8. Supplementary

In the supplementary, we first provide additional details about the motion field Sec. 8.2. Then, we report the sensitivity analysis of the loss weights (Sec. 8.3) and the detailed analysis of the runtime (Sec. 8.4). We also provide details on the proposed initialization of the scene Gaussians Sec. 8.5 and compare it to existing work on event-based structure-from-motion (SfM) (Sec. 8.6). An example of a failure case due to flickering lights is given in Sec. 8.7. Finally, additional results are provided on dense/sparse depth, flow, and rendered intensity (Sec. 8.8).

8.1. Video

We encourage readers to inspect the attached video, which summarizes the method and the results.

8.2. Motion Field from Depth and Camera Pose

Let us provide further details on the geometric equation Eq. (3). Assuming a stationary scene viewed by a moving camera with linear and angular velocities $\mathbf{V}$ and $\boldsymbol{\omega}$, respectively, the scene depth D can be used to compute the apparent motion on the image plane via the well-known motion field equation

$$\mathbf{v}(\mathbf{x}) = \frac{1}{D(\mathbf{x})} A(\mathbf{x})\mathbf{V} + B(\mathbf{x})\boldsymbol{\omega}, \tag{10}$$

where the quantities are assumed to be given at time t (and omitted, for simplicity of this instantaneous equation). The 2×3 matrices $A(\mathbf{x})$ and $B(\mathbf{x})$ solely depend on the pixel coordinates $\mathbf{x} = (x, y)^\top$:

$$A(\mathbf{x}) = \begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix}, \tag{11}$$

$$B(\mathbf{x}) = \frac{1}{f} \begin{bmatrix} xy & -(f^2 + x^2) & f \cdot y \\ f^2 + y^2 & -xy & -f \cdot x \end{bmatrix}, \tag{12}$$

An alternative way to write Eq. (10) is as the product of a 2×6 matrix (called feature sensitivity matrix, interaction matrix, or image Jacobian matrix for a point feature [53]) and the 6×1 twist given by the camera's generalized velocity $(\mathbf{V}^\top, \boldsymbol{\omega}^\top)^\top$.

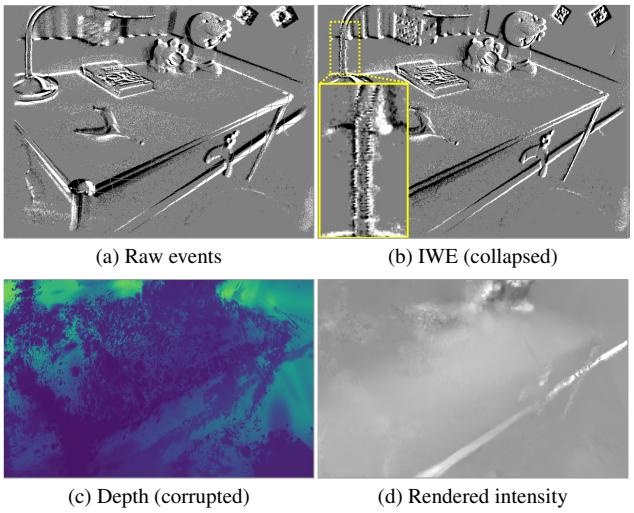
8.3. Sensitivity Analysis

Table 4 reports the sensitivity analysis regarding the loss weights: $\mathcal{L}_c, \mathcal{L}_p$, using the EDS dataset. The metrics are averaged over all five sequences. The results confirm the efficacy of all proposed loss terms in leading to a successful convergence of the GS model.

Notably, we find that *event collapse* (e.g., [40]) occurs with a large weight of the Contrast loss $\mathcal{L}_c$. The collapse is observed in Fig. 7c) as corrupted depth (many small Gaussians with various distances). In Fig. 7b) IWE, the lamp on the desk shows undesired local optima of the warp.

λ_c	λ_s	PSNR ↑	SSIM ↑	LPIPS ↓
0.1	0.1	18.994	0.801	0.389
0.1	1	**19.584**	**0.812**	**0.359**
0.1	10	17.282	0.773	0.423
1	0.1	18.432	0.790	0.398
1	1	19.094	0.805	0.361
1	10	18.593	0.802	0.359
10	0.1	16.666	0.753	0.448
10	1	16.288	0.752	0.436
10	10	16.946	0.770	0.398

Table 4. *Sensitivity analysis of the loss weights.*



(a) Raw events (b) IWE (collapsed)

(c) Depth (corrupted) (d) Rendered intensity

Figure 7. *Examples of corrupted depth for large $\mathcal{L}_c$.*

8.4. Runtime per Each Training Step

In Sec. 4.4, we report the training runtime for the total steps to converge. Here, we provide a detailed runtime analysis of each step in Fig. 8. Larger scenes have more Gaussians (i.e., larger N_g). The runtime of the proposed method scales sub-linearly with the scene size, despite having the warp (i.e., $O(N_e)$) and IWE (i.e., $O(N_e + N_p)$) creation steps. For reference, we also report the render-twice variant of the proposed pipeline. Our pipeline is slightly faster; however, we do not observe any significant differences.

8.5. Initialization

Our method starts from a random distribution of 100k Gaussians. During the initial steps (10k steps out of the total 40k steps), we run the system pipeline in Fig. 2 using $\mathbf{C}(\mathbf{x})$, directly, instead of $\hat{H}(\mathbf{x}; t_{\text{ref}})$ (Eq. (7)). After the initial 10k steps, the Gaussians converge, as shown in Fig. 9 (a). The motivation of the initial steps is to favor initial Gaussians on scene texture and edges, and we find that IWEs produce better initialization than images of pixel-wise accumulation

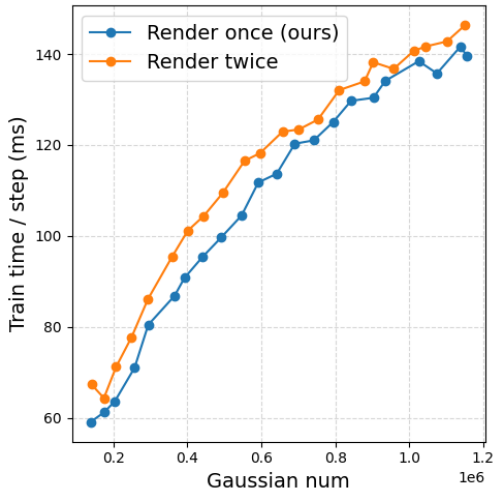
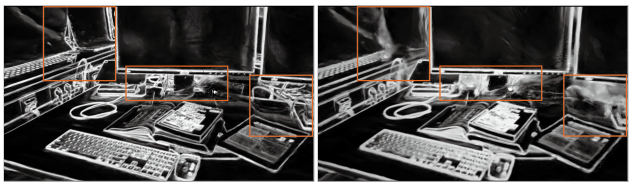


Figure 8. *Detailed analysis on the runtime for different number of Gaussians N_g used to model the scene.*

of events (e.g., Fig. 9 (b)) because of their sharpness, which more concretely determines the location of the centers of the Gaussians.



(a) $\mathbf{C}(\mathbf{x})$ optimized by IWE (b) $\mathbf{C}(\mathbf{x})$ optimized by event frame

Figure 9. IWE-based initialization (a) reconstructs better structure than event-frame–based initialization (b). See the region enclosed by the orange rectangles.

8.6. Comparison with SfM Methods

As discussed in Secs. 3.6 and 8.5, the proposed framework initializes the scene geometry via optimization without polarity information, which has a similar effect as COLMAP, i.e., SfM. Here, using the synthetic dataset from [14], which has accurate ground truth geometry, we now visualize and compare initial point cloud estimation results. For the evaluation of 3D points, we use the Chamfer Distance (CD):

$$\begin{aligned}\mathbf{CD}(X,\hat{X}) =& \frac{1}{|X|}\sum_{x\in X}\min_{\hat{x}\in\hat{X}}\|x-\hat{x}\|_2^2 \\ &+\frac{1}{|\hat{X}|}\sum_{\hat{x}\in\hat{X}}\min_{x\in X}\|x-\hat{x}\|_2^2,\end{aligned} \tag{13}$$

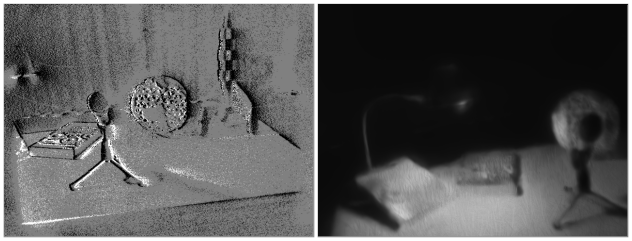
which measures the 3D Euclidean distance between the predicted points $\hat{X}$ and the ground truth (GT) points X.

Figure 11 displays qualitative 3D point estimation results. As baselines, we use the frame-based pipeline ("E2VID [34] + VGGT [60]") and Event-based Multi-view Stereo (EMVS) [58]. Our method consistently recovers fine details of the scene, such as the thin edges of the chair and drum, and the cables of the mic. On the other hand, the event-based baseline, EMVS [58], struggles to recover the entire scene and is limited to the points visible from a small range of viewpoints in the entire trajectory. EMVS is not suitable for the 360-degree trajectory that is typical for the GS and NeRF settings, since the 3D space is represented as voxels (DSIs) with the perspective projection.

Quantitative results are reported in Table 5. Our method achieves the smallest CD among all sequences except for the *hotdog* sequence. "E2VID + VGGT" recovers the *hotdog* sequence the best, possibly due to its simple shape; however, it struggles to estimate correct 3D points for other sequences. The overall results show that the proposed initialization provides more plausible initial geometry than the conventional event-based or event-to-frame SfM methods.

8.7. Failure Cases

As mentioned in Sec. 6, flickering lights are challenging for event-based methods based on brightness constancy. Figure 10 shows a scene from the EDS dataset where many events are generated away from object edges due to flickering lights, which produces noisy reconstructions.



(a) Scene with flicker events (b) Reconstruction

Figure 10. Flickering events produce blurred results.

8.8. Further Qualitative Results

Figure 12 shows further results on depth, flow, and intensity reconstruction using the three datasets.

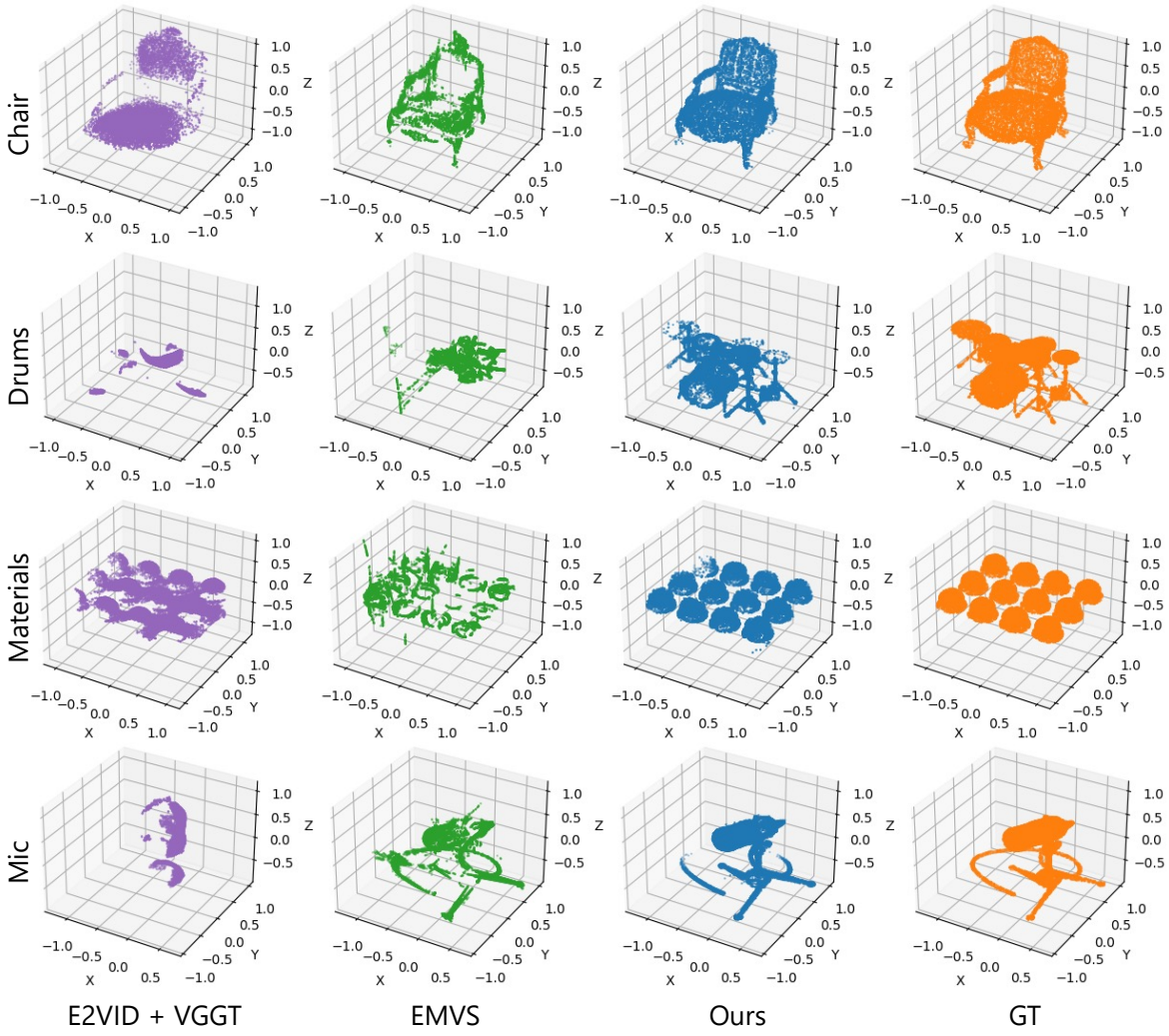
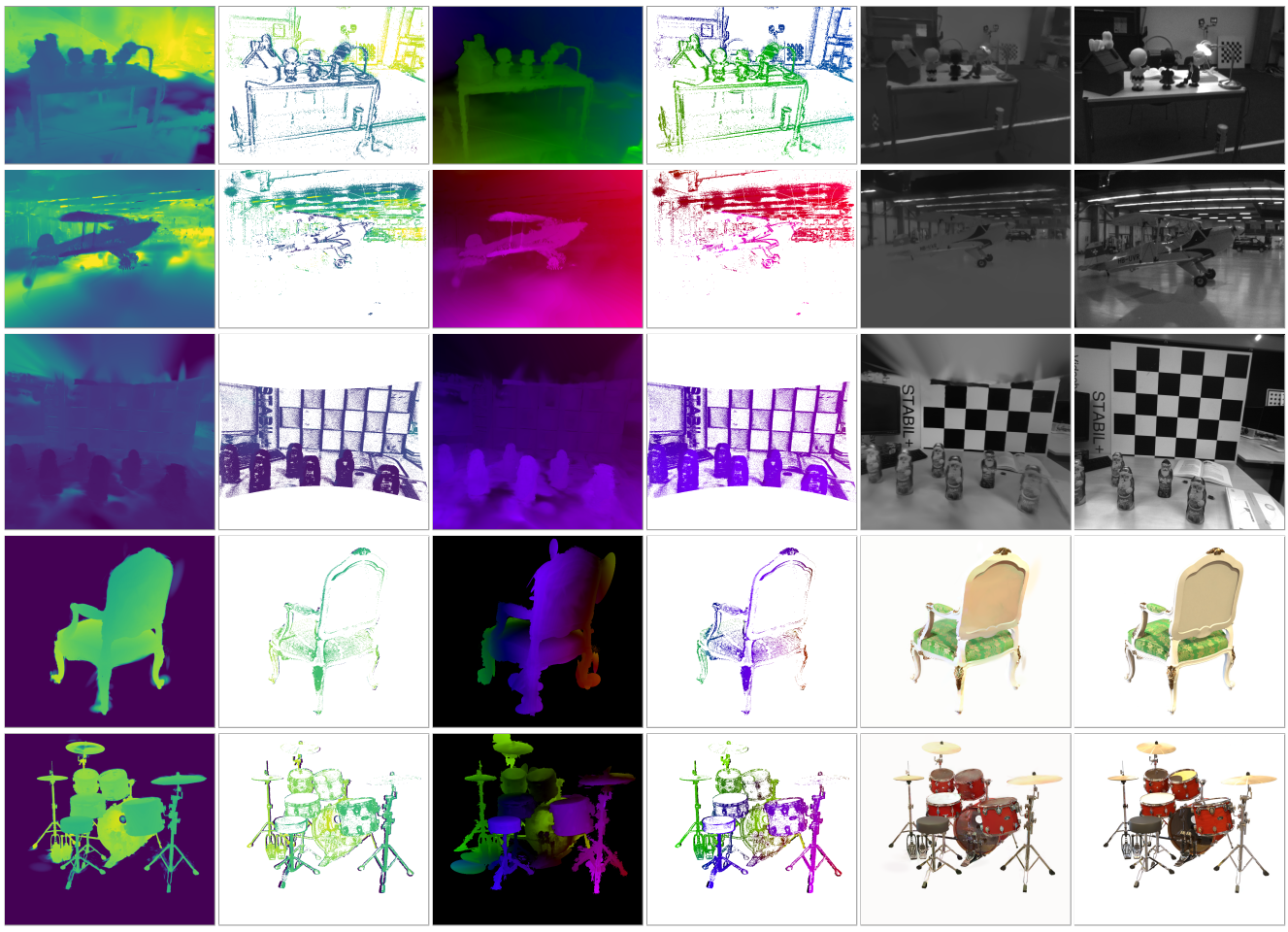


Figure 11. *Results on the point cloud estimation.* For comparison, we use E2VID [34] + VGGT [60] and EMVS [58].

Method	**CD ↓ (Avg.)**	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic
E2VID [34] + VGGT [60]	34.820	25.310	82.340	–	**5.284**	4.568	11.500	79.920
EMVS [58]	35.090	9.757	79.330	7.260	51.530	71.390	18.890	7.490
Ours	**3.559**	**3.127**	**1.204**	**0.949**	11.490	**3.056**	**1.351**	**3.734**

Table 5. *Quantitative results on point cloud estimation using data [14].* The CD is given in mm. "E2VID + VGGT" does not converge on the ficus sequence.



(a) Dense depth (b) Sparse depth (c) Dense flow (d) Sparse flow (e) Rendered intensity (f) GT

Figure 12. Additional depth, flow and intensity reconstruction results on EDS (rows 1 and 2), TUM-VIE (row 3) and color synthetic datasets (rows 4 and 5).