# Implicit Event-RGBD Neural SLAM

## Supplementary Material


### Abstract

*This supplementary material accompanies the main paper by providing more details for reproducibility as well as additional evaluations and qualitative results to to verify the effectiveness and robustness of EN-SLAM:*

▷ ***Sec. 7****: Configurations of DEV-Indoors dataset, including scene assets, event generation, evaluation dataset, ground truth mesh production, and sequence visualization.*

▷ ***Sec. 8****: Configurations of DEV-Reals dataset, including capture system specifications and sequence visualization.*

▷ ***Sec. 9****: Additional implementation details.*

▷ ***Sec. 10****: Additional experimental results, including more ablation studies, detailed tracking comparison, and mapping reconstruction visualization.*

▷ ***Sec. 11****: Video demonstration.*


## 7. Configurations of DEV-Indoors dataset

Table 7. Comparison of different event-centric datasets. We focus on the availability of event data, color images, depth, and ground truth mesh. **I** denotes indoor scenes. **O** denotes outdoor scenes.

| Dataset | event data | RGB/gray image | Depth image | GT mesh | challenging motion blur | lighting change | indoors / outdoors | Synthetic / Real |
|---|---|---|---|---|---|---|---|---|
| ECDS [41] | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | **I+O** | **S+R** |
| RPG [78] | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | **I** | **S** |
| MVSEC [80] | ✓ | ✓ | ✓ | ✗ | ✗ | ✓ | **I+O** | **R** |
| UZH-FPV [9] | ✓ | ✓ | ✗ | ✗ | ✓ | ✗ | **I+O** | **R** |
| DSEC [17] | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ | **O** | **R** |
| TUM-VIE [30] | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | **I** | **R** |
| EDS [22] | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | **I** | **R** |
| Vector [15] | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | **I** | **R** |
| M2DGR [73] | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | **I+O** | **R** |
| VICON [19] | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | **O** | **R** |
| ViVID++ [33] | ✓ | ✓ | ✓ | ✗ | ✗ | ✓ | **O** | **R** |
| VISTA 2.0 [1] | ✓ | ✓ | ✓ | ✗ | ✗ | ✓ | **O** | **S** |
| DEV-Indoors (ours) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | **I** | **S** |
| DEV-Reals (ours) | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | **I** | **R** |

Tab. 7 presents a comparison of the prevalent event-centric datasets available today. In this work, we focus on addressing challenges associated with motion blur and lighting variations within indoor settings rather than ground robot navigation or SLAM from a UAV perspective. A pervasive issue with current datasets is the absence of ground truth depth [9, 17, 19, 22, 30, 41, 73, 78] or mesh data [1, 15, 33, 80], which are essential for the operation and evaluation of NeRF-based SLAM methods. In addition, many outdoor datasets are geared towards large-scale navigation [1, 17, 33, 73] and lack significant motion blur and lighting variation, making them unsuitable for our intended purposes. Besides, most datasets are synthetic [1, 41, 78], which are not representative of real-world scenarios or provide sample motion [5, 41]. To address the existing limitations, we introduce the synthetic dataset DEV-Reals and DEV-Indoors, which consist of 6 scenes and 17 sequences with practical motion blur and lighting changes.

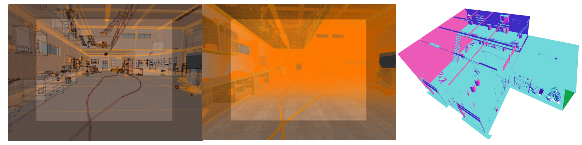



Figure 11. **The ground truth mesh generation process** of `#workshop` in DEV-Indoors dataset.

**Scene Assets of DEV-Indoors.** We use the Blender [7] to construct the synthetic DEV-Indoors dataset, including three high-quality models: `#Room`, `#Apartment` and `#Workshop`. Fig. 13 illustrates the blender models and corresponding camera trajectories. Unlike the camera motion on the Replica dataset [58], our camera trajectory is six degrees of freedom (6-DOF), and the motion is highly complex. The camera trajectory is obtained through manual manipulation of position and orientation and further refined through smoothing operations.

**Event Data Generation.** The simulated event data in DEV-indoors are obtained via the following three steps: first, we render high-quality RGB captures covering norm, motion blur, and dark scenarios by varying the scene lighting and camera exposure time. Second, we perform a video frame interpolation algorithm FILM [53] to convert the rendered images into ultra-high frequency RGB frames. Finally, We use the event camera simulator [16] to generate synthetic event data.


Figure 12. **Extra virtual views** of `#Room`, `#Apartment` and `#Workshop` models in DEV-Indoors dataset.

**Ground Truth Mesh.** As shown in Fig. 11, to obtain a dense mesh that can apply to algorithm reconstruction, we perform detailed and dense triangulation on the models and use the sampling algorithm of Open3D [1] to uniformly sample them to avoid points gathering on the surface of small objects. Then, we further use the mesh culling in [66] to remove the unseen vertices of the models. This process ultimately yields a high-quality mesh that can be used for evaluation. Note that although Blender can directly export point cloud files in PLY format, they cannot be directly used

[1] open3d.geometry.simplify_vertex_clustering

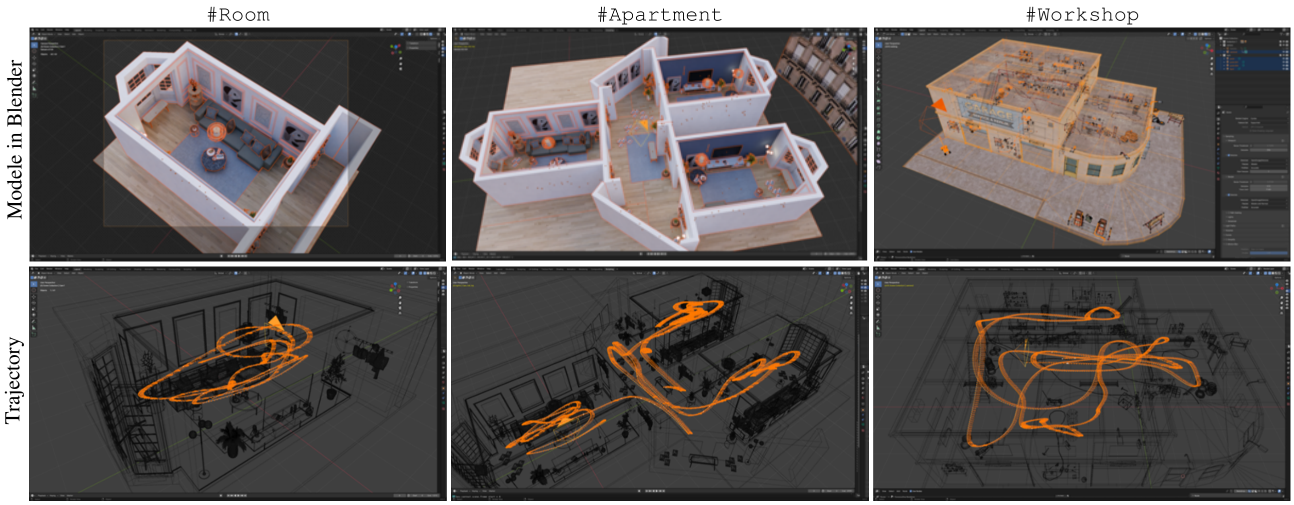



Figure 13. **The models and trajectories** of the DEV-Indoors dataset in Blender [7], including `#room`, `#apartment`, and `#workshop`.

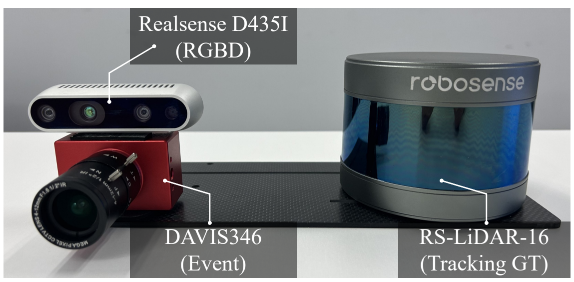



Figure 14. Illustration of the DEV-Reals capture configuration.

for reconstruction evaluation. The reason is that the models created in Blender are highly structured and sparsely connected, where a face may only be covered by a few vertices.

**Evaluation Datasets.** To construct the evaluation subsets, we use frustum + occlusion + virtual cameras that introduce extra virtual views to cover the occluded parts inside the region of interest in CoSLAM [66]. The evaluation datasets are generated by randomly conducting 2000 poses and depths in Blender for each scene. We further manually add extra virtual views to cover all scenes, as shown in Fig. 12. This process helps to evaluate the view synthesis and hole-filling capabilities of the algorithm.

**Dataset Sequence Visualization.** We show the visualization details in Fig. 18, including 9 subsets: #Room Norm, #Room Blur, #Room Dark, #Apartment Norm, #Apartment Blur, #Apartment Dark, #Workshop Norm, #Workshop Blur, and #Workshop Dark, with corresponding RGB frames, event data, and depth images.

## 8. Configurations of DEV-Reals dataset

**Capture System.** As shown in Fig. 14, our capture system comprises a LiDAR (for ground truth pose), a Realsense D435I RGBD camera, and a DAVIS346 event camera. Besides, we report the hardware specifications of our capture system in Tab. 8. All data sequences are recorded on a PC running Ubuntu 18.04 LTS on an Intel Core i7 CPU. We use the Kalibr toolkit to calibrate the extrinsic parameters between IMUs of DAVIS346 and Realsense D435I. The ground truth trajectories are obtained using the advanced implementation of LOAM [74] algorithm. Time calibration across all sensors is synchronized to a **millisecond level**, and spatial calibration accuracy is in **millimeters level**.

Table 8. Capture System Sensors Specifications of DEV-Reals.

| Sensors | Rate / Bandwidth | Specifications |
|---|---|---|
| Realsense D435I | 90 / 30 fps | $1920 \times 1080$ pixels,<br>Depth: 69°H / 42°V, Stereoscopic,<br>RGB: 87°H / 58°V, Rolling Shutter. |
| DAVIS346 | 12 MEvents / s | $346 \times 260$ pixels,<br>DVS: 120 dB, APS: 56.7 dB,<br>f/2.1-12, FoV: 125°D / 97.7°V. |
| RS-LiDAR-16 | 10 hz | 6 DoF ground truth trajectory. |

**Dataset Sequence Visualization.** The dataset is captured in three challenging scenarios: #Pioffice, #Garage, and #Dormitory by changing the lighting conditions and camera movement speed in the environment. We report the visualization details in Fig. 19, including 8 subsets: #Pioffice1, #Pioffice2, #Garage1, #Garage2, #Dormitory1, #Dormitory2, #Dormitory3 and #Dormitory4, with corresponding RGB frames, event data, and depth images. Compared with the synthetic DEV-Indoors dataset, the DEV-Reals dataset is more challenging and realistic, containing depth and event noise, which is more suitable for evaluating the robustness of the algorithm.

## 9. Additional implementation details

**Hyperparameters.** EN-SLAM run 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 $\lambda_{ev} =$

Table 9. **Tracking (RSME)** and **run-time** comparison with detailed iteration setting on **DEV-Indoors** dataset. Our method outperforms previous works in both accuracy and efficiency in most subsets, demonstrating its robustness under motion blur and luminance variation.

| Method | Metric | #Rm **norm** | #Rm **blur** | #Rm **dark** | #Apt **norm** | #Apt **blur** | #Apt **dark** | #Wkp **norm** | #Wkp **blur** | #Wkp **dark** | #all **avg** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| iMAP [60] | ATE RMSE (cm) | 41.08 | 50.58 | 70.77 | 25.75 | 14.41 | $1.06e^5$ | 276.91 | 891.86 | 345.21 | 214.57 |
| | Tracking (ms) ↑ | 24.72×50 | 24.66×50 | 24.66×50 | 24.76×50 | 24.79×50 | 24.70×50 | 24.78×50 | 24.75×50 | 24.73×50 | 24.73×50 |
| | Mapping (ms) ↑ | 45.97×300 | 45.72×300 | 45.68×300 | 45.33×300 | 45.50×300 | 45.34×300 | 45.34×300 | 45.94×300 | 45.76×300 | 41.18×300 |
| | FPS ↑ | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 | 0.07 |
| NICE-SLAM [81] | ATE RMSE (cm) | 17.06 | 29.54 | 30.53 | 25.17 | 44.22 | 48.28 | ✗94 % | ✗33 % | ✗33% | 32.47 |
| | Tracking (ms) ↑ | 7.70×10 | 7.31×10 | 7.44×10 | 5.88×20 | 5.82×20 | 5.93×20 | 5.88×20 | 5.91×20 | 5.89×20 | 6.46×16 |
| | Mapping (ms) ↑ | 27.65×120 | 26.31×120 | 26.45×120 | 26.10×120 | 25.43×120 | 26.53×120 | 26.07×120 | 26.65×120 | 26.59×120 | 26.42×120 |
| | FPS ↑ | 0.30 | 0.32 | 0.32 | 0.32 | 0.33 | 0.31 | 0.31 | 0.31 | 0.31 | 0.31 |
| CoSLAM [66] | ATE RMSE (cm) | 10.71 | 10.88 | 26.64 | 10.02 | 13.03 | 30.75 | 7.96 | 14.37 | 17.88 | 15.80 |
| | Tracking (ms) ↑ | 5.46×15 | 5.50×15 | 5.39×15 | 5.51×15 | 5.09×15 | 5.15×15 | 7.55×15 | 7.48×15 | 7.62×15 | 6.08×15 |
| | Mapping (ms) ↑ | 9.76×15 | 12.84×15 | 12.38×15 | 11.29×15 | 11.47×15 | 14.07×15 | 16.63×15 | 16.61×15 | 16.65×15 | 13.52×15 |
| | FPS ↑ | 12.22 | 12.12 | 12.37 | 12.09 | 13.11 | 12.95 | 8.83 | 8.91 | 8.75 | 11.26 |
| ESLAM [26] | ATE RMSE (cm) | 10.72 | 15.55 | 40.42 | 9.99 | 12.79 | 12.39 | 7.01 | 15.07 | 7.97 | 14.66 |
| | Tracking (ms) ↑ | 2.90×10 | 5.34×10 | 5.30×10 | 5.18×15 | 5.16×15 | 5.30×15 | 5.33×15 | 5.22×15 | 5.32×15 | 5.20×13 |
| | Mapping (ms) ↑ | 17.95×10 | 17.83×10 | 18.20×10 | 15.00×10 | 15.02×10 | 15.11×10 | 17.07×10 | 17.02×10 | 16.92×10 | 16.68×10 |
| | FPS ↑ | **19.18** | **18.71** | **18.86** | 12.87 | 12.92 | 12.58 | 12.51 | 12.76 | 12.53 | 14.77 |
| Ours | Acc (cm) ↓ | **9.62** | **9.72** | **9.94** | **8.62** | **8.77** | **9.21** | **6.74** | **7.51** | **6.94** | **8.56** |
| | Tracking (ms) ↑ | 5.64×10 | 5.83×10 | 5.65×10 | 5.76×10 | 5.69×10 | 5.77×10 | 5.96×10 | 5.80×10 | 5.63×10 | 5.75×10 |
| | Mapping (ms) ↑ | 13.02×10 | 13.33×10 | 13.07×10 | 13.16×10 | 13.23×10 | 13.21×10 | 13.36×10 | 13.04×10 | 12.98×10 | 13.16×10 |
| | FPS ↑ | 17.71 | 17.15 | 17.71 | **17.37** | **17.59** | **17.34** | **16.77** | **17.23** | **17.76** | **17.40** |

$0.05, \lambda_{rgb} = 5.0, \lambda_d = 0.1, \lambda_{sdf} = 1000.0, \lambda_{sf} = 10$. The adaptive event forward query window $w_d$ and neighborhood window $w_k$ are set as 10 and 5 in DEV-Indoors, DEV-Indoors, and the fast subsets of DEV-Reals. Loss threshold $\mathcal{L}_s$ is set as 0.08 by default and 0.1 for DEV-Reals. The patch size of probability-weighted sampling is set as $32 \times 32$ for both RGB and event cameras. The event threshold $C$ is set as 0.2 for the synthetic DEV-Indoors dataset and performs a normalization for real datasets DEV-Reals and Vector. For the camera distortion, we do not perform a pixel-wised undistortion but remove the distortion for each ray of both the RGBD camera and event camera.

We use Realsense RGB frames in DEV-Real for higher resolution compared to DAVIS. The pseudo-exposure is a **equivalent exposure time** of the event CRF rendering model. EN-SLAM renders logarithmic brightness in Eqs. (12) and (13) at $t_\alpha$ and $t_\beta$ rather than all events between $t_\alpha$ and $t_\beta$. Thus, we do not focus on the intrinsic exposure of the event camera but on the equivalent exposure time for volume rendering and training.

For DEV-Reals capture, we enable the auto-exposure to obtain a suitable exposure time and fixed it in a constant, *i.e.*, 7.5 ms for normal scenes and 30 ms for the dark, to ensure the data match the algorithm inputs and support the validation. However, we enable the auto-gain and model the differentiable ISP through neural networks, as mentioned in Sec. 3.2 and [24, 61].

# 10. Additional Experimental Results

## 10.1. More Ablation Studies

**Effect of the Event Temporal Aggregating Optimization Strategy.** To evaluate the effect of each component of the event temporal aggregating optimization strategy (ETA), we conduct an ablation study on the #Rm blur subset of DEV-Indoors and #Dorm2 subset of DEV-Reals. We investigate the performance using a constant interval of 5 frames and 10 frames for forward query, as well as utilize the proposed adaptive query in Tab. 10. The results show that the query interval is critical for EN-SLAM. The adaptive query strategy can significantly reduce the tracking ATE by 1.5 cm on #Rm blur and 16.08 cm on #Dorm2, compared with the constant query interval of 5, respectively. In addition, the implementation with #10 interval is better than #5 interval by providing a longer time window constraint for the event temporal aggregating optimization, but still worse than the adaptive query strategy. The reason is that the event temporal aggregating optimization is sensitive, and the adaptive query strategy can adaptively select events to participate in optimization based on the loss, providing more robust local constraints thus reducing the impact of noise on optimization. Besides, Tab. 10 also shows that the full model surpasses the model w/o PWS by 0.25 and 1.9% in ATE and completion on #Rm blur. For the effectiveness of ETA, our full model achieves lower tracking errors of 9.61 and 15.47 than the model w/o ETA on the #Rm blur and #Dorm2, respectively.

Table 10. **Ablation study of ETA** on the #Rm blur and #Drom2 subset of DEV-Indoors and DEV-Reals (15 iterations).

| Setting | #Rm blur ATE↓ | #Rm blur ACC↓ | #Rm blur Comp↓ | #Rm blur Comp ratio↑ | #Dorm2 Median↓ | #Dorm2 RSME↓ |
|---|---|---|---|---|---|---|
| Forward Query #5 | 11.11 | 8.54 | 8.51 | 83.21 | 27.99 | 28.99 |
| Forward Query #0 | 10.45 | 8.23 | 8.60 | 82.62 | 12.50 | 14.15 |
| w/o PWS | 9.86 | 7.88 | 9.49 | 81.04 | 16.59 | 19.78 |
| w/o ETA | 11.89 | 8.61 | 10.98 | 76.31 | 14.46 | 18.75 |
| Full ETA | **9.61** | **7.88** | **7.59** | **83.51** | **11.91** | **15.47** |

## 10.2. More Detailed Tracking Comparison

In this section, we further provide the accuracy of tracking and its corresponding iteration settings, as well as the runtime. Note that it is unrealistic to strictly control all the iterations or FPS to be the same. Therefore, all the methods are compared under similar runtimes. Besides, we must emphasize that we had to increase the iteration number for certain methods to avoid crashes. Nevertheless, EN-SLAM still achieves superior accuracy with less time-consuming.

**Tracking Comparison on DEV-Indoors.** We provide the detailed iterations and corresponding FPS of the tracking evaluation on the DEV-Indoors dataset in Tab. 9. The re-

Table 11. **Tracking (ATE median [cm]) and run-time comparison** with detailed iteration setting of the proposed method vs. the SOTA methods on **DEV-Reals**. Our method achieves better performance in comparison to NICE-SLAM [81], CoSLAM [66] and ESLAM [26].

| Method | Metric | #Pio1 | #Pio2 | #Gre1 | #Gre2 | #dorm1 | #dorm2 | #dorm3 | #dorm4 | **#avg** |
|---|---|---|---|---|---|---|---|---|---|---|
| NICE-SLAM [81] | ATE RMSE (cm) ↓ | 13.21 | 23.35 | ✗63% | ✗25% | 24.69 | **10.68** | 18.44 | 44.04 | ✗22.40 |
| | Tracking (ms) ↑ | 3.08×100 | 3.61×100 | ✗×100 | ✗×100 | 3.08×100 | 3.15×100 | 3.18×100 | 3.17×100 | 3.21×100 |
| | Mapping (ms) ↑ | 2.97×60 | 2.57×60 | ✗×60 | ✗×60 | 3.86×60 | 3.97 ×60 | 3.27×60 | 3.20×60 | 3.31×60 |
| | FPS ↑ | 0.28 | 0.28 | ✗ | ✗ | 0.31 | 0.32 | 0.32 | 0.32 | 0.31 |
| CoSLAM [66] | ATE RMSE (cm) ↓ | 11.14 | 19.83 | 82.52 | 40.16 | 15.99 | 15.42 | 30.12 | 32.45 | 30.95 |
| | Tracking (ms) ↑ | 8.87×20 | 8.90×20 | 8.96×20 | 8.89×20 | 8.87×20 | 9.09×20 | 9.03×20 | 9.08×20 | 8.96×20 |
| | Mapping (ms) ↑ | 14.86 × 20 | 14.84 × 20 | 14.97 × 20 | 14.71 × 20 | 15.33 × 20 | 14.83 × 20 | 16.09 × 20 | 15.41×20 | 15.13×20 |
| | FPS ↑ | 5.64 | 5.62 | 5.58 | 5.63 | 5.64 | 5.50 | 5.54 | 5.51 | 5.58 |
| ESLAM [26] | ATE RMSE (cm) ↓ | 11.28 | 21.42 | 63.65 | 30.75 | 37.94 | 31.04 | 16.19 | 37.91 | 31.27 |
| | Tracking (ms) ↑ | 5.11 × 20 | 5.15 × 20 | 5.08 × 20 | 5.16 × 20 | 4.84 × 20 | 4.93 × 20 | 4.92 × 20 | 4.84 × 20 | 5.00×20 |
| | Mapping (ms) ↑ | 17.85× 20 | 17.6× 20 | 17.4× 20 | 18.4× 20 | 17.× 20 | 19.05× 20 | 16.2× 20 | 16.46 × 20 | 17.50×20 |
| | FPS ↑ | 9.76 | 9.70 | 9.83 | 9.68 | 10.31 | 10.13 | 10.15 | 10.33 | 9.99 |
| ENSLAM (Ours) | ATE RMSE (cm) ↓ | **8.94** | **19.05** | **43.63** | **21.18** | **11.26** | 11.91 | **16.00** | **19.78** | **18.97** |
| | Tracking (ms) ↑ | 5.75×15 | 5.88×15 | 5.59×15 | 5.91×15 | 5.34×15 | 5.78×15 | 5.77×15 | 6.44×15 | 5.81×15 |
| | Mapping (ms) ↑ | 14.00×15 | 14.70×15 | 14.97×15 | 14.23×15 | 14.90×15 | 13.79×15 | 14.35×15 | 15.32×15 | 14.53×15 |
| | FPS ↑ | **11.59** | **11.33** | **11.92** | **11.28** | **12.48** | **11.53** | **11.55** | **10.35** | **11.50** |

Table 12. **Tracking (ATE mean [cm])** with detailed iteration setting of the proposed method vs. the SOTA NeRF-based methods on **Vector**[15] dataset. EN-SLAM achieves better accuracy and efficiency compared with CoSLAM [66] and ESLAM [26] in most scenes.

| Method | Metric | robot norm | robot fast | desk norm | desk fast | sofa norm | sofa fast | hdr norm | hdr fast | **#all avg** |
|---|---|---|---|---|---|---|---|---|---|---|
| CoSLAM [66] | ATE RMSE (cm) ↓ | **1.00** | 124.69 | **1.76** | 97.65 | **1.74** | 77.89 | 1.47 | 1.42 | 38.45 |
| | Tracking (ms) ↑ | 59.74 × 10 | 5.99 × 10 | 5.51 × 10 | 5.67 × 10 | 5.55 × 10 | 5.47 × 10 | 5.55 × 10 | 5.80 × 10 | 5.69 |
| | Mapping (ms) ↑ | 11.44 × 10 | 11.18 × 10 | 10.41 × 10 | 11.18 × 10 | 12.12 × 10 | 16.90 × 10 | 14.32 × 10 | 11.15 × 10 | 12.34 |
| | FPS ↑ | 16.74 | 16.69 | 18.16 | 17.63 | 18.02 | 18.29 | 18.03 | 17.24 | 17.60 |
| ESLAM [26] | ATE RMSE (cm) ↓ | 1.39 | 3.30 | 2.54 | 3.64 | 7.99 | 19.03 | 7.38 | 12.23 | 7.19 |
| | Tracking (ms) ↑ | 4.94 × 20 | 4.96 × 20 | 4.96 × 20 | 4.67 × 20 | 4.85 × 20 | 5.00 × 20 | 5.10 × 20 | 4.91 × 20 | 4.93 × 20 |
| | Mapping (ms) ↑ | 18.68×20 | 19.49×20 | 17.07×20 | 18.69×20 | 17.97×20 | 17.57×20 | 18.16×20 | 18.08×20 | 18.22 × 20 |
| | FPS ↑ | 10.11 | 10.06 | 10.07 | 10.69 | 10.30 | 9.98 | 9.79 | 10.16 | 10.15 |
| ENSLAM (Ours) | ATE RMSE (cm) ↓ | 1.06 | **1.73** | **1.76** | **2.69** | 2.02 | **1.84** | **1.03** | **1.22** | **1.67** |
| | Tracking (ms) ↑ | 5.58 × 10 | 5.91 × 10 | 5.81 × 10 | 6.01 × 10 | 5.74 × 10 | 6.01 × 10 | 5.76 × 10 | 6.12 × 10 | 5.87 |
| | Mapping (ms) ↑ | 19.05 × 10 | 17.07 × 10 | 18.05 × 10 | 16.28 × 10 | 13.91 × 10 | 13.22 × 10 | 13.42 × 10 | 13.76 | 15.60 |
| | FPS ↑ | **17.92** | **16.92** | **17.21** | **16.63** | **17.42** | **16.63** | **17.36** | **16.33** | **17.05** |

sults show that our method is more efficient and accurate than existing NeRF-based SLAM methods. Specifically, our method reduces the tracking ATE by 23.9, 7.24, and 6.1 cm, compared with the SOTA methods NICE-SLAM [81], CoSLAM [66] and ESLAM [26], respectively. In addition, all the other methods face significant challenges from #norm subsets to #blur and #dark scenarios, with a serious decline in accuracy. Hence, we must increase the tracking or mapping iteration times for some baselines to avoid crushes but slow down the FPS. In contrast, our method uses the invariant iterations 10 times for both tracking and mapping and maintains fast, robust, and accurate results.

**Tracking Comparison on DEV-Reals.** In the main paper, we only report the final tracking ATE. Hence, we further show the detailed performance with tracking and mapping iterations in Tab. 11. EN-SLAM uses 15 iterations for both tracking and mapping and achieves the best performance in accuracy and efficiency in the challenging DEV-Reals dataset. In contrast, the other methods perform worse with an event larger iteration number.

**Tracking Comparison on Vector.** Tab. 12 illustrates the tracking ATE and iterations on Vector [15] dataset. EN-SLAM, CoSLAM [66] and ESLAM [26] set the iterations as 10, 20 and 10 in both tracking and mapping, respectively. CoSLAM and EN-SLAM perform comparably in the normal subsets, but EN-SLAM significantly surpasses CoSLAM on the fast subsets, benefitting from the high-quality event data.

### 10.3. Additional Reconstruction Visualization

**Reconstruction Visualization on DEV-Indoors.** Fig. 15 provides more mesh reconstruction results in DEV-Indoors dataset. Compared with the other SOTA methods, EN-SLAM significantly reduces the presence of holes and ghosting artifacts in reconstructed scenes under blurry scenarios, achieving higher-quality reconstruction results. Under the challenges of dark scenes, *e.g.*, #Apt Dark, previous methods NICE-SLAM and CoSLAM suffer from the weak supervision of color images, resulting in tracking drift. While EN-SLAM maintains robust and accurate.

**Reconstruction Visualization on DEV-Reals.** Fig. 16 Fig. 16 shows the map reconstruction comparison on the challenging DEV-Reals dataset. NICE-SLAM crushes in the #Garage1 and #Garage2 subsets due to the low-lighting environments. CoSLAM reconstructs all the scenarios but causes significant holes and artifacts in the mapping results. ESLAM performs relatively well in the #Pioffice1 and #Pioffice2 subsets but fails in the low-lighting subsets #Garage1, #Dormitory2, and #Dormitory4 due to the low-quality color and depth images. In contrast, EN-SLAM achieves the best performance in all the subsets, demonstrating its robustness and accuracy in the challenging DEV-Reals dataset.

**Reconstruction Visualization on Vector.** For the Vector dataset, we show the mesh visualization results in Fig. 17. All the methods perform comparably in the normal subsets but on the fast subset. All methods show comparable performance on the normal subset. However, in the fast subset, the performance of CoSLAM notably declines, leading to reconstruction ghosting. While ESLAM maintains consistent performance, it falls short in providing detailed reconstruction. Our method achieves consistently excellent performance under both normal and fast camera movements.

## 11. Videos Demonstration

We provide a video of our proposed method EN-SLAM along with this document. The video compares EN-SLAM with existing state-of-the-art under motion blur and low-lighting environments: `./demo.mp4`.

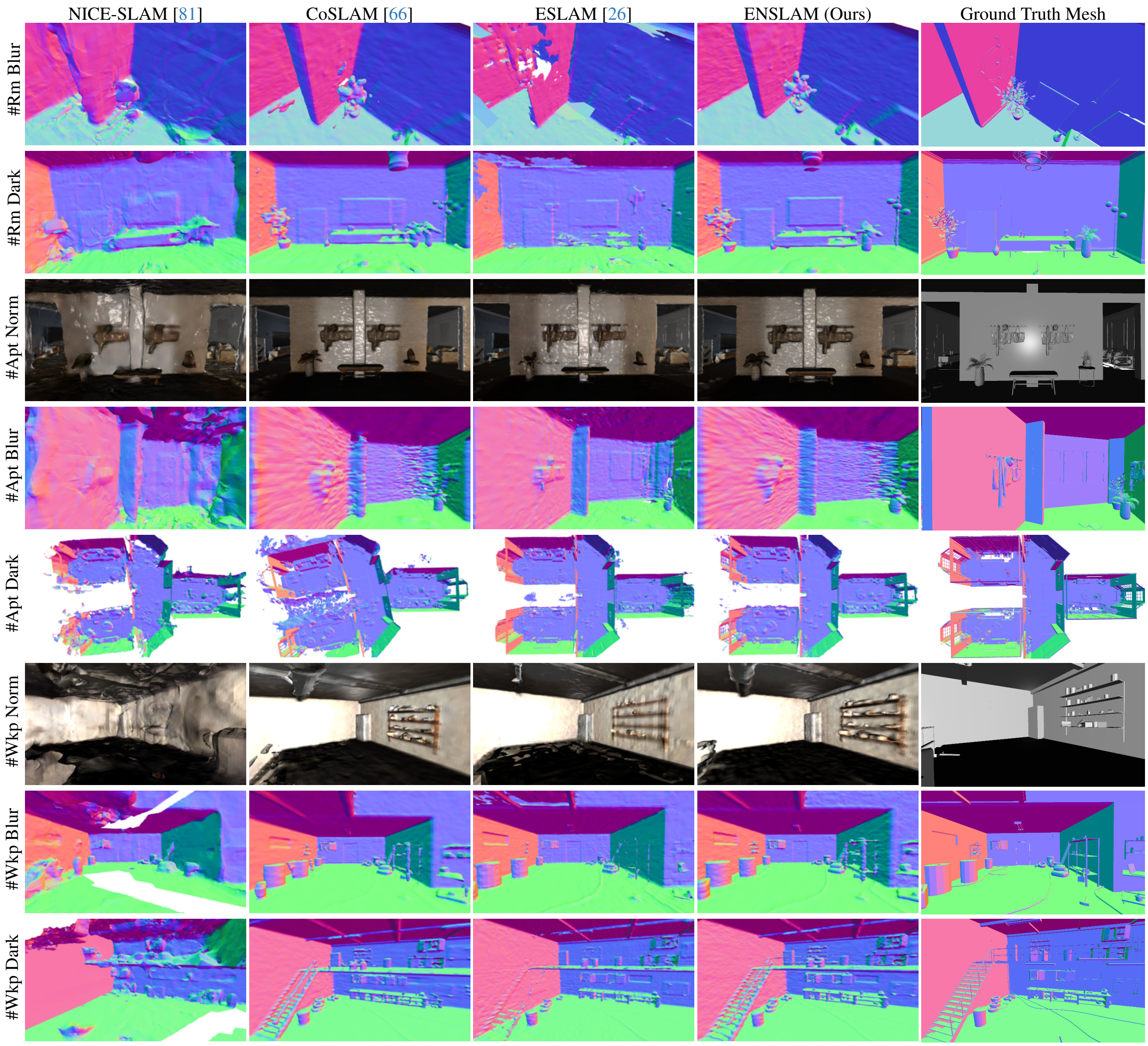



Figure 15. Reconstruction Performance on **DEV-Indoors**. EN-SLAM achieves, on average, more precise reconstruction details than existing methods in motion blur and lighting-varying environments with the assistance of high-quality event streams.

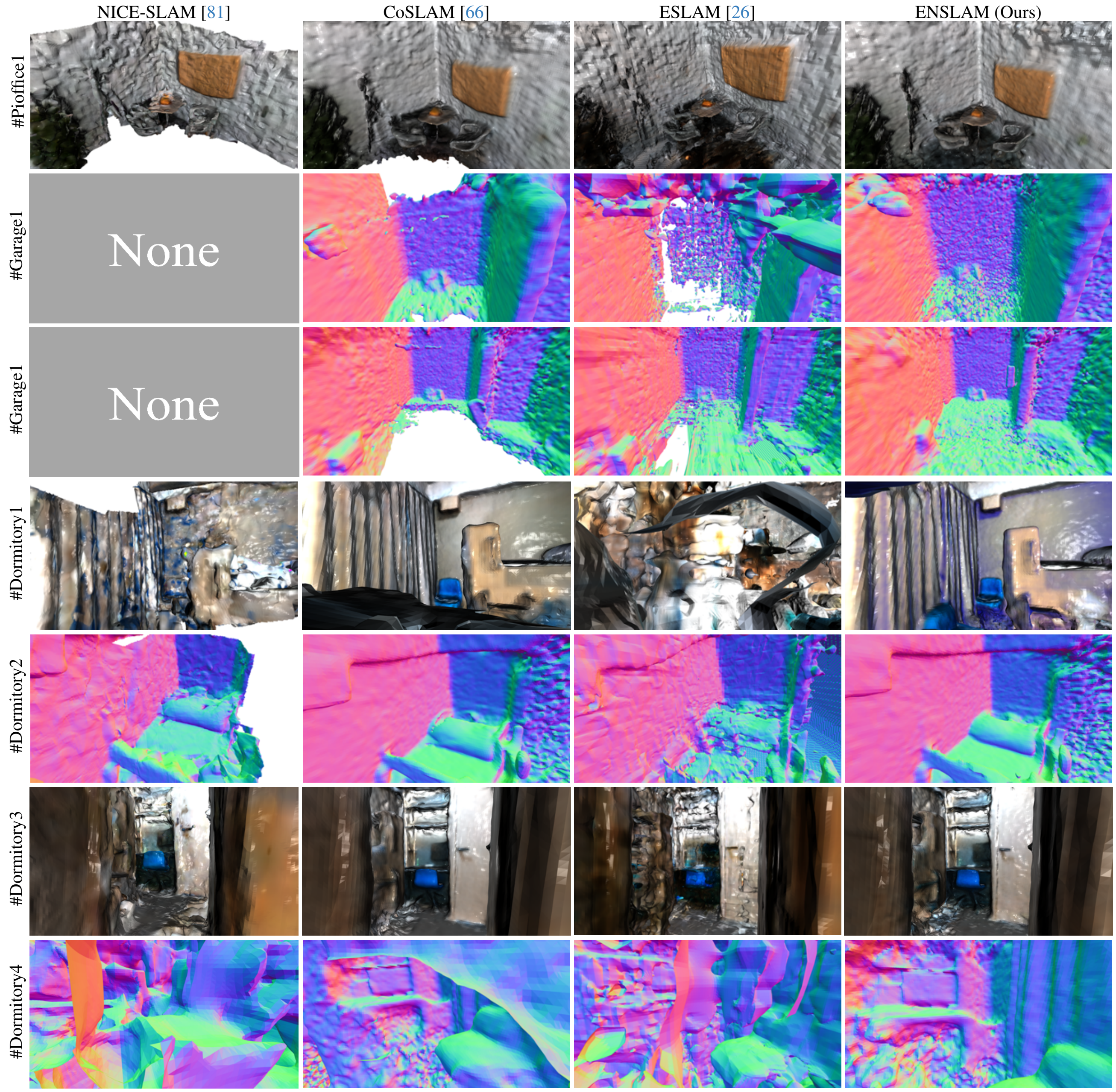



Figure 16. Reconstruction Performance on the challenging **DEV-Reals** dataset. EN-SLAM performs consistently well in all the subsets and obtains more satisfying reconstruction results compared with NICE-SLAM, CoSLAM and ESLAM.

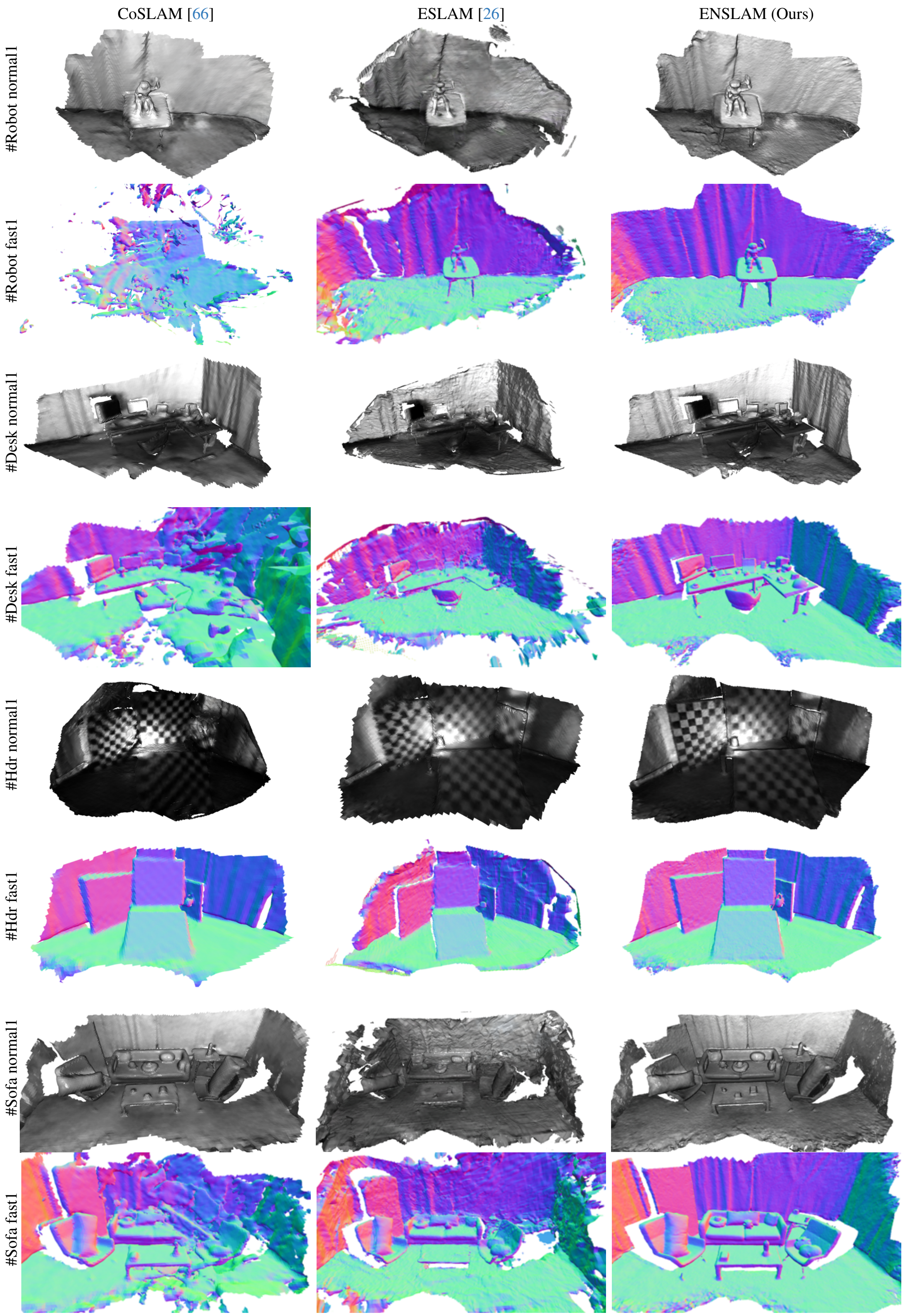



Figure 17. Reconstruction on **Vector**. All the methods perform comparably in normal subsets, but CoSLAM faces challenges in fast subsets, and ESLAM falls short in precise reconstruction. Our method consistently performs better under both normal and fast movements.

Figure 18. **Visualization of the DEV-Indoors dataset**. DEV-Indoors is rendered from Blender models, including 9 subsets containing high-quality color images, depth, meshes, and ground truth trajectories by varying the scene lighting and camera exposure time.

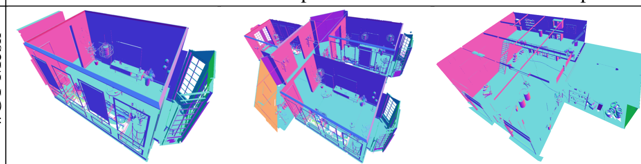
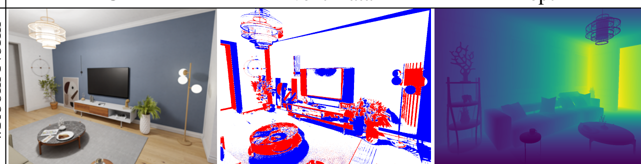
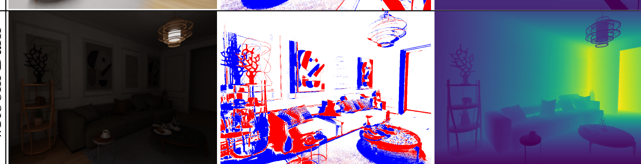
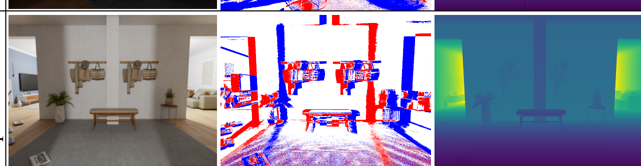
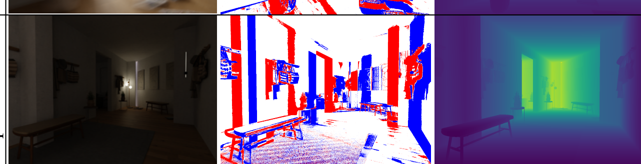
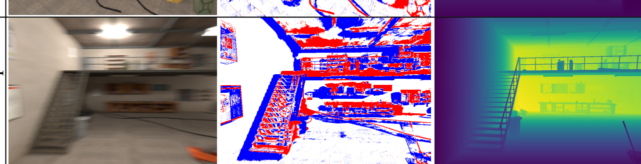
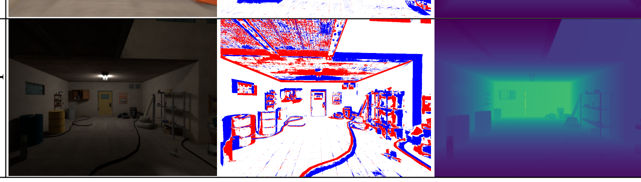
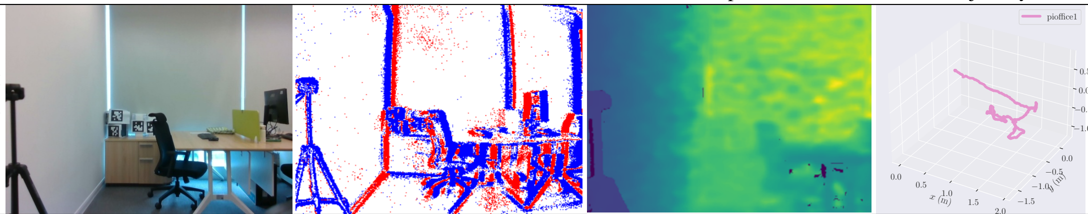
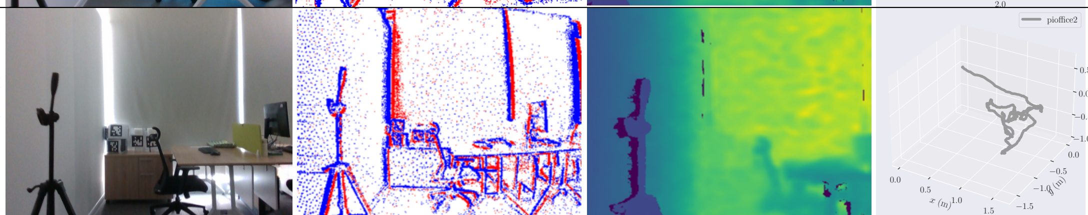
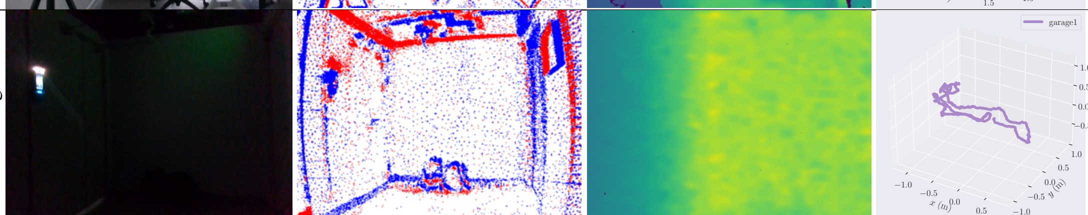
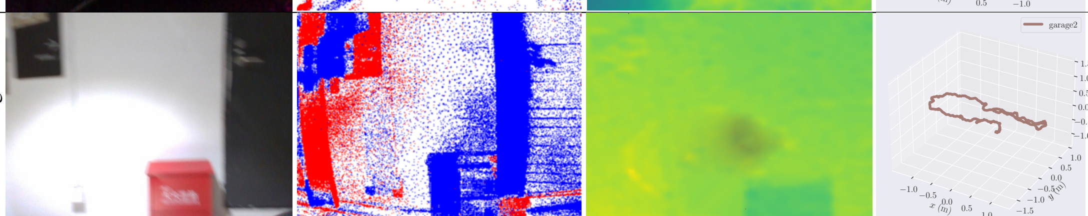
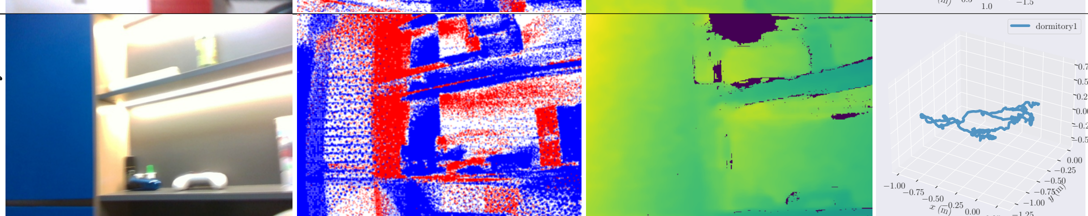
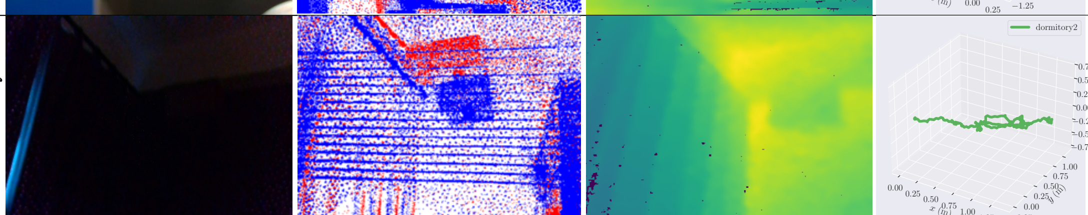
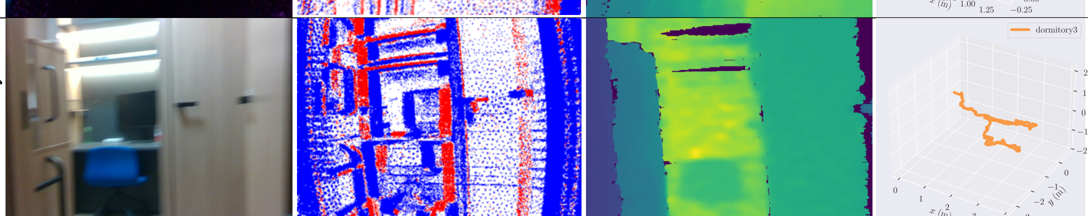
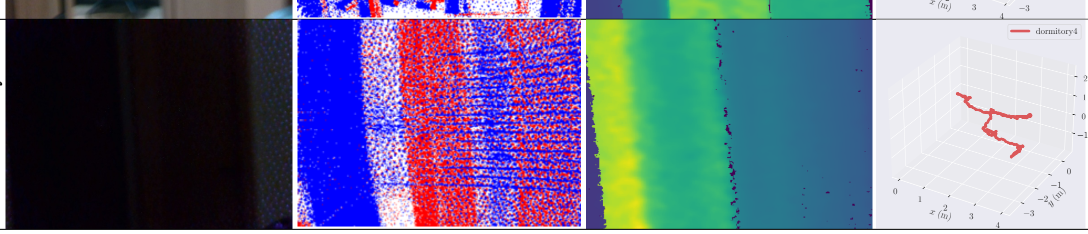
| | #Room | #Apartment | # Workshop | #Length (frame) | #Duration (second) |
|---|---|---|---|---|---|
| #GT Mesh |  | | | — | — |
| | RGB | Event Data | Depth | | |
| #Room Norm |  | | | 1371 | 55 s |
| #Room Blur |  | | | 1371 | 55 s |
| #Room Dark |  | | | 1371 | 55 s |
| #Apartment Norm |  | | | 3000 | 120 s |
| #Apartment Blur |  | | | 3000 | 120 s |
| #Apartment Dark |  | | | 3000 | 120 s |
| #Workshop Norm |  | | | 1800 | 72 s |
| #Workshop Blur |  | | | 1800 | 72 s |
| #Workshop Dark |  | | | 1800 | 72 s |

Figure 19. **Visualization of the DEV-Reals dataset.** DEV-Reals is captured from real scenes: #Piofffice, #Garage, and #Dormitory, providing 8 challenging subsets containing color images, depth, and ground truth trajectories under motion blur and lighting variation.

| | RGB / Event Data / Depth / Trajectory | #Length | #Duration |
|---|---|---|---|
| #Piofffice1 |  | 1209 | 80.6 s |
| #Piofffice2 |  | 1286 | 85.7 s |
| #Garage1 |  | 1384 | 92.7 s |
| #Garage2 |  | 989 | 65.9 s |
| #Dormitory1 |  | 1799 | 119.9 s |
| #Dormitory2 |  | 961 | 64.1 s |
| #Dormitory3 |  | 2726 | 181.7 s |
| #Dormitory4 |  | 1928 | 128.5 s |

# References


[1] Alexander Amini, Tsun-Hsuan Wang, Igor Gilitschenski, Wilko Schwarting, Zhijian Liu, Song Han, Sertac Karaman, and Daniela Rus. Vista 2.0: An open, data-driven simulator for multimodal sensing and policy learning for autonomous vehicles. In *ICRA*. IEEE, 2022.

[2] 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

[3] 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

[4] 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

[5] 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, 1

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

[7] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 6, 1, 2

[8] 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

[9] Jeffrey Delmerico, Titus Cieslewski, Henri Rebecq, Matthias Faessler, and Davide Scaramuzza. Are we ready for autonomous drone racing? the UZH-FPV drone racing dataset. In *ICRA*, 2019.

[10] 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

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

[12] 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

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

[14] 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

[15] 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, 1, 4

[16] 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, 1

[17] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. Dsec: A stereo event camera dataset for driving scenarios. *RA-L*, 2021.

[18] 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

[19] 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, 1

[20] 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

[21] 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

[22] Javier Hidalgo-Carrió, Guillermo Gallego, and Davide Scaramuzza. Event-aided direct sparse odometry. In *CVPR*, pages 5781–5790, 2022. 3, 6, 7, 1

[23] 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

[24] 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, 3

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

[26] 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, 3, 4

[27] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *TOG*, 2023. 2

[28] 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

[29] 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

[30] S Klenk, J Chui, N Demmel, and D Cremers. Tum-vie: The tum stereo visual-inertial event dataset. In *IROS*, 2021.

[31] 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

[32] 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

[33] Alex Junho Lee, Younggun Cho, Young-sik Shin, Ayoung Kim, and Hyun Myung. Vivid++: Vision for visibility dataset. *RA-L*, 7(3):6282–6289, 2022.

[34] 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

[35] 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

[36] Bangyan Liao, Delin Qu, Yifei Xue, Huiqing Zhang, and Yizhen Lao. Revisiting rolling shutter bundle adjustment: Toward accurate and fast solution. In *CVPR*, 2023. 1

[37] 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

[38] 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

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

[40] 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

[41] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbrück, and Davide Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *IJRR*, 36:142 – 149, 2016.

[42] 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

[43] 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

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

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

[46] 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

[47] Delin Qu, Yizhen Lao, Zhigang Wang, Dong Wang, Bin Zhao, and Xuelong Li. Towards nonlinear-motion-aware and occlusion-robust rolling shutter correction. In *ICCV*, 2023. 1

[48] Delin Qu, Bangyan Liao, Huiqing Zhang, Omar Ait-Aider, and Yizhen Lao. Fast rolling shutter correction in the wild. *TPAMI*, 2023. 1

[49] 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

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

[51] 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

[52] 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

[53] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *ECCV*, 2022.

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

[55] 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

[56] 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

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

[58] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Yuheng Ren, Shobhit Verma, Anton Clarkson, Ming Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke Malte Strasdat, Renzo De Nardi, Michael Goesele, S. Lovegrove, and Richard A. Newcombe. The replica dataset: A digital replica of indoor spaces. *ArXiv*, abs/1906.05797, 2019.

[59] 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

[60] 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, 3

[61] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022. 4, 3

[62] 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

[63] 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

[64] 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

[65] 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

[66] 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, 5, 6, 7, 8, 3, 4

[67] 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

[68] 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

[69] 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

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

[71] Chi Yan, Delin Qu, Dong Wang, Dan Xu, Zhigang Wang, Bin Zhao, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *CVPR*, 2024. 2

[72] 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

[73] 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, 1

[74] Ji Zhang and Sanjiv Singh. Loam: Lidar odometry and mapping in real-time. In *RSS*, pages 1–9. Berkeley, CA, 2014.

[75] 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

[76] 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

[77] 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

[78] 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, 1

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

[80] Alex Zihao Zhu, Dinesh Thakur, Tolga Özaslan, Bernd Pfrommer, Vijay R. Kumar, and Kostas Daniilidis. The multivehicle stereo event camera dataset: An event camera dataset for 3d perception. *RA-L*, 3:2032–2039, 2018.

[81] 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, 3, 4

[82] 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