Applications, Software and Hardware for Event-Based Vision

Kynan Eng
Prelude
Will Event-Based Vision Change the World (of vision)?

If yes, then we need:

1. A community of developers (including workshops like this one)
2. Methods for quantifying benefits
Event-Based Vision, 2005

Delbruck, Liu, et al. 2005. CAVIAR EU project
Since 2008

+ Bias generator
+ USB cable

2008
DVS128

2019
>250 organizations

Resolutions
Form factors
Greyscale / Color

- Industrial
- Consumer Electronics
- Automotive
- Aerospace
- Services
- Computing hardware

>250 organizations
Today’s Talk

Software
• Making it easier

Applications
• Does it make sense to do this with DVS?

Hardware
• How to maximize DVS benefits
Software
Open DVS Development
Problems with DVS Development

Problems
• It’s too hard
• I can’t use OpenCV

Solutions
• Development environment
• Pre-built modules
• Interface to everything
Step 1: Prototyping with jAER + libcaer

>250 open-source modules contributed
Step 1: Prototyping with jAER

Up to 2016: >4000 downloads (Sourceforge, before switch to Git)
Step 2: Robust Solution

- High-performance C/C++
- Open API
- Decoupled GUI and engine
- Cross-platform
- Works with different DVS cameras
- Interfaces to CNN hardware/software
Live Development

Network socket

Deployment

DV Instance
(local or remote)

Live Viewer

DVS
Data Flow Definition

+ pre-built high-level modules
Applications
How to select DVS applications
Questions & Assumptions

Where does DVS work best?

How can I decide?

Possible answers
  Low Latency
  HDR
  Energy efficiency?
Questions & Assumptions

Where does DVS work best?

How can I decide?

Single events vs Event frames vs Diff image
  Can we quantify how much DVS can beat frames?
Problem Formulation

Input Variables

\( F = \text{Target frame rate} \)
\( P_s = \text{Total system power available} \)

Constants

\( E_d = \text{Energy per frame diff image} \)
\( E_a = \text{Energy per processing algorithm update step} \)
  (note: assume algorithms need the same energy)

Output

\( P_a = \text{Power margin available per frame for algorithms} \)
Result: Where DVS Works Best

![Graph showing the relationship between effective frame rate and system power budget for DVS and frames. The graph indicates that DVS only works best under certain conditions, specifically when the system power budget is relatively low. The red line represents frames with a power budget of $E_{\text{alg}} = E_{\text{diff}} = 0.0001 \text{ J}$, while the blue line represents DVS with a power budget of $E_{\text{alg}} = 0.0001 \text{ J}$. The graph highlights that DVS benefits more from lower power budgets compared to frames, achieving higher effective frame rates under these conditions.]

© iniVation 2019
Hardware
Maximizing DVS Benefits
Questions, Principles

Question
What hardware works best with DVS?

Principles
Minimal power budget ASICs
Activity-dependent computation
## In-Array Noise Filtering

**Spatio-temporal correlation**

- **Pixel size (μm)**: 10
- **Max Readout Speed (Meps)**: 180
- **Readout Efficiency (event/clock)**: Best: 4, worst: 0.25
- **Power Supply (V)**: 1.2
- **Power (mW)**:
  - High activity: 180Meps: 4.9
  - Low activity: 100keps: 0.25
- **Normalized**:
  - Dynamic energy (pJ/event): 26
  - Static power (nW/pixel): 18

**2x2 pixels**

- 10μm pitch, 20% fill factor, 65nm 1P9M (non-CIS)
In-Array Noise Filtering

40% less data

87% less data

1800 rpm USB fan
1 kefps
300 lux

Flickering monitor
200 efps
300 lux

Li C, IISW 2019
Dynap-SE by aiCTX

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td>28 nm FDSOI</td>
</tr>
<tr>
<td>Supply Voltage</td>
<td>0.73 V</td>
</tr>
<tr>
<td>Neuron Type</td>
<td>Analog</td>
</tr>
<tr>
<td>Neurons per core</td>
<td>256</td>
</tr>
<tr>
<td>Core Area</td>
<td>0.36 mm²</td>
</tr>
<tr>
<td>Fan In/Out</td>
<td>2k/8k</td>
</tr>
<tr>
<td>Synaptic Weight</td>
<td>(4+1) bits</td>
</tr>
<tr>
<td>Energy per SOP</td>
<td>&lt;2 pJ²</td>
</tr>
<tr>
<td>Energy per spike</td>
<td>&lt;1.68 nJ³</td>
</tr>
<tr>
<td>Feature</td>
<td>Value</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>Technology</td>
<td>22 nm</td>
</tr>
<tr>
<td>Neuron Type</td>
<td>Digital</td>
</tr>
<tr>
<td>Neurons</td>
<td>1 M</td>
</tr>
<tr>
<td>Core Area</td>
<td>12 mm²</td>
</tr>
<tr>
<td>Parameters</td>
<td>4 M</td>
</tr>
<tr>
<td>Input sync</td>
<td>Fully event-driven</td>
</tr>
<tr>
<td>Power</td>
<td>&lt; 1 mW (typical)</td>
</tr>
</tbody>
</table>
DynapCNN by aiCTX

Total Sops: 0.36 GHz, and peak layer Sops: 0.28 GHz

No Face Carsten Felix Ole Sadique Qian Sebastian
DVS + DynapCNN

Micropower intelligent scene analysis for mobile and IoT

- Announced CES 2019
- Single-chip DVS + CNN processor
- Ultra-low-power classification
- <1 mW power (typical)
- Low latency
- Sampling Q3 2019
Talk to Us

www.iniVation.com

Live demos @ CVPR: Booth 1554