Real-time 6-DoF Pose Estimation by an Event-based Camera using Active LED Markers

Anonymous WACV Applications Track submission

Paper ID 282

Abstract

Real-time applications for autonomous operations depend largely on fast and robust vision-based localization systems. Since image processing tasks require processing large amounts of data, the computational resources often limit the performance of other processes. To overcome this limitation, traditional marker-based localization systems are widely used since they are easy to integrate and achieve reliable accuracy. However, classical marker-based localization systems significantly depend on standard cameras with low frame rates, which often lack accuracy due to motion blur. In contrast, event-based cameras provide high temporal resolution and a high dynamic range, which can be utilized for fast localization tasks, even under challenging visual conditions. This paper proposes a simple but effective event-based pose estimation system using active LED markers (ALM) for fast and accurate pose estimation. The proposed algorithm is able to operate in real time with a latency below $0.5\,\mathrm{ms}$ *while maintaining output rates of* $3\,\mathrm{kHz}$*. Experimental results in static and dynamic scenarios are presented to demonstrate the performance of the proposed approach in terms of computational speed and absolute accuracy, using the OptiTrack system as the basis for measurement. Moreover, we demonstrate the feasibility of the proposed approach by deploying the hardware, i.e., the event-based camera and ALM, and the software in a real quadcopter application. Our project page is available at: almpose.github.io*

1. Introduction

Fast and reliable spatial localization is essential in a wide range of robotic applications. For example, in collaborative scenarios, the ability to accurately and rapidly estimate the pose of the end effector is a key component for achieving a safe, reliable and robust execution of corresponding tasks. Vision-based methods [3, 14, 37] are the most common approaches for obtaining the relative localization of objects within the line of sight. These methods achieve significantly better accuracy compared to other non-contact localization methods, *e.g.* radio-based localization approaches [11, 33]. Vision-based approaches are, however, computationally expensive and typically require more than one sensor, *e.g.* infrared-based systems [26]. To reduce the computational overhead, classical markers [15,36] serving as easy-to-detect anchors are often integrated into vision-based systems. Since conventional RGB-D cameras are often used in these systems, the latency of detection cannot be reduced beyond the limit determined by the frame rate of the utilized cameras.

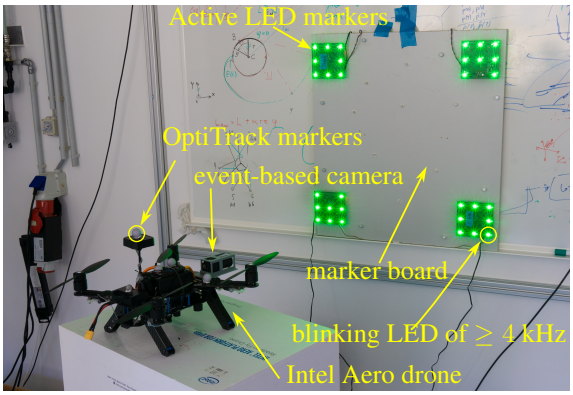


Figure 1. Overview of the experimental setup. Active LED markers (ALM) are attached to a marker board. An event-based camera mounted on a drone is used to estimate the pose of the marker board.

Event-based vision is an emerging field that has attracted much attention in recent years [10, 24]. An event-based camera consists of an array of independent pixels measuring changes in luminosity $L = \log(I)$, based on the photocurrent I [18]. A change in the continuous luminosity signal

$$\Delta L(\mathbf{u}_k, t_k) = L(\mathbf{u}_k, t_k) - L(\mathbf{u}_k, t_k - \Delta t_k) > p_k C \quad (1)$$

triggers an event $\mathbf{e}_k = (\mathbf{u}_k, t_k, p_k)$ at pixel location $\mathbf{u}_k =$

*These authors contributed equally to this work.

(u_k, v_k) due to a temporal contrast threshold $\pm C$, $p_k \in \{+1, -1\}$ being its polarity and Δt_k the time since the last event (at $\mathbf{u}_k$) occurred at t_k [10]. The state-of-the-art event-based sensors can produce up to 1.2 Giga events per second (Geps) [10] with a microsecond range timestamp accuracy.

Compared to frame-based cameras that deliver periodically dense (*i.e.* full-frame) information, the event stream is sparse and contains information that relates only to changes in the scene. Additionally, event-based cameras have a high temporal resolution and a large dynamic range. These features make them ideal for applications requiring fast and accurate detection. Starting from the early years of event-based vision development [9], advantages given by those sensors for robotic applications are noticable. Recent advancements in event-based sensor development [10] have enabled them to compete with the precision of other localization methods [7] due to increased resolution and reduced noise. Avoiding accumulated event representations (*i.e.* frames), markers can be tracked online utilizing the event-based camera's high temporal resolution of up to 1 µs.

An active LED marker (ALM) is a fixed geometric arrangement of individual LEDs, each unambiguously identifiable by its unique blinking frequency. By identifying the individual LEDs in the event stream and knowing the geometric arrangement of the LEDs, the pose of the ALM can be retrieved.

In this paper, we propose a fast and simple method employing an event-based camera together with ALM for simultaneous detection and tracking of the 6 degrees-of-freedom (DoF) pose of a rigid object in the 3D space. An overview of our proposed approach is depicted in Fig. 1 with four ALMs attached to a marker board. The event-based camera is mounted on the drone, which is utilized to estimate the pose of the marker board with respect to the camera's base frame. To estimate the pose, the blinkings of the LEDs are logged with an event-based camera to identify the corresponding frequencies of each LED in the ALM. These blinking frequencies are utilized to identify each individual LED and match it with the known geometry of the ALM. With this mapping of the individual points on the camera's sensor plane and the known geometry of the ALM, the pose of the ALM can be computed by utilizing a Perspective-n-Point (PnP) algorithm. In the presented approach, by tuning the biases, i.e., parameters for tuning the analog front-end of the event-based camera, and using a priori knowledge about timing, the complexity of the ALM tracking can be simplified. This aids in reducing the tracking latency. During the tracking, the initial detection is continuously refined, resulting in subpixel resolution. Such an approach can still precisely estimate the pose even under fast rotational and linear motion. The proposed approach was tested and verified extensively using an external infrared-based positioning system. Our contributions are listed in the following.

- We propose a fast event-based pose estimation system using ALM achieving a latency below $0.5\,\mathrm{ms}$ while maintaining an output rate of 3 kHz.
- We analyze the proposed system in static and dynamic scenarios for several in-depth aspects, e.g., absolute accuracy, static noise, and latency. Translational errors of $34.5\,\mathrm{mm} \pm 16\,\mathrm{mm}$ and $0.74° \pm 0.15°$ orientation errors at distances of $2.1\,\mathrm{m}$ to $4.8\,\mathrm{m}$ between the camera and the marker were achieved. Together with the fast computing speed, this proves that the proposed algorithm is promising for real-time applications.
- We integrate the proposed system into a quadcopter application for the 6-DoF pose estimation task. For indoor experiments, the proposed system outperforms the ORB-SLAM algorithm. Furthermore, in outdoor experiments, the proposed system can simultaneously detect and track the ALM in very aggressive flights at velocities of up to $10\,\mathrm{m\,s^{-1}}$ and up to $10\,\mathrm{m}$ away from marker.

The paper is organized as follows: Section 2 presents the related work in the field of pose estimation with event-based cameras and active markers. Section 3 describes the proposed method for marker detection and tracking. In Section 4, we present the experimental setup and results. Finally, we conclude the paper in Section 5 with a summary of our contributions and suggestions for future work.

2. Related Work

Visual localization systems show improved accuracy compared to systems based on other physical principles [27], [7]. Fiducial marker-based systems [13] constitute the most common choice for robotic applications. Due to the limited range and the dependence on the lighting conditions, some studies proposed LED-based solutions based on standard RGB cameras [35], infrared [34], or ultraviolet [31] spectrum. However, the latency cannot be reduced beyond the camera's frame rate.

One of the first works in the direction of localization based on event-based sensors was the 2D localization method [32]. The known shape (contours) was tracked, and the relative localization was determined by event-based vision. The high temporal resolution of the event-based sensors was used in [21] to localize an Unmanned Aerial Vehicle (UAV) during high-speed maneuvers. The pose information was retrieved using a black square as a known shape. In [16], a visual odometry method was proposed based on the feature tracking algorithm. In this direction, multiple methods were developed [16], [20], which show a significant improvement compared to the RGB-based approach for high-speed applications.

The utilization of ALMs was proposed first in [22], where the authors tracked the 2D position of the LED and used it as a feedback signal for the robot homing and a pan tilt system. Later, the first method for pose estimation using ALMs was presented in [4]. Therein, ALMs were used to detect and estimate the position of a flying quadrocopter. LEDs were recognized and detected using event polarity changes in the event stream. In [4], the authors used an accumulated event representation to decode the frequency and estimate the pose. In [6], a Gaussian mixture probability hypothesis density filter was proposed to localize the camera with respect to the active marker. Therein, online tracking was presented to increase the robustness and reliability of the pose estimation. The achieved results indicate a localization error lower than $3\,\mathrm{cm}$ in scenarios where the camera was within $1\,\mathrm{m}$ relative to the active marker.

Most recent works using ALMs propose the additional fusion of inertial measurements [28]. The error in the predicted relative position is in the subcentimeter range. However, utilizing only the vision-based approach increases the error by the order of one magnitude. Compared to previous methods, the marker size is significantly larger. The LEDs are placed $1\,\mathrm{m}$ apart. Current work in active marker-based solutions also focuses on the visual communication aspect of modulated light.

Different from other approaches in the literature, our approach simplifies the complexity by tuning the biases and using a priori knowledge about timing. This helps to reduce the tracking latency. To the best of the authors' knowledge, this work achieves the lowest latency compared to other methods in the literature.

3. Active Marker Tracking and Pose Estimation

Using ALMs, periodic and dense signals can be generated as a projection of the LED on the camera's sensor plane.

To reduce the computational complexity and the required bandwidth, the biases of the sensor are tuned to generate a single event per pixel on every LED blink while suppressing all other background events to increase the signal-to-noise ratio, as presented in Figure 2. While [4] uses events of both polarities and relatively low frequencies (1-2kHz), the amount of noise can be reduced by using higher frequencies and disabling one polarity.

The ALM's structure is an arrangement of high-frequency blinking LEDs, where a unique frequency of the blinking pattern can individually recognize each LED (*e.g.* different blinking frequencies). The arrangement of the LEDs has to be fixed and determined in the 3D coordinate space. However, it can also be arranged on a plane, as utilized in this work. Based on the 2D projection of the LEDs, knowing their 3D arrangement and camera intrinsics, the relative pose of the marker with respect to the sensor can be reconstructed using a Point-n-Perspective (PnP) algorithm [19]. In this work, the *IPPE* PnP algorithm is used [8].

The proposed approach is divided into four parts, as illustrated in Fig. 2. To reduce the noise in the signal, the bias settings are tuned to produce a single event per pixel on every blink of a LED. Next, events are accumulated over the time $\frac{2}{f_{\text{min}}}$, which is two times the period of the LED's minimal frequency f_{min} (typ. $f_{\text{min}} \approx 2\,\mathrm{kHz}$). For those accumulated event clusters, frequencies are recognized. These frequencies are used to identify newly appearing ALMs. For each of the ALM's LEDs, trackers are spawned that keep track of the LEDs' center points based on single events. The tracking of the LEDs is independent of the detection loop. The pose of the ALM is estimated utilizing the trackers of the ALM. The accuracy of the pose estimation can be obtained using the reprojection error. When an ALM leaves the field of view or the reprojection error exceeds the defined maximal value, the corresponding trackers are deleted. If the ALM enters the field of view again, the detection algorithm respawns it. Such an approach reduces the latency and increases the accuracy of the solution.

3.1. Detection

For the detection of an ALM, the geometrical arrangement and the blinking frequency of the ALM LEDs, have to be provided in advance.

The range of possible frequencies for the LEDs is wide: from tests conducted, frequencies higher than 4kHz and lower than 40kHz work best. To detect lower frequencies, biases have to be adapted to maximize the signal-to-noise ratio. As the timestamp is quantized, it is advisable to use LED frequencies with an integer microsecond period.

Due to the limited noise, detection can be simplified by using only single types of events. In [28] and [4], the detections rely on the transitions between event polarities. Instead, in the proposed approach, we use the timing information between consecutive events generated by a single pixel.

For detection, an event frame generated over the period T_d is used, where T_d has to be larger than $\frac{2}{f_{\text{min}}}$ to ensure that at least two blinks are visible for every LED. Candidates for the blinking LEDs can be retrieved by selecting the connected regions where more than $T_d f_{\text{min}}$ events per pixel are generated. Each region with an area larger than a defined minimal area is selected as a potential candidate. Due to the short accumulation period, even under fast motion, the LEDs' center points can be calculated by computation of the center of mass on a 2D plane. The error introduced by the relative movement of the LED is refined by the tracking procedure.

For the frequency estimation of the LEDs, a histogram of the time differences between events in a given area is used.

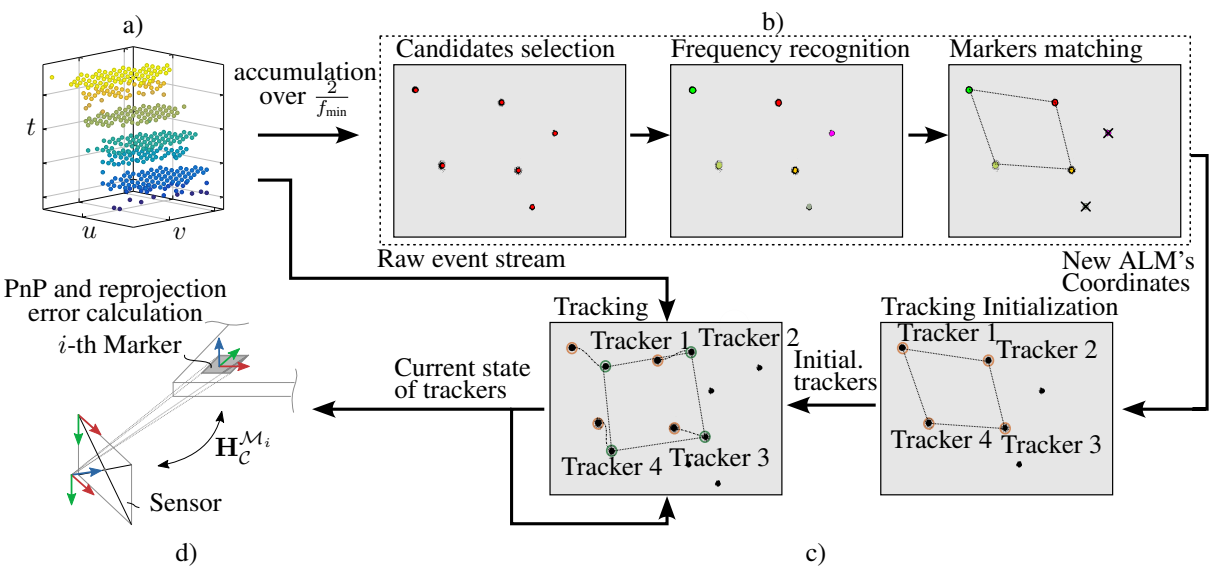


Figure 2. Overview of the proposed approach. Our pipeline consists of four asynchronous parts. *First*, illustrated in (a), to reduce the noise in the signal, biases are tuned to produce a single event per pixel on every blink. The proposed detection algorithm accumulates the events over a short period of time (two times the period of the LED's minimal frequency f_{min} to ensure at least two blinks are visible for every LED). *Second*, for the accumulated blinks, frequencies are recognized and assigned to the specified markers. For every detected marker (b), trackers (c) are spawned for each individual LED. *Third*, using a simple tracking procedure, the LEDs are being tracked independently. *Fourth*, to quantify tracking quality during runtime, the resulting solution (d) is used to compute the reprojection errors.

In the case of frequencies with an integer microsecond period, the histogram has a pronounced peak, while for other frequencies, the histogram follows a wider Gaussian distribution. The frequency estimation follows the procedure proposed in [4].

3.2. Tracking

While detection relies on an accumulated representation of the events, the tracking can be performed online to reduce latency. Using the initial guess from the detection of the ALM's LED center points, trackers are spawned for every LED. The i-th tracker is characterized by its frequency f_i, center point $\mathbf{c}_i = [x_i, y_i]$, and radius r_i. In comparison to the assumptions of [17] and [25], the distribution of the generated events (within one blink) follows a spatially uniform distribution and hence, produces a dense event stream in this region. This allows us to simplify the tracking algorithm while maintaining precise tracking with sub-pixel accuracy.

For every LED's blink, the tracker's center of mass $\bar{\mathbf{c}}_i$ is calculated using all events within its current radius r_i. The update term

$$\bar{\mathbf{c}}_i = \tilde{\beta}\mathbf{u}_k + (1 - \tilde{\beta})\bar{\mathbf{c}}_i \tag{2}$$

introduces low-pass filtering, where every new event $\mathbf{u}_k$ updates the current solution directly with an update factor $\tilde{\beta}$ of typically 0.02. The radius r_i is updated every N events and set to twice the average distance of the events from the center point of the tracker.

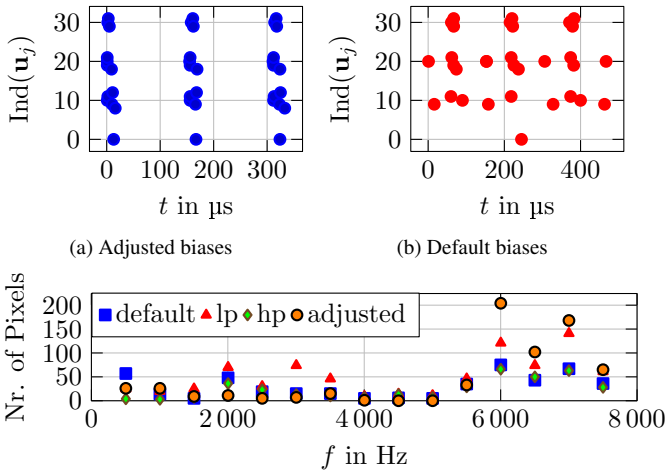
3.3. Pose Estimation

The 6-DoF pose is estimated asynchronously, using the current center points of the ALM's trackers. To increase the update rate of the algorithm, a PnP algorithm is started whenever the previous iteration is done. Due to the simplicity of the tracking, the PnP calculation is decisive in terms of latency and output rate.

To ensure stability and detect tracking failures, the reprojection error is computed and compared to the tracker's center points. When the reprojection error of one tracker exceeds the mean distance of the events from the center point, a tracking lost signal is generated, and tracking is stopped. It is reinitialized with the first new detection of a given marker.

4. Experiments

For the experimental setup, the EVK4 HD evaluation kit from Prophesee is used. It includes the event-based vision sensor IMX636ES providing HD resolution (1280×720 pixels) and the Soyo SFA0820-5M lens. The ALM consists of printed circuit boards with 8 LEDs arranged in a square of $9\,\text{cm}$ side length. Each ALM has a base frequency (first LED), and the remaining frequencies are selected to match integer microsecond period times. For the experiments, four markers are arranged in a square on a marker board with a



(a) Adjusted biases (b) Default biases

(c) Frequency histogram

Figure 3. Visualization of bias adjustment. The vertical axis in (a) and (b) represents the flattened indices of the pixels in the region of interest (ROI) around an LED light. Plot (a) and (b) demonstrate the event distribution with optimal and default bias settings, respectively. In the frequency histogram (c), the effect of the low-pass (lp) and high-pass (hp) settings are illustrated beside the default and adjusted bias settings.

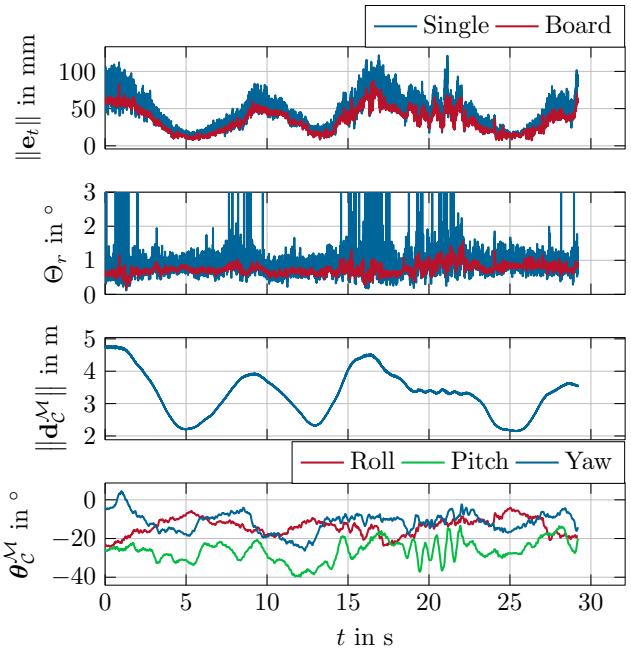


Figure 4. Analysis of the orientation and position error with reference distances and orientations.

side length of 59 cm. The 8 outermost LEDs were chosen to create a single marker. The event stream is processed on a Desktop PC (Ubuntu 20.04, Intel i9-12900K, 32GB RAM) and on an Intel Aero Compute Board. As ground truth, the commercial infrared 3D tracking system OptiTrack is used. The OptiTrack recordings are triggered with the same trigger signal as the event-based camera via its external trigger input.

4.1. Bias Adjustment

The bias adjustment of the event camera is essential for the proposed system's performance. The IMX636ES sensor biases [1] allow control over analog pixel gate thresholds to achieve the desired sensor response. The adjustment goal is to minimize the number of activated pixels between two LED blinks, as shown in Figure Fig. 3, thereby reducing processing complexity. The proposed method employs a single event polarity for simplicity.

By adjusting the refractory period setting, a pixel should be rendered insensitive to subsequent changes in LED brightness. An optimal value during adjustments should filter out all events between two consecutive LED-triggered events, as depicted in Fig. 3a. By utilizing high-pass and low-pass filter setups, the number of environment-generated events (excluding those by LEDs) can be limited to prevent sensor overflow and maintain manageable event blob density. The histogram of frequencies at which events occur over an accumulation time of 10 ms is depicted in Figure 3c. It illustrates the effect of the low-pass (lp) and high-pass (hp) bias settings acting as a band-pass filter for the LED frequencies of the ALM. Please note that the bias values may affect each other [1]. This causes the high event count, with the adjusted bias settings, at 6 kHz in Fig. 3c, where the value exceeds low-pass and high-pass settings.

A detailed explanation of the sensor biases is provided in [1] and the description of the bias adjustment procedure is detailed in the supplementary document.

4.2. Absolute Accuracy

To evaluate the absolute accuracy, the pose estimation of the ALMs and the marker board are compared with the synchronized measurements of the OptiTrack system (ground truth). For this experiment, the marker board is placed statically in the scene. The camera moves from close to far, covering the working distance of the setup, which is limited by the OptiTrack setup. The kinematic relations of the experimental setup are described in detail in the supplementary document.

The magnitude of the absolute position error $\mathbf{e}_t$ and the orientation error Θ_r with the distance between the marker and the camera $\left\|\mathbf{d}_C^{\mathcal{M}}\right\|$, ranging from 2.1 m to 4.8 m, as well as the orientation $\boldsymbol{\Theta}_C^{\mathcal{M}}$ of the marker with respect to the camera, is depicted in Fig. 4. Moreover, Fig. 4 illustrates the difference of using a single ALM with a side length of 9 cm for pose estimation compared to a larger marker board with a side length of 59 cm. From the plot of the position

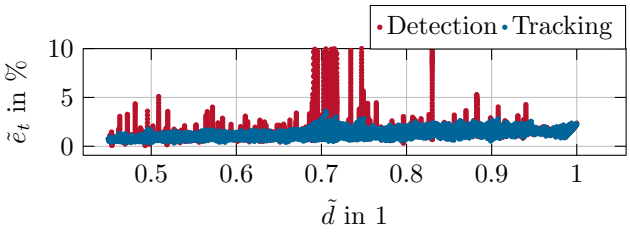


Figure 5. Normalized relative position error $\tilde{e}_t$.

error $\|\mathbf{e}_t\|$ in Fig. 4 it can be seen that the marker board has less noise but a comparable error magnitude. The second plot displaying the orientation error Θ_r shows significant spikes for the single ALM. This indicates flips in the estimated pose, especially for medium to far distances. Hence, the usage of a marker board with increased side length is beneficial for accurate orientation estimations. The plot of the marker orientation $\mathbf{\Theta}_C^{\mathcal{M}}$ in Fig. 4 shows fast orientation changes beginning at 20 s. This demonstrates the ability of the proposed method to accurately estimate pose information even in highly dynamic scenes.

In order to compare the performance between the detection-based pose estimation (Sec. 3.1) and the tracking-based pose estimation (Sec. 3.2), the relative position error $\tilde{e}_t = \frac{\|\mathbf{e}_t\|}{\|\mathbf{d}_C^{\mathcal{M}}\|}$ is displayed in Fig. 5 as a function of the normalized distance $\tilde{d} = \frac{\|\mathbf{d}\|_C^{\mathcal{M}}}{\max\|\mathbf{d}\|_C^{\mathcal{M}}}$. The results for the tracking-based approach indicate a better consistency, *i.e.* less noise, and altogether lower error numbers in the 1 % to 2 % range. The expected linear increase of the position error with the distance can also be inferred from Fig. 5. The statistical values of the data illustrated in Fig. 5 are summarized in Tab. 1. The maximum position error of 87.8 mm at a distance of 4.8 m and the maximum orientation error of 1.55° indicate the excellent performance of the tracking-based approach. The standard deviation of the position and orientation error of 16.2 mm and 0.146°, respectively, show the robustness of our method. In Table 4, we contextualize our results within different types of positioning system.

Table 1. Statistical values of the absolute accuracy measurements.

	Tracking		Detection	
	$\|\mathbf{e}_t\|$	Θ_r	$\|\mathbf{e}_t\|$	Θ_r
Mean	34.5 mm	0.738°	64.9 mm	1.55°
Std. Dev.	16.2 mm	0.146°	121 mm	5.12°
Maximum	87.8 mm	1.55°	1.233 m	71.9°

4.3. Static Noise

In Tab. 2, the noise floor of the proposed method is characterized for different distances between the camera and the marker board. The low standard deviation values indicate the stability of the pose estimation. This data can be utilized to tune Bayesian filters (*e.g.* Kalman filters).

Table 2. Statistics of the noise in static scenes.

$\|\mathbf{d}_C^{\mathcal{M}}\|$	Std. Dev.	Maximum
6 m	1.4 mm	5.5 mm
4 m	0.68 mm	2.95 mm
2 m	0.25 mm	2.17 mm

4.4. Latency Measurement and Output Rate

To determine the latency of the proposed system, the execution priority was elevated. Additionally, the visualization, as well as background tasks of the operating system, were disabled. This avoids unintentional interrupts and stalls during the execution of the pose estimation.

The latency and output rate values are listed in Tab. 3. The output rate of the tracking-based approach outperforms the detection-based method while achieving comparable latency results. As shown in Tab. 3, the proposed method is capable of running even on an embedded PC of a drone. While maintaining real-time performance, we can notice a reduced output rate (limited by the PnP computation time) as well as an increased average delay (limited by a number of concurrent threads) compared to a desktop PC. Latency is measured using a precise synchronization trigger signal and is equal to the time difference between the trigger and the time when pose estimation for this timestamp is available. Our proposed method achieves lower mean latency combined with low standard deviation compared to the state of the art. A large part of the resulting latency is due to communication overhead.

Table 3. Latency and output rates using a Desktop PC (PC) and Intel Aero Compute Board (Drone).

		Tracking		Detection	
		latency	rate	latency	rate
PC	Mean	354 µs	3.81 kHz	699 µs	670 Hz
	SD	92 µs	64 Hz	35 µs	288 Hz
Drone	Mean	1232 µs	1.32 kHz	1953 µs	223 Hz
	SD	194 µs	140 Hz	240 µs	94 Hz

4.5. Application to 6-DoF position estimation for a quadcopter

In this section, indoor and aggressive outdoor flights are considered for 6-DoF position estimation of a quadcopter.

4.5.1 Indoor flight experiments

Compared to stationary robots, e.g. articulated manipulators [29, 30], which are equipped with high-precision encoders to monitor their state, flying robots mainly rely on IMUs, barometers, and vision-based systems to estimate their state. While the pose estimation module equipped with

Table 4. Comparison of different marker based approaches and visual odometry/SLAM methods to contextualize the performance of the system. All the results are taken from corresponding papers. *Possible types E-Events, F-Frames, I-IMU. †Absolute Trajectory Error (RMS) reported in [12] using "kitchen" sequence. ‡Sequences used for evaluation include only slow motion.

Method	Input*	Rate/FPS	Range	Markers	Positioning error	Resolution	Dynamic motion	Marker size
Ours	E	3.8 kHz	up to 10 m	Active	1.89 % 2.11 %	720 p	✓	59 ×59 cm 9 ×9 cm
Censi et al. 2013 [4]	E	250 Hz	-	Active	8.9 cm	128 × 128	✓	20 ×20 cm
Salah et al. 2022 [28]	E+I	200 Hz	up to 7 m	Active	0.074 %	480 p	✗‡	3 ×3 ×3 m
Chen et al. 2020 [6]	E	-	1 m (tests)	Active	3 %	346 × 240	✗	40 ×30 cm
STag [2]	F	56 FPS	up to 3 m	Passive	1.32 %	720 p	✗	15 ×15 cm
ORB-SLAM2 [23]	F	-	-	SLAM	13.0 cm†	480 p	✓	-
EDS [12]	E+F	-	-	VO	9.6 cm†	480 p	✓	-

only IMUs and barometers often suffers from the problem of drift, vision-based systems ensure a more reliable measurement. In this experiment, the Intel® Aero Ready to Fly (RTF) Drone, shown in Fig. 1, is employed as it offers enough computational power for on-board processing.

For absolute drift-free pose information, the ORB-SLAM2 [23] algorithm is utilized. The ORB-SLAM2 was chosen for its impressive performance and open-source implementation. Note that ORB-SLAM3 [3], as the successor of ORB-SLAM2, is a more robust version compared to ORB-SLAM2. However, the accuracy of these approaches on stereo and RGB-D cameras is still comparable since the key concepts of the estimation module and the relocalization method, remain unchanged. We are aware that comparing the proposed system with other SLAM algorithms, e.g., feature-based SLAM [23] and event-based SLAM [5], may not be fair because the key concept is different. However, this comparison could provide a qualitative guide for choosing the right method in a given situation and contextualize results as shown in Table 4. Similar to previous subsections, the OptiTrack serves as the source of ground truth.

For the position errors, the metric for comparison is the difference between a ground-truth position and the estimated position as $\mathbf{p}_e = [p_{e,x}\ p_{e,y}\ p_{e,z}]^{\mathrm{T}} = \mathbf{p}_g - \hat{\mathbf{p}}$, where the hat $(\hat{\ })$ indicates quantities estimated by the ORB-SLAM2 algorithm [23] or with the event-based marker, respectively, the subindex $(\cdot)_g$ stands for the three-dimensional ground-truth quantities, and the subindex $(\cdot)_e$ for the resulting three-dimensional errors. The orientation errors are represented as Euler angles. The difference between the ground-truth quaternion and the estimated quaternion is defined as $\mathbf{q}_e = \hat{\mathbf{q}}^{-1} \otimes \mathbf{q}_g$,with $\otimes$ as the quaternion product. Subsequently, this error quaternion can be transformed into an equivalent representation using three angles, i.e., roll, pitch, and yaw. In this experiment, the quadcopter is moved aggressively in a zigzag pattern in a space of $1.8\,\mathrm{m}$ in the x-direction, $0.6\,\mathrm{m}$ in y-direction, and

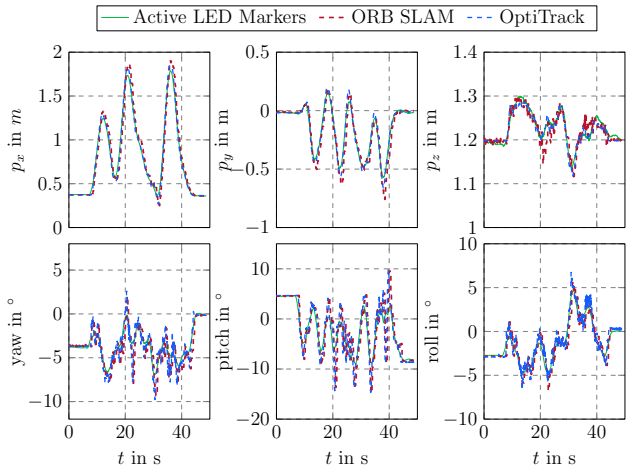


Figure 6. Time evolution of the drone trajectory.

$0.4\,\mathrm{m}$ in z-direction for about $50\,\mathrm{s}$. The position estimates and the resulting errors for this case are illustrated in Fig. 6 and Fig. 7. The errors obtained from the proposed algorithm in the x-, y- and z-direction are bounded within $\pm 0.06\,\mathrm{m}$, $\pm 0.08\,\mathrm{m}$, and $\pm 0.02\,\mathrm{m}$, respectively, while larger errors result from the ORB-SLAM2, i.e., $p_{e,x} = \pm 0.2\,\mathrm{m}$, $p_{e,y} \in [-0.2, 0.1]^{\mathrm{T}}\,\mathrm{m}$, and $p_{e,z} = \pm 0.05\,\mathrm{m}$. The orientation errors achieved with the two methods are similar, shown at the bottom of Fig. 7. However, the orientation errors measured by the proposed system are slightly better since the spikes in ORB-SLAM2 are larger. More experiments can be found in the supplementary material.

4.5.2 Outdoor Flight Experiments

Outdoor experiments were conducted to demonstrate the capability of the proposed system to detect and track motions at very high speeds. In the first scenario, the drone is equipped with an ALM and the event-based camera is mounted vertically on a tripod on the ground, see

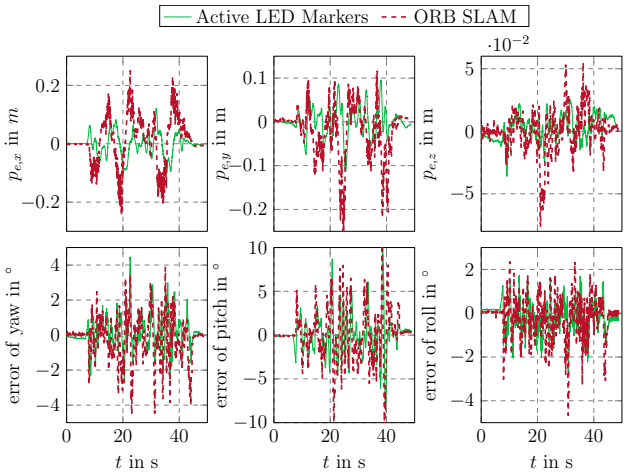


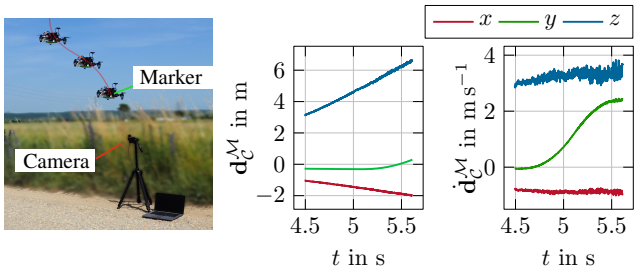
Figure 7. The error plots of the corresponding estimates, depicted in Fig. 6 with respect to the ground-truth measurements.

Fig. 8a. Although the drone moves at a maximum speed of $4.5\,\mathrm{m\,s^{-1}}$ and ascends to a height of $9\,\mathrm{m}$, the proposed system is still able to capture the trajectories of the ALM, depicted in Fig. 8a. The position signals $\mathbf{d}_{\mathcal{C}}^{\mathcal{M}}$ indicate a low noise floor. The velocity signals $\dot{\mathbf{d}}_{\mathcal{C}}^{\mathcal{M}}$ are calculated based on the position signals with additional moving average filtering. Unlike in the first scenario, the camera is mounted on the drone and the ALM is static on the ground in the second scenario, as illustrated in Fig. 8b. The captured trajectories are shown in Fig. 8a when the drone is moving with an average speed of $10\,\mathrm{m\,s^{-1}}$. In both scenarios, the velocity in the z direction is noisy due to the higher noise in the z estimation by the PnP algorithm. Live videos of the two scenarios are provided in the supplementary material.

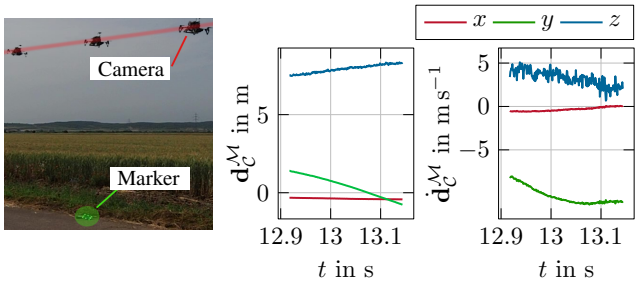
4.6. Limitations

The performance of the proposed system is mainly limited by the resolution of the event-based camera and the power of the light source. Based on the sensor resolution, the ground sampling distance (GSD) using the $8\,\mathrm{mm}$ lens at $10\,\mathrm{m}$ is equal to $0.61\,\mathrm{cm}$. This sets an upper bound on the accuracy of the system, even with sub-pixel tracking precision and precisely calibrated camera. Light intensity decreases with the square of the distance [28]. Hence, LEDs with $1\,\mathrm{W}$ at 10% duty cycle were used in the experiments. Altogether, the maximum working distance is around 10m.

In favor of accuracy, ALMs with occluded LEDs are not tracked. However, occlusion of LEDs is detected to determine if the tracking is lost. An ALM is respawned when all LEDs are visible again. As the proposed system features a small marker size intended for outdoor usage, where other positioning systems are not applicable, LED occlusion is not a primary concern.



(a) Camera on the ground.



(b) Camera on the drone.

Figure 8. The plot of the drone's trajectory and velocities during experiments. The derivative of the positional signal $\mathbf{d}_{\mathcal{C}}^{\mathcal{M}}$ was filtered with a moving average filter with a window length of 100 samples.

5. Conclusion

This paper presents a fast and accurate vision-based localization system using an event-based camera with active led markers. Our proposed method overcomes the limits of traditional marker-based localization systems, *i.e.* low frame rate, motion blur, and high computational costs, by utilizing the advantages of an event-based camera. The proposed algorithm is simple but effective, achieving real-time performance with minimal latency below $0.5\,\mathrm{ms}$ and output rates above $3\,\mathrm{kHz}$ using a regular PC. The proposed tracking-based approach outperforms detection-based methods, especially in applications with very fast movements. the position error normalized to the distance is constantly below $1.87\,\%$ with a mean orientation error of $0.738°$. To the best of the authors' knowledge, the combination of the achieved precision at this output rate and latency was not achieved so far. For the applications, where latenecy is not crucial, output of the system can be filtered and fused with other modalities (IMU, RBG based localistaion systems). Proposed system can be used as a cheap relative localization/reference positioning system for outdoor application as data collection where other systems can not be applied (dynamic scenes, fast motion). Also the proposed method opens new possibilities for robotic applications where the high output rates and high precision of 6-DoF pose estimation are important, *e.g.* dynamic handover tasks and pick-and-place tasks.

References

[1] Biases — metavision SDK docs 4.1.0 documentation. 5

[2] Burak Benligiray, Cihan Topal, and Cuneyt Akinlar. Stag: A stable fiducial marker system. *Image and Vision Computing*, 89:158–169, 2019. 7

[3] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021. 1, 7

[4] Andrea Censi, Jonas Strubel, Christian Brandli, Tobi Delbruck, and Davide Scaramuzza. Low-latency localization by active LED markers tracking using a dynamic vision sensor. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 891–898, 2013. 3, 4, 7

[5] William Chamorro, Joan Solà, and Juan Andrade-Cetto. Event-based line slam in real-time. *IEEE Robotics and Automation Letters*, 7(3):8146–8153, 2022. 7

[6] Guang Chen, Wenkai Chen, Qianyi Yang, Zhongcong Xu, Longyu Yang, Jörg Conradt, and Alois Knoll. A novel visible light positioning system with event-based neuromorphic vision sensor. *IEEE Sensors Journal*, 20(17):10211–10219, 2020. 3, 7

[7] Siyuan Chen, Dong Yin, and Yifeng Niu. A Survey of Robot Swarms' Relative Localization Method. *Sensors*, 22(12):4424, 2022. 2

[8] Toby Collins and Adrien Bartoli. Infinitesimal plane-based pose estimation. *International Journal of Computer Vision*, 109(3):252–286. 3

[9] Tobi Delbruck and Manuel Lang. Robotic goalie with 3 ms reaction time at 4event-based dynamic vision sensor. *Frontiers in Neuroscience*, 7:223, 2013. 2

[10] Guillermo Gallego, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J. Davison, Jorg Conradt, Kostas Daniilidis, and Davide Scaramuzza. Event-Based Vision: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):154–180, 2022. 1, 2

[11] Milad Heydariaan, Hessam Mohammadmoradi, and Omprakash Gnawali. Toward standard non-line-of-sight benchmarking of ultra-wideband radio-based localization. In *IEEE Workshop on Benchmarking Cyber-Physical Networks and Systems*, pages 19–24, 2018. 1

[12] Javier Hidalgo-Carrió, Guillermo Gallego, and Davide Scaramuzza. Event-aided direct sparse odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5781–5790, 2022. 7

[13] Michail Kalaitzakis, Brennan Cain, Sabrina Carroll, Anand Ambrosi, Camden Whitehead, and Nikolaos Vitzilaios. Fiducial markers for pose estimation. *Journal of Intelligent & Robotic Systems*, 101(4):71, 2021. 2

[14] Iman Abaspur Kazerouni, Luke Fitzgerald, Gerard Dooly, and Daniel Toal. A survey of state-of-the-art on visual slam. *Expert Systems with Applications*, 205:117734, 2022. 1

[15] Oguz Kedilioglu, Tomás Marcelo Bocco, Martin Landesberger, Alessandro Rizzo, and Jörg Franke. Arucoe: Enhanced aruco marker. In *International Conference on Control, Automation and Systems*, pages 878–881, 2021. 1

[16] Beat Kueng, Elias Mueggler, Guillermo Gallego, and Davide Scaramuzza. Low-latency visual odometry using event-based feature tracks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 16–23, 2016. 2

[17] Xavier Lagorce, Cédric Meyer, Sio-Hoi Ieng, David Filliat, and Ryad Benosman. Asynchronous event-based multikernel algorithm for high-speed visual features tracking. *IEEE Transactions on Neural Networks and Learning Systems*, 26(8):1710–1720, 2015. 4

[18] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128×128 120 dB 15μs Latency Asynchronous Temporal Contrast Vision Sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. 1

[19] K. Madsen, H.B. Nielsen, O. Tingleff, and Danmarks tekniske universitet. Informatik og Matematisk Modellering. *Methods for Non-linear Least Squares Problems*. Informatics and Mathematical Modelling, Technical University of Denmark, 2004. 3

[20] Elias Mueggler, Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. Continuous-time visual-inertial odometry for event cameras. *IEEE Transactions on Robotics*, 34(6):1425–1440, 2018. 2

[21] Elias Mueggler, Basil Huber, and Davide Scaramuzza. Event-based, 6-DOF pose tracking for high-speed maneuvers. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2761–2768, 2014. 2

[22] Georg R. Muller and Jorg Conradt. A miniature low-power sensor system for real time 2D visual tracking of LED markers. In *IEEE International Conference on Robotics and Biomimetics*, pages 2429–2434, 2011. 3

[23] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017. 7

[24] Anh Nguyen, Thanh-Toan Do, Darwin G. Caldwell, and Nikos G. Tsagarakis. Real-time 6dof pose relocalization for event cameras with stacked spatial lstm networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1638–1645, 2019. 1

[25] David Reverter Valeiras, Xavier Lagorce, Xavier Clady, Chiara Bartolozzi, Sio-Hoi Ieng, and Ryad Benosman. An asynchronous neuromorphic event-driven visual part-based shape tracking. *IEEE Transactions on Neural Networks and Learning Systems*, 26(12):3045–3059, 2015. 4

[26] Nasir Saeed, Haewoon Nam, Tareq Y Al-Naffouri, and Mohamed-Slim Alouini. A state-of-the-art survey on multidimensional scaling-based localization techniques. *IEEE Communications Surveys & Tutorials*, 21(4):3565–3583, 2019. 1

[27] Wilson Sakpere, Michael Adeyeye Oshin, and Nhlanhla Mlitwa. A state-of-the-art survey of indoor positioning and navigation systems and technologies. *South African Computer Journal*, 29:145, 2017. 2

[28] Mohammed Salah, Mohammed Chehadah, Muhammed Humais, Mohammed Wahbah, Abdulla Ayyad, Rana Azzam,

Lakmal Seneviratne, and Yahya Zweiri. A neuromorphic vision-based measurement for robust relative localization in future space exploration missions. *CoRR*, 2022. 3, 7, 8

[29] Yvonne Stürz, Lukas Affolter, and Roy Smith. Parameter identification of the KUKA LBR iiwa robot including constraints on physical feasibility. *IFAC-PapersOnLine*, 50(1):6863–6868, 2017. 6

[30] Minh Nhat Vu, Florian Beck, Christian Hartl-Nesic, Anh Nguyen, and Andreas Kugi. Machine learning-based framework for optimally solving the analytical inverse kinematics for redundant manipulators. *Mechatronics*, 91:102970, 2023. 6

[31] Viktor Walter, Martin Saska, and Antonio Franchi. Fast Mutual Relative Localization of UAVs using Ultraviolet LED Markers. In *International Conference on Unmanned Aircraft Systems*, pages 1217–1226, 2018. 2

[32] David Weikersdorfer and Jörg Conradt. Event-based particle filtering for robot self-localization. In *IEEE International Conference on Robotics and Biomimetics*, pages 866–870, 2012. 2

[33] Henk Wymeersch, Jiguang He, Benoit Denis, Antonio Clemente, and Markku Juntti. Radio localization and mapping with reconfigurable intelligent surfaces: Challenges, opportunities, and research directions. *IEEE Vehicular Technology Magazine*, 15(4):52–61, 2020. 1

[34] Xudong Yan, Heng Deng, and Quan Quan. Active infrared coded target design and pose estimation for multiple objects. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 6885–6890, 2019. 2

[35] Masaki Yoshino, Shinichiro Haruyama, and Masao Nakagawa. High-accuracy positioning system using visible led lights and image sensor. In *IEEE Radio and Wireless Symposium*, pages 439–442, 2008. 2

[36] Xiang Zhang, Stephan Fronz, and Nassir Navab. Visual marker detection and decoding in ar systems: A comparative study. In *International Symposium on Mixed and Augmented Reality*, pages 97–106, 2002. 1

[37] Yong Zhao, Shibiao Xu, Shuhui Bu, Hongkai Jiang, and Pengcheng Han. Gslam: A general slam framework and benchmark. In *IEEE/CVF International Conference on Computer Vision*, pages 1110–1120, 2019. 1