Lab Visit and Exercise - Today

- **Lab visit with live demos** (@Robotics and Perception Group):
  - We will take Tram 10 to Bahnhof Oerlikon Ost
  - Lab address: Andreasstrasse 15, 2nd floor
  - Duration of the visit: ~1 hour

- **Lunch**: There is a UZH mensa in **UZH BIN** building (Binzmuhlestrasse 14 (across the street from the RPG lab), open till 14:00

- **Exercise** Session: Q&A on final VO integration
  - Room **UZH BIN 2.A.10** from **14:00 to 16:00**

Exams Questions

- The oral exam will last 30 minutes
- It will consist of one application question followed by two theoretical questions
- This document contains a "**non exhaustive**" list of possible application questions and an "**exhaustive**" list of all the topics that you should learn about the course, which will be subject of discussion in the theoretical part: http://rpg.ifi.uzh.ch/docs/teaching/2017/Exam_Questions.pdf

University of Zurich[UZH] | ETH zürich

Institute of Informatics – Institute of Neuroinformatics

ROBOTICS & PERCEPTION GROUP

Event based vision

Towards Robust Visual Inertial SLAM:

from Frame-based to Event Cameras

Davide Scaramuzza

Review

- Direct (photometric) based vs Feature based approaches
- Direct (photometric) based methods:
  Influence of the number of pixels:
  - Dense,
  - Semi-dense,
  - Sparse methods
- IMU

Feature-based methods

1. Extract & match features (+RANSAC)
2. Minimize **Reprojection error** minimization

$$T_{k,k-1} = \arg\min_{T} \sum_{i} \| \boldsymbol{u}'_i - \pi(\boldsymbol{p}_i) \|_{\Sigma}^2$$

- ✓ **Large frame-to-frame motions**
- × **Slow due to costly feature extraction and matching**
- × **Matching Outliers (RANSAC)**

Direct (photometric) methods

1. The Pixel is the feature to track
2. Minimize **Photometric error**

$$T_{k,k-1} = \arg\min_{T} \sum_{i} \| I_k(\boldsymbol{u}'_i) - I_{k-1}(\boldsymbol{u}_i) \|_{\sigma}^2$$

- ✓ **All information in the image can be exploited (precision, robustness)**
- ✓ **Increasing camera frame-rate reduces computational cost per frame**
- × **Limited frame-to-frame motion**

[Jin,Favaro,Soatto'03] [Silveira, Malis, Rives, TRO'08], [SVO, Forster'14,-17], [LSD-SLAM, Engel'14], [DSO, Engel'17]

Popular SLAM algorithms

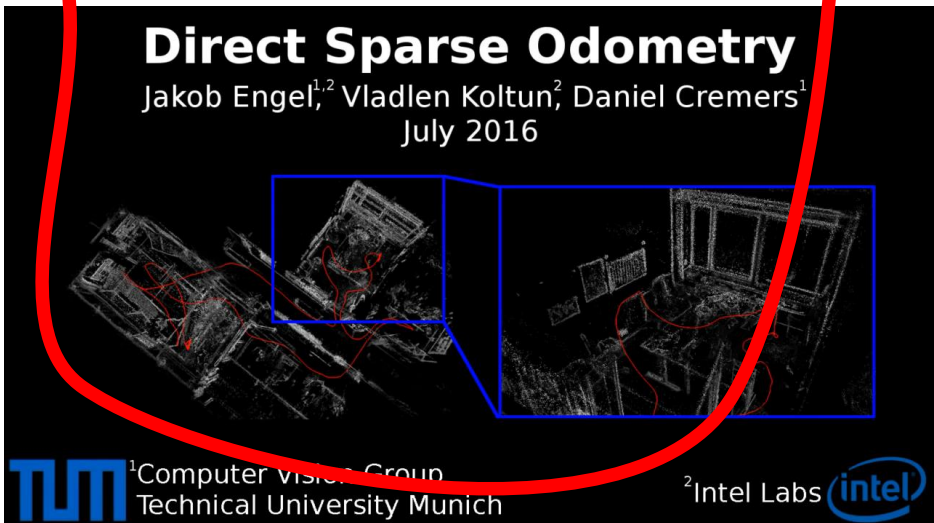
[SVO, ICRA'14] [SVO 2.0 TRO'17]

LSD-SLAM, ECCV'14

Photometric (Direct) methods

ORB-SLAM, TRO'15

DSO, PAMI'17

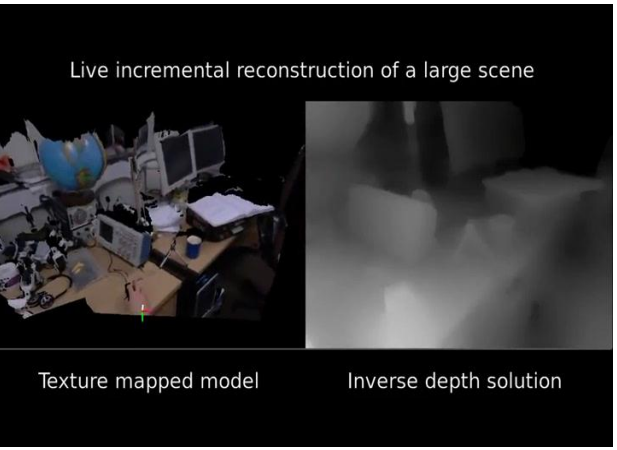


Review

- Direct (photometric) based vs Feature based approaches
- Direct (photometric) based methods:
  Influence of the number of pixels:
  - Dense,
  - Semi-dense,
  - Sparse methods
- IMU

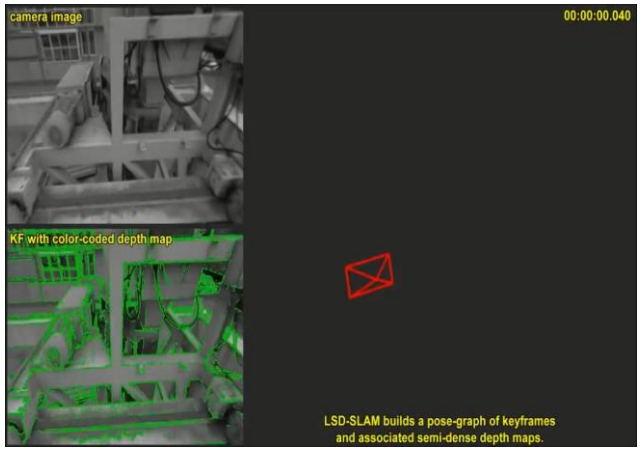
Direct Methods: Dense vs Semi-dense vs Sparse [TRO'16]

Dense



DTAM [Newcombe '11] REMODE [Pizzoli'14]
300'000+ pixels

Semi-Dense



LSD-SLAM [Engel'14]
~10,000 pixels

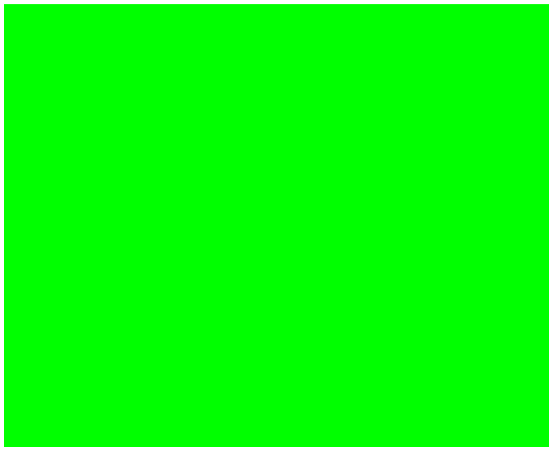
Sparse

SVO [Forster'14] DSO [Engel'17]
100-200 x 4x4 patches ≅ 2,000 pixels

[Forster, et al., SVO: Semi Direct Visual Odometry for Monocular and Multi-Camera Systems, TRO'17]

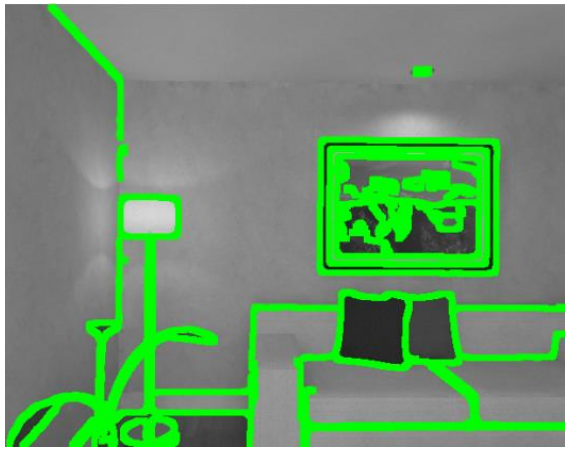
Direct Methods: Dense vs Semi-dense vs Sparse [TRO'16]

Dense



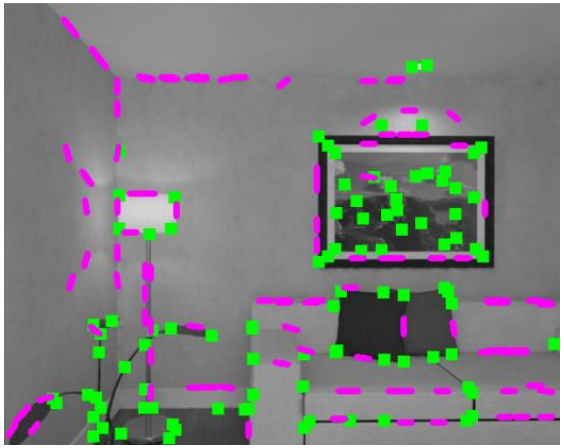
DTAM [Newcombe '11] REMODE [Pizzoli'14]
300'000+ pixels

Semi-Dense



LSD-SLAM [Engel'14]
~10,000 pixels

Sparse

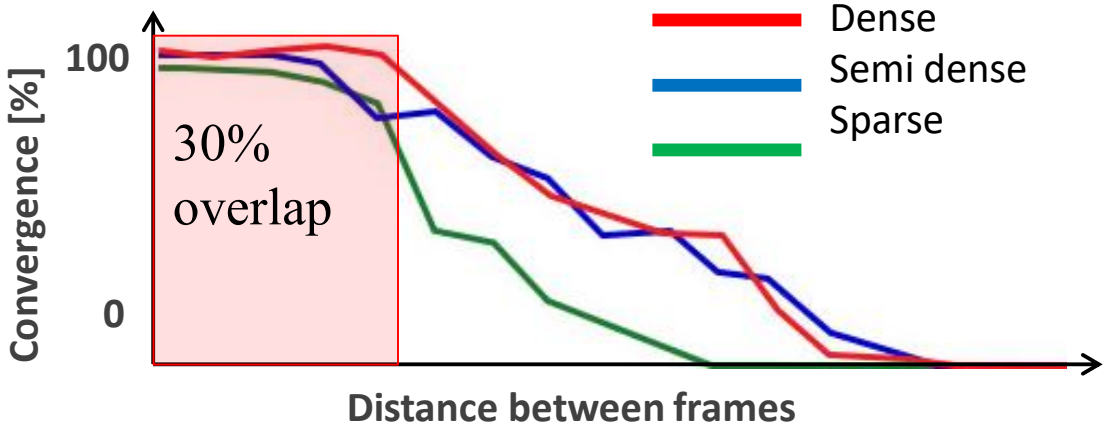


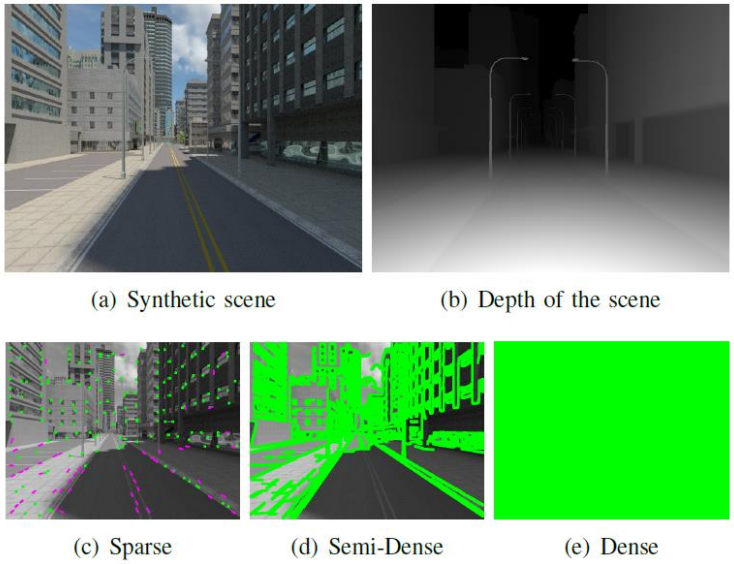
SVO [Forster'14]
100-200 x 4x4 patches ≅ 2,000 pixels

[Forster, et al., SVO: Semi Direct Visual Odometry for Monocular and Multi-Camera Systems, TRO'17]

Direct Methods: Dense vs Semi-dense vs Sparse [TRO'16]

Robustness to motion baseline (computed from 1,000 Blender simulations)





Images from the synthetic Multi-FOV Zurich Urban Dataset

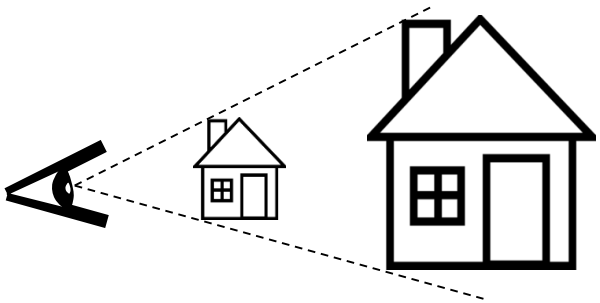
- **Dense** and **Semi-dense behave similarly**
  - weak gradients are not informative for the optimization)
- Dense only useful with **motion blur** and **defocus**
- **Sparse** methods behave equally well for image **overlaps up to 30%**

- Forster, et al., SVO: Semi Direct Visual Odometry for Monocular and Multi-Camera Systems, TRO'17
- Multi-FOV Zurich Urban Dataset: http://rpg.ifi.uzh.ch/fov.html

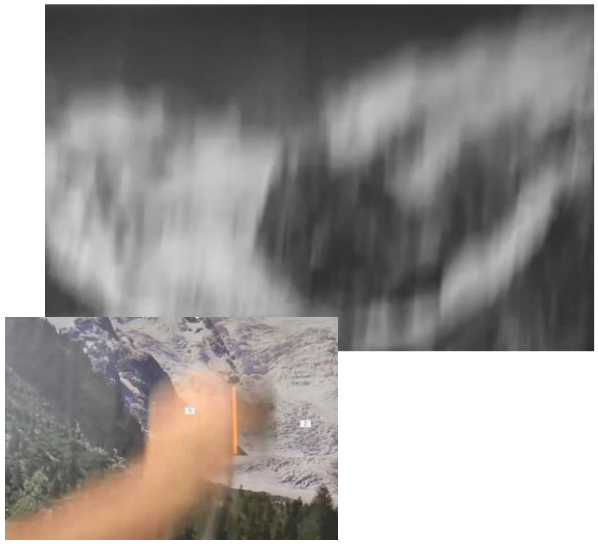
Review

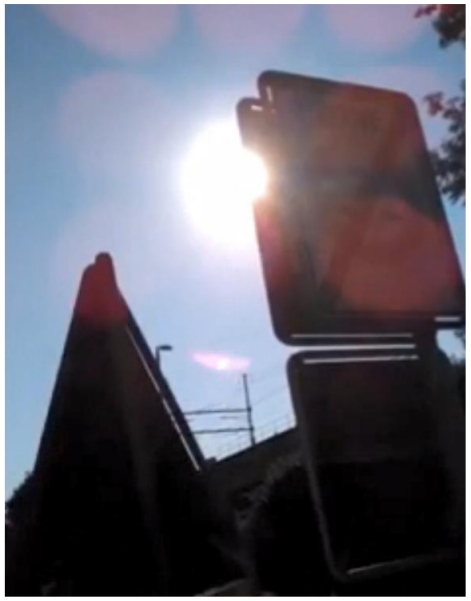
- Direct (photometric) based vs Feature based approaches
- Direct (photometric) based methods:
  Influence of the number of pixels:
  - Dense,
  - Semi-dense,
  - Sparse methods
- IMU

Why do we need an IMU?

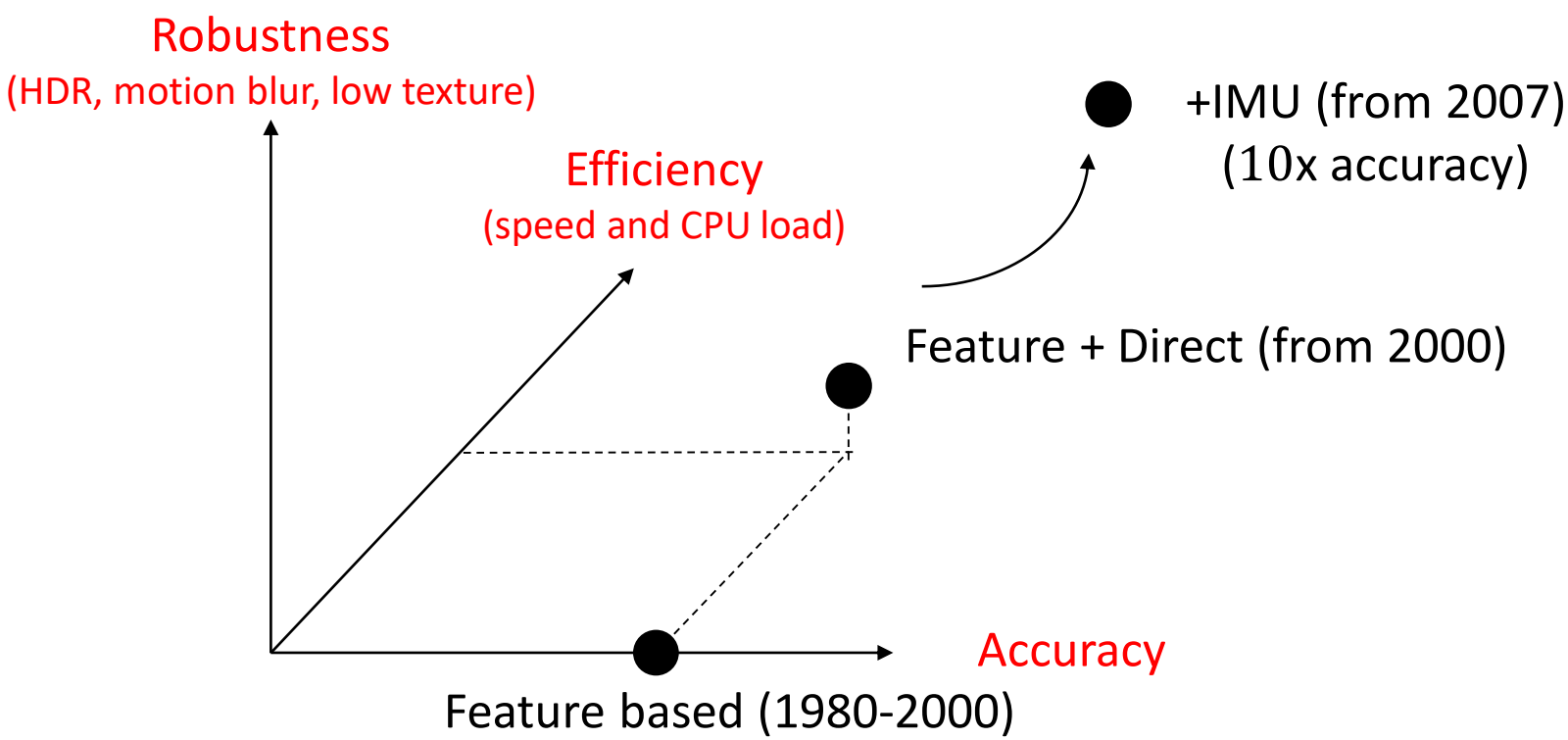


- Monocular vision is scale ambiguous.
- **Pure vision is not robust** enough to
  - Low texture
  - High dynamic range (HDR)
  - High speed motion (because of motion blur)





The 3 Axes of SLAM Research: **Accuracy, Efficiency, Robustness**



Robustness: Challenges of Vision for SLAM

- IMU alone only helpful for short motions; **drifts very quickly** without visual constraint
- Biggest challenges for vision today is robustness to:
  - High Dynamic Range (HDR)
    - Can be handled with active exposure control or Event cameras
  - High-speed motion (i.e., motion blur)
    - Can be handled with event cameras
  - Low-texture scenes
    - Can be handled with Depth cameras or by getting closer to the scene
  - Dynamic environments
    - Deep learning?
- Current VO algorithms and sensors have big latencies (50-200 ms)
  - Can we reduce this to much below a 1ms?
    - Can be handled with event cameras

Active Exposure Control for HDR Scenes

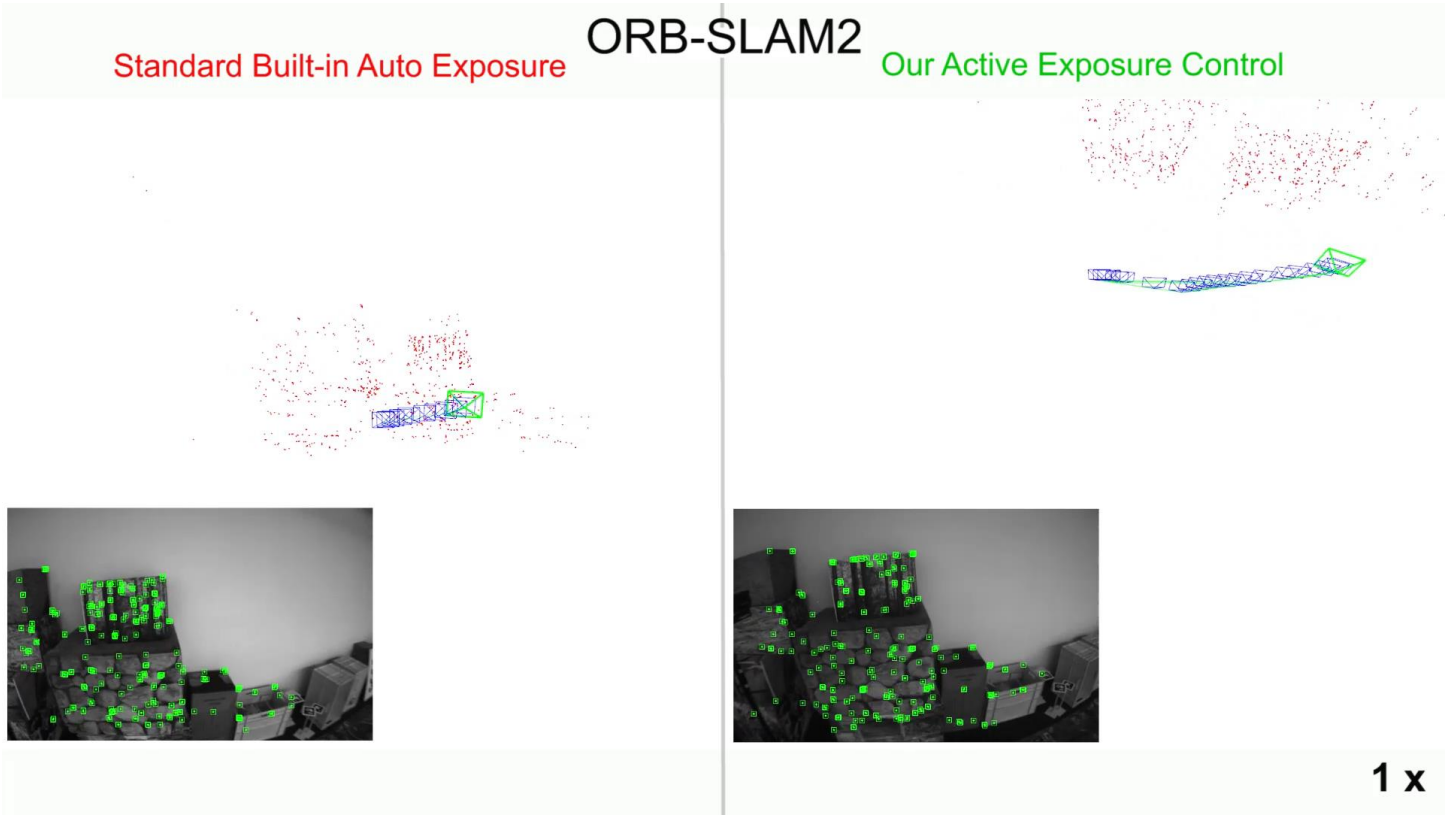
- Goal: Actively control exposure to maximize image gradient information

$$M_{\text{perc}}(p) \quad = \quad \text{percentile}(\{G(\mathbf{u}_i)\}_{\mathbf{u}_i \in \mathbf{I}}, p)$$

- Simple gradient ascent control

$$\Delta t_{\text{next}} = \Delta t + \gamma \boxed{\frac{\partial M_{\text{softperc}}}{\partial \Delta t}}$$

Computed from photometric response function



Zhang, Forster, Scaramuzza, Active Exposure Control for Robust Visual Odometry in HDR Environments, ICRA'17

Event-based Cameras

Outline

- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

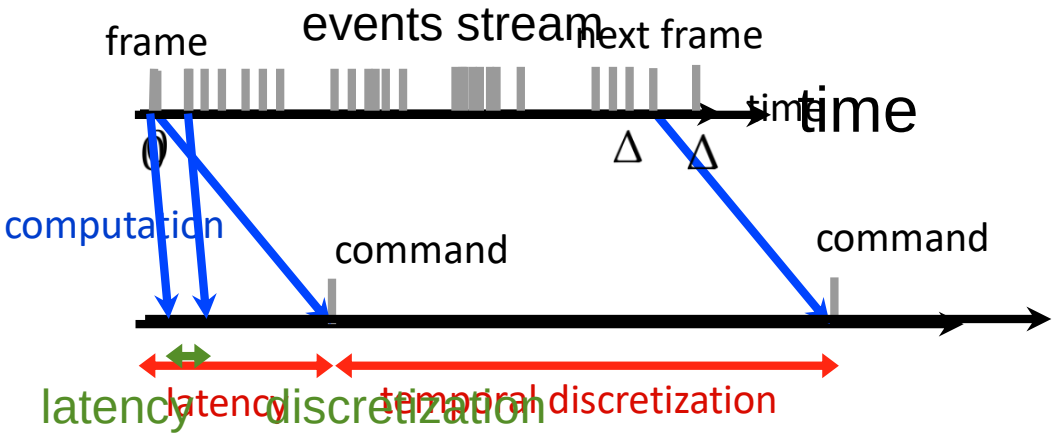
Latency and Agility are tightly coupled!

Current flight maneuvers achieved with onboard cameras are still to slow compared with those attainable by **birds**. We need **low latency sensors and algorithms**!

A sparrowhawk catching a garden bird (National Geographic)

To go faster, we need faster sensors!

- The agility of a robot is limited by the latency and temporal discretization of its sensing pipeline.



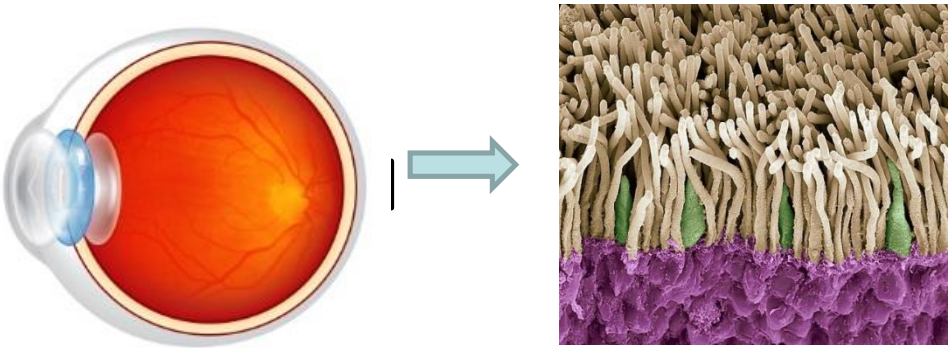
- The average robot-vision algorithms have latencies of 50-200 ms, which puts a hard bound on the agility of the platform

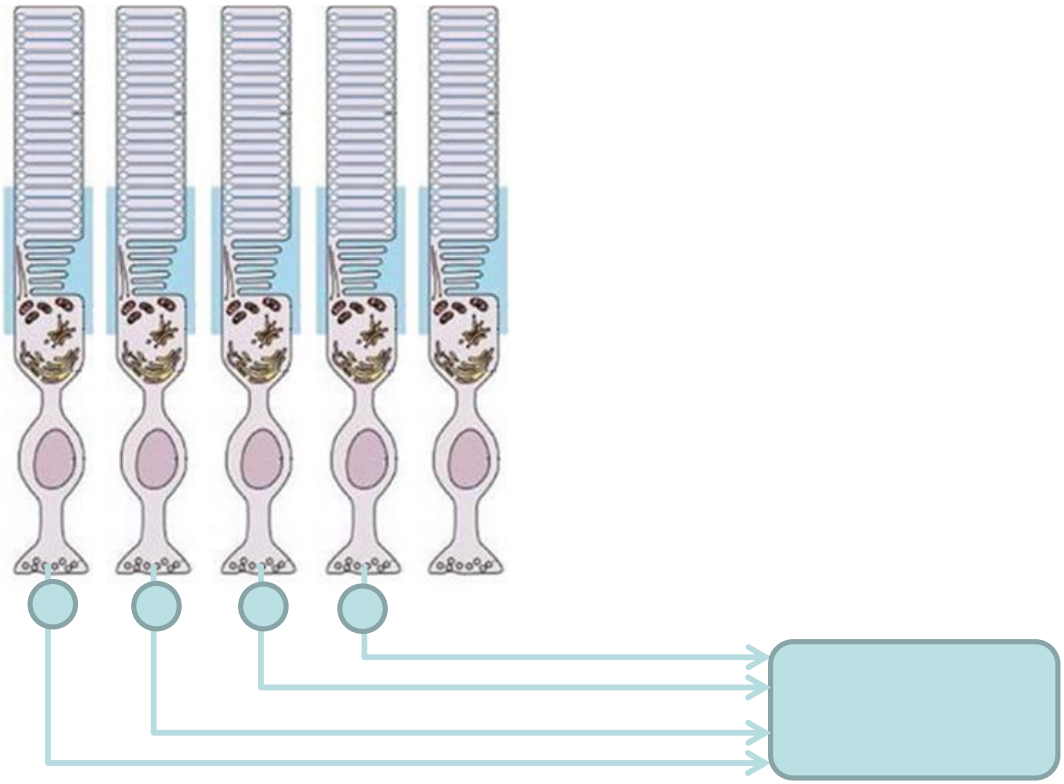
- Event cameras enable **low-latency sensory motor control (<< 1ms)**

Censi, Strubel, Brandli, Delbruck, Scaramuzza, Low-latency localization by Active LED Markers tracking using a Dynamic Vision Sensor, IROS'13

Human Vision System

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

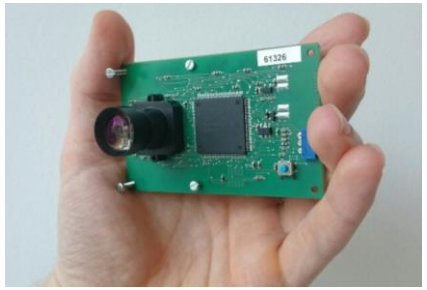




Outline

- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

Dynamic Vision Sensor (DVS)



DVS from inilabs.com

Advantages

- **Low-latency** (~1 micro-seconds)
- **High-dynamic range (HDR)** (140 dB instead 60 dB)
- **High updated rate** (1 MHz)
- **Low power** (10mW instead 1W)

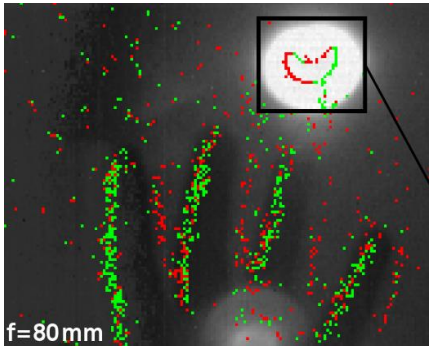
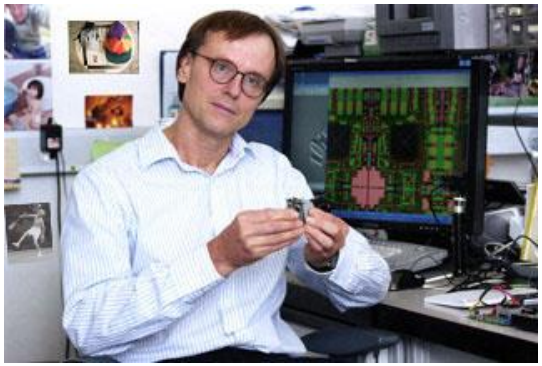


Image of solar eclipse captured by a DVS, without black filter!

Disadvantages

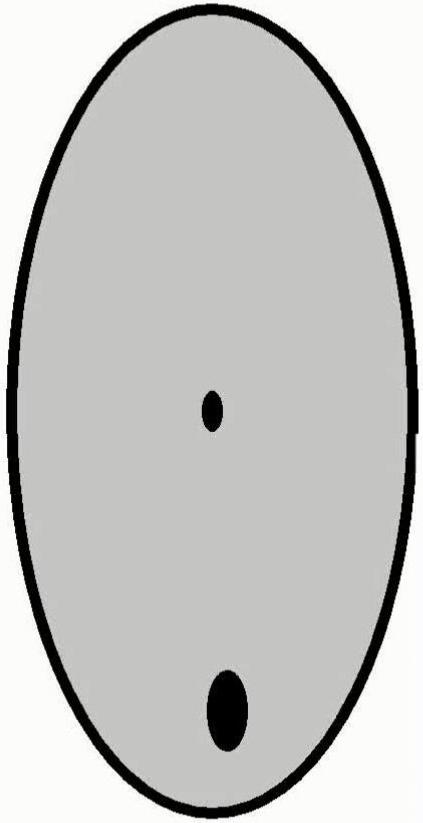
- **Paradigm shift**: Requires totally **new vision algorithms**:
  - **Asynchronous** pixels
  - **No intensity information** (only binary intensity changes)



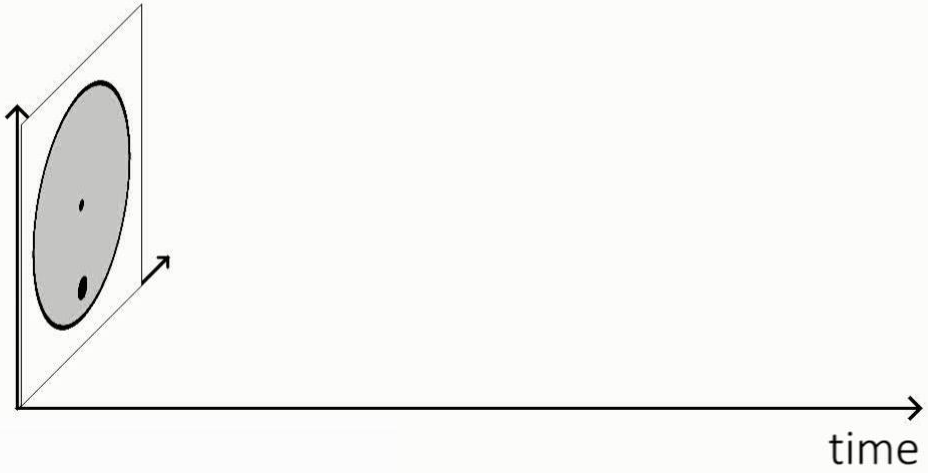
Prof. Tobi Delbruck, UZH & ETH Zurich

1. Lichtsteiner et al., A 128x128 120 dB 15µs Latency Asynchronous Temporal Contrast Vision Sensor, 2008
2. Brandli et al., A 240x180 130dB 3us Latency Global Shutter Spatiotemporal Vision Sensor, JSSC'14.

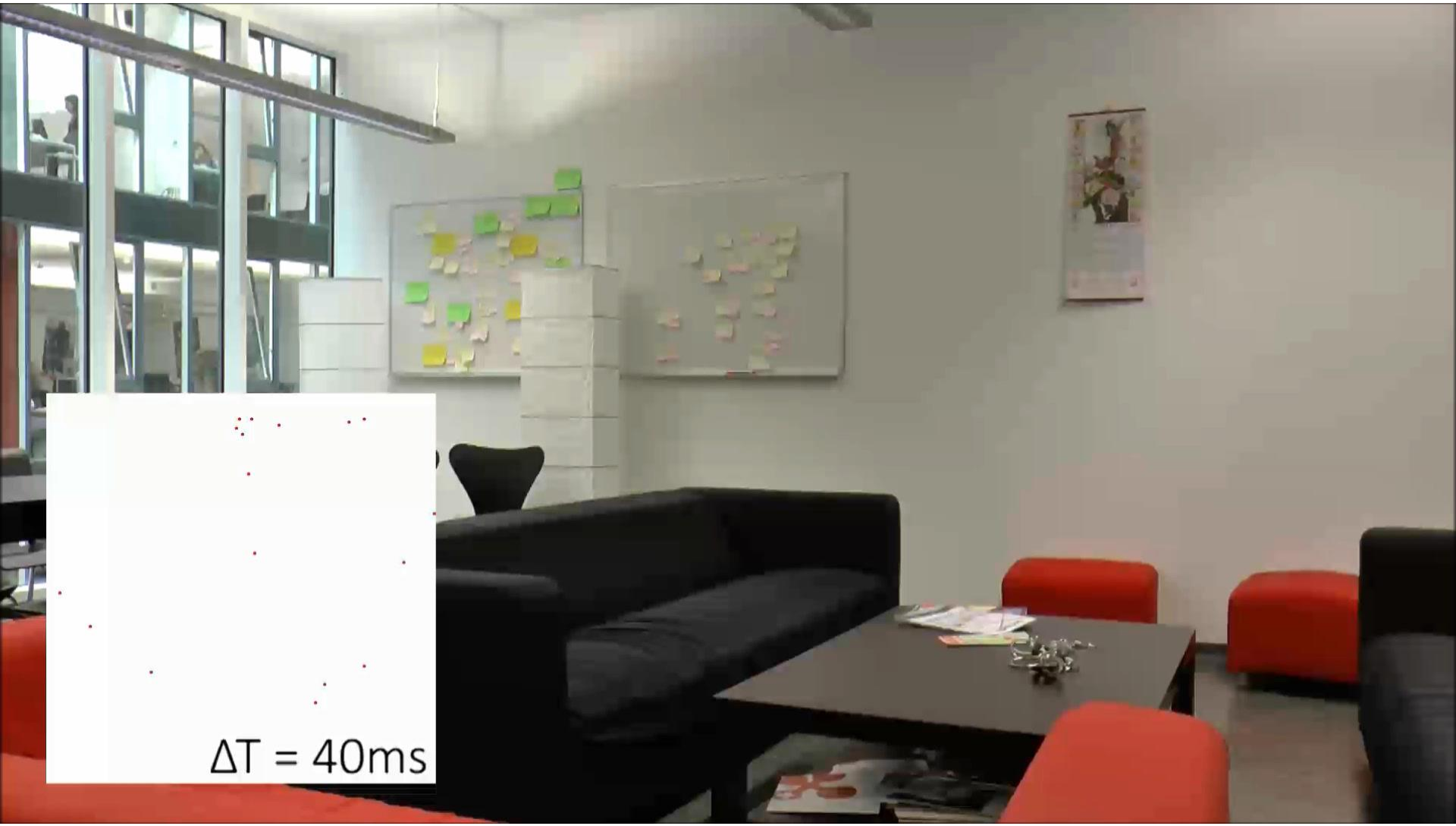
Camera vs Dynamic Vision Sensor



standard camera output:

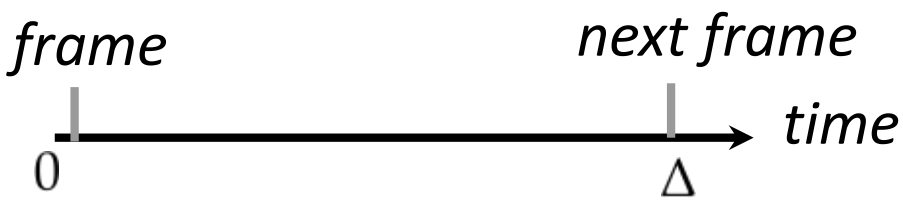


Camera vs Dynamic Vision Sensor

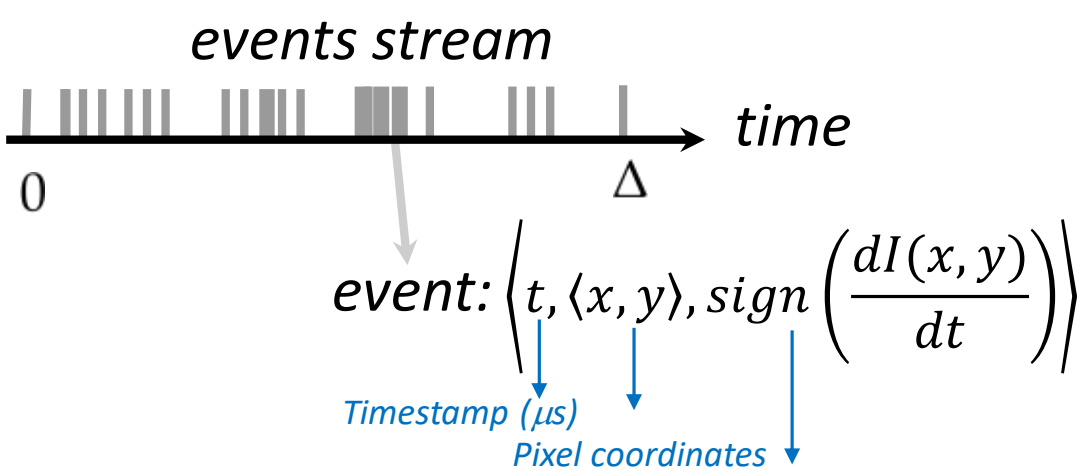


Dynamic Vision Sensor (DVS)

- 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



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

What is an event camera, **precisely**?

- **Asynchronous**: all pixels are *independent* from one another
- Implements ***level-crossing*** sampling (i.e. adaptive sampling)
- Reacts to ***logarithmic*** brightness changes

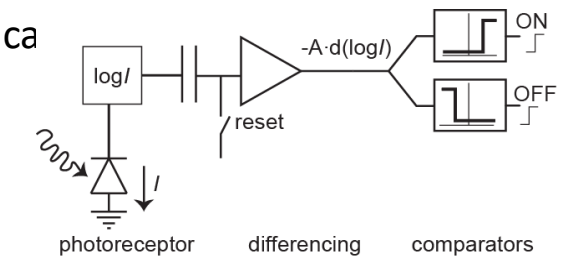
Let´s look at how this works for one pixel in detail

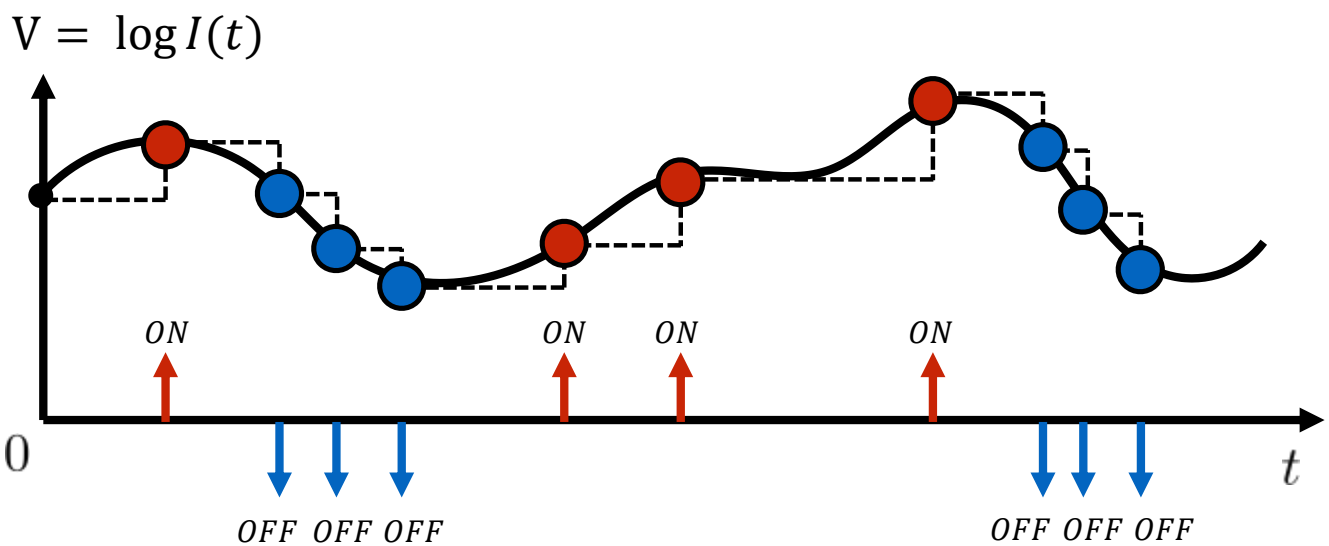
DVS Operating Principle

- A DVS detects and outputs ***asynchronous pixel-level brightness changes***
  - Each pixel is *independent* of all the other pixels
  - Events are generated any time a single pixel sees a change of the logarithm of the brightness that is equal to C, i.e.:

$$|\Delta \log I| \;=\; |\log I(t+\Delta t) - \log I(t)| \;= C$$

  - $C \in [0.15, 0.20]$ is called ***Contrast sensitivity*** and can be tuned by the user
  - Since brightness changes can be either positive or negative, we ca
    - **ON event**: if and only if $\Delta \log I = C$
    - **OFF event**: if and only if $\Delta \log I = -C$

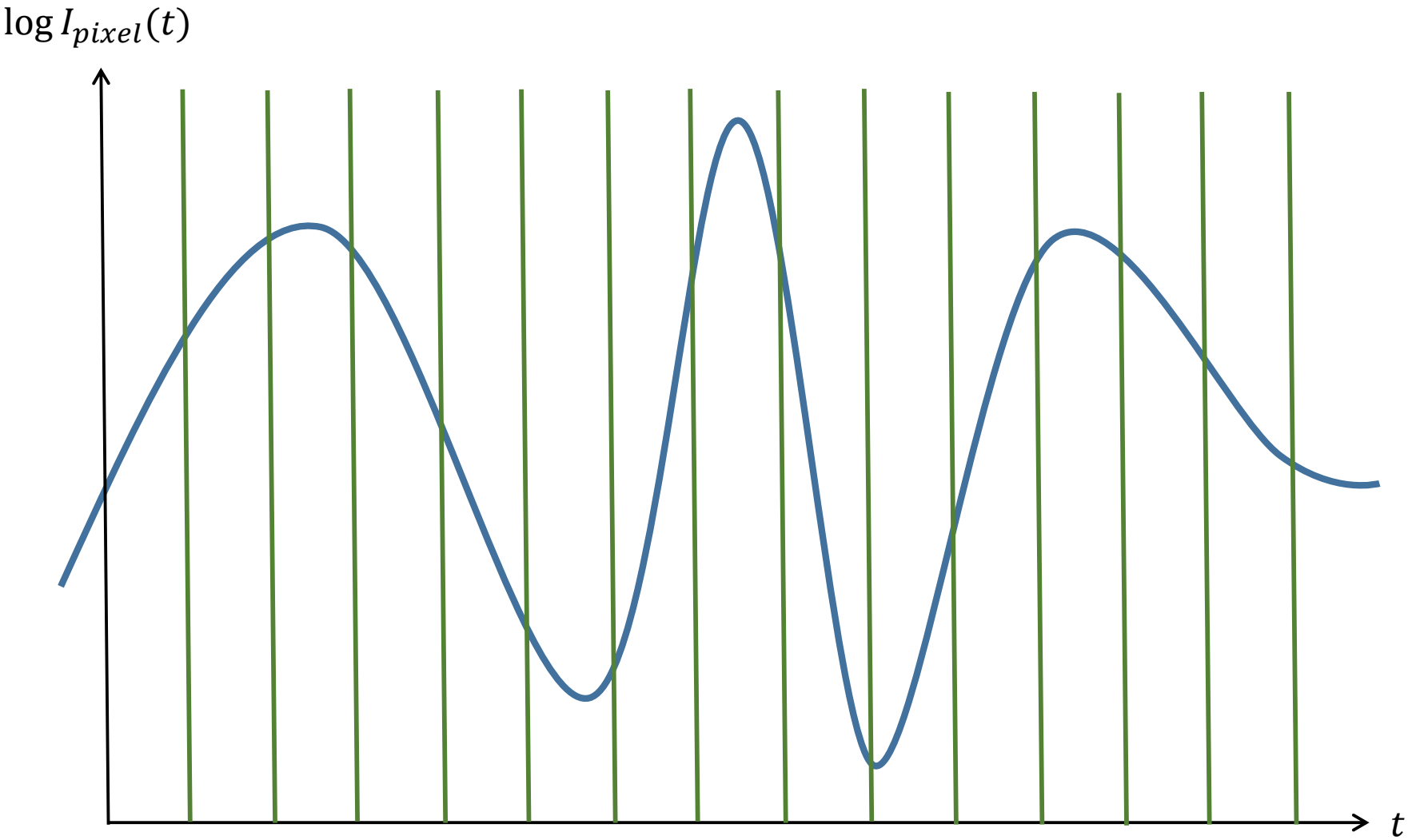




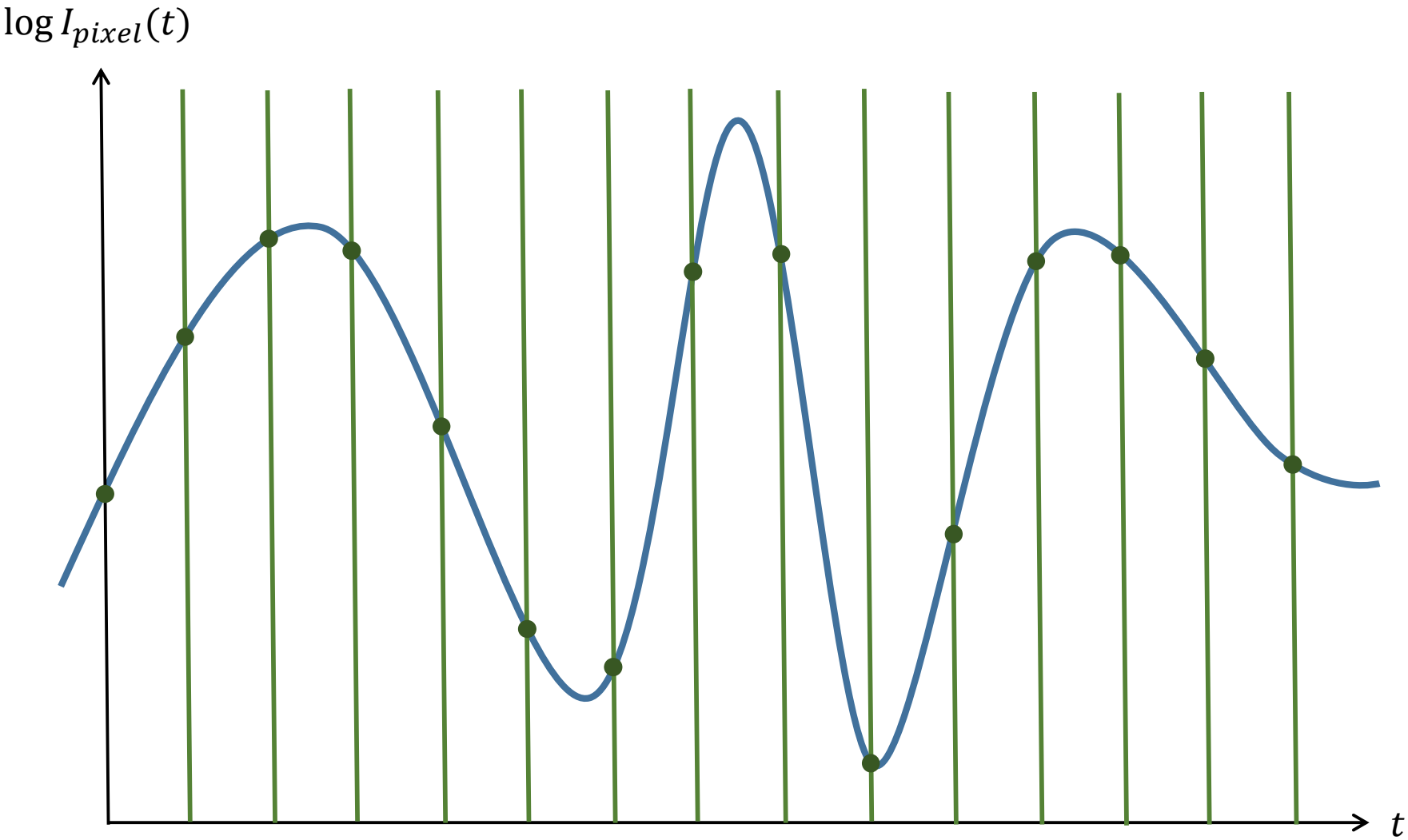
Outline

- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level-crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

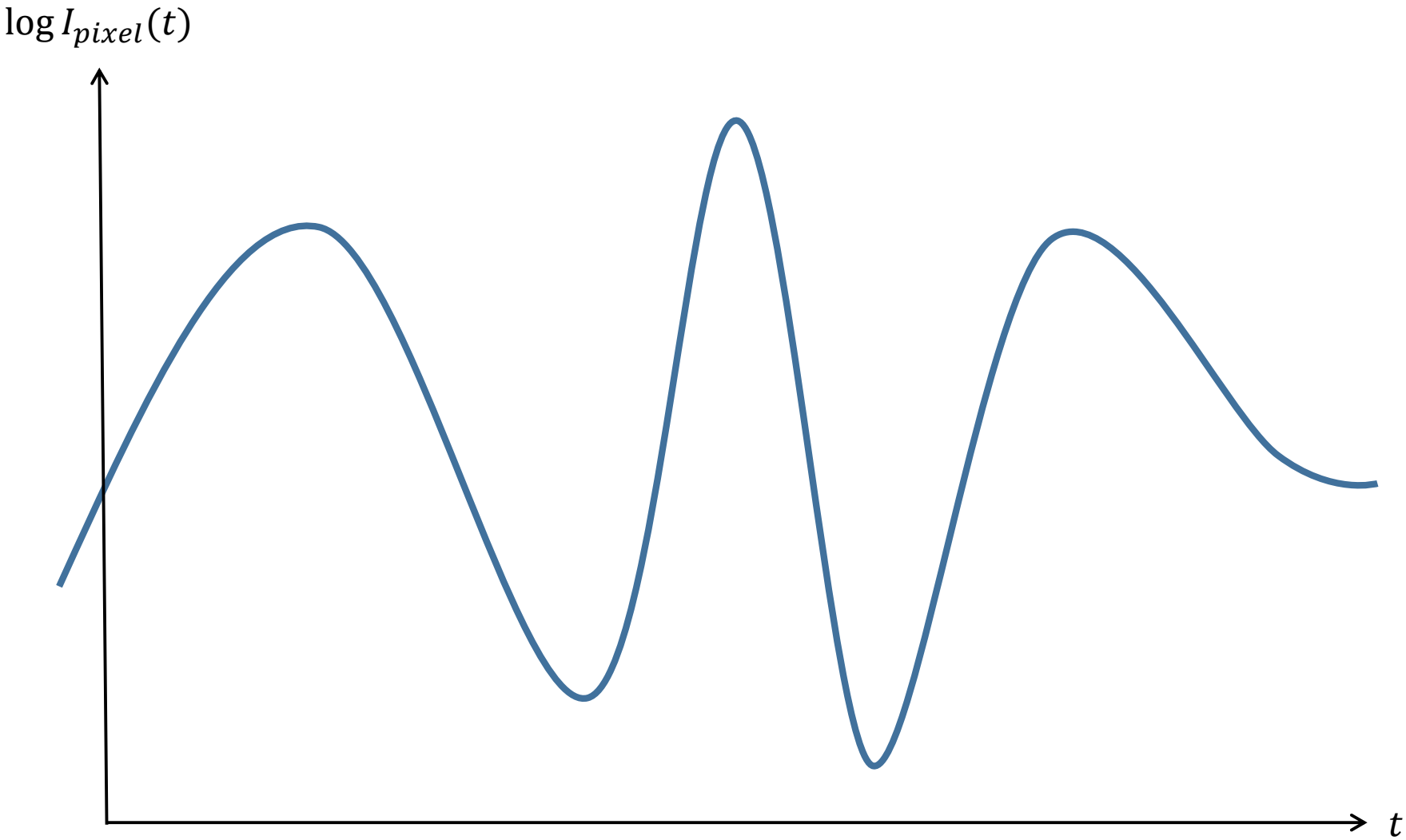
Traditional sampling



Traditional sampling

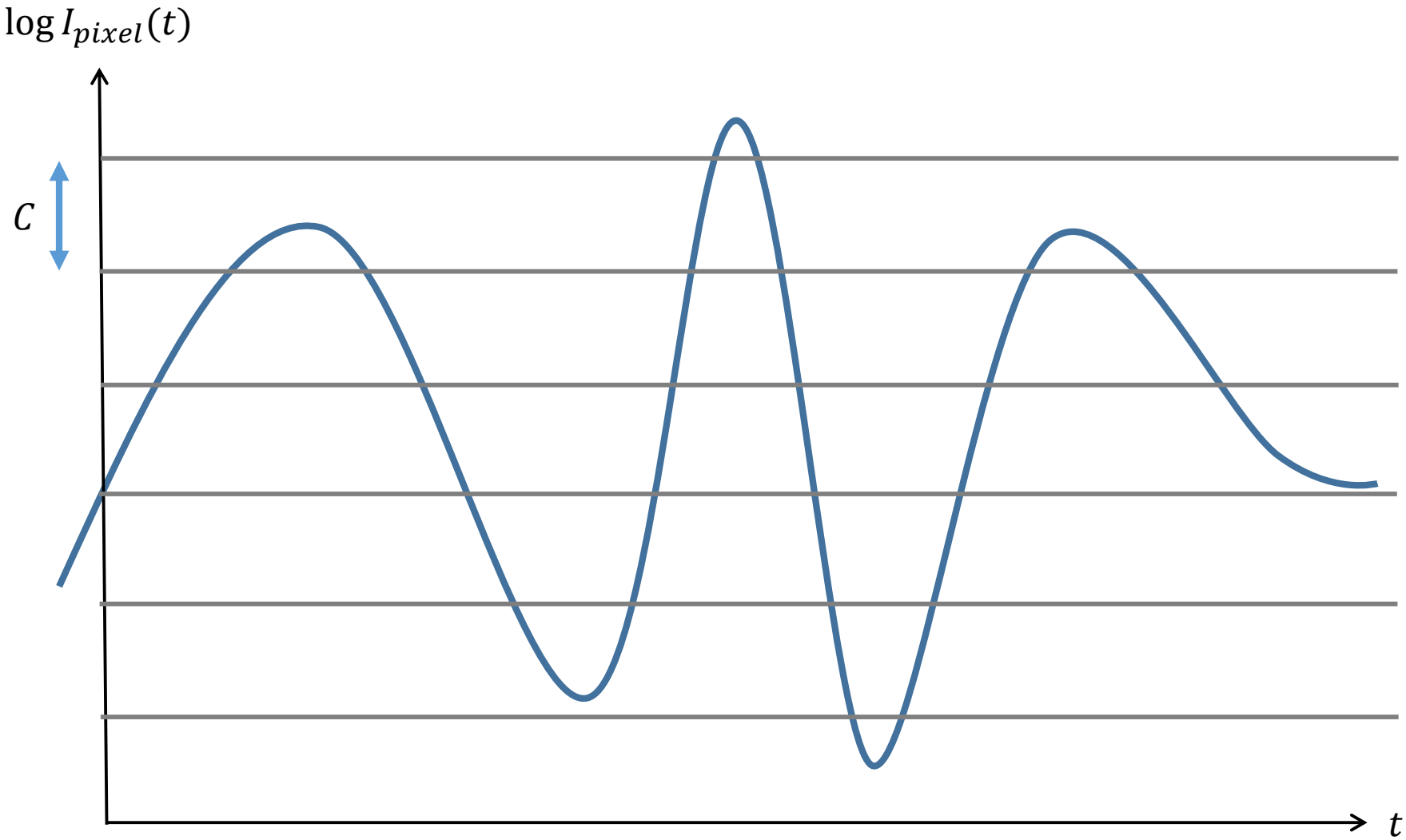


Level-crossing sampling



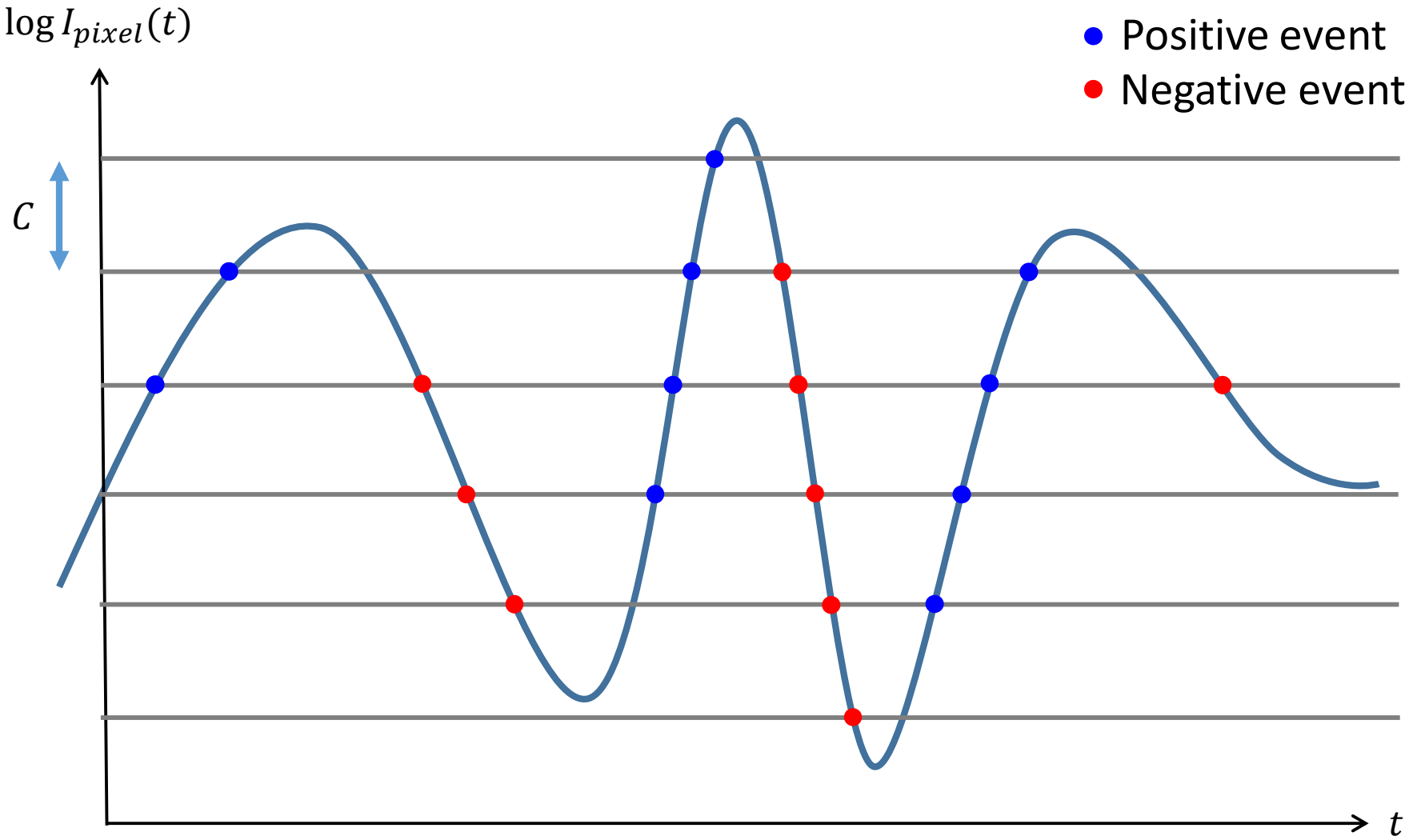
Level-crossing sampling

- An **event** is generated when the signal *change* exceeds C



Level-crossing sampling

- An **event** is generated when the signal *change* exceeds C



Outline

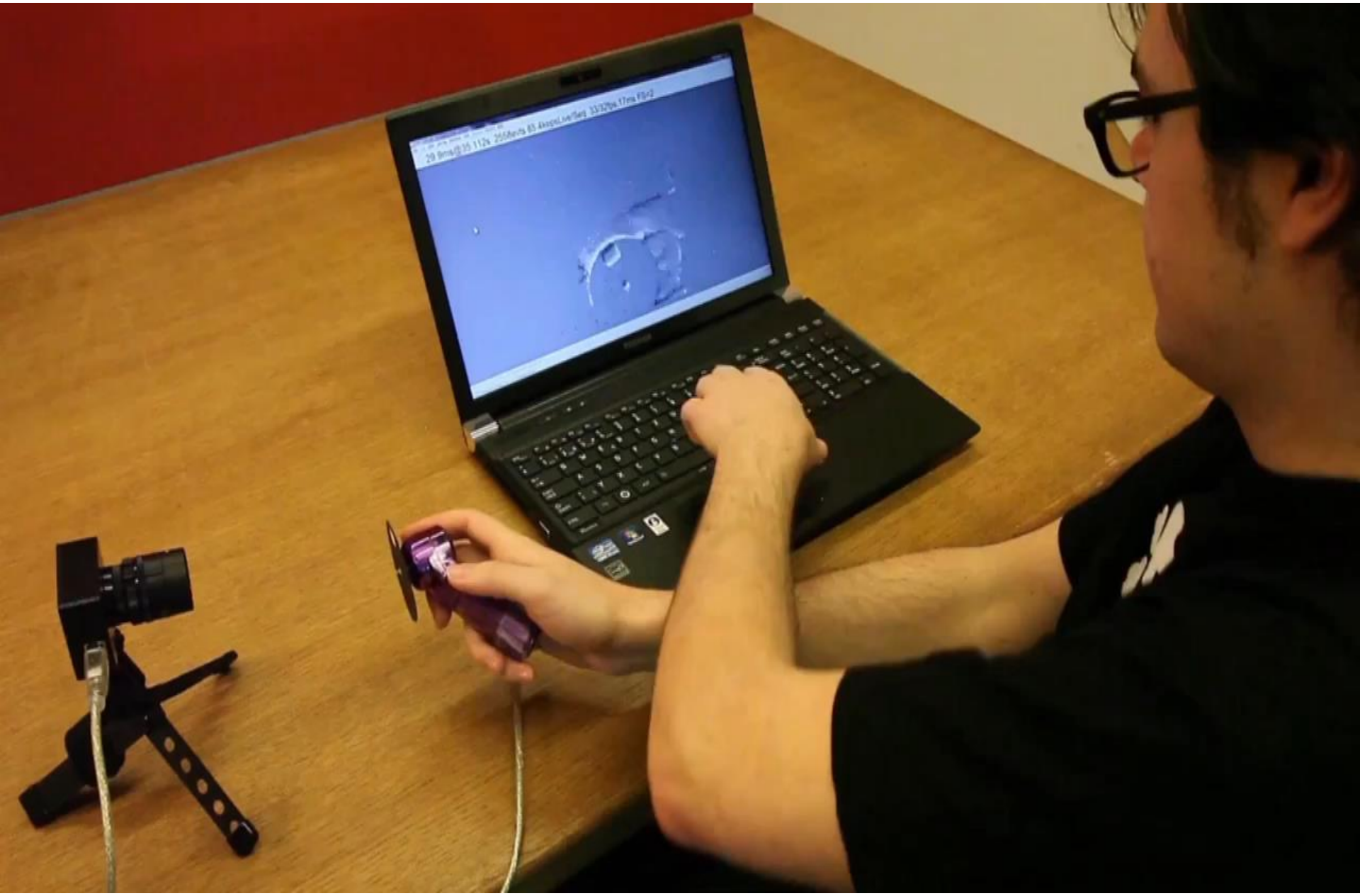
- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level-crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

Current Applications of event cameras

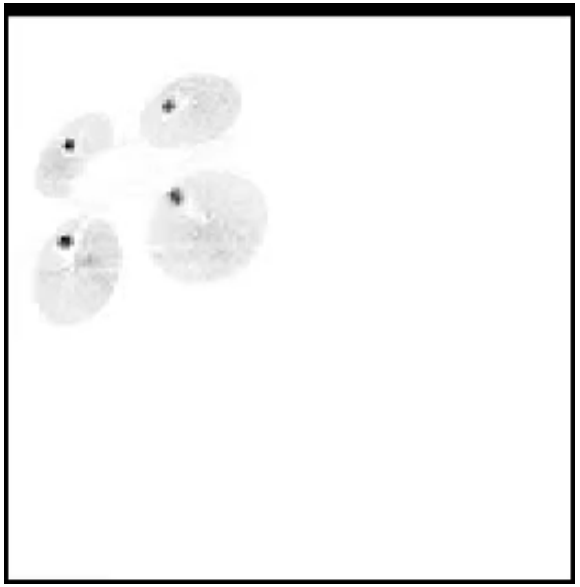
- Low-power Monitoring and Video Surveillance:
  - Traffic and moving object detection and tracking
- Fast closed-loop control
- High-dynamic range imaging
- Low-power gesture recognition (IBM TV control)
- High speed flow speed estimation
- Robust visual SLAM: low-power, HDR, and high speed applications

http://inilabs.com/videos/

The DVS is fast



The DVS is fast



1 frame = 33 ms

1 frame = 1 ms

1 frame = 0.05 ms

Censi, Delbruck, Scaramuzza, *Low-latency localization by active led markers tracking using a dynamic vision sensor*. IROS 2013

High-speed cameras vs DVS

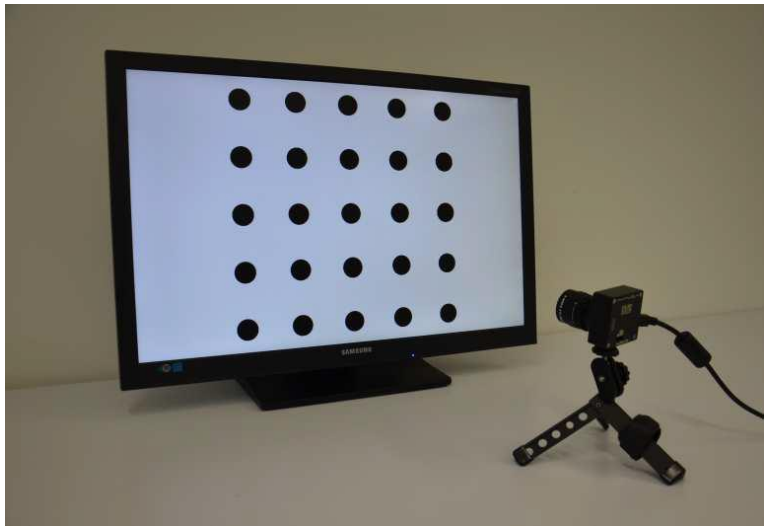
	Photron Fastcam SA5	Matrix Vision Bluefox	DVS
Max fps or measurement rate	1MHz	90 Hz	**1MHz**
Resolution at max fps	64x16 pixels	752x480 pixels	**346x260 pixels**
Bits per pixels	12 bits	8-10	**1 bits**
Weight	6.2 Kg	30 g	**30 g**
Active cooling	yes	No cooling	**No cooling**
Data rate	1.5 GB/s	32MB/s	**~1MB/s on average**
Power consumption	150 W + llighting	1.4 W	**20 mW**
Dynamic range	n.a.	60 dB	**140 dB**

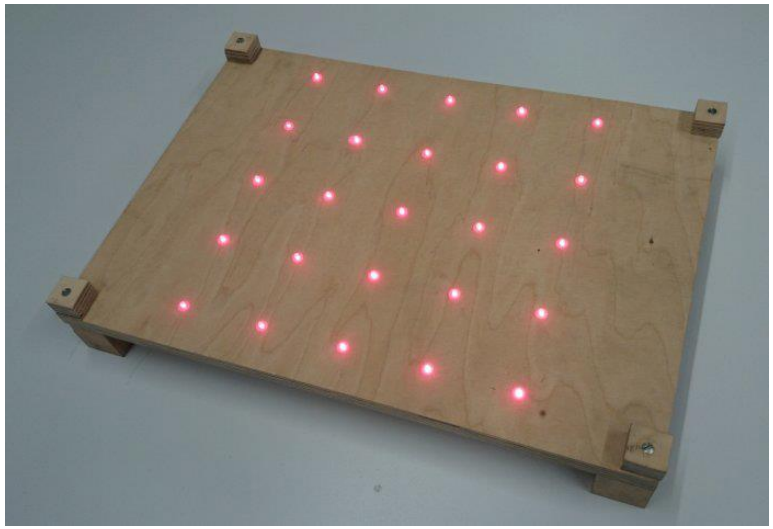
Outline

- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level-crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

Calibration of a DVS [IROS'14]

- Standard **pinhole camera model** still valid (same optics)
- Standard passive calibration patterns **cannot be used**
  - need to move the camera → inaccurate corner detection
- **Blinking patterns** (computer screen, LEDs)
- ROS DVS driver + intrinsic and extrinsic stereo calibration **open source**: https://github.com/uzh-rpg/rpg_dvs_ros





Outline

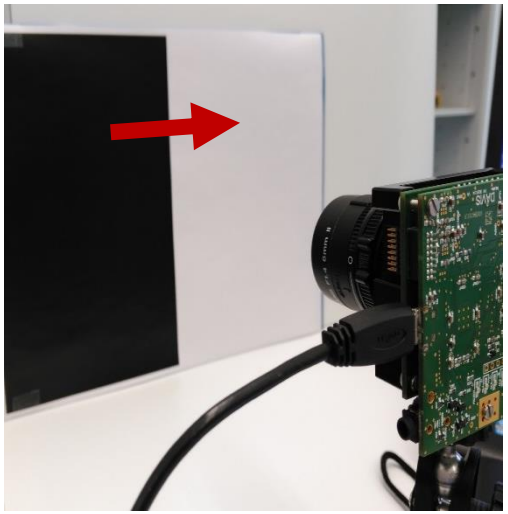
- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level-crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

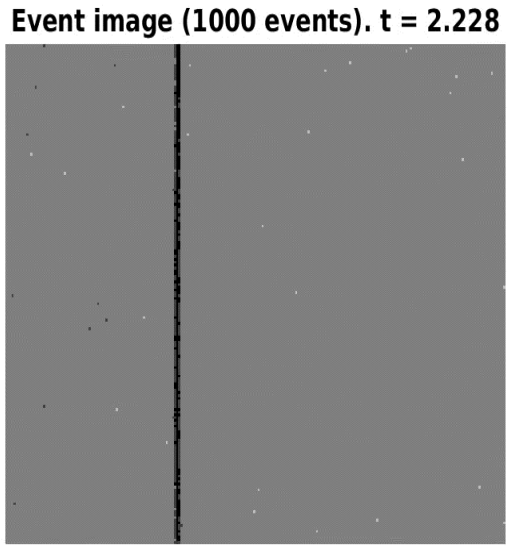
A Simple Optical Flow Algorithm

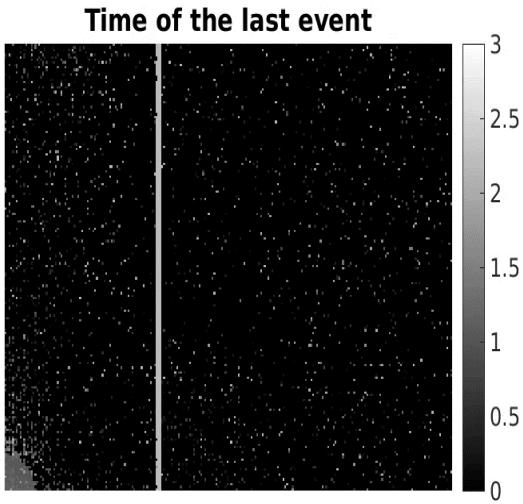
A moving edge

Horizontal motion

White pixels become black → brightness decrease → negative events (in black color)

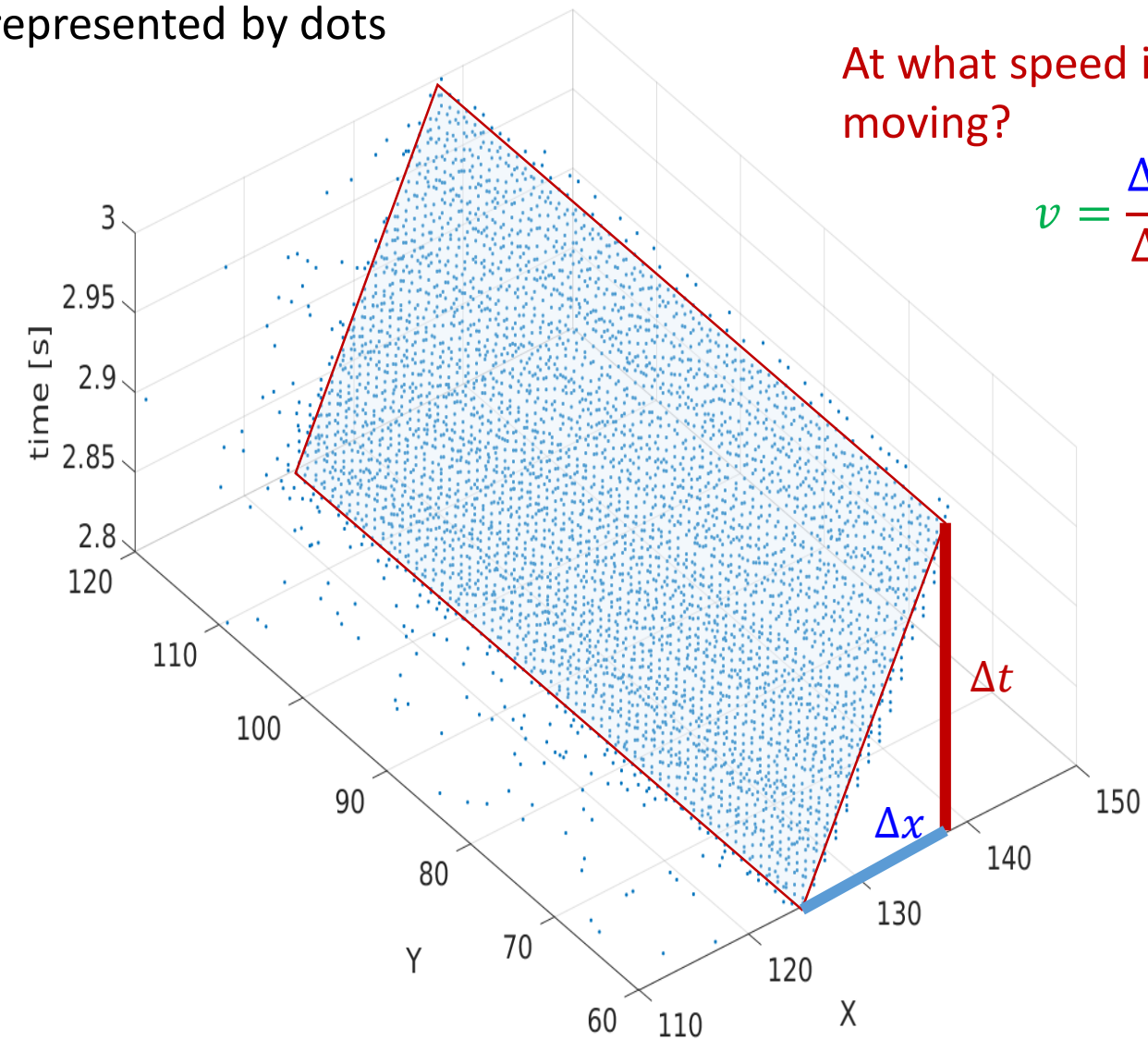






A moving edge

The same edge, visualized in space-time.
Events are represented by dots



At what speed is the edge moving?

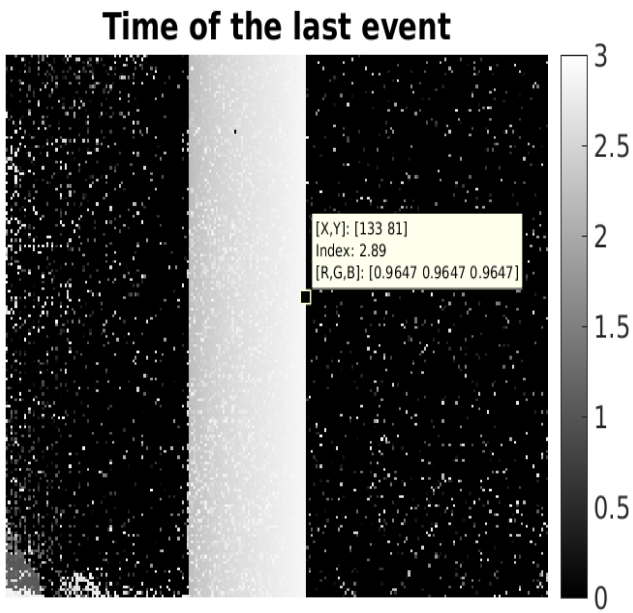
$$v = \frac{\Delta x}{\Delta t}$$

A moving edge

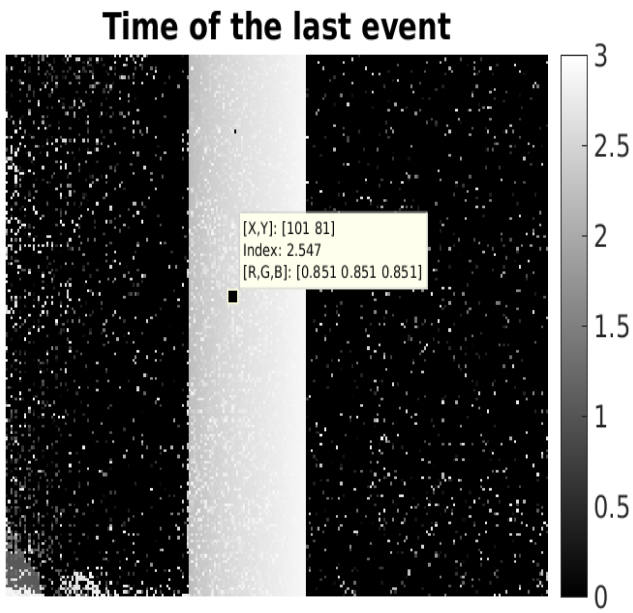
Speed of the edge:

$$v = \frac{(133 - 101)}{(2.89 - 2.547)} = 93.3\ pix/sec$$

At t = 2.89, X = 133



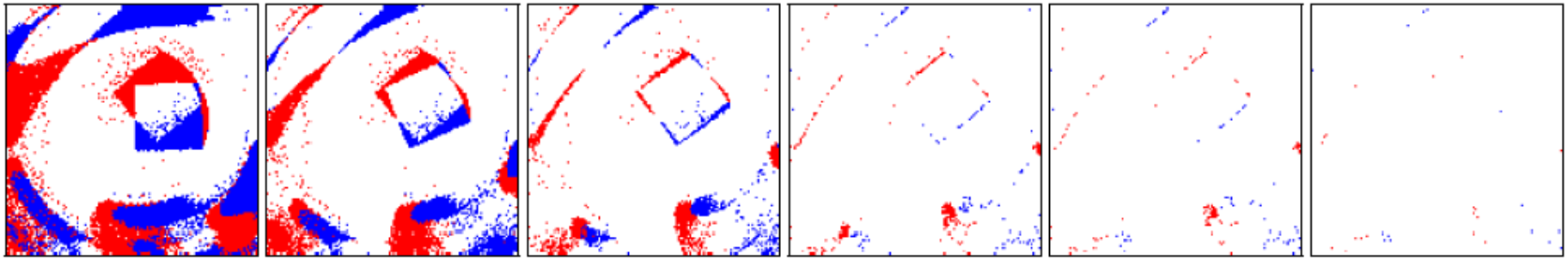
At t = 2.547, X = 101



Outline

- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level-crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

How many events should be used?



(a) $\Delta T = 33\,\text{ms}$ (b) $\Delta T = 15\,\text{ms}$ (c) $\Delta T = 5\,\text{ms}$ (d) $\Delta T = 1\,\text{ms}$ (e) $\Delta T = 0.5\,\text{ms}$ (f) $\Delta T = 0.1\,\text{ms}$

- **Event-by-event processing** (i.e., estimate the state **event by event**)
  - Pros: low latency (in principle down to microseconds)
  - Cons: with high-speed motion -> dozens of millions of events per seconds → GPU

- **Event-packet processing** (i.e., **process the last N events**)
  - Pros: N ca be tuned to allow real-time performance on a CPU
  - Cons: no longer microsecond resolution (when is this really necessary?)

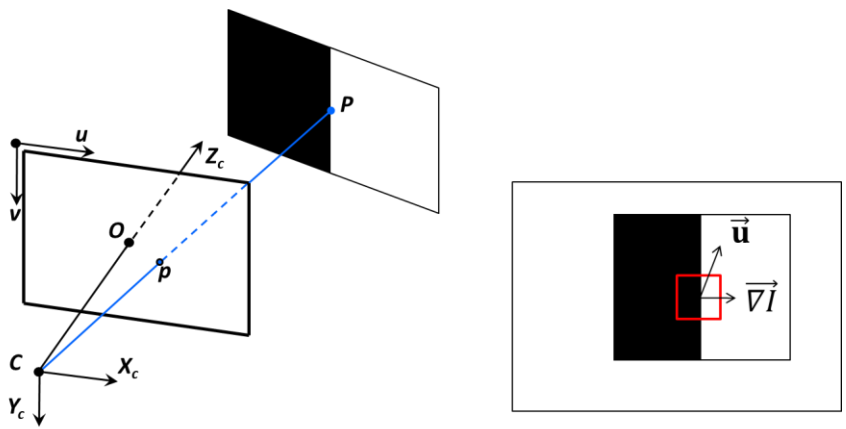
Censi, Strubel, Brandli, Delbruck, Scaramuzza, Low-latency localization by Active LED Markers tracking using a Dynamic Vision Sensor, IROS'13

Event-by-Event based Processing

Event Generation Model

- To simplify the notation, let's assume from now on that $I(x,y,t) = Log(I(x,y,t))$
- Consider a given pixel $p(x,y)$ moving with apparent motion $\mathbf{u} = (u,v)$ (i.e., induced by a moving 3D patch $\mathbf{P}$).
- It can be shown that an event is generated if the scalar product between the the gradient vector $\nabla I(x,y)$ and the apparent motion vector $\mathbf{u} = (u,v)$ is equal to C:

$$-\nabla I \cdot \mathbf{u} = C$$



- Censi & Scaramuzza, Low Latency, Event-based Visual Odometry, ICRA'14
- Gallego, Lund, Mueggler, Rebecq, Delbruck, Scaramuzza, Event-based, 6-DOF Camera Tracking from Photometric Depth Maps, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.

Proof

The proof comes from the brightness constancy assumption, which says that the intensity value of p, before and after the motion, must remained unchanged:

$$I(x, y, t) = I(x + u, y + v, t + \Delta t)$$

By replacing the right-hand term by its 1st order approximation at $t + \Delta t$, we get:

$$I(x, y, t) = I(x, y, t + \Delta t) + \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial x} v$$

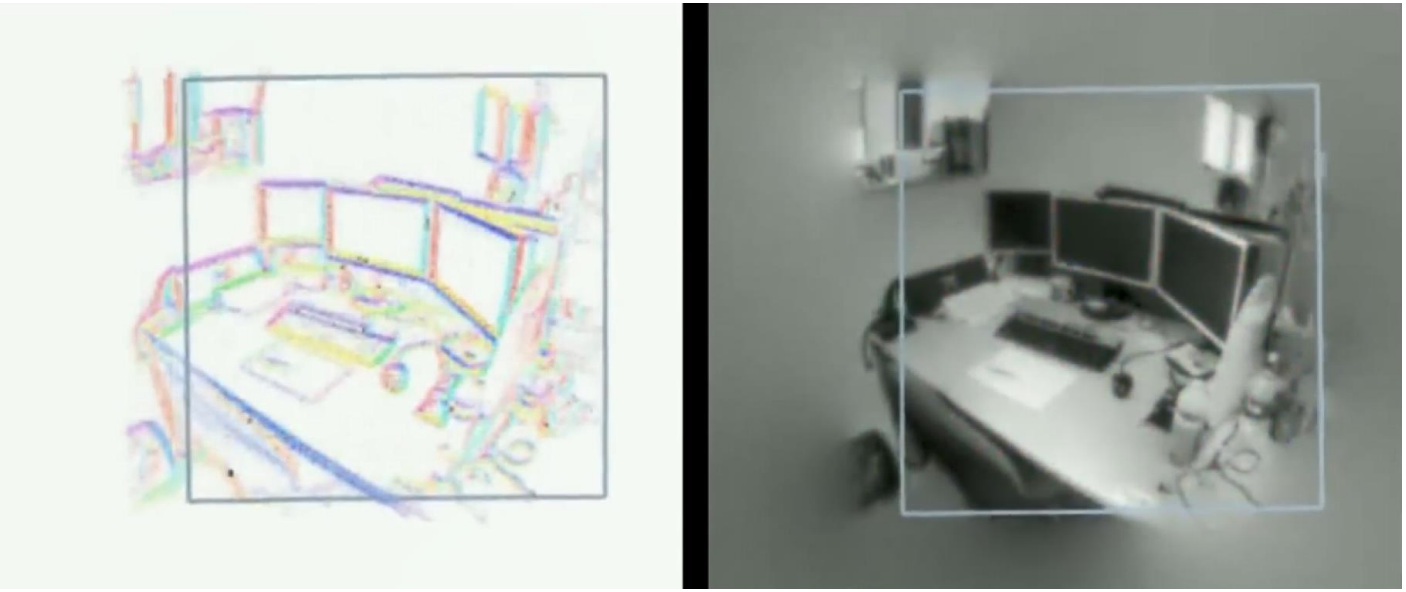
$$\Rightarrow \; I(x, y, t + \Delta t) - I(x, y, t) = -\frac{\partial I}{\partial x} u - \frac{\partial I}{\partial x} v$$

$$\Rightarrow \; \Delta I = \boxed{C = -\nabla I \cdot \mathbf{u}}$$

What if we do not know ∇I and $\mathbf{u}$?

This equation describes the **linearized** event generation equation for an event generated by a gradient ∇I that moved by a motion vector $\mathbf{u}$ (optical flow) during a time interval Δt.

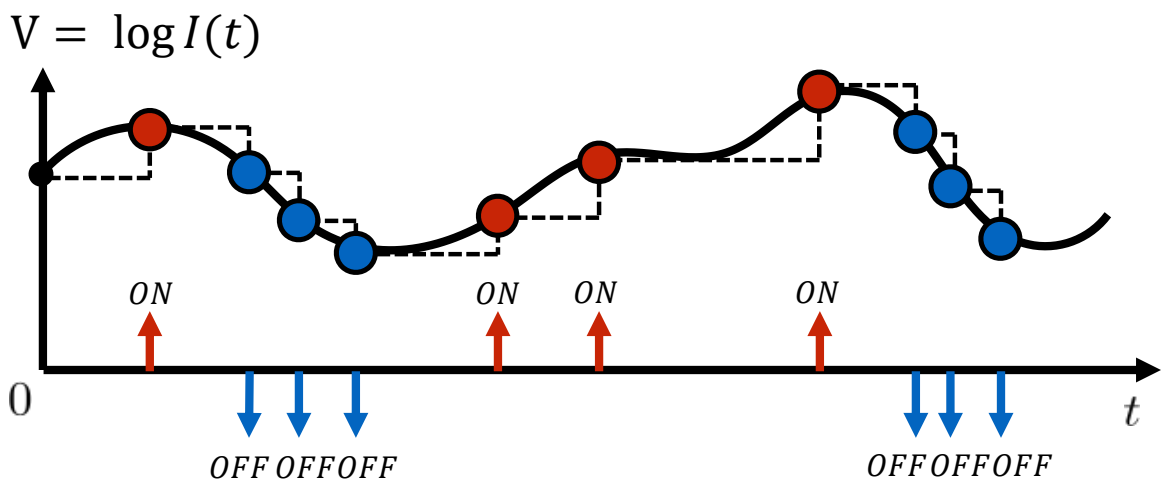
Case Study 1:
Image Intensity Reconstruction



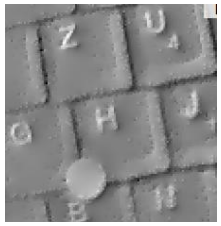
Kim, Handa, Benosman, Ieng, Davison, Simultaneous Mosaicing and Tracking with an Event Camera, BMVC'15

Image reconstruction

Recall: events are generated any time a single pixel sees a change in brightness equal to C



The intensity signal at the event time can be reconstructed by integration of $\pm C$



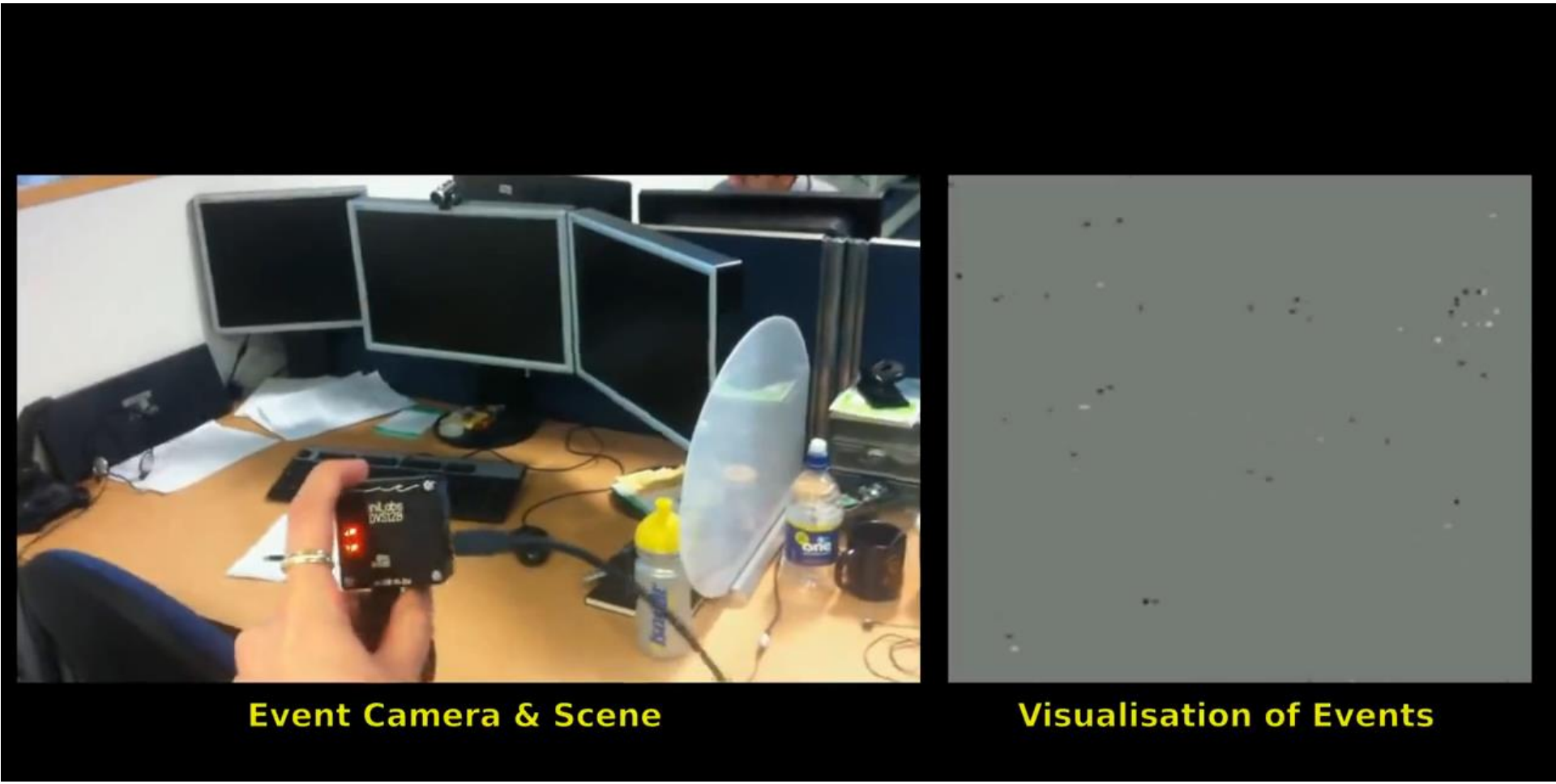
[Cook et al., IJCNN'11]

[Kim et al., BMVC'14]

- Cook, Gugelmann, Jug, Krautz, Steger, *Interacting Maps for Fast Visual Interpretation, Cook et al., IJCNN'11*
- Kim, Handa, Benosman, Ieng, Davison, Simultaneous Mosaicing and Tracking with an Event Camera, BMVC'14

Image reconstruction

Given the events and the camera motion (rotation), recover the absolute brightness.



Kim, Handa, Benosman, Ieng, Davison, Simultaneous Mosaicing and Tracking with an Event Camera, BMVC'14

Image reconstruction

- Given the events and the camera motion (rotation), recover the absolute brightness.
- How is it possible?
- Intuitive explanation: an event camera naturally responds to edges, hence, if we know the motion, we can relate the events to "world coordinates" to get an edge/gradient map. Then, just integrate the gradient map to get absolute intensity.

Steps: 1. Recover the gradient map of the scene

2. Integrate the gradient to obtain brightness

Events

Camera motion $R(t)$

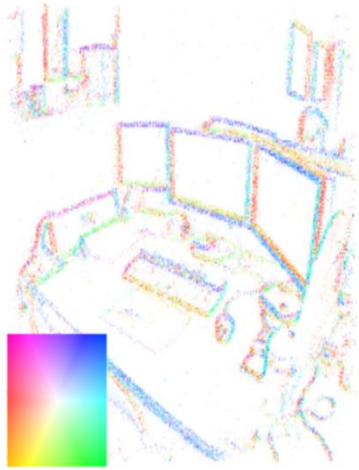


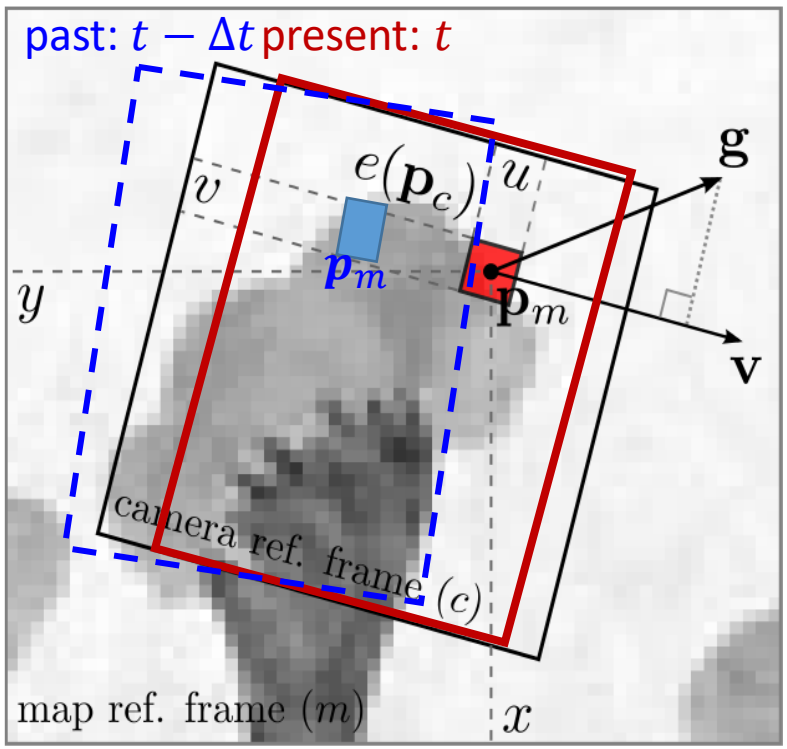
Image gradient
(magnitude and direction)

solve Poisson Eq.

[Kim et al., BMVC'14]

Image reconstruction. Step 1: compute gradient map

Brightness map M



[Kim et al., BMVC'14]

Event generated due to brightness change of size C.

Let $L = \log I$,

$$\Delta L(t) \equiv L(t) - L(t - \Delta t) = C$$

In terms of the brightness map $M(x, y)$ (panorama):

$$M(\boldsymbol{p}_m(t)) - M\big(\boldsymbol{p}_m(t - \Delta t)\big) = C$$

Using Taylor 1st order approximation:

$$\boxed{M\big(\boldsymbol{p}_m(t)\big) - M\big(\boldsymbol{p}_m(t - \Delta t)\big) \approx \boldsymbol{g} \cdot \boldsymbol{v}\Delta t}$$

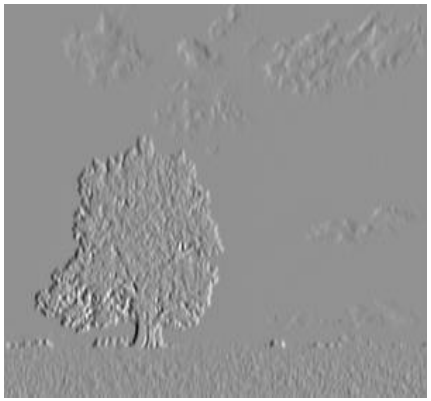
With brightness gradient $\boldsymbol{g} = \nabla M\big(\boldsymbol{p}_m(t)\big)$

Displacement: $\boldsymbol{v}\Delta t = \big(\boldsymbol{p}_m(t) - \boldsymbol{p}_m(t - \Delta t)\big)$

Image reconstruction. Step 2: Poisson reconstruction

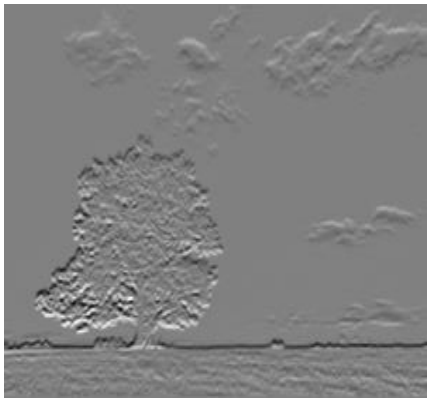
Integrate gradient map $\boldsymbol{g}$ to get absolute brightness M

Original Image



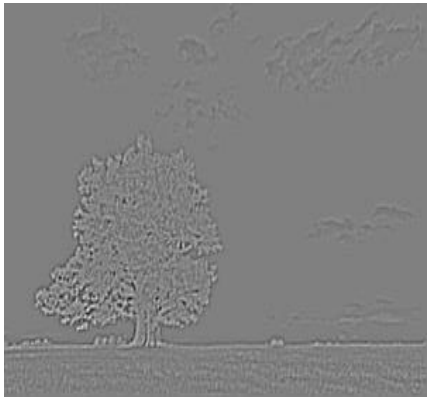
Gradient in *x* direction

$(g_x = \partial_x I)$



Gradient in *y* direction

$(g_y = \partial_y I)$



Divergence

$(\operatorname{div} \boldsymbol{g} = \partial_x g_x + \partial_y g_y)$

Solve Poisson eq:

$(\Delta \tilde{I} = \operatorname{div} \boldsymbol{g})$

Fast using the FFT

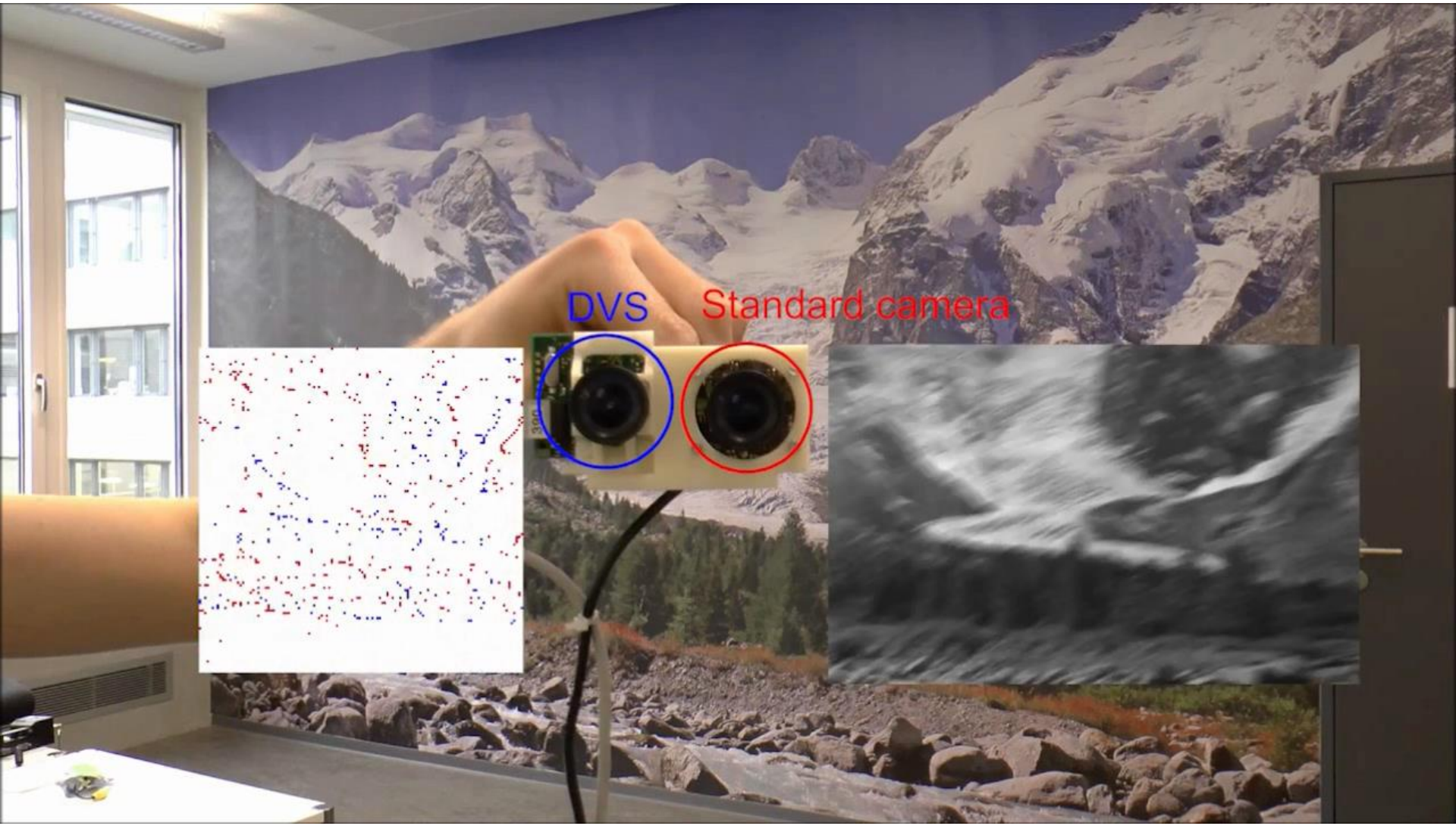
Poisson Image Reconstruction

Reconstructed Image

Case Study 2:
6-DoF Pose Tracking from a Photometric Depth Map

Gallego, Lund, Mueggler, Rebecq, Delbruck, Scaramuzza, Event-based, 6-DOF Camera Tracking from Photometric Depth Maps, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.

Result: High speed 6DoF Camera Tracking



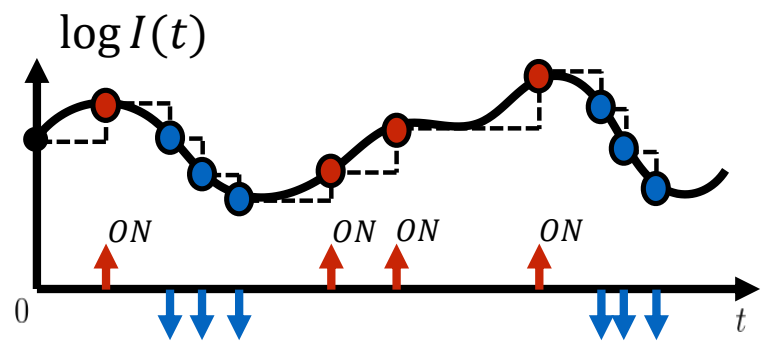
Gallego, Lund, Mueggler, Rebecq, Delbruck, Scaramuzza, Event-based, 6-DOF Camera Tracking from Photometric Depth Maps, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.

Methodology

- **Probabilistic** approach (Bayesian filter): $p(s|e) = p(e|s)p(s)$

  Posterior Likelihood Prior

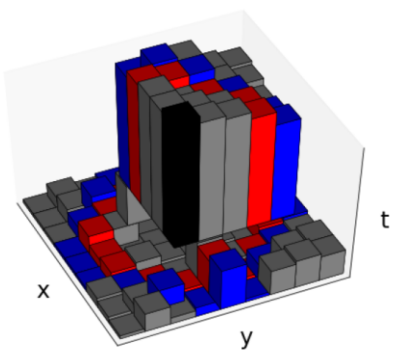
- **State vector**: $s = (R, T, C, \sigma_C, \rho)$
  - pose (R,T),
  - contrast mean value C
  - uncertainty σ_C,
  - inlier ratio ρ

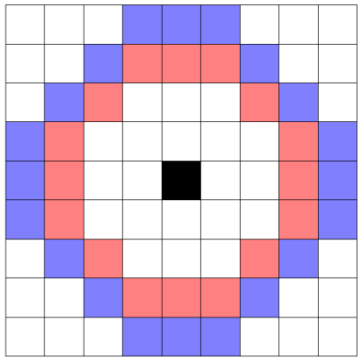


- Motion model: **random walk**
- **Robust sensor model** (likelihood)
  - **Measurement function** derived from generative event model: $-\nabla I \cdot \mathbf{u} = C$

Gallego, Lund, Mueggler, Rebecq, Delbruck, Scaramuzza, Event-based, 6-DOF Camera Tracking from Photometric Depth Maps, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.

Case Study 3:
Event-based Corner Detection

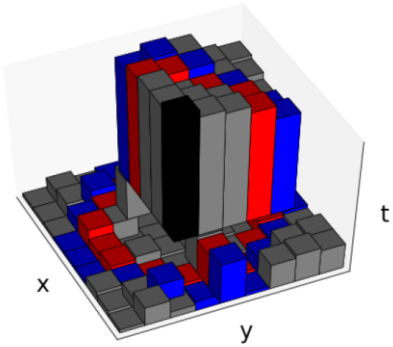


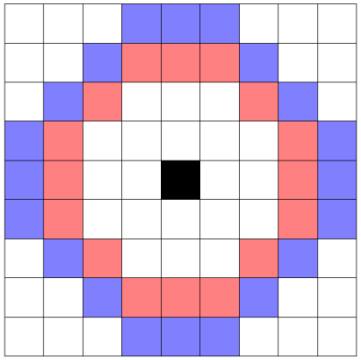


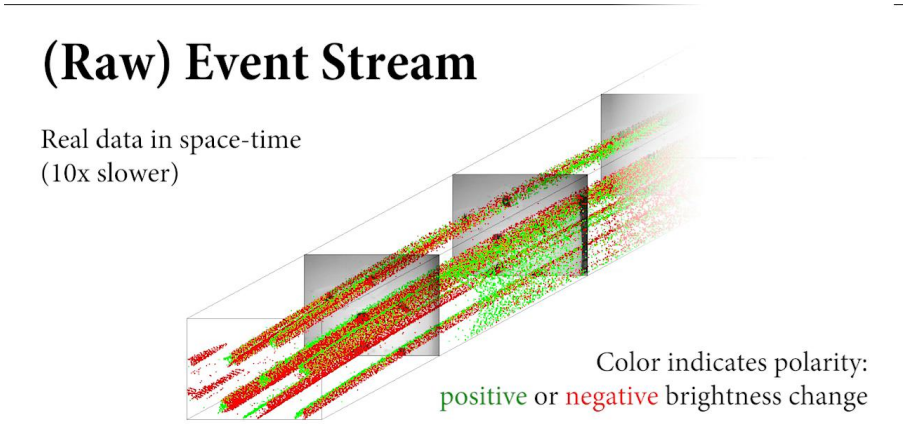
Mueggler, Bartolozzi, Scaramuzza, Fast Event-based Corner Detection, British Machine Vision Conference (BMVC), London, 2017

FAST-like Event-based Corner Detection

- Operates on Surface of Active Events





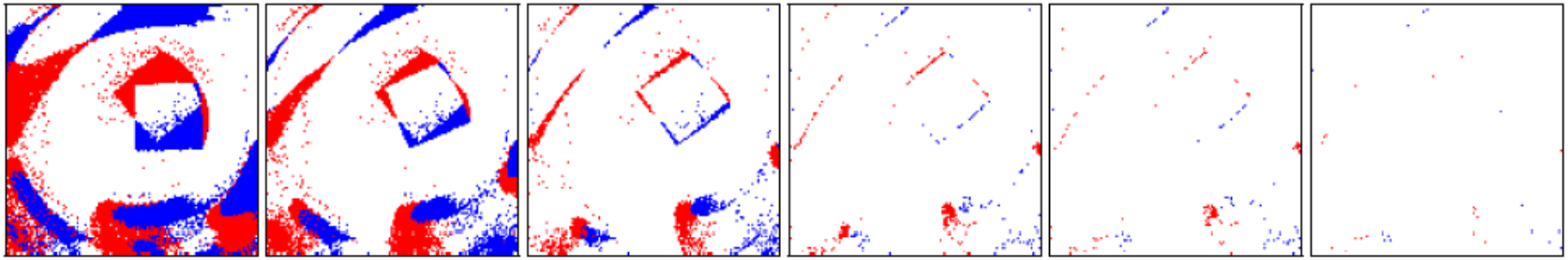


- The event is considered a corner if
  - 3-6 contiguous pixels on red ring are newer than all other pixels on the same ring and
  - 4-6 contiguous pixels on blue ring are newer than all other pixels on the same ring and

Mueggler, Bartolozzi, Scaramuzza, Fast Event-based Corner Detection, British Machine Vision Conference (BMVC), London, 2017

Event-packet based Processing

How many events should be used?



(a) $\Delta T = 33\,\text{ms}$ (b) $\Delta T = 15\,\text{ms}$ (c) $\Delta T = 5\,\text{ms}$ (d) $\Delta T = 1\,\text{ms}$ (e) $\Delta T = 0.5\,\text{ms}$ (f) $\Delta T = 0.1\,\text{ms}$

- **Event-by-event processing** (i.e., estimate the state **event by event**)
  - Pros: low latency (in principle down to microseconds)
  - Cons: with high-speed motion -> dozens of millions of events per seconds → GPU

- **Event-packet processing** (i.e., **process the last N events**)
  - Pros: N ca be tuned to allow real-time performance on a CPU
  - Cons: no longer microsecond resolution (when is this really necessary?)

Censi, Strubel, Brandli, Delbruck, Scaramuzza, Low-latency localization by Active LED Markers tracking using a Dynamic Vision Sensor, IROS'13

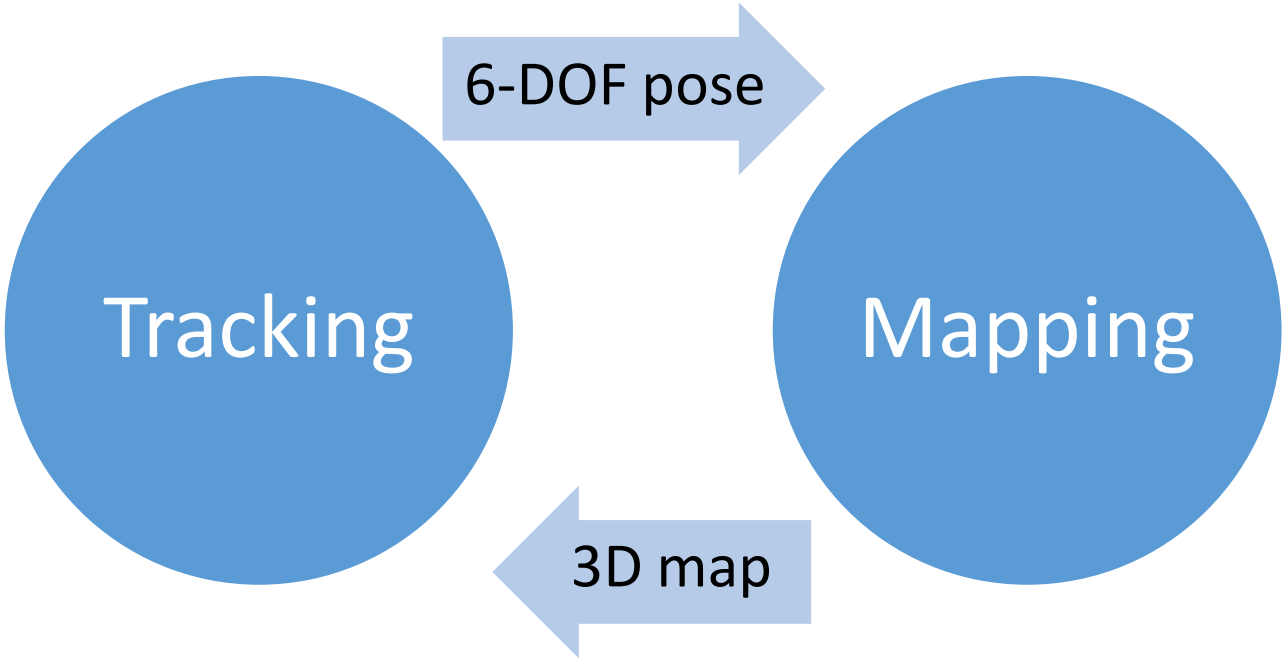
EVO:
A Geometric Approach to Event-based 6-DOF Parallel Tracking and Mapping in Real-time

Rebecq, Horstschäfer, Gallego, Scaramuzza

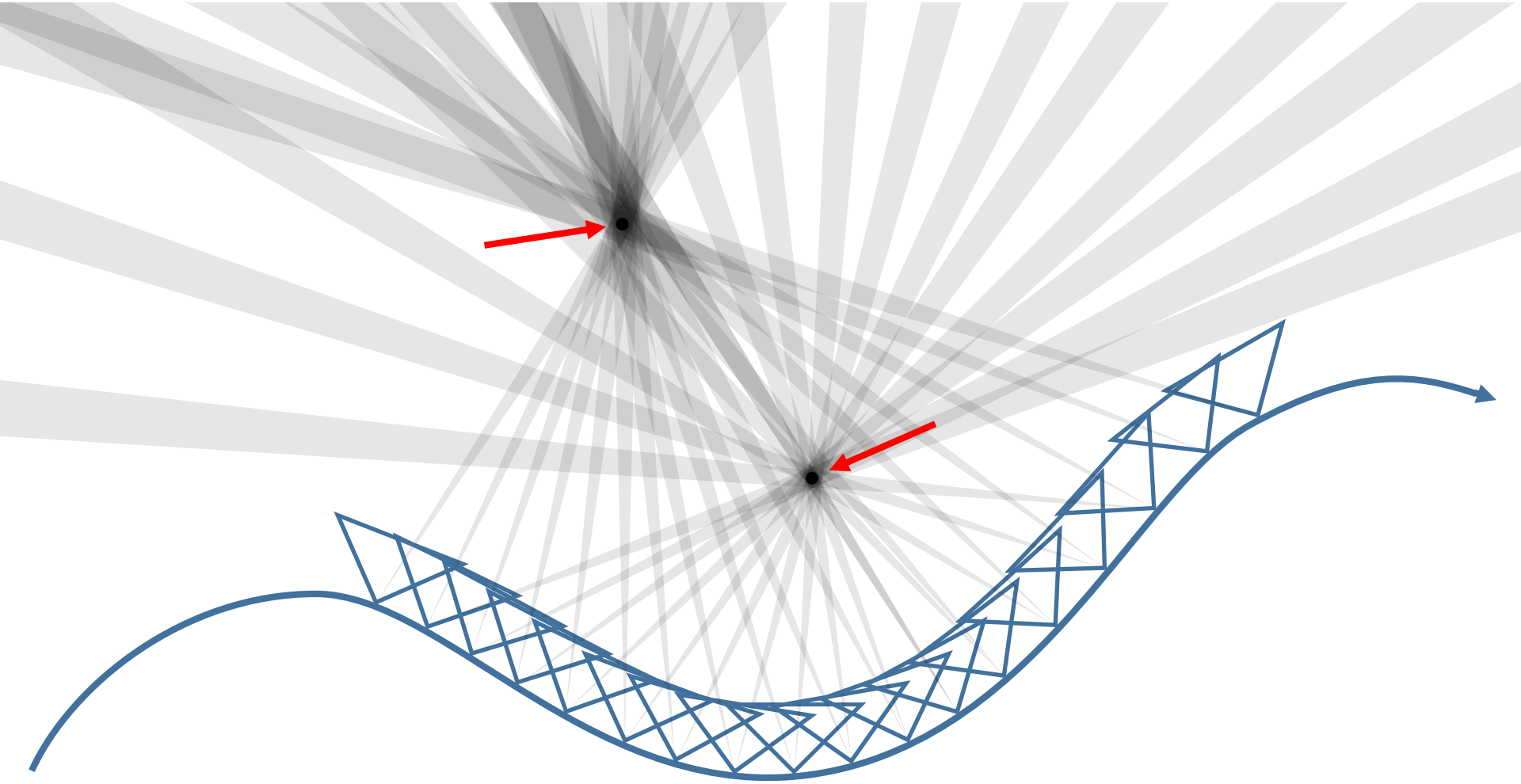
IEEE Robotics & Automation Letters, 01/2017 (presented at ICRA'17)

EU Patent 2017

Parallel Tracking and Mapping



How the 3D mapping works

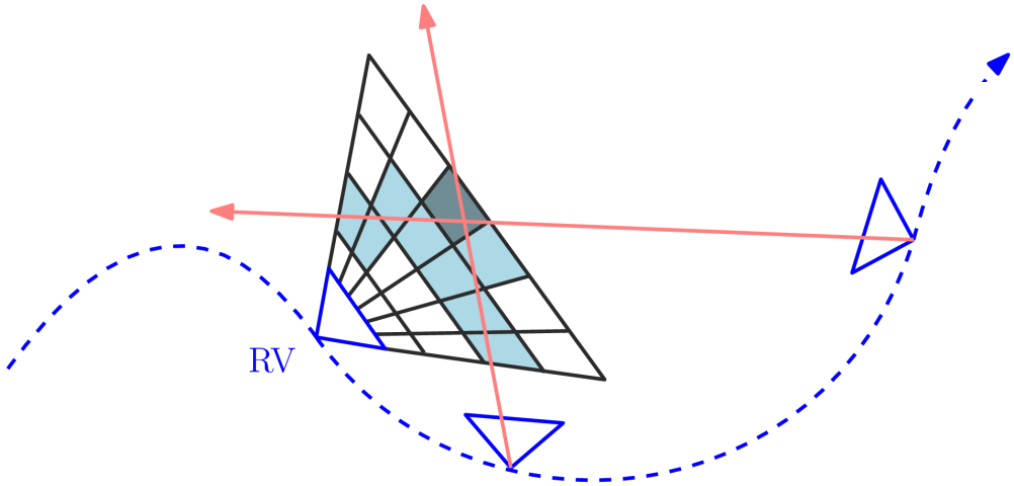


An event camera reacts to strong gradients in the scene

Areas of high ray-density likely indicate the presence of 3D structures

How the 3D mapping works

- Ray-density: Disparity Space Image (DSI)
- Projective sampling grid (DSI)
  + adaptive thresholding

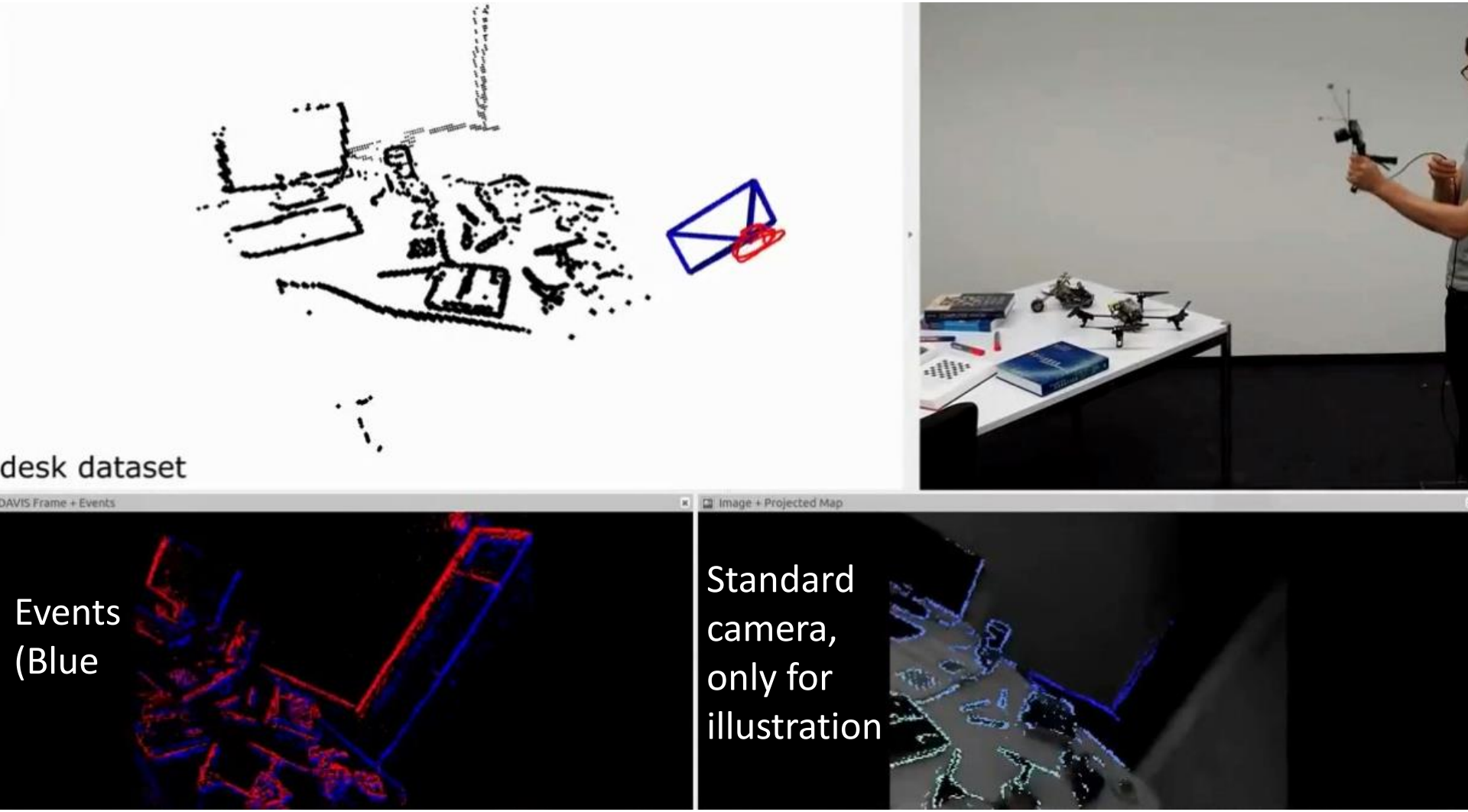


Non-uniform, projective grid,
centered on a reference viewpoint

240 x 180 x 100 voxels

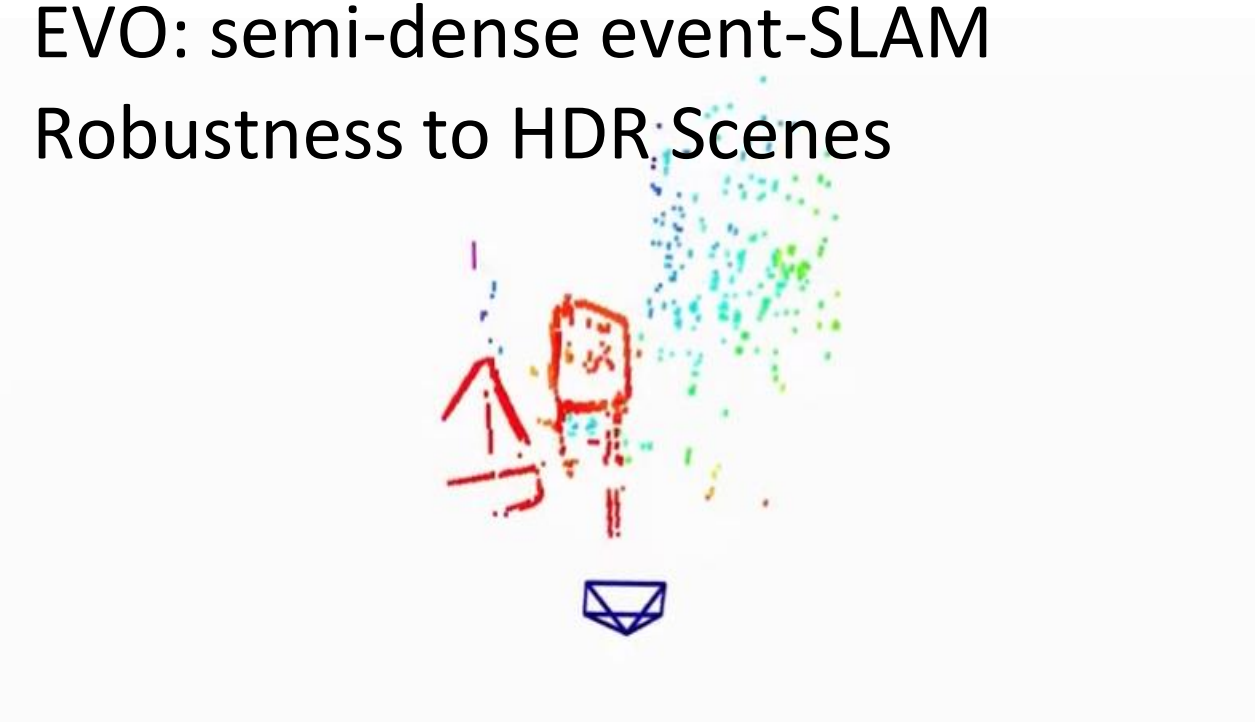
Rebecq, Gallego, Scaramuzza, EMVS: Event-based Multi-View Stereo, BMVC'16, Best Industry Paper Award

EVO: semi-dense event-SLAM



Rebecq, Horstschaefer, Gallego, Scaramuzza, EVO: A Geometric Approach to Event-based 6-DOF Parallel Tracking and Mapping in Real-time, IEEE Robotics and Automation Letters (RA-L), 2017

EVO: semi-dense event-SLAM
Robustness to HDR Scenes

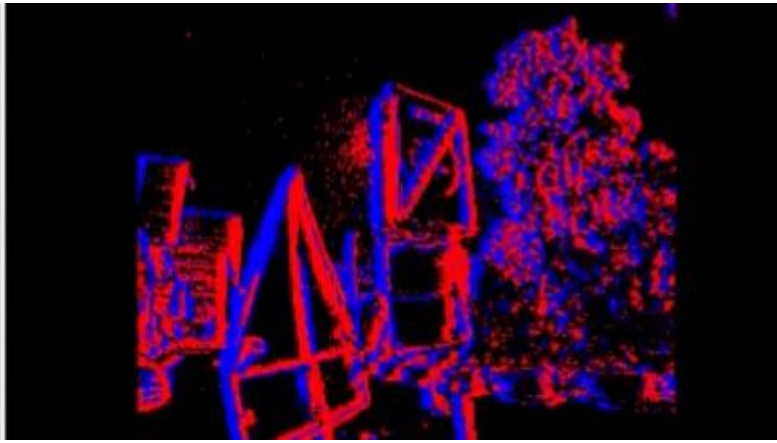


iPhone camera

Frame of a standard camera

Intensity reconstruction from events

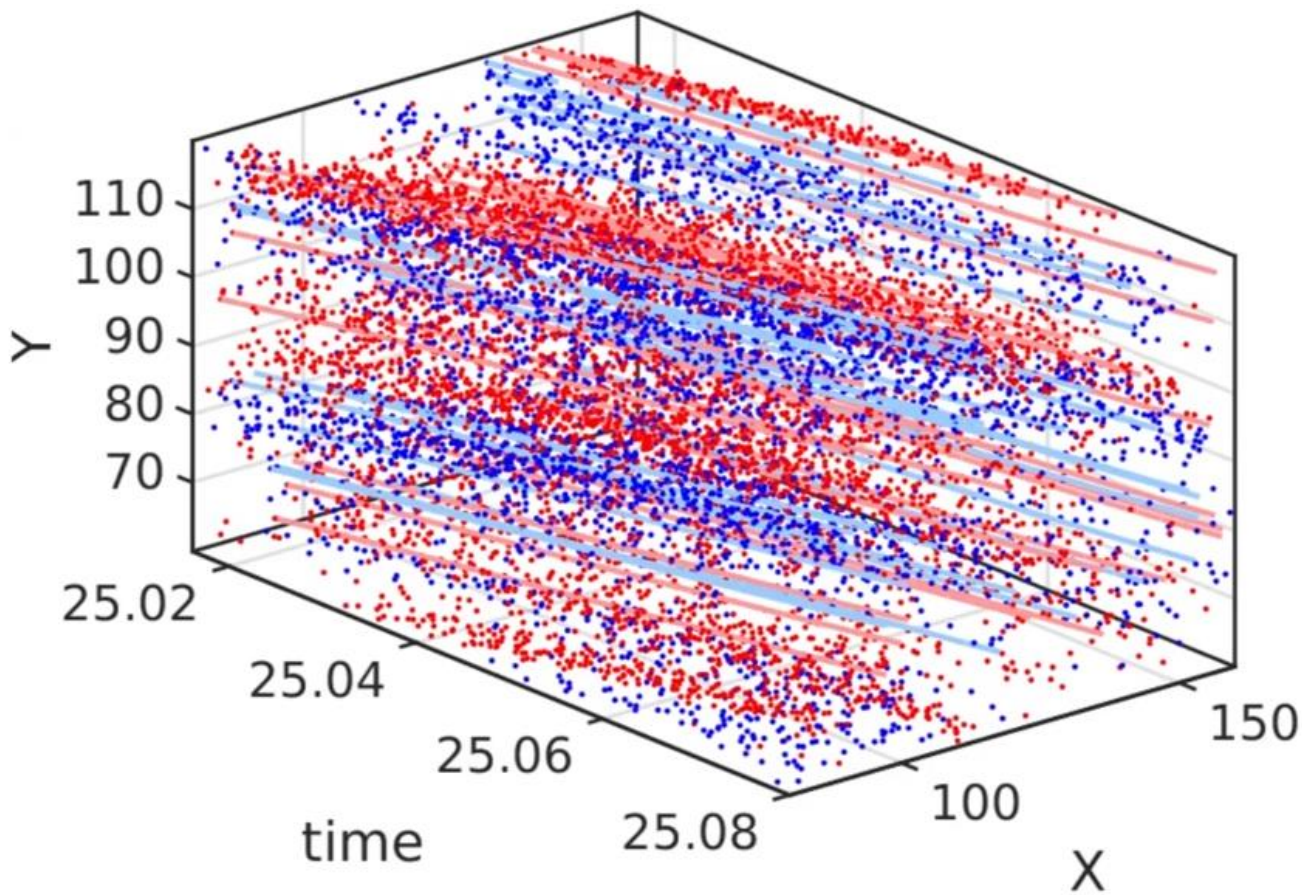
Events only



Rebecq, Horstschaefer, Gallego, Scaramuzza, EVO: A Geometric Approach to Event-based 6-DOF Parallel Tracking and Mapping in Real-time, IEEE Robotics and Automation Letters (RA-L), 2017

Motion-Estimation by Contrast Maximization

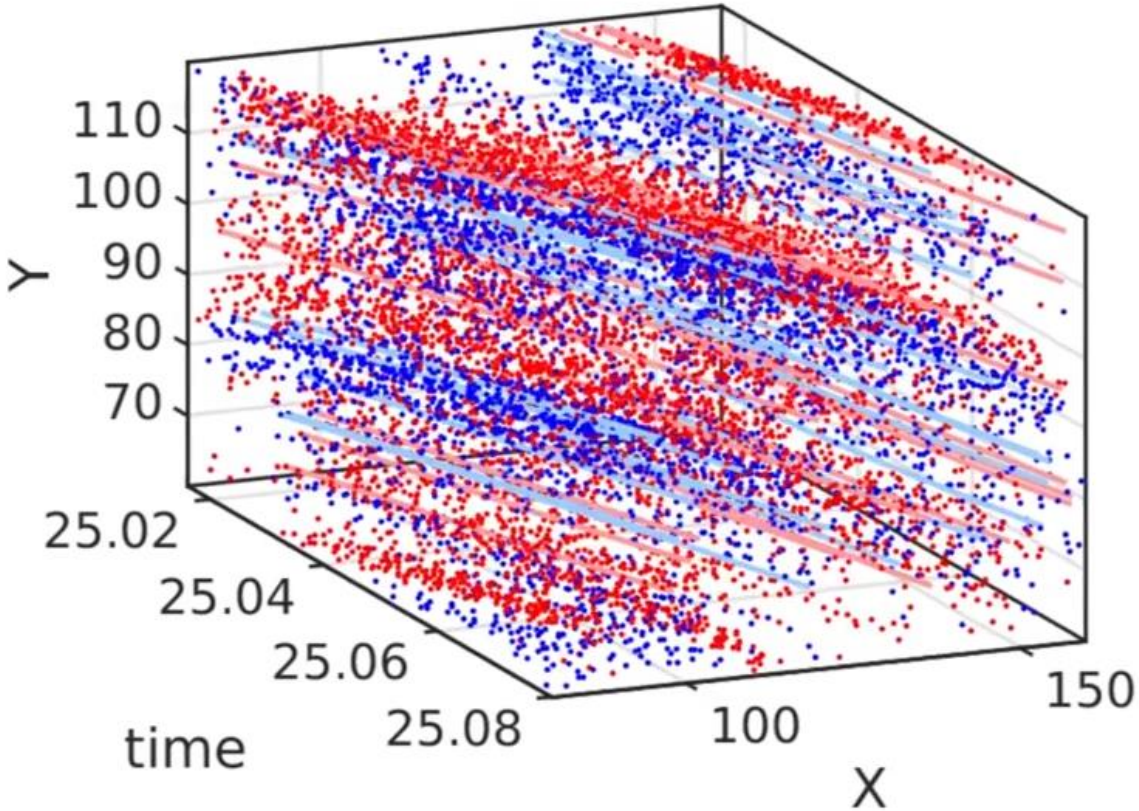
- Directly estimate the motion curves that align the events



Gallego, Scaramuzza, Accurate Angular Velocity Estimation with an Event Camera, IEEE RAL'17

Motion-Estimation by Contrast Maximization

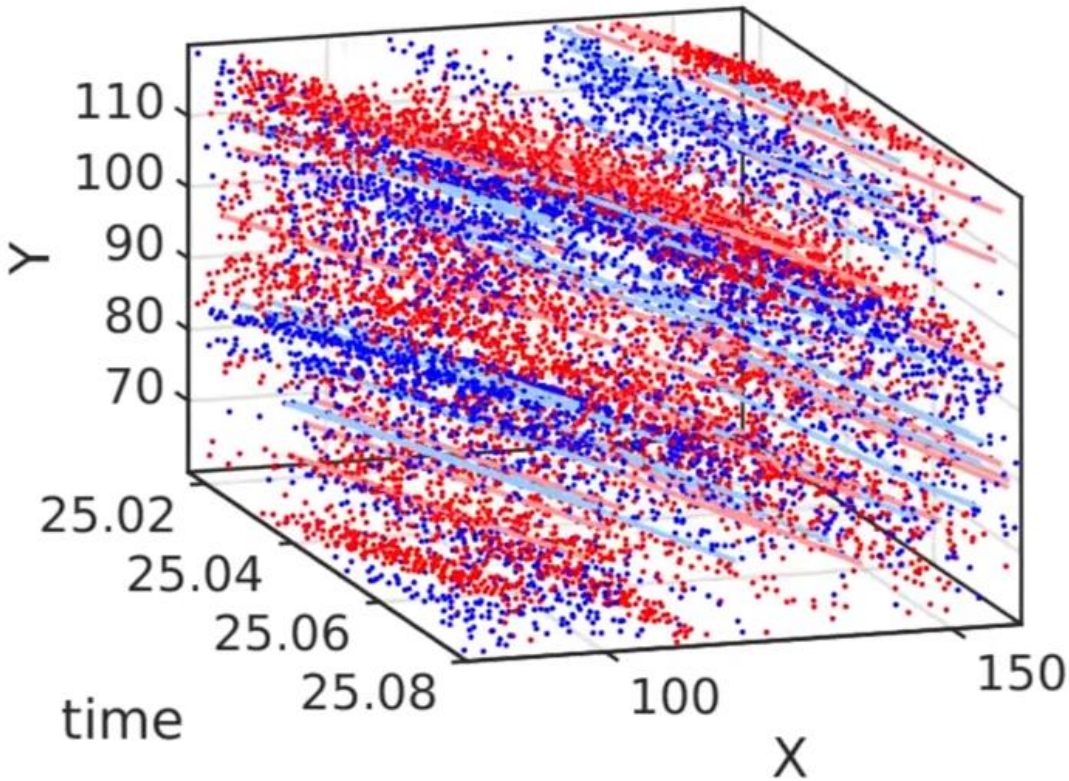
- Directly estimate the motion curves that align the events



Gallego, Scaramuzza, Accurate Angular Velocity Estimation with an Event Camera, IEEE RAL'17

Motion-Estimation by Contrast Maximization

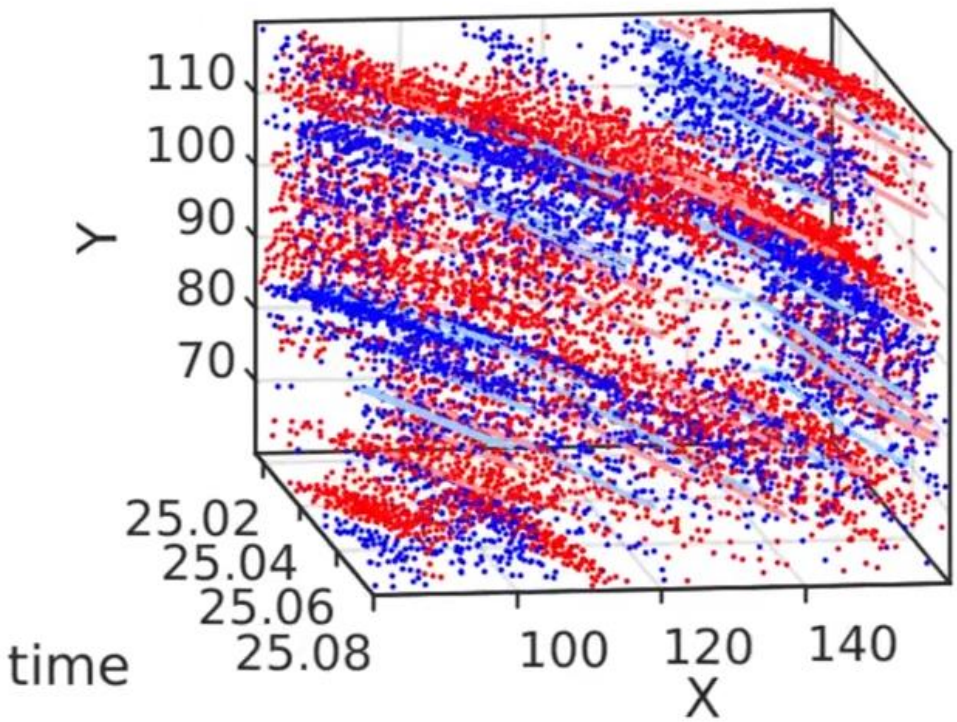
- Directly estimate the motion curves that align the events



Gallego, Scaramuzza, Accurate Angular Velocity Estimation with an Event Camera, IEEE RAL'17

Motion-Estimation by Contrast Maximization

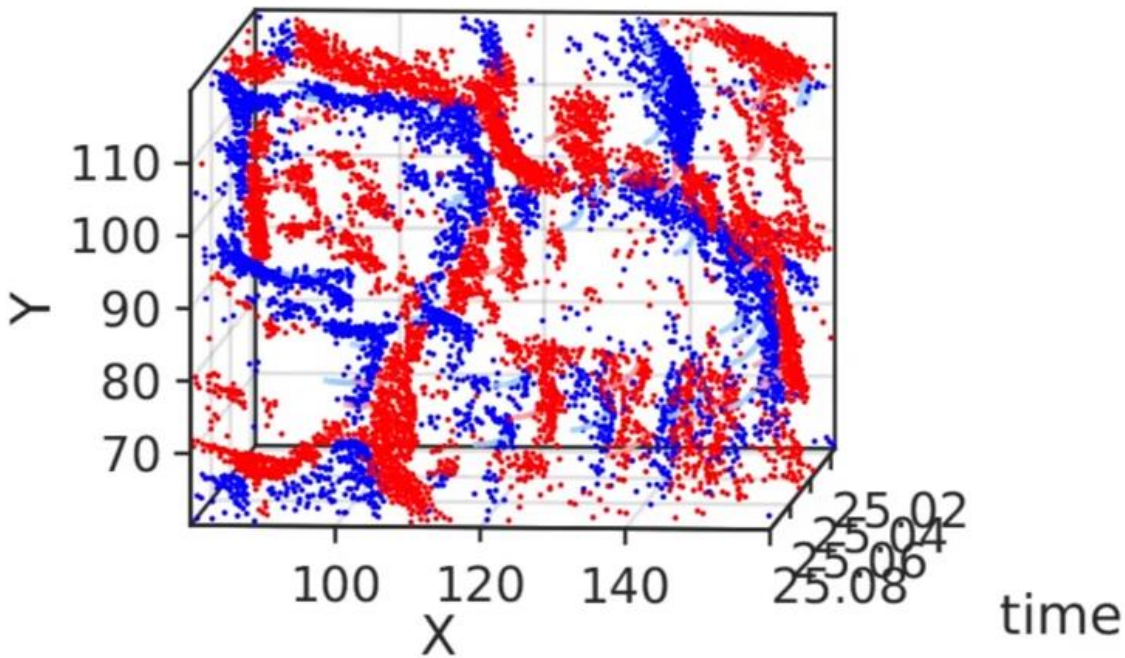
- Directly estimate the motion curves that align the events



Gallego, Scaramuzza, Accurate Angular Velocity Estimation with an Event Camera, IEEE RAL'17

Motion-Estimation by Contrast Maximization

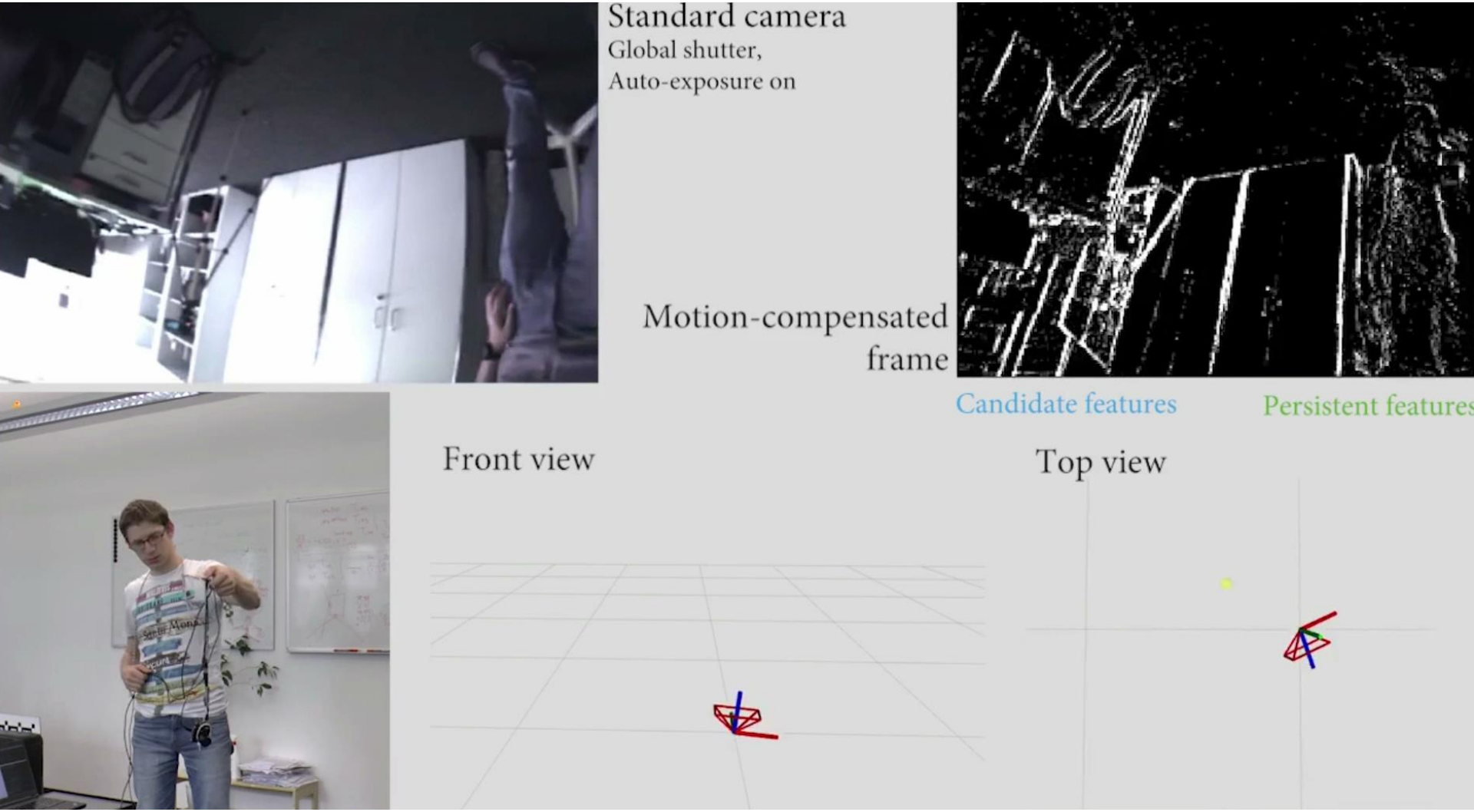
- Directly estimate the motion curves that align the events



Gallego, Scaramuzza, Accurate Angular Velocity Estimation with an Event Camera, IEEE RAL'17

Events + IMU based SLAM

Impact: visual SLAM works even when spinning and event camera attached to a leash



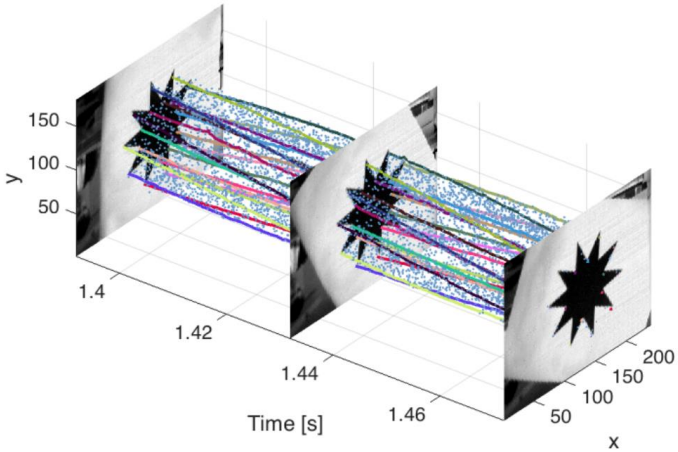
Rebecq, Horstschaefer, Scaramuzza, Real-time Visual-Inertial Odometry for Event Cameras using Keyframe-based Nonlinear Optimization, British Machine Vision Conference (BMVC), London, 2017

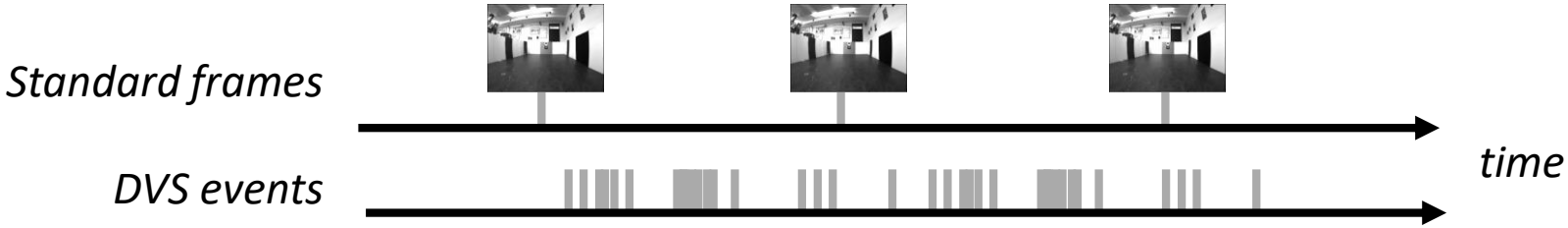
Outline

- Motivation
- DVS sensor and its working principle
- Traditional sampling vs level-crossing sampling
- Current commercial applications
- Calibration patterns
- A simple optic flow algorithm
- Event-by-event vs Event-packet processing
  - Case studies
- DAVIS sensor
  - Case study

DAVIS sensor: Dynamic and Active-pixel Vision Sensor

- Combines an **event** sensor (DVS) and a **standard** camera in the same pixel array
- **Output**: frames (at 30 Hz) and events (asynchronous)



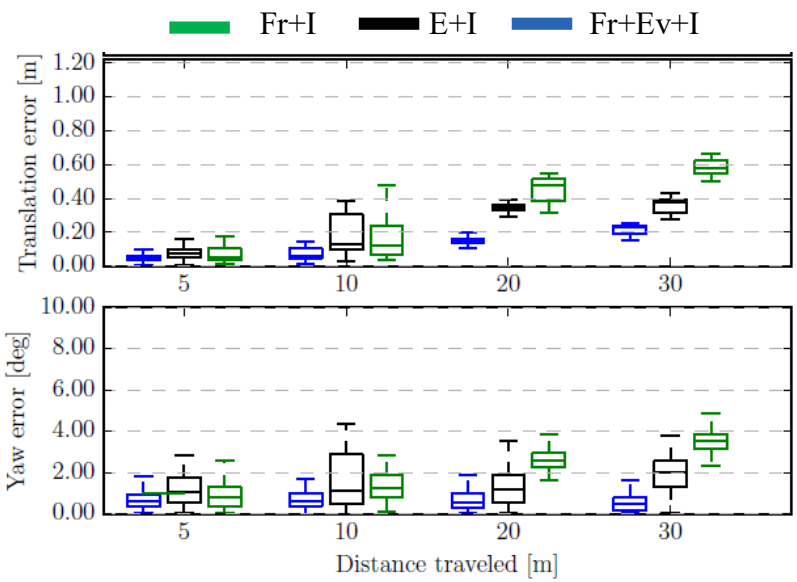


Brandli et al. *A 240x180 130dB 3us latency global shutter spatiotemporal vision sensor*. IEEE JSSC, 2014

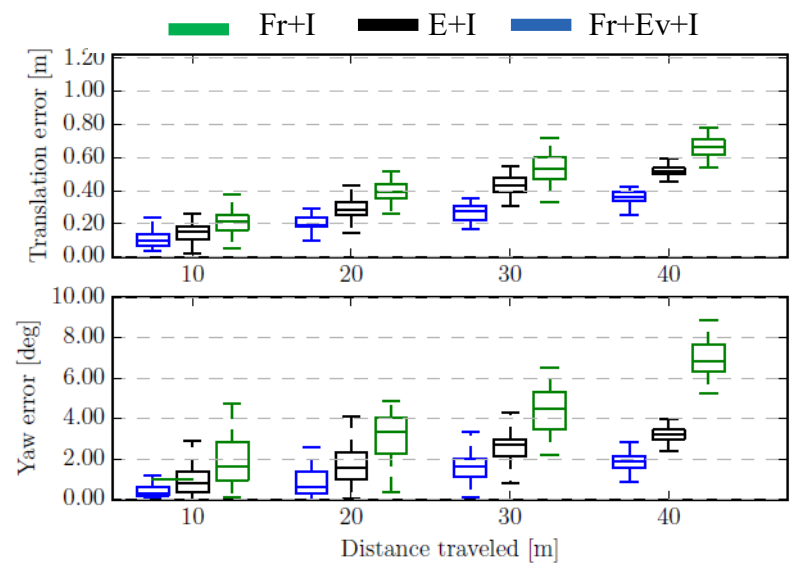
Events + Frames + IMU based SLAM («UltimateSLAM»)

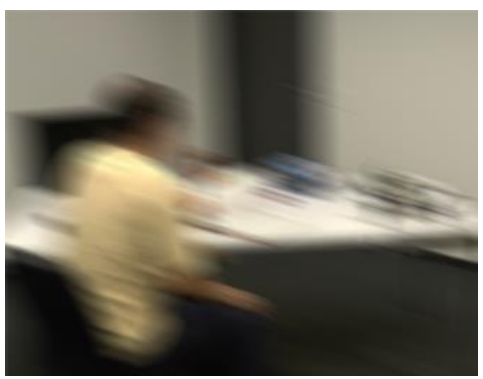
Adding standard frames increases the accuracy further

High-speed sequence



HDR sequence

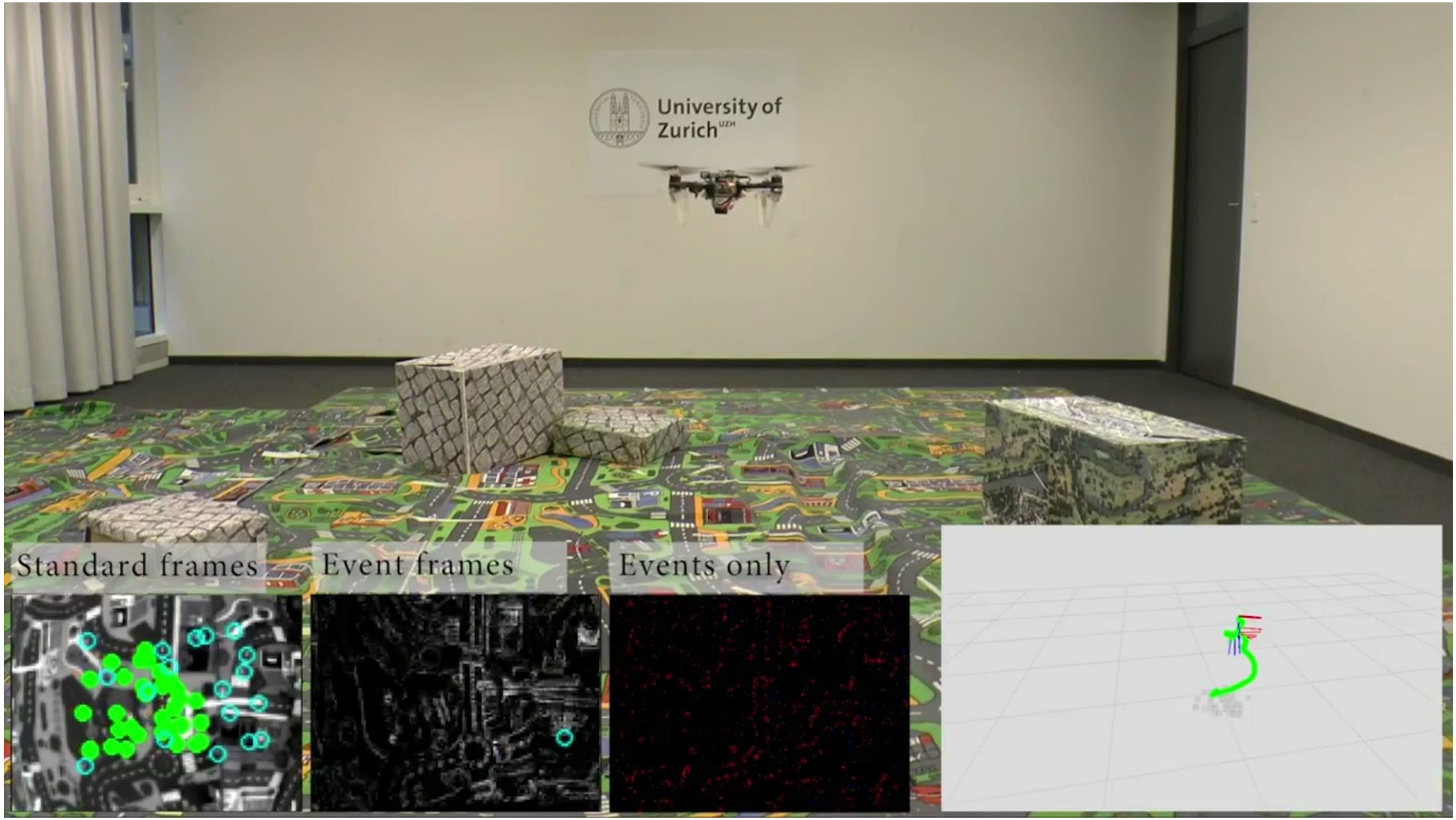




Vidal, Rebecq, Horstschaefer, Scaramuzza, Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High Speed Scenarios

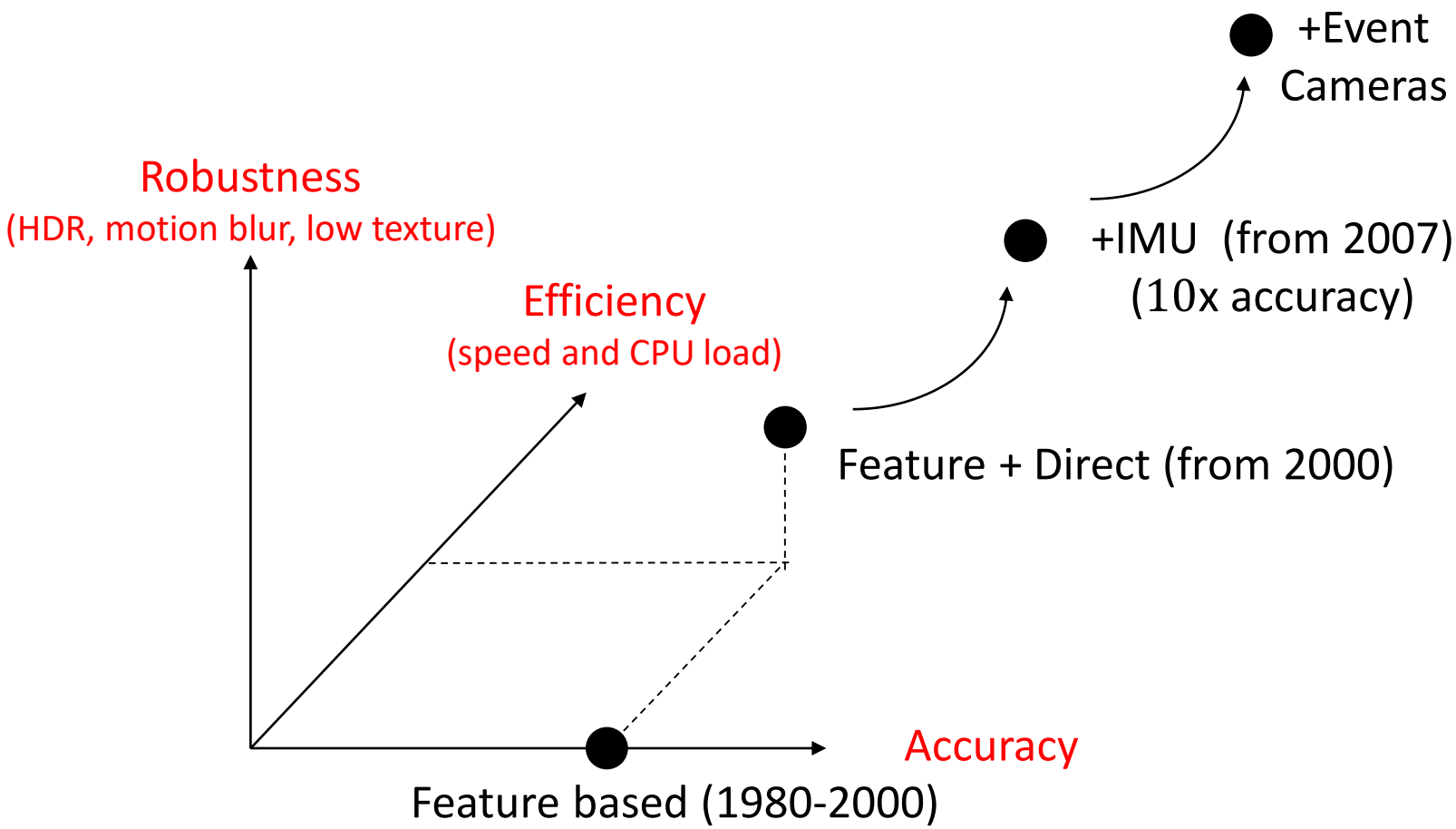
Autonomous Navigation with an Event Camera

Events + frames + IMU, tightly coupled. Fully onboard (Odroid XU4)



Vidal, Rebecq, Horstschaefer, Scaramuzza, Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High Speed Scenarios

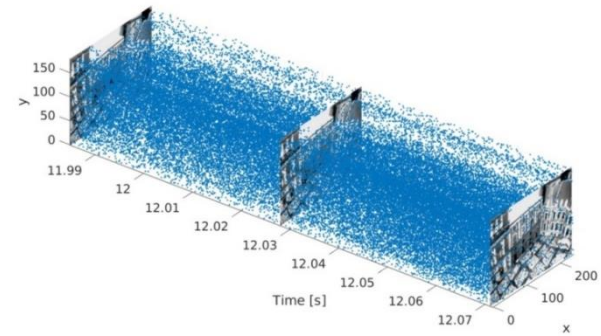
A Short Recap of the last 30 years of Visual Inertial SLAM



C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I.D. Reid, J.J. Leonard
Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age
IEEE Transactions on Robotics, 2016.

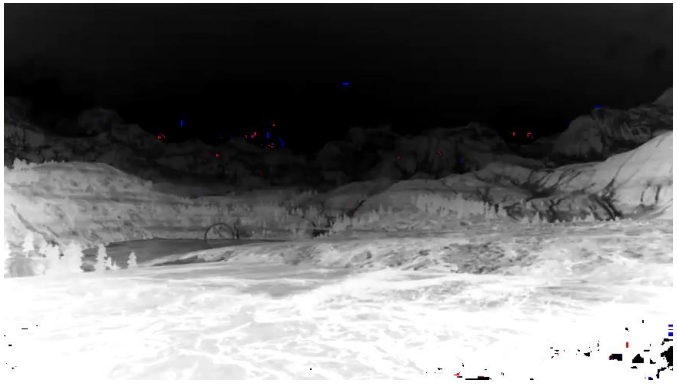
Event Camera Dataset and Simulator [IJRR'17]

- **Publicly available**: http://rpg.ifi.uzh.ch/davis_data.html
- **First event camera dataset** specifically made **for VO and SLAM**
- **Many diverse scenes:** HDR, Indoors, Outdoors, High-speed
- **Blender simulator of event cameras**
- Includes
  - **IMU**
  - **Frames**
  - **Events**
  - **Ground truth** from a motion capture system



➢ **Code, papers, videos, companies on event cameras:**

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



Mueggler, Rebecq, Gallego, Delbruck, Scaramuzza,
The Event Camera Dataset and Simulator: Event-based Data for Pose Estimation, Visual Odometry, and SLAM, International Journal of Robotics Research, IJRR, 2017.

Conclusions

- Visual Inertial SLAM **theory** is **well established**
- Biggest challenges today are **reliability and robustness** to**:**
  - High-dynamic-range scenes
  - High-speed motion
  - Low-texture scenes
  - Dynamic environments
  - Active sensor parameter control (on-the-fly tuning)
- **Event cameras** are revolutionary and provide:
  - Very **low latency** (1 μs) and **robustness** to **high speed motion** and **high-dynamic-range scenes**
  - Standard cameras studied for 50 years
    - event cameras offer have plenty of room for research
  - **Open problems on event cameras**: noise modeling, asynchronous feature and object detection and tracking, sensor fusion, asynchronous learning & recognition, low latency estimation and control, low power computation