

000
001
002054
055
056

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

003
004
005
006
007057
058
059
060
061008
009
010
011062
063
064
065012
013066
067014
015068
069016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107

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 ms while maintaining output rates of 3 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

Paper ID 282

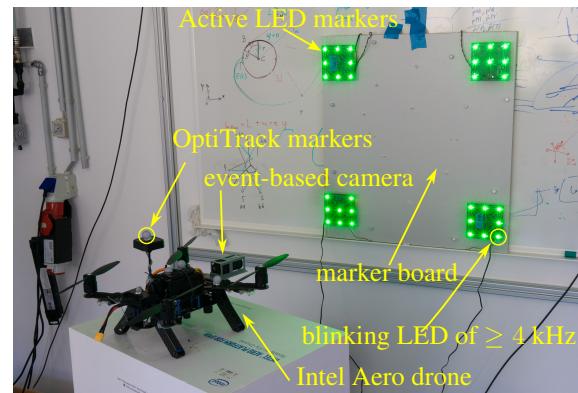


Figure 1. Overview of the experimental setup. Active LED markers (ALM) are attached to a marker board. An event-based camera mounted on a drone is used to estimate the pose of the marker board with respect to the event-based camera's frame of reference. An OptiTrack system is used for assessing the accuracy of the proposed method.

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 [2, 12, 36] are the most common ap-

proaches 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 [10, 32]. Vision-based approaches are, however, computationally expensive and typically require more than one sensor, e.g. infrared-based systems [25]. To reduce the computational overhead, classical markers [13, 35] 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.

Event-based vision is an emerging field that has attracted much attention in recent years [9, 23]. Compared to conventional cameras, event-based cameras capture changes in light intensity asynchronously with high temporal resolution and a large dynamic range. This feature makes them ideal for applications requiring fast and accurate detection.

*These authors contributed equally to this work.

108 Starting from the early years of event-based vision development [8], advantages given by those sensors for robotic
 109 applications are noticeable. Due to the mediocre signal-to-
 110 noise ratio and low resolution capabilities, applications are
 111 limited and fall far behind standard RGB cameras in terms
 112 of performance. However, recent advancements in event-
 113 based sensor development [9] have enabled them to com-
 114 pete with the precision of other localization methods [6] due
 115 to increased resolution and reduced noise. Avoiding accu-
 116 mulated event representations (*i.e.* frames), markers can be
 117 tracked online utilizing the event-based camera's high tem-
 118 poral resolution of up to 1 μ s.
 119

120 In this paper, we propose a fast and simple method em-
 121 ploying an event-based camera together with active LED
 122 markers (ALMs) for simultaneous detection and tracking of
 123 the 6 degrees-of-freedom (DoF) pose of a rigid object in
 124 the 3D space. An overview of our proposed approach is de-
 125 picted in Fig. 1 with four ALMs attached to a marker board.
 126 The event-based camera is mounted on the drone, which is
 127 utilized to estimate the pose of the marker board with re-
 128 spect to the camera's base frame. To estimate the pose, the
 129 blinkings of the LEDs are logged with an event-based cam-
 130 era to identify the corresponding frequencies of each LED
 131 in the ALM. These blinking frequencies are utilized to iden-
 132 tify each individual LED and match it with the known geo-
 133 metry of the ALM. With this mapping of the individual
 134 points on the camera's sensor plane and the known geo-
 135 metry of the ALM, the pose of the ALM can be computed
 136 by utilizing a Perspective-n-Point (PnP) algorithm. In the
 137 presented approach, by tuning the biases, *i.e.*, parameters
 138 for tuning the analog front-end of the event-based camera,
 139 and using a priori knowledge about timing, the complexity
 140 of the ALM tracking can be simplified. This aids in reduc-
 141 ing the tracking latency. During the tracking, the initial de-
 142 tection is continuously refined, resulting in subpixel resolu-
 143 tion. Such an approach can still precisely estimate the pose
 144 even under fast rotational and linear motion. The proposed
 145 approach was tested and verified extensively using an ex-
 146 ternal infrared-based positioning system. Our contributions
 147 are listed in the following.
 148

- 149 • We propose a fast event-based pose estimation system
 150 using ALM achieving a latency below 0.5 ms while
 151 maintaining an output rate of 3 kHz.
- 152 • We analyze the proposed system in static and dynamic
 153 scenarios for several in-depth aspects, *e.g.*, absolute
 154 accuracy, static noise, and latency. Translational errors
 155 of $34.5 \text{ mm} \pm 16 \text{ mm}$ and $0.74^\circ \pm 0.15^\circ$ orientation er-
 156 rors at distances of 2.1 m to 4.8 m between the camera
 157 and the marker were achieved. Together with the fast
 158 computing speed, this proves that the proposed algo-
 159 rithm is promising for real-time applications.
- 160 • We integrate the proposed system into a quadcopter

162 application for the 6-DoF pose estimation task. For
 163 indoor experiments, the proposed system outperforms
 164 the ORB-SLAM algorithm. Furthermore, in outdoor
 165 experiments, the proposed system can simultaneously
 166 detect and track the ALM in very aggressive flights at
 167 velocities of up to 10 m s^{-1} .
 168

169 The paper is organized as follows: Section 2 presents the
 170 related work in the field of pose estimation with event-based
 171 cameras and active markers. Section 3 describes the pro-
 172 posed method for marker detection and tracking. In Section
 173 4, we present the experimental setup and results. Finally,
 174 we conclude the paper in Section 5 with a summary of our
 175 contributions and suggestions for future work.
 176

2. Related Work

177 Visual localization systems show improved accuracy
 178 compared to systems based on other physical principles
 179 [26], [6]. Fiducial marker-based systems [11] constitute
 180 the most common choice for robotic applications. Due to
 181 the limited range and the dependence on the lighting con-
 182 ditions, some studies proposed LED-based solutions based
 183 on standard RGB cameras [34], infrared [33], or ultraviolet
 184 [30] spectrum. However, the latency cannot be reduced
 185 beyond the camera's frame rate.
 186

187 One of the first works in the direction of localiza-
 188 tion based on event-based sensors was the 2D localiza-
 189 tion method [31]. The known shape (contours) was tracked, and
 190 the relative localization was determined by event-based vi-
 191 sion. The high temporal resolution of the event-based sen-
 192 sors was used in [20] to localize an Unmanned Aerial Ve-
 193 hicle (UAV) during high-speed maneuvers. The pose infor-
 194 mation was retrieved using a black square as a known shape.
 195 In [14], a visual odometry method was proposed based on
 196 the feature tracking algorithm. In this direction, multiple
 197 methods were developed [14], [19], which show a signif-
 198 icant improvement compared to the RGB-based approach
 199 for high-speed applications.
 200

201 The utilization of ALMs was proposed first in [21],
 202 where the authors tracked the 2D position of the LED and
 203 used it as a feedback signal for the robot homing and a pan
 204 tilt system. Later, the first method for pose estimation using
 205 ALMs was presented in [3]. Therein, ALMs were used to
 206 detect and estimate the position of a flying quadrocopter.
 207 LEDs were recognized and detected using event polarity
 208 changes in the event stream. In [3], the authors used an ac-
 209 cumulated event representation to decode the frequency and
 210 estimate the pose. In [5], a Gaussian mixture probability hy-
 211 pothesis density filter was proposed to localize the camera
 212 with respect to the active marker. Therein, online tracking
 213 was presented to increase the robustness and reliability of
 214 the pose estimation. The achieved results indicate a local-
 215 ization error lower than 3 cm in scenarios where the camera
 216

216 was within 1 m relative to the active marker. It is worth noting that in [5], neither standard deviation nor mean values
 217 are reported.
 218

219 Most recent works using ALMs propose the additional
 220 fusion of inertial measurements [27]. The error in the pre-
 221 dicted relative position is in the subcentimeter range. How-
 222 ever, utilizing only the vision-based approach increases the
 223 error by the order of one magnitude. Compared to previous
 224 methods, the marker size is significantly larger. The LEDs
 225 are placed 1 m apart. Current work in active marker-based
 226 solutions also focuses on the visual communication aspect
 227 of modulated light.
 228

229 Different from other approaches in the literature, our ap-
 230 proach simplifies the complexity by tuning the biases and
 231 using a priori knowledge about timing. This helps to reduce
 232 the tracking latency. To the best of the authors' knowl-
 233 edge, this work achieves the lowest latency compared to
 234 other methods in the literature. During tracking, initial de-
 235 tection is continuously refined, resulting in subpixel resolu-
 236 tion. Such an approach can still precisely estimate the pose
 237 even under fast rotational and linear motion. The output rate
 238 is independent of the tracking performance and is limited
 239 only by the Perspective-n-Point (PnP) algorithm [17]. The
 240 method allows simultaneous tracking of multiple markers
 241 and is computationally undemanding, allowing it to be run
 242 on a wide range of devices.
 243

3. Active Marker Tracking and Pose Estimation

244 An event-based camera consists of an array of indepen-
 245 dent pixels measuring changes in luminosity $L = \log(I)$,
 246 based on the photocurrent I [16]. A change in the continu-
 247 ous luminosity signal
 248

$$249 \Delta L(\mathbf{u}_k, t_k) = L(\mathbf{u}_k, t_k) - L(\mathbf{u}_k, t_k - \Delta t_k) > p_k C \quad (1)$$

250 triggers an event $\mathbf{e}_k = (\mathbf{u}_k, t_k, p_k)$ at pixel location $\mathbf{u}_k =$
 251 (u_k, v_k) due to a temporal contrast threshold $\pm C$, $p_k \in$
 252 $\{+1, -1\}$ being its polarity and Δt_k the time since the last
 253 event (at \mathbf{u}_k) occurred at t_k [9]. The state-of-the-art event-
 254 based sensors can produce up to 1.2 Giga events per sec-
 255 ond (Geps) [9] with a microsecond range timestamp accu-
 256 racy. Compared to frame-based cameras that deliver period-
 257 ically dense (*i.e.* full-frame) information, the event stream is
 258 sparse and contains information that relates only to changes
 259 in the scene. Using ALMs, periodic and dense signals can
 260 be generated as a projection of the LED on the camera's
 261 sensor plane.
 262

263 To reduce the computational complexity and the required
 264 bandwidth, the biases of the sensor are tuned to generate a
 265 single event per pixel on every LED blink while suppressing
 266 all other background events to increase the signal-to-noise
 267 ratio, as presented in Figure 2. While [3] uses events of
 268

269 both polarities and relatively low frequencies (1-2kHz), the
 270 amount of noise can be reduced by using higher frequencies
 271 and disabling one polarity.
 272

273 The ALM's structure is an arrangement of high-
 274 frequency blinking LEDs, where a unique frequency of the
 275 blinking pattern can individually recognize each LED (*e.g.*
 276 different blinking frequencies). The arrangement of the
 277 LEDs has to be fixed and determined in the 3D coordi-
 278 nate space. However, it can also be arranged on a plane,
 279 as utilized in this work. Based on the 2D projection of the
 280 LEDs, knowing their 3D arrangement and camera intrin-
 281 sics, the relative pose of the marker with respect to the sen-
 282 sor can be reconstructed using a Point-n-Perspective (PnP)
 283 algorithm [18]. In this work, the IPPE PnP algorithm is
 284 used [7].
 285

286 The proposed approach is divided into four parts, as il-
 287 lustrated in Fig. 2. To reduce the noise in the signal, the bias
 288 settings are tuned to produce a single event per pixel on ev-
 289 ery blink of a LED. Next, events are accumulated over the
 290 time $\frac{2}{f_{\min}}$, which is two times the period of the LED's mini-
 291 mal frequency f_{\min} (typ. $f_{\min} \approx 2$ kHz). For those accumu-
 292 lated event clusters, frequencies are recognized. These fre-
 293 quencies are used to identify newly appearing ALMs. For
 294 each of the ALM's LEDs, trackers are spawned that keep
 295 track of the LEDs' center points based on single events. The
 296 tracking of the LEDs is independent of the detection loop.
 297 The pose of the ALM is estimated utilizing the trackers of
 298 the ALM. The accuracy of the pose estimation can be ob-
 299 tained using the reprojection error. When an ALM leaves
 300 the field of view or the reprojection error exceeds the de-
 301 fined maximal value, the corresponding trackers are deleted.
 302 If the ALM enters the field of view again, the detection al-
 303 gorithm respawns it. Such an approach reduces the latency
 304 and increases the accuracy of the solution.
 305

3.1. Detection

306 For the detection of an ALM, the geometrical arrange-
 307 ment and the blinking frequency of the ALM LEDs, have to
 308 be provided in advance.
 309

310 The range of possible frequencies for the LEDs is wide:
 311 from tests conducted, frequencies higher than 4kHz and
 312 lower than 40kHz work best. To detect lower frequencies,
 313 biases have to be adapted to maximize the signal-to-noise
 314 ratio. As the timestamp is quantized, it is advisable to use
 315 LED frequencies with an integer microsecond period.
 316

317 Due to the limited noise, detection can be simplified by
 318 using only single types of events. In [27] and [3], the de-
 319 tections rely on the transitions between event polarities. In-
 320 stead, in the proposed approach, we use the timing informa-
 321 tion between consecutive events generated by a single pixel.
 322

323 For detection, an event frame generated over the period
 324 T_d is used, where T_d has to be larger than $\frac{2}{f_{\min}}$ to ensure that
 325 at least two blinks are visible for every LED. Candidates
 326

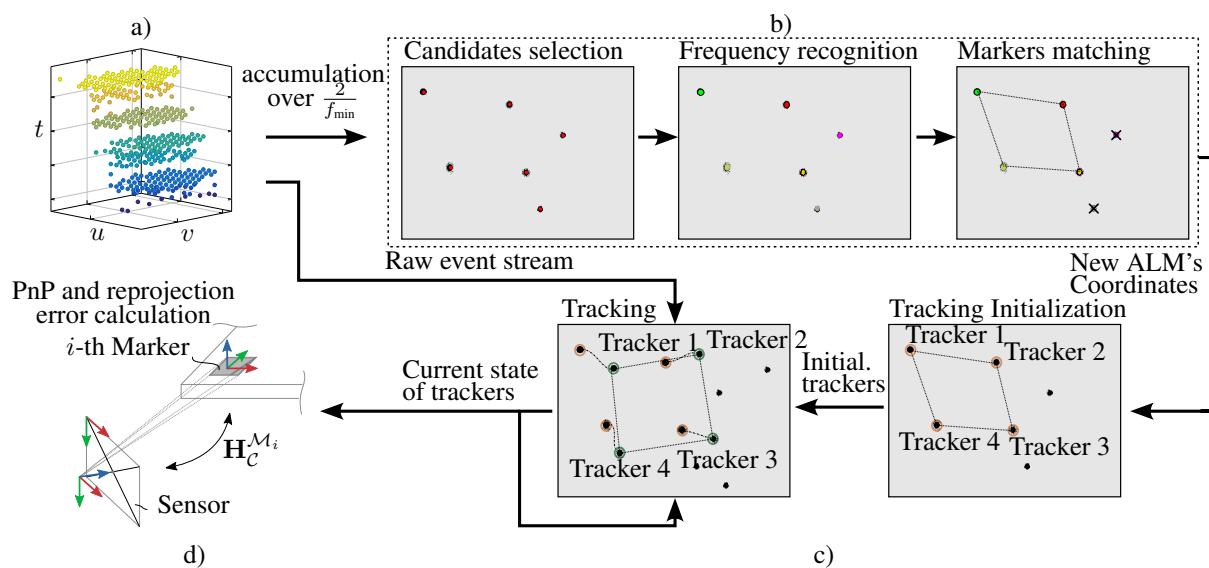


Figure 2. Overview of the proposed approach. Our pipeline consists of four asynchronous parts. *First*, illustrated in (a), to reduce the noise in the signal, biases are tuned to produce a single event per pixel on every blink. The proposed detection algorithm accumulates the events over a short period 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.

for the blinking LEDs can be retrieved by selecting the connected regions where more than $T_d f_{\min}$ 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. 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 [3].

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 $c_i = [x_i, y_i]$, and radius r_i . In comparison to the assumptions of [15] and [24], 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 al-

gorithm while maintaining precise tracking with sub-pixel accuracy.

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

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

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.

324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431

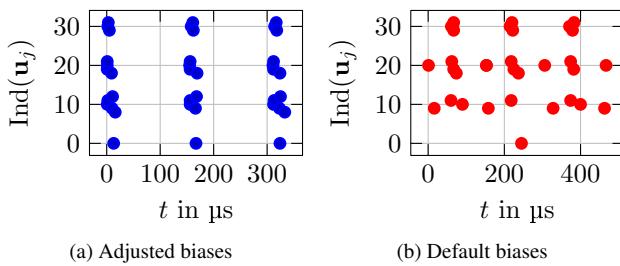


Figure 3. Visualization of bias adjustment in an event camera using the IMX636ES sensor. The vertical axis represents the flattened indices of the pixels in the region of interest (ROI) around an LED light. The left plot demonstrates an optimal bias adjustment, where event fronts - closely packed clusters of events triggered by a rapid change in the scene (like a sudden LED blink) - are clearly distinct. No unwanted spurious events occur between these event fronts. The right plot, in contrast, displays the event distribution with the camera's default bias settings, showing a less distinct separation of event fronts.

4. Experiments

4.1. Setup

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 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 side length of 59 cm. The 8 outermost LEDs were chosen to create a single marker. The camera's biases were chosen to minimize noise while maintaining a high event rate caused by the ALMs. The processing of the event stream is done on a PC running Ubuntu 20.04 with an Intel i9-12900K processor and 32GB of RAM and Intel Aero Compute Board. The camera is connected via USB 3 and the camera's external trigger input is used for time synchronization between the PC and the camera. As ground truth, the commercial infrared 3D tracking system OptiTrack is used, comprising nine Prime 17W cameras. The OptiTrack recordings are triggered with the same external trigger signal as the event-based camera. In order to use the event-based camera simultaneously with the OptiTrack system, an infrared filter is used.

4.2. 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, 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. 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, as shown in Figure Fig. 3.

A detailed explanation of the sensor biases is provided in [1]. The description of the bias adjustment procedure is given in the supplementary document. Although the focus here is on the IMX636ES sensor, the setup procedure should be applicable to cameras from other manufacturers.

4.3. 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 $\|\mathbf{d}_c^M\|$, ranging from 2.1 m to 4.8 m, as well as the orientation Θ_c^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 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 Θ_c^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 $\tilde{\mathbf{e}}_t = \frac{\|\mathbf{e}_t\|}{\|\mathbf{d}_c^M\|}$ is displayed in Fig. 5 as a function of the

432	486
433	487
434	488
435	489
436	490
437	491
438	492
439	493
440	494
441	495
442	496
443	497
444	498
445	499
446	500
447	501
448	502
449	503
450	504
451	505
452	506
453	507
454	508
455	509
456	510
457	511
458	512
459	513
460	514
461	515
462	516
463	517
464	518
465	519
466	520
467	521
468	522
469	523
470	524
471	525
472	526
473	527
474	528
475	529
476	530
477	531
478	532
479	533
480	534
481	535
482	536
483	537
484	538
485	539

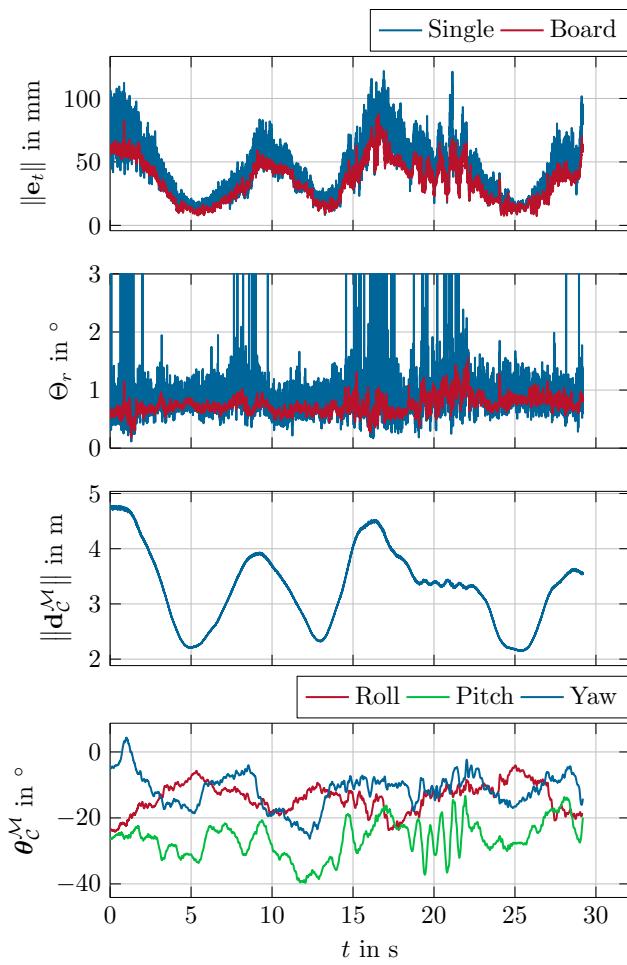


Figure 4. The ALM and the marker board are placed statically in the scene, and the camera moves from close to far. The first plot shows the absolute position error of a single ALM and the whole marker board. The second plot displays the orientation error, with the third and fourth plot indicating the distance $\|\mathbf{d}_c^M\|$ ($\min \|\mathbf{d}_c^M\| = 2.1$ m, $\max \|\mathbf{d}_c^M\| = 4.8$ m) and the orientation Θ_c^M between the camera and the marker.

normalized distance $\tilde{d} = \frac{\|\mathbf{d}\|_c^M}{\max \|\mathbf{d}\|_c^M}$. 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.

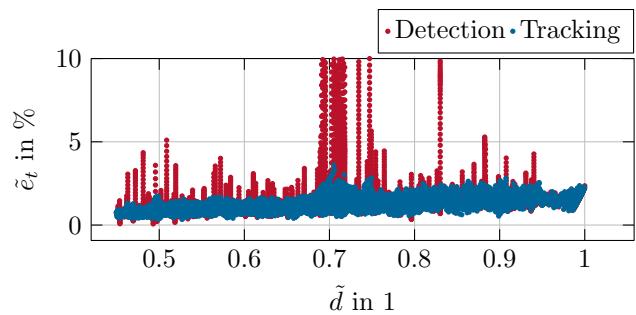


Figure 5. Comparison between the detection and the tracking algorithm for pose estimation. The relative position error \tilde{e}_t is plotted over the normalized distance \tilde{d} .

Table 1. Statistical values of the absolute accuracy measurements for the detection-based and the tracking-based approach.

	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°

4.4. 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) in the subsequent data processing in applications.

Table 2. Statistics of the noise in static scenes at different distances between marker and camera.

$\ \mathbf{d}_c^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

4.5. 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 and Tab. 4. The output rate of the tracking-based approach outperforms the detection-based method while achieving comparable latency results. As shown in Tab. 4, the proposed method is capable of running even on an embedded PC with an Intel Atom processor on an Intel Aero Ready to Fly

648 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. To further reduce the latency, it is possible to implement this solution on different hardware architectures, *e.g.* FPGAs. A large part of the resulting latency is due to communication overhead.

662 Table 3. Latency and output rates for the detection-based and the
663 tracking-based approach. (Desktop PC)

	Tracking		Detection	
	latency	rate	latency	rate
Mean	354 μ s	3.805 kHz	699 μ s	0.670 kHz
Std. Dev.	92 μ s	0.064 kHz	35 μ s	0.288 kHz

670 Table 4. Latency and output rates for the detection-based and the
671 tracking-based approach. (Intel Aero Compute Board)

	Tracking		Detection	
	latency	rate	latency	rate
Mean	1232 μ s	1.32 kHz	1953 μ s	0.223 kHz
Std. Dev.	194 μ s	0.14 kHz	240 μ s	0.094 kHz

679 4.6. Application to 6-DoF position estimation for a 680 quadcopter

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

684 4.6.1 Indoor flight experiments

686 Compared to stationary robots, *e.g.* articulated manipulators [28, 29], 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 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 of all algorithms.

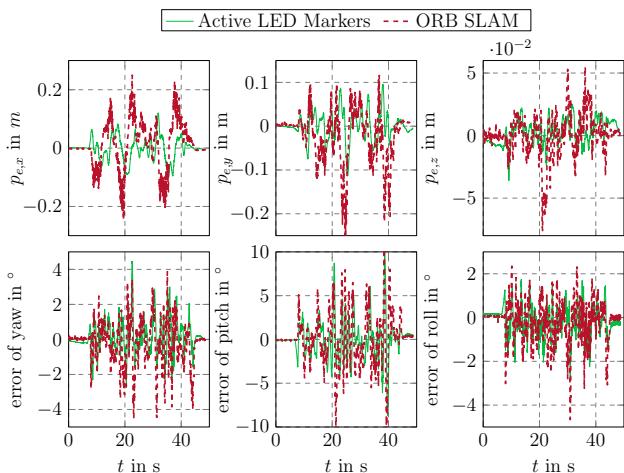
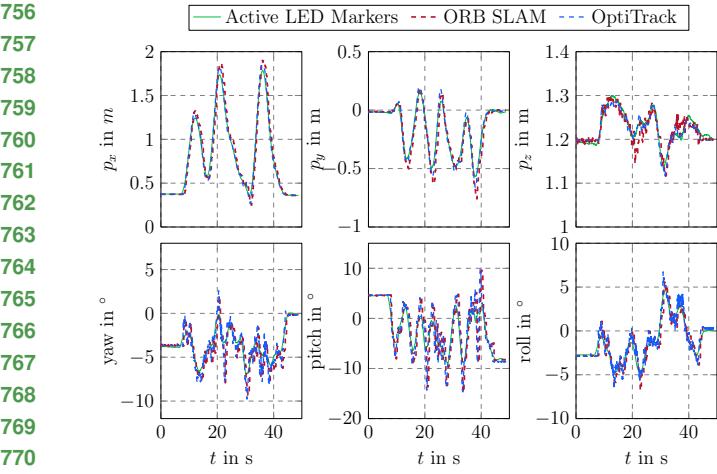
697 For absolute drift-free pose information, the ORB-SLAM2 [22] algorithm is utilized. The ORB-SLAM2 was chosen for its impressive performance and open-source implementation. Note that ORB-SLAM3 [2], as the successor of ORB-SLAM2, is a more robust version compared to

702 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 [22] and event-based SLAM [4], 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. Similar to previous subsections, the OptiTrack serves as the source of ground truth.

703 For the position errors, the metric for comparison is the difference between a ground-truth position and the estimated position as $\mathbf{p}_e = [p_{e,x} \ p_{e,y} \ p_{e,z}]^T = \mathbf{p}_g - \hat{\mathbf{p}}$, where the hat ($\hat{\cdot}$) indicates quantities estimated by the ORB-SLAM2 algorithm [22] 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 m in the x -direction, 0.6 m in y -direction, and 0.4 m in z -direction for about 50 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 ± 0.06 m, ± 0.08 m, and ± 0.02 m, respectively, while larger errors result from the ORB-SLAM2, *i.e.*, $p_{e,x} = \pm 0.2$ m, $p_{e,y} \in [-0.2, 0.1]^T$ m, and $p_{e,z} = \pm 0.05$ 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 ORBSLAM2 are larger. More experiments can be found in the supplementary material.

743 4.6.2 Outdoor Flight Experiments

745 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 Fig. 8a. Although the drone moves at a maximum speed of 4.5 m s^{-1} and ascends to a height of 9 m, the proposed system is still able to capture the trajectories of the ALM, depicted in Fig. 8c. The position signals \mathbf{d}_C^M indicate a low noise floor. The velocity signals $\dot{\mathbf{d}}_C^M$ are calculated based on the position signals with additional moving average fil-



tering. 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. 8d when the drone is moving with an average speed of 10 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.

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

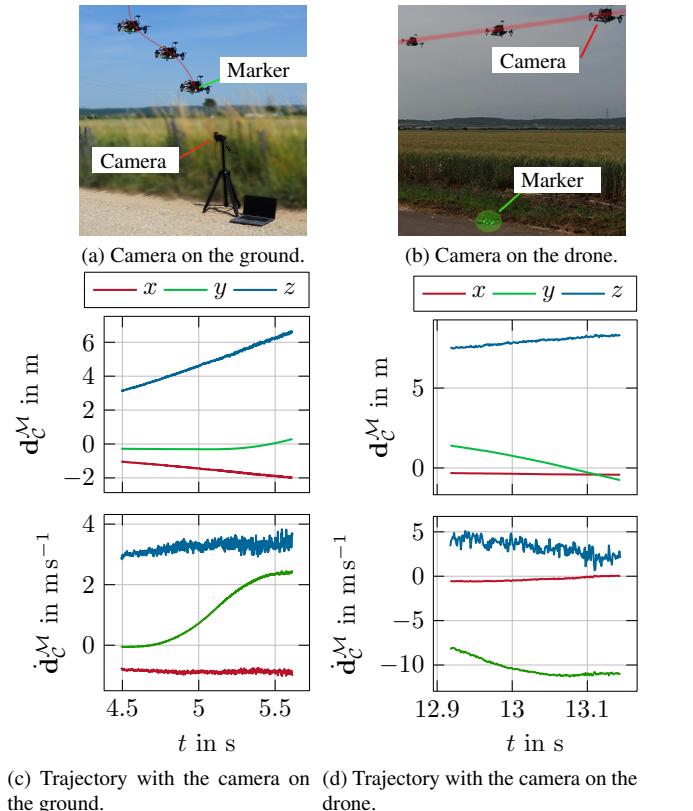


Figure 8. The plot of the drone's trajectory during experiments. The top plot illustrates the position with the corresponding velocity shown in the bottom plot. The derivative of the positional signal d_C^M was filtered with a moving average filter with a window length of 100 samples to obtain the plotted velocity signal.

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 ms and output rates above 3 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. 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.

864

References

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

- [1] Biases — metavision SDK docs 4.1.0 documentation. 5
- [2] 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
- [3] Andrea Censi, Jonas Strubel, Christian Brandli, Tobi Delbrück, 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. 2, 3, 4
- [4] 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
- [5] 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. 2, 3
- [6] Siyuan Chen, Dong Yin, and Yifeng Niu. A Survey of Robot Swarms’ Relative Localization Method. *Sensors*, 22(12):4424, 2022. 2
- [7] Toby Collins and Adrien Bartoli. Infinitesimal plane-based pose estimation. 109(3):252–286. 3
- [8] Tobi Delbrück and Manuel Lang. Robotic goalie with 3 ms reaction time at 4event-based dynamic vision sensor. *Frontiers in Neuroscience*, 7:223, 2013. 2
- [9] Guillermo Gallego, Tobi Delbrück, 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, 3
- [10] 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
- [11] Michail Kalitzakis, 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
- [12] 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
- [13] 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
- [14] 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
- [15] 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
- [16] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbrück. A 128×128 120 dB $15\mu s$ Latency Asynchronous Temporal Contrast Vision Sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. 3
- [17] Xiao Xin Lu. A review of solutions for perspective-n-point problem in camera pose estimation. In *Journal of Physics: Conference Series*, volume 1087, page 052009, 2018. 3
- [18] 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
- [19] 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
- [20] 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
- [21] 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. 2
- [22] 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
- [23] 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
- [24] 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
- [25] 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
- [26] Wilson Sakpere, Michael Adeyeye Oshin, and Nhlanhla Mlitiwa. A state-of-the-art survey of indoor positioning and navigation systems and technologies. *South African Computer Journal*, 29:145, 2017. 2
- [27] Mohammed Salah, Mohammed Chehadah, Muhammed Hu-mais, 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
- [28] 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. 7

- 972 [29] Minh Nhat Vu, Florian Beck, Christian Hartl-Nesic, Anh
973 Nguyen, and Andreas Kugi. Machine learning-based frame-
974 work for optimally solving the analytical inverse kinematics
975 for redundant manipulators. *Mechatronics*, 91:102970, 2023.
976 7
- 977 [30] Viktor Walter, Martin Saska, and Antonio Franchi. Fast Mu-
978 tual Relative Localization of UAVs using Ultraviolet LED
979 Markers. In *International Conference on Unmanned Aircraft
980 Systems*, pages 1217–1226, 2018. 2
- 981 [31] David Weikersdorfer and Jörg Conradt. Event-based particle
982 filtering for robot self-localization. In *IEEE International
983 Conference on Robotics and Biomimetics*, pages 866–870,
984 2012. 2
- 985 [32] Henk Wymeersch, Jiguang He, Benoit Denis, Antonio
986 Clemente, and Markku Juntti. Radio localization and map-
987 ping with reconfigurable intelligent surfaces: Challenges,
988 opportunities, and research directions. *IEEE Vehicular Tech-
989 nology Magazine*, 15(4):52–61, 2020. 1
- 990 [33] Xudong Yan, Heng Deng, and Quan Quan. Active infrared
991 coded target design and pose estimation for multiple objects.
992 In *IEEE/RSJ International Conference on Intelligent Robots
993 and Systems*, pages 6885–6890, 2019. 2
- 994 [34] Masaki Yoshino, Shinichiro Haruyama, and Masao Nakagawa.
995 High-accuracy positioning system using visible led
996 lights and image sensor. In *IEEE Radio and Wireless Sym-
997 posium*, pages 439–442, 2008. 2
- 998 [35] Xiang Zhang, Stephan Fronz, and Nassir Navab. Visual
999 marker detection and decoding in ar systems: A comparative
1000 study. In *International Symposium on Mixed and Augmented
1001 Reality*, pages 97–106, 2002. 1
- 1002 [36] Yong Zhao, Shibiao Xu, Shuhui Bu, Hongkai Jiang, and
1003 Pengcheng Han. Gslam: A general slam framework and
1004 benchmark. In *IEEE/CVF International Conference on Com-
1005 puter Vision*, pages 1110–1120, 2019. 1
- 1006
- 1007
- 1008
- 1009
- 1010
- 1011
- 1012
- 1013
- 1014
- 1015
- 1016
- 1017
- 1018
- 1019
- 1020
- 1021
- 1022
- 1023
- 1024
- 1025
- 1026
- 1027
- 1028
- 1029
- 1030
- 1031
- 1032
- 1033
- 1034
- 1035
- 1036
- 1037
- 1038
- 1039
- 1040
- 1041
- 1042
- 1043
- 1044
- 1045
- 1046
- 1047
- 1048
- 1049
- 1050
- 1051
- 1052
- 1053
- 1054
- 1055
- 1056
- 1057
- 1058
- 1059
- 1060
- 1061
- 1062
- 1063
- 1064
- 1065
- 1066
- 1067
- 1068
- 1069
- 1070
- 1071
- 1072
- 1073
- 1074
- 1075
- 1076
- 1077
- 1078
- 1079