



HAL
open science

DH-PTAM: A Deep Hybrid Stereo Events-Frames Parallel Tracking And Mapping System

Abanob Soliman, Fabien Bonardi, Désiré Sidibé, Samia Bouchafa

► **To cite this version:**

Abanob Soliman, Fabien Bonardi, Désiré Sidibé, Samia Bouchafa. DH-PTAM: A Deep Hybrid Stereo Events-Frames Parallel Tracking And Mapping System. IEEE Transactions on Intelligent Vehicles, In press, pp.1-10. 10.1109/TIV.2024.3412595 . hal-04620377

HAL Id: hal-04620377





<https://hal.science/hal-04620377>

Submitted on 21 Jun 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DH-PTAM: A Deep Hybrid Stereo Events-Frames Parallel Tracking And Mapping System

Abanob Soliman , *Member, IEEE*, Fabien Bonardi , Désiré Sidibé , *Senior Member, IEEE*, and Samia Bouchafa 

Abstract—This paper presents a robust approach for a visual parallel tracking and mapping (PTAM) system that excels in challenging environments. Our proposed method combines the strengths of heterogeneous multi-modal visual sensors, including stereo event-based and frame-based sensors, in a unified reference frame through a novel spatio-temporal synchronization approach. We employ deep learning-based feature extraction and description for estimation to enhance robustness further. We also introduce an end-to-end parallel tracking and mapping optimization layer complemented by a simple loop-closure algorithm for efficient SLAM behavior. Through comprehensive experiments on both small-scale and large-scale real-world sequences of VECtor and TUM-VIE benchmarks, our proposed method (DH-PTAM) demonstrates superior performance in terms of robustness and accuracy in adverse conditions, especially in large-scale HDR scenarios. Our implementation’s research-based Python API is publicly available on GitHub for further research and development: <https://github.com/AbanobSoliman/DH-PTAM>.

Index Terms—Stereo, events, SuperPoint, R2D2, SLAM.

I. INTRODUCTION

SIMULTANEOUS Localization and Mapping (SLAM) is pivotal in robotics and computer vision, aiming to chart unknown terrains while discerning an agent’s location. Among the key SLAM contributions is the Parallel Tracking and Mapping (PTAM) method [1], which uniquely separates tracking and mapping into parallel threads for augmented efficiency and real-time performance in monocular systems. However, PTAM faced scale ambiguity challenges inherent to monocular SLAM. Addressing this issue, its successor, Stereo Parallel Tracking and Mapping (S-PTAM) [2], leveraged stereo vision to extract depth information directly, eliminating scale ambiguity and fortifying robustness. In recent years, learning-based features extraction and description methods [3], [4], and deep learning based approaches [5] have been applied to improve robustness.

Manuscript received XX XX, XX; revised XX XX, XX; accepted XX XX, XX. Date of publication XX XX, XX; date of current version XX XX, XX. This article was recommended for publication by Editor X. XX upon evaluation of the Associate Editor and Reviewers’ comments.

This work was fully funded by a grant from the French Ministry of Higher Education, Research and Innovation as part of the doctoral thesis of A. Soliman, defended on October 5th, 2023, under Candidature No. 88303.

A. Soliman is with Continental Engineering Services Limited, Burgess Hill, United Kingdom. (Contact: abanob.soliman@conti-engineering.com).

F. Bonardi, D. Sidibé, and S. Bouchafa are with Université Paris-Saclay, Univ Evry, IBISC Laboratory, 91020, Evry, France.

Digital Object Identifier XX.XXXX/XXXXXX

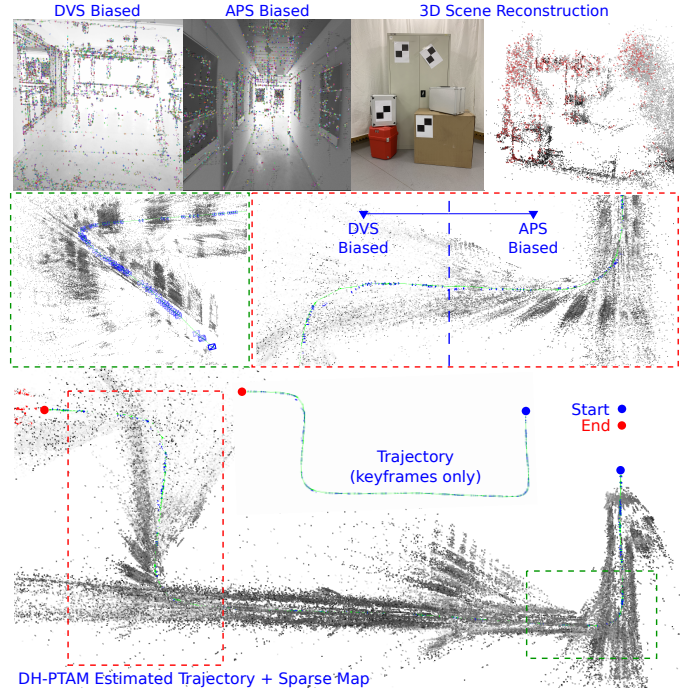


Fig. 1. Experiments on school-scooter and corner-slow sequences from the VECtor dataset show the estimated trajectory with the constructed scene map (green dotted rectangle). The red dotted rectangle highlights an HDR use-case where DH-PTAM estimates the trajectory continuously based on the two fusion-biased modes: Dynamic Vision Sensor (DVS) or Active Pixel Sensor (APS). APS denotes the standard camera’s global shutter frames. For more insights, please refer to: <https://www.youtube.com/watch?v=chAC-vQH9KU>

Visual Odometry (VO), an integral component of SLAM, has predominantly depended on conventional cameras to determine motion. However, such methods often fail in high dynamic range (HDR) scenarios where lighting conditions fluctuate [6] (see Fig. 1). Fortunately, the innovation of event cameras, also termed asynchronous or dynamic vision sensors (DVS) [7], offers a groundbreaking solution. Unlike their traditional counterparts that capture frames at fixed intervals, event cameras relay a continuous stream of “events” showcasing pixel-wise brightness alterations. This not only empowers them to operate at unparalleled speeds and in dimly lit conditions but also markedly diminishes motion blur [8].

With the ability to adeptly handle swift motions, scenes with significant HDR, and challenging lighting, event cameras emerge as superior alternatives, increasing both SLAM and

VO's robustness and precision in environments that would confound traditional cameras. The distinctive event-based design makes these sensors incredibly adapted at tracking fast-moving objects, setting them apart as an invaluable asset to visual odometry, especially under adverse conditions.

The integration of event-based and frame-based sensors in PTAM systems has shown significant potential for enhancing the robustness and accuracy of autonomous navigation technologies [9]. Leveraging the high temporal resolution and low latency of event-based sensors alongside the spatial resolution and contextual richness provided by frame-based imaging, this hybrid sensory approach is well-suited to address the challenges of rapid motion and high dynamic ranges found in diverse operational environments. Researchers have explored various methods for synchronizing spatio-temporal data from both sensor types [10], significantly improving the capabilities of PTAM systems to perform reliable real-time mapping and tracking in complex settings. These innovations are crucial in developing advanced navigation systems for intelligent vehicles.

Deep learning-based features are more robust than traditional methods [3], [4], as they can learn from large amounts of data and generalize well to unseen data. They are also more invariant to changes in viewpoint and lighting, making them suitable for real-world applications. Recently, pre-trained models have been widely adopted in computer vision and have achieved state-of-the-art performance in object detection, semantic segmentation, and image classification tasks.

This paper proposes a deep hybrid stereo events-frames parallel tracking and mapping system (see Fig. 2) that significantly improves SLAM accuracy and robustness in dynamic environments. This system combines the advantages of stereo RGB and event cameras, which can capture visual information at high temporal resolution. The use of deep learning techniques in this system allows for the extraction of robust features from the stereo hybrid image and event frames, which improves the accuracy of the feature-matching process and the estimation of the camera pose.

The main contributions can be summarized as follows:

- We propose an end-to-end parallel tracking and mapping (PTAM) approach based on a novel spatio-temporal synchronization of stereo visual frames with event streams.
- We propose a simple mid-level feature loop-closure algorithm for prompt SLAM behavior based on a learning-based feature description method to maximize robustness.
- DH-PTAM's effectiveness is evaluated in both stereo event-aided and image-based visual SLAM modes, achieving improved accuracy when incorporating event information, shown in an ablation study on the CPU versus the GPU of a consumer-grade laptop.

This paper is organized as follows: Section II gives a brief overview of the state-of-the-art SLAM methods. Section III provides an in-detail summary of the proposed method and offers insights into the novel parts of the algorithm. Section IV comprehensively evaluates the algorithm on the most recent VECtor [11] and TUM-VIE [12] benchmarks, along with defining the limitations. Section V summarizes

the experiments' main observations, the proposed method's behavioral aspects, and the start points for future works.

II. RELATED WORK

Events-Frames hybridization. Event-aided systems leverage the high-quality representations that events can produce after processing, especially in dynamic and dimmed environments where RGB camera frames fail. Some of the well-known event representations are event image (EI) [14], Time Surfaces (TS) [15], Event Spike Tensor (EST) [16], and recently Event 3-Channel Tensor (E3CT) [13]. Others [7] build the front-end on an Event Generation Model (EGM) [17] or construct motion-compensated event frames (MEF) [18] aided by a gyroscope. Towards a traditional frame reconstruction from events, [19] proposes a Log Intensity Reconstruction (LIR), a model-based method, and [20] proposes Spade-e2vid, a learning-based method.

Indirect methods [21], such as frame-based approaches, extract keypoints from the input data in the front-end. This front-end stage typically involves detecting and matching salient features in the sensory data, such as images or event streams. These keypoints are then passed to the back-end, where state estimation algorithms are used to estimate the robot's pose and build a consistent environment map.

Conversely, direct methods [22] attempt to process all available sensor data, such as individual pixel intensity changes in images (events) or all RGB frame pixels, without any intermediate filtering or feature extraction in the front-end, relying on the back-end to handle the entire data. The proposed method adopts a hybrid approach where all events are directly processed during the events-frames fusion in the front-end. In contrast, only the reliable learning-based features from the fusion frames are fed to the back-end (see Fig. 2).

Table I compares the latest event-based and event-aided VO solutions concerning the sensor setup, events pre-processing layer (EPL), direct or indirect event processing, and the loop-closure capability to minimize visual drifts.

Event-aided visual-SLAM. DH-PTAM builds upon the pioneering work of [7], which introduces a monocular 6-DoF visual odometry model that synergistically integrates events and grayscale frames using a direct approach. In this model, the fusion of events and frames is achieved by applying EGM to the intensity frame, transforming it into a unified reference that aligns with the event stream, termed as brightness increments (event-like). This process is followed by an error minimization cost function, which estimates the extrinsics in real-time for accurate mapping and is augmented by a photometric bundle adjustment for tracking. However, DH-PTAM distinguishes itself from [7] by employing an image-like unified reference in its front-end, harmoniously combining stereo event streams with stereo RGB frames.

Loop-closure detection is paramount in visual-SLAM, effectively minimizing drifts by allowing a system to recognize previously traversed locations. The realm of loop-closure detection predominantly features two methodologies: mid-level features [23] and the bag-of-words model [24]. While mid-level features offer a more nuanced representation than low-level features such as edges and corners, they don't consider

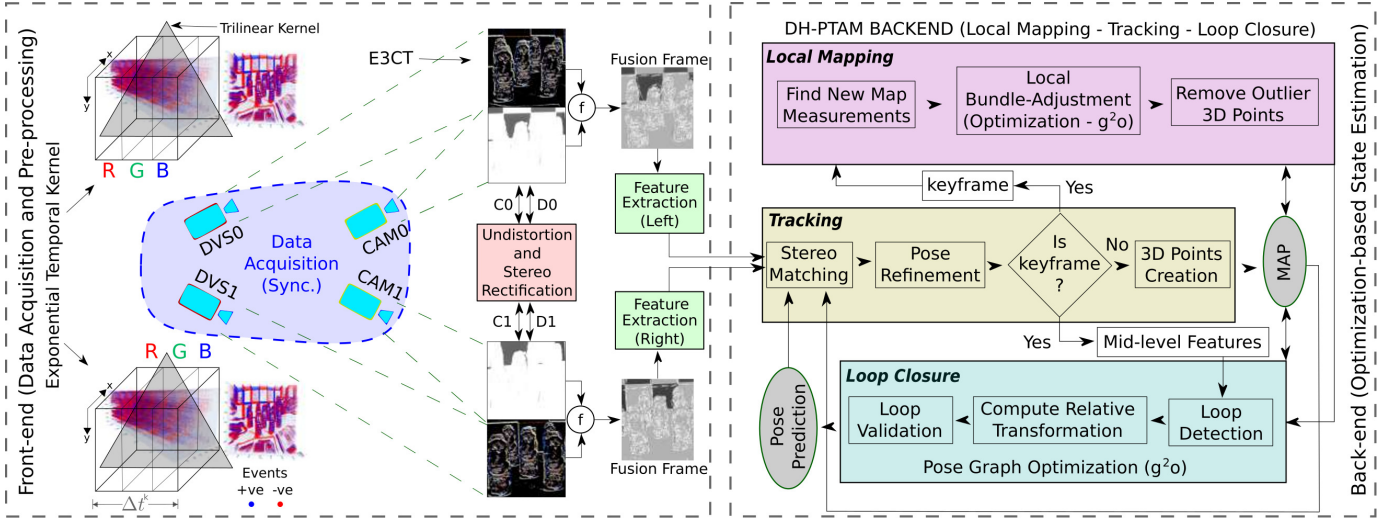


Fig. 2. Block diagram of the proposed hybrid event-aided stereo visual odometry approach (DH-PTAM). f denotes the fusion function defined in (6). Δt^k is the event volume $\mathcal{V}_0(x, y, t)$ accumulation time defined in (2). E3CT denotes the Event 3-Channel Tensor [13], an image-like event representation.

TABLE I

DIRECT AND INDIRECT (D/I) VISUAL ODOMETRY METHODS BASED (B) ON EVENTS AND/OR AIDED (A) BY EVENTS

Method	B/A	D/I	EPL ^a	LC ^b	More Information
Kim [19]	B	D	LIR	×	3 EKFs + Image reconst.
Rebecq [14]	B	D	EI	×	Monocular (PTAM)
Zhou [26]	B	D	TS	×	Stereo (PTAM)
Kueng [27]	A	I	×	×	Event-aided Tracking
Rosinol [18]	A	I	MEF	×	Mono + IMU (front-end)
Hidalgo-Carrió [7]	A	D	EGM	×	Monocular Odometry
Proposed	A	D/I	E3CT	✓	Stereo (PTAM) + DL ^c

^a denotes an Event Pre-processing Layer. ^b denotes Loop-Closure capability. ^c denotes the only method incorporating Deep Learning-aided features.

the specificity of high-level features like object recognition. As referenced in [25], deep learning descriptors typify mid-level features. They extract more sophisticated information from raw data than low-level features like pixel values, yet they don't reach the specificity of direct task-related features, for instance, object labels.

III. METHODOLOGY

A. System Overview

Figure 2 illustrates the main components and the process of DH-PTAM (refer to Algo. 1). The system establishes a global reference frame based on the camera position in the initial frame. A preliminary map is created by identifying and triangulating distinctive points in the first stereo pair of images. For subsequent frames, the tracking thread calculates the 6D pose of each stereo frame by minimizing the discrepancy between the projected map points and their matches. The system chooses a subset of keyframes used in another thread to update the map at a slower pace.

Map points are derived from the stereo matches of each keyframe and added to the map. The mapping thread constantly improves the local discrepancy by adjusting all map points and stereo poses using Bundle Adjustment. A pose

Algorithm 1 DH-PTAM Processing Pipeline

- 1: **System Initialization:**
- 2: Initialize global reference frame and preliminary map with the first stereo image pair.
- 3: **Temporal Synchronization:**
- 4: **for** each new left CAM frame **do**
- 5: Synchronize event and frame streams based on the exposure time and event accumulation window.
- 6: **end for**
- 7: **Spatial Hybridization:**
- 8: **for** each synchronized CAM-DVS frame **do**
- 9: Apply depth scale equivalence and process E3CT events.
- 10: Fuse processed events with RGB frames for enhanced depth estimation.
- 11: **end for**
- 12: **Optimization-based State Estimation:**
- 13: **while** system is running **do**
- 14: Refine poses and map points via Bundle Adjustment.
- 15: Apply loop closure detection and pose graph optimization.
- 16: **end while**
- 17: **Loop Closure:**
- 18: **if** loop detected **then**
- 19: Correct accumulated drift through graph optimization.
- 20: **end if**

graph is utilized to preserve the global map's consistency, which is a shared resource among the tracking, mapping, and loop-closing threads. Point correspondences are actively searched between keyframes to strengthen the constraints of the pose graph optimization smoothing process.

Notations. The odometry state representation comprises the 3D points X_w^k and a 7-increment vector $\mu \in \mathfrak{se}(3)$, which is

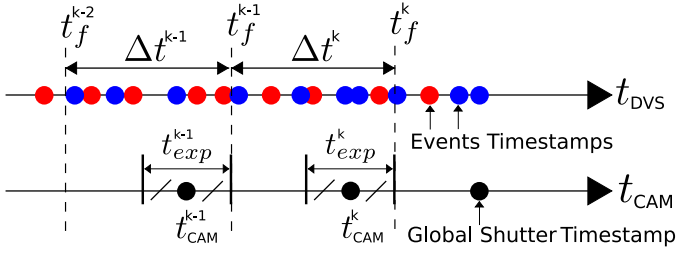


Fig. 3. Temporal synchronization scheme. t_{exp} is the global shutter camera exposure time. Δt is the event representation (E3CT) volume accumulation window. t_f is the fusion frame calculated timestamp. $t_{DVS,CAM}$ are the DVS events, and RGB camera frames timestamps, respectively.

the current pose of the left fusion frame at time k :

$$\mu^k = [\delta x \ \delta y \ \delta z \ \delta q_x \ \delta q_y \ \delta q_z \ \delta q_w]^\top, \quad (1)$$

where $[\delta x \ \delta y \ \delta z]^\top$ is the incremental translation vector and $[\delta q_x \ \delta q_y \ \delta q_z \ \delta q_w]^\top$ is the incremental quaternion vector.

B. Temporal Synchronization Approach

Our temporal synchronization approach (see Fig. 3) considers the general case of global shutter cameras where the exposure time t_{exp} is known. We adopt the constant-time Δt^k events accumulation window k approach where the number of accumulated events during this temporal window is ablated in the qualitative analysis in Fig. 4.

As soon as stereo RGB camera frames are received at timestamps t_{CAM} , we calculate the fusion frames timestamps assuming the hardware synchronization of stereo RGB images and stereo event streams, using:

$$t_f = t_{CAM} + \frac{t_{exp}}{2}, \quad \Delta t^k = t_f^k - t_f^{k-1}, \quad (2)$$

where the left t_{CAM} is the selected stereo keyframe timestamp.

C. Spatial Hybridization Approach

Leveraging the findings of [28], where Scale-Consistent monocular depth learning is explored, our method is firmly

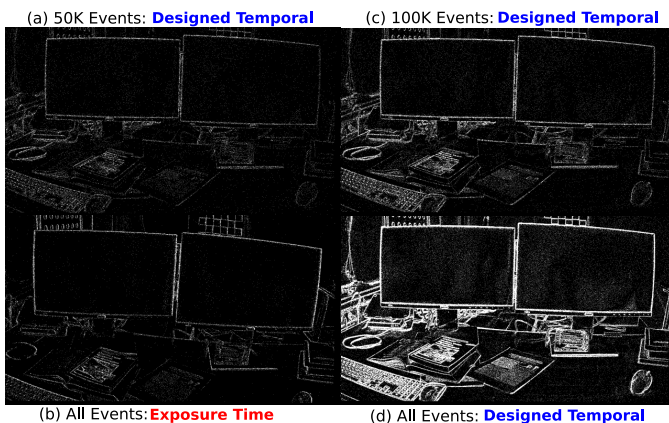


Fig. 4. Visualization for reducing the temporal window width versus controlling the number of events in the designed window. All event frames are post-processed E3CTs by median filtering followed by a binary threshold.

rooted in the depth scale equivalence principle. Rather than just an assumption, this principle is an inference drawn from the inherent properties of closely-spaced APS-DVS sensors. It asserts that the depth scales between frames maintain a consistent relationship. Diverging from the approach in [28], we incorporate the front-end's depth uncertainty into the back-end's stereo bundle adjustment process. Here, it acts as an adaptive mechanism, progressively refining depth estimates, thereby boosting system performance. The E3CT events pre-processing layer is adopted and modeled as two consecutive filtering kernel convolutions on the event volume $\mathcal{V}_0(x, y, t)$ of temporal width Δt^k (see Fig. 2). The first kernel to filter the time decaying events in the volume, is the α -exponential time decay kernel and is modeled as:

$$\mathcal{V}_1(x, y, t) \doteq \exp \left(-\alpha \left(\frac{\mathcal{V}_0(x, y, t) - \eta/2}{\eta/6} \right)^2 \right), \quad (3)$$

where $\alpha = 0.5$ and the decay rate $\eta = 30$ [ms] for our model. Followed by a trilinear voting kernel to stack the events in the three channels tensor so that each event contributes to two consecutive channels depending on their location from a vertex of this trilinear kernel. An event near the top contributes a higher weight to the current channel and a lower weight to the neighboring ones. These contribution weights of the three channels can represent a percentage of an R-G-B color map; hence, the E3CT can be considered a synthetic RGB frame of events. The trilinear voting kernel can be modeled as follows:

$$\mathcal{V}_2(x, y, t_i) \doteq \max \left(0, 1 - \left| \frac{\mathcal{V}_1(x, y, t_i)}{\delta t} \right| \right), \quad (4)$$

where δt is the temporal bin i size as discussed in [16].

After applying the trilinear temporal voting kernel on the exponential-decay time surface, we stack the 3-channel tensor temporal bins together, resulting in a synthetically colored 2D frame called the Event 3-Channel Tensor (E3CT). In Fig. 2, we can observe that the constructed synthetic colors are always consistent, meaning that the stereo left and right constructed E3CTs have identical colors for the same scene.

Conventional frame-based post-processing operations can be applied to the constructed E3CTs, such as adaptive threshold, contrast stretching, color correction and balance, and denoising functions. We consider a fully calibrated stereo RGB and event cameras stack as represented in Fig. 5, so that the rigid-body transformations \mathcal{T}_{cd} and the cameras intrinsic parameters $\mathcal{K}_c, \mathcal{K}_d$ are known.

Given that the same post-processing operations are applied on the current stereo E3CT frames (see Fig. 4), the 2D-to-3D-to-2D consecutive inverse-forward projections of the pixels on the E3CT frames P_d^h to the RGB camera frames $P_{d \in c}^h$ can be performed as follows (comparable to Equation (2) in [28]):

$$P_{d \in c}^h \doteq \frac{1}{\lambda'} \mathcal{K}_c R_{cd} \lambda \mathcal{K}_d^{-1} P_d^h + \delta P_{align}^h, \quad (5)$$

where $(\cdot)^h$ denotes the pixel location in homogeneous coordinates. The term δP_{align}^h denotes the pixel location alignment correction factor for the RGB and event frames so that the same 3D world point X_w^h should correspond exactly to the

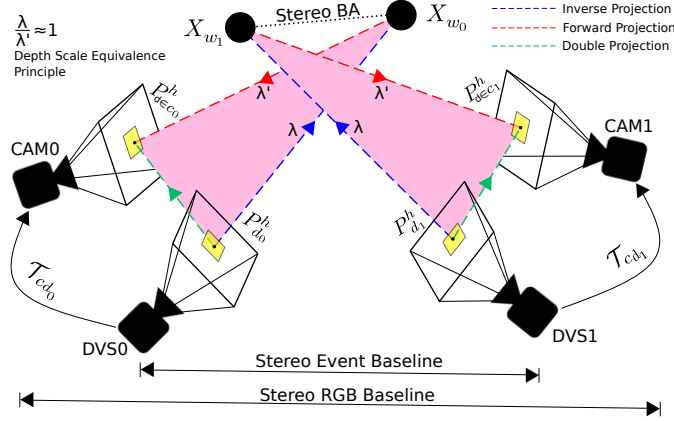


Fig. 5. Geometry of the stereo hybrid event-RGB cameras stack. \mathcal{T}_{cd} denotes the rigid-body transformations. $P_{d \in c}^h$ and P_d^h denote pixels locations.

pixel locations $P_{d \in c}^h$, P_d^h . This alignment term is observed to be constant for the same sensor rig with non-varying intrinsic and extrinsic parameters. δP_{align}^h value can be estimated using an offline optimization process only once on a selected number of frames (the more the accurate) with high confidence feature matches, and this value is given in Section IV for both VECtor and TUM-VIE sequences.

Finally, the fusion function (and frame) $f(\cdot)$ performs a temporal cross-dissolve (linear blending) between both the left (D_0, C_0) and right (D_1, C_1) E3CTs and RGB camera frames, respectively, and is formulated as:

$$f(C, D) = (1 - \beta) * C + \beta * D, \quad (6)$$

where $\beta \in [0, 1]$ is the E3CT contribution weight in the current fusion frame. β value is calculated online and depends on the scene lighting and texture conditions. It is set to chirp-shaped values $\beta = \max(\bar{C}/C^{\max}, 1 - \bar{C}/C^{\max})$ when the RGB camera frame fails to detect features due to adverse HDR conditions and low-textured scenes, and this is the DVS-biased fusion mode. For situations where RGB camera frames can detect reliable scene features with good lighting and enough texture, the β value is harmonic with the scene lighting conditions according to $\beta = \min(\bar{C}/C^{\max}, 1 - \bar{C}/C^{\max})$, and this is the APS-biased fusion mode. To reduce the amount of extracted features and maintain the back-end processing complexity and latency in reasonable ranges, β value can be capped at a certain value, as shown in Fig. 6 by setting $\beta = 0.3$ as an example.

Dynamic scenes with challenging and adverse conditions can easily trigger rapid switching between these two fusion modes during long-term navigation. This causes a critical problem during the feature tracking process using conventional low-level feature detectors, such as ORB, SIFT, SURF, BRIEF, and FAST. Accordingly, applying mid-level feature detectors that depend mainly on learning-based architectures could solve this fusion frame modes alternation problem. We employ the learning-based feature extractors and descriptors [3], [4] for their high robustness and feature detection speed. Fig. 6 elaborates in a real-world case study, the fusion mode

automatic switch using the β parameter in response to the rapid intensity fluctuations of the RGB camera frame C in HDR scenarios represented by the parameter \bar{C}/C^{\max} .

D. Optimization-based State Estimation

As our work is based on the original S-PTAM system, all the optimization Jacobians mentioned in this section can be found with detailed proofs in [2]. All objective functions are minimized with the Levenberg-Marquardt algorithm implemented in the g^2o optimization library. We employ the Huber loss function for outliers rejection $\rho(\cdot)$.

System bootstrapping. The first stereo fusion frames are considered a keyframe. Then, a triangulation for the collected feature matches on the left and right fusion frames is performed to initialize the map.

Pose tracking thread. Each map point is projected into the viewing frustum of the anticipated stereo position, and we then look nearby for the match. A valid prediction of the current pose is required for such a projection. By contrasting the descriptions, map points, and features are matched. The L_2 norm is computed using the binary descriptors of SuperPoint and R2D2 (see Fig. 7). The match is valid if the distance falls below a certain threshold; otherwise, it is ignored. The pose refinement is then applied to recover the current pose knowing the previous one using the following objective function:

$$L^{\text{refine}} = \arg \min_{\mu} \sum_{i \in N} \rho(\|J_i^k \mu_k - \Delta z_i(\mu_{k-1}, X_w^i)\|^2), \quad (7)$$

where $N = \{z_1, \dots, z_M\}$ and M is the number of matched measurements. The measurement $z = [u, v]^T$ is a pixel 2D location of the forward projection of a 3D map point X_w using the pinhole model projection function $\pi(X_w^i) = \mathcal{K}_c \mathcal{T}_{fow}^i X_w^i$. $J_i^k = \frac{\partial \Delta z_i(\mu)}{\partial \mu_k}$ is the re-projection error's Jacobian with respect to the current odometry state vector. Δz is the re-projection error of a matched set of measurements on the current k stereo fusion frames and is defined as:

$$\Delta z_i(\mu, X_w) = z_i - \pi(\exp(\mu) \mathcal{T}_{fow}^{k-1} X_w^i), \quad (8)$$

where the 3D point cloud X_w is considered a constant optimization parameter and not updated in the tracking thread and $\mathcal{T}_{fow}^{k-1} = \exp(\mu) \in SE(3)$ with $\exp(\cdot)$ the exponential map in the $\mathcal{L}ie$ group for the previous increment state vector. If the number of observed points is less than 90% of the points recorded in the previous keyframe, a frame is chosen to be a keyframe after the current pose has been evaluated. Then, new map points are created by triangulating the stereo pair's remaining mismatched features. The keyframe is then placed in the local mapping thread for processing.

Mapping thread. We apply Bundle Adjustment (BA) to fine-tune the camera poses (keyframe map) and the 3D points (point cloud map). Local Bundle Adjustment minimizes the re-projection error of every point in every keyframe f_0^k . Given an initial set of N keyframe poses $\{\mathcal{T}_{fow}^1, \dots, \mathcal{T}_{fow}^N\}$, an initial set of M 3D points X_w^i , and measurement sets $S \in \{S_1, \dots, S_N\}$, where each set comprises the measurement z_i^k of the i^{th} point in the k^{th} keyframe, the local BA is performed

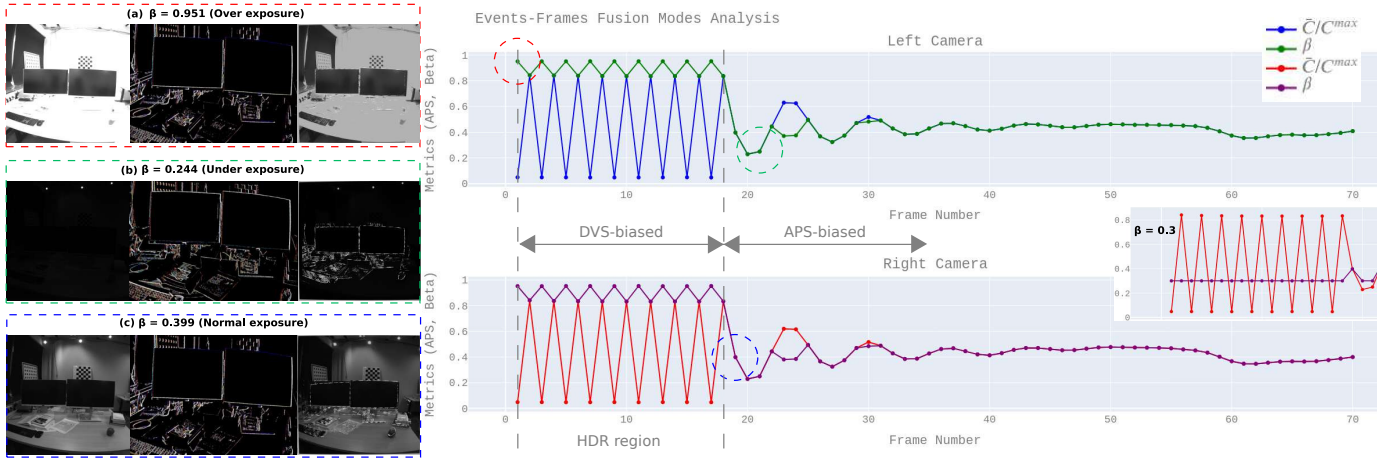


Fig. 6. Experiments on mocap-desk2 TUM-VIE dataset that show the capability of the proposed events-frames fusion method to maintain and track features in bright and dimmed scenes where grayscale-only frames may fail. For each scenario (left to right): grayscale frame, E3CT-color frame and fusion frame.

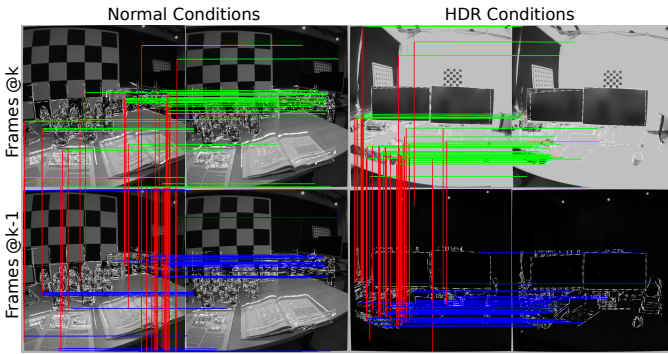


Fig. 7. Normal versus HDR semi-circular spatio-temporal matching for two stereo consecutive fusion frames. Green and blue lines denote the spatial stereo SuperPoints matching at frames k and $k-1$, respectively. Red lines denote the temporal matching for keypoints of two consecutive keyframes (left).

using the following objective function on all keyframes in a pre-defined sliding-window size N :

$$L^{\text{BA}} = \arg \min_{\mu, X_w} \sum_{k=1}^N \sum_{i \in S_k} \rho \left(\left\| J_i^k \begin{bmatrix} \mu_k \\ X_w^i \end{bmatrix} - \Delta z_i(\mu_k, X_w^i) \right\|^2 \right), \quad (9)$$

where the 3D point cloud X_w is considered a variable optimization parameter and is updated in the mapping thread. Hence, the $J_i^k = \left[\frac{\partial \Delta z_i(\mu_k, X_w^i)}{\partial \mu_k}, \frac{\partial \Delta z_i(\mu_k, X_w^i)}{\partial X_w^i} \right]$ is the re-projection error's Jacobian with respect to the current odometry state vector and the 3D point as well.

Loop-closure thread. Instead of the conventional way of keyframe embedding assignments using a bag-of-words, we adopt a simple loop-closure detection method based on the mean of the mid-level learning-based feature descriptors (SuperPoint and R2D2) for each keyframe and assign this mean value as the embedding identity of each keyframe. Once a potential loop closure is detected, the system performs geometric verification through RANSAC-based pose estimation to validate the candidate. If the verification is successful, a loop closure constraint is added to the pose graph, and graph

optimization is performed to distribute the error and update the global map, thus correcting the accumulated drift.

IV. EVALUATION

We perform a thorough, comprehensive evaluation during navigation in real-world, large-scale, and small-scale areas in challenging settings. In subsection IV-A, we compare DH-PTAM with other RGB image-based and event-based/-aided methods on the HDR large-scale sequences of the publicly available dataset VECtor [11] due to its high-quality ground truth values and sensors calibration parameters. In subsection IV-B, we evaluate the small-scale (mocap-) sequences of TUM-VIE [12] to test the quality of the DH-PTAM spatio-temporal synchronization method with degraded event camera calibration parameters. Moreover, the first 45 frames of TUM-VIE sequences suffer a high over-/under-exposure global shutter alternation, which tests the DH-PTAM's pose estimation stability. The accuracy of DH-PTAM is measured with absolute trajectory error (ATE), and relative pose error (RPE) metrics calculated using the baseline SLAM evaluation tool [29]. We perform a comparative quantitative analysis to evaluate the accuracy of our system in Table II and a qualitative analysis in Fig. 8.

Figure 8 exemplifies the efficacy of DH-PTAM in recognizing loop closures, which are crucial for correcting cumulative errors in SLAM systems. By integrating event data, DH-PTAM exhibits superior performance in identifying loop closures, particularly in dynamic and complex environments. The fusion of event data allows for enhanced perception in high dynamic range and fast motion scenes, where traditional frame-based systems falter. This thorough analysis focuses on quantifying the improvement in loop closure detection rates and the reduction of trajectory drift, underscoring the value of event data in augmenting spatial awareness and mapping precision in expansive settings.

The integration of event data on CPUs not only mitigates computational constraints but also elevates feature detection fidelity and tracking accuracy, showcasing the system's adaptability to diverse computational platforms. An in-depth

TABLE II

DH-PTAM QUANTITATIVE COMPARISON AGAINST THE BEST PERFORMING OPEN-SOURCE STATE-OF-THE-ART SLAM SYSTEMS USING ATE [M] METRIC. THE UPPER SUB-TABLE IS FOR STANDARD STEREO VIO METHODS, THE MIDDLE IS FOR EVENT-BASED VO/VIO METHODS, AND THE LOWER IS FOR DH-PTAM WITH RPE [M] METRIC. **BOLD** DENOTES BEST PERFORMING, UNDERLINE FOR SECOND BEST PERFORMING, AND (×) DENOTES FAILURE

Method	VECTor sequences [11]						TUM-VIE sequences [12]					Mean VECTor large-scale	Mean TUM-VIE small-scale
	corridors dolly	corridors walk	units dolly	units scooter	school dolly	school scooter	mocap 1d-trans	mocap 3d-trans	mocap 6dof	mocap desk	mocap desk2		
ORB-SLAM3 (SVIO) [21]	0.802	<u>1.031</u>	18.063	14.504	0.921	<u>0.752</u>	0.007	0.012	0.018	0.007	0.025	6.012	<u>0.013</u>
BASALT (SVIO) [30]	1.625	2.152	11.151	13.256	1.852	1.482	0.003	0.009	0.014	0.016	<u>0.011</u>	5.253	0.011
VINS-Fusion (SVIO) [31]	1.464	0.392	10.391	11.471	1.791	0.562	0.011	0.011	<u>0.017</u>	0.058	0.013	4.345	0.022
EVO (Mono Events) [14]	×	×	×	×	×	×	0.075	0.125	0.855	0.541	0.752	×	0.470
ESVO (Stereo Events) [26]	×	×	×	×	13.710	9.830	0.009	0.028	0.058	0.033	0.032	11.77	0.032
Ultimate SLAM (EVIO) ⁺ [18]	×	×	×	×	×	6.830	0.039	0.047	0.353	0.195	0.341	6.830	0.195
DH-PTAM (Stereo Fusion)	1.884	1.299	5.274	8.433	<u>1.093</u>	0.796	0.103	<u>0.007</u>	0.024	0.016	0.015	<u>3.130</u>	0.033
(SuperPoint on CPU) - RPE (σ)	0.073	0.038	0.055	0.149	0.178	0.074	0.006	0.007	0.009	0.009	0.007	0.095	0.008
DH-PTAM* (Stereo Image)	1.841	1.543	<u>5.738</u>	5.010	1.559	0.877	0.099	0.004	0.045	<u>0.011</u>	0.008	2.761	0.033
(R2D2 on GPU) - RPE (σ)	0.116	0.141	0.134	0.308	0.202	0.331	0.014	0.020	0.022	0.023	0.021	0.205	0.020

⁺ IMU sensor is included since it is integrated into the front-end and cannot be separated for a fair comparison with EVO, ESVO, and DH-PTAM (ours).
^{*} in this ablation case study, the SuperPoint detector is replaced with the R2D2 detector (trained for SLAM tasks), leveraging the GPU performance.

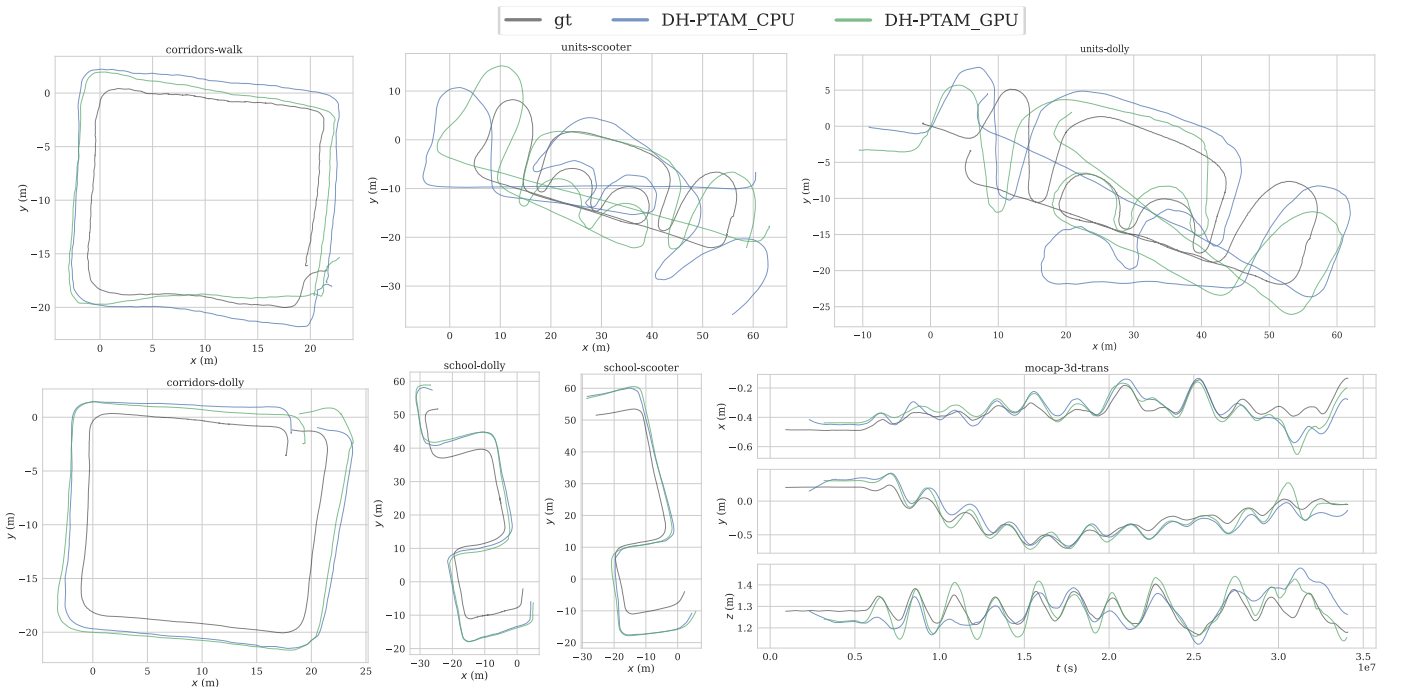


Fig. 8. DH-PTAM (GPU (no events) vs. CPU (event-aided)) qualitative analysis. All trajectories are transformed to a reference frame as the ground truth poses using the extrinsic parameters, followed by an alignment with all poses by Umeyama's SE(3) method implemented by [29]. Large-scale trajectories show high-quality loop closure detection in the case of R2D2 on GPU. Small-scale trajectories show the high accuracy of the event-aided version of DH-PTAM.

discussion explores how the event-driven approach enhances computational efficiency, enabling real-time processing even on less powerful CPUs. Furthermore, this analysis highlights the pivotal role of event data in sustaining system stability and accuracy across various lighting conditions, illustrating the dual benefits of improved performance and computational versatility offered by DH-PTAM.

To prevail the advantages of complementing the sensor stack with events information, we compare our event-aided stereo visual odometry solution (DH-PTAM) to the latest best-performing open-source visual-inertial systems in literature in Table II. Table III gives the system parameters configuration for large-scale and small-scale sequences. We keep these parameters constant for all sequences of the same scale group without an online fine-tuning process.

TABLE III
DH-PTAM PARAMETERS CONFIGURATION

Parameter	VECTor	TUM-VIE	MVSEC
$\delta P_{align}^h(u, v)$ - Left [px]	(-160, -235)	(355, 40)	(0, 0)
$\delta P_{align}^h(u, v)$ - Right [px]	(-160, -235)	(375, 45)	(-20, 0)
frustum_near [m]	0.1	0.1	0.1
frustum_far [m]	30.0	5.0	5.0
matching_cell_size [px]	15	15	15
matching_neighborhood [px]	2	1.8	1.8
matching_distance [px]	25	15	15

A. VECTor large-scale experiments

We notice a prominent estimation failure in Table II while evaluating the event-based methods EVO, ESVO and Ulti-

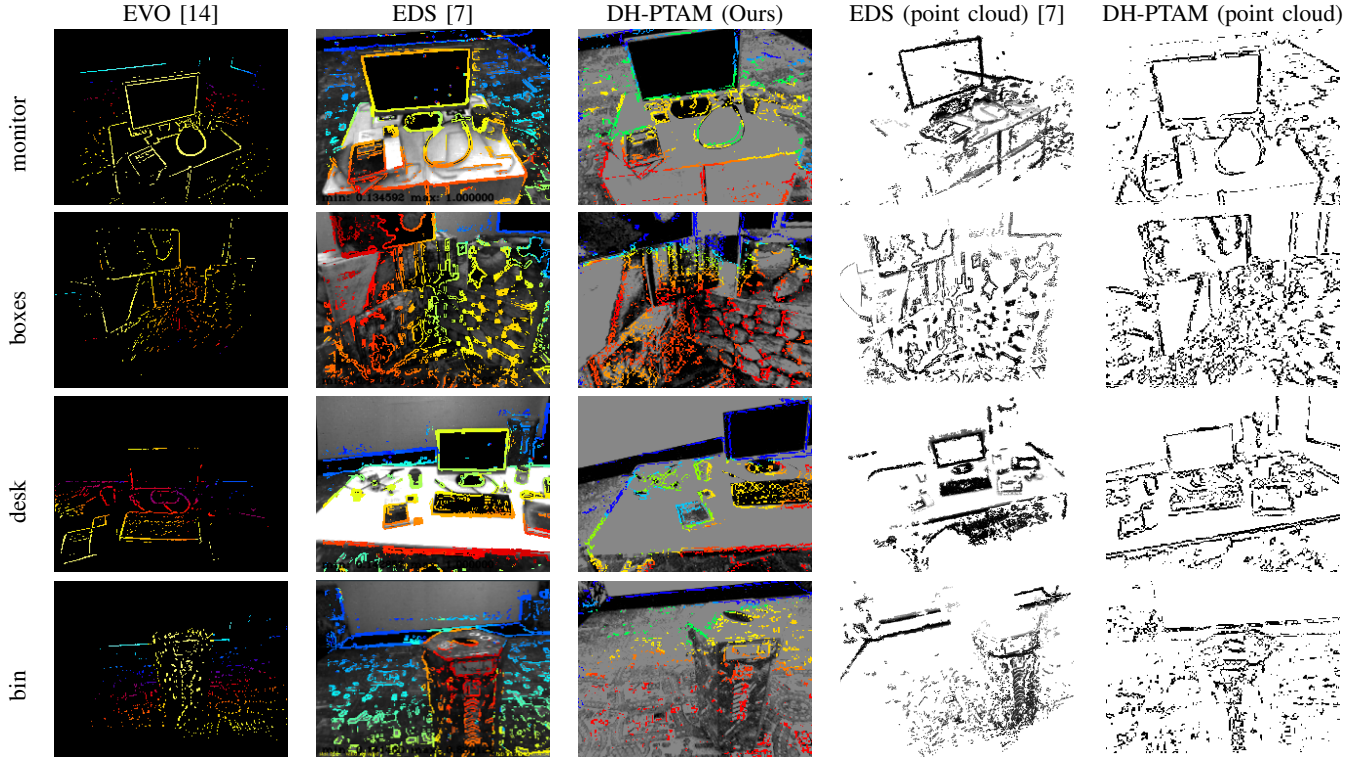


Fig. 9. Stereo DAVIS sequences evaluation from [32] with $\delta P_{align}^{ph} = [0, 0]^T$ since both sensors (event-based and frame-based) share the same photoreceptor pixel array. Initial three columns present pseudo-colored inverse depth maps: EVO (yellow-near, blue-far), EDS and DH-PTAM (red-near, blue-far). All sequences span a 1-7 m depth range. 3D point clouds from EDS and DH-PTAM, utilizing grayscale keyframe values, are presented in the last two columns.

mate SLAM on the large-scale sequences. Numerous factors may contribute to the failure of these systems, including stringent initialization requirements. For instance, the system EVO necessitates running in a sensor-planar scene for several seconds to bootstrap the system. Additionally, these systems are susceptible to parameter tuning, as demonstrated by using different parameters for different sequences in the same scenarios, even within their open-source projects.

Table II shows a good performance for DH-PTAM compared to the competing non-event-based VI-SLAM systems. Although Fig. 8 shows high visual drifts for our vision-only system in the case of units sequences, DH-PTAM could outperform the VI-SLAM systems based on the ATE metric. Fig. 8 gives an overview of the high-quality loop detection of DH-PTAM in the case of corridor sequences. Loop detection failure can be noticed only when the RAM overflows while running the system with enormous point clouds, as in the case of unit sequences. We provide trajectory smoothing and post-processing script with our open-source implementation to join estimated trajectory increments in case of RAM overflow failures.

B. TUM-VIE small-scale experiments

As noticed in [33], the calibrationA (mocap-desk, mocap-desk2) sequences have more accurate depth estimation results than calibrationB (rest of mocap and TUM-VIE large-scale) sequences due to the significant calibration errors in the latter. Hence, we perform our comparative evaluation on TUM-VIE small-scale (mocap-) sequences using calibrationA parameters.

Although the same high-quality calibrationA parameters apply to both desk2 and desk sequences with the same spiral motion, DH-PTAM performs the best with desk2 sequence but the worst with desk sequence. This occurs since the scene of the desk sequence is bounded by a close-by white wall that strict the depth, and hence DH-PTAM front-end detects low quality and fewer features for desk than desk2. Table II shows that the more DoF excited (6dof, desk2) and the consistent loops detection (1d-trans), the better the pose estimation quality.

C. Ablation experiments

No event streams ($\beta = 0$). In Table II, we show an ablation study where we run DH-PTAM on stereo images. We notice estimation failure with all the conventional and learning-based feature detectors except R2D2. Although the ATE metric shows slightly better results without using events, the RPE metric shows much more accurate values when using events. These better ATE values are due to the high performance of the GPU in loop-closures using R2D2 features (see Fig. 8).

Front-end depth maps. Qualitatively, in Fig. 9, we rigorously assess the front-end of our events-frames fusion technique by comparing its inverse depth maps against established methods such as EVO and EDS. By channeling the stereo fusion frames through the SGM matching method [34], our method capitalizes on direct front-end fusion where the inverse depth maps are derived from events projected onto the RGB frames. Remarkably, the point clouds generated through our method exhibited richer details compared to those of EDS.

TABLE IV
DH-PTAM MPE [%] COMPARISON WITH OTHER IMU-AIDED AND DL-BASED ODOMETRY METHODS ON MVSEC DATASET [35]

Method	IndoorFlying1	IndoorFlying2	IndoorFlying3	IndoorFlying4
PL-EVIO [36]	0.36	0.30	0.34	0.44
ESVIO [37]	0.25	0.30	0.25	0.46
DEVO [38]	0.26	0.32	0.19	1.08
DH-PTAM (Ours)	0.28	1.42	0.97	0.26

This affirms the potential of our method to set a new benchmark in event-frame fusion paradigms.

IMU-aided versus deep learning-aided back-end. Table IV presents an ablation study comparing the Mean Positional Error (MPE [%]) of DH-PTAM with other IMU-aided and Deep Learning-based visual odometry methods on the MVSEC dataset. The methods compared include PL-EVIO, ESVIO, and DEVO across four indoor flying sequences.

This ablation study highlights the strengths and limitations of DH-PTAM compared to other state-of-the-art methods. While DH-PTAM shows promising results in specific sequences like IndoorFlying4 (comprising slower 6-DoF excitations), it struggles in others, particularly IndoorFlying2 and IndoorFlying3. These variations suggest that further tuning of DH-PTAM’s parameters for rapid motions and additional training for SuperPoint extractor might be necessary to enhance its overall robustness and accuracy across dynamic environments.

D. GPU vs. CPU Performance Analysis

All experiments are performed on the CPU and the GPU of a 16 GB RAM laptop computer running 64-bit Ubuntu 20.04.3 LTS with AMD(R) Ryzen 7 4800h ×16 cores 2.9 GHz processor and a Radeon RTX NV166 Renoir graphics card. Table V reports a detailed computational complexity analysis for our DH-PTAM system with minimal and maximal system requirements. The high CPU load observed when detecting SuperPoint and R2D2 features can be attributed to the algorithms’ design, which prioritizes feature quality and robustness over computational efficiency. This trade-off is often necessary for computer vision research, where high-quality results are crucial for many applications but come at the cost of increased computational complexity. The back-end runs with real-time performance, and it is recommended to run the front-end on a GPU to achieve a memory-efficient, faster, and more stable performance.

V. CONCLUSION

In this paper, we presented the DH-PTAM system for robust parallel tracking and mapping in dynamic environments using stereo images and event streams. The proposed system builds upon the principles of S-PTAM and extends it with a learning-based approach to handle the sparse and noisy nature of event-based sensors while leveraging the rich information provided by fusion frames. Our experiments demonstrate that DH-PTAM outperforms state-of-the-art visual-SLAM methods, particularly in challenging scenarios such as HDR and occlusions. The proposed system can achieve better performance

TABLE V
COMPUTATIONAL COMPLEXITY ANALYSIS ON CPU vs. GPU

Thread	#Tasks	Operation	CPU [ms]	GPU [ms]
Front-end	3	Stereo E3CT Construction	25-30	76-172
		Events-Frames Fusion	439-521	191-352
		SuperPoint Detection	2478-3256	521-1752
Bootstrapping	1	($\approx 2 \times 10K$) R2D2 Detection	8532-8752	1067-2254
		($\approx 2 \times 4K$) Initialize the Map	106-143	53-120
		Tracking	161-215	142-172
Mapping	2	(10 iter.) Spatio-temporal Matching	11-15	10-12
		Pose Refinement	1-3	0.452-1
Loop-closing	3	(30 iter.) Local Bundle-Adjustment	1-4	1-2
		Loop Detection	14-20	10-15
End-to-End	11	Compute and Validate	2-5	1-3
		(30 iter.) Pose Graph Optimization	1-3	0.524-2
End-to-End	11	SuperPoint Detector	3153-4208	356-1962
		R2D2 Detector	8560-9125	1226-2486

on a GPU and provides a scalable and accurate solution for 3D reconstruction and pose estimation. Future work includes investigating the potential of improving spatial synchronization through online optimization and exploring the integration of an adaptive paradigm for choosing the temporal window width to further boost robustness and system latency. DH-PTAM proves to have the potential to provide robust and accurate 3D mapping and localization, which are crucial for the successful operation of long-term navigation systems.

REFERENCES

- [1] G. Klein and D. Murray, “Parallel tracking and mapping for small ar workspaces,” in *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, 2007, pp. 225–234.
- [2] T. Pire, T. Fischer, G. Castro, P. De Cristóforis, J. Civera, and J. J. Berllés, “S-ptam: Stereo parallel tracking and mapping,” *Robotics and Autonomous Systems*, vol. 93, pp. 27–42, 2017.
- [3] D. DeTone, T. Malisiewicz, and A. Rabinovich, “Superpoint: Self-supervised interest point detection and description,” in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 224–236.
- [4] J. Revaud, C. De Souza, M. Humenberger, and P. Weinzaepfel, “R2d2: Reliable and repeatable detector and descriptor,” *Advances in neural information processing systems*, vol. 32, 2019.
- [5] Z. Teed and J. Deng, “Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 16 558–16 569, 2021.
- [6] Z. Zhang, C. Forster, and D. Scaramuzza, “Active exposure control for robust visual odometry in hdr environments,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3894–3901.
- [7] J. Hidalgo-Carrió, G. Gallego, and D. Scaramuzza, “Event-aided direct sparse odometry,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5781–5790.
- [8] S. Sun, G. Cioffi, C. De Visser, and D. Scaramuzza, “Autonomous quadrotor flight despite rotor failure with onboard vision sensors: Frames vs. events,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 580–587, 2021.
- [9] P. Chen, W. Guan, F. Huang, Y. Zhong, W. Wen, L.-T. Hsu, and P. Lu, “Ecmd: An event-centric multisensory driving dataset for slam,” *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 407–416, 2024.
- [10] J. Huang, S. Zhao, T. Zhang, and L. Zhang, “Mc-veo: A visual-event odometry with accurate 6-dof motion compensation,” *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 1756–1767, 2024.
- [11] L. Gao, Y. Liang, J. Yang, S. Wu, C. Wang, J. Chen, and L. Kneip, “VEctor: A versatile event-centric benchmark for multi-sensor slam,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8217–8224, 2022.
- [12] S. Klenk, J. Chui, N. Demmel, and D. Cremers, “Tum-vie: The tum stereo visual-inertial event dataset,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 8601–8608.

- [13] A. Soliman, F. Bonardi, D. Sidibé, and S. Bouchafa, "IBISCape: A simulated benchmark for multi-modal SLAM systems evaluation in large-scale dynamic environments," *Journal of Intelligent & Robotic Systems*, vol. 106, no. 3, p. 53, Oct 2022.
- [14] H. Rebecq, T. Horstschäfer, G. Gallego, and D. Scaramuzza, "Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 593–600, 2016.
- [15] A. Sironi, M. Brambilla, N. Bourdis, X. Lagorce, and R. Benosman, "Hats: Histograms of averaged time surfaces for robust event-based object classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1731–1740.
- [16] D. Gehrig, A. Loquercio, K. G. Derpanis, and D. Scaramuzza, "End-to-end learning of representations for asynchronous event-based data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 5633–5643.
- [17] D. Gehrig, H. Rebecq, G. Gallego, and D. Scaramuzza, "Ekl: Asynchronous photometric feature tracking using events and frames," *International Journal of Computer Vision*, vol. 128, no. 3, pp. 601–618, 2020.
- [18] A. R. Vidal, H. Rebecq, T. Horstschäfer, and D. Scaramuzza, "Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 994–1001, 2018.
- [19] H. Kim, S. Leutenegger, and A. J. Davison, "Real-time 3d reconstruction and 6-dof tracking with an event camera," in *European conference on computer vision*. Springer, 2016, pp. 349–364.
- [20] P. R. G. Cadena, Y. Qian, C. Wang, and M. Yang, "Spade-e2vid: Spatially-adaptive denormalization for event-based video reconstruction," *IEEE Transactions on Image Processing*, vol. 30, pp. 2488–2500, 2021.
- [21] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "OrbSLAM3: An accurate open-source library for visual, visual-inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.
- [22] J. Niu, S. Zhong, and Y. Zhou, "Imu-aided event-based stereo visual odometry," *arXiv preprint arXiv:2405.04071*, 2024.
- [23] P. Koniusz, F. Yan, and K. Mikolajczyk, "Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection," *Computer vision and image understanding*, vol. 117, no. 5, pp. 479–492, 2013.
- [24] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, October 2012.
- [25] Z. Ji, F. Wang, X. Gao, L. Xu, and X. Hu, "Ssnet: Learning mid-level image representation using salient superpixel network," *Applied Sciences*, vol. 10, no. 1, 2020.
- [26] Y. Zhou, G. Gallego, and S. Shen, "Event-based stereo visual odometry," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1433–1450, 2021.
- [27] B. Kueng, E. Mueggler, G. Gallego, and D. Scaramuzza, "Low-latency visual odometry using event-based feature tracks," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 16–23.
- [28] L. Wang, Y. Wang, L. Wang, Y. Zhan, Y. Wang, and H. Lu, "Can scale-consistent monocular depth be learned in a self-supervised scale-invariant manner?" in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 12 707–12 716.
- [29] M. Grupp, "evo: Python package for the evaluation of odometry and slam," Note: <https://github.com/MichaelGrupp/evo>, 2017.
- [30] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers, "Visual-inertial mapping with non-linear factor recovery," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 422–429, 2019.
- [31] T. Qin, J. Pan, S. Cao, and S. Shen, "A general optimization-based framework for local odometry estimation with multiple sensors," *arXiv preprint arXiv:1901.03638*, 2019.
- [32] Y. Zhou, G. Gallego, H. Rebecq, L. Kneip, H. Li, and D. Scaramuzza, "Semi-dense 3d reconstruction with a stereo event camera," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 235–251.
- [33] S. Ghosh and G. Gallego, "Multi-event-camera depth estimation and outlier rejection by refocused events fusion," *Advanced Intelligent Systems*, vol. 4, no. 12, p. 2200221, 2022.
- [34] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.
- [35] A. Z. Zhu, D. Thakur, T. Özslan, B. Pfrommer, V. Kumar, and K. Daniilidis, "The multivehicle stereo event camera dataset: An event

camera dataset for 3d perception," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2032–2039, 2018.

- [36] W. Guan, P. Chen, Y. Xie, and P. Lu, "PI-evio: Robust monocular event-based visual inertial odometry with point and line features," *IEEE Transactions on Automation Science and Engineering*, 2023.
- [37] P. Chen, W. Guan, and P. Lu, "Esvio: Event-based stereo visual inertial odometry," *IEEE Robotics and Automation Letters*, 2023.
- [38] S. Klenk, M. Motzet, L. Koestler, and D. Cremers, "Deep event visual odometry," *arXiv preprint arXiv:2312.09800*, 2023.



Abanob Soliman received his Ph.D. in Signal and Image Processing from Université Paris-Saclay in 2023, specializing in hybrid visual-inertial odometry, numerical optimization, and neuromorphic vision. His doctoral research was conducted at the IBISC laboratory under the guidance of Prof. Samia Bouchafa, Prof. Désiré Sidibé, and Prof. Fabien Bonardi. After completing his Ph.D., he joined Continental Engineering Services Limited in the United Kingdom as an ADAS Algorithm Engineer in 2023. He holds an M.Sc. in Smart Aerospace and Autonomous Systems from Université Paris-Saclay (2020) and a B.Sc. (Hons.) in Aerospace Engineering from Cairo University (2018). Before his graduate studies, he was a Teaching Assistant in the Electronics and Computer Engineering Department at Nile University in 2019.



Fabien Bonardi received in 2017 the Ph.D. degree in electrical and computer engineering from Normandie University, UNIROUEN (LITIS Lab). He is assistant professor at Université d'Evry Paris-Saclay since 2018. His research concerns computer vision, visual perception in robotics and sensors fusion for non-conventional cameras.



Désiré Sidibé (Senior Member, IEEE) graduated from Ecole Centrale de Nantes (MSc Eng., 2004) and from University of Montpellier (PhD, 2007). He is currently a Full Professor at Université Evry - Paris Saclay, and a member of the IBISC laboratory. His research interests include computer vision for autonomous vehicles and medical image analysis. He is an IEEE senior member and has authored about 100 papers in international journals and conferences. He serves as an Associate Editor for IEEE Robotics and Automation Letters.



Samia Bouchafa received a Ph.D. degree in electrical and computer engineering from University Paris VI and the French Research Institute in Transportation (IFSTTAR, then University Gustave Eiffel) in 1998. In 1999, she was an assistant professor at University Paris XI (IEF Lab), then an associate professor in 2011 (HDR). Since 2012, she has a full professor position at Université d'Evry Paris-Saclay. Since 2015, she has been coordinating the second year of a Master in Electrical Engineering, specializing in "Automatic mobile systems" at the University of Paris-Saclay. Since 2019, she has been the director of the IBISC laboratory. Since 2023, she has been the director of the Faculty of Science and Technology at the Université d'Evry Paris-Saclay. Her research concerns computer vision, visual perception, multimodal vision for autonomous systems, motion analysis, stereovision, visual odometry, and localization.