This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

FastEventDGS: Deformable Gaussian Splatting for Fast Dynamic Scenes from a Single Event Camera

Zijia Dai[1] Nico Messikommer[2] Rong Zou[2] Nikola Zubić[2] Davide Scaramuzza[2] Laurent Kneip[1]
[1]ShanghaiTech University, China [2]University of Zurich, Switzerland

Abstract

The demand for dynamic 3D assets in AR/VR has recently popularized Deformable Gaussian Splatting. However, traditional RGB cameras are limited in their ability to reconstruct high-speed scenes due to motion blur and low temporal resolution. While event cameras offer a promising alternative, reconstructing a complete scene from their sparse and noisy output is a significant challenge. Existing event-based methods rely on an auxiliary sensor, such as a frame camera, thereby inducing tedious hardware and calibration challenges. We introduce FastEventDGS, a novel Deformable Gaussian Splatting-based framework that leverages a single event camera for high-fidelity 4D reconstruction. Our method utilizes a continuous camera trajectory parametrization and integrates two event generation models to provide both photometric and geometric constraints. We further propose a local patch event motion loss to constrain object motion, effectively mitigating overfitting. To ensure robust reconstruction, we employ an off-the-shelf model for depth correction and apply noise regularization terms in the final stage. We demonstrate robust results on both new synthetic and real-world datasets, highlighting our framework's ability to provide a simplified, event-only solution for high-fidelity 4D reconstruction in dynamic scenes. Our code and dataset are available at `github.com/daizijia/FastEventDGS`.

1. Introduction

Driven by the growing demand for dynamic environments in AR/VR, content creation, and robotics, researchers have extended seminal static novel view synthesis frameworks [21] by incorporating a temporal dimension [13, 19]. This approach allows for the flexible representation of diverse dynamic scenes, aiming to recover their appearance and geometry from multi-view sensor observations with corresponding pose information. Sharp, high-temporal-resolution images are essential for high-fidelity, fast dynamic reconstruction. However, conventional frame-based sensors (RGB and RGB-D cameras) are highly susceptible to motion blur and operate at low temporal resolution [39]. Consequently, frame-based methods are often limited to modeling only moderate motion.

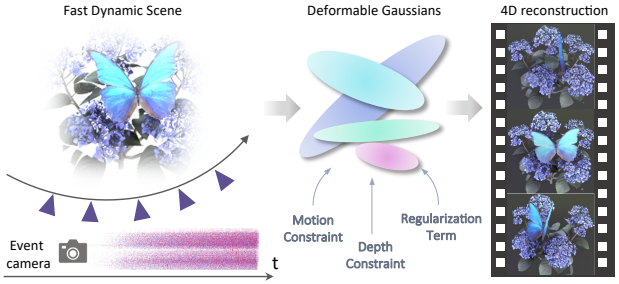


Figure 1. *FastEventDGS* is a novel framework built upon a state-of-the-art Deformable 3D Gaussian Splatting representation, achieving high-fidelity 4D dynamic scene reconstruction from a single event camera stream. This strong performance is attributable to the combined effect of the proposed motion constraint, depth constraint, and regularization term.

Event cameras—or neuromorphic sensors—have recently gained significant attention due to their ability to detect brightness changes asynchronously with microsecond-level latency [7]. This makes them highly effective for capturing high-speed dynamic scenarios. Nevertheless, reconstructing the spatio-temporal representation of a dynamic scene from events is a non-trivial task due to their inherent sparsity, noise, and lack of absolute intensity information. To overcome these obstacles, the majority of event-based novel view synthesis methods leverage RGB frames [32, 41] or perform sensor fusion with Lidar [41] to provide a global contextual prior. To simplify the sensor setup, this work addresses a fundamental research question: *Is it possible to reconstruct a dynamic scene using only a monocular event camera?*

Our approach, named *FastEventDGS*, is a novel framework built upon a state-of-the-art Deformable 3D Gaussian Splatting representation, achieving high-fidelity dynamic 4D scene reconstruction from a single event camera stream. Because event cameras operate at a much higher sensing frequency than standard pose estimation systems (e.g., mo-

tion capture or SLAM), we adopt a continuous representation of the camera trajectory [20]. This enables supervision at arbitrary time intervals between available camera poses, allowing accurate reconstruction of fine-grained temporal motion. Based on this continuous representation, besides the general event generation model used for the brigthness change consistency loss, we propose leveraging a linear event generation model that utilizes rendered image gradients and optical flow to supervise brightness increments. We furthermore exploit the inherent motion information of event cameras by proposing a novel method for supervising the motion of Gaussians, ensuring spatio-temporal consistency with the surrounding event stream. To ground the reconstruction geometrically, we propose leveraging a recent feed-forward scene reconstruction approach, VGGT [26], to correct depth inconsistencies, thereby enhancing the overall quality of the reconstruction. Additionally, regularization terms are introduced to improve robustness against noise and promote a smooth, consistent result. A comprehensive and rigorous evaluation is enabled by two new datasets, comprising both synthetic and real-world data, which will be publicly released.

The main contributions of our work are summarized below:

- We propose a novel event-based Deformable Gaussian Splatting framework for fast dynamic scene reconstruction, which operates on asynchronous data from a single event camera, thereby simplifying the sensor setup for real-world applications.
- We develop our method based on a continuous representation of the camera trajectory, and integrate an event generation model that provides both photometric and geometric constraints by leveraging inter-frame brightness changes and moving edges. Furthermore, we propose a local patch event motion loss to enforce spatio-temporal consistency and mitigate overfitting.
- We refine our reconstruction by leveraging a feed-forward scene reconstruction approach (VGGT) and applying regularization terms in the final optimization stage.

2. Related Works

Novel view synthesis (NVS) for dynamic scenes. Novel view synthesis for dynamic scenes is a challenging task in 3D reconstruction, primarily due to the complexities of modeling object motion and deformation over time. The previous neural radiance fields method, D-NeRF[21], incorporates time as an additional input parameter, enabling the reconstruction of both dynamic and static scenes from a single camera. Recently, NVS methods based on 3D Gaussian Splatting (3DGS) [13] have leveraged their advantage of faster rendering speed to extend explicit representations to dynamic scenes. These methods [29, 31, 33, 34] learn a deformation field or 4D primitives to predict Gaussian deformations at arbitrary timestamps. However, MLP-based neural deformation fields often suffer from overfitting to the training data. Furthermore, their performance typically degrades when encountering complex motion because of the lack of explicit motion guidance. To address these problems, recent works [25, 27, 30, 38] have begun to leverage 2D flow priors to supervise the deformation field of 3DGS, achieving higher rendering quality. Inspired by the importance of motion guidance, our method utilizes a local motion constraint to enforce spatio-temporal consistency between the 3D gaussians and event data, thereby enabling high-fidelity dynamic reconstruction.

Event-based 3D scene reconstruction. Driven by the popularity of neural representations, the reconstruction of 3D scenes from event cameras has become a prominent task in event-based vision. Early works applying the NeRF representation [14, 18, 24] are the first to demonstrate that pure event data can be used to reconstruct a dense 3D representation of a static scene. To overcome the computational bottlenecks of event-based NeRF methods, recent work has shifted toward static 3DGS, including Event-3DGS[9], EventSplat[36], and IncEventGS[12]. In addition to methods that use pure event data, other frameworks [3, 15, 35, 40] focus on mitigating motion blur in RGB frames with the help of the event stream, thus obtaining sharp images for reconstruction. Existing methods for dynamic scene reconstruction with event cameras primarily rely on sensor fusion. For example, [41] combines RGB, depth, and event data to perform Deformable 3D Gaussian Splatting, while others leverage RGB frames together with the event stream [6, 10, 32, 37]. Although some of these works do not apply explicit motion constraints [6, 32, 41], STD-GS [37] uses weak motion guidance by making the distribution of events and Gaussians consistent via spatial histograms.

To the best of our knowledge, our work is the first to successfully tackle the problem of training a Dynamic Gaussian Splatting representation from a monocular event stream.

3. Preliminaries

3.1. Event Generation Model

Event-based cameras, such as [16], are renowned for their low latency, high temporal resolution, and high dynamic range, which enable them to capture high-speed scene changes with minimal latency. These impressive sensing capabilities are possible because event cameras trigger an event at a pixel $\mathbf{x}_k = (x_k, y_k)$ when the brightness change $\Delta I(\mathbf{x}_k, t_k)$ in their logarithmic space $I(x, y) = \log \mathcal{I}(x, y)$ at time t_k reaches a certain threshold during the

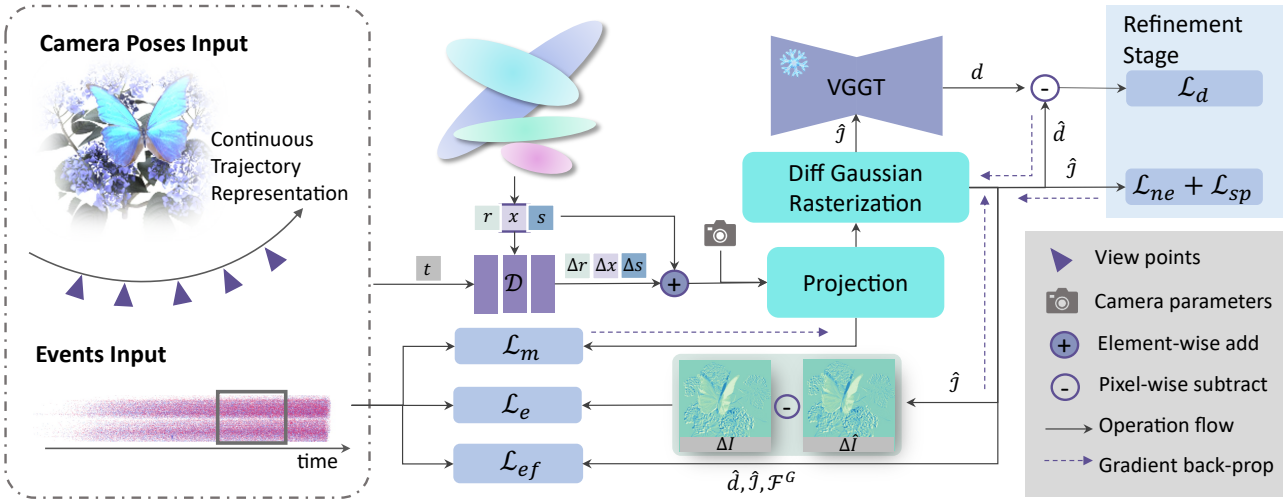


Figure 2. Overview of our proposed *FastEventDGS*. The framework takes a monocular event stream and its continuous camera pose trajectory (modeled via B-spline) as input. The core pipeline is built around Deformable Gaussian Splatting with a temporal dimension. Explicit motion guidance is derived by calculating the temporal motion loss ($\mathcal{L}_m$) and the event flow loss ($\mathcal{L}_{ef}$) to supervise the Gaussian deformation ($\Delta\boldsymbol{r}, \Delta\boldsymbol{x}, \Delta\boldsymbol{s}$). The photometric loss ($\mathcal{L}_e$) provides global consistency. The Refinement Stage improves geometric quality by leveraging VGGT [26], as off-the-shelf depth estimations to enforce a depth loss ($\mathcal{L}_d$), and utilizes texture ($\mathcal{L}_{ne}$) and spatial ($\mathcal{L}_{sp}$) regularization terms to suppress noisy outputs.

past Δt_k, i.e.

$$I(\mathbf{x}_k, t_k) - I(\mathbf{x}_k, t_k - \Delta t_k) = \Delta I(\mathbf{x}_k, t_k) \quad (1)$$

$$\Delta I(\mathbf{x}_k, t_k) = p_k C. \quad (2)$$

The contrast sensitivity threshold $C > 0$ determines the minimum required brightness change, and the polarity $p_k \in \{+1, -1\}$ indicates the sign of the brightness change.

With the assumption of brightness constancy, Equation 1 can be linearized for a short time interval $\Delta\tau$ [8]:

$$\Delta I(\mathbf{x}) \approx -\nabla\mathcal{I}(\mathbf{x}) \cdot \mathbf{v}(\mathbf{x})\Delta\tau, \quad (3)$$

where $\nabla\mathcal{I}(\mathbf{x}) = (\delta_x\mathcal{I}, \delta_y\mathcal{I})^T$ symbolizes the spatial gradients of the brightness image, and $\mathbf{v}(\mathbf{x})$ represents pixel movement (i.e. optical flow).

3.2. Deformable 3D Gaussian Splatting

Deformable Gaussian splatting is renowned for its application in novel view synthesis of dynamic scenes, which focuses on monocular camera inputs. It follows the same concepts as 3D Gaussian Splatting, where the 3D Gaussians are projected into a 2D image plane and rendered on each pixel with the usage of a 2D covariance matrix [13]

$$\Sigma' = JV\Sigma V^T J^T. \quad (4)$$

Here, J is the Jacobian of the projective transformation, V denotes the viewing transform matrix, and Σ symbolizes the 3D covariance matrix. The scaling matrices S and R are used to represent Σ, which simplifies learning of the covariance matrix: $\Sigma = RSS^TR^T$. The color of each pixel $C(\mathbf{p})$ originates from blending N ordered overlapping Gaussians:

$$C(\mathbf{p}) = \sum_i^N c_i\alpha_i \prod_{j=1}^{i-1}(1-\alpha_j). \quad (5)$$

c_i and α_i represent the color and density along the ray, which are obtained from the projected 3D Gaussians.

Compared to the original 3D Gaussian Splatting, Deformable Gaussian Splatting incorporates a temporal dimension, thereby enabling the modeling of scene changes over time. The motion of each Gaussian is learned by a deformation MLP, which transforms the discrete state of time into canonical space [33]. The input of the deformation network is the time t and the position $\boldsymbol{x}$ of the Gaussian. The changes of the scene through time can be obtained from the MLP outputs:

$$(\Delta\boldsymbol{x}, \Delta\boldsymbol{r}, \Delta\boldsymbol{s}) = \mathcal{D}(\boldsymbol{x}, t) \quad (6)$$

where $\boldsymbol{x}$ and t are the center position of the Gaussian and the timestamp, and $\Delta\boldsymbol{x}, \Delta\boldsymbol{r}, \Delta\boldsymbol{s}$ denote changes in position, rotation, and scale over time.

4. Event-Based Deformable Gaussian Splatting

Our framework, which aims for high-fidelity 4D reconstruction from a single event camera, begins with a monocular event stream and its camera poses. We first model the camera trajectory as a continuous representation using B-spline

(§4.1), which is essential for computing the event photometric loss (§4.2) over arbitrary time intervals. We further incorporate flow (§4.2) and motion (§4.3) losses to derive explicit motion guidance. To improve geometric quality, we leverage an off-the-shelf depth estimator (§4.4) for supervision. The final stage also utilizes regularization terms (§4.5) to suppress noisy outputs. An overview of our complete method is shown in Figure 2.

4.1. Trajectory Interpolation

The camera poses, generated by a motion capture system or a Structure from Motion pipeline, are sampled at a significantly lower rate than the events. To establish dense temporal correspondence for supervision, we fit a continuous-time trajectory to the sparse poses using cubic B-spline [20]. This representation enables us to query poses at any time, allowing us to leverage event streams from arbitrary time intervals as a supervision signal. This advantage is critical for the subsequent implementation of the flow loss ($\mathcal{L}_{ef}$) and motion loss ($\mathcal{L}_m$).

4.2. Event Photometric Constancy

Our supervision strategy combines two distinct mechanisms to ensure both global consistency and local detail. The Event Single Integral (ESI) loss provides global supervision by efficiently accumulating scene features through the integration of events over a long time interval. Complementarily, the flow loss is calculated over a small time interval, enabling the preservation of scene details. Derived from the rendered depth and Gaussian flow, this flow loss not only ensures detailed reconstruction but also provides geometry and motion constraints. The details of these losses are as follows:

Event single integral. Due to the inherent lack of absolute intensity information, we leverage the brightness change in the logarithmic space to supervise the training of time-continuous Deformable Gaussian Splatting. This supervisory signal is obtained by rendering two adjacent images at times $t_k - \Delta t_k$ and t_k, and then calculating their difference in the logarithmic space, $\Delta\hat{I}$.

The corresponding ground truth signal, ΔI, is calculated via the temporal integration of events captured by the event camera across the designated time interval:

$$\Delta I = \int_{t_k-\Delta t_k}^{t_k} C \cdot e(\tau)\, d\tau, \quad t_k - \Delta t_k \leqslant \tau \leqslant t_k. \tag{7}$$

This allows us to define a pixel-wise supervision loss, $\mathcal{L}_{esi}$, as the L1 norm of the difference between the predicted and measured brightness changes:

$$\mathcal{L}_{esi} = \left\| \Delta I - \Delta\hat{I} \right\|_1 \odot \mathbf{m}. \tag{8}$$

Here, $\mathbf{m}$ is a binary mask that ensures the loss is only computed for pixels where events were triggered.

We optimize the following loss, $\mathcal{L}_e$, throughout the entire training:

$$\mathcal{L}_e = (1-\lambda_e)\mathcal{L}_{esi}(\Delta I, \Delta\hat{I}) + \lambda_e \mathcal{L}_{D-SSIM}(\Delta I, \Delta\hat{I}), \tag{9}$$

where $\mathcal{L}_{D-SSIM}$ is the structural dissimilarity loss as used in former works [4][12], and λ_e denotes a hyperparameter.

Event flow loss. As detailed in Section 3.1, event generation can be approximated by pixel velocity $\mathbf{v}(\mathbf{x})$ and image gradient $\nabla\hat{\mathcal{I}}(\mathbf{x})$ over a small time interval $\Delta\tau$. Inspired by [38], we derive $\mathbf{v}(\mathbf{x})$ from the optical flow, which is decomposed into camera flow $\mathcal{F}^C_{t,t+\Delta\tau}$ and Gaussian flow $\mathcal{F}^G_{t,t+\Delta\tau}$. The camera flow represents the flow caused solely by camera motion under a static scene and is calculated using the rendered depth map d_t and known camera extrinsics T_t. We have

$$p_t^{t+\Delta\tau} = \pi(K, T_{t+\Delta\tau}, \pi^{-1}(K, T_t, d_t, p_t)) \tag{10}$$

$$\mathcal{F}^C_{t,t+\Delta\tau} = p_t^{t+\Delta\tau} - p_t, \tag{11}$$

where K denotes the intrinsic parameters of the camera. The function π indicates the camera projection from a 3D projective space to a 2D projective space, and π^{-1} is the inverse projection. The Gaussian flow $\mathcal{F}^G$ is the 2D projection of the Gaussian deformation during a short time interval $\Delta\tau$ from the same camera viewpoint. In our implementation, both the Gaussian deformation and the depth map at a specific time t are obtained via a differentiable rasterizer [38]. The optical flow $\mathcal{F}^O$ is then represented as:

$$\mathcal{F}^O_{t,t+\Delta\tau} = \mathcal{F}^C_{t,t+\Delta\tau} + \mathcal{F}^G_{t,t+\Delta\tau}. \tag{12}$$

We utilize this optical flow to construct a photometric loss that supervises brightness increments over small time intervals. This event flow loss $\mathcal{L}_{ef}$ is normalized to eliminate the impact of $\Delta\tau$ and the contrast threshold:

$$\mathcal{L}_{ef} = \left\| \frac{\Delta I'}{\|\Delta I'\|_2} - \frac{\Delta\hat{I}(x; \mathcal{F}^O_{t,t+\Delta\tau})}{\|\Delta\hat{I}(x; \mathcal{F}^O_{t,t+\Delta\tau})\|_2} \right\|_1, \tag{13}$$

where $\Delta\hat{I}(\mathbf{x}; \mathcal{F}^O_{t,t+\Delta\tau}) = -\nabla\hat{\mathcal{I}}(\mathbf{x}) \cdot \mathcal{F}^O_{t,t+\Delta\tau} \cdot \Delta\tau$ and $\nabla\hat{\mathcal{I}}(\mathbf{x})$ is the gradient of the rendered intensity image. $\Delta I'$ represents the integration of events during a short time interval $\Delta\tau$, following Equation 7.

4.3. Local Patch Motion Constraint

A common failure case is the overfitting to the training views, resulting in some Gaussians failing to move coherently with the objects to which they belong. Therefore, we propose an event motion loss that enforces the Gaussian trajectories to be consistent with the event edges, as

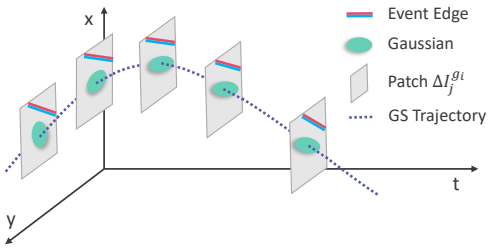


Figure 3. The illustration of our proposed motion loss. It ensures that individual 2D projections of Gaussians do not move far from the neighboring event edges over a short time period.

these edges follow the same underlying object motion. The fundamental idea behind this loss is to ensure that individual 2D projections of Gaussians do not move far from the neighboring event edges over a short time period, as illustrated in Figure 3.

First, we split a time interval Δt that starts at time t into m event sequences of an equal number of events. Subsequently, we project the mean of each Gaussian g^i onto the 2D image plane:

$$g^i = \pi(K, T_t, (\boldsymbol{x} + \Delta \boldsymbol{x})), i \in \{1, ..., n\}. \tag{14}$$

When starting the optimization of the deformation network, we randomly sample a fixed number of Gaussians n and apply a depth filter to exclude occlusion cases, i.e., Gaussians occluded by others are disregarded. We then select a square patch of fixed size in the 2D image plane, centered at the projection of the sampled Gaussians. The successive m patches following the same Gaussian center, $\Delta I_j^{g_i}$, are calculated via the single integral of events over the corresponding time interval $[t_j, t_{j+1}]$. The final motion loss is aggregated by summing the local patch residuals across all sampled individual Gaussians:

$$\mathcal{L}_m = \sum_i \sum_j \|\Delta I_j^{g_i} - \Delta I_{j+1}^{g_i}\|_1, j \in \{1, ..., m\}. \tag{15}$$

This loss ensures that the motion of Gaussians is consistent with the event trajectory in the temporal domain. For Gaussians belonging to the static background, the loss penalizes overfitting-induced trembling behavior. Conversely, for Gaussians associated with dynamic objects, it imposes a rigid constraint to enforce coherent group motion.

4.4. Expert Model-based Depth Optimization

During optimization, we observe that the model can render high-quality intensity images after a few iterations. However, due to the monocular nature of the input and the limited number of viewpoints, some Gaussians tend to float in flat areas, leading to depth inconsistencies. This issue suggests the need for relative depth constraints to encourage Gaussians on the same plane to maintain a consistent depth. Therefore, we utilize a state-of-the-art scene reconstruction framework [26] as a depth estimator to refine the geometry. We optimize the depth using a scale-invariant log loss (SiLog) [5]:

$$\mathcal{L}_d = \frac{1}{n} \sum_i \alpha_i^2 - \frac{1}{n^2} \left(\sum_i \alpha_i \right)^2, \tag{16}$$

where $\alpha_i = \log d_i - \log \hat{d}_i$ is the log difference between rendered depth $\hat{d}_i$ and estimated depth d_i.

4.5. Regularization for Noise Resiliance

During optimization, the model exhibits artifacts in textureless or uniform regions, which correspond to areas with sparse or no event data. Moreover, trailing events are a common issue in real-world datasets [11, 17], often leading to noisy reconstructions. Therefore, we introduce two regularization terms to alleviate this problem.

Non-Event Region Constraint. While events are located in areas of brightness change, the non-event regions also provide a crucial constraint: the change in brightness in these areas should not exceed a certain threshold [1]. We leverage this property by introducing a non-event region constraint $\mathcal{L}_{ne}$:

$$\mathcal{L}_{ne} = ReLU(|\Delta \hat{I}| - C) \odot \neg \mathbf{m}, \tag{17}$$

Here, $\neg \mathbf{m}$ represents the non-event region. This loss provides additional supervision specifically for non-event regions.

Event spatial loss. Given the event camera's inherent characteristic of providing only relative brightness changes across time intervals, the resulting absence of absolute scene intensity information often leads to unnatural artifacts in the reconstruction. To mitigate this issue, following [28], we also adopt a spatial gradient loss $\mathcal{L}_{sp}$:

$$\mathcal{L}_{sp} = \| \delta_x \mathcal{I} \|_1 + \| \delta_y \mathcal{I} \|_1 . \tag{18}$$

This loss penalizes excessive spatial gradients in the rendered image, which helps to mitigate noise.

4.6. The Complete Training Pipeline

Combining all individual loss components discussed above, we define the ultimate loss function $\mathcal{L}$ used for training our framework as:

$$\mathcal{L} = \mathcal{L}_e + \lambda_{ef} \mathcal{L}_{ef} + \lambda_m \mathcal{L}_m + \lambda_d \mathcal{L}_d + \lambda_{ne} \mathcal{L}_{ne} + \lambda_{sp} \mathcal{L}_{sp}, \tag{19}$$

Table 1. Quantitative comparison of different methods on the synthetic BlenderDynamicEvent dataset.

	Butterfly			Duck			Alarm			Ball		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Event3GS[13]	14.46	0.6464	0.2997	16.21	0.8025	0.5104	16.47	0.7578	0.3492	15.46	0.5889	0.6963
EvDNeRF[2]	12.10	0.4635	0.6310	8.18	0.5162	0.8485	-	-	-	16.64	0.5764	0.7322
E2vidDGS[23]	16.13	0.7615	0.2737	19.54	0.8526	0.2687	16.61	0.7790	0.3099	18.74	0.6918	0.3869
Ours	**24.27**	**0.9020**	**0.1195**	**21.25**	**0.8914**	**0.2435**	**23.24**	**0.8986**	**0.1466**	**22.89**	**0.8763**	**0.2714**

where $\lambda_{ef}, \lambda_m, \lambda_d, \lambda_{ne}, \lambda_{sp}$ represent the hyperparameters that weigh the contribution of their corresponding loss components.

Among these loss functions, $\mathcal{L}_e$ serves as the fundamental loss component and is applied throughout the entire training process. We begin to use the motion-related losses ($\mathcal{L}_{ef}$ and $\mathcal{L}_m$) after the initial warm-up stage, which optimizes the position, rotation, and scale of the Gaussians as vanilla Gaussian Splatting. Subsequently, once the model renders a relatively high-quality image, the remaining terms ($\mathcal{L}_d, \mathcal{L}_{ne}, \mathcal{L}_{sp}$) are introduced in the final refinement stage.

5. Experiments

Dataset Given the absence of event camera datasets for fast dynamic scenes, we introduce two new datasets: BlenderDynamicEvent and Gen4Dynamic.

- **BlenderDynamicEvent.** We construct a synthetic dataset using Blender and the ESIM simulator [22], featuring four distinct scenes. For each scene, we render a sequence of 4.5-6 seconds at a rate of a thousand frames per second. Color events are generated by applying a Bayer filter. The spatial resolution of most sequences is 400×400, except the *Duck* scene, which is rendered at 600×400.
- **Gen4Dynamic.** To acquire a real-world dataset, we record two fast-moving objects using a well-calibrated and synchronized setup consisting of a Prophesee Gen4 event camera and a color FLIR Blackfly S camera (10 fps). The dataset features a spatial resolution of 945×649 and contains accurate camera poses provided by a motion capture system.

More detailed information regarding the construction and content of our datasets is provided in the supplementary material.

Implementation Details. All experiments are conducted on a machine equipped with an NVIDIA RTX 4090 GPU and a 16-core AMD EPYC 9354 CPU. For each experiment, we train the model for 30,000 epochs to obtain the reported results. The initial warm-up stage is set to 3,000 epochs. Following this, we enter the refinement stage after 20,000 epochs, at which point the depth constraints and regularization terms are introduced. For testing, intermediate frames located between two training viewpoints are used. Hyperparameters and Gaussian initialization are detailed in the supplementary material.

Baselines. To the best of our knowledge, our framework, *FastEventDGS*, is the first method focusing on fast dynamic scene reconstruction using only a single event camera. Thus, a direct comparison to another open-source, event-only Deformable Gaussian Splatting framework is not possible. Nevertheless, we establish a robust comparative analysis by selecting four baselines applicable in related settings. We compare our method against EvDNerf [2], a NeRF-based method that also reconstructs dynamic scenes using multi-view event streams, allowing for a comparison across different representations. To illustrate the importance of explicitly modeling a motion field, we further compare against Event3GS, which is implemented upon vanilla 3GS[13]. As a third baseline, we select E2VID-DGS [23], a two-stage approach that first uses E2VID to reconstruct frames before applying a general DGS framework. Finally, Frame-DGS[33] serves as a baseline that utilizes motion-blurred RGB frames for scene reconstruction, highlighting the advantages of event data in dynamic settings with fast-moving objects.

Evaluation Metrics. We rely on established image quality metrics for a quantitative comparison. Specifically, for the synthetic dataset with available ground truth, we report the PSNR and SSIM for fidelity assessment, along with the LPIPS to evaluate perceptual quality. In the case of the real-world dataset, the low temporal resolution of the RGB camera combined with rapid object motion precludes the capture of reliable ground truth. Consequently, our comparison with each baseline on the real-world dataset relies primarily on qualitative analysis. Following EventNerf[24], before calculating the evaluation metrics, each results is processed using a fitted single linear color transform with the ground truth image in the logarithmic space to mitigate the influence of color balance.

5.1. Benchmark Comparison

Synthetic Dataset. The quantitative results for the synthetic dataset are reported in Table 1 while the qualitative results are shown in Figure 4. As can be observed, our method significantly outperforms all baselines in both quantitative

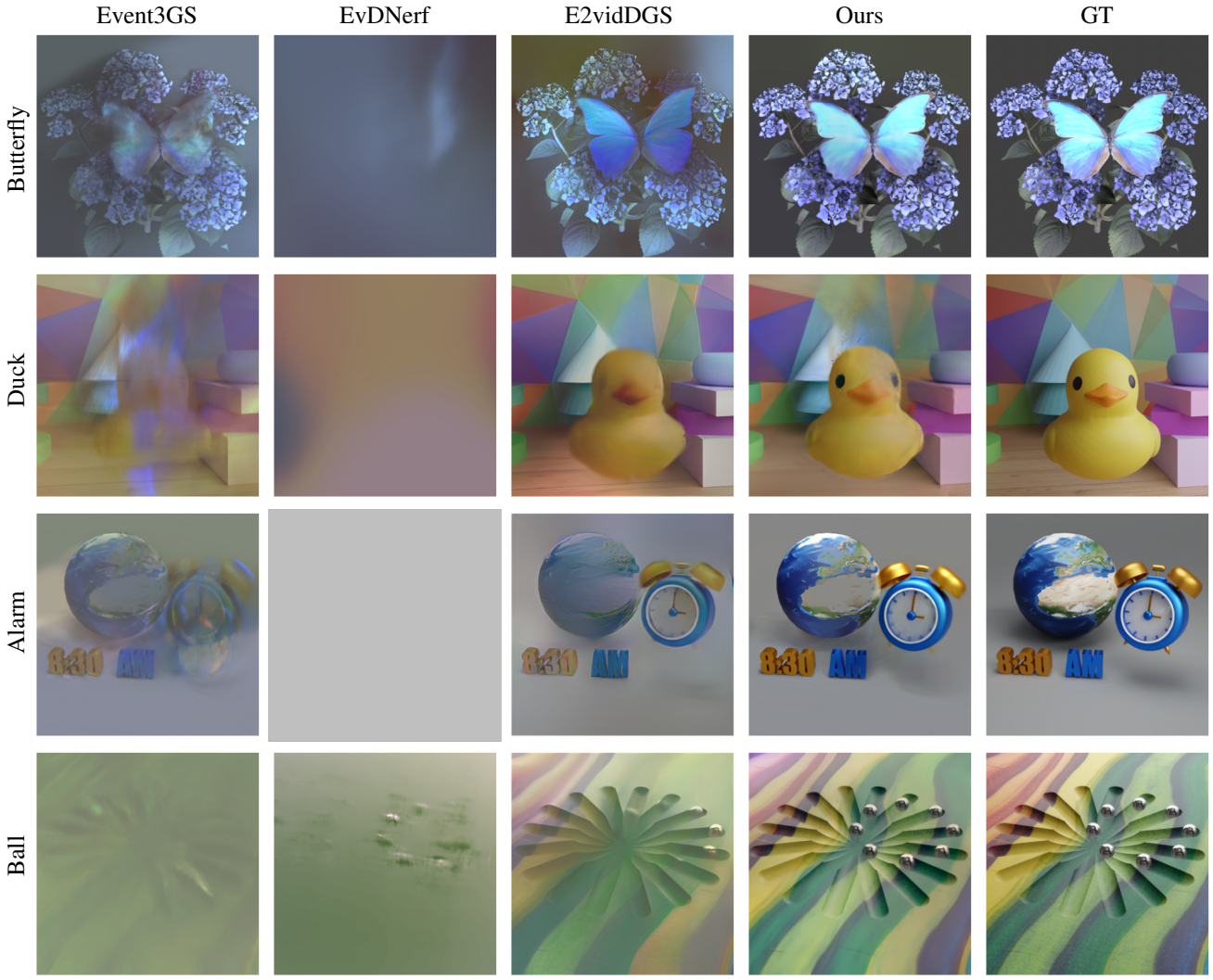


Figure 4. Qualitative evaluation for novel view image synthesis on our synthetic BlenderDynamicEvent dataset. The results show that our method consistently outperforms all baselines.

metrics and visual reconstruction quality. Event3GS only reconstructs static elements, yielding blurred results for dynamic regions due to the absence of a learned motion field. Moreover, EvDNerf struggles with the monocular and fast-motion characteristics of our dataset, leading to a failure in scene reconstruction. Finally, despite its high reconstruction performance on simpler scenes (Duck), the two-stage E2vidDGS exhibits limitations when encountering complex scene structures, where its limited generalization capacity leads to unnatural colors and structural degradation.

Real-world Dataset. The qualitative results in Figure 5 confirm that our framework excels at reconstructing the dynamic parts of the scene, especially compared to the FrameDGS method. Since FrameDGS relies on an absolute brightness signal, it typically performs better at preserving the details of static scene regions. However, despite setting the RGB camera frame rate to 10fps, it still misses high-speed dynamic information, placing dynamic objects at incorrect locations.

The sequence duration is only 0.5 seconds, with the object's drop motion lasting merely 0.25 seconds. Despite this short time interval, we successfully reconstruct the object along with its background. Conversely, the high temporal resolution property of the event camera, which our method fully leverages, allows for a more accurate reconstruction of dynamic objects, resulting in superior temporal consistency.

5.2. Ablation Studies

We conduct our ablation studies on the synthetic dataset to explore two key areas: the impact of object motion speed and the effectiveness of each proposed loss component.

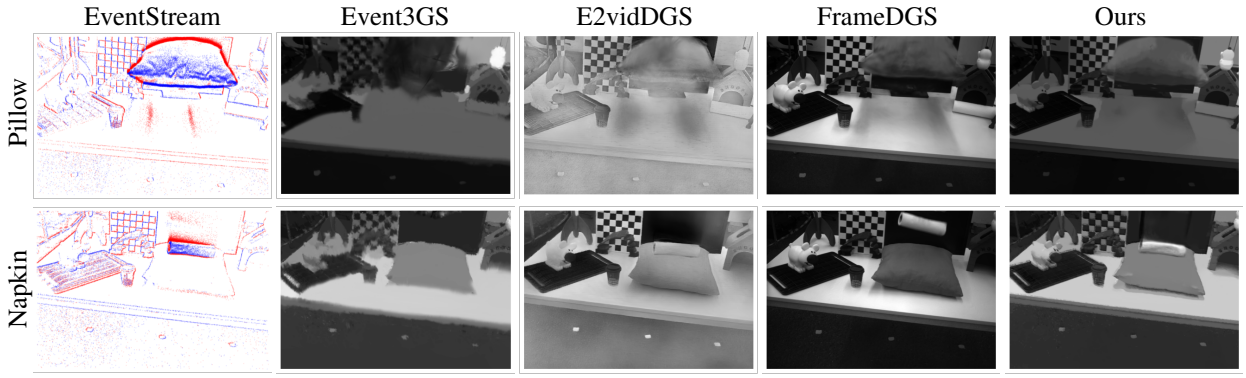


Figure 5. Qualitative evaluation for novel view image synthesis on our real-world Gen4Dynamic dataset. Despite lacking an absolute brightness signal, our framework excels at reconstructing the dynamic parts of the scene, a benefit derived from the high temporal resolution property of the event camera.

Table 2. Ablation study on the object motion speed.

	PSNR↑	SSIM↑	LPIPS↓
1×	22.91	0.8921	0.1952
2×	19.51	0.8456	0.2369
3×	18.43	0.8361	0.2532
4×	18.31	0.8308	0.2578

Object Motion Speed. We control the object motion speed within this experiment by reducing the initial frame rate used to simulate the event stream by four speed factors, ranging from 1 to 4. The results presented in Table 2 show a clear trend: as the motion speed increases, the PSNR decreases. This drop is primarily due to the diminished information available for estimating relative intensity changes, which causes a noticeable color bias in the rendered images. However, the SSIM does not decrease significantly, confirming that our model can still reconstruct images with high structural integrity despite the color shift. Figure 6 provides a visual comparison of our method for the *Ball* scene under these different motion speeds.

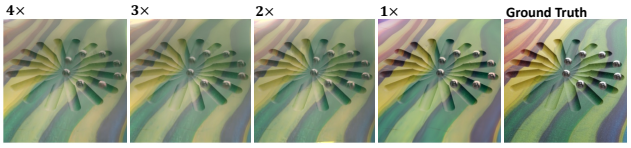


Figure 6. Comparison of different motion speeds.

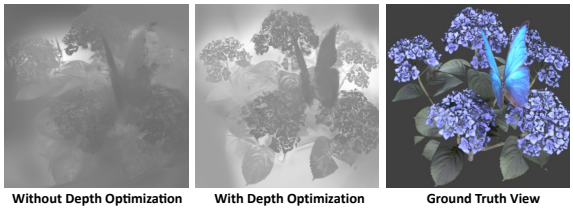


Figure 7. Comparison of depth reconstruction.

Table 3. Ablation study on the proposed loss components.

	PSNR↑	SSIM↑	LPIPS↓
$\mathcal{L}_e$	18.61	0.8478	0.2267
$\mathcal{L}_e + \mathcal{L}_{ef}$	19.35	0.8570	0.2231
$\mathcal{L}_e + \mathcal{L}_{ef} + \mathcal{L}_m$	20.52	0.8692	0.2160
$\mathcal{L}_e + \mathcal{L}_{ef} + \mathcal{L}_m + \mathcal{L}_d$	22.49	0.8911	0.1944
All	22.91	0.8921	0.1952

Effectiveness of Losses. As evident from the results reported in Table 3, the depth constraint makes the most significant contribution to the overall performance. This can be observed in Figure 7, where the depth maps rendered from the trained Gaussian model demonstrate that depth optimization significantly improves the geometric structure, producing much clearer depth maps. In addition to the depth loss, the flow loss ($\mathcal{L}_{ef}$) and motion loss ($\mathcal{L}_m$) contribute nearly equally to the reconstruction quality. Finally, even if it has the most negligible effect, the regularization terms ($\mathcal{L}_{ne} + \mathcal{L}_{sp}$) still lead to a consistent improvement, as can be seen by the performance of all losses(All).

6. Conclusion

This paper presents *FastEventDGS*, a novel event-based Deformable Gaussian Splatting framework for fast dynamic scene reconstruction. By operating on asynchronous data from a single event camera, our method demonstrates the high potential for utilizing event cameras in future research, particularly for the NVS of fast dynamic scenes.

Limitation Our current method relies on known camera poses, which introduces practical inconvenience during deployment. Future work should focus on developing a technique that is agnostic to camera poses, simplifying the data acquisition process.

Acknowledgement

The authors would like to thank the fund support from Natural Science Foundation of China(W2531052).

References

[1] Patrick Bardow, Andrew J Davison, and Stefan Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 884–892, 2016. 5

[2] Anish Bhattacharya, Ratnesh Madaan, Fernando Cladera, Sai Vemprala, Rogerio Bonatti, Kostas Daniilidis, Ashish Kapoor, Vijay Kumar, Nikolai Matni, and Jayesh K Gupta. Evdnerf: Reconstructing event data with dynamic neural radiance fields. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5846–5855, 2024. 6

[3] Marco Cannici and Davide Scaramuzza. Mitigating motion blur in neural radiance fields with events and frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9286–9296, 2024. 2

[4] Hiroyuki Deguchi, Mana Masuda, Takuya Nakabayashi, and Hideo Saito. E2gs: Event enhanced gaussian splatting. In *2024 IEEE International Conference on Image Processing (ICIP)*, pages 1676–1682. IEEE, 2024. 4

[5] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014. 5

[6] Chaoran Feng, Zhenyu Tang, Wangbo Yu, Yatian Pang, Yian Zhao, Jianbin Zhao, Li Yuan, and Yonghong Tian. E-4dgs: High-fidelity dynamic reconstruction from the multi-view event cameras. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 7356–7365, 2025. 2

[7] Guillermo Gallego, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J. Davison, Jorg Conradt, Kostas Daniilidis, and Davide Scaramuzza. Event-based vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):154–180, 2022. 1

[8] Daniel Gehrig, Henri Rebecq, Guillermo Gallego, and Davide Scaramuzza. Asynchronous, photometric feature tracking using events and frames. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 750–765, 2018. 3

[9] Hanqian Han, Jianing Li, Henglu Wei, and Xiangyang Ji. Event-3dgs: Event-based 3d reconstruction using 3d gaussian splatting. *Advances in Neural Information Processing Systems*, 37:128139–128159, 2024. 2

[10] Junhao He, Jiaxu Wang, Jia Li, Mingyuan Sun, Qiang Zhang, Jiahang Cao, Ziyi Zhang, Yi Gu, Jingkai Sun, and Renjing Xu. Degs: Deformable event-based 3d gaussian splatting from rgb and event stream. *IEEE Transactions on Visualization and Computer Graphics*, 32(2):1698–1712, 2026. 2

[11] Yuhuang Hu, Shih-Chii Liu, and Tobi Delbruck. v2e: From video frames to realistic dvs events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1312–1321, 2021. 5

[12] Jian Huang, Chengrui Dong, Xuanhua Chen, and Peidong Liu. Inceventgs: Pose-free gaussian splatting from a single event camera. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26933–26942, 2025. 2, 4

[13] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 1, 2, 3, 6

[14] Simon Klenk, Lukas Koestler, Davide Scaramuzza, and Daniel Cremers. E-nerf: Neural radiance fields from a moving event camera. *IEEE Robotics and Automation Letters*, 8 (3):1587–1594, 2023. 2

[15] Seungjun Lee and Gim Hee Lee. Diet-gs: Diffusion prior and event stream-assisted motion deblurring 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21739–21749, 2025. 2

[16] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43 (2):566–576, 2008. 2

[17] Haoyue Liu, Shihan Peng, Lin Zhu, Yi Chang, Hanyu Zhou, and Luxin Yan. Seeing motion at nighttime with an event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25648–25658, 2024. 5

[18] Weng Fei Low and Gim Hee Lee. Robust e-nerf: Nerf from sparse & noisy events under non-uniform motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18335–18346, 2023. 2

[19] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1

[20] Hartmut Prautzsch, Wolfgang Boehm, and Marco Paluszny. *Bézier and B-spline techniques*. Springer Science & Business Media, 2002. 2, 4

[21] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10318–10327, 2021. 1, 2

[22] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. ESIM: an open event camera simulator. *Conf. on Robotics Learning (CoRL)*, 2018. 6

[23] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):1964–1980, 2019. 6

[24] Viktor Rudnev, Mohamed Elgharib, Christian Theobalt, and Vladislav Golyanik. Eventnerf: Neural radiance fields from a

single colour event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4992–5002, 2023. 2, 6

[25] Su Sun, Cheng Zhao, Zhuoyang Sun, Yingjie Victor Chen, and Mei Chen. Splatflow: Self-supervised dynamic gaussian splatting in neural motion flow field for autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 27487–27496, 2025. 2

[26] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025. 2, 3, 5

[27] Shizun Wang, Xingyi Yang, Qiuhong Shen, Zhenxiang Jiang, and Xinchao Wang. Gflow: Recovering 4d world from monocular video. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7862–7870, 2025. 2

[28] Zipeng Wang, Yunfan Lu, and Lin Wang. Revisit event generation model: Self-supervised learning of event-to-video reconstruction with implicit neural representations. In *European Conference on Computer Vision*, pages 321–339. Springer, 2024. 5

[29] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20310–20320, 2024. 2

[30] Liuyue Xie, Joel Julin, Koichiro Niinuma, and Laszlo Attila Jeni. Gaussian splatting lucas-kanade. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[31] Jiawei Xu, Zexin Fan, Jian Yang, and Jin Xie. Grid4d: 4d decomposed hash encoding for high-fidelity dynamic gaussian splatting. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2

[32] Wenhao Xu, Wenming Weng, Yueyi Zhang, Ruikang Xu, and Zhiwei Xiong. Event-boosted deformable 3d gaussians for dynamic scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 28334–28343, 2025. 1, 2

[33] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20331–20341, 2024. 2, 3, 6

[34] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2024. 2

[35] Wangbo Yu, Chaoran Feng, Jianing Li, Jiye Tang, Jiashu Yang, Zhenyu Tang, Meng Cao, Xu Jia, Yuchao Yang, Li Yuan, et al. Evagaussians: Event stream assisted gaussian splatting from blurry images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24780–24790, 2025. 2

[36] Toshiya Yura, Ashkan Mirzaei, and Igor Gilitschenski. Eventsplat: 3d gaussian splatting from moving event cameras for real-time rendering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26876–26886, 2025. 2

[37] Hanyu Zhou, Haonan Wang, Haoyue Liu, Yuxing Duan, Luxin Yan, and Gim Hee Lee. Std-gs: Exploring frame-event interaction for spatiotemporal-disentangled gaussian splatting to reconstruct high-dynamic scene. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24801–24810, 2025. 2

[38] Ruijie Zhu, Yanzhe Liang, Hanzhi Chang, Jiacheng Deng, Jiahao Lu, Wenfei Yang, Tianzhu Zhang, and Yongdong Zhang. Motiongs: Exploring explicit motion guidance for deformable 3d gaussian splatting. *Advances in Neural Information Processing Systems*, 37:101790–101817, 2024. 2, 4

[39] Michael Zollhöfer, Patrick Stotko, Andreas Görlitz, Christian Theobalt, Matthias Nießner, Reinhard Klein, and Andreas Kolb. State of the art on 3d reconstruction with rgb-d cameras. In *Computer graphics forum*, pages 625–652. Wiley Online Library, 2018. 1

[40] Rong Zou, Marco Cannici, and Davide Scaramuzza. Event-aided sharp radiance field reconstruction for fast-flying drones. *IEEE Transactions on Robotics*, pages 1–16, 2026. 2

[41] Zihao Zou, Ziyuan Qu, Xi Peng, Vivek Boominathan, Adithya Pediredla, and Praneeth Chakravarthula. High-speed dynamic 3d imaging with sensor fusion splatting, 2025. 1, 2