Implicit Event-RGBD Neural SLAM

Delin Qu[1,2*] Chi Yan[2*] Dong Wang[2] Jie Yin[3]
Dan Xu[5] Bin Zhao[2,4] Xuelong Li[2,4]
[1]Fudan University [2]Shanghai AI Laboratory [3]Shanghai Jiao Tong University
[4]Northwestern Polytechnical University [5]The Hong Kong University of Science and Technology

Abstract

*Implicit neural SLAM has achieved remarkable progress recently. Nevertheless, existing methods face significant challenges in non-ideal scenarios, such as motion blur or lighting variation, which often leads to issues like convergence failures, localization drifts, and distorted mapping. To address these challenges, we propose **EN-SLAM**, the first event-RGBD implicit neural SLAM framework, which effectively leverages the high rate and high dynamic range advantages of event data for tracking and mapping. Specifically, EN-SLAM proposes a differentiable CRF (Camera Response Function) rendering technique to generate distinct RGB and event camera data via a shared radiance field, which is optimized by learning a unified implicit representation with the captured event and RGBD supervision. Moreover, based on the temporal difference property of events, we propose a temporal aggregating optimization strategy for the event joint tracking and global bundle adjustment, capitalizing on the consecutive difference constraints of events, significantly enhancing tracking accuracy and robustness. Finally, we construct the simulated dataset **DEV-Indoors** and real captured dataset **DEV-Reals** containing 6 scenes, 17 sequences with practical motion blur and lighting changes for evaluations. Experimental results show that our method outperforms the SOTA methods in both tracking ATE and mapping ACC with a real-time* 17 *FPS in various challenging environments. The code and dataset will be released soon.*

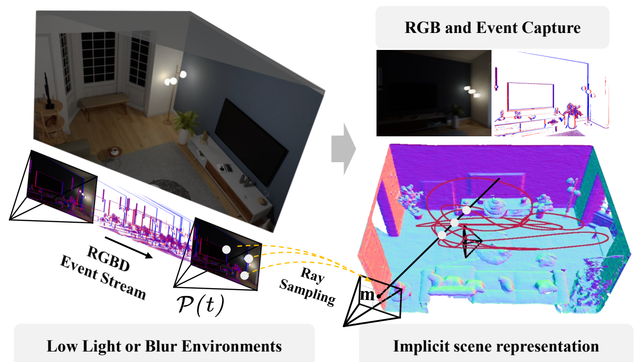


Figure 1. Illustration of the proposed implicit event-RGBD neural SLAM system **EN-SLAM** under non-ideal environments. The dynamic range of RGB sensors is relatively low and suffers from motion blur. Instead, event cameras show great potential in non-ideal environments due to their high dynamic range and low latency advantages. Our method samples rays from two independent RGBD and event cameras to jointly train a single implicit neural field with both modalities. This hybrid shared mechanism provides a natural fusion approach, avoiding alignment issues. It also leverages the advantages of both modalities, resulting in dense, more robust, and higher-quality reconstruction results.

1. Introduction

Simultaneous Localization and Mapping (SLAM) is an essential problem in computer vision and robotics, widely applied in tasks such as virtual and augmented reality [8], robot navigation [18] and autonomous driving [3] over last decades. Exploration in extreme environments [10] remains challenging for visual SLAM (vSLAM) systems due to the lack of visual features caused by motion blur and lighting variation in diverse environments [55, 63].

As a novel representation for myriad of signals, Neural Radiance Fields (NeRF) [33] has innovated great progress in SLAM recently, demonstrating significant improvements in map memory consumption, hole filling, and mapping quality [23, 45, 48, 54, 62, 66]. While the existing NeRF-based neural vSLAM methods address the limitations of traditional SLAM frameworks [11, 34, 35, 46, 58] in accurate dense 3D map reconstruction, they are primarily designed for well-lit scenes and always fail in practical SLAM scenarios with motion blur [38, 61] and lighting variation [30, 44]. These methods produce unsatisfying results under non-ideal environments [54] because of the following limitations: **1) View-inconsistency:** When the camera encounters rapid velocity variation in Fig. 2 (2nd), the scene may exhibit discontinuous blur, leading to view-inconsistency among frames, further causing heavy artifacts in the reconstructed map. **2) Low dynamic range:** In lighting variation scenes illustrated in Fig. 1, the dynamic range of the RGB sensor is relatively low, and the information on the dark and overexposure areas is lost, leading to tracking drifts and mapping distortions.

To address the issues in non-ideal scenarios of existing neural vSLAM, we introduce utilizing the advantages of high dynamic range (HDR) and temporal resolution of event data to compensate for the lost information, thereby improving the robustness, efficiency, and accuracy of current neural vSLAM in extreme environments. Fig. 2 shows the event generation model that an event is triggered at a single pixel if the corresponding logarithmic change in luminance exceeds a threshold C. This asynchronous mechanism shows excellent potential in non-ideal environments due to its advantages in low latency [25, 39, 65], high dynamic range [42], and high temporal resolution [50, 51]. Fig. 1 and Fig. 2 illustrate its superiority in dark and fast motion, and event sensors capture higher-quality signals than RGB sensors. However, applying events into NeRF-based vSLAM is challenging due to the significant distinction in imaging mechanisms between event and RGB cameras. Moreover, the requirement of highly accurate camera poses and careful optimization in traditional surface density estimation [30] further complicates the integration.

In order to overcome these obstacles, we present ***EN-SLAM***, the first event-RGBD implicit neural SLAM framework that effectively harnesses the advantages of event and RGBD streams. The overview of EN-SLAM is shown in Fig. 3. Our method models the differentiable imaging process of two distinct cameras and utilizes shared radiance fields to jointly learn a hybrid unified representation from events and RGBD data. By integrating the event generation model into the optimization process, we introduce the event temporal aggregating (ETA) optimization strategy for event joint tracking and global bundle adjustment (BA). This strategy effectively leverages the temporal difference property of events, providing efficient consecutive difference constraints and significantly improving the performance. Additionally, we construct two datasets: the simulated dataset **DEV-Indoors** and the real captured dataset **DEV-Reals**, which consist of 6 scenes and 17 sequences with practical motion blur and lighting changes. Contributions can be summarized as follows:

- We present **EN-SLAM**, the first event-RGBD implicit neural SLAM framework that efficiently leverages event stream and RGBD to overcome challenges in extreme motion blur and lighting variation scenes.
- A differentiable CRF rendering technique is proposed to map a unified representation in the shared radiance field to RGB and event camera data for addressing the significant distinction between event and RGB. A temporal aggregating optimization strategy that capitalizes the consecutive difference constraints of the event stream is present and significantly improves the camera tracking accuracy and robustness.
- We construct a simulated **DEV-Indoors** and real captured **DEV-Reals** dataset containing 17 sequences with practical motion blur and lighting changes. A wide range of evaluations demonstrate competitive real-time performance under various challenging environments.

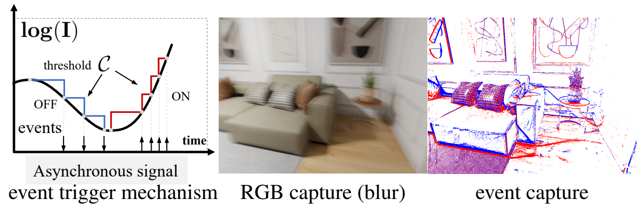


Figure 2. **Illustration of the Event Generation Model (EGM).** An event is triggered at a single pixel if the corresponding logarithmic change in luminance exceeds a threshold C.

2. Related Work

Neural Implicit vSLAM. Existing NeRF-based visual SLAM methods have made significant improvements in dense map reconstruction. iMAP [48] first introduces neural implicit mapping into SLAM, compactly representing the scene with an MLP. NICE-SLAM [66] expands the reconstructable environment size by introducing multi-scale feature grids, allowing for larger-scale environmental reconstruction. Vox-Fusion [59] utilizes an octree-based structure to expand the scene dynamically, enabling dynamic map partition. Besides, CoSLAM [54] combines coordinate and sparse parametric encoding to achieve fast convergence and surface hole filling in reconstruction. The parallel works ESLAM [23] and Point-SLAM [45] represent the scene as a set of multi-scale feature planes and neural point cloud, respectively, to improve the efficiency and accuracy of tracking and mapping. However, these methods are designed for well-lit indoor scenes and commonly encounter challenges in non-ideal SLAM processes, such as motion blur and lighting variation. In contrast, we introduce utilizing the advantages of the high dynamic range and temporal resolution of events to compensate for the lost information, thereby improving the robustness, efficiency, and accuracy of current neural implicit methods.

Event-based SLAM. Events have been incorporated into several traditional visual SLAM systems to address the motion blur and lighting variation. These methods can be divided into three main types: *feature-based methods*, *direct methods*, and *motion-compensation methods*. Feature-based methods, such as USLAM [53], EIO [16] and PL-EVIO [17], extract and track point or line features from event data [28, 40] and perform the camera tracking and mapping in parallel threads. However, the feature extraction algorithms [2, 31, 43, 52] rely heavily on frame-based feature detection, facing challenges for motion-dependent event data appearance. Instead of extracting features from

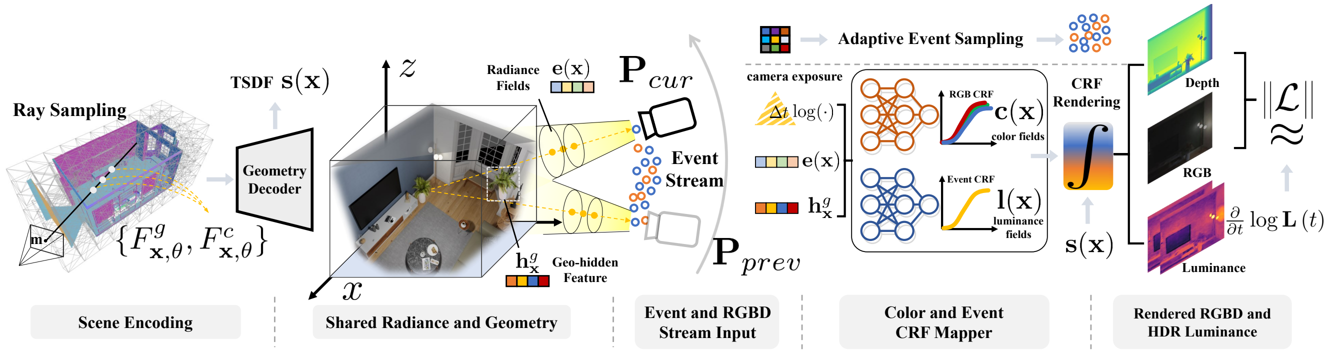


Figure 3. **Overview of EN-SLAM**. EN-SLAM decodes the scene encoding to a shared geometry and radiance representation, and decomposes the radiance into RGB color $\mathbf{c}(\mathbf{x})$ and event luminance $\mathbf{l}(\mathbf{x})$ via differentiable CRF Mappers. We iteratively optimize the pose and scene representation by minimizing losses, in tracking and global BA with the event temporal aggregating techniques in Algorithm 1.

events, direct methods employ events without explicit data association by aligning the photometric event image [19, 24, 25] or utilizing the spatiotemporal information for event representations alignment [39, 41, 65]. Direct methods are well suited for event data, but they mainly focus on event-based visual odometry, leaving the visual dense mapping unexplored. Motion-compensation methods optimize the event alignment in motion-compensated event frames by maximizing the contrast [37, 56], minimizing the dispersion [36] and align probabilistic [15]. However, they suffer from the collapse in a broad range of camera motions. Thus, currently, the event-based SLAM demonstrates significant potential but lacks sufficient exploration in dense map reconstructions [20].

Neural Radiance Fields using Events. Event-based NeRF is in the nascent stages, and several studies have demonstrated the possibility of view synthesis from events via implicit neural fields. Event-NeRF [44] proposes an approach for inferring NeRF from a monocular color event stream that enables novel view synthesis. E-NeRF [26] and E^{2}NeRF [38] tackle the NeRF estimation from event cameras under strong motion blur. They develop normalized and rendering loss to address varying contrast thresholds and enhance neural volumetric representation. The parallel work Ev-NeRF [22] conducts a threshold-bound loss with the ReLU function to address the lack of RGB images. In addition to reconstruction, Δ_tNeRF [32] proposes an event camera tracker by minimizing the error between sparse events and the temporal gradient of the scene representation on the simplified intensity-change events. However, the traditional surface density estimation in NeRF requires highly accurate camera poses and careful optimization [30], thus making it exceptionally challenging to apply NeRF to the event-based SLAM.

3. Methodology

The overview of our method is shown in Fig. 3. Given an input RGBD stream $\{\mathbf{I}_i, \mathbf{D}_i\}_{i=1}^J$ and event stream $\{\mathbf{E}_k\}_{k=1}^N$ with known camera intrinsics $\mathbf{K} \in \mathbb{R}^{3\times3}$ and $\mathbf{K}' \in \mathbb{R}^{3\times3}$, we aim to leverage event and RGBD to reconstruct the camera poses $\{\mathbf{P}_i\}_{i=1}^J$ and the implicit scene representation. In Sec. 3.1, the scene encoding is decoded to a unified geometry and radiance representation. Then, the shared radiance is decomposed into RGB color $\mathbf{c}(\mathbf{x})$ and event luminance $\mathbf{l}(\mathbf{x})$ via differentiable CRF Mappers in Secs. 3.2 and 3.3 to address the imaging distinction of event and RGB cameras. Finally, EN-SLAM iteratively optimizes the pose and scene representation by minimizing the re-rendering loss between the observed RGBD-E (RGBD and events) and rendering results in tracking and global BA of Sec. 3.4.

3.1. Unified Implicit Scene Representation

As shown in Fig. 3, we represent the scene $\mathbf{S}$ with multi-resolution geometric features and color grid features:

$$\mathbf{S} = \{(F^g_{\mathbf{x},\theta}, F^c_{\mathbf{x},\theta}) \,|\, \theta = 1, ..., \Theta\} \ , \tag{1}$$

where $\mathbf{x}$ and θ denote the coordinate and resolution level. There are two challenges that hinder us from learning a scene representation from different RGB and event modalities. Firstly, the event data is sparse and records logarithmic changes in luminance. Secondly, different cameras hold distinct physical imaging process mechanisms. Despite this, the geometry and radiance fields remain consistent during the camera imaging. In this case, we propose to learn a shared unified geometry hidden feature $\mathbf{h}^g_{\mathbf{x}}$ and radiance representation $\mathbf{e}(\mathbf{x})$ across distinct cameras. The geometric grid feature $F^g_{\mathbf{x},\theta}$ and color feature $F^c_{\mathbf{x},\theta}$ are simultaneously mapped to geometry hidden vector $\mathbf{h}^g_{\mathbf{x}}$, radiance fields $\mathbf{e}(\mathbf{x})$ and TSDF (truncated signed distance function) $\mathbf{s}(\mathbf{x})$, by a geometry decoder $\boldsymbol{f_g}$:

$$\boldsymbol{f_g}\left(F^g_{\mathbf{x},\theta}, F^c_{\mathbf{x},\theta}\right) \mapsto (\mathbf{h}^g_{\mathbf{x}}, \mathbf{e}(\mathbf{x}), \mathbf{s}(\mathbf{x})). \tag{2}$$

The geometry hidden vector $\mathbf{h}^g_{\mathbf{x}}$ and radiance fields $\mathbf{e}(\mathbf{x})$ are shared by the color and event CRF decoders.

3.2. Decomposition of the Radiance Fields

In standard imaging devices, the incoming radiance undergoes linear and nonlinear image processing before being mapped into pixel values and stored in images. This entire image processing can be represented by a single function $\boldsymbol{f_c}$ called the camera response function (CRF) [9]. However, the traditional NeRF method [33] simplifies the imaging process, leading to discrepancies between the rendering and actual images. This deviation is further amplified in multi-modal implicit representations. As Fig. 10 shows, the captures of RGB and event cameras are significantly distinct, which can lead to joint optimization fluctuation. To address this issue, we model the radiance $\mathbf{e}(\mathbf{x})$ and exposure $\Delta \mathbf{t}_c$ of a ray $\mathbf{r}$ but take the aperture and others as implicit factors to obtain the color field $\mathbf{c}(\mathbf{x})$ [21, 49] by differentiable tone-mapping:

$$\mathbf{c}(\mathbf{x}) = \boldsymbol{f_c}\left(\mathbf{e}(\mathbf{x})\Delta \mathbf{t}_c\right). \tag{3}$$

To facilitate optimization, we convert all numerical values into the logarithmic domain and present the inverse function of $\left(\ln \boldsymbol{f_c}^{-1}\right)^{-1}$ as $\boldsymbol{\Psi}_\mathbf{c}$:

$$\begin{aligned} \mathbf{c}(\mathbf{x}) &= \left(\ln \boldsymbol{f_c}^{-1}\right)^{-1}\left(\ln \mathbf{e}(\mathbf{r}) + \ln \Delta \mathbf{t}_c\right) \\ &= \boldsymbol{\Psi}_\mathbf{c}\left(\ln \mathbf{e}(\mathbf{x}) + \ln \Delta \mathbf{t}_c\right), \end{aligned} \tag{4}$$

As for the event camera, directly obtaining the event data is not feasible. However, we can predict high dynamic range luminance $l(\mathbf{x})$ and derive events using the event generation model: As shown in Fig. 2, an event $E_k = (u_k, v_k, t_k, p_k)$ at image coordinate $\mathbf{m}_k = [u_k, v_k, 1]^T$ is triggered if the corresponding logarithmic brightness change $\mathbf{L}(\mathbf{m}, t)$ exceeds a threshold C:

$$\mathbf{L}(\mathbf{m}_k, t_k) - \mathbf{L}(\mathbf{m}_k, t_{k-1}) = p_k C, \ \ p_k \in \{+1, -1\}. \tag{5}$$

The logarithmic brightness $\mathbf{L}(\mathbf{m}, t)$ can be obtained by:

$$\mathbf{L}(\mathbf{m}, t) = \operatorname{lin}\log(\mathbf{I}_e(\mathbf{m})) = \begin{cases} \mathbf{I}_e(\mathbf{m}) \cdot \ln(B)/B, & \text{if } \mathbf{I}_e(\mathbf{m}) < B \\ \ln(\mathbf{I}_e(\mathbf{m})), & \text{else} \end{cases}, \tag{6}$$

where B denotes the linear region threshold [7] and the imaging brightness $\mathbf{I}_e(\mathbf{m})$ of event camera equals to the corresponding luminance of ray $\mathbf{r}$. By applying the modeling approach in Eqs. (3) and (4) to the CRF of an event camera, we establish the relation among the luminance field $l(\mathbf{x})$, the radiance $\mathbf{e}(\mathbf{x})$ and exposure:

$$l(\mathbf{x}) = \boldsymbol{\Psi}_l\left(\ln \mathbf{e}(\mathbf{x}) + \ln \Delta \mathbf{t}_l\right), \tag{7}$$

where $\boldsymbol{\Psi}_l$ and $\Delta \mathbf{t}_l$ denote the luminance tone-mapping and pseudo exposure of the event camera. In this way, we decompose the shared radiance field $\mathbf{e}(\mathbf{x})$ into the RGB and event camera imaging processes through two differentiable tone-mapping processes.

3.3. Differentiable CRF Rendering

Upon obtaining the color and luminance fields in Sec. 3.2, we render the final imaging RGB, luminance, and depth by integrating predicted values along the samples in a ray $\mathbf{r}$:

$$\mathbf{x}_i = \mathbf{O} + z_i \mathbf{d}, \quad i \in \{1, ..., M\}, \tag{8}$$

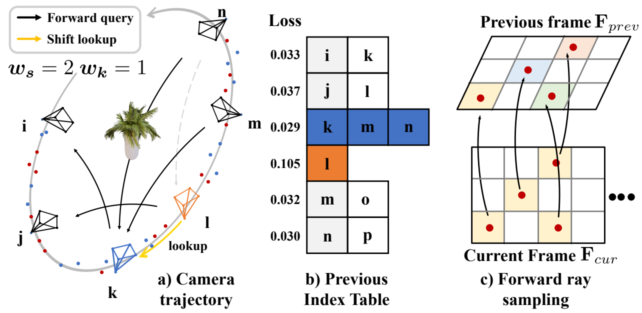


Figure 4. The illustration of event temporal aggregating optimization strategy. In the tracking and global BA stages, EN-SLAM adaptively forwards query the previous frame according to the previous index table, and sample rays from different views perform joint optimization in Eq. (13).

where $\mathbf{O} \in \mathbb{R}^3$ is the camera origin, $\mathbf{d} \in \mathbb{R}^3$, $\|\mathbf{d}\| = 1$ is the ray direction, and $z_i \in \mathbb{R}$ denotes the depth. Hence, we obtain the final imaging color $\hat{\mathbf{c}}$, luminance $\hat{l}$ and depth $\hat{d}$:

$$\begin{aligned} \hat{\mathbf{c}}(\mathbf{r}, \Delta t_c) &= \sum_{i=1}^{i=M} w_i \boldsymbol{\Psi}_\mathbf{c}(\ln \mathbf{e}(\mathbf{x}_i) + \ln \Delta t_c), \\ \hat{l}(\mathbf{r}, \Delta t_l) &= \sum_{i=1}^{i=M} w_i \boldsymbol{\Psi}_l(\ln \mathbf{e}(\mathbf{x}_i) + \ln \Delta t_l), \\ \hat{d}(\mathbf{r}) &= \sum_{i=1}^{i=M} w_i z_i. \end{aligned} \tag{9}$$

We utilize the simple bell-shaped model [1] and compute weights w_i by two sigmoid functions $\sigma(\cdot)$ to convert predicted TSDF $\mathbf{s}(\mathbf{x}_i)$ into weight w_i:

$$w_i = \sigma\left(\frac{\mathbf{s}(\mathbf{x}_i)}{tr}\right)\sigma\left(-\frac{\mathbf{s}(\mathbf{x}_i)}{tr}\right), \tag{10}$$

where tr is the truncation distance of a ray $\mathbf{r}$.

3.4. Tracking and Bundle Adjustment

In this section, to leverage the HDR and temporal difference properties of events, we propose an event joint tracking and global BA strategy in Sec. 3.4.1 that incorporates events into optimization, thus improving the accuracy and robustness. Besides, we introduce adaptive forward-query and sampling strategies in Sec. 3.4.2 and Sec. 3.4.3, which select event data and ray samples with more elevated confidence for optimization, thereby boosting the convergence.

3.4.1 Event Temporal Aggregating Optimization

The overview of ETA is shown in Fig. 4 and Algorithm 1. **For tracking**, we representate the camera pose $\mathbf{P}_{cur} = \exp\left(\xi_t^\wedge\right) \in \mathbb{SE}(3)$ of current frame $\mathbf{F}_{cur}$ and initialize with constant assumption. By selecting N_t rays within $\mathbf{F}_{cur}$ and performing an adaptive event forward query in Sec. 3.4.2 with a probability-weighted sampling in Sec. 3.4.3, we get the event stream and previous rays. Then, we iteratively optimize the pose by minimizing objective functions. **For global BA**, N_{ba} rays from the global keyframe list are sampled to be the subset of pixels $\{PX_{cur}\}$. And then, the forward query and probability-weighted sampling are performed for each sample to get the previous subset $\{PX_{prev}\}$ and events $\{E_k\}_{k=prev}^{cur}$. Finally, a joint optimization is performed to optimize the geometry decoder $\boldsymbol{f_g}$, differentiable CRF $\{\boldsymbol{\Psi_c}, \boldsymbol{\Psi}_l\}$, and poses $\{\mathbf{P_i}\}_{i=0}^{cur}$.

3.4.2 Adaptive Forward Event Query

As shown in Fig. 4, ETA performs an adaptive event forward window selection in tracking and BA by constructing a previous index table to prioritize reliable prior frames for optimization. Specifically, ETA uses a default window w_d and performs a forward query. If the loss of a queried frame exceeds a threshold $\mathcal{L}_s$, we conduct a shift lookup within a neighborhood of length $2 \times w_k$, selecting the frame with the minimum loss as the forward frame for event loss calculation Eq. (13). This event temporal constraint provides a stable local constraint between the participating frames and effectively leverages the high HDR property of events.

Algorithm 1: Event temporal aggregating optimization

Input : RGBD-E stream $\{\mathbf{I}_i, d_i\}_{i=1}^{J}$ and $\{\mathbf{E}_k\}_{k=1}^{N}$.
Output: Loss $\{\mathcal{L}_{ev}, \mathcal{L}_{rgb}, \mathcal{L}_d, \mathcal{L}_{sdf}, \mathcal{L}_{fs}\}$.

while $j < J$ **do**
  for $i \in \{j\}$ **if not BA else** $\{0, 1, ..., j\}$ **do**
    /* Forward Query Sec. 3.4.2 */
    $\mathbf{F}_{cur}, \mathbf{F}_{prev} \leftarrow$ Tab(i), Tab($i - w_d$);
    if $\mathcal{L}_{total} > \mathcal{L}_s$ **then**
      $\mathbf{F}_{prev} \overset{\min}{\leftarrow}$ Tab($i - w_s - w_d, i + w_s$).Loss
    end
    Probability-weighted ray sampling: Rays$_{cur}$, Rays$_{prev} \leftarrow \mathbf{F}_{cur}, \mathbf{F}_{prev}$; // Sec. 3.4.3
    Ray rendering: $\hat{\mathbf{L}}(\mathbf{m}, t_\beta) - \hat{\mathbf{L}}(\mathbf{m}, t_\alpha)$; // Sec. 3.3
    Event accumulation: $\sum_{t_k=t_{cur}}^{t_k=t_{prev}} p_k C$;
    Calculate loss: $\mathcal{L}_{ev}, \mathcal{L}_{rgb}, \mathcal{L}_d, \mathcal{L}_{sdf}, \mathcal{L}_{fs}$; // Eqs. (13) to (15)
    Tab(i) $\leftarrow \{\mathcal{L}_{total}$, cur, prev$\}$
  end
end

3.4.3 Probability-weighted Sampling Strategy.

To take advantage of hybrid multimodality and reduce computational costs, we propose to utilize the RGB loss to guide ray sampling in the event plane. As shown in Fig. 5, the algorithm starts by dividing the RGB image into $h \times w$ patches and randomly sampling N_c rays from each patch to obtain the loss for each sample. Then, we calculate the average loss of each patch and project the center $\mathbf{m}_c$ to a downsampled mini-plan plane of the event camera:

$$\mathbf{m} = \frac{1}{Z_e} \mathbf{K}_m \left[\mathbf{I}_{3\times3} | \mathbf{0}_{3\times1}\right] \begin{bmatrix} \mathbf{T}_{ec}\mathbf{K}^{-1}\mathbf{m}Z_c \\ 1 \end{bmatrix}, \tag{11}$$

where Z_c and Z_e are the depths of two planes, $\mathbf{K}_m$ is the intrinsic of event mini-plane, and $\mathbf{T}_{ec}$ denotes the transformation between cameras. We apply the bilinear interpolation to compute the loss for each pixel in the mini-plan. Finally, the divided patches of the event plane query the loss $\{\mathcal{L}_e^q\}_{q=0}^{Q}$ from the mini-plane and sample rays with probability distribution $\boldsymbol{f}(j) = \frac{\mathcal{L}_e^q}{\sum_{q=1}^{Q} \mathcal{L}_e^q}$.

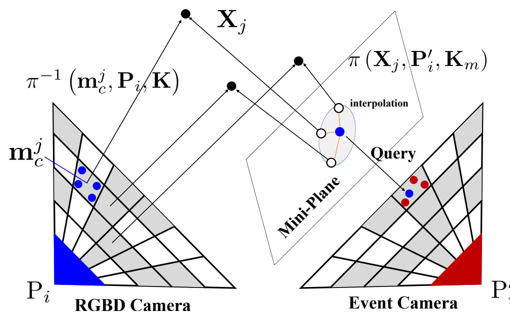


Figure 5. Illustration of the proposed probability-weighted sampling strategy. We utilize the loss of the RGBD plane (left) to guide ray sampling in the event plane (right).

3.4.4 Objective Functions.

According to the EGM in Eq. (5), although it is not possible to directly model the luminance signals supervision, the logarithmic brightness differences $\hat{\mathbf{L}}(\mathbf{m}, t_\beta) - \hat{\mathbf{L}}(\mathbf{m}, t_\alpha)$ can be rendered from two camera poses $\mathbf{P}_\alpha$ and $\mathbf{P}_\beta$ with Eq. (9). By integrating it with Eqs. (5) and (6), we obtain:

$$\mathbf{L}(\mathbf{m}, t_\beta) - \mathbf{L}(\mathbf{m}, t_\alpha) = \sum_{t_k=t_\alpha}^{t_k=t_\beta} p_k C \approx \hat{\mathbf{L}}(\mathbf{m}, t_\beta) - \hat{\mathbf{L}}(\mathbf{m}, t_\alpha). \tag{12}$$

Thus, we establish the relation between events and rendering, and define event reconstruction loss as:

$$\mathcal{L}_{\mathrm{ev}}(t_\beta, t_\alpha) = \mathrm{MSE}\left(\sum_{t_k=t_\alpha}^{t_k=t_\beta} p_k C - \hat{L}(\mathbf{m}, t_\beta) + \hat{L}(\mathbf{m}, t_\alpha)\right). \tag{13}$$

In our implementation, we perform a normalization on Eq. (13) to eliminate C when it is unavailable. The color and depth rendering losses in a valid ray batch R between the rendering and observations are also utilized:

$$\mathcal{L}_{\mathrm{rgb}} = \frac{1}{|R|} \sum_{\mathbf{r}\in\mathbf{R}} (\hat{\mathbf{c}}(\mathbf{r}, \Delta t_c)) - c(\mathbf{r}))^2, \mathcal{L}_{\mathrm{d}} = \frac{1}{|R|} \sum_{\mathbf{r}\in\mathbf{R}} (\hat{d}(\mathbf{r}) - d(\mathbf{r}))^2, \tag{14}$$

where $c(\mathbf{r})$ and $d(\mathbf{r})$ are the ground truth color and depth. To achieve an accurate geometry reconstruction, we apply the approximated SDF loss and free-space loss to sampled point $\mathbf{x}$ near the surface ($S_r^{tr} = \{\mathbf{x} \mid |d(\mathbf{r}) - d(\mathbf{x})| \le tr\}$) and far from the surface ($S_r^{fs} = \{\mathbf{x} \mid |d(\mathbf{r}) - d(\mathbf{x})| > tr\}$):

$$\begin{aligned} \mathcal{L}_{sdf} &= \frac{1}{|R|} \sum_{r\in R} \frac{1}{|S_r^{tr}|} \sum_{\mathbf{x}\in S_r^{tr}} \left(\mathbf{x} - (\hat{d}(\mathbf{r}) - d(\mathbf{r}))\right)^2, \\ \mathcal{L}_{fs} &= \frac{1}{|R|} \sum_{r\in R} \frac{1}{\left|S_r^{fs}\right|} \sum_{\mathbf{x}\in S_r^{fs}} (\mathbf{x} - tr)^2. \end{aligned} \tag{15}$$

4. Dataset

To our knowledge, there is currently no SLAM dataset that satisfactorily tackles challenges posed by strong motion blur and lighting variations while encompassing RGBD and

Figure 6. **Overview of the DEV-Indoors and DEV-Reals datasets**. **DEV-Indoors** is obtained through Blender [6] and simulator [14], covering normal, motion blur, and dark scenes, providing 9 subsets with RGB images, depth maps, event streams, meshes, and trajectories. **DEV-Reals** is captured from real scenes, providing 8 challenging subsets under motion blur and lighting variation.

event streams for NeRF-based SLAM. Common datasets lack depth [19] or ground truth meshes [57]. Additionally, they are primarily focused on outdoor scenes [60], without significant motion blur [4] or lighting variation [29]. Therefore, in this paper, we construct simulated Dynamic Event RGBD Indoor (DEV-Indoors) and Dynamic Event RGBD Real captured (DEV-Reals) datasets, as shown in Fig. 6. Besides, we use Vector [13] dataset for evaluation as well.

1) DEV-Indoors is rendered from 3 Blender [6] models: `#room`, `#apartment`, and `#workshop`. We generated 9 subsets containing high-quality color images, depth, meshes, and ground truth trajectories by varying the scene lighting and camera exposure time. The events are further generated via the events simulator [14].

2) Dev-Reals is captured from 3 real scenes: `#Pioffice`, `#Garage` and `#Dormitory`. Our capture system comprises a LiDAR (for ground truth pose), a Realsense D435I RGBD camera, and a DAVIS346 event camera. Eight subsequences are captured by modifying the lighting conditions and camera movement speed in the environment.

5. Experiment

Baselines. To the best of our knowledge, there is currently no event-based RGBD dense vSLAM with available public code that can be directly compared with our method. We opt EVO [39], ESVO [64], USLAM [53] as a reference from the most relevant event-based methods [5, 12, 19, 27, 57, 67]. We also compare our method with the existing SOTA NeRF-based methods: iMAP [48], NICE-SLAM [66], CoSLAM [54], and ESLAM [23].

Metric. We use the absolute trajectory error [47] (ATE) (cm) to measure the localization accuracy. For map reconstruction, we use the 2D Depth L1 (cm) [66], 3D accuracy (cm), completion (cm), and completion ratio (%) to measure the scene geometry with mesh culling [1, 23]. The evaluation datasets are generated by randomly conducting 2000

Table 1. **Tracking (ATE RSME [cm]) comparison on DEV-Indoors**. Our method outperforms previous works, demonstrating its robustness under motion blur and luminance variation.

Method	#Rm norm	#Rm blur	#Rm dark	#Apt norm	#Apt blur	#Apt dark	#Wkp norm	#Wkp blur	#Wkp dark	#all avg
iMAP [48]	41.08	50.58	70.77	25.75	14.41	$1.06e^5$	276.91	891.86	345.21	214.57
NICE-SLAM [66]	17.06	29.54	30.53	25.17	44.22	48.28	✗94 %	✗33 %	✗33%	32.47
CoSLAM[54]	10.71	10.88	26.64	10.02	13.03	30.75	7.96	14.37	17.88	15.80
ESLAM [23]	10.72	15.55	40.42	9.99	12.79	12.39	7.01	15.07	7.97	14.66
Ours	**9.62**	**9.72**	**9.94**	**8.62**	**8.77**	**9.21**	**6.74**	**7.51**	**6.94**	**8.56**

poses and depths in Blender [6]. We run all the methods 5 times and report the average results or ✗+ tracking success ratio if a method crashes.

Implementation Details. EN-SLAM is implemented in Python and trained on a desktop PC with a 5.50GHz Intel Core i9-13900K CPU and NVIDIA RTX 4090 GPU. We run EN-SLAM at 17 FPS and sample 1024 and 2048 rays in tracking and BA stages with 10 iterations by default. The event joint global BA is performed every 5 frames with 5% of pixels from all keyframes. The model is trained using Adam optimizer with learning rate $lr_{rot} = 1e^{-3}, lr_{trans} = 1e^{-3}$, and loss weights $\lambda_{ev} = 0.05, \lambda_{rgb} = 5.0, \lambda_d = 0.1$. Default window w_d and neighborhood window are set as 5 and 2, respectively. The exposures of RGB and event cameras are $5.21e^{-5}$ in DEV-Indoors. We use two sigmoid functions to fit the exposures if they are unavailable. Detailed settings can be found in the supplemental materials.

5.1. Evaluation of Tracking and Mapping

Evaluation on DEV-Indoors. We report the trajectory accuracy and reconstruction quality in Tab. 1 and Tab. 2. As shown in Tab. 1, EN-SLAM performs the best in all 9 scenes and remains stable (fluctuation below 0.8) under all the blur and dark sequences. While the others perform sensitive and unstable when facing non-ideal environments, especially iMAP [48] and NICE-SLAM [66], which completely crushes in the blur and dark scenes. ESLAM [66]

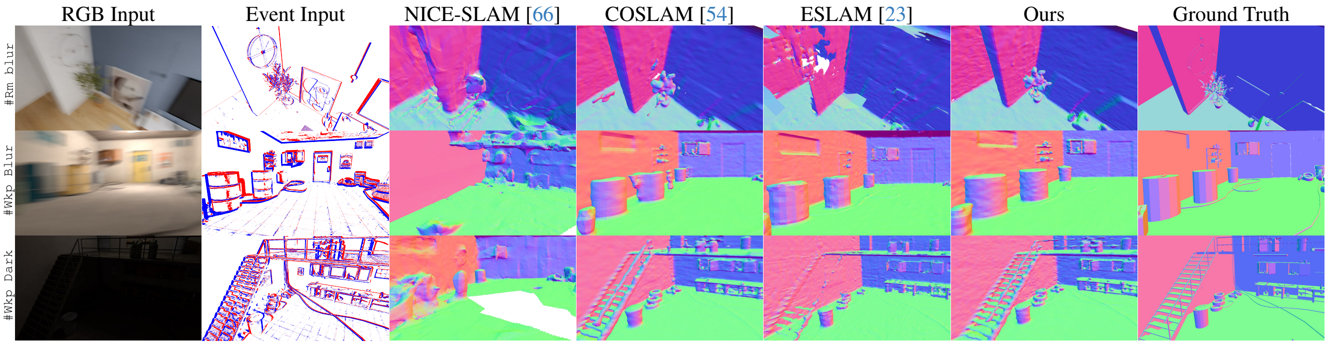


Figure 7. **Reconstruction Performance on DEV-Indoors.** EN-SLAM achieves, on average, more precise reconstruction details than existing methods in motion blur and lighting varying environments with the assistance of high-quality event streams.

Table 2. **Reconstruction Performance [cm]** of the proposed method vs. the state-of-the-art methods on **DEV-Indoors** dataset.

Method	Metric	#Rm **norm**	#Rm **blur**	#Rm **dark**	#Apt **norm**	#Apt **blur**	#Apt **dark**	#Wkp **norm**	#Wkp **blur**	#Wkp **dark**	#all **avg**
iMAP [48]	Acc↓	37.16	37.60	34.30	25.55	44.288	54.47	56.40	51.74	38.12	42.18
	Comp↓	48.69	55.97	33.76	19.28	23.94	49.61	65.40	27.18	47.11	41.22
	Comp Ratio↑	37.56	37.76	39.11	60.03	46.27	42.56	12.14	48.44	40.41	40.48
	Depth L1↓	79.93	97.98	64.96	40.36	128.45	140.00	131.79	115.07	124.69	102.58
NICESLAM [66]	Acc↓	18.49	18.86	16.69	21.27	16.51	19.17	26.35	22.09	28.40	20.87
	Comp↓	20.26	21.93	21.43	20.67	18.70	21.29	28.04	49.82	77.19	31.04
	CompRatio ↑	60.40	59.03	58.14	59.60	63.49	56.55	50.91	46.00	35.58	54.41
	Depth L1↓	40.59	42.09	40.26	41.54	24.00	34.89	62.48	104.20	106.09	55.13
CoSLAM [54]	Acc↓	10.66	11.36	12.77	15.47	16.42	30.71	13.02	17.59	19.85	16.43
	Comp↓	13.24	12.44	12.23	14.09	15.36	21.67	13.92	18.26	19.46	15.63
	Comp Ratio↑	69.22	76.87	77.26	70.61	66.75	55.26	67.92	61.26	60.70	67.3
	Depth L1↓	24.78	20.91	20.65	32.29	35.90	64.14	28.69	39.17	42.85	34.38
ESLAM [23]	Acc↓	9.48	**8.58**	11.81	12.86	16.25	14.85	**9.01**	10.01	10.02	11.43
	Comp↓	7.94	**7.54**	**7.08**	8.511	12.37	10.40	**8.89**	10.95	10.44	**9.35**
	Comp Ratio↑	**84.60**	**85.69**	**86.70**	82.93	71.53	80.90	**83.17**	80.02	81.64	81.91
	Depth L1↓	15.34	**12.27**	19.08	11.07	27.80	15.50	30.03	29.06	28.02	20.91
Ours	Acc↓	**7.48**	10.53	**7.07**	**9.46**	**9.91**	**9.34**	9.23	**9.28**	**9.25**	**9.06**
	Comp↓	**7.70**	12.51	7.70	**9.87**	**9.28**	**9.61**	9.95	**9.92**	**9.92**	9.61
	Comp Ratio↑	83.00	74.48	84.26	**85.36**	**83.40**	**84.01**	82.27	**82.38**	**82.35**	**82.39**
	Depth L1↓	**15.10**	23.36	**11.92**	**19.86**	**11.94**	**19.21**	**23.16**	**23.14**	**23.39**	**19.01**

Table 3. **Tracking comparison (ATE median [cm])** of the proposed method vs. the state-of-the-art methods on **DEV-Reals**.

Method	Pio1	Pio2	Gre1	Gre2	dorm1	dorm2	dorm3	dorm4	**avg**
ORBSLAM [19]	✗63%	✗63%	✗63%	✗63%	✗63%	✗63%	✗63%	✗63%	✗63%
NICE-SLAM [66]	13.21	23.35	✗63%	✗25%	24.69	10.68	18.44	44.04	✗22.40
COSLAM [54]	11.14	19.83	82.52	40.16	15.99	15.42	30.12	32.45	30.95
ESLAM [23]	11.28	21.42	63.65	30.75	37.94	31.04	16.19	37.91	31.27
Ours	**8.94**	**19.05**	**43.63**	**21.18**	**11.26**	**11.91**	**16.00**	**19.78**	**18.97**

and CoSLAM [54] exhibit similar trends but achieve relatively stable results. Specifically, in `#room` sequences, they achieved 1.02 and 1.45 times the error on the blur subset and 2.49 and 3.77 times the error on the dark subset, respectively. The **reconstruction quality** in Tab. 2 and Fig. 7 show that EN-SLAM performs more accurately and robustly than the other methods. Specifically, our method reduces the error by 2.37, 0.48, and 1.90 in ACC, Comp, and Depth L1 compared with the second ESLAM [23]. Fig. 7 also shows that our method reconstructs the details of the scenes more accurately and produces fewer artifacts.

Evaluation on DEV-Reals. Tab. 3 illustrates the tracking performance of our method and the state-of-the-art methods on the DEV-Reals dataset. Our method achieves the best performance in all the scenes (18.97), and the average error is 1.63 times lower than the second-best CoSLAM [54] (30.95). Note that DEV-Reals is challenging due to the large motion and varying light, leading to the crushes of ORB-SLAM [34] and NICE-SLAM [66].

Evaluation on Vector. Tab. 4 compares EN-SLAM with

Table 4. **Tracking comparison (ATE mean [cm])** of the proposed method vs. the Event-based SLAM system on **Vector**[13] dataset.

Method	robot norm	robot fast	desk norm	desk fast	sofa norm	sofa fast	hdr norm	hdr fast	**#all avg**
EVO [39]	3.25	✗	✗	✗	✗	✗	✗	✗	3.25
ESVO [64]	✗	✗	✗	✗	**1.77**	✗	✗	✗	1.77
USLAM [53] (EVIO)	1.18	**1.65**	2.24	**1.08**	5.74	2.54	5.69	2.61	2.84
CoSLAM [54] (DV)	**1.00**	124.69	**1.76**	97.65	1.74	77.89	1.47	1.42	38.45
ESLAM [23] (DV)	1.39	3.30	2.54	3.64	7.99	19.03	7.38	12.23	7.19
Ours (EDV)	1.06	1.73	**1.76**	2.69	2.02	**1.84**	**1.03**	**1.22**	**1.67**

E: event, V: RGB or gray image, D: depth, I: IMU, S: stereo, O: odometry.

Table 5. **Run-time comparison** on DEV-Indoors. EN-SLAM is comparable to the most efficient ESLAM and keeps lightweight.

Method	Tracking [ms×it] ↓	Mapping [ms×it] ↓	FPS ↑	#parama.
iMAP [48]	24.73×50	41.18×300	0.36	0.22 M
NICE-SLAM [66]	6.46×16	26.42×120	1.55	5.86 M
CoSLAM[54]	6.08×15	13.52×15	11.26	1.71 M
ESLAM [23]	5.20×13	16.68×10	14.77	7.85 M
Ours	5.75×10	13.16×10	**17.40**	1.95 M

the other event-aided SLAM systems in vector [13] dataset. Our method outperforms USLAM [53] on average by 1.14 cm and achieves the best performance in 5 of 8 scenes. We must emphasize that Tab. 4 is not a strictly fair comparison because all the methods use different modalities (RGB, depth, and IMU) or camera systems (monocular and stereo).

5.2. Runtime Analysis

We evaluate all the frameworks on an NVIDIA RTX 4090 GPU and report average tracking and mapping iterations spending, FPS, and parameters number of the model in Tab. 5. The experimental results indicate that our method is fast, with an average of 17 FPS, comparable to the currently most efficient ESLAM [23]. Meanwhile, our method remains lightweight, with only 1.95M parameters, yet achieves the best accuracy.

5.3. Evaluation of Rendering

We compare the rendering performance in Fig. 5.3 (left), EN-SLAM outperforms most SOTA works in image quality. The thumbnails in Fig. 5.3 (right) show that EN-SLAM achieves more precise rendering details than previous methods. Specifically, on `#Rm Blur`, EN-SLAM yields more refined results, while CoSLAM and ESLAM exhibit ghosting. Note that in `#Rm Dark` and `#Wkp Dark`, all the RGB rendering is dark and blurred, while our method can still generate high-quality luminance results with the assistance of the HDR event stream.

Method	Metric	#Rm norm	#Rm blur	#Rm dark	#Apt norm	#Apt blur	#Apt dark	#Wkp norm	#Wkp blur	#Wkp dark
NICESLAM [66]	PSNR	13.65	18.24	28.09	14.23	17.50	28.37	✗	✗	✗
	SSIM	0.457	0.623	0.828	0.445	0.573	0.853			
	LPIPS	0.646	0.485	0.349	0.673	0.552	0.325			
CoSLAM [54]	PSNR	23.16	24.86	31.22	22.79	23.85	**32.45**	**24.12**	**25.11**	39.13
	SSIM	0.785	0.830	0.883	0.768	0.799	**0.925**	**0.821**	0.846	0.962
	LPIPS	0.487	0.428	0.392	0.515	0.523	0.289	**0.462**	0.451	0.183
ESLAM [23]	PSNR	19.52	20.70	28.48	18.68	15.11	31.15	16.38	18.35	31.08
	SSIM	0.670	0.715	0.841	0.614	0.518	0.895	0.519	0.603	0.905
	LPIPS	0.522	0.487	0.414	0.606	0.836	**0.285**	0.688	0.664	0.255
Ours	PSNR	**23.72**	**25.11**	**32.64**	**23.08**	**24.53**	31.26	23.83	**25.11**	**39.38**
	SSIM	**0.808**	**0.840**	**0.911**	**0.777**	**0.821**	0.909	0.810	**0.848**	**0.963**
	LPIPS	**0.468**	**0.423**	**0.349**	**0.510**	**0.493**	0.358	0.481	**0.448**	**0.182**

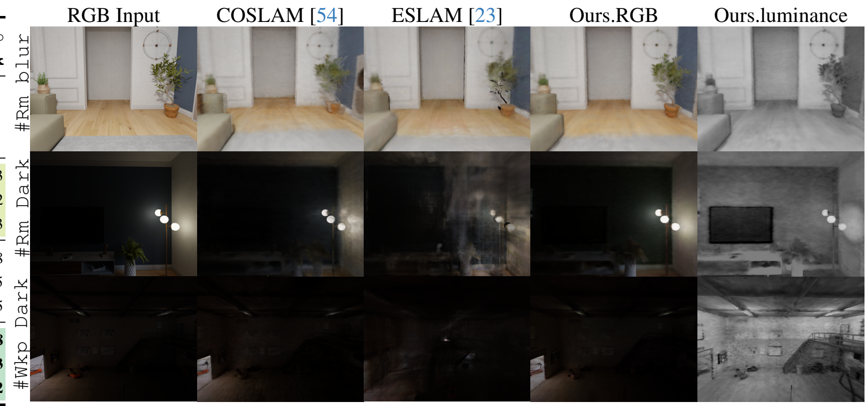
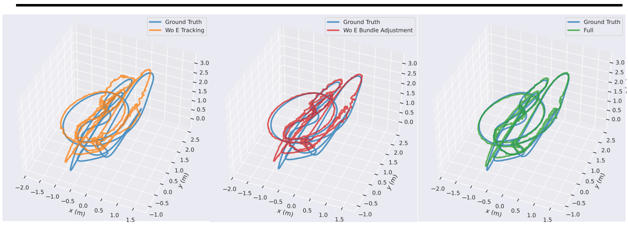


Figure 8. **Rendering Performance on DEV-Indoors.** left): Our method outperforms most previous works in image quality evaluation under non-ideal environments. right): EN-SLAM achieves more precise rendering details on average than previous methods.

Tracking Event	Mapping Event	#Rm blur ATE↓	ACC↓	Comp↓	Comp ratio↑	#Dorm2 Median↓	RSME↓
✗	✗	11.89	8.61	10.98	76.31	14.46	18.75
✗	✓	10.73	8.32	9.53	81.83	14.17	17.07
✓	✗	11.68	8.28	10.28	79.05	16.09	19.72
✓	✓	**9.61**	**7.88**	**7.59**	**83.51**	**11.91**	**15.47**
RGB 1st-2nd only		10.92	9.02	9.15	82.81	13.52	17.80
W/o RGB		12.07	11.12	11.05	76.27	22.50	26.48



W/o E Mapping. ATE: 11.68 W/o E Tracking. ATE: 10.73 FULL. ATE: 9.61

Figure 9. **Ablation study of modalities** on the `#Rm blur` and `#Drom2` subset of DEV-Indoors and DEV-Reals (15 iterations).

5.4. Ablation Study

Effect of event and RGB modalities. Fig. 9 illustrates quantitative evaluation using ETA in tracking and mapping. Our full model achieve lower tracking error of 9.61 and 15.47 than the model w/o ETA in tracking (10.73 and 17.07) and mapping (11.68 and 19.72) on the `#Rm blur` and `#Dorm2`, respectively. The results also show that the full model surpasses the model w/o ETA by 0.73 and 3.39% in ACC and completion. In addition, the RGB is also critical, but with the initialization of RGB in 2 frames, the performance is significantly improved, benefiting from the CRF.

Effect of CRF and probability-weighted sampling. Fig. 9 show the performance of differentiable CRF and probability-weighted sampling strategy (PWS). The system without CRF might suffer from the distinct event and RGB imaging process, resulting in fluctuating training and poor performance, especially in real datasets. It significantly reduces the tracking ATE RSME from 30.18 to 15.47 in `#Drom2` and reconstruction completion from 9.78 to 7.59 in `#Rm blur`. The results also show that the full model surpasses the model w/o PWS by 0.25 and 1.9% in ATE and completion on `#Rm blur`. A visualization of CRF is shown in Fig. 10, on dark scene `#Drom2`, the model with CRF renders HDR luminance results and accurate mesh benefiting from events. In contrast, the model w/o CRF suffers from the low dynamic range RGB input.

Setting	# Rm blur ATE↓	ACC↓	Comp↓	Comp ratio↑	#Dorm2 Median↓	RSME↓
w/o CRF	12.12	8.29	9.78	83.57	27.67	30.18
w/o PWS	9.86	7.88	9.49	81.04	16.59	19.78
Full model	**9.61**	**7.88**	**7.59**	**83.51**	**11.91**	**15.47**

Table 6. **Ablation study of CRF and PWS** on the `#Rm blur` and `#Drom2` subset of DEV-Indoors and DEV-Reals (15 iterations).

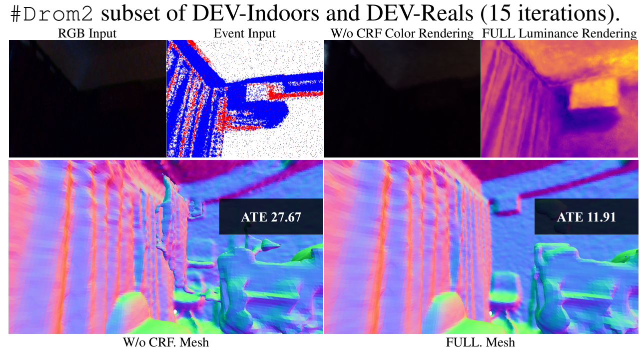


Figure 10. **CRF ablation** on the `#Dorm2` of DEV-Reals.

6. Conclusion and Limitation

This paper first integrates the event stream into the implicit neural SLAM framework to overcome challenges in scenes with motion blur and lighting variation. A differentiable CRF rendering technique that maps the unified representation to color and luminance is proposed to address the significant distinction between event and RGB. An event temporal aggregating optimization strategy that capitalizes the consecutive difference constraints of events is presented to enhance the optimization. We construct two challenging datasets and evaluate the effectiveness and robustness of EN-SLAM under various environments.

Limitations: EN-SLAM is designed for indoor scenes and does not consider outdoor open scenarios. In future work, we aim to extend EN-SLAM to large-scale outdoor environments and enhance its generalization capability.

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, pages 6290–6301, 2022. 4, 6

[2] Christian Brändli, Jonas Strubel, Susanne Keller, Davide Scaramuzza, and Tobi Delbruck. Elised—an event-based line segment detector. In *EBCCSP*, pages 1–7. IEEE, 2016. 2

[3] Guillaume Bresson, Zayed Alsayed, Li Yu, and Sébastien Glaser. Simultaneous localization and mapping: A survey of current trends in autonomous driving. *T-IV*, 2(3):194–220, 2017. 1

[4] Samuel Bryner, Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. Event-based, direct camera tracking from a photometric 3d map using nonlinear optimization. In *ICRA*, pages 325–331. IEEE, 2019. 6

[5] Andrea Censi and Davide Scaramuzza. Low-latency event-based visual odometry. In *ICRA*, pages 703–710. IEEE, 2014. 6

[6] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 6

[7] Tobi Delbruck, Yuhuang Hu, and Zhe He. V2e: From video frames to realistic dvs event camera streams. *arXiv e-prints*, pages arXiv–2006, 2020. 4

[8] Parth Rajesh Desai, Pooja Nikhil Desai, Komal Deepak Ajmera, and Khushbu Mehta. A review paper on oculus rift-a virtual reality headset. *arXiv preprint arXiv:1408.1173*, 2014. 1

[9] Frederic Dufaux, Patrick Le Callet, Rafal Mantiuk, and Marta Mrak. *High dynamic range video: from acquisition, to display and applications*. Academic Press, 2016. 4

[10] Kamak Ebadi, Lukas Bernreiter, Harel Biggie, Gavin Catt, Yun Chang, Arghya Chatterjee, Christopher E Denniston, Simon-Pierre Deschênes, Kyle Harlow, Shehryar Khattak, et al. Present and future of slam in extreme underground environments. *arXiv preprint arXiv:2208.01787*, 2022. 1

[11] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *ECCV*, pages 834–849. Springer, 2014. 1

[12] Guillermo Gallego, Jon EA Lund, Elias Mueggler, Henri Rebecq, Tobi Delbruck, and Davide Scaramuzza. Event-based, 6-dof camera tracking from photometric depth maps. *TPAMI*, 40(10):2402–2412, 2017. 6

[13] Ling Gao, Yuxuan Liang, Jiaqi Yang, Shaoxun Wu, Chenyu Wang, Jiaben Chen, and Laurent Kneip. Vector: A versatile event-centric benchmark for multi-sensor slam. *RA-L*, 7(3): 8217–8224, 2022. 6, 7

[14] Daniel Gehrig, Mathias Gehrig, Javier Hidalgo-Carrió, and Davide Scaramuzza. Video to events: Recycling video datasets for event cameras. In *CVPR*, pages 3586–3595, 2020. 6

[15] Cheng Gu, Erik Learned-Miller, Daniel Sheldon, Guillermo Gallego, and Pia Bideau. The spatio-temporal poisson point process: A simple model for the alignment of event camera data. In *ICCV*, pages 13495–13504, 2021. 3

[16] Weipeng Guan and Peng Lu. Monocular event visual inertial odometry based on event-corner using sliding windows graph-based optimization. In *IROS*, pages 2438–2445. IEEE, 2022. 2

[17] Weipeng Guan, Peiyu Chen, Yuhan Xie, and Peng Lu. Pl-evio: Robust monocular event-based visual inertial odometry with point and line features. *T-ASE*, 2023. 2

[18] Christian Häne, Christopher Zach, Jongwoo Lim, Ananth Ranganathan, and Marc Pollefeys. Stereo depth map fusion for robot navigation. In *IROS*, pages 1618–1625. IEEE, 2011. 1

[19] Javier Hidalgo-Carrió, Guillermo Gallego, and Davide Scaramuzza. Event-aided direct sparse odometry. In *CVPR*, pages 5781–5790, 2022. 3, 6, 7

[20] Kunping Huang, Sen Zhang, Jing Zhang, and Dacheng Tao. Event-based simultaneous localization and mapping: A comprehensive survey. *arXiv preprint arXiv:2304.09793*, 2023. 3

[21] Xin Huang, Qi Zhang, Ying Feng, Hongdong Li, Xuan Wang, and Qing Wang. Hdr-nerf: High dynamic range neural radiance fields. In *CVPR*, pages 18398–18408, 2022. 4

[22] Inwoo Hwang, Junho Kim, and Young Min Kim. Ev-nerf: Event based neural radiance field. In *WACV*, pages 837–847, 2023. 3

[23] Mohammad Mahdi Johari, Camilla Carta, and François Fleuret. Eslam: Efficient dense slam system based on hybrid representation of signed distance fields. In *CVPR*, pages 17408–17419, 2023. 1, 2, 6, 7, 8

[24] Hanme Kim, Ankur Handa, Ryad Benosman, Sio-Hoi Ieng, and Andrew J Davison. Simultaneous mosaicing and tracking with an event camera. *JSSC*, 43:566–576, 2008. 3

[25] Hanme Kim, Stefan Leutenegger, and Andrew J Davison. Real-time 3d reconstruction and 6-dof tracking with an event camera. In *ECCV*, pages 349–364. Springer, 2016. 2, 3

[26] Simon Klenk, Lukas Koestler, Davide Scaramuzza, and Daniel Cremers. E-nerf: Neural radiance fields from a moving event camera. *RA-L*, 8(3):1587–1594, 2023. 3

[27] Beat Kueng, Elias Mueggler, Guillermo Gallego, and Davide Scaramuzza. Low-latency visual odometry using event-based feature tracks. In *IROS*, pages 16–23. IEEE, 2016. 6

[28] Ruoxiang Li, Dianxi Shi, Yongjun Zhang, Kaiyue Li, and Ruihao Li. Fa-harris: A fast and asynchronous corner detector for event cameras. In *IROS*, pages 6223–6229. IEEE, 2019. 2

[29] Wenbin Li, Sajad Saeedi, John McCormac, Ronald Clark, Dimos Tzoumanikas, Qing Ye, Yuzhong Huang, Rui Tang, and Stefan Leutenegger. Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. In *BMVC*, 2018. 6

[30] Qi Ma, Danda Pani Paudel, Ajad Chhatkuli, and Luc Van Gool. Deformable neural radiance fields using rgb and event cameras. In *ICCV*, pages 3590–3600, 2023. 1, 2, 3

[31] Jacques Manderscheid, Amos Sironi, Nicolas Bourdis, Davide Migliore, and Vincent Lepetit. Speed invariant time surface for learning to detect corner points with event-based cameras. In *CVPR*, pages 10245–10254, 2019. 2

[32] Mana Masuda, Yusuke Sekikawa, and Hideo Saito. Event-based camera tracker by ∇tnerf. *IEEE Access*, 11:96626–96635, 2023. 3

[33] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1, 4

[34] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *T-RO*, 33(5):1255–1262, 2017. 1, 7

[35] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. Dtam: Dense tracking and mapping in real-time. In *ICCV*, pages 2320–2327. IEEE, 2011. 1

[36] Urbano Miguel Nunes and Yiannis Demiris. Robust event-based vision model estimation by dispersion minimisation. *TPAMI*, 44(12):9561–9573, 2021. 3

[37] Xin Peng, Ling Gao, Yifu Wang, and Laurent Kneip. Globally-optimal contrast maximisation for event cameras. *TPAMI*, 44(7):3479–3495, 2021. 3

[38] Yunshan Qi, Lin Zhu, Yu Zhang, and Jia Li. E2nerf: Event enhanced neural radiance fields from blurry images. In *ICCV*, pages 13254–13264, 2023. 1, 3

[39] Henri Rebecq, Timo Horstschäfer, Guillermo Gallego, and Davide Scaramuzza. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time. *RA-L*, 2(2):593–600, 2016. 2, 3, 6, 7

[40] Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization. In *BMVC*, 2017. 2

[41] Henri Rebecq, Guillermo Gallego, Elias Mueggler, and Davide Scaramuzza. Emvs: Event-based multi-view stereo—3d reconstruction with an event camera in real-time. *IJCV*, 126 (12):1394–1414, 2018. 3

[42] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *TPAMI*, 43(6):1964–1980, 2019. 2

[43] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *ECCV*, pages 430–443. Springer, 2006. 2

[44] Viktor Rudnev, Mohamed Elgharib, Christian Theobalt, and Vladislav Golyanik. Eventnerf: Neural radiance fields from a single colour event camera. In *CVPR*, pages 4992–5002, 2023. 1, 3

[45] Erik Sandström, Yue Li, Luc Van Gool, and Martin R Oswald. Point-slam: Dense neural point cloud-based slam. In *ICCV*, pages 18433–18444, 2023. 1, 2

[46] Thomas Schops, Torsten Sattler, and Marc Pollefeys. Bad slam: Bundle adjusted direct rgb-d slam. In *CVPR*, pages 134–144, 2019. 1

[47] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *IROS*, 2012. 6

[48] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. imap: Implicit mapping and positioning in real-time. In *ICCV*, pages 6229–6238, 2021. 1, 2, 6, 7

[49] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022. 4

[50] Stepan Tulyakov, Daniel Gehrig, Stamatios Georgoulis, Julius Erbach, Mathias Gehrig, Yuanyou Li, and Davide Scaramuzza. Time lens: Event-based video frame interpolation. In *CVPR*, pages 16155–16164, 2021. 2

[51] Stepan Tulyakov, Alfredo Bochicchio, Daniel Gehrig, Stamatios Georgoulis, Yuanyou Li, and Davide Scaramuzza. Time lens++: Event-based frame interpolation with parametric non-linear flow and multi-scale fusion. In *CVPR*, pages 17755–17764, 2022. 2

[52] Valentina Vasco, Arren Glover, and Chiara Bartolozzi. Fast event-based harris corner detection exploiting the advantages of event-driven cameras. In *IROS*, pages 4144–4149. IEEE, 2016. 2

[53] Antoni Rosinol Vidal, Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios. *RA-L*, 3(2):994–1001, 2018. 2, 6, 7

[54] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *CVPR*, pages 13293–13302, 2023. 1, 2, 6, 7, 8

[55] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. in 2020 ieee. In *IROS*, pages 4909–4916. 1

[56] Yifu Wang, Jiaqi Yang, Xin Peng, Peng Wu, Ling Gao, Kun Huang, Jiaben Chen, and Laurent Kneip. Visual odometry with an event camera using continuous ray warping and volumetric contrast maximization. *Sensors*, 22(15):5687, 2022. 3

[57] David Weikersdorfer, David B Adrian, Daniel Cremers, and Jörg Conradt. Event-based 3d slam with a depth-augmented dynamic vision sensor. In *ICRA*, pages 359–364. IEEE, 2014. 6

[58] Thomas Whelan, Michael Kaess, Maurice F. Fallon, Hordur Johannsson, John J. Leonard, and John B. McDonald. Kintinuous: Spatially extended kinectfusion. In *AAAI*, 2012. 1

[59] Xingrui Yang, Hai Li, Hongjia Zhai, Yuhang Ming, Yuqian Liu, and Guofeng Zhang. Vox-fusion: Dense tracking and mapping with voxel-based neural implicit representation. In *ISMAR*, pages 499–507. IEEE, 2022. 2

[60] Jie Yin, Ang Li, Tao Li, Wenxian Yu, and Danping Zou. M2dgr: A multi-sensor and multi-scenario slam dataset for ground robots. *RA-L*, 7(2):2266–2273, 2021. 6

[61] Xiang Zhang, Lei Yu, Wen Yang, Jianzhuang Liu, and Gui-Song Xia. Generalizing event-based motion deblurring in real-world scenarios. In *ICCV*, pages 10734–10744, 2023. 1

[62] Youmin Zhang, Fabio Tosi, Stefano Mattoccia, and Matteo Poggi. Go-slam: Global optimization for consistent 3d instant reconstruction. In *ICCV*, pages 3727–3737, 2023. 1

[63] Shibo Zhao, Damanpreet Singh, Haoxiang Sun, Rushan Jiang, YuanJun Gao, Tianhao Wu, Jay Karhade, Chuck Whittaker, Ian Higgins, Jiahe Xu, et al. Subt-mrs: A subterranean, multi-robot, multi-spectral and multi-degraded dataset for robust slam. *arXiv preprint arXiv:2307.07607*, 2023. 1

[64] Yi Zhou, Guillermo Gallego, Henri Rebecq, Laurent Kneip, Hongdong Li, and Davide Scaramuzza. Semi-dense 3d reconstruction with a stereo event camera. In *ECCV*, pages 235–251, 2018. 6, 7

[65] Yi Zhou, Guillermo Gallego, and Shaojie Shen. Event-based stereo visual odometry. *T-RO*, 37(5):1433–1450, 2021. 2, 3

[66] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *CVPR*, pages 12786–12796, 2022. 1, 2, 6, 7, 8

[67] Yi-Fan Zuo, Jiaqi Yang, Jiaben Chen, Xia Wang, Yifu Wang, and Laurent Kneip. Devo: Depth-event camera visual odometry in challenging conditions. In *ICRA*, pages 2179–2185. IEEE, 2022. 6