DOI 10.5162/SMSI2025/B6.1

Fast Tracking using Event-based Vision Sensors and Binary Frames

Johannes Baßler[1] *and Michael Heizmann*[1]
[1] Karlsruhe Institute of Technology, Karlsruhe, Germany
johannes.bassler@kit.edu

Summary: A novel tracking algorithm for high-speed object tracking using event-based Vision (EBV) sensors and binary frame representation is presented. The method transforms sparse event data into binary frames, enabling precise and robust particle tracking at $2\,\mathrm{kHz}$ frequency by combining correlation-based matching and center of gravity alignment.

Keywords: Tracking, Event-Based-Vision, Realtime-Capable, Binary Frames, High-Speed

Introduction

This work is based on two key design elements: the use of Event-Based Vision sensors for high-speed tracking, and the representation of events through binary frames.
Event-Based Vision (EBV) sensors represent a promising new paradigm in the field of computer vision, offering high temporal resolution and low latency. These characteristics make **EBV sensors** particularly well-suited for tracking rapid movements. This paper presents a real-time, robust tracker that processes event data using binary frames, demonstrating its potential in applications such as bulk material sorting, where precise object tracking is crucial for pneumatic actuator control.
The proposed tracker operates at $2\,\mathrm{kHz}$, enabling precise tracking of particle trajectories in real-time. This performance level is challenging to achieve with conventional frame-based camera sensors due to limitations in costs, latency, and algorithm runtime. For comparison, the work of Maier et al. [1] utilizes conventional industrial-grade frame cameras, achieving only a $93\,\mathrm{Hz}$ tracking frequency. By leveraging the properties of EBV sensors, our approach overcomes these constraints, offering a significant advancement in high-speed object tracking. Moreover, there is potential to further increase the sampling rate if required for specific applications.
While existing event-based tracking literature, exemplified by Barranco et al. [2], focuses on direct sparse event processing, our work presents a paradigm shift through **binary frame representation**. The sparse spatio-temporal nature of event data poses significant challenges for classical computer vision algorithms, particularly in real-time scenarios with time-dependent data volumes. Our approach strikes a balance by generating binary frames that efficiently capture relevant information. These binary frames enable efficient morphological operations, support compression techniques like `RLE` and `CSR`, and minimize storage requirements through their boolean representation. This allows high frame rates and therefore a small but inherent loss in temporal resolution.

Method

Our tracking algorithm combines correlation-based matching to estimate object displacement with center of gravity alignment to refine tracking by maintaining the object's position within the reference frame. The method assumes a known initial position $\xi_{t_0,p}$ and reference frame $\mathbf{r}_{t_0,p}$ containing the tracking object, which in the context of bulk material sorting can be determined when a particle crosses a predefined boundary region in the direction of motion. The algorithm operates on binary frames accumulated over a fixed time interval τ. This accumulation period also defines the tracking frequency, as each new frame triggers a new iteration. This dual-step process is executed in parallel for each polarity p individually, providing shifts in both x and y directions.

Binary Frame Generation Binary frames are generated through temporal accumulation of the event stream. For each pixel location (x, y) and polarity $p \in \{+, -\}$, events are accumulated over a time window τ, resulting in binary frames $\mathbf{b}_{t,p}(x, y)$ defined as:

$$\mathbf{b}_{t,p}(x,y) = \begin{cases} 1 & \text{if } \exists e(x,y,t,p) \text{ in } [t, t+\tau] \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $e(x, y, t, p)$ represents an event at position (x, y) with polarity p at time t. This results in two separate binary frames $\mathbf{b}_{t,+}(x, y)$ and $\mathbf{b}_{t,-}(x, y)$ for positive and negative events respectively. The accumulation time τ is a crucial parameter that needs to balance two competing requirements: it should be large enough to create coherent structures in the binary frames, yet small enough to maintain sufficient temporal resolution for the tracking task. In our implementation, we set $\tau = 500\,\mu s$ to achieve a tracking frequency of $2\,\mathrm{kHz}$.

Correlation-based Shift We compute the correlation map $\mathbf{c}_{t,p}$ by correlating the reference frame $\mathbf{r}_{t-1,p}$ with a constrained region $\tilde{\mathbf{b}}_{t,p}$ of $\mathbf{b}_{t,p}$. This region is delimited around $\xi_{t-1,p}$, extending d pixels beyond the size of $\mathbf{r}_{t-1,p}$ to capture the maximum expected particle displacement during τ, thereby reducing computational complexity significantly.

$$\mathbf{c}_{t,p} = \mathbf{r}_{t-1,p} \star \tilde{\mathbf{b}}_{t,p} \tag{2}$$

The correlation-based shift $\delta^{corr}_{t,p}$ is then determined by:

$$\delta^{corr}_{t,p} = \arg\max(\mathbf{c}_{t,p}) - C(\tilde{\mathbf{b}}_{t,p}) \tag{3}$$

where C represents the geometric center operation. Note, that $C(\tilde{\mathbf{b}}_{t,p})$ is equivalent to $\xi_{t-1,p}$ but referenced in the local coordinate system of $\tilde{\mathbf{b}}_{t,p}$ instead of the global coordinate system.

Center of Gravity Alignment Now, we extract an expanded preliminary reference frame $\tilde{\mathbf{r}}_{t,p}$ centered at the shifted position $\xi_{t-1,p}+\delta^{corr}_{t,p}$. The second shift $\delta^{grav}_{t,p}$ is computed as:

$$\delta^{grav}_{t,p} = \mathtt{CoG}(\tilde{\mathbf{r}}_{t,p}) - C(\tilde{\mathbf{r}}_{t,p}) \tag{4}$$

where `CoG` represents the center of gravity calculation. The actual reference frame $\mathbf{r}_{t,p}$ is then cut out of $\tilde{\mathbf{r}}_{t,p}$ w.r.t $\delta^{grav}_{t,p}$. This centering adjustment maintains the tracked particle within the reference frame, enhancing robustness by preventing positional drift across iterations. Morphological filtering of each reference frame enhances robustness against noise.

Final Position Estimation The total shift per iteration and polarity is $\delta_{t,p} = \delta^{corr}_{t,p} + \delta^{grav}_{t,p}$, yielding the next point $\xi_{t,p} = \xi_{t-1,p} + \delta_{t,p}$. Robustness is enhanced further by independent processing of positive and negative events, creating redundancy in the tracking process. The final object position is determined by $\xi_t = \frac{\xi_{t,+}+\xi_{t,-}}{2}$ where $\xi_{t,+}$ and $\xi_{t,-}$ represent the positions obtained from positive and negative events, respectively.

Results

The proposed tracking algorithm has been tested with numerous particle trajectories. Our experiments demonstrate the method's robustness and real-time capability, with processing times of approximately $200\,\mu s$ per iteration on a modern mid-range desktop processor, significantly lower than

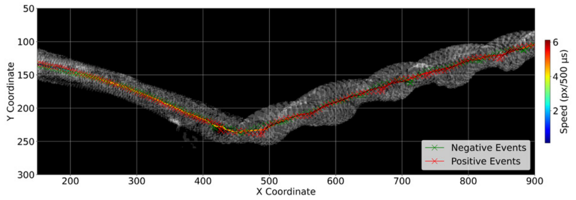

Fig. 1: Tracked particle trajectory with velocity estimation. The background image outlines the swept area of the particle during the flight.

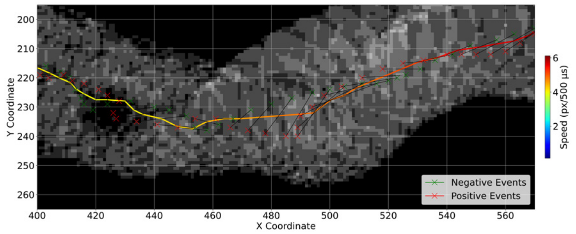

Fig. 2: Tracked particle trajectory with velocity estimation zoomed in at the interaction area between particle and pneumatic system.

$\tau = 500\,\mu s$ required for binary frame accumulation, posing the real time boundary. Figure 1 illustrates a representative tracking result, while 2 zooms in in order to reveal more details. The trajectory includes post-processed velocity estimations. More examples and videos illustrating the results for various examples in the context of bulk sorting can be found here: https://github.com/uwupl/FT_SMSI

Summary & Outlook

The tracking algorithm demonstrates significant potential for further development and practical implementation. Future work will focus on extending the method to track multiple objects simultaneously and incorporating rotational velocity estimation of the tracked objects. This extension would provide valuable information for more precise control of sorting actuators.

For optimal industrial performance, we are exploring FPGA implementation options. This hardware acceleration approach could significantly reduce processing latency and enable even higher tracking frequencies, making the system more suitable for high-throughput sorting applications.

References

[1] G. Maier, F. Pfaff, C. Pieper, R. Gruna, B. Noack, H. Kruggel-Emden, T. Längle, U. D. Hanebeck, S. Wirtz, V. Scherer, and J. Beyerer, "Experimental evaluation of a novel sensor-based sorting approach featuring predictive real-time multiobject tracking," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 2, pp. 1548–1559, 2021.

[2] F. Barranco, C. Fermüller, and E. Ros, "Real-time clustering and multi-target tracking using event-based sensors," pp. 5764–5769, 10 2018.