

SpikeGS: 3D Gaussian Splatting from Spike Streams with High-Speed Camera Motion

Jiyuan Zhang
Peking University, Beijing, China
jyzhang@stu.pku.edu.cn

Kang Chen
Wuhan University, Beijing, China
ck@stu.pku.edu.cn

Shiyan Chen
Peking University, Beijing, China
2001212818@stu.pku.edu.cn

Yajing Zheng*
Peking University, Beijing, China
yj.zheng@pku.edu.cn

Tiejun Huang
Peking University, Beijing, China
tjhuang@pku.edu.cn

Zhaofei Yu*
Peking University, Beijing, China
yuzf12@pku.edu.cn

ABSTRACT

Novel View Synthesis plays a crucial role by generating new 2D renderings from multi-view images of 3D scenes. However, capturing high-speed scenes with conventional cameras often leads to motion blur, hindering the effectiveness of 3D reconstruction. To address this challenge, high-frame-rate dense 3D reconstruction emerges as a vital technique, enabling detailed and accurate modeling of real-world objects or scenes in various fields, including Virtual Reality or embodied AI. Spike cameras, a novel type of neuromorphic sensor, continuously record scenes with an ultra-high temporal resolution, showing potential for accurate 3D reconstruction. Despite their promise, existing approaches, such as applying Neural Radiance Fields (NeRF) to spike cameras, encounter challenges due to the time-consuming rendering process. To address this issue, we make the first attempt to introduce the 3D Gaussian Splatting (3DGS) into spike cameras in high-speed capture, providing 3DGS as dense and continuous clues of views, then constructing **SpikeGS**. Specifically, to train SpikeGS, we establish computational equations between the rendering process of 3DGS and the processes of instantaneous imaging and exposing-like imaging of the continuous spike stream. Besides, we build a very lightweight but effective mapping process from spikes to instant images to support training. Furthermore, we introduced a new spike-based 3D rendering dataset for validation. Extensive experiments have demonstrated our method possesses the high quality of novel view rendering, proving the tremendous potential of spike cameras in modeling 3D scenes.

KEYWORDS

View Synthesis, Dense 3D reconstruction, Spike Camera, Gaussian Splatting

1 INTRODUCTION

Novel View Synthesis (NVS) involves the generation of new, unseen 2D renderings of a viewpoint from a sequence of multi-view images of a given 3D scene. This task holds significant importance in the realm of 3D scene reconstruction topic, playing a crucial role in computer vision and imaging research. The introduction of Neural Radiance Fields (NeRF) [24] has particularly drawn attention to this

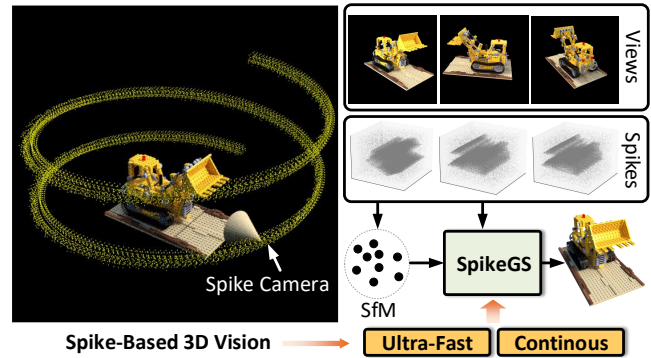


Figure 1: Illustration of spike-based Novel View Synthesis (NVS). Spike cameras, with ultra-high-speed continuous imaging capability, capture dense views and overcome the blurring effects of exposure imaging. We make the first step to present Spike-based Gaussian Splatting (*SpikeGS*), proving the potential of spike cameras in real-time 3D Imaging.

field. NeRF combines implicit neural representations with volume rendering techniques, paving the way for innovative approaches in NVS. In recent years, there have been remarkable developments in NeRF-related technologies, including enhanced rendering methods for higher scene quality [33, 37], strategies tailored for handling more complex [22] and dynamic scenes [7, 27], and techniques for image deblurring [20]. NeRF learns the continuous volumetric density and color that implicitly represents scenes by training a Multi-Layer Perception (MLP) network. However, rendering a new viewpoint still requires a great amount of sampling and integration through MLP, imposing limitations on rendering speed.

Why 3DGS standing out? Recently, 3D Gaussian Splatting (3DGS) [10] has been proposed to achieve real-time render speed and more reliable performance. Different from NeRF which models scenes implicitly, 3DGS represents scenes explicitly with a series of 3D Gaussians, which is initialized by Structure-from-Motion (SfM) [32]. Each Gaussian is parameterized by the mean position, the full 3D covariance matrix, the opacity, and its color. 3DGS projects 3D Gaussians to the 2D image plane with the differentiable Gaussian rasterization, which makes it able to be optimized by gradients of 3D Gaussians. It achieves very short training time and rendering speed and possesses great potential on NVS.

*Corresponding Authors.

Degradation of 3DGS with camera motion. Despite the remarkable efficacy demonstrated by 3DGS methodologies, their performance encounters inherent limitations imposed by the procedural characteristics of traditional exposure-based photo capturing. The conventional cameras, predicated by discrete exposure mechanisms, capture each frame within a predetermined temporal exposure window. This paradigm introduces a significant constraint when the image sequence intended for training 3DGS is subjected to blurring attributable to the high-velocity capture process. Such conditions lead to two profound detriments of the 3DGS framework. Firstly, the prerequisite quality of the initial point cloud essential for 3DGS is severely compromised. Training high-quality 3D Gaussians requires accurate assumptions about camera poses, which is difficult to achieve in some real-world scenarios. Secondly, the blurry images would affect optimizing the covariance matrix of the 3D Gaussians [14]. Moreover, the intrinsic interval between successive frames in traditional cameras entails a temporal void during which no visual information is captured. This hiatus in data acquisition may result in the omission of pivotal viewpoint information for scenes demanding dense perspective sampling for high-grade rendering, thereby adversely affecting the integrity of novel view synthesis. If we can accurately capture dense and continuous views, the performance of 3D reconstruction may make progress.

Introducing spike cameras for 3D reconstruction. The spike camera represents a novel class of neuromorphic visual sensors, boasting advantages such as ultra-high temporal resolution and a higher dynamic range. Inspired by the mechanism of the fovea in the retinas of primates, each unit on a spike camera asynchronously and continuously receives photons and accumulates photoelectric current, immediately emitting a spike when the voltage reaches a preset threshold. Event cameras [2, 18, 25] are also kind of neuromorphic cameras which also possess high temporal resolution. Several studies integrate event cameras with NeRF for NVS. However, events encode the change of light and do not have absolute intensity information. **spike cameras** encode the absolute light intensity of a scene at extremely high speeds, which reduces the significance of exposure time. This characteristic naturally mitigates the presence of blur and alleviates the speed requirements for the camera during the shooting process. Existing studies [43] have proved the temporal and spatial completeness of spikes in 2D reconstruction. In 3D scenes, spikes provide a denser and more continuous set of viewpoints. We believe that spike cameras hold tremendous potential for 3D scene reconstruction. Recently, pioneering work has been carried out with SpikeNeRF [46], demonstrating the feasibility of using spikes in modeling 3D scenes. However, SpikeNeRF faces several challenges: first, due to its complex spike simulation process, both training and rendering speeds are suboptimal; second, its training requires noise estimation to be recalibrated for different scenes, indicating a lack of adaptability; third, it fails to leverage the high temporal resolution advantage of spike cameras fully. This paper aims to fully exploit the high-speed and continuous imaging advantages of spiking cameras, constructing a spike-based 3D Gaussian Splatting model for the first time, and overcoming the limitations of training 3DGS on traditional RGB sequences.

What attempts we have made for spike-based GS? In this work, we make the first attempt to introduce the 3D Gaussian Splatting (3DGS) into spike cameras in high-speed capture, providing 3DGS as great supervision signals and constructing **SpikeGS**.

To be specific, we first build the framework of **SpikeGS** based on continuous spikes. we focus on two characteristics to assist the training of high-quality 3D scenes from the fast-moving spike camera: *Instantaneous Imaging from spikes*, and *Exposing-like Imaging from spikes*. On the one hand, to meet the instantaneous imaging assumption in 3DGS rendering, we aim to establish a ‘simple but effective’ mapping from continuous spikes to instant images, which can offer good signals for supervising the training. On the other hand, building the equality constraint between spikes and continuous camera poses better utilizes the continuity of spikes. By accumulating spikes and rendering images in SpikeGS in series, the exposure-like imaging equation is achieved for training. Secondly, we propose a very simple but effective mapping network *Spike-based Instant Mapping (SIM)* from spikes to instant images to support the *Instantaneous Imaging*, to offer reliable supervision signals for rendering SpikeGS. SIM is simply composed of several convolutional layers and incorporates blind spots to enable self-supervised training through spike firing frequency. Our SIM achieves an ultra-lightweight (30K params) design with a very fast inference speed (>1200FPS). In addition, we generate a high-quality spike-based 3D dataset for training and validation. Experiments demonstrate the superior 3D scene reconstruction capabilities of SpikeGS, proving the potential of spiking cameras in 3D vision. The contributions of this work can be summarized as follows:

- We make the first attempt to introduce the 3D Gaussian Splatting (3DGS) with spike cameras in high-speed capture, and constructing **SpikeGS**.
- To train SpikeGS efficiently and effectively, we establish computational equations that relate the rendering process of 3DGS to the instantaneous imaging and exposure-like imaging processes of continuous spikes.
- We establish a very lightweight but effective mapping process from spikes to instant images to assist training.
- Experiments demonstrate the superior 3D scene reconstruction capabilities of SpikeGS on existing and our proposed datasets.

2 RELATED WORKS

2.1 Spike-based Image Reconstruction

Spike cameras [9], as a novel type of bio-inspired camera, feature the capability of emitting spike bit stream with extremely low latency, thus endowing spike cameras with substantial advantages in the realm of the high-speed image reconstruction area. Specifically, Zhu et al. [45] initially proposes a straightforward spike-based reconstruction method “texture from play-back (TFP)”, which closely aligns with the imaging principles of conventional cameras. Inspired by the spike camera’s biological principles, studies like [43, 44] have employed short-term synaptic plasticity and retinal imaging principles to transform the spike stream into a high frame rate video sequence. However, these approaches often suffer from significant image quality degradation in real-world scenarios due to inadequate modeling of spike noise. Addressing this, Zhao et al. [40]

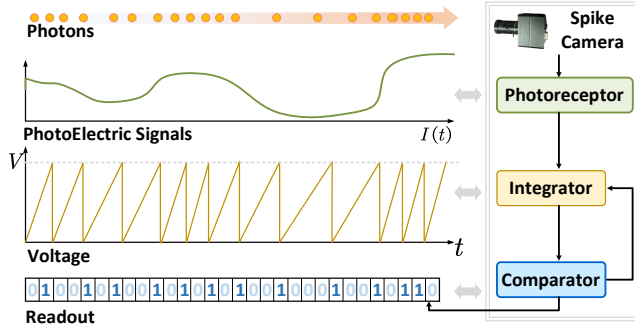


Figure 2: Illustration of the principle of spike cameras.

and Zhang et al. [39] leveraged the powerful nonlinear fitting capabilities of CNNs to train an end-to-end model for converting the spike stream into sharp images on synthetic datasets. While supervised methods trained on the synthetic dataset suffer from significant performance degradation when applied to real-world scenarios, Chen et al. [4] constructed a self-supervised spike-based reconstruction framework that jointly predicts optical flow and grayscale images. Some works focus on color spike cameras [6] or deblurring reconstruction [5].

2.2 Novel View Synthesis

Neural Radiance Fields Since the introduction of Neural Radiance Fields (NeRF) [24], which represents scenes implicitly and constructs one differentiable 3D scenes reconstruction framework, a significant amount of research has been garnered [15, 20, 34]. However, the quality of 3D scenes reconstructed by NeRF significantly deteriorates when the input image quality is severely degraded, such as in cases of motion blur. To this end, recent studies resort to bio-inspired event cameras, which output events with low temporal latency. For instance, Klenk et al. [11] utilized an event camera and established the E-NeRF framework, which can recover sharp scenes from events under high-speed camera movement. Rudnev et al. [29] learned the 3D RGB representation using a color event camera. Low and Lee [19] established a real event generation physical model and proposed Robust e-NeRF, capable of reconstructing high-quality scenes from sparse and noisy events produced by non-uniform moving cameras. Qi et al. [28] leveraged the complementary information between event and blurry images. Some studies [1, 21] focus on constructing dynamic NeRFs, *i.e.*, utilizing events to recover dynamic scenes with rigid transformations, which is challenging for traditional cameras owing to the limited frame rates.

3D Gaussian Splatting Kerbl et al. [10] proposes the novel real-time radiance field rendering approach with the 3D Gaussian splatting which has recently become a potent tool in computer graphics and vision. Scenes are represented with 3D Gaussians whose anisotropic covariance is optimized by gradients. Fu et al. [8] utilized geometric information and continuity in the video to get rid of the Structure-from-Motion(SfM) preprocessing. Yu et al. [38] emphasizes the importance of frequency constraints in 3DGS to avoid artifacts when sampling rates vary. Some works have been proposed to deal with the blurry images led by camera motion [14, 26, 30, 42]. Deblurring 3DGS [14] consists of a small network predicting the

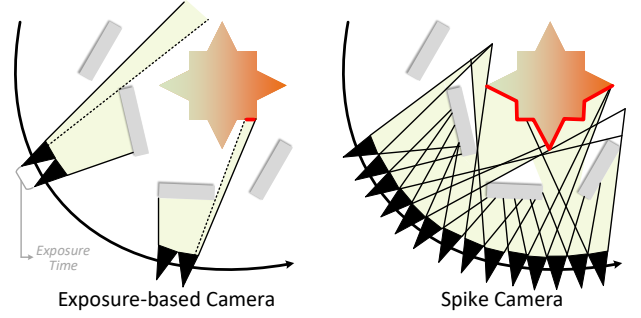


Figure 3: An example of why spike cameras possess potential on 3D scene understanding. The green area indicates the visible area. *Left*: The frame-based camera captures discretely with the exposure window, which may lead to blind areas of the scene. *Right*: The spike camera continuously records the scene, which offers more clues for the complex scene.

covariance offset of 3D Gaussians which represents the blur level of the image. BAD-Gaussian [42] models the physical process of motion blur by optimizing the camera trajectory with the exposure time. BAGS [26] models blur by a Blur Proposal Network (BPN) capable of predicting kernels and masks that indicate the blur region and types. Seiskari et al. [30] proposed to utilize the physical image formation process and velocities to incorporate rolling-shutter and motion blur effects.

However, limitations persist regarding the speed of camera motion across different scenes. The exposure photography principle inherent in traditional cameras also hampers 3DGS-based model performance. For the first time, we introduce the use of spike cameras, leveraging their advantage of ultra-high-speed continuous imaging to effectively model 3D scenes.

3 METHOD

3.1 Preliminary

Principle of the Spike Camera. In the spike camera, each pixel is equipped with a photoreceptor that receives photons at a high frequency, as shown in fig. 2. The arrival of photons alters the photoelectric signals of the receptor sensor and there is an integrator continuously accumulating the voltage. This accumulation continues until the voltage V reaches a predefined threshold Θ . At this moment t_e , the pixel emits a spike, and the voltage of the integrator is reset to zero, mathematically formulated as follows:

$$V(t) = \int_{t_s}^t \sigma \cdot I(t) dt \bmod \Theta, \quad (1)$$

where $I(t)$ represents the instant light intensity at time t , t_s is the moment when the previous spike was emitted, and σ is the constant photoelectric conversion coefficient. The emitted spike S will be read out at extremely short and uniform intervals τ ($25\mu s$), which can be formulated as:

$$S_{x,y,k} = \begin{cases} 1, & \text{if } \exists t \in ((k-1)\tau, k\tau], V_{x,y}(t) = 0, \\ 0, & \text{if } \forall t \in ((k-1)\tau, k\tau], V_{x,y}(t) > 0, \end{cases} \quad (2)$$

where (x, y) is the pixel coordinate on the imaging plane and k is the k -th readout of spikes.

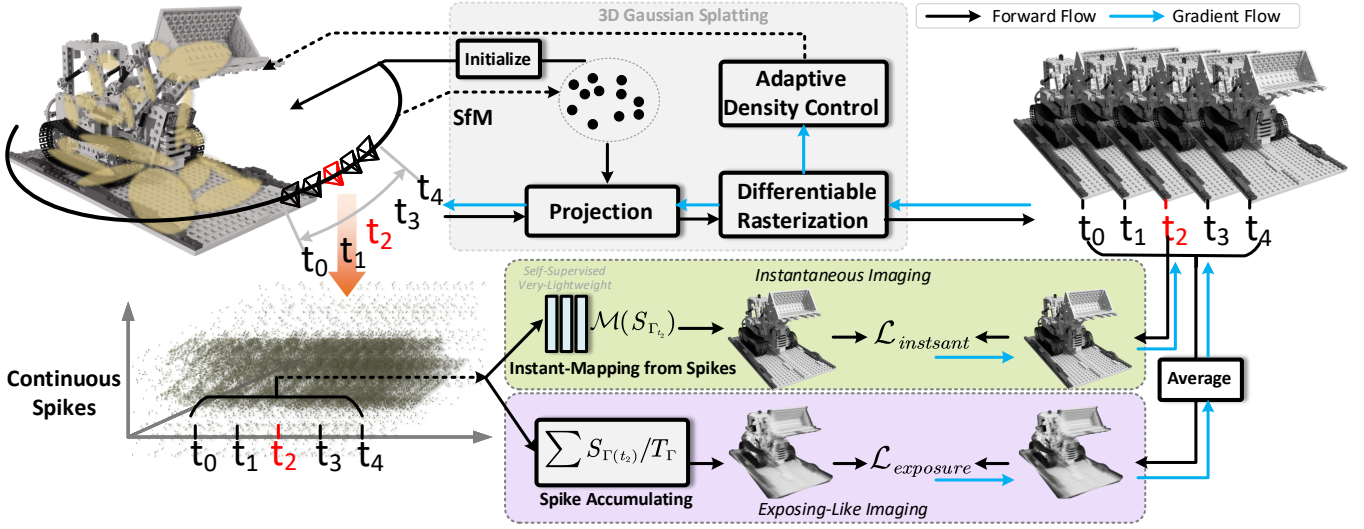


Figure 4: The schematic diagram of our SpikeGS. Combining the *Instantaneous Imaging* and *Exposing-Like Imaging* in spikes, together with the self-supervised lightweight mapping module from spikes to instant images, SpikeGS can be trained effectively.

3D Gaussian Splatting. 3D Gaussian Splatting stands out as a sophisticated point-based method for 3D scene reconstruction, offering notable advancements beyond the capabilities of Neural Radiance Fields. The core of 3DGS lies in its utilization of a series of 3D Gaussian primitives $\{\mathcal{G}_n | n = 1, \dots, N\}$ to encapsulate the scene’s spatial attributes.

Each Gaussian primitive is anchored by the central point \mathbf{p}_n and shaped by the covariance matrix Σ_n , which shape the Gaussian’s influence at any selected point \mathbf{v} in 3D space, described mathematically as:

$$\mathcal{G}_n(\mathbf{v}) = e^{-\frac{1}{2}(\mathbf{v}-\mathbf{p}_n)^T \Sigma_n^{-1}(\mathbf{v}-\mathbf{p}_n)}. \quad (3)$$

In the rendering phase, these 3D Gaussians are projected onto a 2D plane along the ray r , resulting in 2D Gaussian forms \mathcal{G}_n^{2D} . Throughout this process, the 3D Gaussians are endowed with additional properties, such as opacity α and color c , which play a crucial role in the rendering equation:

$$C(r) = \sum_{k=1}^N T_i \alpha_i c_i \mathcal{G}_i^{2D}, i = D_k, T_i = \prod_{j=1}^{i-1} (1 - \alpha_j \mathcal{G}_j^{2D}), \quad (4)$$

where D represents the index of the 3D Gaussian primitives set rearranged according to their depth over the rendered tile.

3.2 Analysis on Spike-based Views

Potentials on Continuous Imaging in 3D Scenes. In high-speed motion settings, using traditional cameras for 3D reconstruction faces two challenges: 1) insufficient frame rates of traditional cameras lead to missed details due to occlusion in the scene and the camera at certain viewpoints; 2) images captured by traditional cameras in high-speed scenarios tend to blur, as in Fig. 3(left). The spike camera offers a solution by outputting the continuous spikes at 40,000 Hz with minimal latency, which ensures that the full details of the captured object are visible even under high-speed camera

motion settings, as in Fig. 3(right). Recent spike-based image reconstruction methods [39, 40] have demonstrated the capability to recover sharp images from spikes at any given timestamp.

3.3 Spike-based Gaussian Splatting

We aim to train a high-quality 3DGS with spikes as supervision. During recording the scene with the spike camera, spikes are continuous. Thus, at training, a 3DGS model can be denoted as:

$$\hat{I}(t) = \mathcal{F}_{3\text{dgs}}(\mathcal{P}\mathcal{C}, v(t)) \quad (5)$$

where $\mathcal{P}\mathcal{C}$ is the initial points, $v(t)$ is the camera pose at some timestamp t and $\hat{I}(t)$ is the rendered image. Spikes are irregularly binary data, which can be viewed directly. Then the key problem in spike-based 3DGS is raised:

how to deal with the supervision for $\hat{I}(t)$ at view $v(t)$?

To supervise the training of continuous 3D scenes based on the discrete spike stream, a straightforward idea is to construct a virtual exposure window as in traditional RGB cameras, *i.e.*, accumulating a large number of spikes and summing up them to get the image. However, in the experimental setting of this paper, the camera moves around the scene at extremely high speeds, leading to significant motion blur in the images obtained through this virtual exposure method. In our SpikeGS, we focus on two factors to assist the training of high-quality 3D scenes from the fast-moving spike camera: (A) *Instantaneous Imaging from spikes*, (B) *Exposing-like Imaging from spikes*. The SpikeGS framework is shown in Fig. 4.

(A) Instantaneous Imaging from spikes. In 3DGS, it generally follows the assumption of instantaneous exposure where the images for supervision should denote the instant light intensity. To address this, an *ideal* mapping \mathcal{M} from spikes to instant images is essential, as follows:

$$\bar{I}(t) = \mathcal{M}(S_{\Gamma}(t)), \quad (6)$$

where $S_{\Gamma(t)}$ is a segment of spikes in a time interval Γ around the t , and $\bar{I}(t)$ is the instant image at t predicted from spikes.

Benefit from the continuity of spikes, plenty of motion and texture information are contained in the spikes. If the mapping \mathcal{M} can be established, the instant imaging loss can be formulated as follows to offer SpikeGS supervision as in Fig. 4 for training:

$$\mathcal{L}_{instant} = \|\hat{I}(t) - \bar{I}(t)\|_1. \quad (7)$$

(B) Exposing-like Imaging from spikes. In our setting, a spike camera records a 3D scene continuously, indicating that the camera poses are also non-uniformly continuous. In the time interval $\Gamma(t)$ around the view $v(t)$ at t (the duration of Γ is denoted as T_{Γ}), spike streams and camera poses are both continuous. We aim to build the mathematical formulation between spikes and camera poses. For camera poses, 3DGS itself inputs camera poses and outputs the rendered image at the corresponding view. Assuming the output of 3DGS is an instant clear image, then the mean of its rendering results with continuous poses will approximate an exposure-like blurred image \hat{B}_t with an exposure time of T_{Γ} , as follows:

$$\hat{B}(t) = \frac{1}{T_{\Gamma}} \int_{t-T_{\Gamma}/2}^{t+T_{\Gamma}/2} \mathcal{F}_{3dgs}(\mathcal{P}\mathcal{C}, v(t)). \quad (8)$$

However, in real-world data capturing, the camera pose cannot be read out at any timestamps. They are recorded discretely. Suppose that there are K poses in T_{Γ} , then Eq.8 can be re-write as:

$$\hat{B}(t) = \frac{1}{K} \sum_{k=0}^K \mathcal{F}_{3dgs}(\mathcal{P}\mathcal{C}, v(t_k)). \quad (9)$$

In the spike stream, the exposure-like image in the $\Gamma(t)$ can be achieved by accumulating spikes along the time axis. with the characteristics of spikes, we can formulate an approximate equation between poses and spikes aiming to train the SpikeGS:

$$\mathcal{L}_{exposure} = \|\hat{B}(t) - \frac{S_{\Gamma(t)}}{T_{\Gamma}}\|_1. \quad (10)$$

The illustration is shown in Fig. 4. In this way, utilizing the equation Eq. 7 and Eq. 10, the SpikeGS can be trained.

3.4 Ideal Mapping of Spikes to Instant Image

For the **instantaneous imaging from spikes**, recall that to successfully achieve the training of SpikeGS, an ideal mapping \mathcal{M} from the segment of spikes to instantaneous image (as in Eq. 6) is needed to satisfy the supervision loss in Eq. 7. Thus, in this section, we are dedicated to dealing with the problem:

How to get the ideal mapping \mathcal{M} from spikes?

What is an ideal mapping from spikes to images? (1) the mapping should be **High-Quality** and **Generalized** in the 3D scene, which means that the image recovered from spikes has sharp textures across all the views. (2) To meet the feature of real-time rendering and very-fast training of the 3DGS, the mapping should be **Simple but Effective**. Upon these requirements, we build a new **Spike Instant Mapping (SIM)** network ($\mathcal{M}(\cdot)$).

Several approaches have been proposed to recover sharp images from spike segments. Although TFI and TFP [45], as the most basic and fast numerical analysis spike reconstruction algorithms, can recover images fast, their quality is poor with noise and blurring.

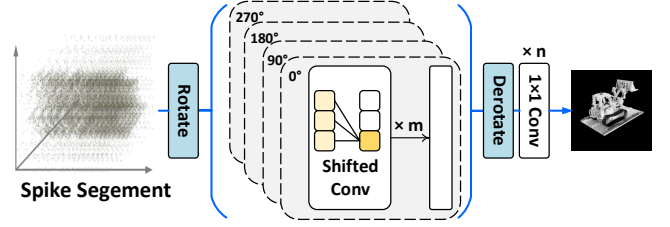


Figure 5: Model Architecture for mapping spikes to images.

Some supervised deep learning-based methods proposed to recover high-quality grayscale images from spike streams rely on training on large synthetic datasets. The capability of generalization is poor and the model is also complex and has a low speed of inference.

In SpikeGS, from the perspective of generalization ability, we aim to accomplish the mapping $\mathcal{M}(S_{\Gamma})$ in a self-supervised manner, training the specific model with spikes in each 3D scene.

SSML [3] is the first self-supervised reconstruction algorithm tailored for spike cameras. It adopts a blind spot network (BSN) structure [13, 16, 17, 35, 36] to predict the current pixel from neighboring spikes. However, SSML’s network structure and computing processes are relatively complex, and its training cost is lagging compared to the fast 3DGS process. Therefore, we aim to build a ‘simple but effective’ self-supervised module and strive to achieve a lightweight design, motivated by BSN.

Principle of BSN. The BSN was initially utilized in self-supervised image-denoising tasks. It relies on the assumption of Noise2Void [12], which assumes that under the premise of noise mean being zero and noise having no spatial correlation, the optimization objective of self-supervised denoising is approximately equivalent to supervised denoising, namely:

$$\arg \min \mathbb{E}\{(f_{\theta}(x) - x)^2\} \approx \arg \min \mathbb{E}\{(f_{\theta}(x) - y)^2\}, \quad (11)$$

where $f_{\theta}(\cdot)$ represents the denoising network, x denotes the noisy input and y represents the potential sharp image. The equation between the x and y is:

$$x = y + m, \quad (12)$$

where m is the noise. To avoid the identity mapping of the noisy image itself. BSN is thus introduced to address this issue, where the receptive field of each pixel does not include the pixel itself, preventing the identity mapping of the noise.

The spike stream mainly suffers from stochastic thermal noise. Accumulation of dark current can lead the accumulator to reach the firing threshold prematurely, resulting in unexpected binary spike noise. Methods like TFP [45] can construct a low-quality image from spikes with noise, as $I_{noisy}(t) = \text{TFP}(S_{\Gamma(t)})$. Then, between the desired clean image $\bar{I}(t)$ and the $I_{noisy}(t)$, the formulation like Eq: 12 can be established as:

$$I_{noisy}(t) = \bar{I}(t) + m, \quad (13)$$

Thus, for self-supervised BSN-based module \mathcal{M} , the optimal module \mathcal{M}^* can be obtained by:

$$\mathcal{M}^* = \arg \min \mathbb{E}\{(\mathcal{M}(S_{\Gamma(t)}) - I_{noisy}(t))^2\}. \quad (14)$$



Figure 6: Qualitative results compared with other methods on the synthetic dataset. We compare with SpikeNeRF [46], and three baseline methods that are the cascading two-stage model TFP(33)+3DGS, TFP(33)+3DGS and TFI+3DGS.

With such a definition, we can use the results of TFP [45] from spikes as the self-supervision for spike-to-image mapping \mathcal{M} .

Constructing the Lightweight Mapping Module for Spikes.

Following the assumptions in SSML [3], the current pixel value can be inferred from the neighboring spike stream. By excluding the central pixel position, the network is unable to learn the noise value at the current position from the spike stream. Fig.5 illustrates the spike reconstruction network we designed. Specifically, we employ a blind spot construction scheme similar to SSML, using shift-based convolutions. We propose to design the **Spike Instant Mapping (SIM)** network ($\mathcal{M}(\cdot)$) most simply with only sequential Conv layers, as shown in Fig. 5.

Since shift-based convolutions cause the network’s receptive field to grow in a single direction, we rotate the input spike stream into four parts to obtain a complete receptive field in four directions. The rotated spike stream is fed into m layers of shift-based 3×3 convolutional layers. In this way, the network possesses a receptive field with a $(2m+1, 2m+1)$ size. In the end, the extracted features are combined through n layers of 1×1 convolutions and the re-rotation operation to obtain the mapped clean image.

Since spikes reach a time resolution of 40,000 Hz, we can safely assume that small pixel displacements are caused by motions within a segment of spikes during a very short interval in T_F . Thus, the $(2m + 1, 2m + 1)$ receptive field is enough when m is small. In our implementation, we only use $m = 3$ and $n = 3$. Thus we construct a very lightweight but robust network for mapping spikes to images.

To train the BSN-based \mathcal{M} , we simply utilize L1 loss:

$$\mathcal{L}_{\mathcal{M}} = \|\mathcal{M}(S_{\Gamma}) - I_{noisy}\|_1 = \|\mathcal{M}(S_{\Gamma}) - \text{TFP}(S_{\Gamma})\|_1. \quad (15)$$

3.5 Training the SpikeGS

The loss in Eq. 7 holds with the achievement of idea mapping \mathcal{M} from spikes to images which meets the requirement of Eq. 6. Thus, the loss function for training SpikeGS is as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{instant} + \lambda \cdot \mathcal{L}_{exposure}, \quad (16)$$

where λ is a hyperparameter to balance the weight of two losses.

4 EXPERIMENTS

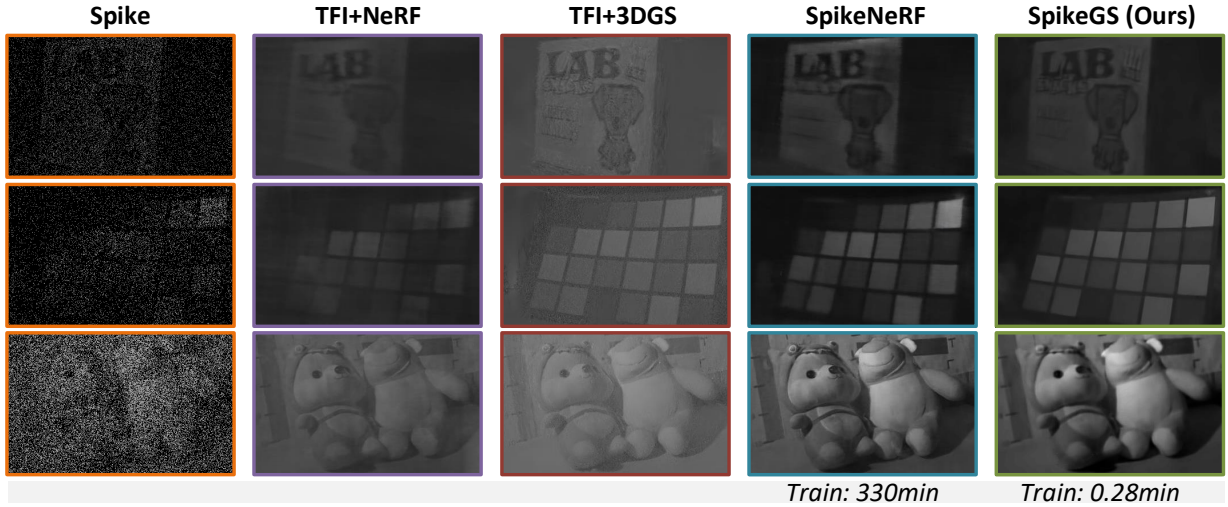
4.1 Datasets

To our knowledge, there is a related dataset [46] focusing on spike-based 3D reconstruction. However, this dataset has converted the spike stream into image format, rendering it impractical to implement our SpikeGS on this dataset. Therefore, we construct a new synthetic dataset and provide the raw spike stream instead of the image format, as described in Zhu et al. [46].

Synthetic Dataset. To evaluate the quantitative performance of our approach, we first conduct experiments on synthetic scenes provided by Mildenhall et al. [24]. We begin by designing a virtual camera path in Blender that orbits and ascends in a spiral manner around the captured scene. Following this, we render the video

Table 1: Quantitative Results on the built Synthetic dataset.

Method	Lego	Chair	Materials	Drums	Mic	Hotdog	Ficus
	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow
TFP(33)+3DGS	26.03/89.18	28.65/95.45	29.17/93.85	25.76/91.97	31.03/95.80	33.36/96.43	30.12/96.31
TFP(65)+3DGS	23.19/84.55	25.54/93.11	26.49/91.24	24.12/89.23	29.04/94.1	30.82/95.11	26.71/93.60
TFI+3DGS	24.52/86.33	25.12/87.99	25.12/87.99	27.04/93.08	30.88/95.36	25.97/88.05	29.54/94.81
SpikeNeRF	19.24/89.77	19.63/90.23	25.48/94.34	22.07/89.91	29.47/95.39	23.18/93.62	25.42/95.30
SpikeGS (Ours)	32.67/96.44	35.53/98.53	34.18/97.40	28.82/96.07	34.80/98.29	37.39/98.05	36.81/99.00

**Figure 7: Real-World Comparison with other methods on the dataset in SpikeNeRF [46]. We mainly compare with SpikeNeRF, and the two baseline methods are the cascading two-stage model TFI+NeRF and TFI+3DGS.**

sequence along the designed camera path and employ the XVFI frame interpolation algorithm [31] to generate 7 additional frames between each pair of adjacent frames. Finally, we turn to a physically based spike simulator [41], which adopts the Poisson model into the spike simulation process, to convert the high-frame-rate video sequences into the spike stream with low latency.

Real-world Dataset. We evaluate the performance of SpikeGS on the real-world spike dataset released by Zhu et al. [46]. This dataset is captured utilizing a spike camera with a spatial resolution of 250×400 . Five real-world scenarios are recorded, each comprising 35 spike images captured by the fast-moving spike camera. The dataset is organized in the LLFF [23] format, slightly different from the 360 synthetic scenes constructed in the synthetic dataset.

4.2 Training Details

All experiments were conducted on a single NVIDIA GTX 4090, with PyTorch. SpikeGS is trained for 30k iterations taking about 15 minutes, and the learning rate and scheduler settings are identical to those of standard 3DGS. In the implementation, the number of continuous camera poses $K = 5$ used for calculating $\mathcal{L}_{exposure}$, corresponds to a spike stream length of 33. During training, the parameters of the self-supervised network \mathcal{M} were frozen to provide the supervisory signal for $\mathcal{L}_{instant}$.

4.3 Quantitative and Qualitative Comparison

We compare our SpikeGS with the SpikeNeRF [46], the only spike-based 3D reconstruction work to our knowledge, in the synthetic and real-world scenarios for quantitative and qualitative comparisons. As for the baseline methods, we chose two direct spike-to-image reconstruction methods, TFI [45] and TFP [45] (window size = 33 and 65), and cascaded them with standard 3DGS [10], completing the comparison using the two-stage approach. We choose them to thoroughly show the effectiveness of the 3DGS and our proposed self-supervised BSN network. In the following, We compare our SpikeGS against other methods mainly from two aspects: image quality and training speed.

Synthetic Results. In Tab. 1, we present the results of our method compared to others across all 7 scenes of the synthetic dataset. PSNR and SSIM are used as the quantitative metrics. The results show that the quality of novel view synthesis by SpikeGS significantly surpasses other methods. Specifically, compared to those that use TFI and TFP as cascading modules with 3DGS as the baseline model, SpikeGS surpasses them by approximately 5.3dB, 7.7dB, and 7.5dB in PSNR, respectively. Meanwhile, compared to SpikeNeRF, SpikeGS exceeds by more than 10.8dB. This indicates the high effectiveness of the SpikeGS in 3D reconstruction from spikes. Fig. 6 provides a comparison of visual results. As shown

Table 2: Ablation Study on Losses in Modules in the SpikeGS.

Method	Lego	Chair	Materials	Drums	Mic	Hotdog	Ficus
	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow	PSNR/SSIM \uparrow
Only $\mathcal{L}_{exposure}$	29.23/92.94	31.73/96.91	31.42/95.56	26.90/94.19	32.72/97.34	35.48/96.92	34.74/98.41
Only $\mathcal{L}_{instant}$	32.28/96.30	34.62/98.42	33.62/97.17	28.09/96.01	34.22/98.09	36.11/97.87	36.47/98.95
$\mathcal{L}_{exposure} + \mathcal{L}_{instant}$	32.67/96.44	35.53/98.53	34.18/97.40	28.82/96.07	34.80/98.29	37.39/98.05	36.81/99.00

in the results, the images predicted by SpikeNeRF exhibit blurring effects as well as poor adaptability to the motion-induced spike stream; in contrast, images predicted by SpikeGS maintain clear textures and smooth edges, with enhanced realism. Moreover, the training time of SpikeNeRF is about 10 hours (**600min**), while SpikeGS only needs **15min** for training.

Real-world Results. Regarding training speed, our SpikeGS demonstrates a substantial efficiency gain against other methods as evidenced in Fig. 7. SpikeGS completes the training in merely 0.28 minutes, a significant reduction from the 330 minutes required by SpikeNeRF. This marked decrease in training time, by over three orders of magnitude, is mainly attributed to the employment of the 3DGS and our designed extremely lightweight BSN, which has faster speed compared to the NeRF framework and SNN as in SpikeNeRF. In terms of image quality, the presented visual outputs indicate that SpikeGS maintains higher reconstruction fidelity than other methods. Specifically, the ‘Box’ and ‘Grid’ examples, which are typically challenging due to their regular geometries and uniform patterns, are rendered with greater clarity and less noise by SpikeGS. Moreover, the ‘Dolls’ instance, characterized by intricate textures and shading, appears to be reconstructed with greater fidelity, indicating the superior handling of subtle image features by our SpikeGS.

5 ABLATION STUDY

5.1 Ablation on Modules

We conduct ablation experiments on the two types of loss, specifically $\mathcal{L}_{instant}$ and $\mathcal{L}_{exposure}$. The results in Tab. 2 show that if each loss is adopted individually, the model performance decreases. When they are trained together, the model performance significantly increases by 2.57dB and 0.68dB, respectively. This experiment demonstrates the rationality and effectiveness of our architectural design. Through exploring two types of imaging characteristics of spikes, SpikeGS is efficiently and effectively trained.

5.2 Ablation on Continuous Rendering Loss

Table 3: Ablation study on the Continuous Rendering Loss with different lengths of spikes for supervision.

Render Images	Spikes	PSNR \uparrow	SSIM \uparrow	Train
13	97	33.97	97.39	25.5min
9	65	34.28	97.66	22.1min
5	33	34.31	97.68	15.0min

We utilize the average firing rate of the continuous spike stream as the supervision for the Continuous Rendering Loss, simulating the real long-exposure process to optimize the pixel distribution of images with only the short-exposure imaging loss. We conduct ablation experiments on the number of continuous rendering images and the length of the spikes for the loss. The results from Tab. 3 indicate that excessively long exposure times lead to a certain degree of performance degradation and increased training time. Therefore, We ultimately adopt the setting of rendering 5 images.

5.3 Discussion on Reconstruction Model

Table 4: Performance of the lightweight self-supervised reconstruction model.

Scene	PSNR \uparrow	SSIM \uparrow	Speed	Param
Lego	33.36	96.12		
Chair	35.45	98.13		
Materials	37.24	98.70		
Drums	32.10	97.00	1200+FPS	30K
Mic	35.07	98.01		
Hotdog	37.64	97.30		
Ficus	39.28	99.08		

In this section, we aim to highlight and analyze the advantages and significance of our hyper-quantized self-supervised reconstruction model, tinySpkRecon. (1) In the context of 3D scene understanding based on spike cameras, there is no image information available for designing reconstruction models with supervised learning, and supervised models often lack generalizability; Thus, exploring high-performance self-supervised models is essential. (2) Supervised reconstruction models, such as Spk2ImgNet [40], suffer from slow inference time, poor generalizability, and extremely high computational complexity. Complex reconstruction models contradict the real-time rendering characteristics of 3DGS; therefore, designing ultra-lightweight, fast-inference, and highly generalizable self-supervised models is necessary. Tab. 4 demonstrates that our designed self-supervised model performs well in scene reconstruction with strong generalizability; at the same time, our model can infer at **speeds exceeding 1200 FPS** (frames per second) on a single 4090 GPU, with only **30K parameters** required. Such design and performance will not increase any computational burden on the 3DGS pipeline and greatly enhance the rendering performance of our SpikeGS.

6 CONCLUSION

We make the first attempt to introduce the 3D Gaussian Splatting (3DGS) with spike cameras in high-speed capture, and constructing **SpikeGS**. A lightweight self-supervised model *tinySpkRecon* is proposed for recovering images from spikes. The loss combined with *Instantaneous imaging* and *Exposure-like imaging* is designed to improve rendering quality. Experiments demonstrate the superior 3D scene reconstruction capabilities of SpikeGS both on the synthetic and the real-world datasets.

REFERENCES

- [1] Anish Bhattacharya, Ratnesh Madaan, Fernando Cladera, Sai Vemprala, Rogério Bonatti, Kostas Daniilidis, Ashish Kapoor, Vijay Kumar, Nikolai Matni, and Jayesh K Gupta. 2024. EvDNeRF: Reconstructing Event Data with Dynamic Neural Radiance Fields. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 5846–5855.
- [2] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. 2014. A 240×180 130 db $3 \mu\text{s}$ latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits* 49, 10 (2014), 2333–2341.
- [3] Shiyuan Chen, Chaoteng Duan, Zhaofei Yu, Ruiqin Xiong, and Tiejun Huang. 2022. Self-Supervised Mutual Learning for Dynamic Scene Reconstruction of Spiking Camera. In *Proceedings of the International Joint Conference on Artificial Intelligence*. 2859–2866.
- [4] Shiyuan Chen, Zhaofei Yu, and Tiejun Huang. 2023. Self-supervised joint dynamic scene reconstruction and optical flow estimation for spiking camera. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 350–358.
- [5] Shiyuan Chen, Jiyuan Zhang, Yajing Zheng, Tiejun Huang, and Zhaofei Yu. 2024. Enhancing Motion Deblurring in High-Speed Scenes with Spike Streams. *Advances in Neural Information Processing Systems* 36 (2024).
- [6] Yanchen Dong, Ruiqin Xiong, Jing Zhao, Jian Zhang, Xiaopeng Fan, Shuyuan Zhu, and Tiejun Huang. 2024. Joint Demosaicing and Denoising for Spike Camera. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 1582–1590.
- [7] Yilun Du, Yinan Zhang, Hong-Xing Yu, Joshua B Tenenbaum, and Jiajun Wu. 2021. Neural radiance flow for 4d view synthesis and video processing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. IEEE Computer Society, 14304–14314.
- [8] Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A Efros, and Xiaolong Wang. 2023. Colmap-free 3d gaussian splatting. *arXiv preprint arXiv:2312.07504* (2023).
- [9] Tiejun Huang, Yajing Zheng, Zhaofei Yu, Rui Chen, Yuan Li, Ruiqin Xiong, Lei Ma, Junwei Zhao, Siwei Dong, Lin Zhu, et al. 2023. 1000× faster camera and machine vision with ordinary devices. *Engineering* 25 (2023), 110–119.
- [10] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics* 42, 4 (2023), 1–14.
- [11] Simon Klenk, Lukas Koestler, Davide Scaramuzza, and Daniel Cremers. 2023. E-nerf: Neural radiance fields from a moving event camera. *IEEE Robotics and Automation Letters* 8, 3 (2023), 1587–1594.
- [12] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. 2019. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2129–2137.
- [13] Samuli Laine, Tero Karras, Jaakko Lehtinen, and Timo Aila. 2019. High-quality self-supervised deep image denoising. In *Advances in Neural Information Processing Systems*.
- [14] Byeonghyeon Lee, Howoong Lee, Xiangyu Sun, Usman Ali, and Eunbyung Park. 2024. Deblurring 3D Gaussian Splatting. *arXiv preprint arXiv:2401.00834* (2024).
- [15] Dogyoon Lee, Minhyeok Lee, Chajin Shin, and Sangyoung Lee. 2023. DP-NeRF: Deblurred Neural Radiance Field With Physical Scene Priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12386–12396.
- [16] Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. 2022. AP-BSN: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 17725–17734.
- [17] Junyi Li, Zhilu Zhang, Xiaoyu Liu, Chaoyu Feng, Xiaotao Wang, Lei Lei, and Wangmeng Zuo. 2023. Spatially Adaptive Self-Supervised Learning for Real-World Image Denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9914–9924.
- [18] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. 2008. A 128×128 120 db $15 \mu\text{s}$ latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits* 43, 2 (2008), 566–576.
- [19] Weng Fei Low and Gim Hee Lee. 2023. Robust e-NeRF: NeRF from Sparse & Noisy Events under Non-Uniform Motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 18335–18346.
- [20] Li Ma, Xiaoyu Li, Jing Liao, Qi Zhang, Xuan Wang, Jue Wang, and Pedro V Sander. 2022. Deblur-nerf: Neural radiance fields from blurry images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12861–12870.
- [21] Qi Ma, Danda Pani Paudel, Ajad Chhatkuli, and Luc Van Gool. 2023. Deformable Neural Radiance Fields using RGB and Event Cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3590–3600.
- [22] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. 2021. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7210–7219.
- [23] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. 2019. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. *ACM Transactions on Graphics* (2019).
- [24] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.
- [25] Diederik Paul Moeys, Federico Corradi, Chenghan Li, Simeon A Bamford, Luca Longinotti, Fabian F Voigt, Stewart Berry, Gemma Taverni, Fritjof Helmchen, and Tobi Delbruck. 2017. A sensitive dynamic and active pixel vision sensor for color or neural imaging applications. *IEEE Transactions on Biomedical Circuits and Systems* 12, 1 (2017), 123–136.
- [26] Cheng Peng, Yutao Tang, Yifan Zhou, Nengyu Wang, Xijun Liu, Deming Li, and Rama Chellappa. 2024. BAGS: Blur Agnostic Gaussian Splatting through Multi-Scale Kernel Modeling. *arXiv preprint arXiv:2403.04926* (2024).
- [27] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. 2021. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10318–10327.
- [28] Yunshan Qi, Lin Zhu, Yu Zhang, and Jia Li. 2023. E2NeRF: Event Enhanced Neural Radiance Fields from Blurry Images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13254–13264.
- [29] Viktor Rudnev, Mohamed Elgharib, Christian Theobalt, and Vladislav Golyanik. 2023. EventNeRF: Neural radiance fields from a single colour event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4992–5002.
- [30] Otto Seiskari, Jerry Ylilammi, Valteri Kaatrasalo, Pekka Rantalankila, Matias Turkulainen, Juho Kannala, Esa Rahtu, and Arno Solin. 2024. Gaussian Splatting on the Move: Blur and Rolling Shutter Compensation for Natural Camera Motion. *arXiv preprint arXiv:2403.13327* (2024).
- [31] Hyeonjun Sim, Jihyong Oh, and Munchul Kim. 2021. Xvfi: extreme video frame interpolation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14489–14498.
- [32] Noah Snavely, Steven M Seitz, and Richard Szeliski. 2006. Photo tourism: exploring photo collections in 3D. In *ACM SIGGRAPH*. 835–846.
- [33] Chen Wang, Xian Wu, Yuan-Chen Guo, Song-Hai Zhang, Yu-Wing Tai, and Shi-Min Hu. 2022. Nerf-sr: High quality neural radiance fields using supersampling. In *Proceedings of the 30th ACM International Conference on Multimedia*. 6445–6454.
- [34] Peng Wang, Lingzhe Zhao, Ruijie Ma, and Peidong Liu. 2023. BAD-NeRF: Bundle Adjusted Deblur Neural Radiance Fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4170–4179.
- [35] Zichun Wang, Ying Fu, Ji Liu, and Yulun Zhang. 2023. LG-BPN: Local and Global Blind-Patch Network for Self-Supervised Real-World Denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18156–18165.
- [36] Xiaohe Wu, Ming Liu, Yue Cao, Dongwei Ren, and Wangmeng Zuo. 2020. Unpaired learning of deep image denoising. In *European Conference on Computer Vision*.
- [37] Xianggang Yu, Jiapeng Tang, Yipeng Qin, Chenghong Li, Xiaoguang Han, Linchao Bao, and Shuguang Cui. 2022. PVSeRF: joint pixel-, voxel- and surface-aligned radiance field for single-image novel view synthesis. In *Proceedings of the 30th ACM International Conference on Multimedia*. 1572–1583.
- [38] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. 2023. Mip-splatting: Alias-free 3d gaussian splatting. *arXiv preprint arXiv:2311.16493* (2023).
- [39] Jiyuan Zhang, Shanshan Jia, Zhaofei Yu, and Tiejun Huang. 2023. Learning temporal-ordered representation for spike streams based on discrete wavelet transforms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 137–147.
- [40] Jing Zhao, Ruiqin Xiong, Hangfan Liu, Jian Zhang, and Tiejun Huang. 2021. Spk2imgnet: Learning to reconstruct dynamic scene from continuous spike stream. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11996–12005.
- [41] Junwei Zhao, Shiliang Zhang, Lei Ma, Zhaofei Yu, and Tiejun Huang. 2022. Spikingsim: A bio-inspired spiking simulator. In *IEEE International Symposium on Circuits and Systems*. 3003–3007.
- [42] Lingzhe Zhao, Peng Wang, and Peidong Liu. 2024. BAD-Gaussians: Bundle Adjusted Deblur Gaussian Splatting. *arXiv preprint arXiv:2403.11831* (2024).

- [43] Yajing Zheng, Lingxiao Zheng, Zhaofei Yu, Tiejun Huang, and Song Wang. 2023. Capture the moment: High-speed imaging with spiking cameras through short-term plasticity. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
- [44] Yajing Zheng, Lingxiao Zheng, Zhaofei Yu, Boxin Shi, Yonghong Tian, and Tiejun Huang. 2021. High-speed image reconstruction through short-term plasticity for spiking cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6358–6367.
- [45] Lin Zhu, Siwei Dong, Tiejun Huang, and Yonghong Tian. 2019. A retina-inspired sampling method for visual texture reconstruction. In *IEEE International Conference on Multimedia and Expo*. 1432–1437.
- [46] Lin Zhu, Kangmin Jia, Yifan Zhao, Yunshan Qi, Lizhi Wang, and Hua Huang. 2024. SpikeNeRF: Learning Neural Radiance Fields from Continuous Spike Stream. *arXiv preprint arXiv:2403.11222* (2024).