DeblurSplat: SfM-free 3D Gaussian Splatting with Event Camera for Robust Deblurring

Pengteng Li, Yunfan Lu, Pinhao Song, Weiyu Guo, Huizai Yao, F. Richard Yu, *Fellow, IEEE*, and Hui Xiong, *Fellow, IEEE*

arXiv:2509.18898v1 [cs.CV] 23 Sep 2025

***Abstract*—In this paper, we propose the first Structure-from-Motion (SfM)-free deblurring 3D Gaussian Splatting method via event camera, dubbed *DeblurSplat*. We address the motion-deblurring problem in two ways. First, we leverage the pretrained capability of the dense stereo module (DUSt3R) to directly obtain accurate initial point clouds from blurred images. Without calculating camera poses as an intermediate result, we avoid the cumulative errors transfer from inaccurate camera poses to the initial point clouds' positions. Second, we introduce the event stream into the deblur pipeline for its high sensitivity to dynamic change. By decoding the latent sharp images from the event stream and blurred images, we can provide a fine-grained supervision signal for scene reconstruction optimization. Extensive experiments across a range of scenes demonstrate that DeblurSplat not only excels in generating high-fidelity novel views but also achieves significant rendering efficiency compared to the SOTAs in deblur 3D-GS.**

I. Introduction

Scene reconstruction and novel-view synthesis (NVS) from 2D image collections remain fundamental yet challenging problems in computer vision and computer graphics. While point-based representations, such as 3D-GS [1] and NeRF [2], have achieved impressive results in generating high-fidelity novel views, they typically assume high-quality inputs captured in ideal conditions. Unfortunately, real-world captures often contain motion blur [3] due to low-light or long-exposure settings, leading to inaccurate geometry estimation across multiple views. As a result, the performance of both NeRF and 3D-GS deteriorates significantly in blurred scenarios.

To address this issue, researchers have explored deblur-based NeRF and 3D-GS approaches that explicitly model the latent physical blur process [6], [7], [8], [3]. Recently, the imaging characteristics of event cameras—such as no motion blur and high dynamic range—have made them particularly attractive for deblurring tasks [9], [10], [11], [12], [13], [14], [15]. By leveraging the physical imaging process [16], event-based methods can recover clearer scene details, outperforming purely RGB-based pipelines.

Even when event cameras are employed, the overall reconstruction quality is often constrained by the inaccurate initialization provided by traditional Structure-from-Motion (SfM) pipelines (e.g., COLMAP). Motion-blurred images tend to smear out edges and flatten texture details, creating extensive regions of semantically similar pixels [12], [8]. This confuses the feature detection and matching process, yielding fewer valid correspondences and a higher likelihood of mismatches. As a result, camera pose estimation is inaccurate, leading to noticeable misalignment across multiple views. Since 3D-GS and NeRF frameworks rely heavily on precise multi-view alignment to fuse geometry and appearance, these pose errors propagate into the final reconstruction, manifesting as distorted geometry, incomplete surfaces, and artifacts in the rendered views. Consequently, the initial point clouds generated by COLMAP are often sparse or inaccurately positioned, further undermining the performance of the entire deblurring pipeline. Moreover, using SfM to estimate blur scenes results with several images in longer training time consuming and is prone to failure, which we conduct relative experiments as shown in Figure 4.

Recently, SfM-free novel-view synthesis approaches [17], [5], [18], [19], [20] have emerged, showing potential to circumvent SfM pitfalls by directly estimating scene geometry in an end-to-end manner. In particular, DUSt3R [5] employs a dense stereo module pretrained on large-scale datasets, demonstrating greater robustness in producing point clouds

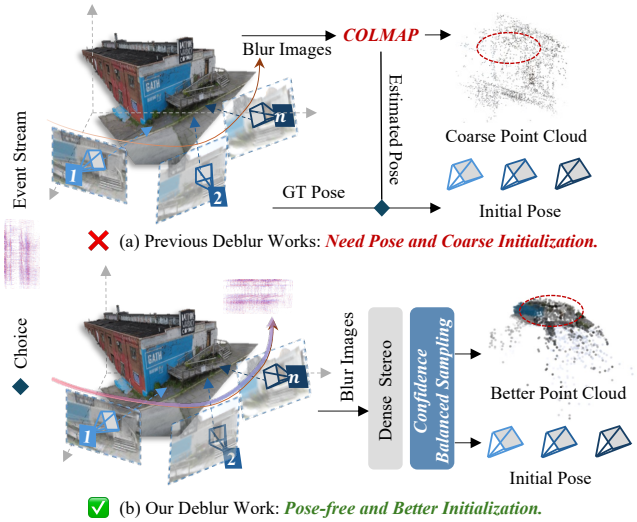
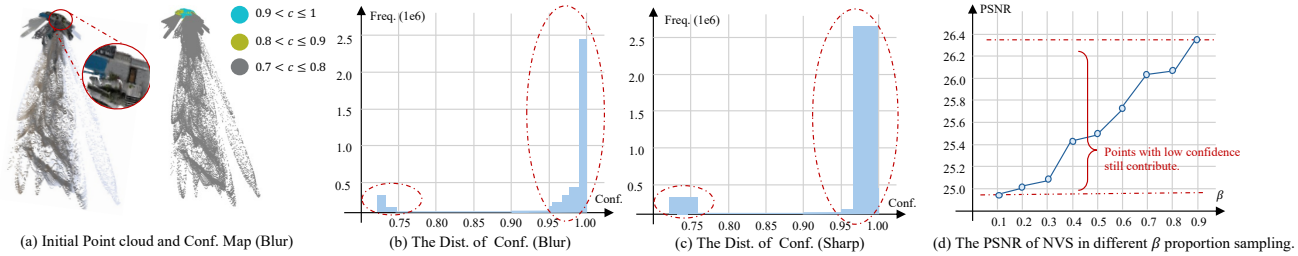


Fig. 1. Illustration of differences between our work and previous works for deblur reconstruction. (a) Most previous works directly estimate camera poses and point clouds by COLMAP [4], leading to coarse point clouds initialization and poses due to the blur geometry uncertainty. (b) Our work mainly utilize sufficient semantic prior from the dense stereo module (DUSt3R [5]) to initialize better point clouds and camera poses, while leverages event streams to guarantee the overall optimization.

Corresponding authors: Hui Xiong.
Pengteng Li, Yunfan Lu, Weiyu Guo, Huizai Yao and Hui Xiong are with the Thust of the Artificial Intelligence, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511453, China (email: {pli807, ylu066, wguo395, hyao032}@connect.hkust-gz.edu.cn), xionghui@ust.hk)
Pinhao Song is with the KU Leuven (email: pinhao.song@kuleuven.be).
F. Richard Yu is with Carleton University (e-mail: richard.yu@carleton.ca).



(a) Initial Point cloud and Conf. Map (Blur) (b) The Dist. of Conf. (Blur) (c) The Dist. of Conf. (Sharp) (d) The PSNR of NVS in different β proportion sampling.

Fig. 2. Empirical study of initial point clouds generated from blur images by DUSt3R. Here, "Freq." denotes frequency, "Dist." denotes distribution, "Conf." denotes confidence. (d) means that we sample the points that have a confidence level greater than their corresponding quantile threshold of $1-\beta$, which we sample a proportion of the point cloud equal to β in total.

even under challenging conditions. This motivates us to adopt DUSt3R for more accurate and robust initialization when dealing with motion-blurred imagery.

Building on these insights, we propose **DeblurSplat**, the first SfM-free motion-deblurring framework for 3D-GS that integrates event-camera priors with DUSt3R's robust initialization (see Figure 1). To sufficiently excavate the information from DUSt3R, we introduce a Confidence Balanced Sampling strategy, which considers the confidences of points as well as maintains a basic spatial information of the scene. Furthermore, to tackle residual defects caused by motion-blur, we propose Progressive Alignment, which exploits event-camera information [10], [11], [12], [13], [14], [15] to iteratively refine motion-blurry pixels by aligning them with clearer semantics derived from event streams. To further evaluate the method performance in more complicated motion-blur degrees, we propose 7 complex scenes crafted from Blender which consist of difference texture, diverse lightning condition and multiple objects compared to normal datasets [6]. Finally, we also design different motion-blur degrees real-world datasets by adjusting the exposure time for comprehensive evaluation. Finally, our contributions can be listed as follows:

- We propose DeblurSplat, a SfM-free 3D-GS framework designed for robust deblurring, which effectively integrates the extensive prior knowledge from DUSt3R and the high dynamic range information provided by event cameras. As we known, this work constitutes the first attempt to leverage a dense stereo module to tackle the challenge of motion-deblurring in 3D reconstruction.
- We propose an initial point clouds sampling strategy named Confidence Balanced Sampling that leverages DUSt3R's confidence estimates while preserving necessary spatial coverage.
- We propose Progressive Alignment to combine the cues from event stream for an accurate scene reconstruction. By estimating the latent sharp images and latent camera poses, we provide a fine-grained guidance for scene reconstruction optimization.
- Extensive experiments on diverse scenes demonstrate that DeblurSplat achieves SOTA results and outperforms motion-deblur reconstruction counterparts significantly.

II. Related Work

A. Motion-Deblur Reconstruction

Motion blur is a frequent occurrence in real-world imaging scenarios and poses a significant challenge for accurate 3D scene reconstruction. Efforts to address this issue have progressed in both the RGB domain and the event-based domain. In the RGB domain, early methods [6], [7] learned a blur formation kernel to approximate the image blurring process, while subsequent approaches [3], [8] integrated blur modeling with bundle adjustment for deblurring. Nevertheless, even when synthesizing motion blur, recovering precise texture details across multiple views remains non-trivial. By contrast, event cameras provide asynchronous event streams with several notable advantages, including a high dynamic range (>120dB vs. 60dB in conventional sensors) [9], which has attracted considerable attention for deblurring. Early event-based methods [21], [22], [23], [24] convert event streams into intensity images or temporal differences, using these signals to guide novel-view synthesis. Later works [25], [26], [27] incorporate additional event-imaging factors, such as event-motion. More recently, approaches [10], [28], [14], [29] integrate motion-blur modeling [3], [8] with event-camera decoupling [12], [30] to jointly optimize latent poses.

Despite these advances, most event-based deblurring methods still rely on SfM (e.g., COLMAP) for camera pose estimation, thereby obtaining initial point clouds. Under blurred conditions, however, such pose estimates are prone to inaccuracies, leading to cumulative errors in the resulting point clouds. Furthermore, the latent-pose optimization process can be computationally expensive; for instance, works like [10], [28], [12] must first estimate camera poses, which adds significant overhead. In contrast, our proposed method leverages the powerful, pretrained DUSt3R to directly generate initial point clouds, eliminating the need for pose estimation from blurred inputs. By effectively combining these point clouds with event information, our framework achieves more accurate and efficient 3D scene reconstruction under motion blur.

B. SfM-free Novel-view Synthesis

Recent advances in novel-view synthesis from uncalibrated images have been greatly facilitated by progress in Multi-view Stereo (MVS) methods [31], [20], [19], [32]. Cutting-edge MVS approaches [5], [33], [20] can directly estimate the geometry of visible surfaces without requiring explicit camera parameters. For instance, PF3plat [34] aligns 3D Gaussians coarsely by leveraging pre-trained models for monocular depth estimation and visual correspondence. Meanwhile, Splatt3R [35] and NopoSplat [36] employ DUSt3R [5] and MASt3R [33] to predict point maps as proxy geometry, subsequently

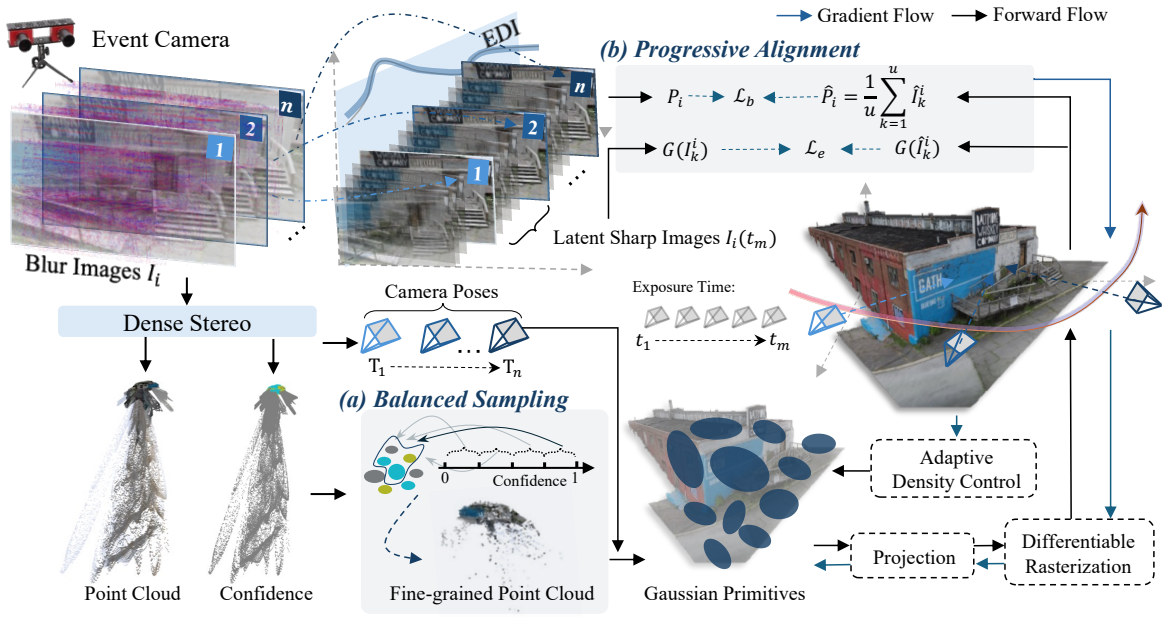


Fig. 3. Illustration of our proposed DeblurSplat. We first input the blur images into the dense stereo module (DUSt3R [5]) for getting estimated point clouds and corresponding confidence. Then, we propose the Confidence Balanced Sampling to extract high-quality point clouds by conforming the original points distribution. In the mean time, we utilize the event stream to decouple blur images into multiple latent sharp images by EDI [16]. Finally, we perform Progressive Alignment to recover the clear semantics in coarse to fine manner.

learning 3D Gaussians for sparse-view reconstruction. Although these methods primarily explore generalization in sparse-view 3D reconstruction, our work is the first to fully exploit dense stereo techniques for deblurring reconstruction and novel-view synthesis, thus extending their applicability to challenging blur scenarios.

III. Method

The proposed DeblurSplat pipeline is illustrated in Figure 3. The following two subsections detail its main components. In Subsection III-A, we generate dense point clouds with associated confidence values via DUSt3R, then analyze various sampling strategies to address potential issues. Inspired by the analysis, we introduce Confidence Balanced Sampling, which employs confidence scores to exclude unreliable points while preserving essential spatial coverage. In Subsection III-B, we utilize event streams to recover both latent sharp images and latent motion poses, providing fine-grained optimization cues for scene reconstruction.

A. Sampling Initial Point Clouds from DUSt3R

Motivation. As outlined in Section I, DUSt3R generates initial point clouds with associated confidence scores by converting multi-view images into a 3D representation. In contrast to COLMAP, which requires camera pose estimation directly from blurred images, DUSt3R can alleviate pose-related inaccuracies, thus producing more reliable point clouds. However, existing SfM-free novel-view synthesis approaches [37], [36], [35] generally operate on sharp images, prompting two crucial questions for blurred-image scenarios: *(1) To what extent can we rely on DUSt3R's point clouds? (2) How can we effectively utilize these point clouds for deblur reconstruction?*

A naive strategy would be to employ all points predicted by DUSt3R. Yet, as illustrated in Figure 2(a, b), the majority of high-confidence points cluster within a limited region where the input images overlap. Such an approach over-represents these regions during Gaussian primitive initialization, increasing computational cost [38], [39]. Furthermore, erroneous points arising from blurry inputs are indiscriminately included, adversely impacting NVS accuracy.

A second, more straightforward strategy involves selecting only high-confidence points. Intuitively, these points appear more trustworthy under blur. However, our empirical analysis (Figure 2(b–d)) challenges this assumption. First, we observe similar confidence histogram distributions for both sharp and blurred images, with notable proportions of points around 0.75 confidence in each, suggesting that confidence alone is not a reliable discriminator of erroneous points. Second, when reconstructing scenes with BAD-GS [3], relaxing the confidence threshold and thereby including more low-confidence points, actually improves the final performance. This indicates that low-confidence points still carry essential spatial cues, and discarding them entirely can lead to a loss of geometric information. Motivated by these observations, we propose a sampling strategy named *Confidence Balanced Sampling* that leverages DUSt3R's confidence estimates while preserving necessary spatial coverage. By carefully balancing the trade-off between high-confidence selection and overall scene structural completeness, we aim to mitigate the detrimental effects of blur-induced errors without forfeiting valuable geometric information.

Confidence Balanced Sampling. Given blurry image sets $P_i \in \mathbb{R}^{H\times W\times 3}$ where $i \in \{1, 2, \ldots, N\}$, we process these images through DUSt3R to generate initial coarse point clouds $\mathcal{P} \in \mathbb{R}^{K\times 3}$ and their associated confidence scores $\mathcal{C} \in \mathbb{R}^{K}$, where K denotes the number of points. The goal of Confidence Balanced Sampling is to sample L points as Gaussian primitives for scene reconstruction. We partition

TABLE I
QUANTITATIVE COMPARISONS OF NOVEL VIEW SYNTHESIS ACROSS SIMPLE, MEDIUM, DIFFICULT LEVEL AND REAL-WORLD SCENES. THE TABLE REPORTS THE AVERAGE PERFORMANCE FOR EACH KIND, DEMONSTRATING THAT OUR METHOD CONSISTENTLY SURPASSES PREVIOUS STATE-OF-THE-ART APPROACHES ACROSS ALL METRICS. BEST RESULTS IN **BOLD**, SECOND-BEST UNDERLINED.

Scene Type	Metric	B-NeRF	B-3DGS	UFP-GS	EFN-GS	E^{2}GS	BAD-NeRF	BAD-GS	EDNeRF	Ours
Hard	PSNR↑	21.46	22.01	23.53	23.44	23.57	24.07	23.34	24.20	**26.94**
	SSIM↑	.6222	.6456	.6979	.6998	.7066	.7253	.7408	.7657	**.8235**
	LPIPS↓	.3701	.3373	.2775	.2825	.2761	.2325	.2120	.1979	**.1355**
Normal	PSNR↑	22.31	22.89	26.88	26.26	28.07	29.35	31.25	31.49	**32.09**
	SSIM↑	.7467	.7623	.8353	.8155	.8821	.9033	.9255	.9389	**.9453**
	LPIPS↓	.3177	.2148	.2084	.2326	.1337	.0998	.0540	.0541	**.0406**
Real	PSNR↑	15.18	16.69	18.98	19.57	20.61	21.47	22.42	22.57	**25.93**
	SSIM↑	.5777	.6200	.6868	.6972	.6863	.6937	.7129	.7049	**.7995**
	LPIPS↓	.3870	.3296	.2289	.2135	.2124	.2084	.1972	.1940	**.0840**

the confidence range $[c_{\min}, c_{\max}]$ into M intervals $\{B_m\}_{m=1}^{M}$, defined as $B_m = [c_{\min} + (m-1)\Delta, c_{\min} + m\Delta)$ with $\Delta = \frac{c_{\max}-c_{\min}}{M}$. For each interval B_m, we extract the subset $\mathcal{S}_m = \{\mathbf{p}_k \in \mathcal{P} \,|\, c_k \in B_m\}$, where c_k is the confidence of point $\mathbf{p}_k$. The sampling process is designed as:

$$\mathcal{P}_{\text{sampled}} = \bigcup_{m=1}^{M} \{\mathbf{p}_k \,|\, \mathbf{p}_k \sim \mathbb{P}(\mathcal{S}_m),\, \mathbb{P}(\mathbf{p}_k) \propto c_k\}, \quad (1)$$

where points are randomly selected within each $\mathcal{S}_m$ with probabilities proportional to their confidence. To ensure balanced coverage across confidence levels, we allocate a fixed number of points $\frac{L}{M}$ per interval B_m. This enforces sparsity in low-confidence regions (prone to noise) and dense sampling in high-confidence regions (critical structures). The resulting subset $\mathcal{P}_{\text{sampled}}$ provides high-quality initialization for Gaussian primitives in deblurring NVS.

One might consider purely spatial sampling (e.g., Farthest Point Sampling, FPS [39], [38]) to maintain geometric coverage. Although spatial sampling can ensure uniform coverage in clear point clouds, in blurred point clouds it risks selecting outlier clusters (e.g., duplicated or drifted points), further distorting the reconstruction. By contrast, our approach treats DUSt3R's confidence map as a relative metric in blurred conditions, striking a balance between high-quality point extraction and overall scene fidelity. We provide a comprehensive evaluation of this design in our ablation study (Table IV).

B. Progressive Alignment

Blur Images Decouple. Event-based deblurring methods [12], [40], [24], [25] have recently garnered significant attention. By leveraging the continuous event stream captured alongside standard RGB frames, these approaches can reconstruct high-fidelity scenes even under challenging conditions such as low-light or motion blur. Motivated by these advances, we introduce event streams into our deblurring pipeline as a high-dynamic-range semantic prior for guidance. Specifically, following the strategy of previous works, we adopt the Event-based Double Integral (EDI) [16] to decouple motion-blurred images using event data. Detailed event-camera imaging principles are omitted here for brevity and can be found in the supplementary material.

In detail, a motion-blurred image P_i is acquired over the exposure interval $[t_0, t_u]$ in tandem with corresponding event bins $\{E_k\}_{k=1}^{u}$. We denote the latent sharp image at time t_1 is I_1. According to the event imaging process, the sharp image I_k at time t_k within each event bin can be expressed as:

$$I_k = I_0 e^{\Theta \sum_{i=1}^{k} E_i}, k > 0, \quad (2)$$

where Θ is the event-camera response threshold. By assuming that the blur in P_i is a time-weighted combination of multiple images [9], and given that the exposure time is uniformly divided into u segments, P_i can be approximated as the average of these latent images:

$$P_i = \frac{I_0}{u+1}(1 + e^{\Theta \sum_{i=1}^{1} E_i} + \cdots + e^{\Theta \sum_{i=1}^{u} E_i}). \quad (3)$$

From this, we can further infer I_0 and I_k at any moment k:

$$\begin{aligned} I_0 &= \frac{(u+1)P_i}{1 + e^{\Theta \sum_{i=1}^{1} E_i} + \cdots + e^{\Theta \sum_{i=1}^{u} E_i}}, \\ I_k &= \frac{(u+1)P_i e^{\Theta \sum_{i=1}^{k} E_i}}{1 + e^{\Theta \sum_{i=1}^{1} E_i} + \cdots + e^{\Theta \sum_{i=1}^{u} E_i}}. \end{aligned} \quad (4)$$

Accordingly, each blurred image P_i can be decomposed into a set of latent sharp images $\{I_k^i\}_{k=1}^{u}$, each containing richer texture information.

Latent Motion Modeling. During the exposure period, a motion-blurred image can be viewed as a sequence of latent images captured along a camera trajectory, which can be discretized by poses $\{\boldsymbol{T}_k^i\}_{k=1}^{u} \in \mathbf{SE}(3)$. We thus parameterize and optimize $\{\boldsymbol{T}_k^i\}_{k=1}^{u}$ in a bundle-adjustment framework [41] to jointly recover the camera trajectory responsible for blur formation. Concretely, we parameterize $\Delta\boldsymbol{T}_1^i$ and $\Delta\boldsymbol{T}_u^i$ and linearly interpolate intermediate poses in the Lie algebra of $\mathbf{SE}(3)$, where $\Delta\boldsymbol{T}^i$ represents the pose adjustment. Hence, at any moment k, the actual pose $\boldsymbol{T}_k^i$ for the corresponding latent image I_k^i is:

$$\boldsymbol{T}_k^i = \hat{\boldsymbol{T}}^i \cdot \Delta\boldsymbol{T}_1^i \cdot \text{Exp}(\frac{k}{u} \cdot \text{Log}((\Delta\boldsymbol{T}_1^i)^{-1}\Delta\boldsymbol{T}_u^i) \quad (5)$$

where $k \in \{1, 2, ..., u\}$ and $\hat{\boldsymbol{T}}^i$ denotes the initial pose for blur images P_i. We optimize $\Delta\boldsymbol{T}_1^i$ and $\Delta\boldsymbol{T}_u^i$ jointly with the learnable parameters of the Gaussian primitives. Note that we also refine the original pose for blurred images, as the DUSt3R-generated poses [38], [37] may be suboptimal.

Event Guided Optimization. In each training iteration, we simultaneously render u latent sharp images $\{\hat{I}_k^i\}_{k=1}^{u}$ from 3D-GS along the camera trajectories $\boldsymbol{T}_k^i$ corresponding to

TABLE II
QUANTITATIVE COMPARISONS OF NOVEL VIEW SYNTHESIS ACROSS DIFFERENT LEVEL BLUR OF REAL-WORLD SCENES. THE TABLE REPORTS THE AVERAGE PERFORMANCE FOR EACH KIND, DEMONSTRATING THAT OUR METHOD CONSISTENTLY SURPASSES PREVIOUS SOTAS.

Level	Metric	B-3DGS	E^{2}GS	BAD-GS	EDNeRF	Ours
1	PSNR↑	18.25	22.88	24.92	24.75	**26.84**
	SSIM↑	.6656	.7502	.7511	.7317	**.8272**
	LPIPS↓	.2728	.1437	.1457	.1727	**.0648**
2	PSNR↑	17.74	21.32	22.22	21.48	**26.27**
	SSIM↑	.6385	.6905	.6965	.6706	**.7858**
	LPIPS↓	.3061	.2124	.2087	.2219	**.1060**
3	PSNR↑	15.71	20.54	21.53	22.45	**25.90**
	SSIM↑	.6025	.6629	.6804	.7080	**.7862**
	LPIPS↓	.3580	.2205	.2097	.1979	**.0875**
4	PSNR↑	15.06	17.71	21.00	21.57	**24.72**
	SSIM↑	.5732	.6415	.6835	.6968	**.7824**
	LPIPS↓	.3815	.2728	.2247	.1994	**.0862**

each blurred view P_i. A direct strategy would be to align these renders $\{\hat{I}_k^i\}_{k=1}^u$ with the actual latent images $\{I_k^i\}_{k=1}^u$. However, despite containing more texture details than P_i, these renders do not fully recover ideal latent frames and often exhibit low visual fidelity (especially in color), thereby degrading optimization outcomes. To mitigate this, we transform the renders into grayscale for alignment, thereby removing color interference while retaining sufficient texture and shape cues:

$$\mathcal{L}_e = \frac{1}{u}\sum_{k=1}^{u} \parallel G(\hat{I}_k^i) - G(I_k^i) \parallel_1, \tag{6}$$

where $G(\cdot)$ denotes the transfer function for gray space. Next, to simulate the formation of motion-blurred images, we approximate Eq. 3 by averaging the u latent renders: $\hat{P}^i = \frac{1}{u}\sum_{k=1}^{u} \hat{I}_k^i$ and adopt the same reconstruction loss as in the original 3D-GS [1]:

$$\mathcal{L}_b = (1-\lambda_b)\cdot \parallel \hat{P}^i - P^i \parallel_1 + \lambda_b \cdot \mathcal{L}_{\text{D-SSIM}}(\hat{P}^i, P^i), \tag{7}$$

Our total loss combines the above objectives:

$$\mathcal{L} = \mathcal{L}_b + \lambda_e \mathcal{L}_e. \tag{8}$$

Equation 6 promotes convergence toward the correct pose by incorporating the EDI-based event prior. Moreover, it helps the model avoid suboptimal solutions stemming from homogeneous pose updates under blurred-image gradients (Eq. 7). By jointly enforcing geometric, textural, and color fidelity across multiple views, this overall loss formulation pushes the model's deblurring performance to new heights. More discussion can be referred to our supplementary.

IV. EXPERIMENTS

A. Implementation Details

Our model is implemented based on the official code of 3D-GS [1]. Both the optimization of Gaussians and camera poses are performed using the Adam optimizer. The learning rate for Gaussians remains identical to the original 3D-GS. The training process spans 20,000 iterations, with an event reconstruction loss introduced after a 3,000-iteration warm up. We set $m = 40$ and $Q_m = 125$, which denotes we assume the exposure time of each blur image can be divided into 40 internals and sample a total of 5k points. Then, we set $\lambda_b = 0.2$ and $\lambda_e = 5 \times 10^{-3}$ for balancing the prior guidance. We set $u = 10$ and $\Theta = 0.27$ following manner of previous works [8], [3], [41]. All experiments were conducted using a single NVIDIA RTX A6000 GPU.

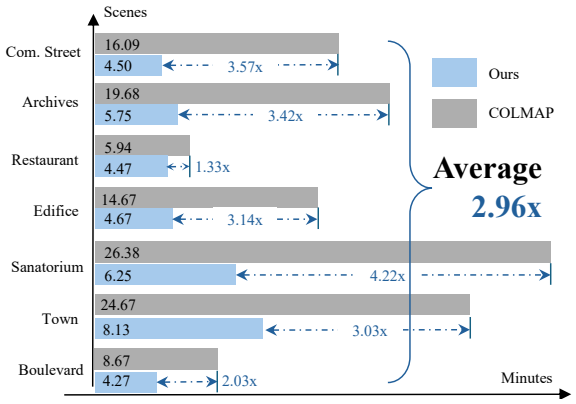


Fig. 4. Quantitative comparisons of time consuming (min.) of initialization between COLMAP and DeblurSplat applied in hard scenes. Note that the COLMAP initialization is based on sharp images (almost fail on blur images).

TABLE III
ANALYSIS OF DEBLURSPLAT. "BALA." DENOTES THE CONFIDENCE BALANCED SAMPLING. "PROG." DENOTES PROGRESSIVE ALIGNMENT.

Bala.	Prog.	Hard			Normal		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
-	-	22.29	.6985	.3037	24.98	.7545	.3123
-	✓	23.82	.7621	.1908	29.85	.9273	.0448
✓	-	23.32	.7337	.2547	26.19	.8470	.1130
✓	✓	**26.94**	**.8235**	**.1355**	**32.09**	**.9453**	**.0406**

B. Datasets

Referring to [6], [28], we evaluate our method on both synthetic and real-world datasets. To comprehensively assess model performance, we introduce three types of benchmarks. In addition to normal dataset [6], we consider more complicated scenes and propose the hard scenes crafted from Blender and real-world dataset. No limited in complex environment condition, we also design the large-scale movement of camera motion, poses a great challenge to general deblurring reconstruction models. Notably, for baseline evaluations in hard scenes, we initialize COLMAP using sharp images, as COLMAP consistently fails under blurry conditions. As shown in Figure 4, even when processing sharp images, COLMAP remains computationally expensive whereas our method achieves almost 2.96× faster performance. *More dataset details are provided in the supplementary material.*

C. Experiment Settings

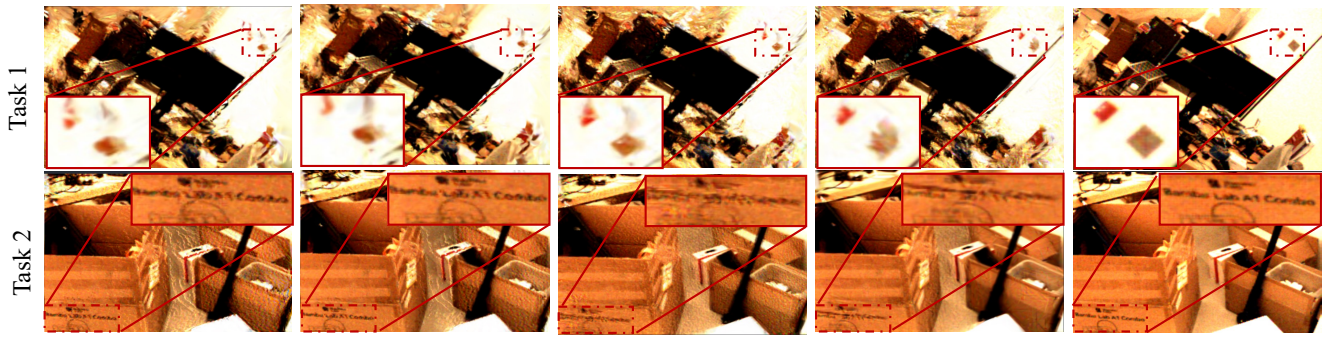
Baselines. We compare our method with three types of baselines: 1) NeRF [2] and 3D-GS [1] directly trained on the blurry images, referring to as B-NeRF and B-3DGS. 2) Deblur rendering methods, including BAD-NeRF [8], BAD-GS [3], E^2 GS [10] and EDNeRF [29]. 3) Image deblur methods, including UFP [42] (single-image deblurring) and EFNet [43] (learnable event-based deblurring). We process



(a) Overview of Rendering Novel Views in Hard Scenes



(b) Overview of Rendering Novel Views in Normal Scenes



(c) Overview of Rendering Novel Views in Real-world Scenes

Fig. 5. Qualitative comparison on the synthetic and real dataset. It shows that our method achieves better performance in rendering novel views compared to other SOTAs in deblur reconstruction. More results about NVS and deblurring are presented in the supplementary.

input blurry images with them and train the vanilla 3D-GS with pre-deblurred images.

Evaluation Metrics. We employ the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [44], and VGG-based Learned Perceptual Image Patch Similarity (LPIPS) [45] to evaluate the similarity between rendered views and ground truth views.

D. Performance

Synthetic Data Experiments. We evaluate our approach across a diverse set of synthetic scenes, including both normal and challenging (hard) scenarios. Quantitative results for novel view synthesis are presented in Table I, while detailed deblurring results of the input views can be found in the supplementary material. Our method demonstrates significant improvements across most metrics, particularly in challenging scenes, achieving a +2.74 dB increase in PSNR, a +5.78% boost in SSIM, and a +6.24% improvement in LPIPS. Specifically, both B-NeRF and B-3DGS generate blurry novel views as they are directly trained on blurred images. Compared to the BAD-Series methods [8], [3], our results highlight the advantages of leveraging event priors and dense stereo guidance.

TABLE IV
ANALYSIS OF DIFFERENT SAMPLING METHODS IN INITIALIZATION.

Method	Hard			Normal		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Center	26.20	.8078	.1746	30.25	.9296	.0770
Random	26.63	.8175	.1386	31.84	.9451	.0413
Spatial	26.57	.8166	.1406	32.08	**.9459**	**.0405**
Balanced	**26.94**	**.8235**	**.1355**	**32.09**	.9453	.0406

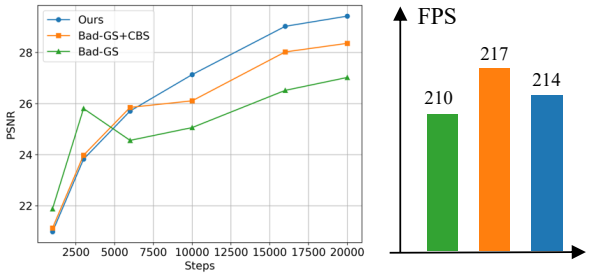


Fig. 6. Analysis of efficiency of Confidence Balanced Sampling (CBS) in Hard Scenes. Here, we utilize the PSNR in deblur view rendering as the evaluation metric.

Additionally, while EDNeRF [30] attempts to decouple each blurry image into a single sharp prior, it fails to capture the continuous features and fine-grained variations in the event stream, ultimately leading to inferior performance.

Real Data Experiments. In this section, we analyze our model's performance independently from the overall evaluation, focusing on different degrees of blur. As shown in Table I, our method outperforms state-of-the-art approaches, achieving a notable improvement of +3.36 dB in PSNR, +8.66% in SSIM, and +6.24% in LPIPS, demonstrating its practical applicability in real-world scenarios. We further investigate the impact of varying blur levels, as detailed in Table II. Our findings indicate that as the degree of blurring increases, the performance of all deblurring methods declines to varying extents. However, our approach consistently maintains superior performance, achieving a +2.15 dB gain in PSNR, highlighting its effective utilization of knowledge from DUSt3R and event streams. Additional results can be found in the supplementary.

Qualitative Results. Qualitative results are presented in Figure 5, which demonstrates that our method effectively generates high-quality scene representations with intricate details, even when trained on a series of blurred images. EDNeRF [30] shows overall strong performance, but it still exhibits some deficiencies, particularly in areas with significant color and light variation. This highlights the advantage of using explicit Gaussian splatting representation over implicit neural representations. In Figure 5, it is clear that 3DGS-based methods [3], [10] struggle to model regions with significant depth variations, mainly due to their heavy reliance on simulating blurry imaging processes or using latent sharp images. When applied to more complex scenes, such as those in Figure 5(a,c), these methods experience performance collapse.

E. Ablation Study

In this section, we perform detailed study to investigate our designed modules in DeblurSplat by utilizing the NVS task. More ablation study can be referred to supplementry.

Overview. As shown in Table I, we first analyze the impact of each designed module. For our baseline, we adopt the standard 3DGS performance using only COLMAP initialization (B-3DGS). When incorporating only the "Bala" module, performance improves significantly over COLMAP initialization (+1.03 dB PSNR in hard scenes), benefiting from the fine-grained point clouds generated by a well-pretrained model. However, this approach still struggles with deblurring reconstruction, as it learns solely from blurred images without additional refinements. On the other hand, utilizing only the "Prog." module yields high-fidelity deblurred reconstructions, leveraging the rich semantic information from the event stream. Finally, by integrating both techniques, we achieve the best overall performance for NVS.

Sampling Techniques. We investigate the impact of different sampling methods for generating point clouds, as shown in Table IV. The "Spatial" method employs Farthest Point Sampling (FPS) within adaptive voxel grids [38]. The "Random" method directly samples points from the point cloud without any constraints. The "Center" method selects a high-confidence point cloud center and randomly samples points within its average geometric distance. Our findings indicate that the "Center" method leads to performance degradation, as it filters out distant points that play a crucial role in enhancing generalization ability. Meanwhile, the "Random" method, despite the possibility of sampling less relevant points, maintains high NVS quality. This supports our hypothesis in Section III-B—that even lower-confidence points can carry meaningful semantics and contribute to performance. Although the "Spatial" method performs well in simple scenes, it struggles in more challenging scenarios. By relying on geometric distance, it increases the likelihood of sampling less informative points, ultimately limiting its robustness in complex environments.

Initialization Efficiency. We first evaluate the efficiency of the Confidence Balanced Sampling (CBS) module. As shown in Figure 6, applying CBS to Bad-GS [3] leads to both higher performance and faster convergence. Notably, with only 12,500 training steps, our method achieves performance comparable to Bad-GS trained for 20,000 steps. Moreover, all configurations maintain a rendering speed of over 210 FPS, demonstrating the efficient knowledge utilization of the dense stereo module (DUSt3R).

V. CONCLUSION

In this paper, we introduce DeblurSplat, the first SfM-free deblurring framework for 3D-GS that integrates event-camera priors with DUSt3R's robust initialization. By leveraging Confidence Balanced Sampling, we ensure reliable point cloud initialization, while Progressive Alignment effectively refines motion-blurry pixels using event-stream semantics. Our method outperforms previous motion-deblur reconstruction approaches across diverse synthetic and real-world datasets, demonstrating superior robustness under varying blur levels. These results highlight the effectiveness of SfM-free deblurring and event-based priors in enhancing 3D scene reconstruction.

REFERENCES

[1] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM TOG*, vol. 42, no. 4, pp. 139–1, 2023.

[2] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[3] L. Zhao, P. Wang, and P. Liu, "Bad-gaussians: Bundle adjusted deblur gaussian splatting," in *ECCV*. Springer, 2025, pp. 233–250.

[4] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016.

[5] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "Dust3r: Geometric 3d vision made easy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 697–20 709.

[6] L. Ma, X. Li, J. Liao, Q. Zhang, X. Wang, J. Wang, and P. V. Sander, "Deblur-nerf: Neural radiance fields from blurry images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 861–12 870.

[7] D. Lee, M. Lee, C. Shin, and S. Lee, "Dp-nerf: Deblurred neural radiance field with physical scene priors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 386–12 396.

[8] P. Wang, L. Zhao, R. Ma, and P. Liu, "Bad-nerf: Bundle adjusted deblur neural radiance fields," in *CVPR*, 2023, pp. 4170–4179.

[9] G. Gallego, T. Delbrück, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis *et al.*, "Event-based vision: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 1, pp. 154–180, 2020.

[10] H. Deguchi, M. Masuda, T. Nakabayashi, and H. Saito, "E2gs: Event enhanced gaussian splatting," in *ICIP*. IEEE, 2024, pp. 1676–1682.

[11] Y. Qi, J. Li, Y. Zhao, Y. Zhang, and L. Zhu, "E3nerf: Efficient event-enhanced neural radiance fields from blurry images," *arXiv:2408.01840*, 2024.

[12] W. Yu, C. Feng, J. Tang, X. Jia, L. Yuan, and Y. Tian, "Evagaussians: Event stream assisted gaussian splatting from blurry images," *arXiv:2405.20224*, 2024.

[13] T. Yura, A. Mirzaei, and I. Gilitschenski, "Eventsplat: 3d gaussian splatting from moving event cameras for real-time rendering," *arXiv:2412.07293*, 2024.

[14] G. R. Matta, R. Trisha, and K. Mitra, "Besplat–gaussian splatting from a single blurry image and event stream," *arXiv:2412.19370*, 2024.

[15] Z. Zhang, K. Chen, and L. Wang, "Elite-evgs: Learning event-based 3d gaussian splatting by distilling event-to-video priors," *arXiv:2409.13392*, 2024.

[16] L. Pan, C. Scheerlinck, X. Yu, R. Hartley, M. Liu, and Y. Dai, "Bringing a blurry frame alive at high frame-rate with an event camera," in *CVPR*, 2019, pp. 6820–6829.

[17] S. Elflein, Q. Zhou, S. Agostinho, and L. Leal-Taixé, "Light3r-sfm: Towards feed-forward structure-from-motion," *arXiv preprint arXiv:2501.14914*, 2025.

[18] J. Lu, T. Huang, P. Li, Z. Dou, C. Lin, Z. Cui, Z. Dong, S.-K. Yeung, W. Wang, and Y. Liu, "Align3r: Aligned monocular depth estimation for dynamic videos," *arXiv preprint arXiv:2412.03079*, 2024.

[19] J. Xu, S. Gao, and Y. Shan, "Freesplatter: Pose-free gaussian splatting for sparse-view 3d reconstruction," *arXiv preprint arXiv:2412.09573*, 2024.

[20] S. Zhang, J. Wang, Y. Xu, N. Xue, C. Rupprecht, X. Zhou, Y. Shen, and G. Wetzstein, "Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views," *arXiv preprint arXiv:2502.12138*, 2025.

[21] V. Rudnev, M. Elgharib, C. Theobalt, and V. Golyanik, "Eventnerf: Neural radiance fields from a single colour event camera," in *CVPR*, 2023, pp. 4992–5002.

[22] S. Klenk, L. Koestler, D. Scaramuzza, and D. Cremers, "E-nerf: Neural radiance fields from a moving event camera," *IEEE RAL*, vol. 8, no. 3, pp. 1587–1594, 2023.

[23] T. Xiong, J. Wu, B. He, C. Fermuller, Y. Aloimonos, H. Huang, and C. Metzler, "Event3dgs: Event-based 3d gaussian splatting for high-speed robot egomotion," in *8th Annual Conference on Robot Learning*, 2024.

[24] H. Han, J. Li, H. Wei, and X. Ji, "Event-3dgs: Event-based 3d reconstruction using 3d gaussian splatting," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2023.

[25] W. F. Low and G. H. Lee, "Robust e-nerf: Nerf from sparse & noisy events under non-uniform motion," in *ICCV*, 2023, pp. 18 335–18 346.

[26] ——, "Deblur e-nerf: Nerf from motion-blurred events under high-speed or low-light conditions," in *ECCV*. Springer, 2025, pp. 192–209.

[27] J. Wu, S. Zhu, C. Wang, B. Shi, and E. Y. Lam, "Sweepevgs: Event-based 3d gaussian splatting for macro and micro radiance field rendering from a single sweep," *arXiv:2412.11579*, 2024.

[28] Y. Qi, L. Zhu, Y. Zhang, and J. Li, "E2nerf: Event enhanced neural radiance fields from blurry images," in *ICCV*, 2023, pp. 13 254–13 264.

[29] J. Wu, S. Zhu, C. Wang, and E. Y. Lam, "Ev-gs: Event-based gaussian splatting for efficient and accurate radiance field rendering," in *MLSP*. IEEE, 2024, pp. 1–6.

[30] M. Cannici and D. Scaramuzza, "Mitigating motion blur in neural radiance fields with events and frames," in *CVPR*, 2024, pp. 9286–9296.

[31] Y. Chen, C. Yang, J. Fang, X. Zhang, L. Xie, W. Shen, W. Dai, H. Xiong, and Q. Tian, "Liftimage3d: Lifting any single image to 3d gaussians with video generation priors," *arXiv preprint arXiv:2412.09597*, 2024.

[32] J. Yang, A. Sax, K. J. Liang, M. Henaff, H. Tang, A. Cao, J. Chai, F. Meier, and M. Feiszli, "Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass," *arXiv preprint arXiv:2501.13928*, 2025.

[33] V. Leroy, Y. Cabon, and J. Revaud, "Grounding image matching in 3d with mast3r," in *European Conference on Computer Vision*. Springer, 2024, pp. 71–91.

[34] S. Hong, J. Jung, H. Shin, J. Han, J. Yang, C. Luo, and S. Kim, "Pf3plat: Pose-free feed-forward 3d gaussian splatting," *arXiv preprint arXiv:2410.22128*, 2024.

[35] B. Smart, C. Zheng, I. Laina, and V. A. Prisacariu, "Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs," *arXiv preprint arXiv:2408.13912*, 2024.

[36] B. Ye, S. Liu, H. Xu, X. Li, M. Pollefeys, M.-H. Yang, and S. Peng, "No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images," *arXiv preprint arXiv:2410.24207*, 2024.

[37] Y. Tang, Y. Guo, D. Li, and C. Peng, "Spars3r: Semantic prior alignment and regularization for sparse 3d reconstruction," *arXiv preprint arXiv:2411.12592*, 2024.

[38] Z. Fan, W. Cong, K. Wen, K. Wang, J. Zhang, X. Ding, D. Xu, B. Ivanovic, M. Pavone, G. Pavlakos *et al.*, "Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds," *arXiv preprint arXiv:2403.20309*, vol. 2, no. 3, p. 4, 2024.

[39] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[40] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "High speed and high dynamic range video with an event camera," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 6, pp. 1964–1980, 2019.

[41] J. Huang, C. Dong, and P. Liu, "Inceventgs: Pose-free gaussian splatting from a single event camera," *arXiv:2410.08107*, 2024.

[42] Z. Fang, F. Wu, W. Dong, X. Li, J. Wu, and G. Shi, "Self-supervised non-uniform kernel estimation with flow-based motion prior for blind image deblurring," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 18 105–18 114.

[43] L. Sun, C. Sakaridis, J. Liang, Q. Jiang, K. Yang, P. Sun, Y. Ye, K. Wang, and L. V. Gool, "Event-based fusion for motion deblurring with cross-modal attention," in *European conference on computer vision*. Springer, 2022, pp. 412–428.

[44] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[45] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[46] F. Plastria, "The weiszfeld algorithm: proof, amendments, and extensions," *Foundations of location analysis*, pp. 357–389, 2011.

[47] M. FISCHLER AND, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[48] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Ep n p: An accurate o (n) solution to the p n p problem," *International journal of computer vision*, vol. 81, pp. 155–166, 2009.

[49] D. Gehrig, M. Gehrig, J. Hidalgo-Carrió, and D. Scaramuzza, "Video to events: Recycling video datasets for event cameras," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2020.

[50] D. Gehrig and D. Scaramuzza, "Video to events: Recycling video datasets for event cameras," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, June 2020.

APPENDIX A
PRELIMINARIES

A. Event generation model

Unlike frame-based cameras that record the brightness of each pixel at a fixed frame rate [9], the event camera asynchronously generates an event $\boldsymbol{e}(x, y, \tau, p)$ when the changes of the brightness of pixel (x, y) reach threshold Θ in the log domain at time τ as follows:

$$p_{x,y,\tau} = \begin{cases} -1, & \log(\mathcal{I}_{x,y,\tau}) - \log(\mathcal{I}_{x,y,\tau-\Delta\tau}) < -\Theta, \\ +1, & \log(\mathcal{I}_{x,y,\tau}) - \log(\mathcal{I}_{x,y,\tau-\Delta\tau}) > \Theta, \end{cases} \tag{9}$$

$$E_k = \{\boldsymbol{e}_i(x_i, y_i, \tau_i, p_i)\}_{t_{k-1}<\tau_i<t_k}, \tag{10}$$

where p indicates the direction of brightness change, $\mathcal{I}_{(x,y,\tau)}$ is the brightness value of pixel (x, y) at time τ. Since the event camera does not have the concept of frame, to facilitate processing and representation, we usually divide the events into m event bins by time. In this work, given a blurred image with exposure time from t_{start} to $t_\{end\}$ and the corresponding event data $\{\boldsymbol{e}_i\}_{t_{start}<\tau_i\leq t_{end}}$, we can generate $\{E_k\}_{k=1}^{u}$, where $t_k = t_{start} + \frac{k}{b} t_{exp}$ is the time division point between bins and $t_{exp} = t_{start} - t_{end}$ is exposure time.

B. DUSt3R Details

Training Details. The training objective of DUSt3R is based on regression of the unprojected and normalized point maps of two input views. Here, the ground-truth pointmaps from the datasets are $\hat{\boldsymbol{P}}_{1,1}$ and $\hat{\boldsymbol{P}}_{2,1}$ obtained from two corresponding views 1,2 where the camera origin as view 1, on which the ground-truth is defined. The regression loss for training DUSt3R is as:

$$\mathcal{L}_{reg} = \| \frac{1}{z_i} \cdot \boldsymbol{P}_{v,1} - \frac{1}{\hat{z}_i} \cdot \hat{\boldsymbol{P}}_{v,1} \|, \tag{11}$$

where view $v \in \{1, 2\}$, $\boldsymbol{P}$ and $\hat{\boldsymbol{P}}$ are the prediction and ground-truth, separately. To handle the scale ambiguity between prediction and ground-truth, DUSt3R normalizes the predicted and ground-truth pointmaps by scaling factors $z_i = \text{norm}(\boldsymbol{P}_{1,1}, \boldsymbol{P}_{2,1})$ and $\hat{z}_i = \text{norm}(\hat{\boldsymbol{P}}_{1,1}, \hat{\boldsymbol{P}}_{2,1})$ respectively, which simply represent the average distance, denoted as $\boldsymbol{D}$ of all valid points to the origin. Then, the pixel-wise confidence $\boldsymbol{C}^i_{v,1}$ is defined and optimized as:

$$\mathcal{L}_{conf} = \sum_{v\in\{1,2\}} \sum_{i\in\boldsymbol{D}^v} \boldsymbol{C}^i_{v,1} \cdot \mathcal{L}_{reg}(v, i) - \alpha \cdot \log \boldsymbol{C}^i_{v,1}, \tag{12}$$

where α is a hyper-parameter controlling the regularization term, encouraging the model to extrapolate in harder areas.

Recovering Camera Parameters. We assume the camera intrinsics are the same for all the views in the trajectory and obtain 1:1 mapping from the pixel lattice to pointmap where we can build the mapping from 2D to the camera coordinate system. We can first solve the simple optimization problem to calculate per-camera focal:

$$f^* = \arg\min_f \sum_{i=0}^{W} \sum_{j=0}^{H} \boldsymbol{O}^{i,j} \left\| (i', j') - f \frac{(\boldsymbol{P}^{i,j,0}, \boldsymbol{P}^{i,j,1})}{\boldsymbol{P}^{i,j,2}} \right\|, \tag{13}$$

where $i' = i - \frac{W}{2}$ and $j' = j - \frac{H}{2}$ denotes the normalized 2D image coordinate, $\boldsymbol{P}^{i,j,k}, k \in \{0, 1, 2\}$ denotes the predictive pointmaps, W and H denotes the shape of point maps. Following DUSt3R [5], we use the Weiszfeld algorithm [46] to estimate f^*. Assuming a single-camera setup similar to that used in COLMAP for a single scene capture, we propose stabilizing the estimated focal length by averaging across all training views: $\overline{f} = \frac{1}{N}\sum_{i=1}^{N} f_i^*$, which is utilized in subsequent processes. Relative transformation $\boldsymbol{T} = [\boldsymbol{R}|\boldsymbol{t}]$ can be computed by RANSAC [47] with PnP [48] for each image pair.

Pair-wise to Globally Aligned Poses. DUSt3R originally takes image pairs as inputs where post-processing is required to align the scale if more than two views are captured from the scene. The underlying reason is that the predicted point map are at the normalized scales within its own, and such misalignment of each independent calculated relative poses leads to significant scale variance, resulting in inaccurate camera poses. Similar to DUSt3R we first construct a complete connectivity graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ of all the N input views, where the vertices $\mathcal{V}$ and each edge $e = (n, m) \in \mathcal{E}$ indicate a shared visual content between images I_n and I_m. To convert the initially predicted point map, $\{(\boldsymbol{P}_i \in \mathbb{R}^{H\times W\times 3})\}_{i=1}^{N}$ to be a globally aligned one $\{(\tilde{\boldsymbol{P}}_i \in \mathbb{R}^{H\times W\times 3})\}_{i=1}^{N}$, we update the point maps, transformation matrix, and a scale factor: For the complete graph, any image pair $e = (n, m) \in \mathcal{E}$, the point maps $\boldsymbol{P}_{n,n}, \boldsymbol{P}_{m,n}$ and confidence maps $\boldsymbol{O}_{n,n}, \boldsymbol{O}_{m,n}$. For clarity, let us define $\boldsymbol{P}_{n,e} := \boldsymbol{P}_{n,n}$, $\boldsymbol{P}_{m,e} := \boldsymbol{P}_{m,n}$, $\boldsymbol{O}_{n,e} := \boldsymbol{O}_{n,n}$, and $\boldsymbol{O}_{m,e} := \boldsymbol{O}_{m,n}$. The optimization for the transformation matrix of edge $\boldsymbol{T}_e$, scaling factor σ_e and point map $\tilde{\boldsymbol{P}}$ are given by:

$$\tilde{\boldsymbol{P}}^* = \arg\min_{\tilde{\boldsymbol{P}},\boldsymbol{T},\sigma} \sum_{e\in\mathcal{E}} \sum_{v\in e} \sum_{i=1}^{HW} \boldsymbol{O}^i_{v,e} \left\| \tilde{\boldsymbol{P}}^i_v - \sigma_e \boldsymbol{T}_e \boldsymbol{P}^i_{v,e} \right\|. \tag{14}$$

Here, we slightly abuse notation and use $v \in e$ for $v \in \{n, m\}$ if $e = \{n, m\}$. The idea is that, for a given pair e, the same rigid transformation $\boldsymbol{T}_e$ should align both pointmaps $\boldsymbol{P}_{n,e}$ and $\boldsymbol{P}_{m,e}$ are by definition both expressed in the same coordinate frame. To avoid the trivial optimum where $\sigma_e = 0, \forall e \in \mathcal{E}$, DUSt3R enforces $\prod_e \sigma_e = 1$. This post-processing step yields a globally aligned point cloud within only seconds.

C. 3D Gaussian Splatting

Following 3D-GS [1], the scene is represented by a series of 3D Gaussians. Each Gaussian, denoted as $\mathbf{G}$, is parameterized by its mean position $\boldsymbol{\mu} \in \mathbb{R}^3$, 3D covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{3\times3}$, opacity $\mathbf{o} \in \mathbb{R}$ and color $\mathbf{c} \in \mathbb{R}^3$. The distribution of each scaled Gaussian is defined as:

$$\mathbf{G}(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}. \tag{15}$$

To ensure that the 3D covariance $\boldsymbol{\Sigma}$ remains positive semi-definite, which is physically meaningful, and to reduce the optimization difficulty, 3D-GS represents $\boldsymbol{\Sigma}$ using a scale $\mathbf{S} \in \mathbb{R}^3$ and rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$ stored by a quaternion $\mathbf{q} \in \mathbb{R}^4$:

$$\boldsymbol{\Sigma} = \mathbf{RSS}^T\mathbf{R}^T. \tag{16}$$

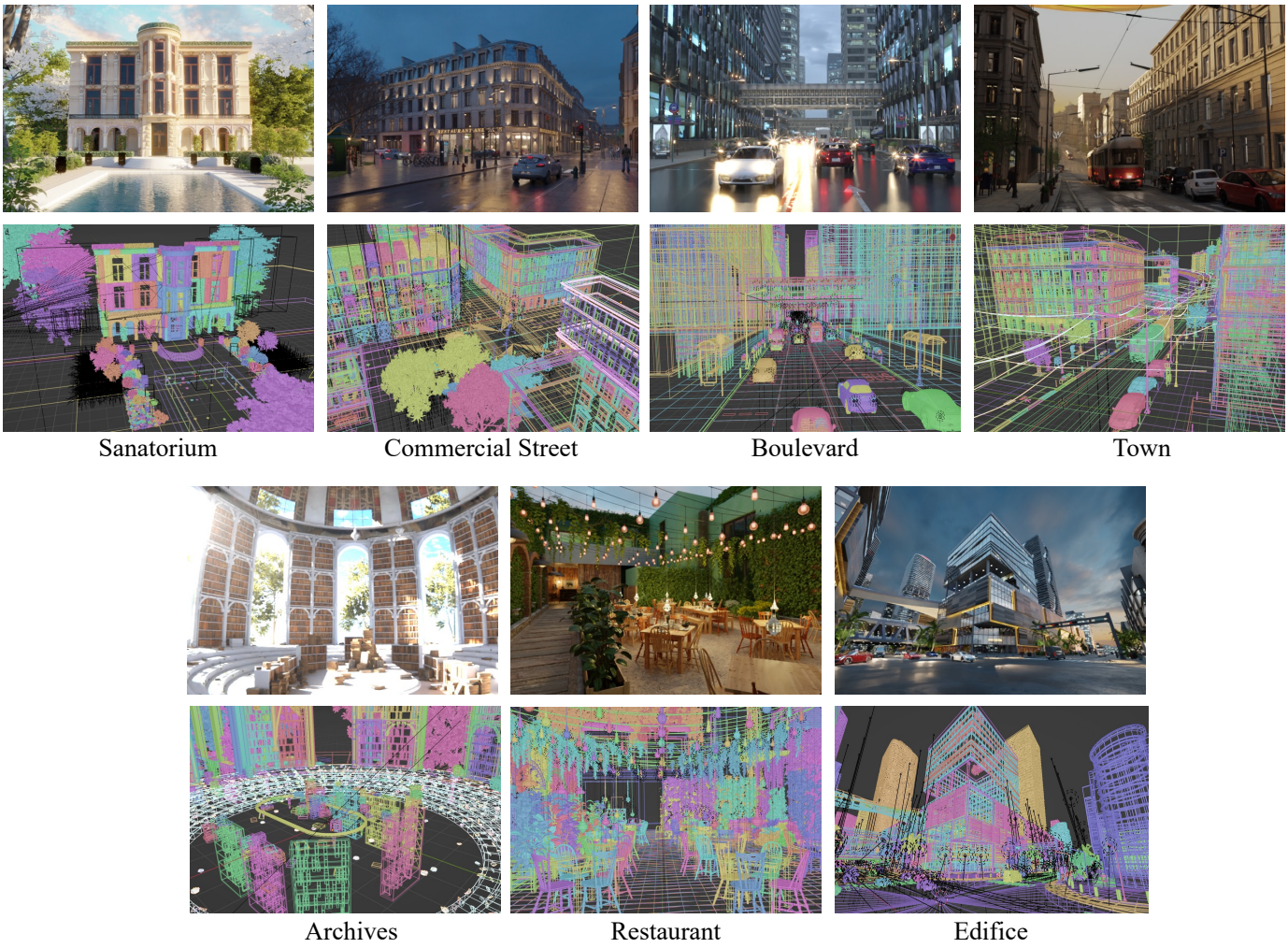


Fig. 7. Illustration of our crafted hard scene datasets. For each scene, the first line is the rendered mode and the second line is the wire frame mode with random colors. Each scene has diverse and multiple objects and complex lightning involving daylight and night, which is more complicated than synthesis dataset from BAD-NeRF [8].

In order to enable differentiable Gaussian rasterization, 3D-GS projects 3D Gaussians to 2D from a given camera pose $\mathbf{T}_c = \{\mathbf{R}_c \in \mathbb{R}^{3\times3}, \mathbf{t}_c \in \mathbb{R}^3\}$, for rasterizing and rendering using the following equation,

$$\mathbf{\Sigma}' = \mathbf{J}\mathbf{R}_c\mathbf{\Sigma}\mathbf{R}_c^T\mathbf{J}^T, \tag{17}$$

where $\mathbf{\Sigma}' \in \mathbb{R}^{2\times2}$ is the 2D covariance matrix, $\mathbf{J} \in \mathbb{R}^{2\times3}$ is the Jacobian of the affine approximation of the projective transformation. Afterward, each pixel color is rendered by rasterizing these N sorted 2D Gaussians based on their depths, following the formulation:

$$\mathbf{C} = \sum_i^N \mathbf{c}_i\alpha_i \prod_j^{i-1}(1-\alpha_j), \tag{18}$$

where $\mathbf{C}_i$ is the learnable color of each Gaussian, and α_i is the alpha value computed by evaluating a 2D covariance $\mathbf{\Sigma}'$ multiplied with the learned Gaussian opacity $\mathbf{o}$:

$$\alpha_i = \mathbf{o}_i \cdot \exp(-\sigma_i), \quad \sigma_i = \frac{1}{2}\Delta_i^T\mathbf{\Sigma}'^{-1}\Delta_i, \tag{19}$$

where $\Delta \in \mathbb{R}^2$ is the offset between the pixel center and the 2D Gaussian center. The derivations presented above demonstrate that the rendered pixel color, denoted as C in Eq. 18, is a function that is differentiable with respect to all of the learnable Gaussians $\mathbf{G}$ and the camera poses $\mathbf{T}_c$.

Appendix B
Discussion of Progressive Alignment

The main idea is to use the latent images decoupled from the event stream for simulated pose supervision through EDI. Regarding the utilization of events as a deblur prior, previous works [10], [28], [12] require inputting latent sharp images to obtain the corresponding camera poses, which is time-consuming. Moreover, sparse point clouds initialized from $I_{k\,k=1}^{i\,m}$ often contain inaccurate semantic points due to biases introduced during the EDI decoupling process [12]. Ev-DeblurNeRF [30] uses only the middle latent sharp image as a prior, failing to take advantage of the continuous response generated by the high dynamic characteristics of

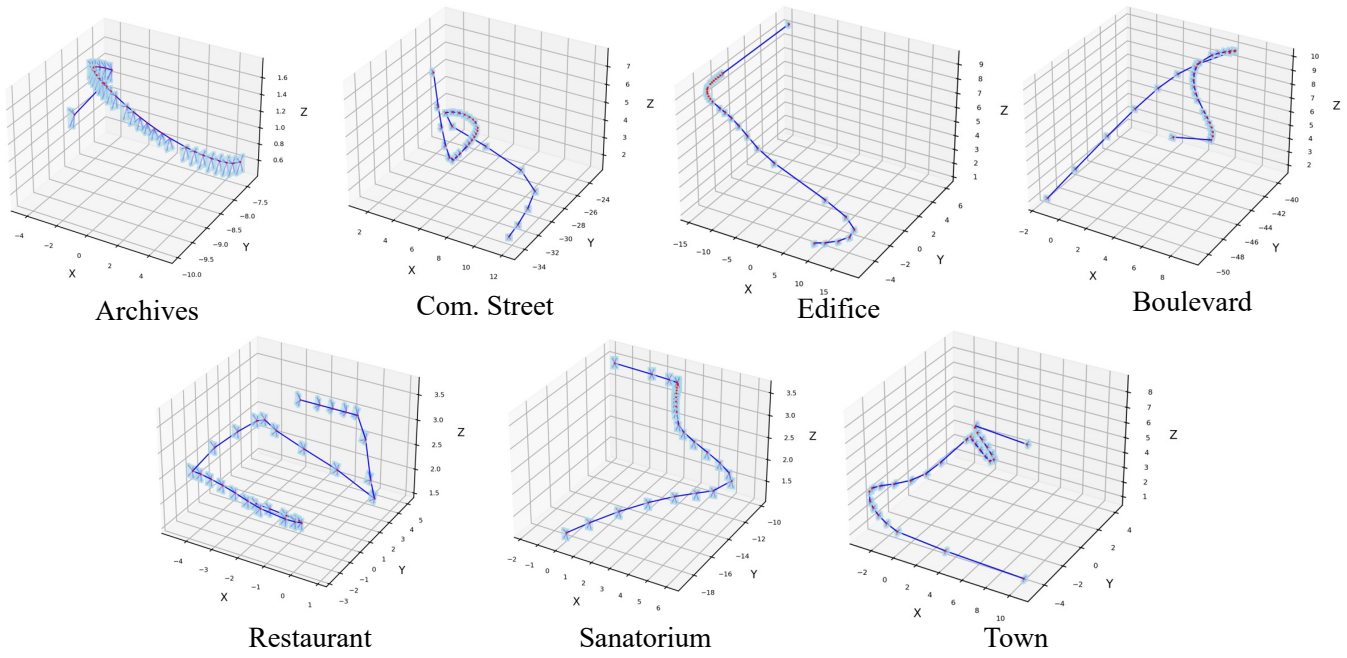


Fig. 8. Illustration of our crafted camera trajectory for hard scenes. Compared to normal scene datasets [6]. The average movement distance of the camera trajectory we proposed is over 10 meters, and the focus view is constantly switched every frame, which poses a great challenge to general deblurring reconstruction models.

events. This results in the loss of important scene details within the exposure time. In contrast, our approach utilizes the decoupled latent sharp images to correct inaccurate camera poses initialized from DUSt3R. The priors from DUSt3R and the event camera are efficiently fused, complementing each other within this deblur framework.

APPENDIX C
DATASET

A. Synthetic Dataset Details

We construct a synthetic dataset that spans a diverse range of scene scales, incorporating various camera trajectories and event data. To systematically evaluate model performance, we categorize the Blender-synthesized dataset into two difficulty levels based on object count, scene complexity, and other critical factors. For the normal level, we directly adopt the dataset proposed in Deblur-NeRF [6], which comprises five scenes: Factory, Wine, Tanabata, Pool, and Cozyroom. For the hard level, we create seven additional indoor and outdoor scenes using Blender and further integrate five 3D scenes from BAD-NeRF [8]. To simulate motion-blurred images, we design diverse camera trajectories across these scenes and generate corresponding event streams using ESIM [49]. Motion blur is introduced by manually placing multi-view cameras, randomly perturbing camera poses, and interpolating linearly between the original and adjusted positions for each view. The final blurred images are obtained by rendering images from these interpolated poses and blending them in the RGB space. *These seven crafted scenes serve as challenging benchmarks due to their high object density, complex lighting conditions, rich texture details, and large-scale camera movements. As*



Fig. 9. Illustration of our equipment for creating real datasets.

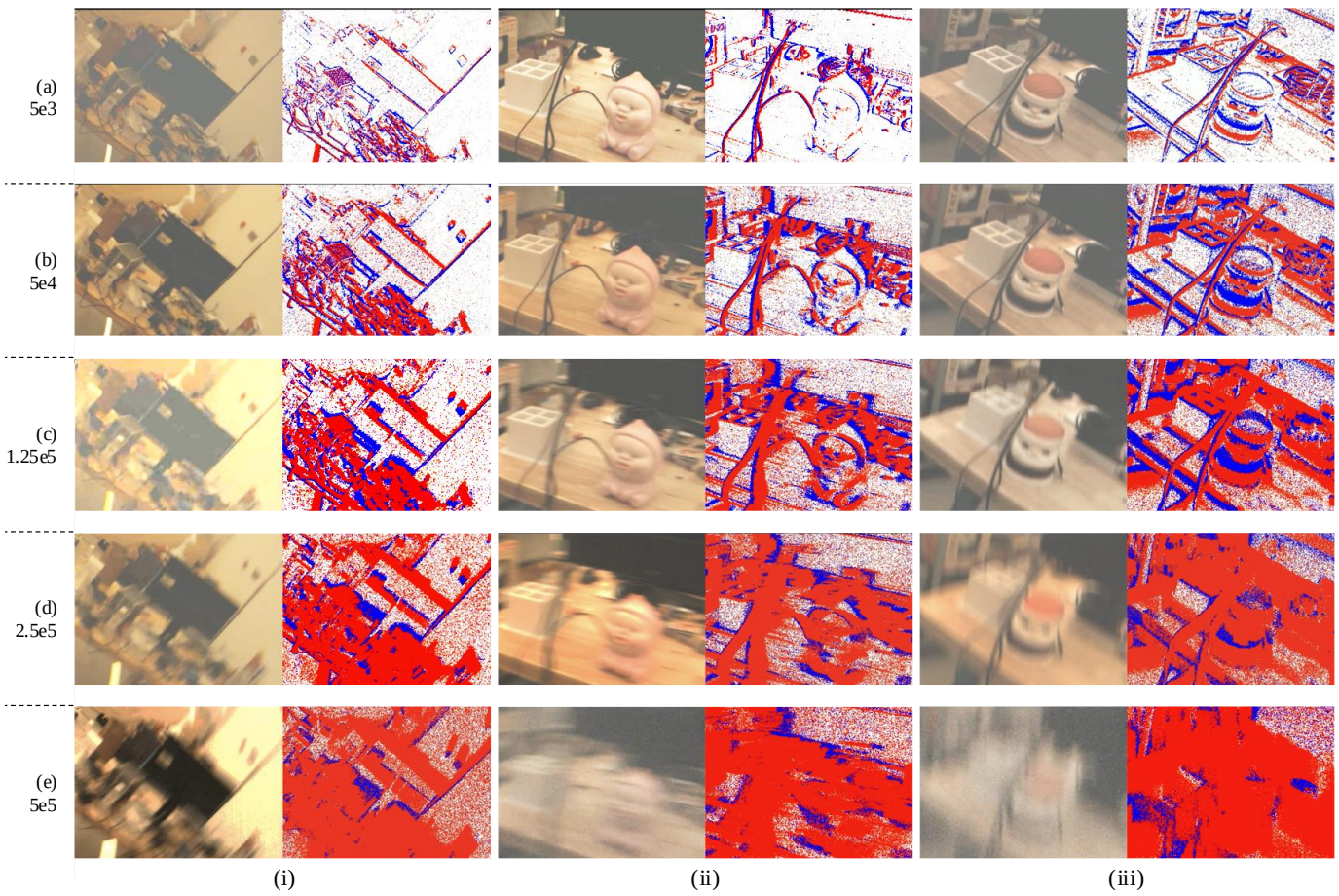


Fig. 10. Different blur degrees for each scene in the real-world dataset captured using the DVS346 camera. To capture videos with varying blur levels, we controlled the exposure times across different scenes. The dataset contains frames with varying exposure durations, displayed in the following order: (a) 5e3, (b) 5e4, (c) 1.25e5, (d) 2.5e5, and (e) 5e5 μs.

shown in Figure 8, the proposed camera trajectories feature an average displacement exceeding 10 meters, with the focal view dynamically shifting in every frame, imposing significant challenges for general deblurring and reconstruction models. Following prior works [2], [6], we set the number of rendered images to 34 for both normal and hard scenes. Figure 7 presents visualizations of our crafted scenes. A detailed overview of the design objectives for these challenging scenes is provided below:

- *Town*: We created this urban scene with a tram and parked cars to test the robustness of deblurring and 3D reconstruction algorithms in handling dynamic urban environments with various moving and stationary objects.
- *Sanatorium*: We created this scene featuring a grand mansion and its surrounding garden to evaluate the accuracy and detail preservation of deblurring and 3D reconstruction algorithms in capturing intricate architectural elements and lush natural environments.
- *Commercial Street*: We created this nighttime urban scene with complex lighting, reflective surfaces and dynamic elements like vehicles and pedestrians.
- *Boulevard*: We want to test the model to handle multiple light sources, reflectionsand the complexity of urban traffic at night, which includes numerous vehicles and varying light intensities.
- *Archives*: This scene was designed in an interior setting with abundant natural light, complex architectural details, and varying depths, simulating the challenges of capturing a well-lit, intricate space with multiple little books.
- *Restaurant*: This scene was crafted to test model to handle complex lighting scenarios, multiple objects, and varying degrees of occlusion in a dimly lit outdoor setting with reflective surfaces and string lights.
- *Edifice*: This urban scene was created to evaluate model to process complex materials like glass and metal, handle dynamic lighting conditions, and capture intricate details amidst the hustle of traffic and architectural variety.

Appendix D
Real-World Dataset with Multiple Levels of Motion Blur

To evaluate performance in real-world scenarios, we manually recorded five scenes using the Color DAVIS346 event camera [50], which features a resolution of 340×260 pixels and an exposure time of 100 milliseconds for RGB frames. To

TABLE V
PERFORMANCE COMPARISON ACROSS DIFFERENT MODELS AND ENVIRONMENTS FOR PSNR↑ IN NOVEL VIEWS SYNTHESIS.

Models	Hard							Normal				
	Sanatorium	Com. Street	Archives	Restaurant	Boulevard	Edifice	Town	Cozyroom	Factory	Pool	Tanabata	Wine
Blurry-NeRF	20.58	21.64	17.93	18.70	23.39	22.97	25.03	26.48	23.41	24.45	19.65	17.60
Blurry-GS	20.90	21.60	18.44	18.99	24.10	23.39	26.69	27.73	23.91	24.56	20.11	18.18
UFP-GS	21.77	23.97	20.11	21.86	26.49	23.02	27.54	28.90	27.16	29.56	24.62	24.19
EFN-GS	21.85	23.98	20.28	21.29	26.52	22.77	27.41	28.09	25.84	29.42	23.88	24.07
E^2GS	22.00	24.18	19.92	21.29	26.89	23.13	27.64	30.44	26.96	31.02	26.05	25.92
BAD-NeRF	22.16	24.43	20.50	21.73	27.84	23.48	28.42	30.92	27.96	32.60	28.16	27.12
BAD-Gaussians	25.59	18.95	**22.34**	**23.80**	28.32	25.42	25.97	32.35	28.18	33.30	31.00	**31.44**
EvDeblurNeRF	22.20	24.67	20.82	21.82	27.85	23.61	28.44	32.84	28.95	35.82	30.27	29.59
Ours	**24.88**	**26.63**	22.06	23.72	**33.30**	**26.55**	**31.59**	**33.14**	**30.25**	**36.30**	**31.34**	29.45

TABLE VI
PERFORMANCE COMPARISON ACROSS DIFFERENT MODELS AND ENVIRONMENTS FOR SSIM↑ IN NOVEL VIEWS SYNTHESIS.

Models	Hard							Normal				
	Sanatorium	Com. Street	Archives	Restaurant	Boulevard	Edifice	Town	Cozyroom	Factory	Pool	Tanabata	Wine
Blurry-NeRF	.6215	.5375	.5405	.4604	.7366	.7285	.7306	.8621	.8178	.6119	.8026	.6393
Blurry-GS	.6316	.5504	.5553	.4733	.7719	.7519	.7855	.8774	.8379	.6149	.8198	.6617
UFP-GS	.6431	.6687	.5987	.5867	.8410	.7524	.7948	.9035	.8307	.8199	.8311	.7916
EFN-GS	.6504	.6724	.6080	.5872	.8457	.7434	.7920	.8731	.8079	.8154	.7949	.7866
E^2GS	.6613	.6782	.6010	.5929	.8536	.7571	.8023	.9331	.8761	.8634	.8828	.8555
BAD-NeRF	.6684	.6955	.6221	.6131	.8834	.7676	.8275	.9362	.8905	.9064	.9057	.8779
BAD-Gaussians	**.8140**	.4563	**.7253**	**.7438**	.8894	.8166	.7404	.9278	.9164	.8964	.9427	**.9443**
EvDeblurNeRF	.7057	.7372	.6554	.6618	.9235	.8022	.8744	.9566	.9158	.9521	.9471	.9233
Ours	.7828	**.7982**	.6976	.7359	**.9657**	**.8642**	**.9202**	**.9607**	**.9293**	**.9575**	**.9562**	.9229

TABLE VII
PERFORMANCE COMPARISON ACROSS DIFFERENT MODELS AND ENVIRONMENTS FOR LPIPS↓ IN NOVEL VIEWS SYNTHESIS.

Models	Hard							Normal				
	Sanatorium	Com. Street	Archives	Restaurant	Boulevard	Edifice	Town	Cozyroom	Factory	Pool	Tanabata	Wine
Blurry-NeRF	.3722	.4437	.4735	.4428	.3140	.2880	.2567	.2718	.2656	.4159	.1894	.4463
Blurry-GS	.3559	.4178	.4400	.4328	.2707	.2485	.1955	.1250	.1579	.3629	.1261	.3025
UFP-GS	.4126	.3032	.2940	.4055	.1237	.2233	.1804	.1035	.2369	.2241	.2251	.2528
EFN-GS	.4108	.3194	.2928	.4136	.1314	.2320	.1775	.1433	.2520	.2275	.2731	.2673
E^2GS	.3914	.3129	.3001	.4141	.1222	.2187	.1736	.0628	.1708	.1387	.1375	.1590
BAD-NeRF	.3412	.2705	.2433	.3613	.0879	.1906	.1331	.0501	.1375	.0937	.0954	.1226
BAD-Gaussians	**.1591**	.3719	.1542	**.2257**	.0909	.1388	.3439	.0340	.0990	.0551	.0478	**.0345**
EvDeblurNeRF	.2781	.2386	.2011	.3345	.0647	.1608	.1082	.0314	.1011	.0377	.0437	.0569
Ours	.1944	**.1424**	**.1532**	.2492	**.0348**	**.0952**	**.0797**	**.0253**	**.0778**	**.0254**	**.0279**	.0467

further facilitate the reality demand, we construct the this real-world blur datasets. We equip the DAVIS346 event camera on the UR-5E robotic arm as shown in Figure 9. The goal of our dataset is to capture video frames and event data from the same scene with varying blur levels. To achieve this, we controlled the camera's motion using a robotic arm that performed non-linear movements while adjusting the exposure time to induce different levels of blur. For accurate alignment of the videos with varying blur degrees, we used an IMU with a 1000 Hz sampling rate in combination with a registration algorithm to achieve millisecond-level temporal synchronization. We define 5 tasks and each task contains 5 kinds of blur in different exposure time (5e3, 5e4, 1.25e5, 2.5e5 and 5e5). We design a time interval alignment algorithm to match and correct all images and event streams. Finally, we utilize the blur degree of 5e3 as the ground truth sets and the scenes at other exposure times are used as a blur dataset. The images we captured are shown in Figure 10. We found that each scenes with different blur and limited image numbers (less than 30) easily fails in initialization with COLMAP. Hence, we initialize these scenes by our method and simulate the sparse condition by random sampling 1k points on point clouds.

APPENDIX E
DETAILED EXPERIMENTS

A. Synthetic Experiments

Here, we present the detailed performance evaluation of various state-of-the-art (SOTA) methods across different syn-

TABLE VIII
QUANTITATIVE COMPARISONS OF DEBLUR VIEW RENDERING ACROSS REAL-WORLD SCENES. BEST RESULTS IN **BOLD**.

Scene Type	Metric	B-NeRF	B-3DGS	UFP-GS	EFN-GS	E^{2}GS	BAD-NeRF	BAD-GS	EDNeRF	Ours
Task1	PSNR↑	19.08	20.87	22.52	22.63	25.05	25.75	26.28	26.31	**27.07**
	SSIM↑	.6876	.7402	.7900	.7899	.8253	.8289	.8307	.8303	**.8546**
	LPIPS↓	.4115	.3311	.1114	.1172	.0972	.0932	.0929	.0916	**.0679**
Task2	PSNR↑	13.21	14.23	12.03	12.77	13.84	18.63	20.47	20.60	**29.03**
	SSIM↑	.5844	.6112	.4267	.5061	.4292	.5030	.5264	.5326	**.7766**
	LPIPS↓	.4180	.3452	.3807	.3320	.3925	.3492	.3211	.3165	**.0645**
Task3	PSNR↑	16.25	17.86	20.82	21.98	23.89	24.50	24.82	24.66	**28.39**
	SSIM↑	.6016	.6721	.6726	.7198	.7281	.7323	.7259	.7162	**.8364**
	LPIPS↓	.3555	.2665	.3108	.2745	.2373	.2272	.2223	.2332	**.1051**
Task4	PSNR↑	17.27	20.28	16.38	17.99	19.26	18.42	19.34	20.32	**25.76**
	SSIM↑	.7107	.7798	.6791	.6871	.6803	.6612	.6768	.6914	**.8016**
	LPIPS↓	.3018	.1965	.2471	.2305	.2337	.2605	.2500	.2276	**.0923**
Task5	PSNR↑	18.01	19.98	22.92	22.66	24.21	24.34	24.85	24.53	**25.98**
	SSIM↑	.6982	.7541	.8195	.8150	.8313	.8307	.8330	.8295	**.8556**
	LPIPS↓	.3122	.2330	.1038	.1162	.0986	.0921	.0880	.0931	**.0651**

TABLE IX
ANALYSIS OF WEIGHT λ_e IN OPTIMIZATION OBJECTIVES.

λ_e	Hard			Normal		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
5e-3	26.94	.8235	.1355	**32.09**	.9453	**.0406**
5e-2	23.14	.7149	**.1114**	32.61	**.9476**	.0471
5e-1	27.21	.8357	.1301	31.89	.9420	.0557
1.0	**27.33**	**.8372**	.1298	31.24	.9320	.0670

TABLE X
ANALYSIS OF DIFFERENT SAMPLING NUMBER IN INITIALIZATION.

Num.	Hard			Normal		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
5,000	**26.94**	**.8235**	.1355	**32.09**	**.9453**	**.0406**
10,000	26.76	.8198	.1341	31.97	.9459	.0458
50,000	26.43	.8070	.1402	31.35	.9336	.0511
100,000	26.25	.8012	**.1064**	30.73	.9237	.0553

TABLE XI
ANALYSIS OF INTERVAL M NUMBERS IN INITIALIZATION.

M	Hard			Normal		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
5	26.53	.8141	.1411	31.95	.9442	.0424
10	26.81	.8188	.1383	31.92	.9442	**.0357**
20	26.86	.8222	**.1328**	31.11	**.9463**	.0404
40	**26.94**	**.8235**	.1355	**32.09**	.9453	.0406
80	26.73	.8188	.1399	32.00	.9439	.0419

TABLE XII
ANALYSIS OF THE PROPOSED MODULES FOR POSE ESTIMATION. THE RESULTS ARE IN THE ABSOLUTE TRAJECTORY ERROR METRIC (ATE). NOTE THAT COLMAP INITIALIZATION IS BASED ON SHARP IMAGES, WHILE OUR METHOD UTILIZE BLUR IMAGES.

Methods	Before Training		After Training	
	COLMAP	Ours	Ours w/o event	Ours w/ event
Average	1.7590	**0.2742**	0.0560	**0.0471**
Archives	1.1187	**0.1459**	0.1127	**0.0964**
Edifice	2.6226	**0.0181**	0.0292	**0.0171**
Com. Street	1.1329	**0.0949**	0.0632	**0.0582**
Restaurant	1.1627	**0.0536**	0.0214	**0.0111**
Sanatorium	2.7585	**0.3272**	0.1312	**0.1192**
Town	1.9840	**0.7073**	0.0203	**0.0177**
Boulevard	1.6330	**0.5720**	0.0183	**0.0079**

thetic scenarios, as shown in Tables V, VI, and VII. In terms of PSNR, our method consistently achieves state-of-the-art performance across multiple scenarios, obtaining the highest PSNR values in 8 out of the 12 evaluated scenes. Notably, it significantly outperforms existing approaches in challenging environments such as Boulevard (33.30 dB), Town (31.59 dB), and Tanabata (31.34 dB), surpassing the second-best results by margins of 4.98 dB, 3.15 dB, and 0.34 dB, respectively. These results demonstrate the superior robustness of our method to motion blur in complex scenarios compared to existing SOTA methods. Furthermore, while maintaining competitive performance in Factory (30.25 dB) and Pool (36.30 dB), our approach exhibits strong robustness in both Hard and Normal scenes. The consistent improvement over baseline models, such as BAD-GS (e.g., +1.74 dB in Sanatorium), highlights its enhanced capability in handling complex scene degradations. This performance underscores the effectiveness of our method in balancing high-quality reconstruction with strong generalization across diverse environments.

B. Real-World Experiments

For deblurring reconstruction, DeblurSplat demonstrates superior performance across all tasks, achieving the highest PSNR, SSIM, and the lowest LPIPS values in all five scenarios, as shown in Table VIII. Notably, it significantly outperforms existing approaches in the challenging Task2, achieving a PSNR of 29.03 dB (compared to 20.60 dB from

TABLE XIII
QUANTITATIVE COMPARISONS OF NOVEL VIEW SYNTHESIS (NVS) ACROSS REAL-WORLD SCENES. BEST RESULTS IN **BOLD**.

Scene Type	Metric	B-NeRF	B-3DGS	UFP-GS	EFN-GS	E^{2}GS	BAD-NeRF	BAD-GS	EDNeRF	Ours
Task1	PSNR↑	18.33	19.96	22.65	22.40	23.64	23.89	24.45	24.44	**25.16**
	SSIM↑	.6921	.7421	.7889	.7815	.7861	.7867	.7883	.7894	**.7982**
	LPIPS↓	.2331	.1775	.1176	.1169	.1006	.0967	.0970	.0971	**.0806**
Task2	PSNR↑	19.14	20.74	11.98	12.97	14.11	19.41	21.53	21.50	**27.98**
	SSIM↑	.6843	.7339	.4554	.4718	.4492	.5279	.5522	.5538	**.7582**
	LPIPS↓	.2550	.1950	.3736	.3449	.3846	.3365	.3058	.3028	**.0591**
Task3	PSNR↑	9.240	10.15	21.52	22.19	22.43	22.89	23.37	23.53	**25.64**
	SSIM↑	.3877	.4185	.7068	.7292	.6882	.6913	.6907	.6863	**.7814**
	LPIPS↓	.5203	.4840	.2908	.2651	.2604	.2540	.2485	.2523	**.1423**
Task4	PSNR↑	13.15	13.37	15.67	18.10	18.53	17.28	18.18	19.19	**24.22**
	SSIM↑	.4638	.4615	.6606	.6952	.6810	.6463	.6613	.6786	**.7773**
	LPIPS↓	.5012	.4859	.2597	.2216	.2317	.2684	.2593	.2325	**.1004**
Task5	PSNR↑	16.08	19.24	23.10	22.22	24.38	23.94	24.57	24.20	**26.69**
	SSIM↑	.6596	.7440	.8225	.8084	.8273	.8166	.8221	.8168	**.8620**
	LPIPS↓	.4258	.3058	.1029	.1195	.0848	.0867	.0757	.0854	**.0486**

EDNeRF) and reducing LPIPS by 74.3%, highlighting its robustness in handling severe motion blur. The consistent improvements in Task1 (PSNR: 27.07 dB, +0.76 dB over BAD-GS) and Task5 (SSIM: 0.8556, +0.0226 over BAD-GS) further underscore the efficacy of our balanced sampling strategy and event-guided progressive alignment. Additionally, DeblurSplat excels in texture recovery, achieving the lowest LPIPS of 0.0923 in Task4, surpassing prior methods by approximately 60%. These results validate the advantages of integrating dense stereo priors and event streams, effectively mitigating pose initialization errors and blur-induced geometry misalignment. Our approach establishes a new state-of-the-art in deblurring reconstruction.

APPENDIX F
ABLATION STUDY

Optimization Objectives. We explore different λ_e settings from Eq. 8 to assess model performance. As shown in Table IX, we conclude that event-based guidance and latent blur simulation are crucial for reconstruction performance in challenging scenarios when $\lambda_e \geq$ 5e-1. On the other hand, setting λ_e = 5e-2 leads to performance collapse, likely due to confusion between the event-based prior and the original blurred images, causing the model to fall into sub-optimal solutions.

Initialization Efficiency. We analyze the effect of different initialization interval values (M), as presented in Table XI. Our findings indicate that setting $5 \leq M \leq 20$ hinders fine-grained semantic extraction, while increasing M to 80 overly prioritizes low-confidence points, many of which lack meaningful information. Finally, we examine the impact of different sampling quantities on overall optimization. As shown in Table X, we observe that increasing the sampling quantity during initialization makes global optimization more challenging, aligning with findings from previous works [39], [1]. To further improve the reconstruction performance and robust deblurring, we set M to 40.

Pose Estimation. As shown in Table XII, we employ the Absolute Trajectory Error (ATE) metric to assess the impact of different modules on pose estimation. Compared to our proposed approach, COLMAP initialization, which relies on sharp images, produces significantly higher errors. This is attributed to its limited robustness when handling blurred images, where semantic confusion degrades pose estimation accuracy. In contrast, our method leverages the rich priors encoded in a dense stereo module pretrained on large-scale sharp datasets, enabling more accurate camera pose predictions. Furthermore, the results in Table XII demonstrate that the event-based Progressive Alignment module effectively refines pose estimation by implicitly guiding optimization in challenging scenarios, ultimately contributing to improved deblurring reconstruction.

APPENDIX G
LIMITATION AND FUTURE

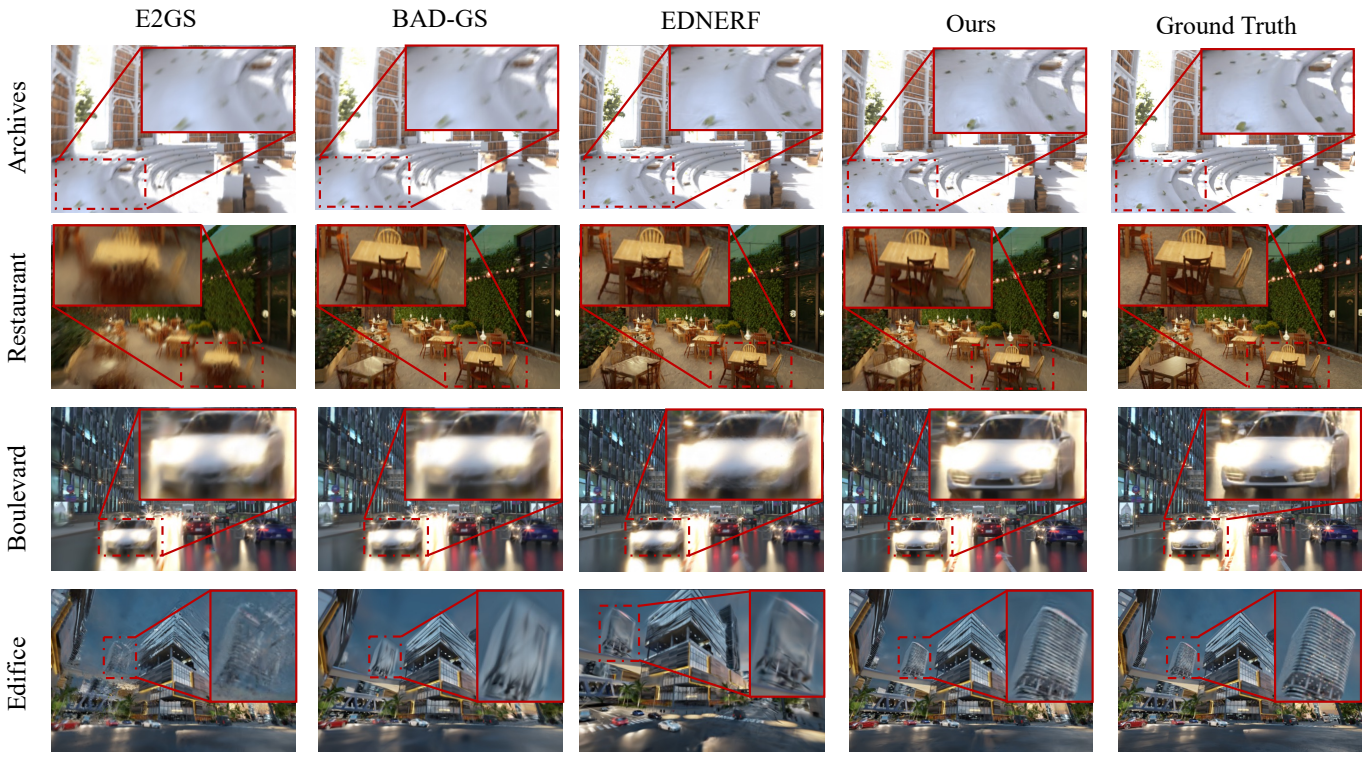
Confidence Evaluation. We rely on confidence estimation to filter point clouds generated by DUSt3R. However, its accuracy under ambiguous conditions remains unclear, especially given that DUSt3R is trained on high-quality, sharp images. A more rigorous quantitative standard for assessing point cloud quality is needed to ensure reliable initialization of gaussian primitives.

Large-Scale Test. The core concept of our approach leverages the rich knowledge from DUSt3R and event streams to tackle motion-blur reconstruction. Naturally, this raises the question of whether the same mechanism can be extended to more complex reconstruction scenarios, such as rolling-shutter effects.

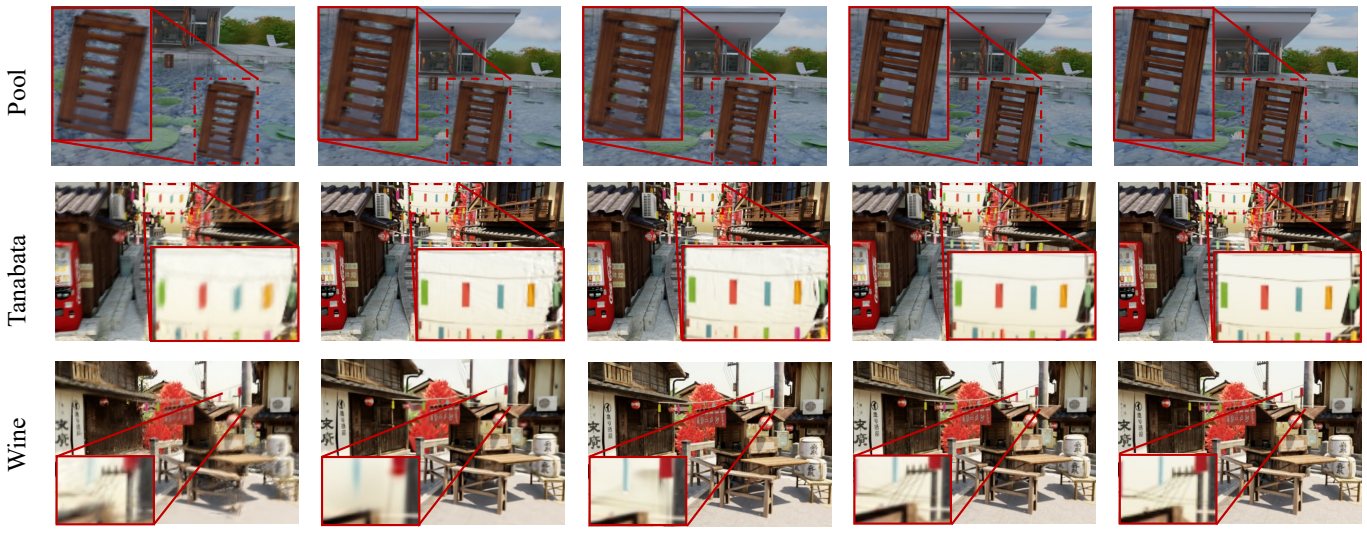
APPENDIX H
MORE VISUALIZATION

Qualitative results are presented in Figure 11, 12, 13 and 14. Our findings indicate that while previous state-of-the-art methods perform well in normal scenes, they struggle in

challenging cases like hard scenes, particularly when handling complex textures and large-scale camera motion, often leading to severe blurring. In contrast, our method significantly outperforms these baselines by rendering high-fidelity deblur views and novel views.



(a) Overview of Rendering Novel Views in Hard Scenes



(b) Overview of Rendering Novel Views in Normal Scenes

Fig. 11. Qualitative comparison on the remaining synthetic dataset on novel views synthesis. It shows that our method achieves better performance in rendering novel views compared to the other SOTA methods.



Fig. 12. Qualitative comparison on the normal scenes dataset on deblur views rendering. It shows that our method achieves better performance in deblur reconstruction compared to the other SOTA methods.

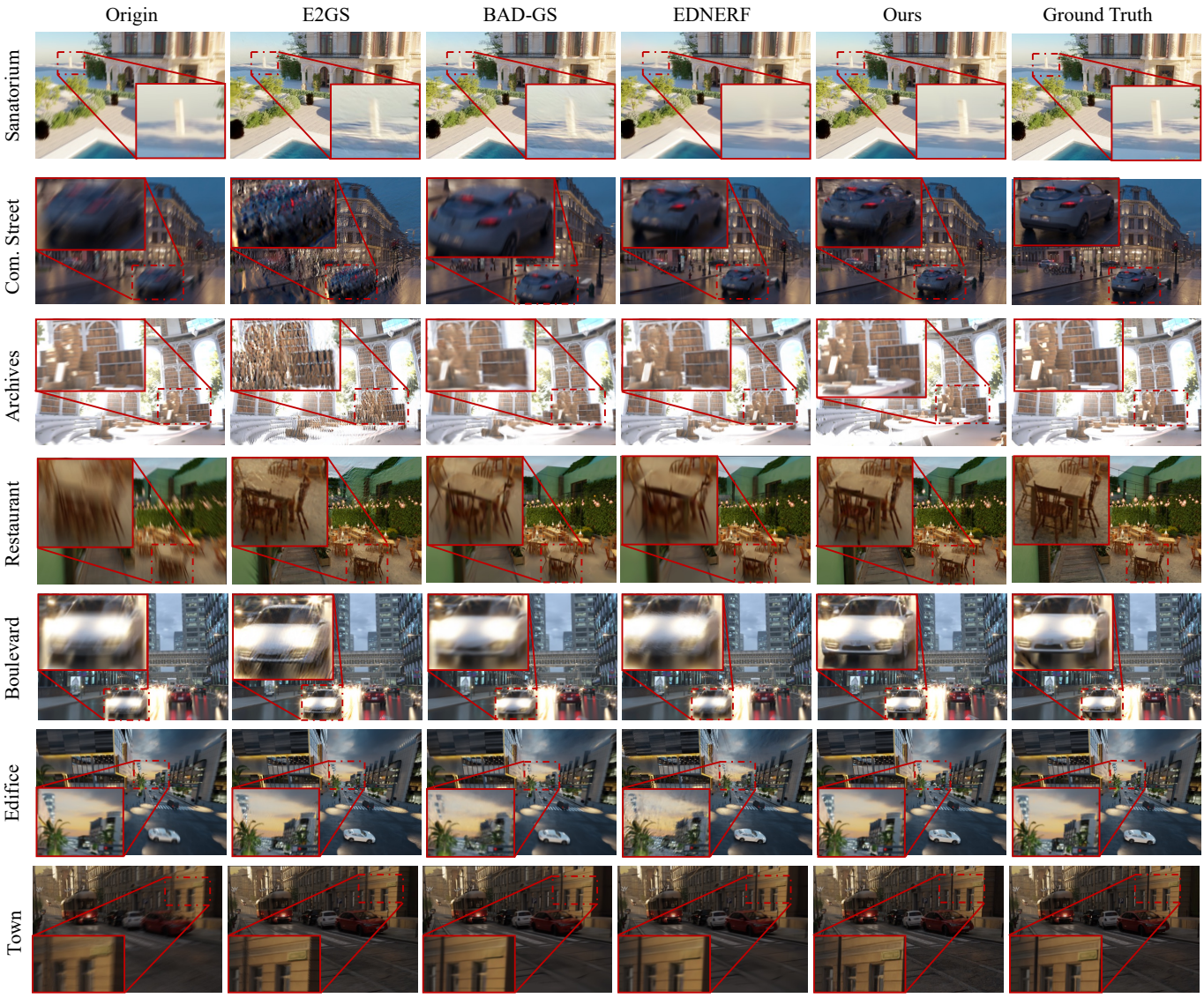


Fig. 13. Qualitative comparison on the hard scenes dataset on deblur views rendering. It shows that our method achieves better performance in deblur reconstruction compared to the other SOTA methods.

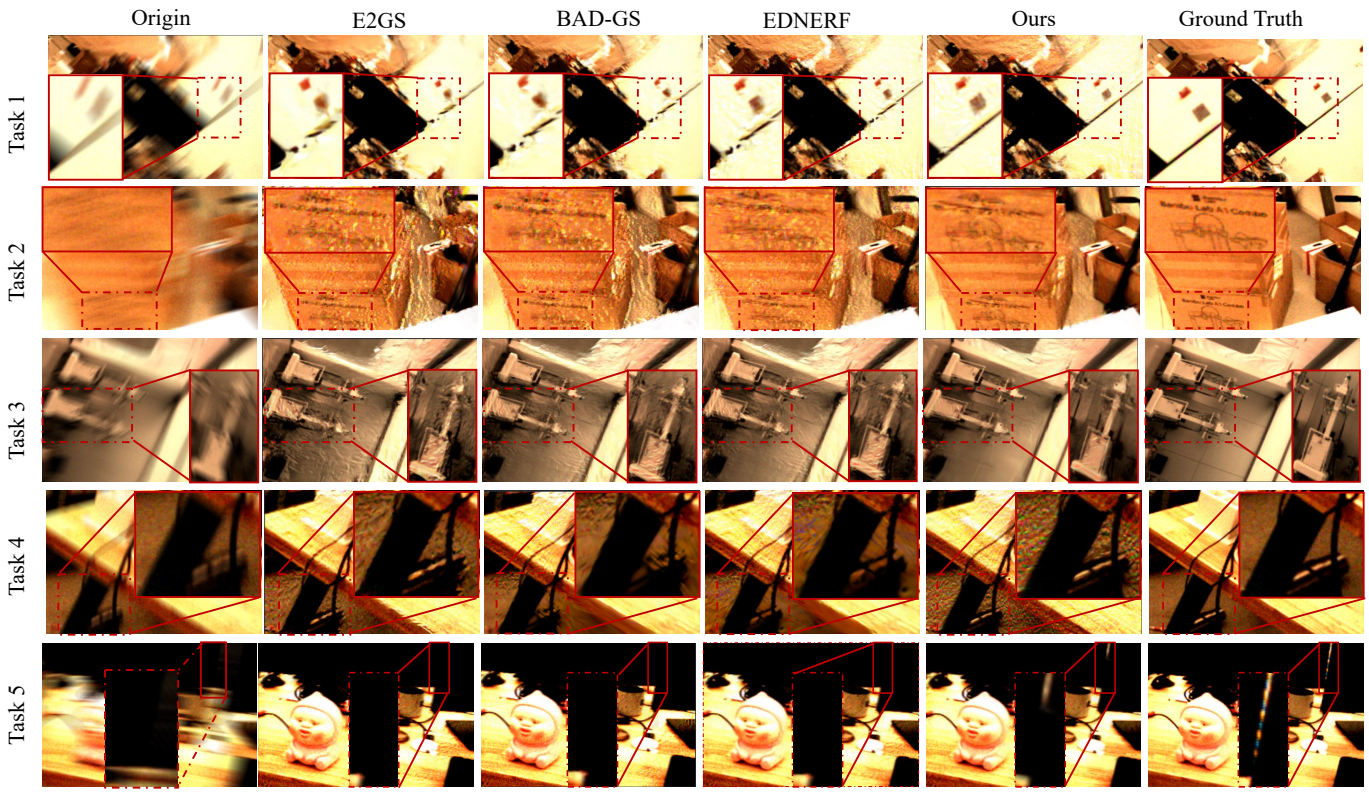


Fig. 14. Qualitative comparison on the real-world scenes on deblur views rendering.