Lecture 14: Event Based Vision

1 Review: Feature based vs. Photometric (direct) methods

1.1 Feature based methods

Feature based methods follow the following procedure:

1. Extract and match features (+RANSAC)
2. Minimize *reprojection error*, which we defined as

$$T_{k,k-1} = \underset{T}{\operatorname{argmin}} \sum_i \|u_i' - \pi(p_i)\|_\Sigma^2. \tag{1.1}$$

The major advantage of this method consists in the ability to handle large frame-to-frame motions. The major disadvantages are

- They are slow due to costly feature extraction and matching.
- Matching outliers (RANSAC).

1.2 Direct (photometric) Methods

These methods follow the procedure:

1. The pixel is the feature to track.
2. Minimize the *photometric error*, previously defined as

$$T_{k,k-1} = \underset{T}{\operatorname{argmin}} \sum_i \|I_k(u_i') - I_{k-1}(u_i)\|_\sigma^2. \tag{1.2}$$

The major advantages of this method are:

- All information in the image can be exploited (precision, robustness).
- Increasing camera frame rate reduces computational cost.

The major disadvantage is the scarse ability to handle frame-to-frame motion. Pure vision is not robust enough to handle low texture, HDR, high speed motion.

1.2.1 Influence of the number of pixels:

Dense and semi-dense methods behave similarly: weak gradients are not informative for optimization. Dense could be useful with motion blur and defocus. On the other hand, sparse methods behave equally well for image overlaps up to 30%.

1.2.2 Accuracy, Efficiency, Robustness: Challenges for Vision

As we have learned in the previous segment, an IMU alone is only helpful for short motions, as it drifts very quickly without visual constraints. In the following, we list some of the biggest challenges for vision today. In general, the challenge is about robustness to:

- High Dynamic Range (HDR). This can be handled with *active exposure control* in event cameras (more on this later).
- High-speed motion (i.e., motion blur). This can be handled with event cameras.
- Low-texture scenes. This can be handled with depth cameras or by getting closer to the scene.
- Dynamic environments. This could be handled with deep learning.

Current Visual Odometry algorithms and sensors show big latencies (in the order of 50 to 200 ms). Are we able to reduce this to much below 1 ms? In the next sections, we will investigate a possible solution to this: event based cameras.

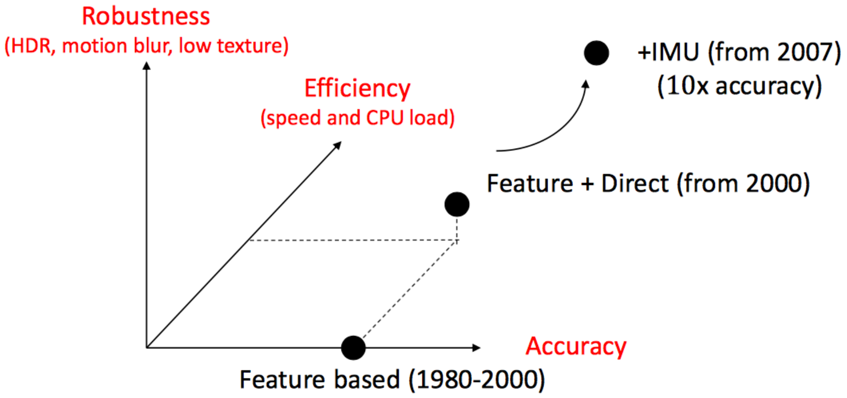

Robustness
(HDR, motion blur, low texture)
Efficiency
(speed and CPU load)
+IMU (from 2007)
(10x accuracy)
Feature + Direct (from 2000)
Accuracy
Feature based (1980-2000)

Figure 1: SLAM research: accuracy, efficiency, robustness.

2 Event-based Vision

2.1 Motivation

Current flight maneuvers achieved with onboard cameras are still too slow compared with those attainable by birds. We need low latency sensors and algorithms. 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** < 1 ms.

2.2 The Dynamic Vision Sensor (DVS) and its Working Principle

2.2.1 Human Vision System

The eye contains 130 million hotoreceptors (similar to pixels) but only 2 millions axons (wires that connect).

2.2.2 The DVS

In the following, we start listing the advantages and the disadvantages of this novel type of sensors:
Advantages:

- Low latency (circa 1 micro second)
- High dynamic range (140 dB instead of 60 dB).
- High updated rate (1 MHz).
- Low power: 10mW instead of 1W

Disadvantages:

- Paradigm shift: requires totally new vision algorithms, principally because of:
  - Asynchronus pixels,
  - No intensity information (only binary intensity changes).

A traditional camera outputs frames at fixed time intervals. By contrast, the dynamic vision sensor outputs asynchronous events at microsecond resolution. An event is generated each time a single pixel detects an intensity changes value:

$$\text{event: } \left\langle t, \langle x, y\rangle, \operatorname{sign}\left(\frac{\mathrm{d}I(x,y)}{\mathrm{d}t}\right)\right\rangle, \tag{2.1}$$

where t represents the timestamp at which the event has been produced, x and y the pixel coordinates at which the event has been produced and the third term the event polarity, i.e did we register and increase (+1 polarity) or a decrease (-1 polarity) in brightness? The asynchronous nature of the sensor is given from the fact that all pixels are independent from each other. Furthermore, the sensor implements **level-crossing** sampling and reacts to **logarithmic** brightness changes.

2.2.3 DVS Operating Principle

Each pixel is independent of all the other pixels. Events are generated everytime a single pixel sees a change of the logarithm of the brightness that is equal to C, i.e.

$$|\log(I)| = |\log(I(t+\Delta t) - \log(I(t))| = C, \tag{2.2}$$

where $C \in [0.15, 0.20]$ is called **contrast sensitivity** and can be tuned by the user. Since brightness can be either positive or negative, we have ON event if $\Delta\log(I) = C$ and OFF event if $\Delta\log(I) = -C$. Traditional sampling is performed with the discriminant (time) on x-axis. Level-crossing sampling works with the change in intensity, in the y-axis.

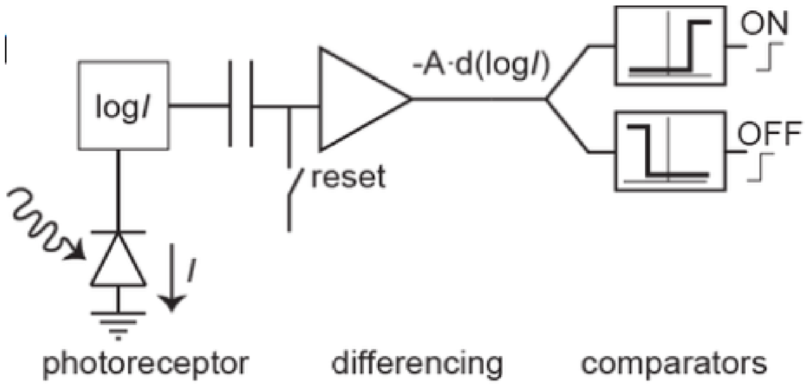


Figure 2: DVS circuit

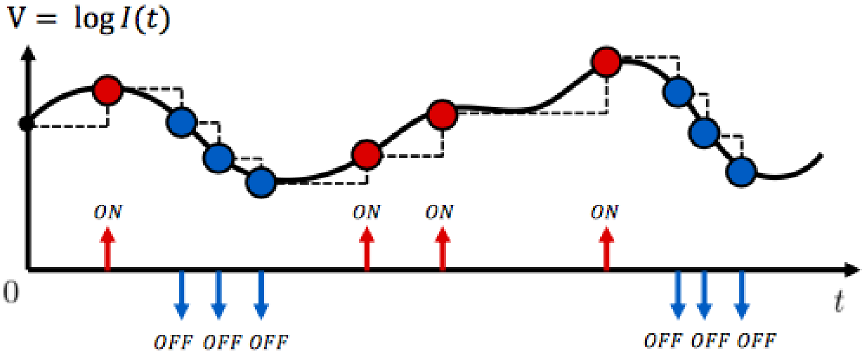


Figure 3: DVS.

Current Applications

Low power monitoring, fast closed-loop contro, high dynamic range imaging, low power gesture recognition, high speed flow speed estimation.

2.2.4 DVS vs High speed cameras

Calibration of a DVS

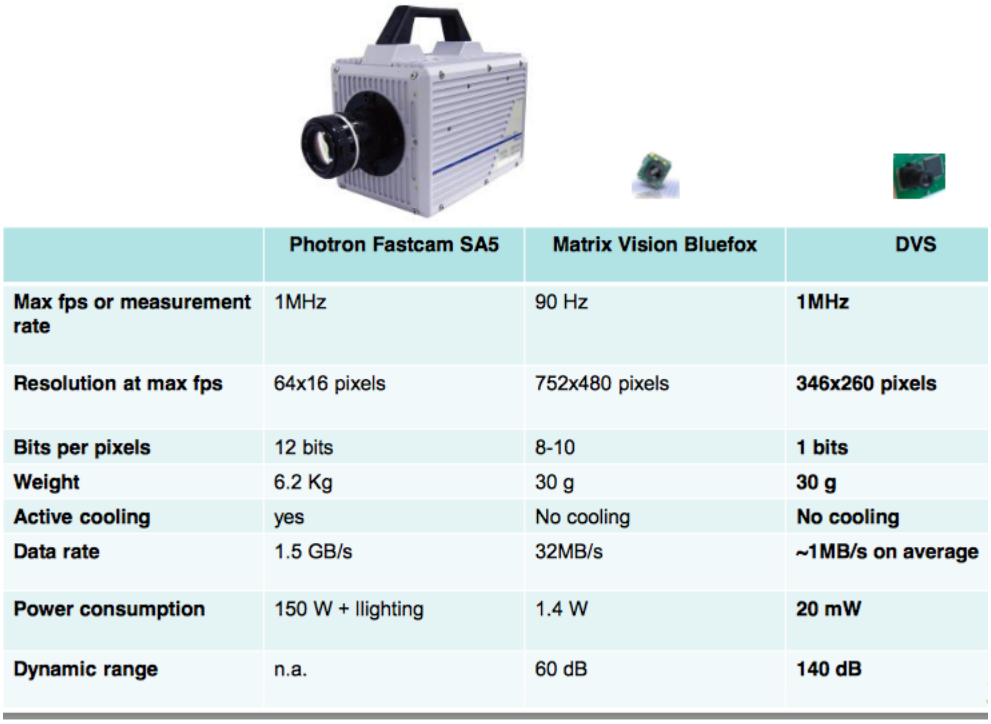
The standard pinhole camera model is still valid (same optics apply). Standard passive calibration cannot be used: we would need to move the camera. **Blinking patterns** (computer screen, LEDs) are used.

2.2.5 A simple optical flow algorithm: a moving edge

White pixels become black, i.e. the brightness decrease, i.e. negative events (black color). Events are represented by dots. At what speed is the edge moving? $v = \frac{\Delta x}{\Delta t}$.

How many events should be used?

Two different approaches



	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**

Figure 4: DVS vs High Speed Cameras.

- **Event-by-event processing** (i.e. estimate the state event by event): Pros: low latency, Cons: with high speed motion, there are dozens of millions of events per seconds (GPU)
- **Event-packet processing** (i.e. process the last N events): Pros: N can be tuned to allow real-time performance on a CPU. Cons: no longer microsecond resolution (when is this really necessary=)

2.2.6 Event-by-event based Processing

Let's start with an approximation:

$$\begin{aligned}\Delta\log(I) &= \frac{\partial\log(I)}{\delta t}\Delta t\\ &= \frac{1}{I}\frac{\delta I}{\delta t}\Delta t\\ &= \frac{\partial I}{I}.\end{aligned} \tag{2.3}$$

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

$$-\nabla I\cdot u=C \tag{2.4}$$

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

$$I(x,y,t)=I(x+u,y+v,t+\Delta t) \tag{2.5}$$

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

$$\begin{aligned} I(x,y,t) &= I(x,y,t+\Delta t) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v \\ I(x,y,t+\Delta t) - I(x,y,t) &= -\frac{\partial I}{\partial x}u - \frac{\partial I}{\partial y}v \\ \Rightarrow \Delta I &= C = -\nabla \cdot u. \end{aligned} \tag{2.6}$$

This equation described the linearized event generation equation for an event generated by a gradient ∇I that moved by a motion vector u (optical flow) during a time interval Δt. 1 Equation, 2 Unknowns, solution is to add events. □

Case Study 1: Image Intensity Reconstruction

The intensity signal at the event time can be reconstructed by integration of $\pm C$. Given the events and the camera motion (rotation), recover the absolute brightness.
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.

1. Recover the gradient map of the scene. Let $L = \log(I)$. Then

   $$\Delta L(t) = L(t) - L(t-\Delta t) = C. \tag{2.7}$$

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

   $$M(p_m(t)) - M(p_m(t-\Delta t)) \approx g \cdot v \cdot \Delta t, \tag{2.8}$$

   with $g = \nabla M(p_m(t))$.

2. Integrate the gradient to obtain brightness. Poisson reconstruction: integrate the gradient map g to get absolute brightness M.

Case Study 2: Event-based Corner Detection

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

2.2.7 Event-packet based processing

EVO: parallel tracking and mapping in real-time. Tracking, 6DOF pose, Mapping, 3D Map. How does a 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. The ray-density can be seen as tue Disparity Space Image (DSI). This is a projective sampling grid (with adaptive thresholding).

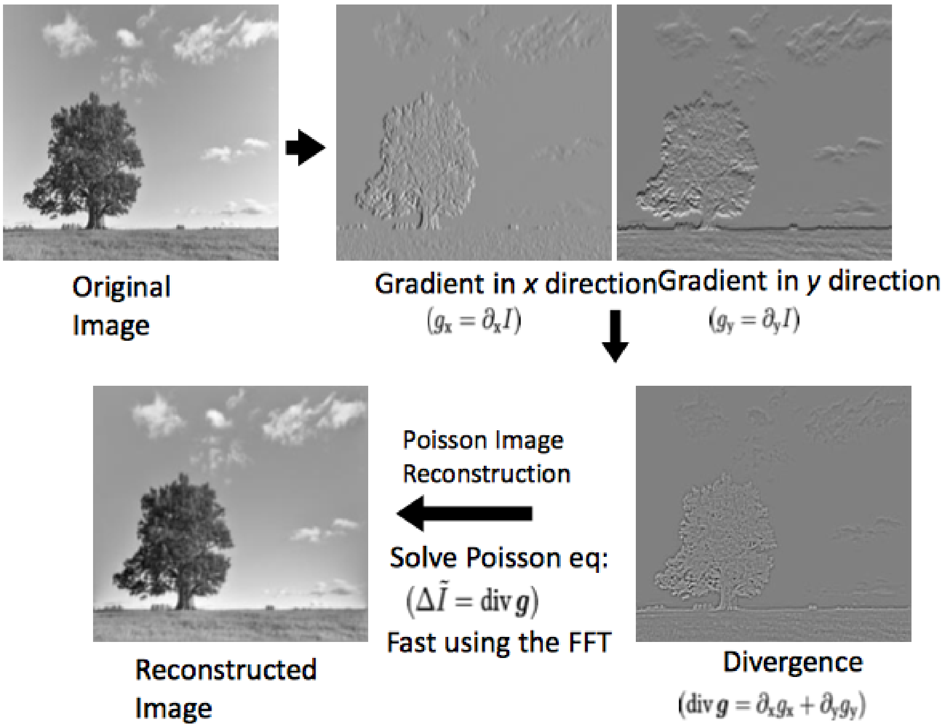


Figure 5: DVS vs High Speed Cameras.

DAVIS: Dynamic and Active-pixel Vision Sensor

Combines an event sensor (DVS) with a standard camera in the same pixel array. Output are frames (at 30 Hz) and events (asynchronous). One can them perform SLAM with an IMU, which increases robustness and accuracy.
Open problems for DVS are: noise modeling, asynchronous feature and object detection and tracking, sensor fusion, asynchronous learning and recognition, estimation and control, low power computation.

2.3 Understanding Check

Are you able to answer the following questions?

- *What is a DVS and how does it work?*
- *What are its pros and cons vs. standard cameras?*
- *Can we apply standard camera calibration techniques?*
- *How can we compute optical flow with a DVS?*
- *Could you intuitively explain why we can reconstruct the intensity?*
- *What is the generative model of a DVS?*
- *What is a DAVIS sensor?*
- *Can you write the equation of the event generation model and its proof?*