High-Throughput Event-Based Feature Detection and Tracking on an Embedded CPU

EVENT-BASED VISION · FEATURE DETECTION AND TRACKING

Ethan Elms · Yasir Latif · Tat-Jun Chin — AI for Space Group, Australian Institute for Machine Learning, Adelaide University

Adelaide University | Australian Institute for Machine Learning

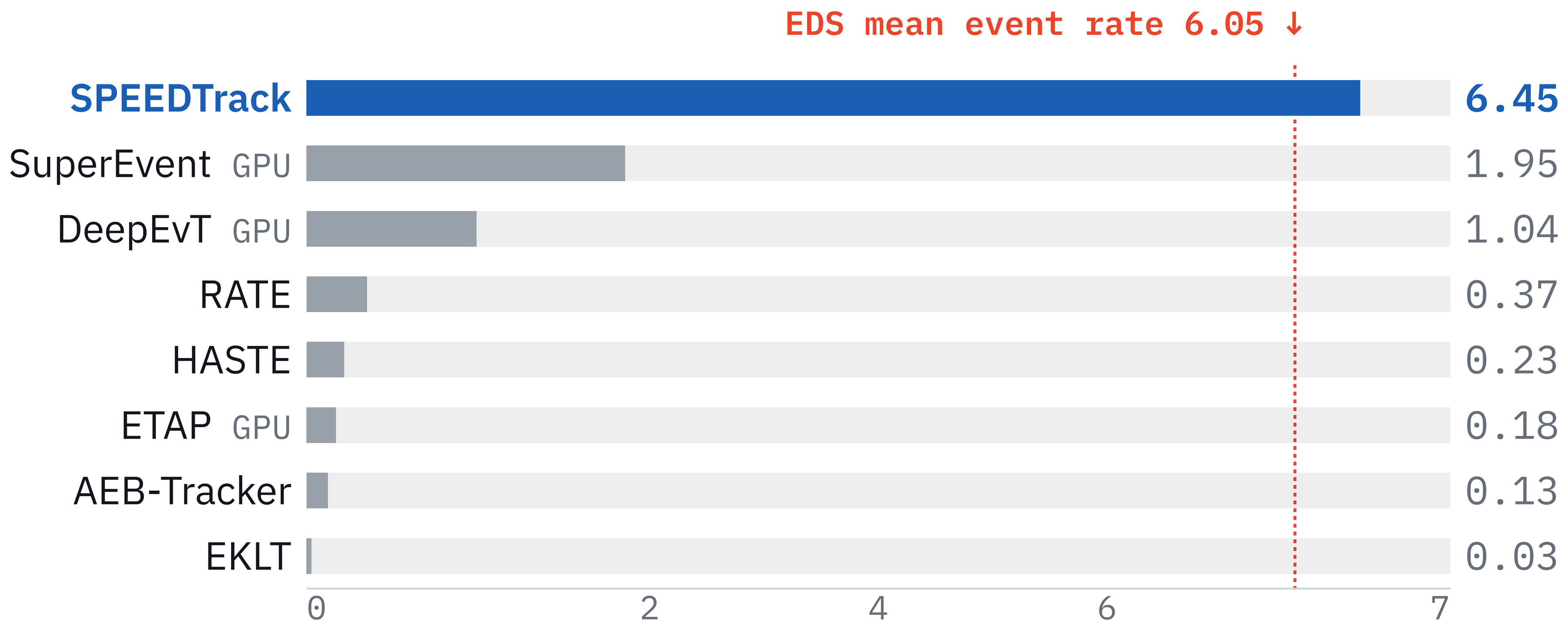
ECCV 2026 | MALMÖ | SEPT 8–12

SPEEDTrack Project page

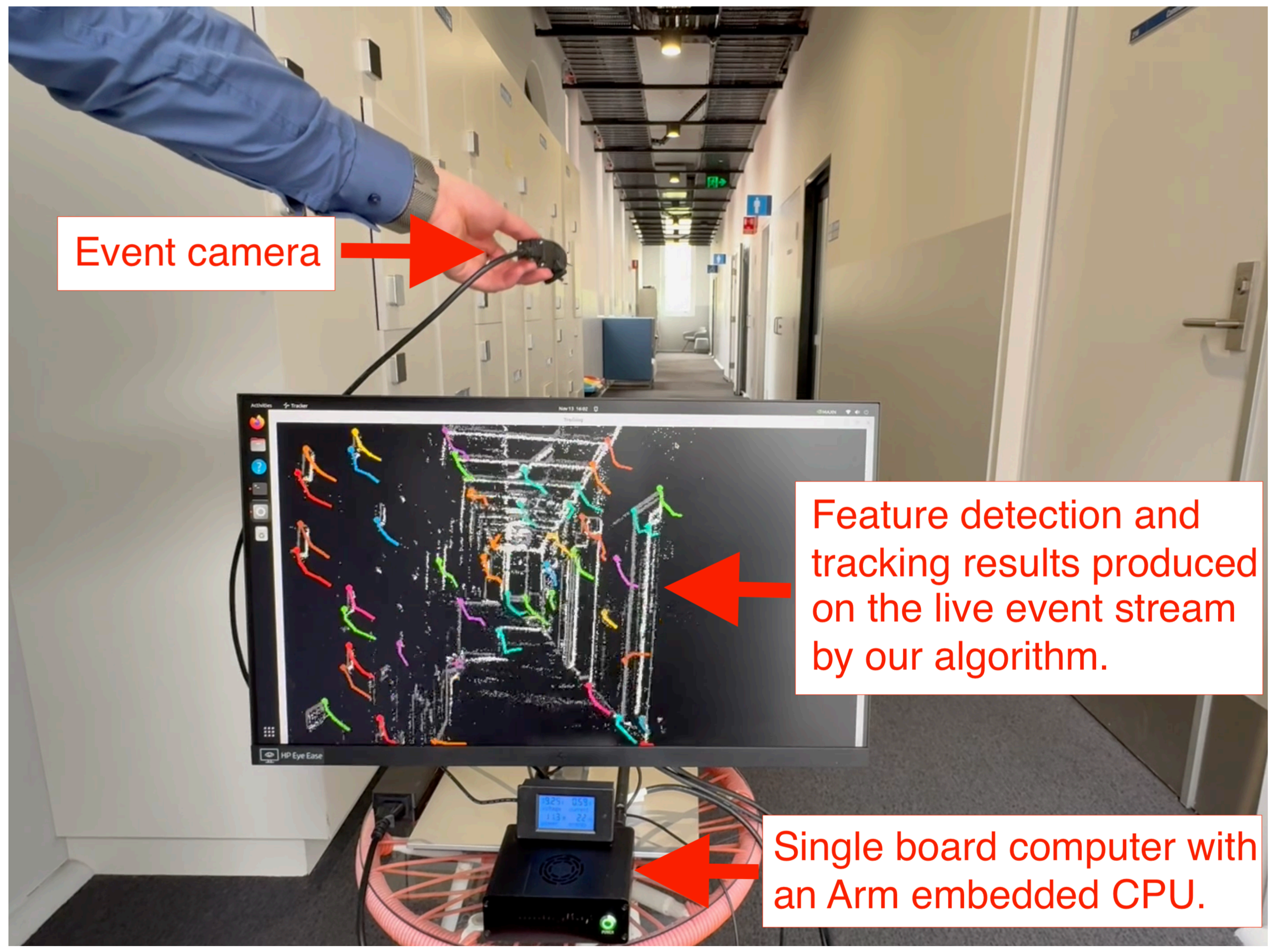
S₁ The problem

An event camera produces events faster than any tracker we measured can consume them. SPEEDTrack is the first that keeps up.

Throughput is the rate at which a method processes events; the event rate is the rate the sensor emits them. Below that rate, subsampling becomes mandatory to maintain live operation, and the sampling policy must discard events before their information content is known. On an embedded platform this margin has to be achieved within a fixed power budget.



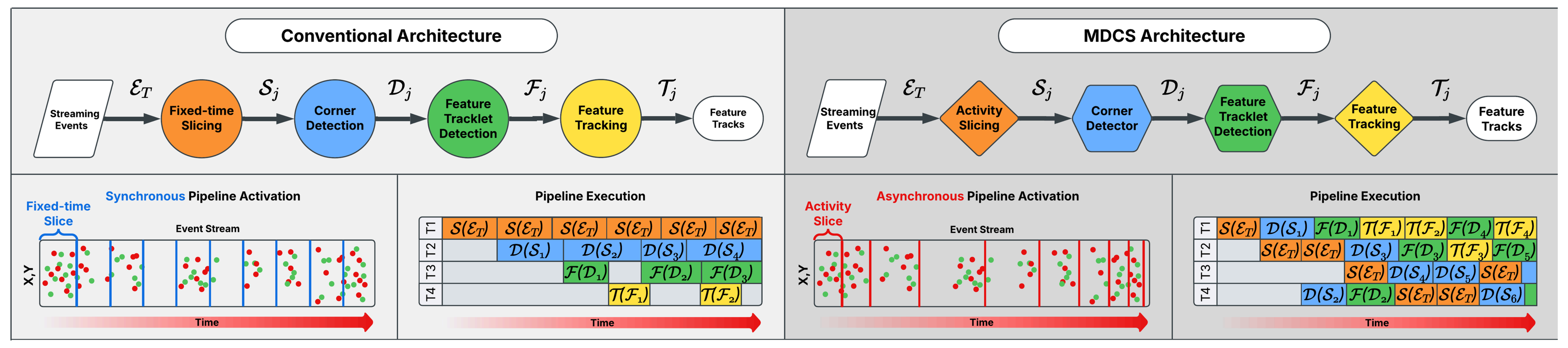
Mean throughput on EDS, million events per second. Everything to the left of the red line falls behind the sensor.



Live event stream, tracked in real time on eight Arm Cortex-A78AE cores without using the GPU. 12 W peak power consumption (camera and compute together).

S₂ Key idea

Conventional pipelines slice the event stream on a clock and at best achieve thread-based parallelism for each stage. Still scenes burn compute, fast motion blurs, and cores sit idle waiting on each other. We change both halves of that.



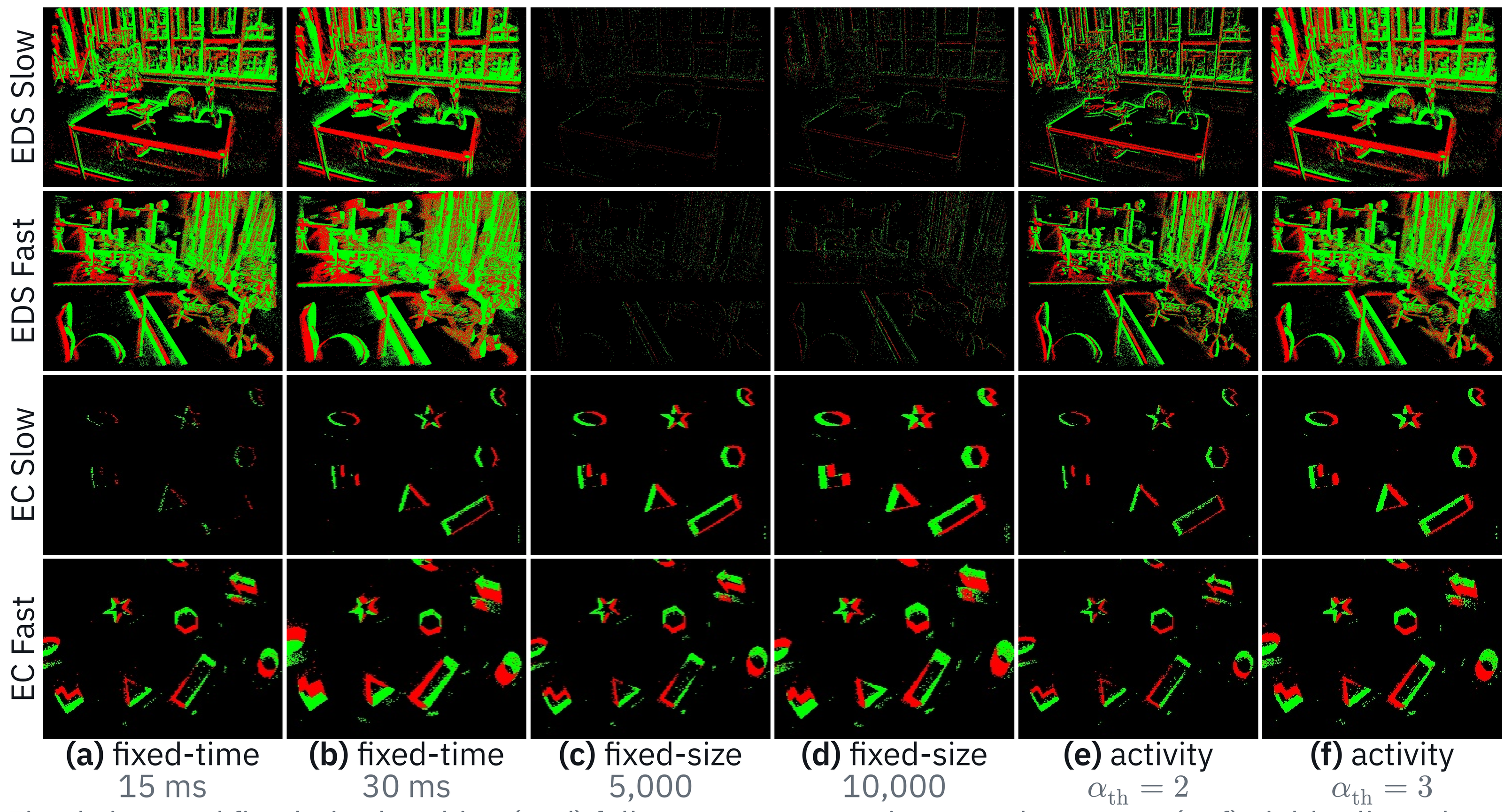
Conventional scheduling leaves threads idle; the Motion-Defined Concurrent Slices architecture keeps them fed.

Each slice is defined by motion

Fill a slice until the mean number of events per *active* pixel crosses a threshold. Fast, dense motion produces slices quickly; a still scene never produces one, so compute and power scale directly with motion.

$$\alpha = \frac{|\mathcal{S}_j|}{A_{\text{active}}} > \alpha_{\text{th}}$$

A single threshold, α_{th}, for every benchmark and every live demo.



Fixed-time and fixed-size batching (a–d) fall apart across motion speeds; one α_{th} (e–f) yields slices that look the same fast or slow.

Each slice is self-contained

Detection and tracklet fitting are slice-local, with no shared state. Only the association stage reaches across slices — so everything before it can run on any core, and slices need not finish in order.

Any core takes any stage

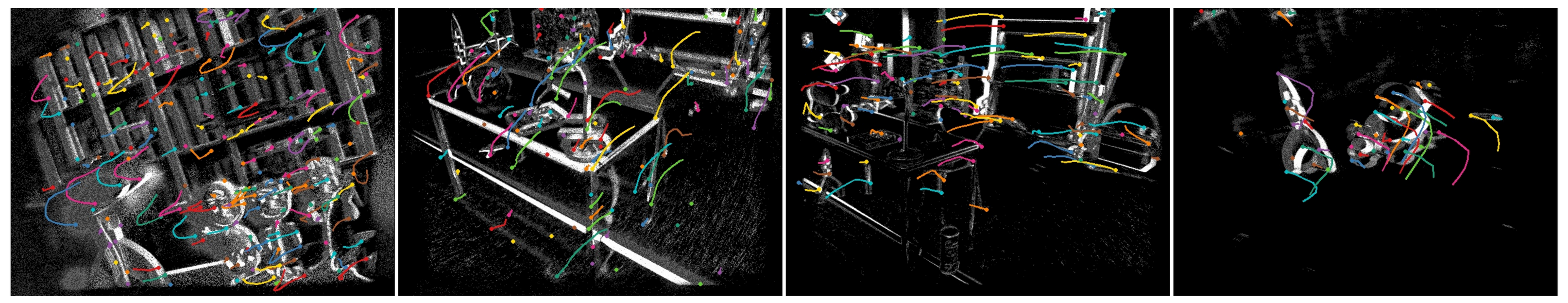
With slices independent, the pipeline becomes a task graph: a stage runs the moment its inputs land, on whichever core is free. Synchronisation is minimal and almost all CPU time goes to useful parallel work.

S₃ Results

Speed is not paid for in accuracy. Seeded with the same detections, our tracker outperforms every event-only method that runs without a GPU, and both hybrid event+frame trackers on EDS.

Feature age	EDS SFA↑	EDS EFA↑	EC SFA↑	EC EFA↑
HASTE	0.070	0.050	0.425	0.410
AEB-Tracker	0.300	0.290	0.540	0.460
EKLT +frames	0.230	0.150	0.795	0.760
DeepEvT +frames	0.480	0.400	0.815	0.805
ETAP GPU	0.742	0.613	0.870	0.851
SPEEDTrack	**0.558**	**0.421**	**0.822**	**0.713**

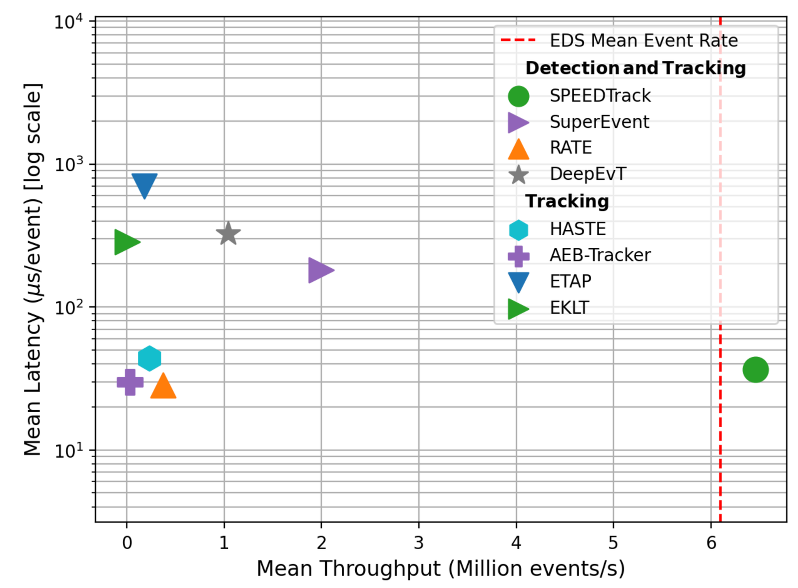
On relative pose we place second of five, behind only SuperEvent (AUC@10°: 20.8 on EC, 14.0 on EDS). Our corners also associate longest at the lowest reprojection error on the ATIS Corner Dataset — 2.27 s at 2.71 px.

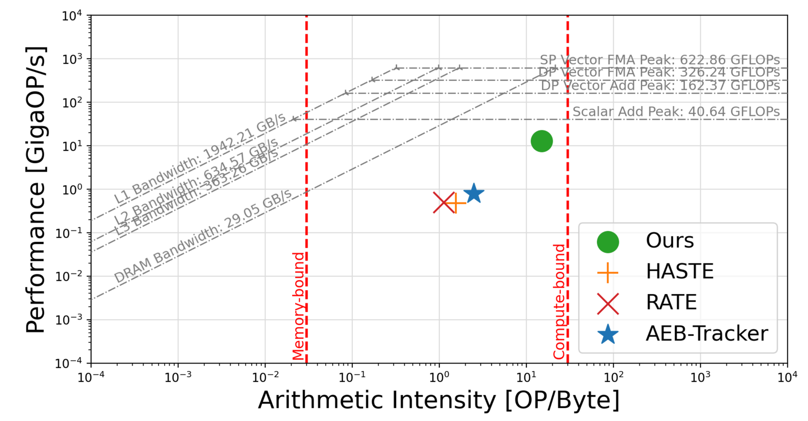


Tracks on peanuts dark, ziggy flying pieces, ziggy in the arena and rocket earth light from EDS.

Runtime performance

On eight Arm cores and 12 W, the full pipeline keeps pace with handheld motion in typical indoor and outdoor scenes, without a GPU.





Left: latency against throughput, both axes measured; aim for the bottom right. Learned trackers sit an order of magnitude higher in latency, while we are at 36 μs per event and still past the event rate. **Right:** roofline for the CPU. Prior trackers are stuck in the memory-bound region, far below what the hardware can do; ours sits near the compute ceiling. That is the difference the architecture makes — and it means more cores buy more throughput.

Align the algorithm with the hardware and an event-only, learning-free pipeline keeps up with a live event stream.

Open questions

- Events are dropped at the USB interface near 100 Mev/s.
- AC-flickering lights are indistinguishable from egomotion and seed phantom tracks.
- A lost track stays lost until we have a SLAM map to recover it from.