

3D Gaussian splatting From Unposed Spike Streams

Supplementary Material

A. Limitations and Future Works

Although our Nope-SGS demonstrates robust capabilities in novel view synthesis, pose estimation, and reconstruction efficiency, it exhibits some limitations. Firstly, the current approach does not account for dynamic scenes with fast-moving objects, limiting its generality to some extent. Furthermore, all our assumptions are based on the strict continuity of the spike stream. Like most spike-based 3D reconstruction methods, this approach struggles to handle discontinuous event sequences. To address these limitations, several promising directions can be explored. In future, we plan to extend Nope-SGS to handle dynamic scenes by incorporating motion-aware neural representations or hybrid spike-frame fusion strategies, enabling robust reconstruction of fast-moving objects. Moreover, we aim to relax the strict continuity assumption by developing spike-based generative methods, such as spike-based diffusion to complete the discontinuous spike stream and improve performance under discontinuous spike streams.

B. Differences With USP-Gaussian

Although both Nope-SGS and USP-Gaussian [1] utilize spike camera data, our method is different from USP-Gaussian fundamentally in both methodology and objectives. Below, we highlight the key distinctions to clarify our novel contributions and emphasize the more challenging nature of the problem we address.

Different task settings: USP-Gaussian aims to mitigate erroneous reconstruction caused by imprecise pose estimation, which is obtained by COLMAP. However, in many practical scenarios, the severely degraded imaging quality of the spike camera prevents even imprecise camera pose estimation using COLMAP. Meanwhile, pose estimation for spike streams introduces significant time consumption. Our Nope-SGS solves these problems by eliminating the dependency of spike-based 3D reconstruction on camera pose priors. **Different methodology:** Unlike USP-Gaussian that employs MLPs for joint scene-pose optimization, our Nope-SGS decouples these processes: the scene representation follows standard 3D Gaussian Splatting (3DGS) while camera poses are explicitly optimized through Lie algebra. This fundamental architectural difference is further accentuated by distinct loss formulations and computational frameworks. Beyond their shared foundation in 3DGS, the two approaches exhibit no meaningful methodological commonality. Our Nope-SGS's significant superiority in efficiency also originates from eschewing MLPs entirely and leveraging the convergence advan-

tages of Lie algebra for pose optimization. **Different general applicability:** USP-Gaussian's designs and experiments are all based on the grayscale spike camera (Spike Camera-001-Gen2 in Fig. 1). **USP-Gaussian cannot process RGGB (Bayer-pattern) spike streams, forcing reliance on grayscale ground truth for evaluation.** On the contrary, Nope-SGS achieves general 3D reconstruction from spike streams by relying solely on fundamental spike emission principles, ensuring compatibility with all formats and future spike camera generations.

C. Why Spike data can be modeled as Bernoulli distribution?

If we assume that spike camera has triggered K times from 0 to t , equation:

$$A(\mathbf{x}, t) = \left(\int_0^t L_C(\mathbf{x}, \tau) d\tau + V_x \right) \bmod \phi, \quad (1)$$

can be rewritten as:

$$A(\mathbf{x}, t) = \left(\int_0^t L_C(\mathbf{x}, \tau) d\tau + V_x \right) - K = A_x(t) + V_x - K \quad (2)$$

$\hat{A}(x, t)$ is the accumulation without reset as defined before. The coding method of the Spike camera ensures $A(\mathbf{x}, t) \in [0, \phi]$ while $\hat{A}(x, t)$ and K is constant (with known t) and V_x following a uniform distribution $V_x \sim U[0, \phi]$, we have $A(\mathbf{x}, t) \sim U[0, \phi]$. If we want to generate a "1" from time t to $t + \mathbb{T}$, We need to ensure:

$$A(\mathbf{x}, t) + \int_0^{\mathbb{T}} L_C(\mathbf{x}, \tau) d\tau \geq \phi \quad (3)$$

That is:

$$A(\mathbf{x}, t) \geq \phi - \int_t^{t+\mathbb{T}} L_C(\mathbf{x}, \tau) d\tau \quad (4)$$

We assume that there is no overexposure:

$$\int_t^{t+\mathbb{T}} L_C(\mathbf{x}, \tau) d\tau \leq \phi \quad (5)$$

Since $A(\mathbf{x}, t) \sim U[0, \phi]$, the probability of "1" output is $\int_0^{\mathbb{T}} L_C(\mathbf{x}, \tau) d\tau / \phi$ and the probability of "0" output is $1 - \int_t^{t+\mathbb{T}} L_C(\mathbf{x}, \tau) d\tau / \phi$. Obviously, the output of given frame of the spike stream $s(x, y, k)$ can be modeled as a Bernoulli distribution $s(x, y, k) \sim B(1, p(x, y, k))$

D. Lie algebra optimization

We first provide details of sub-frame gradient computation. We use $\frac{\partial \mathcal{L}}{\partial s}$ as an example.

$$C_{sub}(t_1, t_N) = C(t_N) - C(t_1) \quad (6)$$

and:

$$C_i = \sum_{i \in N} c_i(P_i) \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j) \quad (7)$$

So we have:

$$\frac{\partial \mathcal{L}}{\partial T} = \frac{\partial \mathcal{L}}{\partial C_N} \frac{\partial C_N}{\partial T} - \frac{\partial \mathcal{L}}{\partial C_1} \frac{\partial C_1}{\partial T} \quad (8)$$

Following [6] we have:

$$\frac{\partial \mathcal{C}}{\partial T} = K \frac{\partial \mathcal{C}}{\partial \mu_I} \frac{\partial \mu_I}{\partial T} + M \frac{\partial \mathcal{C}}{\partial \Sigma_I} \frac{\partial \Sigma_I}{\partial T} \quad (9)$$

Among them, K and M can be obtained through automatic differentiation. Further we have:

$$\frac{\partial \mu_I}{\partial T} = \frac{\partial \mu_I}{\partial \mu_C} \frac{\mathcal{D} \mu_C}{\mathcal{D} T} \quad (10)$$

and:

$$\frac{\partial \mu_I}{\partial T} = \frac{\partial \mu_I}{\partial \mu_C} \frac{\mathcal{D} \mu_C}{\mathcal{D} T} \quad (11)$$

By using the properties of Lie algebra, we can obtain

$$\frac{\mathcal{D} \mu_C}{\mathcal{D} T} = [I, -\mu_C^\times] \quad (12)$$

E. DETAILS OF SPIKE CAMERA

As illustrated in [5], the exceptional advantage of the spike camera over traditional counterparts lies in its effective retention of temporal changes in light at each sampling position (see Figure 3). By virtue of its sensitivity to a single photon, the spike camera can capture the precise and complete physical process when a photon excites a spike. In contrast, conventional photosensitive devices generate a spike only upon capturing a set of photons, providing a coarse representation of the physical process. Importantly, the time relationship of the physical process is preserved to a significant extent with the spike camera, as opposed to the uniform time relationships imposed at tens of hertz through artificial rules in traditional cameras. Presently, CMOS photosensitive devices, widely employed, exhibit a time sensitivity in the range of tens of nanoseconds. Leveraging the innovative spike stream model, the spike camera achieves high-speed temporal domain sampling at ten million hertz,

facilitating the recording of extremely rapid physical processes. It is acknowledged that everyday vision applications do not demand such a high sampling frequency. The initial chip developed for this purpose features a sampling frequency of 40,000 Hz, surpassing the human vision and traditional camera sampling frequency by a factor of 1000. This chip is well-suited for capturing high-speed rail movements at 350 km/h and the rotation of a hard drive at 7200 rpm.

In practice, the spike camera excels at recording subtle changes in light at various positions within a defined spatial range. Its physical significance is evident, providing a clear representation of the underlying processes.

Therefore, it is expected that spike cameras can be used to generate traditional images and videos. In fact, for any given moment, the scene radiance at each position and the pixel value of each pixel can be estimated from the vit covering that moment, and more detailed scene radiance and pixel value can be estimated by referring to the previous and spatially adjacent vits, thereby obtaining fine images at arbitrary moments. The ability of the spike camera to reconstruct the scene radiance at any moment is called full-time imaging or continuous imaging. Please refer to <https://www.spikevision.cn/> for more details.

Additionally, we shows the difference between Gray-scale spike camera (Spike Camera-001T-Gen2) and color spike camera (Spike M1K40-H4-Gen3) in Fig . 1. Apart from resolution, the main difference between two cameras is that the color spike camera adds a Bayer lens. Therefore, the color spike camera will output Bayer-pattern spike streams. Importantly, both Gray-scale and color spike camera have the same coding method. As a result, our SpikeGS is fully compatible with the Gray-scale spike camera. We also systematically introduce the temporal noise and spatial noise in spike camera according to its unique circuit according to [4].

Spatial Noise Temporal noise is a random variation in the signal that fluctuates over time. The temporal noise in spike camera mainly includes shot noise and thermal noise. The shot noise originates from randomness caused by photon reception and, for pixel (x, y) , the probability that n photons are received between time t and $t + \delta t$ is given by the Poisson probability distribution, i.e.,

$$P(ph(x, y, t) = n) = \frac{\mu_{ph(x, y, t)}}{n! e^{\mu_{ph(x, y, t)}}} \quad (13)$$

where $n \in \mathbb{N}$, $ph(x, y, t)$ is a random variable representing the number of received photons from t to $t + \delta t$, and $\mu_{ph(x, y, t)}$ is the expectation of $ph(x, y, t)$. The random number of photons can affect the luminance intensity. Since δt is enough small, we consider the luminance intensity at time t is proportional to the number of photons between time t and $t + \delta t$, i.e., $L(x, y, t) \propto ph(x, y, t)$. Different from the previous simulator [3] which gets the input current

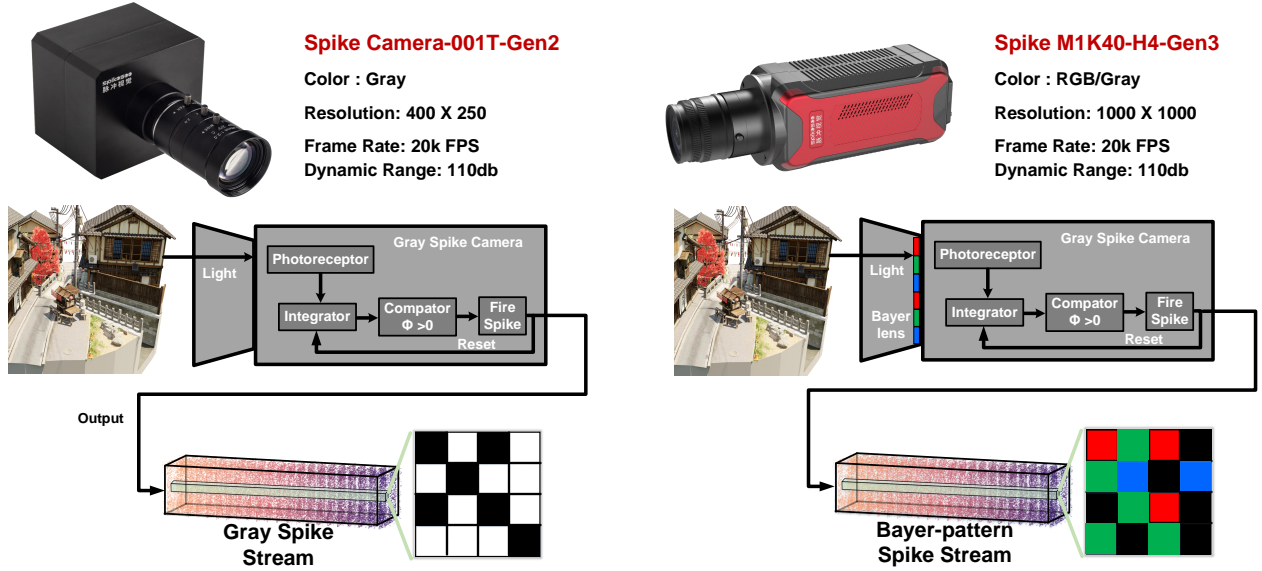


Figure 1. Comparison between Gray-scale spike camera (Spike Camera-001T-Gen2) and color spike camera (Spike M1K40-H4-Gen3). Both Gray-scale and color spike camera have the same coding method. However, different from Gray-scale spike camera, color spike camera's output is Bayer-pattern spike stream.

(see (1)) refer to the ideal luminance intensity $\mu_{L(x,y,t)}$, we first sample $L(x, y, t)$ at time t during simulation, i.e.,

$$L(x, y, t) = \mu_{L(x,y,t)} \frac{ph(x, y, t)}{\mu_{ph(x,y,t)}}. \quad (14)$$

Further, the input current $I_{in}(x, y, t)$ at time t can be expressed as $\alpha L(x, y, \tau)$ where α is the photoelectric conversion rate and it can be estimated through the dynamic range of spike camera (see appendix). The threshold ϕ (as (1)) in the ideal spike camera model also fluctuates over time. We start by writing the ideal threshold in the form of the circuit, i.e.,

$$\phi = CV_d = C(V_D - V_{ref}) \quad (15)$$

The voltage fluctuation in the camera can be caused by the reset transistor, which is affected by temperature where V_D is reset voltage and V_{ref} is reference voltage. As shown in Fig. 2(b), the reset transistor affected by temperature can cause voltage fluctuation. We use the random variable $V^{T_0}(x, y, t)$ to describe fluctuating voltage in pixel (x, y) at time t and $V^{T_0}(x, y, t)$ can be considered to obey Gaussian distribution, i.e.,

$$V^{T_0}(x, y, t) \sim N(0, (\sigma^{T_0})^2), \quad \sigma^{T_0} = \sqrt{\frac{kT_0}{C}}, \quad (16)$$

where σ^{T_0} is the standard deviation of $V^{T_0}(x, y, t)$, k is Boltzmann constant, T_0 is absolute temperature.

Spatial Noise Spatial noise i.e., fixed-pattern noise, is a random variation in the signal that has nothing to do with

time. For the spatial noise in spike camera, we mainly consider dark current, capacitor mismatch and bias voltage. The dark current is from photodiodes and it can generate extra electricity as accumulation in spike camera. The dark current $I_{dark}(x, y)$ in pixel (x, y) obeys Gaussian distribution as,

$$I_{dark}(x, y) \sim N(\mu_{dark}, (\sigma_{dark}^S)^2), \quad (17)$$

where μ_{dark} and σ_{dark}^S contribute to the expectation and variance of $I_{dark}(x, y)$. Besides, capacitance difference in each pixel obeys Gaussian distribution and can be written as,

$$C^S(x, y) \sim N(0, (\sigma_C^S)^2), \quad (18)$$

where $C^S(x, y)$ is the random variable describing the capacitance nonuniformity at pixel (x, y) , σ_C^S is the standard deviation of $C^S(x, y)$. The bias voltage $V^S(x, y)$ mainly comes from reset voltage nonuniformity in check module as shown in Fig. 2(c). We can assume the bias voltage $V^S(x, y)$ at pixel (x, y) obeys Uniform distribution.

F. Details of spike accumulation and spike interval

F.1. Spike Accumulation

According to SpikeGS [2], spike accumulation is defined as:

$$I_{acc}(t_1, t_N) = \phi / N \sum_{t_i} S(P_{t_i}) \quad (19)$$

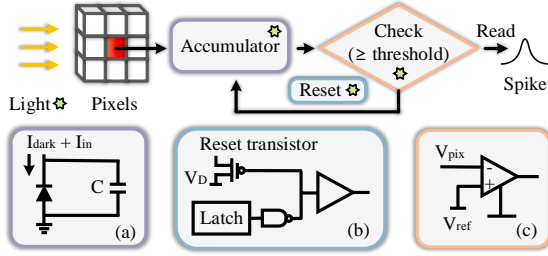


Figure 2. The principle and circuit of spike camera. (a) is the work principle of accumulator in pixel circuit of spike camera. (b) is reset model in pixel circuit of spike camera. (c) is check mechanism in pixel circuit of spike camera. “Star” means that we have considered the corresponding noise.

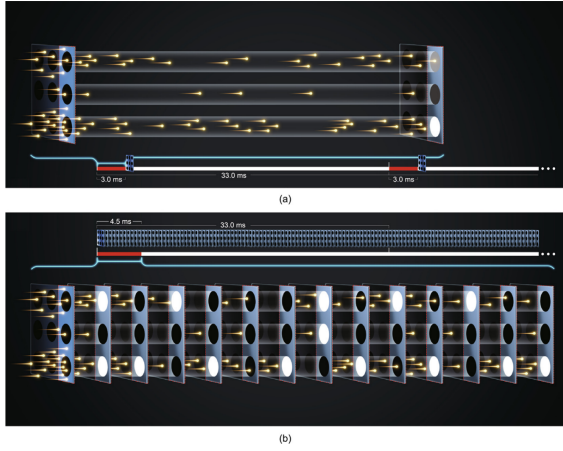


Figure 3. Overall comparison of image and vidar in terms of visual information representation [5].

Where $I_{acc}(t_1, t_N)$ is the result of spike stream accumulation from t_1 to t_N , P_{t_i} is the camera pose at time $t_i \in \{t_1, t_2, \dots, t_N\}$, ϕ is the threshold and $S(P_{t_i})$ is the spike at camera pose P_{t_i} . If the photoelectric conversion process is disregarded, from equation 19 we have:

$$I_{acc}(t_1, t_N) = \phi/N \sum_{t_i} \int_{t_{i-1}}^{t_i} c(P_\tau) d\tau \bmod \phi \quad (20)$$

Where $c(P_\tau)$ is the radiance at camera pose P_τ . Further, we have:

$$I_{acc}(t_1, t_N) \approx 1/N \int_{t_1}^{t_N} c(P_\tau) d\tau \quad (21)$$

To align the accumulation process of the spike stream, we also improved the rasterization of 3DGS to a time accumulation rasterization by simulating the physical accumulation

process of photons:

$$C_{acc}(t_1, t_N) = N^{-1} \int_{t_1}^{t_N} \sum_{i \in N} c_i(P_\tau) \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j) d\tau \quad (22)$$

Considering computability, equation 22 can be approximated by a discrete summation:

$$C_{acc}(t_1, t_N) = N^{-1} \sum_{\tau=t_1}^{t_N} \sum_{i \in N} c_i(P_\tau) \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j) \quad (23)$$

Finally, we employ an accumulation loss to minimize their photometric error, expressed as:

$$\mathcal{L}_{acc} = (1 - \lambda_1) \|C_{acc} - I_{acc}\|_2 + \lambda_1 SSIM(C_{acc}, I_{acc}) \quad (24)$$

F.2. Spike Interval

We use spike intervals and interval loss to initialize our 3D scene, following SpikeGS, spike interval is defined as:

$$I_{in} = \frac{\phi}{t_1 - t_2} \quad (25)$$

Where ϕ represents the threshold. t_1 and t_2 represent the times corresponding to two adjacent trigger spikes ($S(t_1) = S(t_2) = 1$). Rendering results corresponding to the spike intervals can be represented as:

$$C_{in} = \sum_{i \in N} c_i(P_{(t_1+t_2)/2}) \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j) \quad (26)$$

We also employ an interval loss which can be expressed as :

$$\mathcal{L}_{in} = (1 - \lambda_1) \|C_{in} - I_{in}\|_2 + \lambda_1 SSIM(C_{in}, I_{in}) \quad (27)$$

G. Details of Nope-SGS Dataset

To construct our real-world dataset, we deliberately apply rapid, high-amplitude shaking to the spike camera while keeping its optical axis approximately forward-facing. This setup induces strong motion dynamics that closely resemble the challenging conditions present in fast-moving robotic or handheld scenarios. Our dataset features significantly higher resolution and larger spatial extent compared to the real-world dataset provided by USP-Gaussian. A detailed comparison is presented in Tab. 1.

H. Additional Experimental results

The subsequent sections present an in-depth evaluation of camera pose estimation and novel view synthesis, leveraging both the Tanks and Temples and Deblur-NeRF datasets. The details of datasets are listed in Tab. 2

Datasets	Scenes	Type	Pattern	Resolution
USP-Gaussian	3	Indoor/Outdoor	Gray	400*250
Ours	8	Indoor/Outdoor	Gray/RGB	1000*1000

Table 1. Comparison of different real-world dataset.

H.1. Camera Pose Estimation

Supplementary Results. We conduct experiments on 7 scenes from Tanks and Temples and 3 scenes from Deblur-NeRF. The results are meticulously documented in Tab. 3, 4, and 5, which respectively correspond to the metrics Relative Translation Error (RPE_t), Relative Rotation Error (RPE_r), and Absolute Trajectory Error (ATE). As shown in the tables, our method achieves the best performance across all scenes. Notably, it leaves all other methods far behind in terms of RPE_t and RPE_r , demonstrating its superior accuracy in estimating camera poses.

Additional Visualization. We provide qualitative results for camera pose estimation on the Tanks and Temples dataset, as illustrated in Fig. 4. Our method performs well in every scenario. Especially in challenging scenes such as Francis, Horse, and Ignatius, where other methods exhibit significant deviations in estimated camera rotation from the ground truth, our approach consistently aligns closely with the true camera poses.

H.2. Novel View Synthesis

Supplementary Results. Tab. 6, 7 and 8 provide the PSNR, SSIM, and LPIPS results accordingly for novel view synthesis across ten scenes. In terms of PSNR and SSIM, our approach achieves average PSNR gain over 6.58 dB and SSIM gain over 0.16, surpassing all baselines. For LPIPS, our method attains average reduction $> 50\%$.

Additional Visualization. Fig.5 and Fig.6 show the rendered image results on the Tanks and Temples and Deblur-NeRF datasets, respectively. Our SpikeGS creates images that look the closest to the real scenes. Even in tough cases like motion blur, we show clearer details and fewer errors than other methods. These examples prove our approach works better at keeping scenes sharp and consistent.

References

- [1] Kang Chen, Jiyuan Zhang, Zecheng Hao, Yajing Zheng, Tiejun Huang, and Zhaofei Yu. Usp-gaussian: Unifying spike-based image reconstruction, pose correction and gaussian splatting. *arXiv preprint arXiv:2411.10504*, 2024. 1
- [2] Yijia Guo, Liwen Hu, Lei Ma, and Tiejun Huang. Spikegs: Reconstruct 3d scene via fast-moving bio-inspired sensors. 2024. 3
- [3] Liwen Hu, Rui Zhao, Ziluo Ding, Lei Ma, Boxin Shi, Ruiqin Xiong, and Tiejun Huang. Optical flow estimation for spiking camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17844–17853, 2022. 2
- [4] Liwen Hu, Lei Ma, Yijia Guo, and Tiejun Huang. Scsim: A realistic spike cameras simulator. 2024. 2
- [5] Tiejun Huang, Yajing Zheng, Zhaofei Yu, Rui Chen, Yuan Li, Ruiqin Xiong, Lei Ma, Junwei Zhao, Siwei Dong, Lin Zhu, et al. 1000× faster camera and machine vision with ordinary devices. *Engineering*, 25:110–119, 2023. 2, 4
- [6] Hidenobu Matsuki, Riku Murai, Paul HJ Kelly, and Andrew J Davison. Gaussian splatting slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18039–18048, 2024. 2

	Scenes	Type	Seq. length	Frame rate	Pattern
Tanks and Temples	ballroom	indoor	960	40000	RGGB
	barn	outdoor	960	40000	RGGB
	church	indoor	960	40000	RGGB
	family	outdoor	960	40000	RGGB
	francis	outdoor	960	40000	RGGB
	horse	outdoor	960	40000	RGGB
	ignatius	outdoor	960	40000	RGGB
Deblur-NeRF	factory	outdoor	480	40000	RGGB
	pool	outdoor	480	40000	RGGB
	tanabata	outdoor	480	40000	RGGB
USP-Gaussian	sheep	indoor	1280	20000	Grayscale
	keyboard	indoor	1280	20000	Grayscale
	outdoor	outdoor	12800	20000	Grayscale

Table 2. Details of selected scenes.

Datasets	Tanks							Deblur-NeRF		
Methods Scenes	ballroom	barn	church	family	francis	horse	ignatius	factory	pool	tanabata
CF-3DGS	3.550	3.897	3.631	4.349	3.746	3.778	3.674	11.575	10.306	11.638
InstantSplat	0.603	1.549	1.359	0.660	3.046	2.432	2.834	8.574	4.849	12.163
Spikerecon+CF-3DGS	3.742	4.144	3.826	4.648	4.243	3.949	3.639	7.508	11.376	10.336
Spikerecon+Instant	0.311	0.753	0.448	0.283	1.602	0.511	1.001	4.629	1.956	11.125
SpikeGS+Colmap	0.260	1.640	1.240	0.821	2.930	1.701	0.832	3.975	4.627	3.839
Ours	0.213	0.232	0.239	0.203	0.274	0.145	0.301	2.966	1.909	0.577

Table 3. RPE_t : Quantitative comparisons of camera pose estimation on synthetic datasets for 10 scenes. The best performances are highlighted in bold. Our SpikeGS demonstrates the best performance.

Datasets	Tanks							Deblur-NeRF		
Methods Scenes	ballroom	barn	church	family	francis	horse	ignatius	factory	pool	tanabata
CF-3DGS	0.550	0.623	0.156	0.669	1.323	1.134	0.914	0.118	0.501	0.541
InstantSplat	0.141	0.947	0.129	0.146	1.423	1.022	0.769	0.989	0.365	5.507
Spikerecon+CF-3DGS	0.574	0.673	0.168	0.715	1.413	1.207	0.975	0.145	0.450	0.442
Spikerecon+Instant	0.068	0.299	0.041	0.061	0.820	0.096	0.366	0.509	0.236	3.442
SpikeGS+Colmap	0.065	0.833	0.110	0.136	1.549	0.422	0.262	0.022	0.402	0.426
Ours	0.038	0.133	0.037	0.026	0.151	0.039	0.063	0.461	0.226	0.174

Table 4. RPE_r : Quantitative comparisons of camera pose estimation on synthetic datasets for 10 scenes. The best performances are highlighted in bold. Our SpikeGS demonstrates the best performance.

Datasets	Tanks							Deblur-NeRF		
Methods Scenes	ballroom	barn	church	family	francis	horse	ignatius	factory	pool	tanabata
CF-3DGS	0.018	0.045	0.027	0.042	0.093	0.035	0.056	0.028	0.034	0.053
Instantsplat	0.014	0.026	0.027	0.015	0.041	0.099	0.044	0.121	0.064	0.205
Spikerecon+CF-3DGS	0.015	0.053	0.012	0.052	0.070	0.041	0.035	0.050	0.152	0.038
Spikerecon+Instant	0.003	0.013	0.006	0.005	0.020	0.010	0.028	0.044	0.027	0.212
SpikeGS+Colmap	0.003	0.020	0.019	0.010	0.043	0.017	0.020	0.042	0.070	0.053
Ours	0.002	0.002	0.003	0.002	0.002	0.002	0.008	0.069	0.019	0.003

Table 5. **ATE**: Quantitative comparisons of **camera pose estimation** on synthetic datasets for 10 scenes. The best performances are highlighted in **bold**. **Our SpikeGS demonstrates the best performance**.

Datasets	Tanks							Deblur-NeRF		
Methods Scenes	ballroom	barn	church	family	francis	horse	ignatius	factory	pool	tanabata
CF-3DGS	19.750	19.322	22.025	20.974	20.694	20.236	18.720	24.836	24.459	19.030
Instantsplat	20.047	20.975	22.290	22.151	22.737	20.919	19.265	25.161	24.494	16.422
Spikerecon+CF-3DGS	21.458	22.174	24.097	22.716	23.156	21.858	20.470	23.929	21.963	15.648
Spikerecon+Instant	21.896	22.605	24.109	23.503	24.169	22.378	20.909	24.716	25.871	16.372
SpikeGS+Colmap	19.390	20.513	23.770	19.140	22.145	19.951	20.699	24.985	23.938	18.743
Ours	30.251	28.305	30.168	31.353	33.271	30.101	27.843	30.840	30.220	23.115

Table 6. **PSNR**: Quantitative comparisons of **novel view synthesis** on synthetic datasets for 10 scenes. The best performances are highlighted in **bold**. **Our SpikeGS demonstrates the best performance**.

Datasets	Tanks							Deblur-NeRF		
Methods Scenes	ballroom	barn	church	family	francis	horse	ignatius	factory	pool	tanabata
CF-3DGS	0.582	0.685	0.675	0.701	0.685	0.716	0.510	0.751	0.733	0.503
Instantsplat	0.582	0.692	0.680	0.719	0.710	0.720	0.495	0.745	0.739	0.420
Spikerecon+CF-3DGS	0.646	0.720	0.739	0.738	0.725	0.746	0.556	0.720	0.601	0.368
Spikerecon+Instant	0.655	0.727	0.751	0.757	0.735	0.753	0.549	0.699	0.750	0.405
SpikeGS+Colmap	0.612	0.738	0.745	0.681	0.720	0.732	0.609	0.749	0.738	0.561
Ours	0.925	0.895	0.918	0.937	0.914	0.932	0.857	0.796	0.838	0.662

Table 7. **SSIM**: Quantitative comparisons of **novel view synthesis** on synthetic datasets for 10 scenes. The best performances are highlighted in **bold**. **Our SpikeGS demonstrates the best performance**.

Datasets	Tanks							Deblur-NeRF		
Methods Scenes	ballroom	barn	church	family	francis	horse	ignatius	factory	pool	tanabata
CF-3DGS	0.405	0.430	0.357	0.366	0.456	0.358	0.512	0.409	0.480	0.527
Instantsplat	0.424	0.449	0.373	0.396	0.454	0.375	0.556	0.412	0.498	0.611
Spikerecon+CF-3DGS	0.364	0.396	0.319	0.339	0.423	0.340	0.485	0.423	0.536	0.590
Spikerecon+Instant	0.367	0.410	0.313	0.352	0.431	0.340	0.502	0.421	0.492	0.628
SpikeGS+Colmap	0.293	0.341	0.247	0.370	0.415	0.322	0.385	0.371	0.447	0.419
Ours	0.086	0.149	0.101	0.090	0.160	0.104	0.170	0.221	0.323	0.341

Table 8. **LPIPS**: Quantitative comparisons of **novel view synthesis** on synthetic datasets for 10 scenes. The best performances are highlighted in **bold**. **Our SpikeGS demonstrates the best performance**.



Figure 4. **Qualitative comparison for Camera Pose Estimation on Tanks and Temples.** The ground-truth trajectory and the estimated one are shown in red and green, respectively.

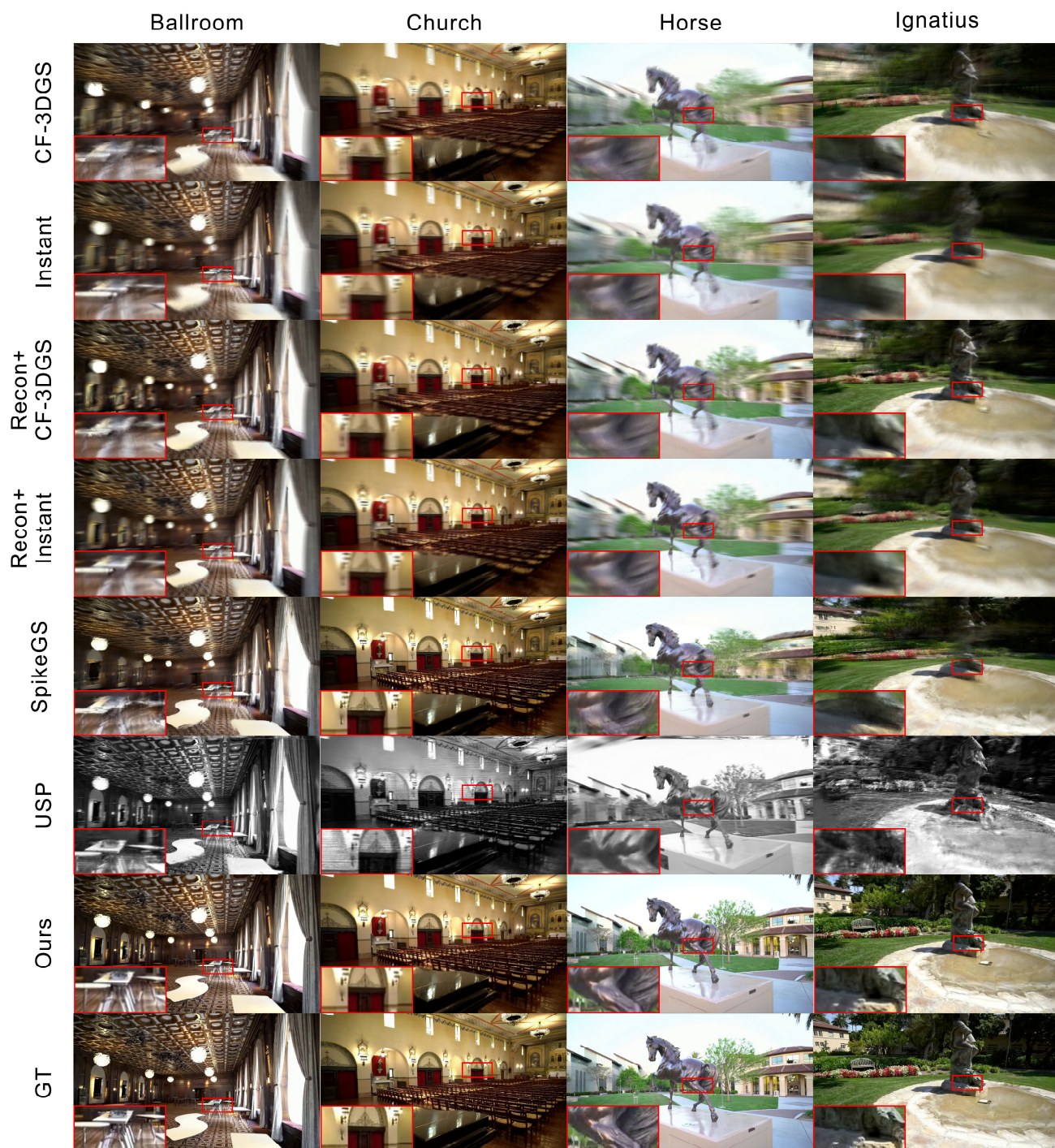


Figure 5. Supplementary results for novel view synthesis on Tanks and Temples.

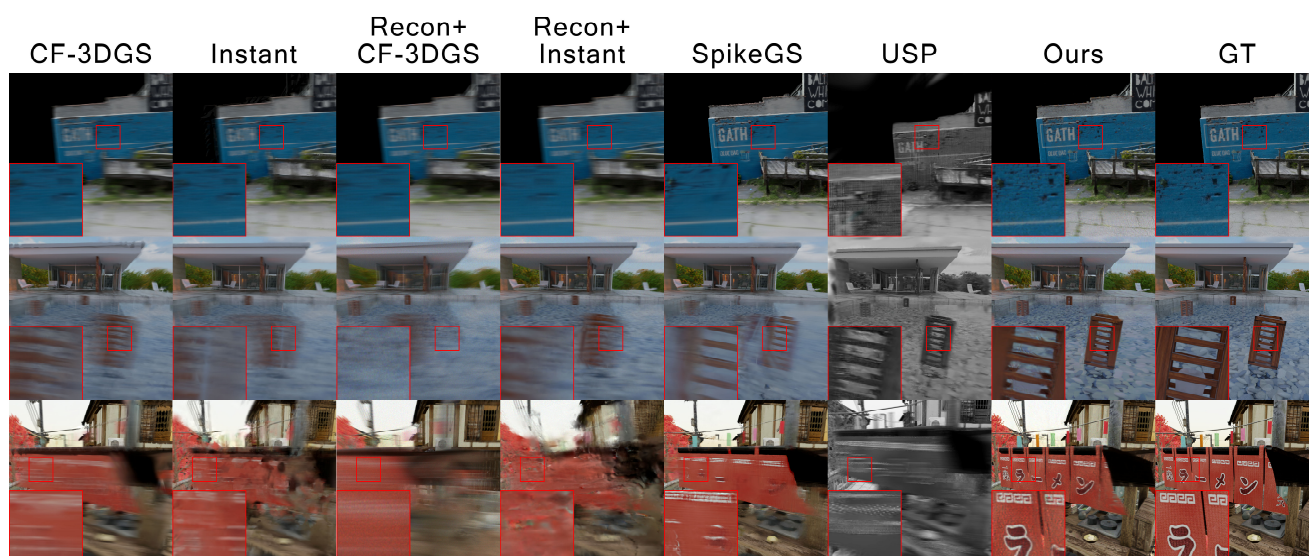


Figure 6. Qualitative comparison for novel view synthesis on Deblur-NeRF.