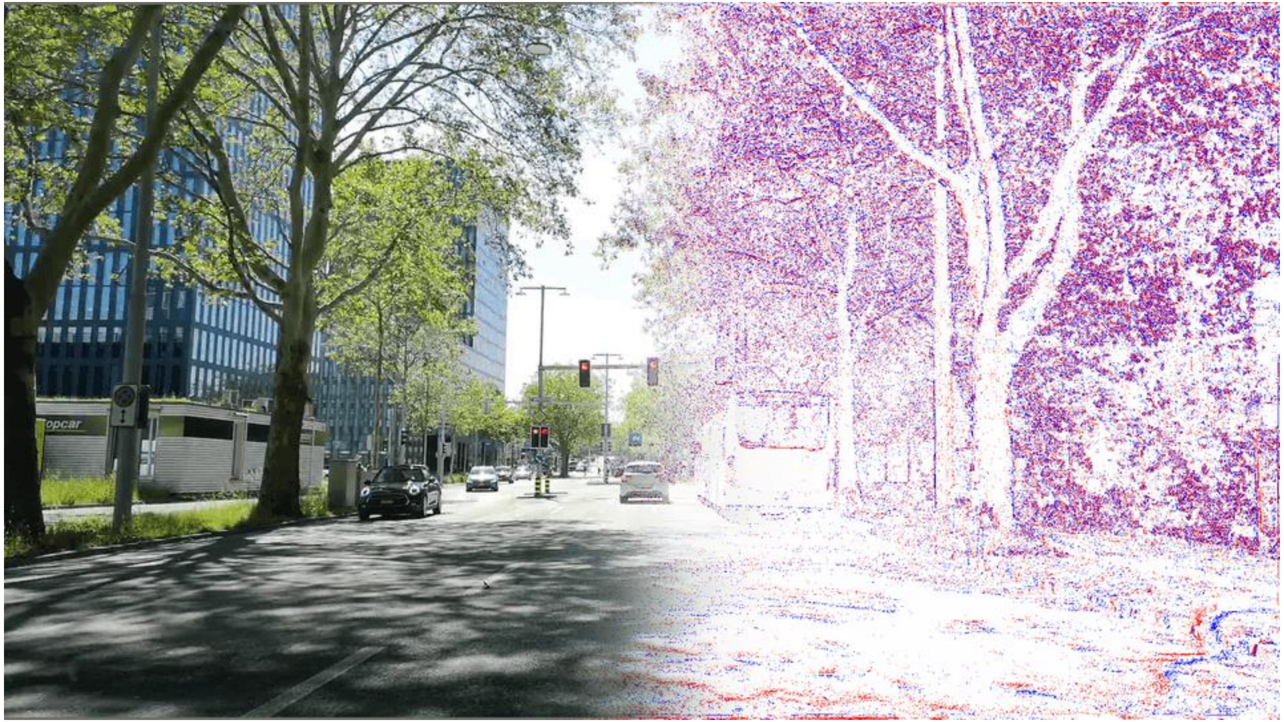
These results were only to show that it should be possible to design **more efficient** algorithms that process events **end-to-end without passing through image reconstruction**!

However, to design end-to-end approaches for event cameras, **we need more data!** But event cameras are new, so there is a **shortage of large scale datasets** compared to standard cameras!

Is it possible to **recycle existing** large-scale **video datasets** recorded with standard cameras **for event cameras**?

Task 4.4: How to get more event data?

Idea: convert Standard videos to events!

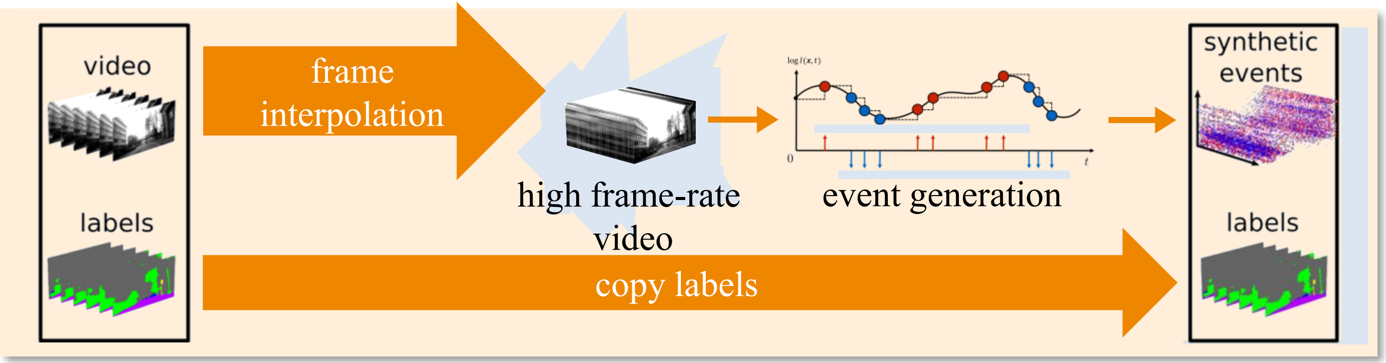


Code: https://github.com/uzh-rpg/rpg_vid2e

Gehrig et al., “Video to Events: Recycling Video Datasets for Event Cameras”, CVPR20. PDF Video Code.

How to we convert a standard video to events?

- **Typical video has a low temporal resolution and needs to upsampled first**
- We use off-the-shelf upsampling techniques (**Super SloMo** [Jiang, CVPR'18])
- Event generation using our event camera simulator (http://rpg.ifi.uzh.ch/esim.html)
- **Noise free simulation**. **We randomize the contrast sensitivity**



video

labels

frame interpolation

high frame-rate video

event generation

synthetic events

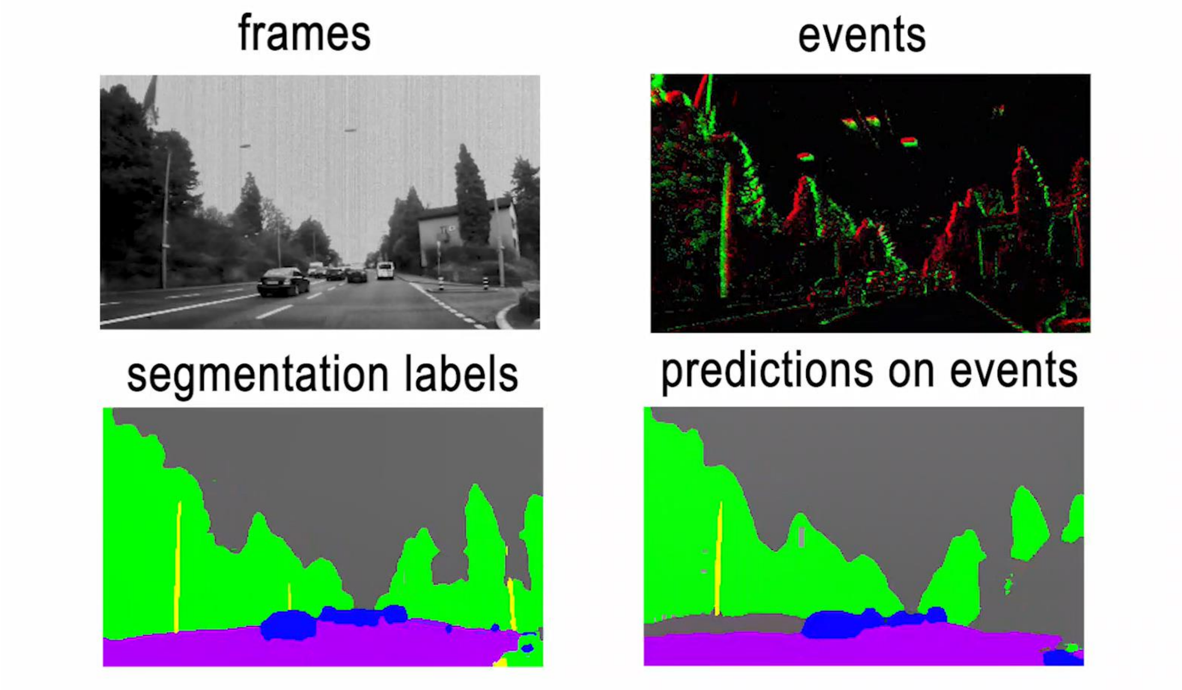
labels

copy labels

Code: https://github.com/uzh-rpg/rpg_vid2e

Gehrig et al., "Video to Events: Recycling Video Datasets for Event Cameras", CVPR20. PDF Video Code.

Experiments on Semantic Segmentation

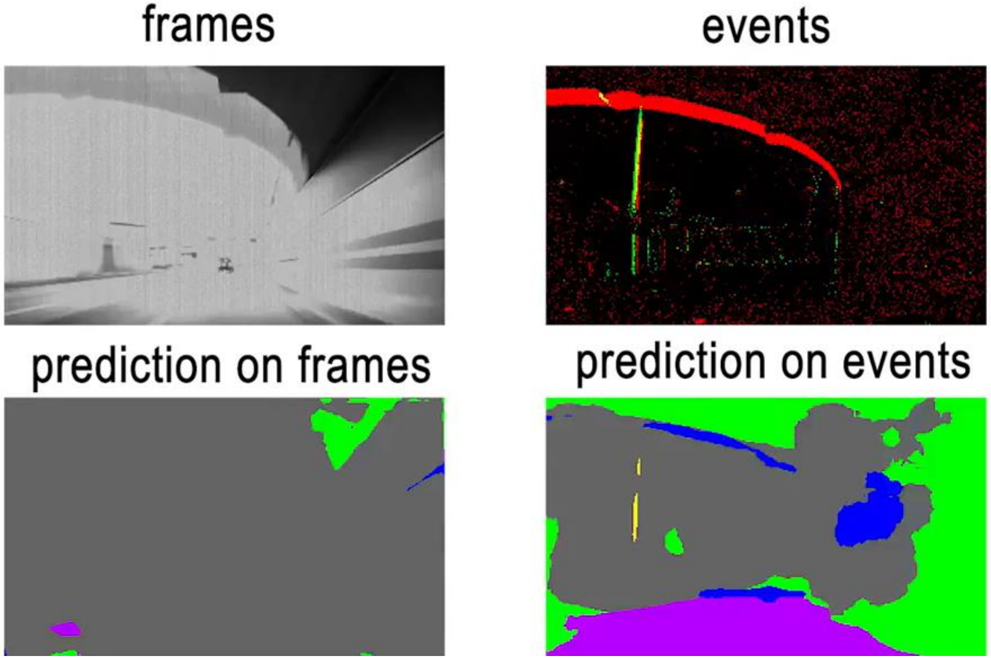


Code: https://github.com/uzh-rpg/rpg_vid2e

Method: Gehrig et al., “Video to Events: Recycling Video Datasets for Event Cameras”, CVPR20. PDF Video Code.

Dataset: Binas et al., “DDD17: End-To-End DAVIS Driving Dataset”, ICMLW’17. Dataset

Generalization to challenging HDR scenario



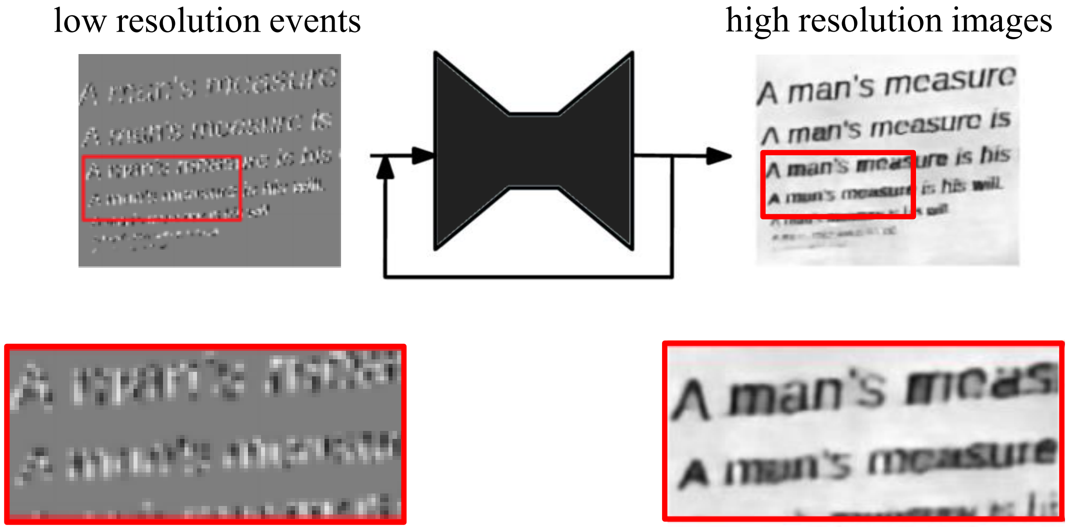
Code: https://github.com/uzh-rpg/rpg_vid2e

Gehrig et al., "Video to Events: Recycling Video Datasets for Event Cameras", CVPR20. PDF Video Code.
Binas et al., "DDD17: End-To-End DAVIS Driving Dataset", ICMLW'17.

Computational Photography Revolution

Task 4.5: Event-based super-resolution

- **Given low-resolution events as input, reconstruct a high-resolution image**
- For standard images, the spatial resolution **is fixed and cannot change**
- For event data**, high spatial resolution may be hidden in the temporal** resolution of the data. Networks can exploit this!



Mostafavi I. et al. , “Learning to Super Resolve Intensity Images from Events”, CVPR20
Wang et al. “EventSR: From Events to Image Reconstruction, Restoration, and Super-Resolution”, CVPR20

Learning with Event Cameras

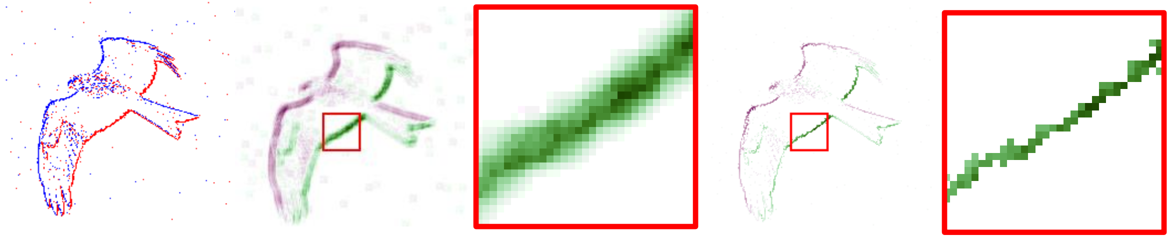
- **Synchronous**, **Dense**, Artificial Neural Networks (ANNs) designed for standard images
- **Asynchronous**, **Sparse** ANNs
- **Asynchronous**, **Spiking** Neural Networks (SNNs)

Adapting Neural Networks To Event-based Data

Messikommer et al., “Event-based Asynchronous Sparse Convolutional Networks”, ECCV’20. PDF Video.

Tasks 4.6: Event-based Asynchronous Sparse Convolutional Networks

- Convolutional neural networks process events as dense image
- However, event-data is inherently sparse and asynchronous, meaning that wasteful computation is being performed
- We can save computation by adopting **sparse convolutions** [Graham CVPR'18] which only compute the convolution at active pixels



input events

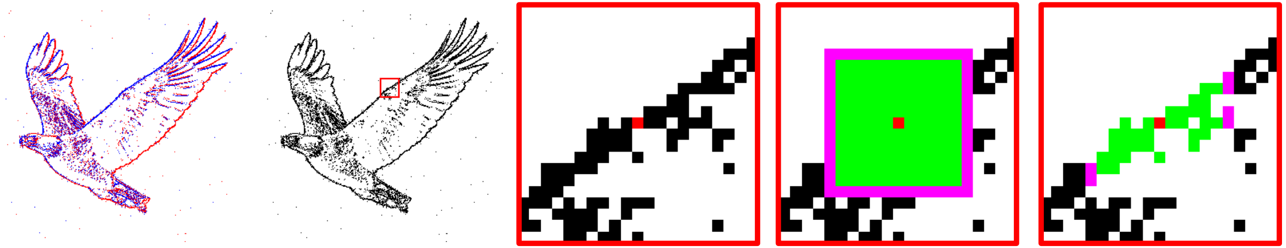
dense network activations

sparse network activations [Graham CVPR'18]

Messikommer et al., "Event-based Asynchronous Sparse Convolutional Networks", ECCV'20. PDF Video.

Tasks 4.6: Event-based Asynchronous Sparse Convolutional Networks

- For each new event we do not have to update the full network layers. Just the pixels which are in the receptive field of the pixel which triggered the event.
- For regular convolutions this receptive field grows quadratically with the depth of the network. However, for sparse convolutions it grows much more slowly.
- This growth rate of the receptive field is related to the **fractal dimension**, which is an intrinsic property of event data.



input events

active pixels

update at a single pixel

receptive field for standard CNN

receptive field for sparse CNN

Messikommer et al., “Event-based Asynchronous Sparse Convolutional Networks”, ECCV’20. PDF Video.

Learning with Event Cameras

- **Synchronous**, **Dense**, Artificial Neural Networks (ANNs) designed for standard images
- **Asynchronous**, **Sparse** ANNs
- **Asynchronous**, **Spiking** Neural Networks (SNNs)

Task 4.7: Spiking Neural Networks (SNN)

- Common processing units based on **Von Neumann architectures** (**CPU and GPU)** are **inefficient & very power consuming for event-by-event processing** [1]
- There exists very efficient, **specialized hardware for event-by-event inference**: **IBM TrueNorth** [1], **Intel Loihi** [2], DynapSE & Speck (AiCTX) [3]
- Promising for Robotics, IoT, VR/AR/MR
  1. Low power
  2. Low latency
  3. Leverage event-based sensing
- **Promise ultra-low carbon footprint!**

[1]: Merolla et. al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*. 2014
[2]: Davies M. et. al. Loihi: A neuromorphic manycore processor with on-chip learning. IEEE Micro. 2018
[3]: Moradi S. et. al.. A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors. IEEE transactions on biomedical circuits and systems. 2017. https://aictx.ai/

The Cost of Current Computer Technologies is Not Sustainable

- **In 2017, > 10 zettabytes** of data were

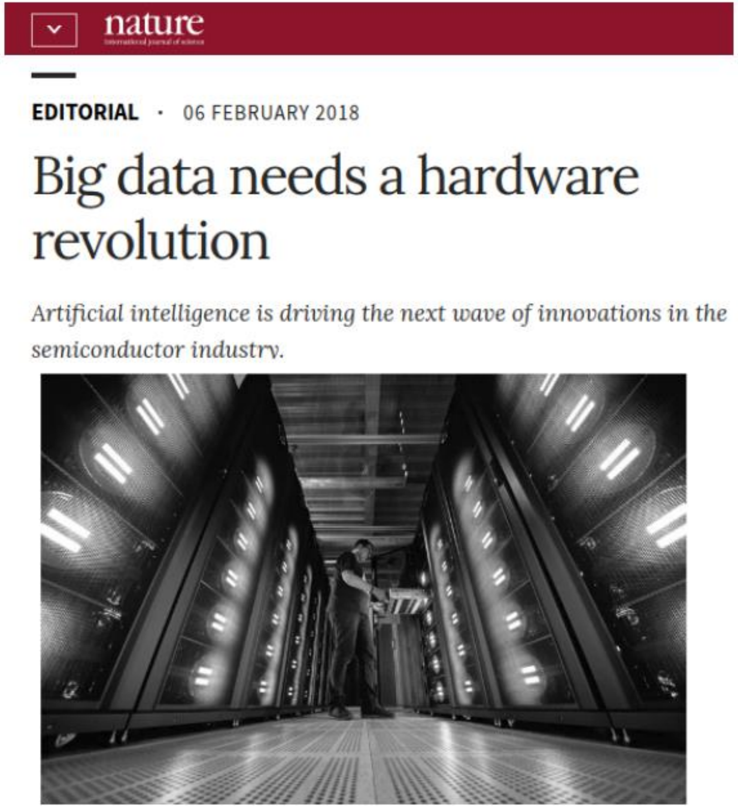
- **IT infrastructures and consumer ele**

  electricity supply.

- By **2025**, over **50 billion of Internet-c** nected.

- **Over 180 zettabytes of data will be g** to a

  consumption of one-fifth of global elec



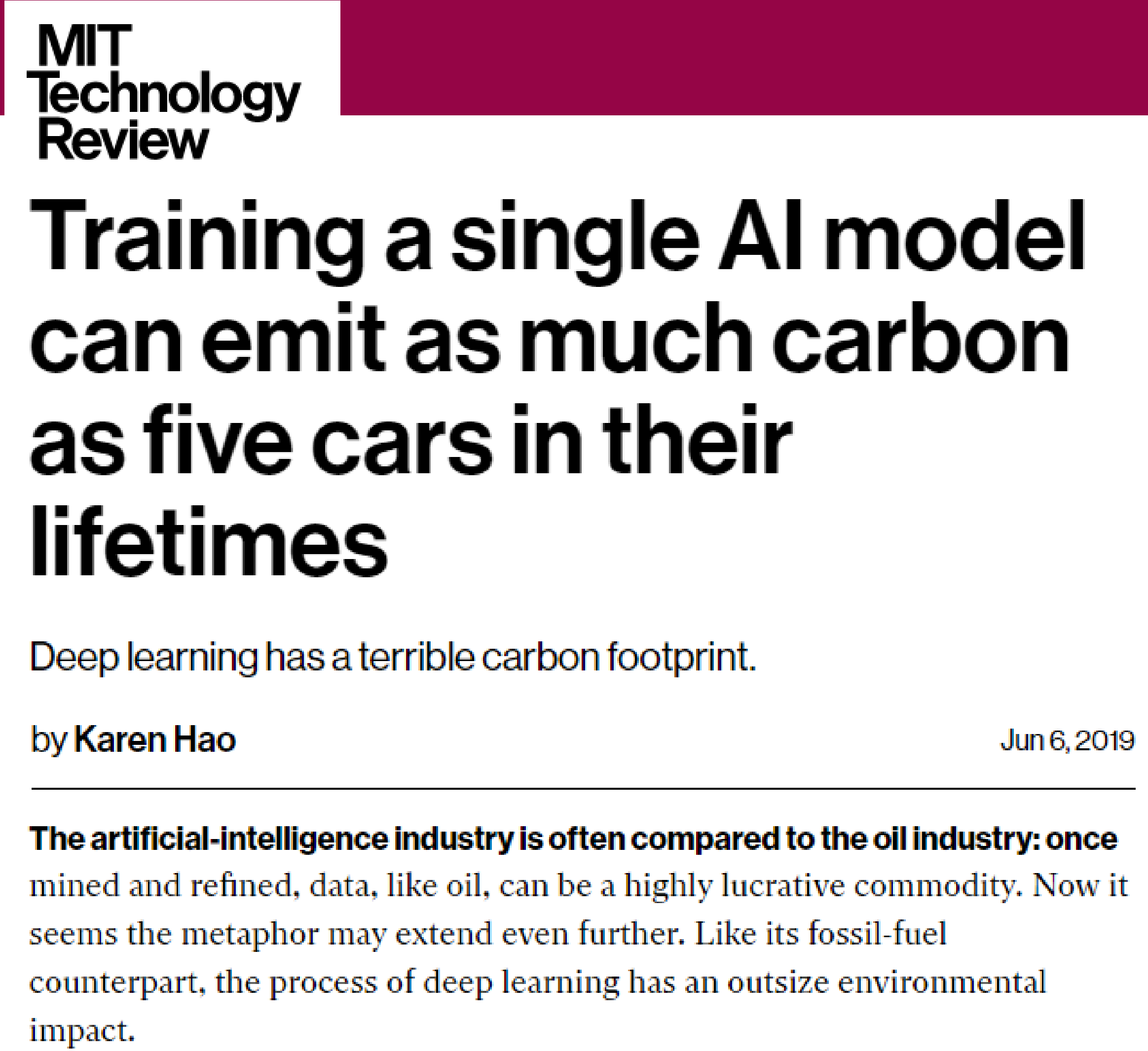
nature

EDITORIAL · 06 FEBRUARY 2018

Big data needs a hardware revolution

Artificial intelligence is driving the next wave of innovations in the semiconductor industry.

"Software companies make headlines but research on computer could bring bigger rewards. "



MIT Technology Review

Training a single AI model can emit as much carbon as five cars in their lifetimes

Deep learning has a terrible carbon footprint.

by Karen Hao

Jun 6, 2019

The artificial-intelligence industry is often compared to the oil industry: once mined and refined, data, like oil, can be a highly lucrative commodity. Now it seems the metaphor may extend even further. Like its fossil-fuel counterpart, the process of deep learning has an outsize environmental impact.

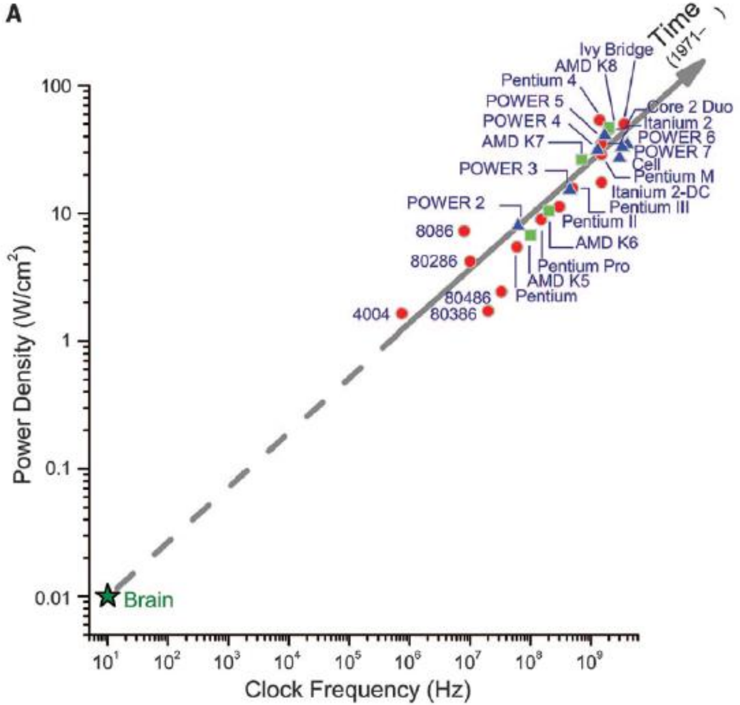
Radical paradigm shift in computer hardware technologies

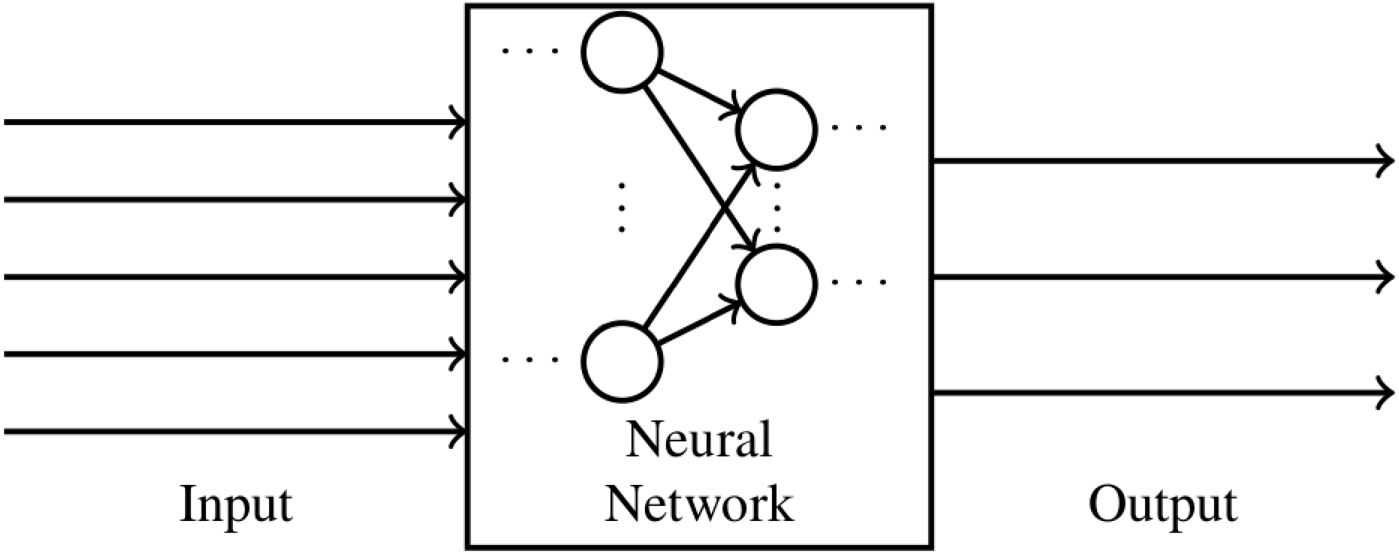
- ➢ Our brain is slow, noisy (**“speed” is no**

- ➢ Massively parallel distributed computat)

- ➢ Real-time interaction with the environn

- ➢ **Complex spatio-temporal pattern re**

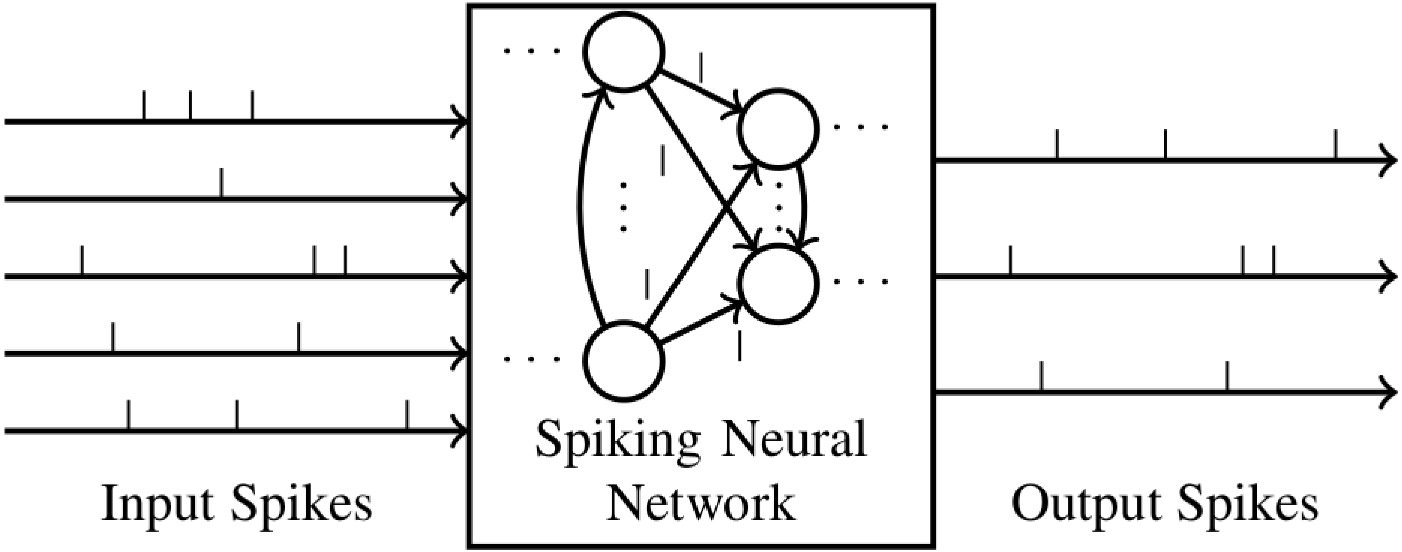




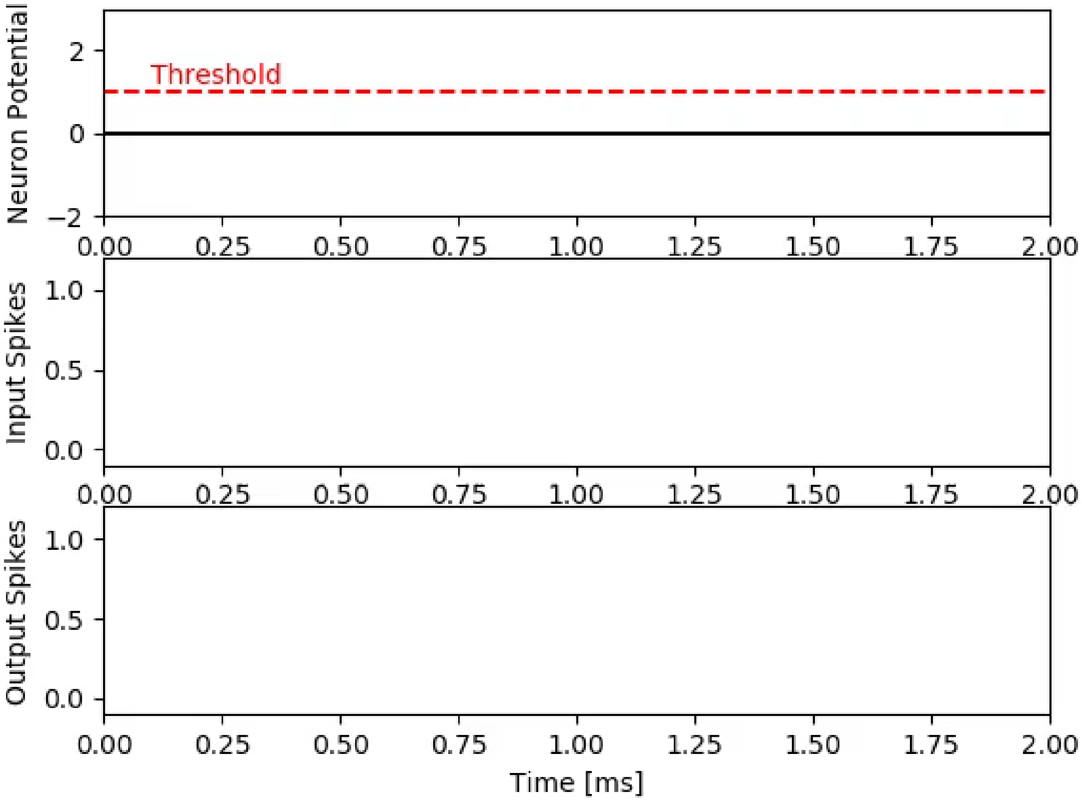
Input

Neural Network

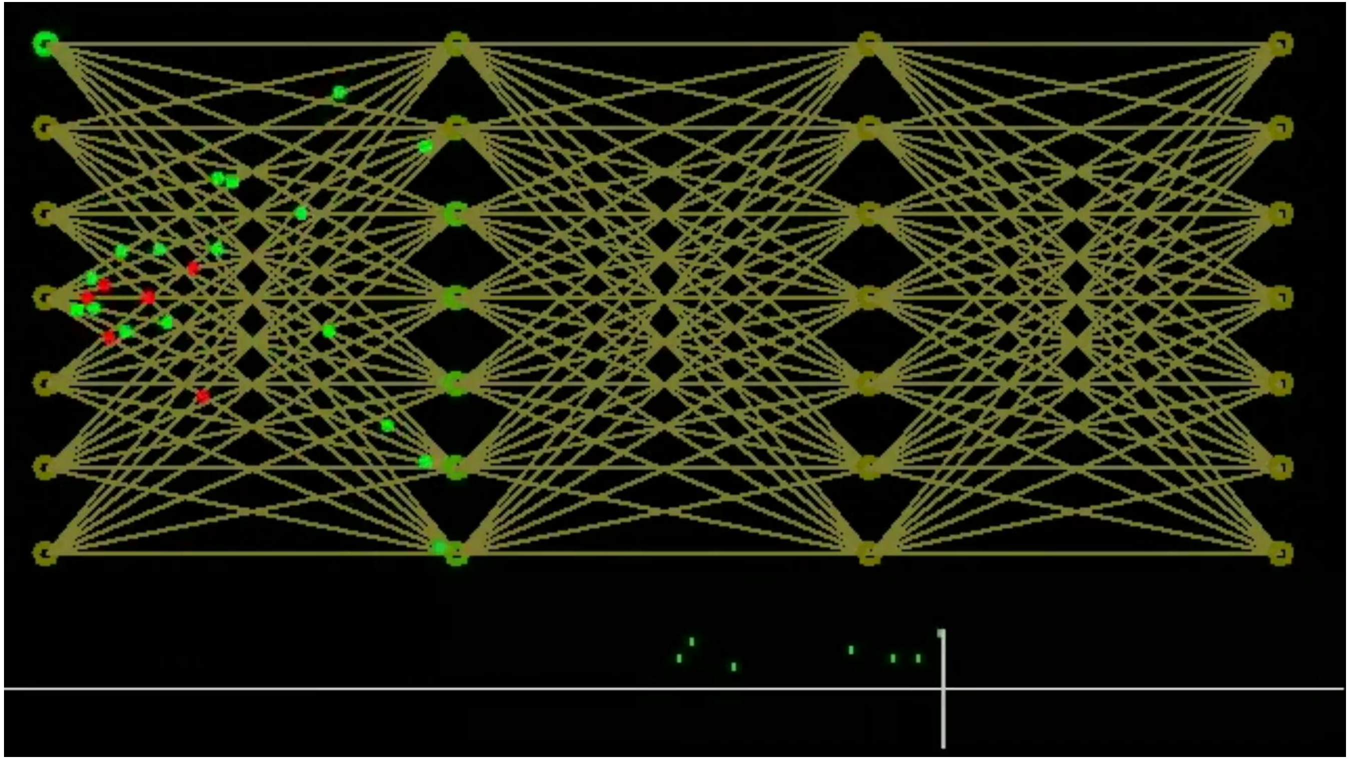
Output


Spiking Neural
Network
Input Spikes
Output Spikes

Model of a Spiking Neuron



Specialized Hardware



Mior Roberto (2010). Spiking Neural Network. https://youtu.be/lhIdisK7akI

Origins of Spiking Neural Networks

- **First model** (integrate-and-fire) of **a spiking neuron in 1907** by Louis Lapicque [1]
- **First computational model for neural networks in 1943** [2]: Neural network research split into biological processes in the brain and the application for artificial intelligence
- First scientific model of biological spike propagation by Hodgkin and Huxley in **1951** [3] (**Nobel Prize in Physiology**)
- A range of more general spiking neuron models are available nowadays [4]

[1]: Lapicque L. Recherches quantitatives sur l'excitation electrique des nerfs traitee comme une polarization. *Journal de Physiologie et de Pathologie Generale*. 1907
[2]: McCulloch WS, Pitts W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*. 1943
[3]: Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*. 1952
[4]: Gerstner W. Time structure of the activity in neural network models. *Physical review*. 1995

SNNs: Current Applications and Demos

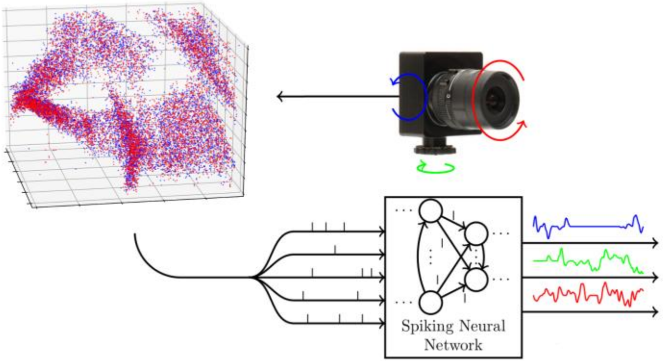
- IBM TrueNorth
  1. Targeting ultra low-power gesture control, video-surveillance and IoT with SNN on **digital processor**
  2. At CVPR 2017 gesture recognition demo (10 gestures)
  3. 96.5 % recognition accuracy
  4. **200 mW power consumption** (event-camera + processing)
- Intel Loihi:
  1. Targeting ultra low-power surveillance and IoT with **SNN** on **analog processor**
- aiCTX (Zurich-based startup):
  1. Targeting ultra low-power surveillance and IoT with **SNN** on **analog processor**
  2. At CES'19 and CVPR'19, they demonstrated face recognition from event data on an SNN processor; total power consumption: **1mW**

[1]: https://aictx.ai/
[2]: Merolla et. al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*. 2014

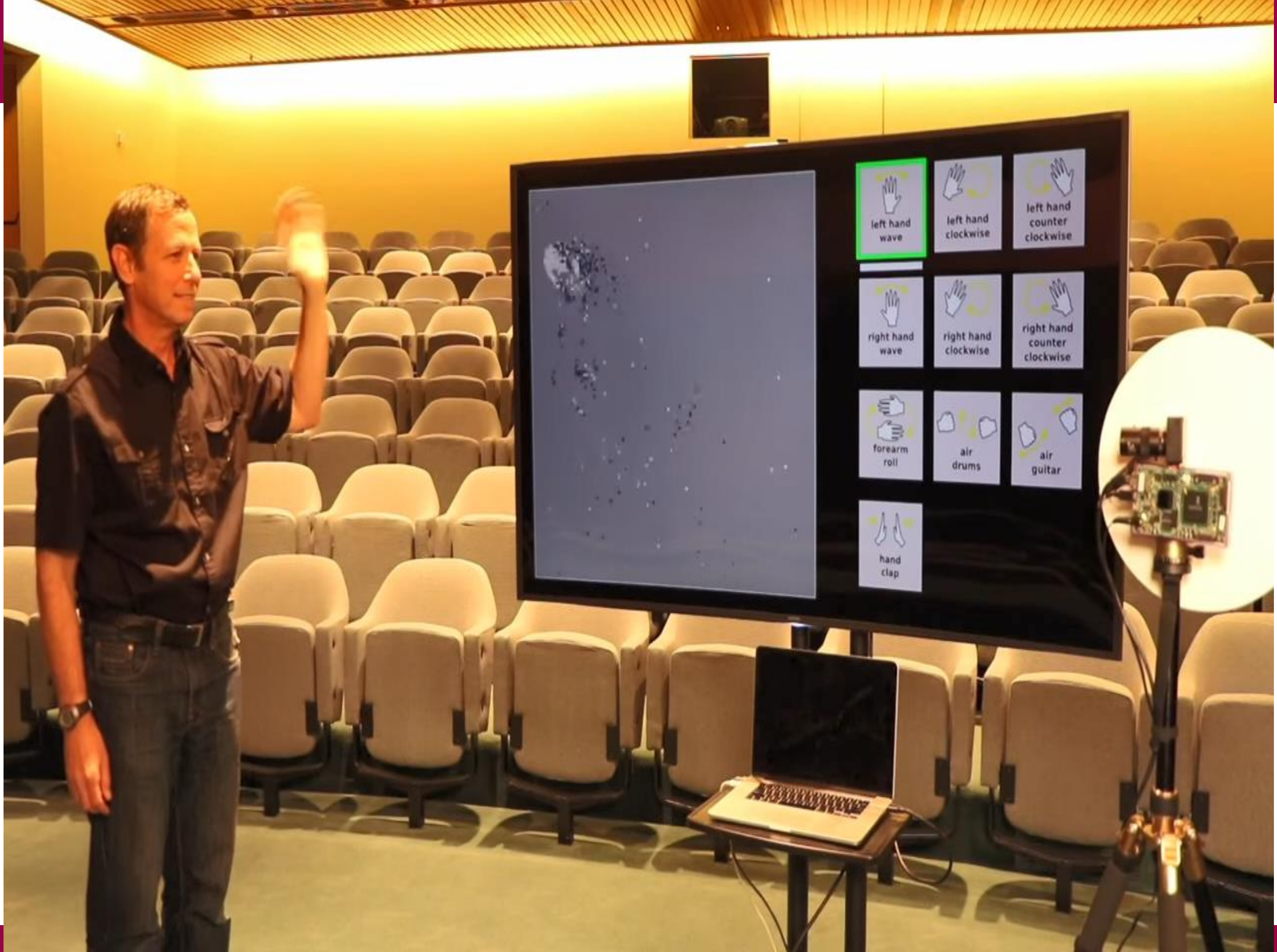
Spiking Neural Network (SNN) Regression

Task: Estimate angular velocity of an event camera with an SNN

- SNNs are **asynchronous networks** that consume events
- Hence,
  - no additional latency
  - no preprocessing necessary
- We show that **SNNs are competitive to ANNs** on this task



Gehrig et al., "Event-Based Angular Velocity Regression with Spiking Networks", ICRA20 PDF


left hand wave
left hand clockwise
left hand counter clockwise
right hand wave
right hand clockwise
right hand counter clockwise
forearm roll
air drums
air guitar
hand clap

Conclusions, Takeaways, Resources

Recap

Event cameras have many **advantages**:

- high dynamic range (HDR)
- high speed
- low latency
- low power

Current commercial applications

- IoT

monitoring and surveillance

- Automotive:

low-latency detection, object classification, low-power and low-memory storage

- AR/VR

low-latency, inter-frame pose estimation, low-power

- Industrial automation

Fast pick and place

Research Challenges with Event Cameras

Quantify the trade-offs:

- Latency vs. power consumption and accuracy
- Sensitivity vs. bandwidth and processing capacity

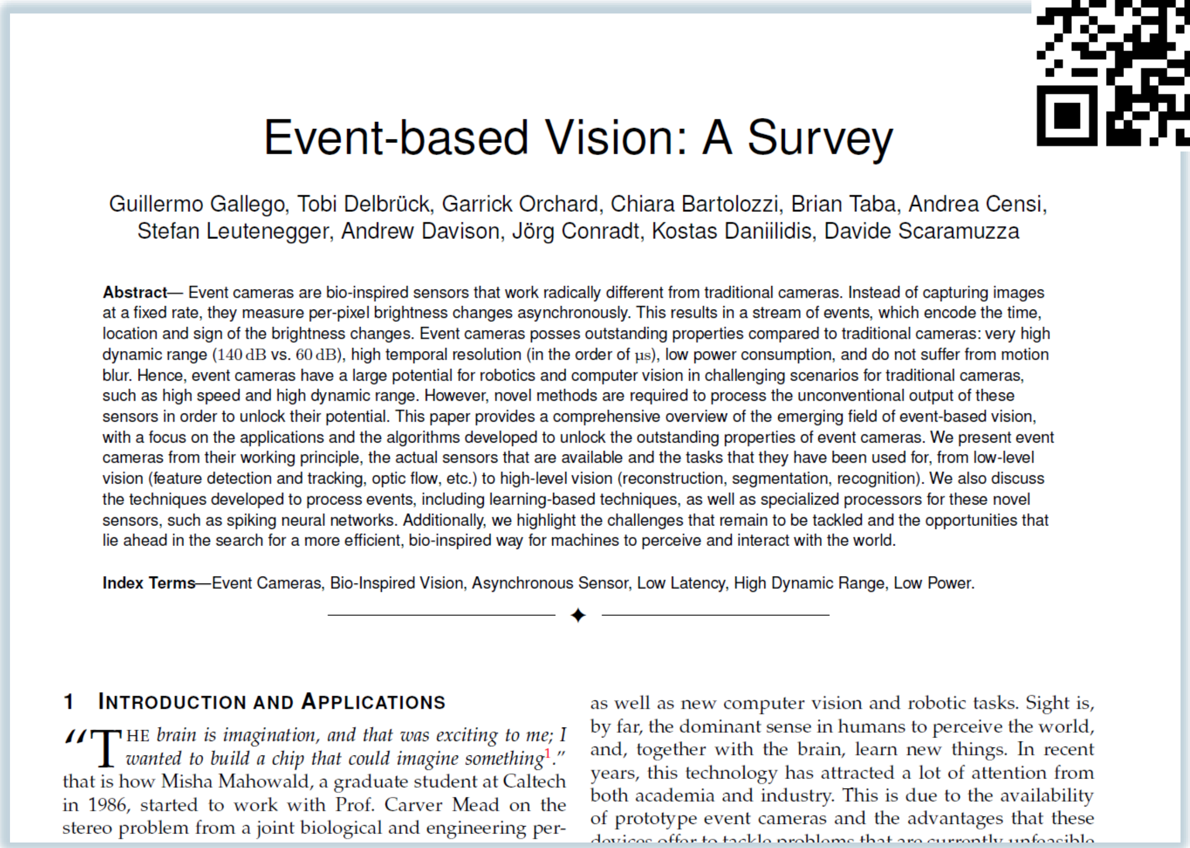
Active parameter adaptation Hardware:

1. pairing event cameras with dedicated hardware (SNN hardware, e.g., Intel Loihi, aiCTX Speck)
2. How do we make sparse convolution in space and time efficient?

Learning with event cameras:

- How do we exploit knowledge from image-based learning to event cameras?
- Asynchronous inference
- Where do we find learning data? Event data is much more rare than frames. Potential solutions: unsupervised Learning, learning in simulation, transfer learning from frames to events

Reference: T-PAMI 2020 paper



Event-based Vision: A Survey

Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew Davison, Jörg Conradt, Kostas Daniilidis, Davide Scaramuzza

Abstract— Event cameras are bio-inspired sensors that work radically different from traditional cameras. Instead of capturing images at a fixed rate, they measure per-pixel brightness changes asynchronously. This results in a stream of events, which encode the time, location and sign of the brightness changes. Event cameras posses outstanding properties compared to traditional cameras: very high dynamic range (140 dB vs. 60 dB), high temporal resolution (in the order of µs), low power consumption, and do not suffer from motion blur. Hence, event cameras have a large potential for robotics and computer vision in challenging scenarios for traditional cameras, such as high speed and high dynamic range. However, novel methods are required to process the unconventional output of these sensors in order to unlock their potential. This paper provides a comprehensive overview of the emerging field of event-based vision, with a focus on the applications and the algorithms developed to unlock the outstanding properties of event cameras. We present event cameras from their working principle, the actual sensors that are available and the tasks that they have been used for, from low-level vision (feature detection and tracking, optic flow, etc.) to high-level vision (reconstruction, segmentation, recognition). We also discuss the techniques developed to process events, including learning-based techniques, as well as specialized processors for these novel sensors, such as spiking neural networks. Additionally, we highlight the challenges that remain to be tackled and the opportunities that lie ahead in the search for a more efficient, bio-inspired way for machines to perceive and interact with the world.

Index Terms—Event Cameras, Bio-Inspired Vision, Asynchronous Sensor, Low Latency, High Dynamic Range, Low Power.

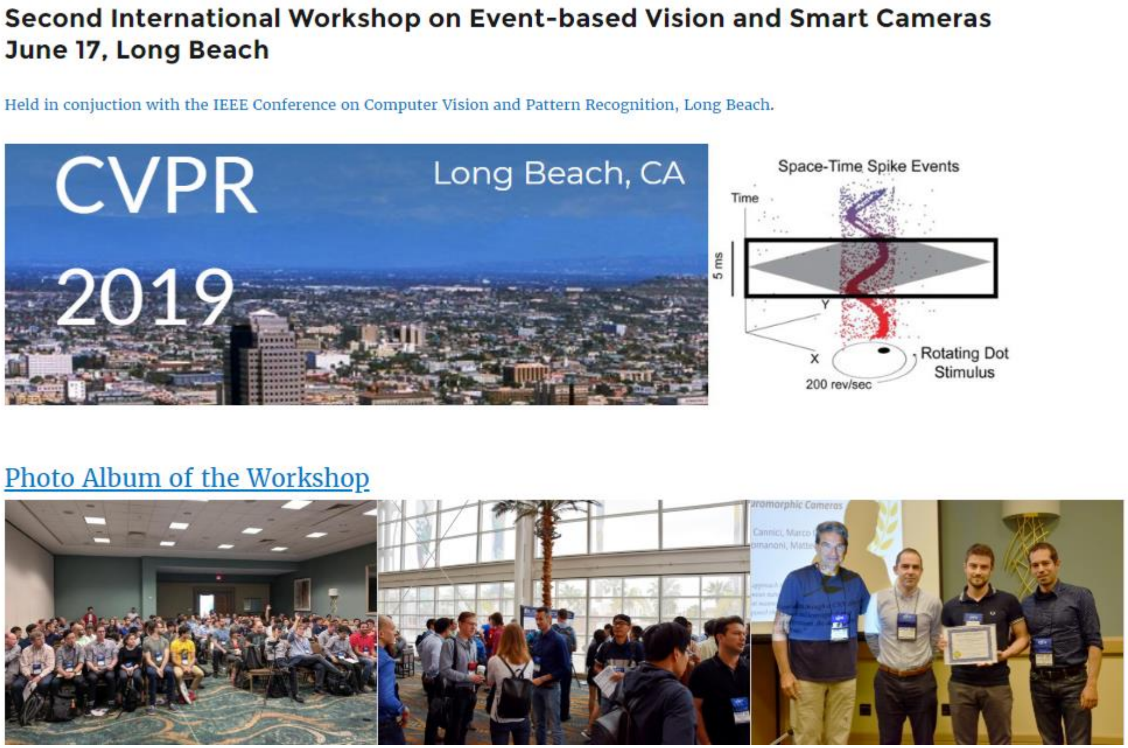
1 Introduction and Applications

"THE *brain is imagination, and that was exciting to me; I wanted to build a chip that could imagine something*[1]." that is how Misha Mahowald, a graduate student at Caltech in 1986, started to work with Prof. Carver Mead on the stereo problem from a joint biological and engineering per-

as well as new computer vision and robotic tasks. Sight is, by far, the dominant sense in humans to perceive the world, and, together with the brain, learn new things. In recent years, this technology has attracted a lot of attention from both academia and industry. This is due to the availability of prototype event cameras and the advantages that these devices offer to tackle problems that are currently unfeasible

http://rpg.ifi.uzh.ch/docs/EventVisionSurvey.pdf

CVPR19 Workshop on Event-based Vision



Second International Workshop on Event-based Vision and Smart Cameras
June 17, Long Beach

Held in conjuction with the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach.

Photo Album of the Workshop

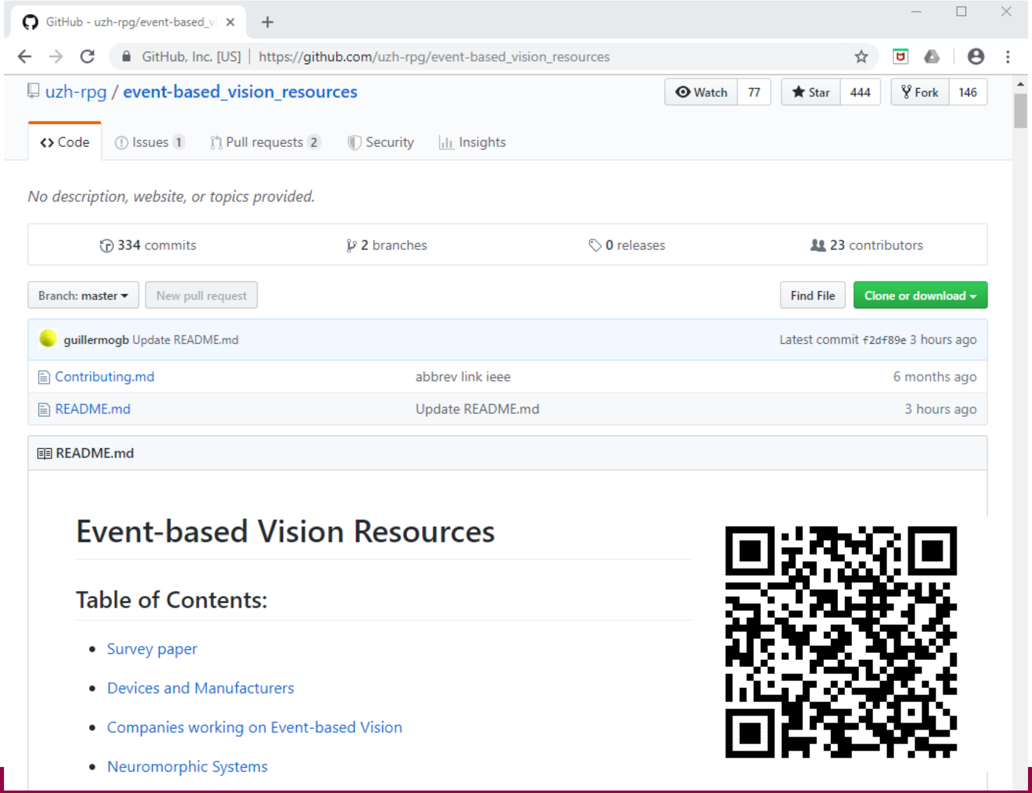
Full-day workshop with talks by 23 researchers on event-based cameras, including Samsung, Intel, and event-camera companies

Slides and video recordings: http://rpg.ifi.uzh.ch/CVPR19_event_vision_workshop.html

List of Event-based Vision resources

Code, datasets, papers, videos, companies on event cameras

https://github.com/uzh-rpg/event-based_vision_resources



uzh-rpg / event-based_vision_resources

No description, website, or topics provided.

334 commits | 2 branches | 0 releases | 23 contributors

guillermogb Update README.md — Latest commit f2df89e 3 hours ago

Contributing.md	abbrev link ieee	6 months ago
README.md	Update README.md	3 hours ago

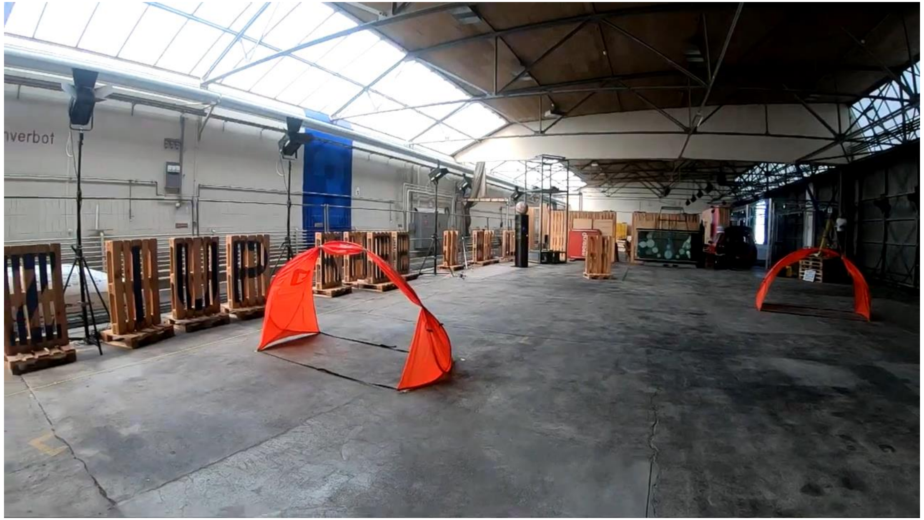
README.md

Event-based Vision Resources

Table of Contents:

- Survey paper
- Devices and Manufacturers
- Companies working on Event-based Vision
- Neuromorphic Systems

UZH-FPV Drone Racing Dataset & Competition



Recorded with a drone flown by a **professional pilot up to over 20m/s**

Contains over 30 sequences with **images, events, IMU**, and **ground truth from a robotic total station**:
http://rpg.ifi.uzh.ch/uzh-fpv.html

IROS 2019 Drone Racing VIO competition: https://github.com/uzh-rpg/IROS2019-FPV-VIO-Competition **Win up to $2,000 plus invited talk** . **Submission deadline: Sep. 27, 2020**

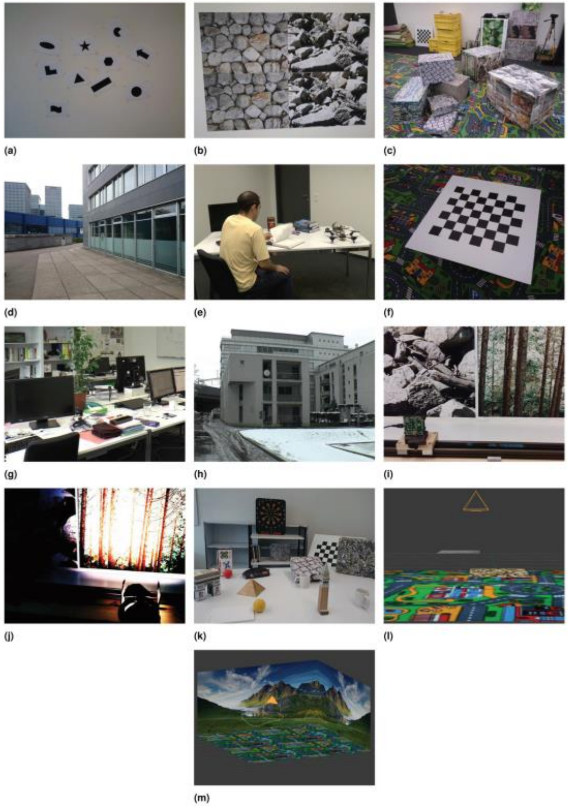
Delmerico et al. "Are We Ready for Autonomous Drone Racing? The UZH-FPV Drone Racing Dataset" ICRA'19
PDF. Video. Datasets.

Other tasks:

Other tasks including obstacle avoidance, exploration, control on UAV etc. can be found:

http://rpg.ifi.uzh.ch/research.html

Dataset



RPG dataset
http://rpg.ifi.uzh.ch/research_dvs.html

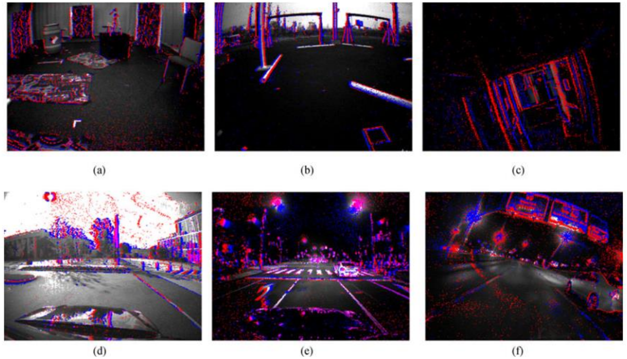
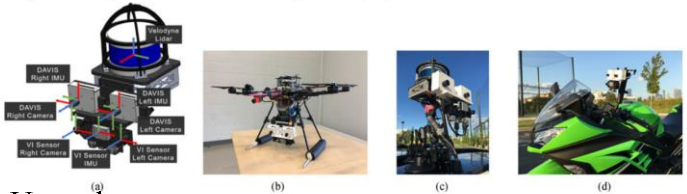
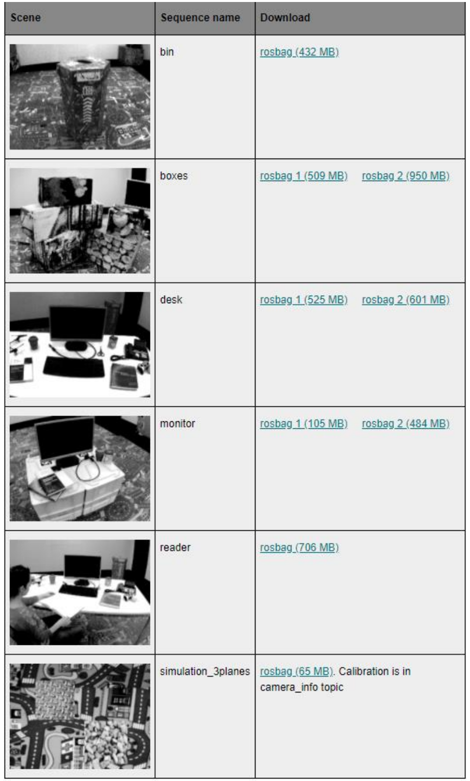


Fig. 3. Sample images with overlaid events (blue and red) from indoor and outdoor sequences, during day and evening. Best viewed in color. (a) Hexacopter indoor flight with Vicon motion capture. (b) Hexacopter outdoor flight with qualisys motion capture. (c) Handheld with difficult lighting conditions. (d) Car day 1. (e) Outdoor car evening. (f) Motorcycle highway 1.

Fig. 4. Motion capture arenas. Left: Indoor Vicon arena, right: Outdoor Qualisys arena.



Upen dataset
https://daniilidis-group.github.io/mvsec/



Scene	Sequence name	Download
	bin	rosbag (432 MB)
	boxes	rosbag 1 (509 MB) rosbag 2 (950 MB)
	desk	rosbag 1 (525 MB) rosbag 2 (601 MB)
	monitor	rosbag 1 (105 MB) rosbag 2 (484 MB)
	reader	rosbag (706 MB)
	simulation_3planes	rosbag (65 MB). Calibration is in camera_info topic

Stereo event cameras
http://rpg.ifi.uzh.ch/ECCV18_stereo_davis.html

Online Resource

- Event camera paper collection: https://github.com/uzh-rpg/event-based_vision_resources
- VO/VSLAM/VIO
  - code: Life time estimation: https://github.com/uzh-rpg/rpg_event_lifetime
  - code: Corner detection: https://github.com/uzh-rpg/rpg_corner_events
  - code: **EventEMin**: https://github.com/ImperialCollegeLondon/EventEMin
  - code: dvs-panotracking: https://github.com/VLOGroup/dvs-panotracking
  - code: EVIO_feature tracker: https://github.com/alexzzhu/event_feature_tracking
  - code: **EMVS_stereo depth reconstruction:** https://github.com/uzh-rpg/rpg_emvs
  - code: **ESVO**: https://github.com/HKUST-Aerial-Robotics/ESVO
  - https://git.ram-lab.com/gogojjh/M-LOAM/src/mloam_gf/eloam
- Other tasks
  - code: EKLT_feature tracker: https://github.com/uzh-rpg/rpg_eklt
    - evaluation code: https://github.com/uzh-rpg/rpg_feature_tracking_analysis
  - code: End-to-end learing representation for event camera: https://github.com/uzh-rpg/rpg_event_representation_learning
  - code: SNN for angular velocity: https://github.com/uzh-rpg/snn_angular_velocity
  - code: Video to event image reconstruction: https://github.com/uzh-rpg/rpg_vid2e
  - code: Event image to frame image reconstruction: https://github.com/uzh-rpg/rpg_e2vid
  - code: Event-based Asynchronous Sparse Convolutional Networks: https://github.com/uzh-rpg/rpg_asynet
  - code: UZH drone racing: https://github.com/uzh-rpg/rpg_feature_tracking_analysis
  - code: Simulator: http://rpg.ifi.uzh.ch/esim.html

Thank you !
Q & A

Acknowledgement:
Some materials are from the tutorial on event cameras from RPG, ETH Zurich
http://rpg.ifi.uzh.ch/docs/scaramuzza/Tutorial_on_Event_Cameras_Scaramuzza.pdf