BI-FLOWGS: BRIDGING GENERATIVE VIEW COMPLETION AND GAUSSIAN GEOMETRY THROUGH BIDIRECTIONAL FLOW CO-REFINEMENT

Anonymous authors
Paper under double-blind review

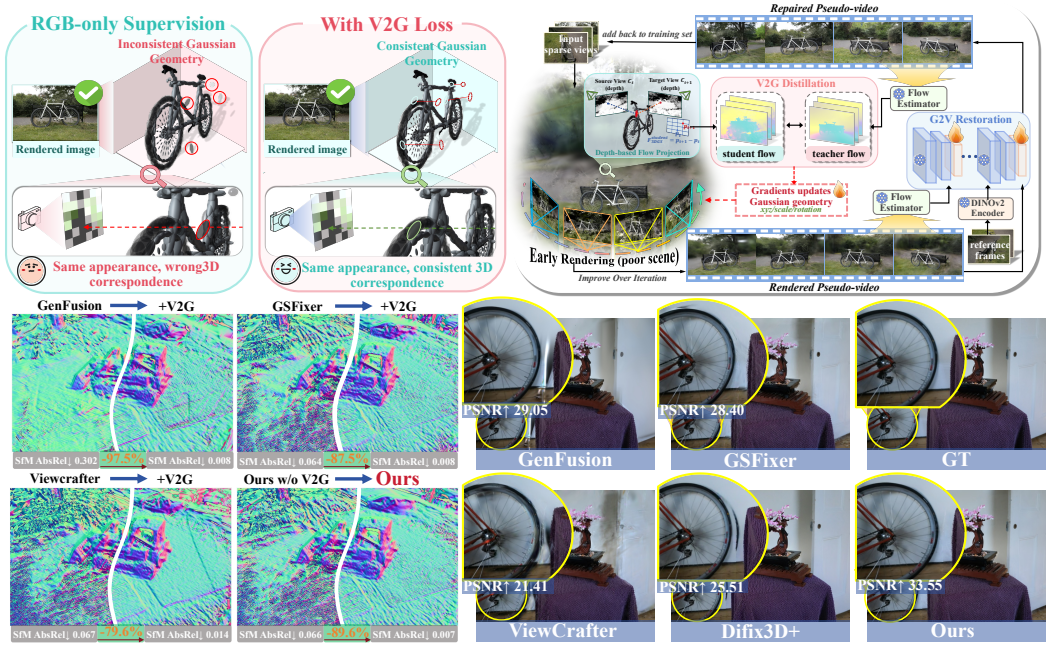


Figure 1: **Bi-FlowGS.** *Top left:* V2G alleviates *Geometry Cheating*. *Top right:* V2G and G2V form a bidirectional flow loop between video restoration and Gaussian geometry optimization. *Bottom:* V2G consistently improves diverse diffusion-based reconstruction frameworks, while the complete Bi-FlowGS achieves strong rendering quality and geometric consistency.

ABSTRACT

Sparse-view 3D scene reconstruction with 3D Gaussian Splatting (3DGS) is inherently underconstrained. Plausible renderings can also coexist with erroneous Gaussian geometry, as errors in positions or depths may be concealed by opacity, scale, and appearance; we term this failure mode ***Geometry Cheating***. Existing regularization methods constrain geometry but remain limited to observed views, while video-diffusion-based methods complete unseen views yet mainly use them as RGB pseudo-supervision, underusing motion and temporal priors and lacking explicit geometry supervision. We present **Bi-FlowGS**, which capitalizes on optical flow to bridge generative view completion and Gaussian geometry regularization. Our plug-and-play **Video-to-Geometry Flow Distillation (V2G)** distills temporal correspondence priors from restored videos into Gaussian geometry to alleviate *Geometry Cheating*. Conversely, **Geometry-to-Video Flow-Guided Restoration (G2V)** uses the current 3DGS geometry to guide temporally consistent video restoration, providing more reliable generative supervision. Together, V2G and G2V form an **implicit bidirectional co-refinement** process, enabling restored videos and the optimized 3DGS scene to iteratively improve each other. Experiments demonstrate improved rendering quality and geometric consistency across wide-baseline and unbounded 360° benchmarks.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

3D Gaussian Splatting (3DGS) enables high-quality scene reconstruction and novel-view synthesis (Kerbl et al., 2023), but relies heavily on dense multi-view observations. Under sparse inputs, insufficient supervision often leads to overfitting, floaters, and unstable geometry. More critically, plausible RGB renderings can still arise from inaccurate Gaussian geometry: erroneous positions or depths can be concealed through compensation among opacity, scale, and appearance. We refer to this discrepancy between rendering fidelity and geometric correctness as **Geometry Cheating**.

Existing sparse-view reconstruction methods mainly follow two directions. **(a) Regularization-based methods** reduce ambiguity through semantic, depth, structural, or geometric priors (Niemeyer et al., 2022; Wang et al., 2023; Xiong et al., 2025; Li et al., 2024; Chen et al., 2025), but remain largely tied to observed views and provide limited supervision for unseen regions. **(b) Generative completion-based methods** expand scene observability by synthesizing or restoring unseen views with image or video diffusion models (Zhou & Tulsiani, 2023; Gao et al., 2024; Yu et al., 2024; Wu et al., 2025b; Yin et al., 2025). However, generated views are typically used mainly as RGB pseudo-supervision, leaving video temporal-motion priors underexploited and Gaussian geometry without explicit supervision. Taken together, these limitations reveal a **barrier between generative view completion and geometry regularization**: **(i)** geometry regularization can refine Gaussian geometry but cannot expand beyond observed views; **(ii)** generative completion can recover unseen views but does not directly optimize Gaussian geometry.

To bridge this barrier, we present **Bi-FlowGS**, which uses **optical flow as the central hub** between generative view completion and Gaussian geometry optimization. **(i)** We introduce plug-and-play **Video-to-Geometry Flow Distillation (V2G)**, which aligns teacher flow from restored videos with differentiable geometry-induced flow from 3DGS, distilling temporal-motion and implicit geometric priors into Gaussian geometry to directly supervise geometry optimization and alleviate *Geometry Cheating*. **(ii)** We further introduce **Geometry-to-Video Flow-Guided Restoration (G2V)** to improve the temporal correspondences of restored videos. Since reliable inter-frame correspondences are crucial for V2G supervision, G2V goes beyond conventional camera, coarse-geometry, or appearance conditioning (He et al., 2025; Kwak et al., 2024; Yu et al., 2024) by injecting cross-view flow and geometric cues from the current reconstructed 3DGS scene into video diffusion, translating static-scene cross-view correspondences into direct temporal-motion consistency guidance. Together with DINOv2 reference features (Oquab et al., 2023), it produces videos with scene-consistent temporal motion and fine-grained texture details. **(iii)** Together, V2G and G2V form an **implicit bidirectional co-refinement loop**: improved videos provide more reliable teacher flow and pseudo-view supervision for 3DGS, while the optimized 3DGS scene provides more accurate flow and geometric guidance for subsequent video restoration, allowing both to iteratively improve each other.

As shown in Figure 1, V2G consistently improves Gaussian geometry across representative video-diffusion-based reconstruction frameworks, while G2V promotes temporally consistent video restoration to provide more reliable teacher-flow and pseudo-view supervision. Together, Bi-FlowGS further improves rendering fidelity and cross-view geometric consistency. Extensive experiments on wide-baseline and unbounded 360° sparse-view benchmarks demonstrate its effectiveness and generality.

Our contributions are summarized as follows:

- **Video-to-Geometry Flow Distillation.** We introduce plug-and-play V2G distillation that converts the temporal-motion and implicit geometric priors of video diffusion models into explicit supervision for Gaussian geometry, providing a new geometry-optimization mechanism for diffusion-completion-based sparse-view reconstruction.
- **Geometry-to-Video Flow-Guided Restoration.** We inject cross-view flow and geometric cues from the current reconstructed 3DGS scene into video diffusion, translating static-scene cross-view correspondences into temporal-motion consistency guidance, while reference features preserve fine-grained texture details, thereby improving the reliability of teacher-flow and pseudo-view supervision.
- **Implicit Bidirectional Co-Refinement.** We formulate an implicit bidirectional optimization loop that bridges generative view completion and Gaussian geometry refinement through optical-flow feedback, allowing video supervision and geometric guidance to mutually strengthen over iterations.

2 Related Work

Conventional Sparse-View 3D Reconstruction. Sparse-view reconstruction is inherently underconstrained. NeRF-based methods reduce ambiguity through semantic, frequency, depth, or solution-space regularization (Jain et al., 2021; Niemeyer et al., 2022; Yang et al., 2023; Wang et al., 2023). Sparse-view 3DGS further introduces depth, structural, and Gaussian regularization (Xiong et al., 2025; Li et al., 2024; Zhu et al., 2024; Huang et al., 2024; Zhang et al., 2024; Park et al., 2025), while feed-forward methods directly predict Gaussian representations from sparse inputs (Charatan et al., 2024; Chen et al., 2024; Xu et al., 2025; Jiang et al., 2025). These methods improve reconstruction under sparse observations but cannot explicitly complete unseen viewpoints to provide additional supervision.

Diffusion-Based Generative View Completion. Diffusion models alleviate missing observations by synthesizing or refining unseen views (Zhou & Tulsiani, 2023; Gao et al., 2024; Sargent et al., 2024; Wu et al., 2024; 2025a; Paliwal et al., 2025; Topaloğlu et al., 2026). Recent video-based methods further alternate video generation or restoration with 3D reconstruction (Liu et al., 2026a; Wu et al., 2025b; Yin et al., 2025; Tang et al., 2026; Zhong et al., 2025; Wang et al., 2026). GenFusion conditions video diffusion on reconstruction-derived RGB-D renderings (Wu et al., 2025b), while GSFixer introduces semantic and geometric reference features for video restoration (Yin et al., 2025). Other approaches incorporate camera, structural, or geometric priors into video generation (Yu et al., 2024; Liu et al., 2026a; Tang et al., 2026; Zhu et al., 2026; Ni et al., 2026; Liu et al., 2026b). Nevertheless, generated views are typically used as RGB or pseudo-view supervision, leaving their temporal correspondences underexploited for geometry optimization.

Direct Geometric Constraints for Sparse-View 3DGS. Beyond general regularization, recent methods explicitly introduce geometric constraints into 3DGS optimization (Hong et al., 2025). FewViewGS uses cross-view feature matching to enforce consistency on sampled novel views (Yin et al., 2024), while FDS leverages pretrained matching flow to guide geometry-induced radiance flow (Chen et al., 2025). These methods strengthen geometric supervision, but remain tied to observed-view correspondences or pretrained matching priors and do not exploit generatively completed views.

3 Method

Figure 2 illustrates **Bi-FlowGS**, which couples video restoration with Gaussian geometry optimization through optical flow. Given K sparse images $\{I_i\}_{i=1}^K$ and poses $\{P_i\}_{i=1}^K$, we reconstruct an initial Gaussian scene $\mathcal{G}$ and render an artifact-prone sequence $\mathcal{V}^G$, which is restored to $\hat{\mathcal{V}}$ by a video diffusion model. The restored video serves two roles: its frames provide pseudo-view supervision, while the extracted optical flow serves as teacher flow for geometry optimization. Our V2G loss aligns this teacher flow with differentiable geometry-induced flow to explicitly supervise Gaussian geometry (Sec. 3.1). In the reverse direction, we introduce G2V, which uses flow and geometric cues from the optimized 3DGS scene with DINOv2 reference features to guide video restoration toward scene-consistent temporal motion and fine-grained textures (Sec. 3.2). Repeating these two directions enables restored videos and the 3DGS scene to iteratively improve each other through bidirectional feedback (Sec. 3.3).

3.1 Video-to-Geometry Flow Distillation

Teacher flow from restored videos. Given a restored video $\widehat{\mathbf{V}} = \{\widehat{I}_t\}_{t=1}^T$ along a known camera trajectory, we sample $(\widehat{I}_i, \widehat{I}_j)$ and estimate teacher flow with a frozen WAFT model (Wang & Deng, 2026): $\mathbf{F}^T_{i\to j} = \mathcal{F}_{\mathrm{WAFT}}(\widehat{I}_i, \widehat{I}_j)$. Forward–backward consistency and flow confidence produce a validity mask $\mathbf{M}^T$ and confidence map $\mathbf{C}^T$. All teacher quantities are detached during Gaussian optimization.

Differentiable geometry-induced flow. For cameras c_i and c_j, the current Gaussian scene renders depths D_i^G and D_j^G. Each source pixel $\mathbf{p}$ is back-projected using $D_i^G(\mathbf{p})$, transformed to camera j, and reprojected onto the target view:

$$\mathbf{p}' = \pi_j\left(T_j T_i^{-1} \pi_i^{-1}\left(\mathbf{p}, D_i^G(\mathbf{p})\right)\right), \qquad \mathbf{F}^S_{i\to j}(\mathbf{p}) = \mathbf{p}' - \mathbf{p}. \tag{1}$$

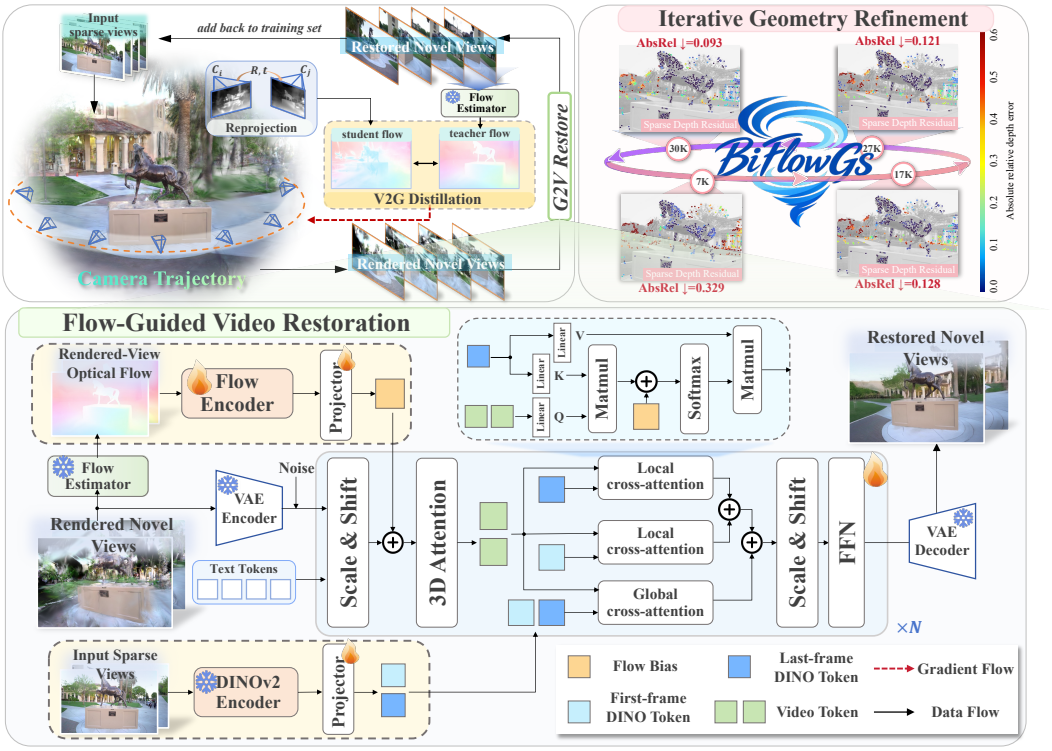


Figure 2: **Bi-FlowGS pipeline.** *Top left:* V2G aligns teacher flow from restored views with differentiable geometry-induced flow, distilling temporal correspondence priors into Gaussian geometry. *Top right:* Iterative co-refinement progressively improves geometric consistency. *Bottom:* G2V uses 3DGS-derived flow and geometric cues, together with DINOv2 reference features, to guide temporally consistent video restoration, whose outputs provide pseudo-view and teacher-flow supervision, forming implicit bidirectional co-refinement.

Here, T_i and T_j are world-to-camera transformations, and π_i^{-1} and π_j denote back-projection and projection. A geometric-validity mask $\mathbf{M}^G$ excludes invalid depths, out-of-bound projections, and occluded correspondences determined by D_j^G.

With fixed cameras, the student flow is governed by rendered depth and directly reflects Gaussian geometry. Its principal gradient path is

$$\frac{\partial \mathcal{L}_{\text{V2G}}}{\partial \mathbf{\Theta}_{\mathcal{G}}} = \frac{\partial \mathcal{L}_{\text{V2G}}}{\partial \mathbf{F}^S_{i\to j}} \frac{\partial \mathbf{F}^S_{i\to j}}{\partial D_i^G} \frac{\partial D_i^G}{\partial \mathbf{\Theta}_{\mathcal{G}}}, \tag{2}$$

where $\mathbf{\Theta}_{\mathcal{G}}$ denotes the Gaussian parameters involved in depth rendering, including the center coordinates $\boldsymbol{\mu}$. Thus, flow errors provide explicit 3D supervision for Gaussian centers, reducing sparse-view spatial ambiguity and mitigating Geometry Cheating.

Reliability-aware V2G loss. We define the pixel-wise reliability weight as $\mathbf{W}(\mathbf{p}) = \mathbf{M}^T(\mathbf{p})\mathbf{M}^G(\mathbf{p})(\mathbf{C}^T(\mathbf{p}))^\gamma$, where γ controls confidence weighting. The V2G loss is

$$\mathcal{L}_{\text{V2G}} = \frac{\sum_{\mathbf{p}} \mathbf{W}(\mathbf{p})\, \rho\left(\mathbf{F}^S_{i\to j}(\mathbf{p}) - \mathbf{F}^T_{i\to j}(\mathbf{p})\right)}{\sum_{\mathbf{p}} \mathbf{W}(\mathbf{p})}, \tag{3}$$

where ρ is the Smooth-L_1 penalty over both flow components. The exact denominator stabilization is specified in Appendix B.

3.2 Geometry-to-Video Flow-Guided Restoration

Flow-Geometry conditioning (FlowCond) for video diffusion. We initialize the restoration model from a pretrained DiT-based CogVideoX model (Yang et al., 2025) and fine-tune it in latent space.

Given a rendered clip $\mathbf{V} = \{I_t\}_{t=1}^{T}$ with endpoint reference views I^0 and I^1, a frozen WAFT model (Wang & Deng, 2026) estimates consecutive flow $\mathbf{F}_t = (\mathbf{F}_{x,t}, \mathbf{F}_{y,t})$ for $t = 1, \ldots, T-1$.

To improve robustness to unreliable intermediate renderings, we construct $\mathbf{Q}^F = \mathrm{Concat}(\mathbf{M}^D, \mathbf{G}^D, \mathbf{F}_x, \mathbf{F}_y)$, where $\mathbf{M}^D$ measures consistency between rendered and monocular depths, and $\mathbf{G}^D$ captures structural boundaries from normalized monocular depth. We adopt the flow encoder from FloVD (Jin et al., 2025) to extract block-aligned multi-scale features $\{\mathbf{\Phi}_l^F\}_{l=1}^{S} = E_{\text{flow}}(\mathbf{Q}^F)$. In parallel, a frozen DINOv2 encoder (Oquab et al., 2023) extracts endpoint reference tokens $\mathbf{S}^r = E_{\text{DINO}}(I^r)$ for $r \in \{0, 1\}$.

Geometry-to-Video flow-guided feature injection. We use flow for two complementary purposes: motion-consistent feature propagation and correspondence-aligned reference retrieval. First, multi-scale flow features are injected before self-attention:

$$\mathbf{H}_l^{\text{SA}} = \mathbf{H}_l + \mathrm{Attention}_l^{\text{SA}}\left(\mathrm{Norm}_1(\mathbf{H}_l) + \gamma_l \mathbf{\Phi}_l^F\right), \tag{4}$$

where $\mathbf{H}_l$ denotes the input tokens of the l-th Transformer block and γ_l controls the injection strength. This encourages spatiotemporal feature propagation to follow the motion structure of the rendered sequence.

Second, flow guides appearance retrieval from the endpoint references. Let $\mathbf{X}_l = \mathrm{Norm}_{\text{cross}}(\mathbf{H}_l^{\text{SA}})$. A global branch retrieves appearance from all reference tokens as $\mathbf{A}_l^{\text{global}} = \mathrm{CrossAttention}_l(\mathbf{X}_l, [\mathbf{S}^0; \mathbf{S}^1])$, while a local branch uses flow-derived correspondence biases for spatially aligned retrieval:

$$\mathbf{A}_{l,t}^{\text{local}} = \sum_{r=0}^{1} w_t^r \ \mathrm{CrossAttention}_l\left(\mathbf{X}_{l,t}, \mathbf{S}^r; \mathbf{B}_{l,t}^{r,F}\right). \tag{5}$$

Here, $\mathbf{B}_{l,t}^{r,F}$ denotes the flow-derived attention bias. We define $\tau_t = (t-1)/(T-1)$, $w_t^0 = 1 - \tau_t$, and $w_t^1 = \tau_t$ to smoothly interpolate the two endpoint references along the camera trajectory.

Finally, the global and local features are fused and modulated by depth consistency. Defining $\widehat{\mathbf{M}}_l^D = \mathrm{Resize}(\mathbf{M}^D)$, we compute

$$\mathbf{A}_l = \left[\mathbf{A}_l^{\text{global}} + \alpha_l^{\text{local}}\left(\mathbf{A}_l^{\text{local}} - \mathbf{A}_l^{\text{global}}\right)\right] \odot \left[1 + \alpha_l^{\text{conf}}\left(1 - \widehat{\mathbf{M}}_l^D\right)\right]. \tag{6}$$

This design combines global appearance consistency with flow-guided local correspondence, while strengthening reference conditioning in geometrically unreliable regions.

3.3 Restored-Video Feedback Optimization

Optimization objective. We apply the same photometric objective to both real and restored views (Kerbl et al., 2023; Wang et al., 2004):

$$\mathcal{L}_{\text{photo}} = (1 - \lambda_{\text{SSIM}})\,\mathcal{L}_1 + \lambda_{\text{SSIM}}\,(1 - \text{SSIM}). \tag{7}$$

The full objective is

$$\mathcal{L} = \mathcal{L}_{\text{real}} + \lambda_{\text{pseudo}}(t)\mathcal{L}_{\text{pseudo}} + \lambda_{\text{V2G}}\mathcal{L}_{\text{V2G}}, \tag{8}$$

where $\lambda_{\text{pseudo}}(t)$ is annealed over time and λ_{V2G} controls the V2G term.

This yields an implicit bidirectional co-refinement loop: restored videos provide pseudo-view and teacher-flow supervision for 3DGS optimization, while the optimized 3DGS scene supplies more accurate flow guidance for subsequent video restoration.

4 Experiments

4.1 Experimental Setup

Training Data and Evaluation Benchmarks. We sample 800 scenes from DL3DV-10K (Ling et al., 2024). For each scene, we optimize 3DGS from 3, 6, or 9 views for 7,000 iterations to obtain imperfect reconstructions, then render two non-overlapping 49-frame clips and pair them with

aligned real videos for fine-tuning. Real endpoint frames serve as references, while rendered clips are restoration inputs. DINOv2 features (Oquab et al., 2023) provide appearance cues. Adjacent-frame flow is estimated by a frozen WAFT model (Wang & Deng, 2026) and encoded by the FloVD flow encoder (Jin et al., 2025), whose reprojection-flow design suits our rendered-view conditions.

We evaluate on 18 DL3DV-Benchmark scenes disjoint from training (Ling et al., 2024), 9 Mip-NeRF 360 scenes (Barron et al., 2022), 6 Tanks and Temples (T&T) scenes (Knapitsch et al., 2017), and 16 CO3D scenes (Reizenstein et al., 2021), using PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018).

Implementation Details. We use the pretrained CogVideoX-5B-I2V (Yin et al., 2025; Yang et al., 2025) as the restoration backbone and fine-tune it on 49-frame clips at 480×720. Training lasts 10,000 steps. We first optimize the conditioning modules for 2,000 steps with the diffusion backbone frozen, and then unfreeze the Transformer self-attention and cross-normalization layers for the remaining 8,000 steps. We use AdamW (Loshchilov & Hutter, 2017) with BF16 precision and an effective batch size of 8. Learning rates are 2×10^{-5} for other trainable parameters and 1×10^{-4} for newly introduced scaling and gating parameters. Training is conducted on two NVIDIA RTX PRO 6000 Blackwell GPUs.

4.2 Comparison with State-of-the-Art Methods

Quantitative Comparison. Tables 1 and 2 summarize results on four benchmarks. On Mip-NeRF 360, our method achieves the best PSNR and SSIM for all input-view settings, improving PSNR over the strongest baselines by 0.43, 0.34, and 0.38 dB with 3, 6, and 9 views, respectively. On Tanks and Temples, DL3DV-Benchmark, and CO3D, it ranks first in PSNR across all nine settings with an average gain of 0.71 dB, while achieving the best SSIM and LPIPS in eight of nine settings each. These results demonstrate consistent gains across datasets and sparsity levels.

Table 1: Rendering-quality comparison for sparse-view reconstruction on Mip-NeRF 360 under 3-, 6-, and 9-view settings. * denotes results reproduced from ReconFusion, GenFusion, or GSFixer.

Method	Source	PSNR↑			SSIM↑			LPIPS↓		
		3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view
ZeroNVS* (Sargent et al., 2024)	CVPR'24	14.44	15.51	15.99	0.316	0.337	0.350	0.680	0.663	0.655
ReconFusion* (Wu et al., 2024)	CVPR'24	15.50	16.93	18.19	0.358	0.401	0.432	0.585	0.544	0.511
3DGS* (Kerbl et al., 2023)	SIGGRAPH'23	13.06	14.96	16.79	0.251	0.355	0.447	0.576	0.505	0.446
2DGS* (Huang et al., 2024)	SIGGRAPH'24	13.07	15.02	16.67	0.243	0.338	0.423	0.580	0.506	0.449
FSGS* (Zhu et al., 2024)	ECCV'24	14.17	16.12	17.94	0.318	0.415	0.492	0.578	0.517	0.468
ViewCrafter (Yu et al., 2024)	TPAMI'25	13.33	15.21	16.87	0.281	0.362	0.436	0.586	0.503	0.442
GenFusion (Wu et al., 2025b)	CVPR'25	15.05	16.80	18.15	0.357	0.427	0.489	0.577	0.494	0.442
Difix3D+* (Wu et al., 2025a)	CVPR'25 Oral	13.92	15.94	17.54	0.298	0.382	0.452	0.578	**0.468**	**0.391**
GSFixer (Yin et al., 2025)	ICML'26	15.65	17.36	18.68	0.370	0.430	0.485	0.559	0.475	0.419
Bi-FlowGS (Ours)	-	**16.08**	**17.70**	**19.06**	**0.388**	**0.446**	**0.501**	**0.541**	**0.468**	0.413

Table 2: Rendering-quality comparison for sparse-view reconstruction on Tanks and Temples, DL3DV-Benchmark, and CO3D under sparse-view settings.

Method	Source	PSNR↑			SSIM↑			LPIPS↓		
T&T		4-view	6-view	9-view	4-view	6-view	9-view	4-view	6-view	9-view
ViewCrafter (Yu et al., 2024)	TPAMI'25	12.38	14.00	15.83	0.452	0.496	0.542	0.533	0.475	0.420
GenFusion (Wu et al., 2025b)	CVPR'25	13.00	14.56	16.10	0.497	0.543	**0.587**	0.531	0.478	0.439
Difix3D+ (Wu et al., 2025a)	CVPR'25 Oral	13.44	14.65	15.62	0.451	0.473	0.492	**0.481**	0.446	0.419
GSFixer (Yin et al., 2025)	ICML'26	13.97	15.47	16.62	0.498	0.528	0.561	0.503	0.451	0.405
Bi-FlowGS (Ours)	-	**14.37**	**15.93**	**16.93**	**0.525**	**0.555**	0.585	0.486	**0.437**	**0.398**
DL3DV-Benchmark		**3-view**	**6-view**	**9-view**	**3-view**	**6-view**	**9-view**	**3-view**	**6-view**	**9-view**
ViewCrafter (Yu et al., 2024)	TPAMI'25	12.79	15.97	17.90	0.385	0.508	0.583	0.549	0.437	0.370
GenFusion (Wu et al., 2025b)	CVPR'25	14.52	17.20	18.82	0.491	0.577	0.634	0.516	0.412	0.357
Difix3D+ (Wu et al., 2025a)	CVPR'25 Oral	14.17	16.70	18.30	0.429	0.528	0.589	0.492	0.389	0.336
GSFixer (Yin et al., 2025)	ICML'26	15.02	17.25	18.62	0.491	0.562	0.615	0.502	0.403	0.347
Bi-FlowGS (Ours)	-	**15.72**	**17.82**	**19.15**	**0.525**	**0.594**	**0.642**	**0.471**	**0.377**	**0.327**
CO3D		**3-view**	**6-view**	**9-view**	**3-view**	**6-view**	**9-view**	**3-view**	**6-view**	**9-view**
ViewCrafter (Yu et al., 2024)	TPAMI'25	13.97	17.30	19.35	0.480	0.576	0.636	0.543	0.459	0.404
GenFusion (Wu et al., 2025b)	CVPR'25	15.24	17.39	19.45	0.590	0.616	0.660	0.512	0.470	0.420
Difix3D+ (Wu et al., 2025a)	CVPR'25 Oral	14.46	17.05	18.53	0.487	0.558	0.602	0.545	0.470	0.430
GSFixer (Yin et al., 2025)	ICML'26	15.04	17.74	18.97	0.571	0.605	0.634	0.509	0.458	0.422
Bi-FlowGS (Ours)	-	**15.92**	**19.33**	**20.78**	**0.595**	**0.650**	**0.682**	**0.474**	**0.398**	**0.359**

Qualitative Comparison. Figures 3 and 4 compare rendering quality and geometric consistency. To expose *Geometry Cheating*, we back-project shared COLMAP tracks (Schonberger & Frahm, 2016) using each method's rendered depth and reproject them into held-out views. Several baselines produce plausible renderings despite larger reprojection errors, revealing discrepancies between appearance fidelity and underlying geometry. Across Tanks and Temples, CO3D, and DL3DV-Benchmark, our method better preserves the red railing, sign text and graphics, chair structures, and storefront details highlighted by the yellow insets.

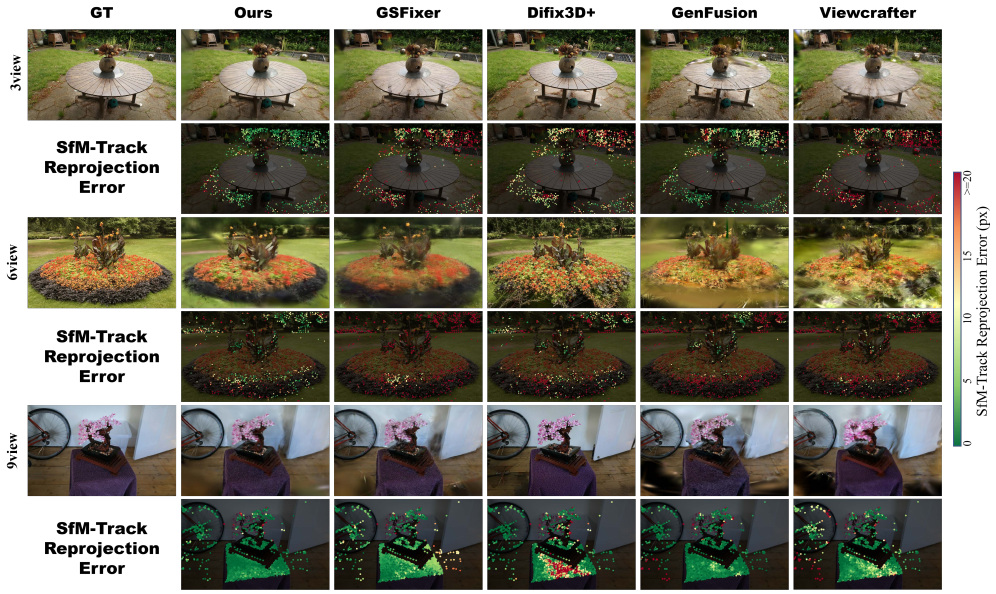


Figure 3: Rendering quality and geometric consistency on Mip-NeRF 360. Novel-view renderings are paired with SfM-track reprojection error maps. Lower reprojection error indicates better geometric consistency.



Figure 4: Qualitative comparison of reconstruction renderings across sparse-view benchmarks. Results are shown on Tanks and Temples, CO3D, and DL3DV-Benchmark.

4.3 Ablation Studies

Plug-and-Play Generality of V2G. We integrate V2G into GenFusion, ViewCrafter, and GSFixer under identical training conditions for each baseline and its V2G variant. Since Mip-NeRF 360 (Barron et al., 2022) lacks dense depth ground truth, geometry is evaluated against COLMAP/SfM sparse depths (Schonberger & Frahm, 2016) on identical robust tracks using standard depth metrics (Eigen et al., 2014). As shown in Table 3, V2G consistently improves all three frameworks, yielding

10.70%–47.77% relative geometry gains. Our full model further achieves 48.01%, 51.49%, and 50.78% gains for 3, 6, and 9 views, with PSNR gains of 0.26, 0.35, and 0.29 dB with an additional cost of only 21.71–26.51 ms/step. Figure 5 further confirms the visual gains, demonstrating V2G generality and effective bidirectional co-refinement.

Table 3: V2G ablation on Mip-NeRF 360 (3/6/9 views). Geo. Avg is the mean relative gain over geometry metrics; inference time (ms/step) is measured on one RTX A6000 for all methods.

Method	PSNR↑	SSIM↑	LPIPS↓	AbsRel↓	SqRel↓	RMSE↓	RMSE-log↓	MedianRel↓	δ_1 ↑	δ_2 ↑	δ_3 ↑	Geo. Avg↑	Time↓
3-view													
GenFusion	15.05	0.357	0.577	0.0322	0.0079	0.1030	0.0903	0.0118	0.9689	0.9906	0.9962	–	27.39
GenFusion+V2G	15.20	0.360	0.574	0.0232	0.0068	0.0983	0.0855	0.0080	0.9839	0.9923	0.9959	+10.70%	37.58
ViewCrafter	13.33	0.281	0.586	0.1311	0.2636	0.6804	0.2381	0.0575	0.8125	0.9274	0.9709	–	27.03
ViewCrafter+V2G	13.37	0.285	0.583	0.0825	0.0525	0.3082	0.1948	0.0240	0.8804	0.9477	0.9806	+32.48%	42.68
GSFixer	15.65	0.370	0.559	0.1135	0.0824	0.3847	0.2328	0.0538	0.8407	0.9361	0.9667	–	27.38
GSFixer+V2G	15.77	0.371	0.545	0.0464	0.0116	0.1239	0.1276	0.0136	0.9376	0.9760	0.9936	+43.91%	50.55
Ours w/o V2G	15.82	0.374	0.555	0.1098	0.0595	0.3156	0.2239	0.0530	0.8508	0.9333	0.9689	–	26.40
Ours	**16.08**	**0.388**	**0.541**	0.0310	0.0080	0.1024	0.1008	0.0097	0.9647	0.9870	0.9928	+48.01%	48.11
6-view													
GenFusion	16.80	0.427	0.494	0.0301	0.0100	0.1064	0.1240	0.0070	0.9603	0.9766	0.9843	–	27.96
GenFusion+V2G	16.94	0.431	0.494	0.0191	0.0065	0.0912	0.0890	0.0054	0.9799	0.9820	0.9974	+17.39%	38.43
ViewCrafter	15.21	0.362	0.503	0.0801	0.0813	0.3929	0.1749	0.0286	0.9046	0.9681	0.9850	–	44.14
ViewCrafter+V2G	15.34	0.369	0.498	0.0326	0.0117	0.1483	0.0987	0.0078	0.9629	0.9878	0.9958	+41.64%	81.47
GSFixer	17.36	0.430	0.475	0.0851	0.0388	0.2555	0.1730	0.0376	0.8828	0.9654	0.9888	–	32.90
GSFixer+V2G	17.43	0.432	0.468	0.0318	0.0078	0.1069	0.1016	0.0090	0.9686	0.9933	0.9946	+41.42%	60.94
Ours w/o V2G	17.35	0.428	0.473	0.0831	0.0500	0.2986	0.1692	0.0361	0.8997	0.9652	0.9869	–	27.46
Ours	**17.70**	**0.446**	**0.468**	0.0182	0.0037	0.0738	0.0516	0.0066	0.9895	0.9968	0.9995	+51.49%	52.21
9-view													
GenFusion	18.15	0.489	0.442	0.0158	0.0044	0.0740	0.0679	0.0054	0.9877	0.9932	0.9971	–	26.57
GenFusion+V2G	18.25	0.489	0.442	0.0115	0.0026	0.0613	0.0416	0.0047	0.9947	0.9980	0.9992	+17.18%	35.10
ViewCrafter	16.87	0.436	0.442	0.0651	0.0839	0.4374	0.1570	0.0218	0.9362	0.9754	0.9873	–	37.25
ViewCrafter+V2G	17.06	0.443	0.435	0.0278	0.0155	0.1806	0.1192	0.0059	0.9705	0.9854	0.9918	+37.47%	65.37
GSFixer	18.68	0.485	0.419	0.0662	0.0323	0.2528	0.1363	0.0284	0.9318	0.9802	0.9931	–	30.20
GSFixer+V2G	18.75	0.487	0.415	0.0174	0.0036	0.0836	0.0446	0.0067	0.9921	0.9984	0.9997	+47.77%	55.86
Ours w/o V2G	18.77	0.486	0.416	0.0762	0.0327	0.2450	0.1586	0.0342	0.9061	0.9702	0.9887	–	29.55
Ours	**19.06**	**0.501**	**0.413**	0.0158	0.0032	0.0708	0.0470	0.0062	0.9943	0.9974	0.9988	+50.78%	56.06

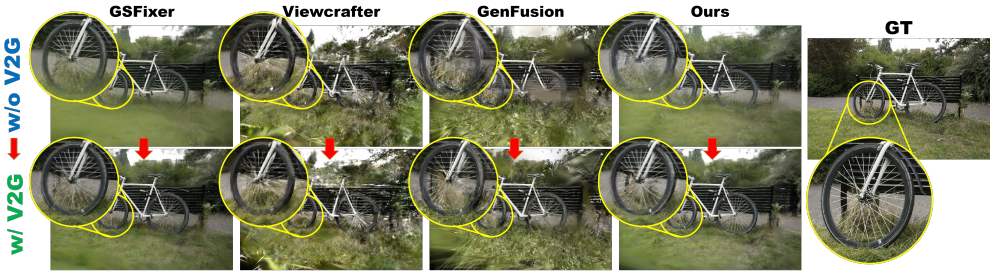


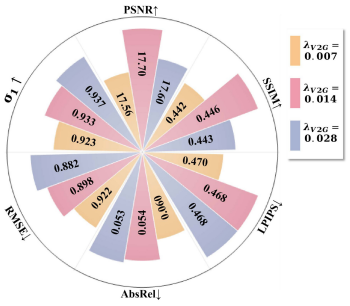
Figure 5: Plug-and-play V2G improves 3D reconstruction quality across different reconstruction frameworks. The top and bottom rows show reconstruction renderings without and with V2G, respectively.

Table 4: Component ablation and V2G-weight sensitivity on Mip-NeRF 360 with 6 views.

(a) Component ablation

DINOv2	FlowCond	V2G	PSNR↑	SSIM↑	LPIPS↓
✓	✓	✓	17.70	0.446	0.468
✓	✓	×	17.35	0.428	0.473
✓	×	✓	17.50	0.439	0.467
✓	×	×	17.28	0.438	0.479
×	✓	✓	17.52	0.439	0.477
×	✓	×	17.23	0.435	0.489
×	×	✓	17.47	0.440	0.473
×	×	×	17.18	0.434	0.483

(b) V2G weight sensitivity



Component Ablation and Co-Refinement Dynamics. We first ablate DINOv2, FlowCond, and V2G on Mip-NeRF 360 under the 6-view setting. Table 4(a) shows that removing V2G drops PSNR by 0.35 dB and SSIM by 0.018. With FlowCond and V2G, DINOv2 improves SSIM by 0.007 and reduces LPIPS by 0.009, indicating better texture preservation; with DINOv2 and V2G, FlowCond adds 0.20 dB PSNR, reflecting effective motion guidance. Their combination balances appearance fidelity and motion consistency. Figure 6 further shows sharper details and lower

SfM-track reprojection errors for the full model. Figure 7 further analyzes V2G alone and with G2V FlowCond. On Ignatius from Tanks and Temples, V2G improves both F-score and PSNR using ground-truth (GT) point clouds, validating geometry and rendering gains. On Bonsai, joint V2G and FlowCond achieves the lowest AbsRel and highest PSNR, demonstrating their mutual reinforcement and supporting implicit V2G–G2V co-refinement.

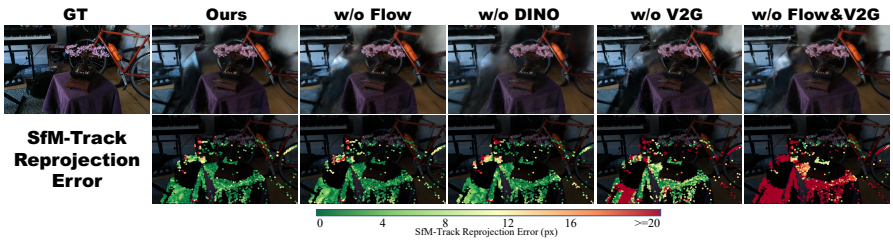


Figure 6: Visualization of component ablation. The top row shows rendered results, and the bottom row visualizes SfM-track reprojection errors.

Sensitivity to the V2G Loss Weight. We evaluate λ_{V2G} on Mip-NeRF 360 under the 6-view setting. As shown in Table 4(b), $\lambda_{\mathrm{V2G}} = 0.014$ achieves the best overall rendering quality, improving PSNR by 0.14 and 0.10 dB over 0.007 and 0.028, respectively. A larger weight slightly improves geometry but degrades PSNR and SSIM. We therefore use $\lambda_{\mathrm{V2G}} = 0.014$ in all other experiments.

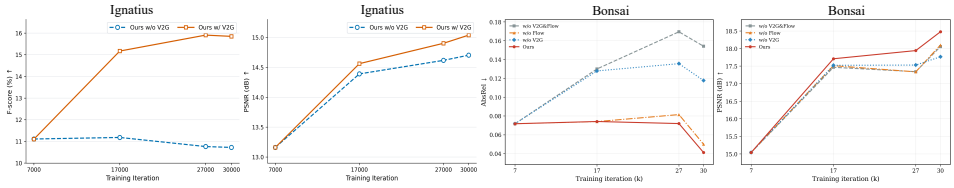
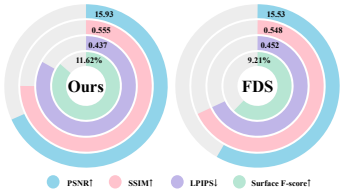


Figure 7: Ablation training dynamics. *Left:* PSNR and Surface F-score with/without V2G on Ignatius (Tanks and Temples). *Right:* AbsRel and PSNR for V2G/FlowCond ablations on Bonsai (Mip-NeRF 360).

Comparison with FDS. We compare V2G with FDS (Flow Distillation Sampling (Chen et al., 2025)), a representative geometry-regularization method using observed-view correspondences, on all six Tanks and Temples scenes. Surface F-score is reported on the four scenes with ground-truth point clouds. Under identical settings, Figure 8(a) shows that V2G achieves better rendering metrics and Surface F-score, while Figure 8(b) shows stronger PSNR and F-score throughout training on Ignatius. This improvement reflects their different supervision sources: FDS relies on observed-view correspondences, whereas V2G distills motion and temporal correspondence priors from restored videos into Gaussian geometry, yielding better geometry and rendering quality.

(a) Rendering and geometry metrics



(b) Training dynamics

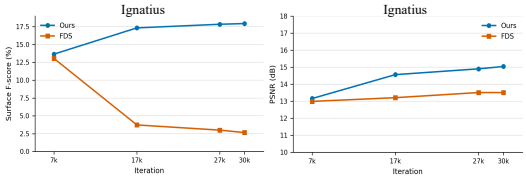


Figure 8: Comparison with FDS on Tanks and Temples under 6 views. Rendering metrics are averaged over 6 scenes, and Surface F-score over 4 scenes with ground-truth point clouds.

5 Conclusion

Bi-FlowGS couples generative view completion and sparse-view 3DGS geometry optimization through bidirectional optical-flow feedback. V2G distills restored-video correspondences into geometry, alleviating *Geometry Cheating*, while G2V uses geometry-derived cues to guide restoration. Experiments show improved rendering and geometric consistency, with V2G generalizing across reconstruction frameworks.

REPRODUCIBILITY STATEMENT

Sections 3.1–3.3 specify the mathematical formulation of Bi-FlowGS, including differentiable geometry-induced flow, reliability-aware V2G supervision, FlowCond, and the bidirectional co-refinement objective. Section 4.1 documents the training data, evaluation benchmarks, sparse-view settings, optimization schedule, precision, batch size, learning rates, and compute resources. Appendix A provides the G2V architecture, conditioning inputs, training and inference configurations, and dataset-specific sampling trajectories. Appendix B details V2G frame-pair sampling, teacher-flow filtering, loss settings, numerical stabilization, optimization, and geometry-evaluation definitions, while Appendix C specifies the controlled comparison protocol. Additional qualitative results, per-scene results, and controlled ablations are reported in the remaining appendix sections. All datasets, pretrained backbones, and external components used in our experiments are identified by name and citation in the paper. Our code is provided with the anonymous supplementary material and includes the complete implementation framework. We will release the full codebase, model weights, and related documentation subsequently.

AI USE STATEMENT

In this work, generative AI tools were used for manuscript writing and polishing, including readability editing and LaTeX formatting. All AI-assisted text and formatting were reviewed by the authors, who take responsibility for the final content.

REFERENCES

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5460–5469. IEEE, 2022.

David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19457–19467. IEEE, 2024.

Lin-Zhuo Chen, Kangjie Liu, Youtian Lin, Zhihao Li, Siyu Zhu, Xun Cao, and Yao Yao. Flow distillation sampling: Regularizing 3d gaussians with pre-trained matching priors. In *International Conference on Learning Representations*, volume 2025, pp. 39372–39384, 2025.

Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European conference on computer vision*, pp. 370–386. Springer, 2024.

Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pp. 303–312, 1996.

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.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for video diffusion models. In *International Conference on Learning Representations*, volume 2025, pp. 100433–100464, 2025.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Kai Hong, Yue Ming, Chuanchen Luo, Jiangwan Zhou, and Haotian Guo. Flowgs: End-to-end correspondence-guided 3d gaussian splatting from sparse unposed images. *Neurocomputing*, pp. 132104, 2025.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pp. 1–11, 2024.

Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5865–5874. IEEE, 2021.

Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *ACM Transactions on Graphics (TOG)*, 44(6):1–16, 2025.

Wonjoon Jin, Qi Dai, Chong Luo, Seung-Hwan Baek, and Sunghyun Cho. Flovd: Optical flow meets video diffusion model for enhanced camera-controlled video synthesis. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2040–2049. IEEE, 2025.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

Jeong-gi Kwak, Erqun Dong, Yuhe Jin, Hanseok Ko, Shweta Mahajan, and Kwang Moo Yi. Vivid-1-to-3: Novel view synthesis with video diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6775–6785. IEEE, 2024.

Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20775–20785. IEEE, 2024.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PmLR, 2023.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22160–22169. IEEE, 2024.

Fangfu Liu, Wenqiang Sun, Hanyang Wang, Yikai Wang, Haowen Sun, Junliang Ye, Jun Zhang, and Yueqi Duan. Reconx: Reconstruct any scene from sparse views with video diffusion model. *IEEE Transactions on Image Processing*, 2026a.

Xinhui Liu, Can Wang, Wei Jiang, Wei Wang, and Dong Xu. Gaussvid: Sparse-view gaussian splatting with 3d-aware video diffusion priors. *arXiv preprint arXiv:2608.21849*, 2026b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Junfeng Ni, Yixin Chen, Zhifei Yang, Yu Liu, Ruijie Lu, Song-Chun Zhu, and Siyuan Huang. G4splat: Geometry-guided gaussian splatting with generative prior. In *International Conference on Learning Representations*, volume 2026, pp. 138924–138951, 2026.

Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5470–5480. IEEE, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Avinash Paliwal, Xilong Zhou, Wei Ye, Jinhui Xiong, Rakesh Ranjan, and Nima Khademi Kalantari. Ri3d: Few-shot gaussian splatting with repair and inpainting diffusion priors. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25094–25103. IEEE, 2025.

Hyunwoo Park, Gun Ryu, and Wonjun Kim. Dropgaussian: Structural regularization for sparse-view gaussian splatting. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21600–21609. IEEE, 2025.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10881–10891. IEEE, 2021.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, et al. Zeronvs: Zero-shot 360-degree view synthesis from a single image. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9420–9429. IEEE, 2024.

Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4104–4113, 2016.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Jimin Tang, Wenyuan Zhang, Junsheng Zhou, Zian Huang, Kanle Shi, Shenkun Xu, Yu-Shen Liu, and Zhizhong Han. Vidsplat: Gaussian splatting reconstruction with geometry-guided video diffusion priors. *arXiv preprint arXiv:2605.11424*, 2026.

Atakan Topaloğlu, Kunyi Li, Michael Niemeyer, Nassir Navab, A Murat Tekalp, and Federico Tombari. Oraclegs: Grounding generative priors for sparse-view gaussian splatting. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 77–87. IEEE, 2026.

Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9031–9042. IEEE, 2023.

Xinliang Wang, Yifeng Shi, and Zhenyu Wu. Artifactworld: Scaling 3d gaussian splatting artifact restoration via video generation models. *arXiv preprint arXiv:2604.12251*, 2026.

Yihan Wang and Jia Deng. Waft: Warping-alone field transforms for optical flow. In *International Conference on Learning Representations*, volume 2026, pp. 157389–157402, 2026.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Jay Zhangjie Wu, Yuxuan Zhang, Haithem Turki, Xuanchi Ren, Jun Gao, Mike Zheng Shou, Sanja Fidler, Zan Gojcic, and Huan Ling. Difix3d+: Improving 3d reconstructions with single-step diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26024–26035. IEEE, 2025a.

Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor Verbin, Jonathan T Barron, Ben Poole, et al. Reconfusion: 3d reconstruction with diffusion priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 21551–21561, 2024.

Sibo Wu, Congrong Xu, Binbin Huang, Andreas Geiger, and Anpei Chen. Genfusion: Closing the loop between reconstruction and generation via videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6078–6088. IEEE, 2025b.

Haolin Xiong, Sairisheek Muttukuru, Hanyuan Xiao, Rishi Upadhyay, Pradyumna Chari, Yajie Zhao, and Achuta Kadambi. Sparsegs: Sparse view synthesis using 3d gaussian splatting. In *2025 International Conference on 3D Vision (3DV)*, pp. 1032–1041. IEEE, 2025.

Jiale Xu, Shenghua Gao, and Ying Shan. Freesplatter: Pose-free gaussian splatting for sparse-view 3d reconstruction. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25442–25452. IEEE, 2025.

Jiawei Yang, Marco Pavone, and Yue Wang. Freenerf: Improving few-shot neural rendering with free frequency regularization. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8254–8263. IEEE, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Ruihong Yin, Vladimir Yugay, Yue Li, Sezer Karaoglu, and Theo Gevers. Fewviewgs: Gaussian splatting with few view matching and multi-stage training. *Advances in Neural Information Processing Systems*, 37:127204–127225, 2024.

Xingyilang Yin, Qi Zhang, Jiahao Chang, Ying Feng, Qingnan Fan, Xi Yang, Chi-Man Pun, Huaqi Zhang, and Xiaodong Cun. Gsfixer: Improving 3d gaussian splatting with reference-guided video diffusion priors. *arXiv preprint arXiv:2508.09667*, 2025.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024.

Jiawei Zhang, Jiahe Li, Xiaohan Yu, Lei Huang, Lin Gu, Jin Zheng, and Xiao Bai. Cor-gs: sparse-view 3d gaussian splatting via co-regularization. In *European conference on computer vision*, pp. 335–352. Springer, 2024.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Yingji Zhong, Zhihao Li, Dave Zhenyu Chen, Lanqing Hong, and Dan Xu. Taming video diffusion prior with scene-grounding guidance for 3d gaussian splatting from sparse inputs. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6133–6143. IEEE, 2025.

Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *2023 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 12588–12597. IEEE, 2023.

Liyuan Zhu, Manjunath Narayana, Michal Stary, Will Hutchcroft, Gordon Wetzstein, and Iro Armeni. Gaussfusion: Improving 3d reconstruction in the wild with a geometry-informed video generator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15432–15442, 2026.

Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. In *European conference on computer vision*, pp. 145–163. Springer, 2024.

A G2V IMPLEMENTATION DETAILS

Backbone and conditioning inputs. We build G2V on CogVideoX-5B-I2V (Yang et al., 2025). The video diffusion Transformer processes the channel-wise concatenation of the noisy target latent and the latent of an artifact-corrupted 3DGS rendering. Each clip has 49 frames at 480×720. Separate DINOv2 and FlowCond pathways provide endpoint appearance and rendered-sequence motion cues, respectively.

FlowCond. The four-channel condition comprises a rendered–monocular depth-consistency mask, a monocular-depth gradient map, and normalized horizontal and vertical adjacent-frame flow. We estimate monocular depth and flow with frozen Depth Anything 3 (DA3MONO-LARGE) (Lin et al., 2025) and WAFT (Wang & Deng, 2026), respectively. A FlowEncoder initialized from FloVD (Jin et al., 2025) extracts multi-scale features, which are spatially aligned, channel-projected, and injected as residual biases before Transformer self-attention.

Flow-guided reference feature injection. A frozen DINOv2 ViT-L/14 (Oquab et al., 2023) extracts patch features from the endpoint frames, and a learnable layer projects them into the Transformer hidden space. Cross-attention between self-attention and the feed-forward layer injects these features through two branches: global retrieval attends to all endpoint tokens, whereas local retrieval uses rendered-view flow to aggregate nearby correspondences and interpolates endpoint contributions by frame position. The branches share cross-attention projections, and depth consistency modulates their injection strength.

Diffusion objective and training. Given a clean video latent $\mathbf{z}_0$, Gaussian noise $\boldsymbol{\epsilon}$, and diffusion timestep t, we use the standard v-prediction objective

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{z}_0,\boldsymbol{\epsilon},t}\left[\|v_\theta(\mathbf{z}_t, t, \mathcal{C}) - v_t\|_2^2\right], \tag{9}$$

where $\mathcal{C}$ comprises the rendered-video, FlowCond, DINOv2, and text conditions. Here, velocity is the diffusion parameterization (Salimans & Ho, 2022), not optical flow or physical motion. We use no additional pixel-space, perceptual, or flow loss.

We train on 800 DL3DV-10K scenes (Ling et al., 2024) for 10,000 steps. For the first 2,000, only the FlowEncoder, DINOv2 projector, multi-scale flow projectors, depth-gradient gates, and conditioning fusion parameters are optimized. For the remaining 8,000, Transformer self-attention and cross-normalization layers are also unfrozen. The DINOv2 backbone, VAE, shared cross-attention projections, and other Transformer modules remain frozen.

We use AdamW (Loshchilov & Hutter, 2017), BF16, an effective batch size of 8, and a 10-step linear warmup followed by constant learning rates: 2×10^{-5} for standard trainable modules and 1×10^{-4} for new scaling and gating parameters.

Inference. We use 50-step DDIM sampling (Song et al., 2020) with classifier-free guidance (Ho & Salimans, 2022) at scale 6.0. The 3DGS rendering is the control video, its endpoint frames are references, and FlowCond is injected throughout denoising. BLIP-2 (Li et al., 2023) generates prompts from the first reference frame, followed by a fixed quality-oriented suffix. This configuration is fixed across experiments and uses 26,405 MiB (25.8 GiB) peak memory on one NVIDIA RTX A6000 (48 GB), without sequential CPU offloading.

A.1 VIDEO SAMPLING TRAJECTORIES

We adapt sampling trajectories to each dataset's scene scale and camera distribution.

Mip-NeRF 360 and Tanks and Temples. Sparse input cameras define a data-adaptive elliptical trajectory: the center minimizes distances to their optical axes, and their spatial distribution determines the horizontal radii. A small height variation expands coverage across restoration rounds and decays for more conservative refinement.

CO3D and DL3DV-Benchmark. We construct bounded circular trajectories from sparse poses. For CO3D, PCA of camera centers estimates the dominant viewing plane, and observed coverage bounds the radius. For DL3DV-Benchmark, the camera distribution determines the center and plane,

while robust statistics set the radius and height. Both trajectories remain within observed coverage, with phase varied across rounds.

B V2G Implementation Details

Frame-pair sampling and teacher reliability. Each V2G update samples two restored-frame pairs with temporal strides randomly selected from one and two. Frozen WAFT (Wang & Deng, 2026) estimates bidirectional flow; forward flow supervises the model, and reverse flow is used only for consistency checking. We reject invalid warps, flow magnitudes above 96 pixels, and cycle errors above $0.08\min(H, W)$. Teacher confidence depends on cycle error and flow magnitude and is thresholded at 0.4. All teacher quantities are detached.

Geometric validity and loss settings. Geometry-induced flow follows Sec. 3.1. Source rendered depth remains differentiable; target depth only masks invalid, out-of-bound, and occluded correspondences. Teacher and student flows are in pixels. We average Smooth-L_1 ($\beta = 1$) over both components, set $\gamma = 1.5$ and $\lambda_{\mathrm{V2G}} = 0.014$, and clamp the normalization denominator to 10^{-6}.

Optimization details. We linearly warm up the V2G weight over 100 valid updates and cap its weighted loss at 0.9 times the current base reconstruction loss. Restored clips, control videos, rendered depths, and FlowCond inputs are periodically regenerated from the latest Gaussian scene.

B.1 Sparse-Depth Geometry Metrics and Reprojection Visualization

Sparse-depth geometry metrics. Because Mip-NeRF 360 lacks dense geometry ground truth, we evaluate against sparse COLMAP/SfM points (Schonberger & Frahm, 2016). For each test view, we project the points into the image, use their camera-space depths as references, and bilinearly sample predicted depths from the rendering.

A point is valid if both depths are finite and positive and rendered opacity exceeds 0.05. For each input-view setting, we intersect valid points across methods and retain the nearest 85% by reference depth, excluding distant points with weak parallax and large depth outliers.

On the retained points, we compute (Eigen et al., 2014)

$$\text{AbsRel} = \frac{1}{N}\sum_i \frac{|d_i - \hat{d}_i|}{d_i}, \qquad \text{SqRel} = \frac{1}{N}\sum_i \frac{(d_i - \hat{d}_i)^2}{d_i},$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_i (d_i - \hat{d}_i)^2}, \qquad \text{RMSE-log} = \sqrt{\frac{1}{N}\sum_i (\log d_i - \log \hat{d}_i)^2},$$

$$\text{MedianRel} = \operatorname{median}\left(\frac{|d_i - \hat{d}_i|}{d_i}\right),$$

where d_i and $\hat{d}_i$ denote reference and predicted depths. We also report

$$\delta_k = \frac{1}{N}\sum_i \mathbb{1}\left(\max\left(\frac{\hat{d}_i}{d_i}, \frac{d_i}{\hat{d}_i}\right) < 1.25^k\right), \qquad k = 1, 2, 3.$$

Metrics pool retained points across scenes and test views. A displayed $\delta_k = 1.0000$ means that all retained points satisfy the threshold up to the reported precision.

Geo. Avg. **Geo. Avg.** is the mean relative improvement over all eight metrics: $(m_{\mathrm{base}} - m)/m_{\mathrm{base}}$ for lower-is-better metrics and $(m - m_{\mathrm{base}})/m_{\mathrm{base}}$ for δ_1, δ_2, and δ_3. It is computed from unrounded metric values.

SfM-track reprojection error visualization. We visualize cross-view reprojection error on fixed COLMAP/SfM tracks shared across methods. For points visible in paired source and target views,

we back-project each source pixel using rendered depth and reproject it into the target camera:

$$e(\mathbf{p}) = \|\hat{\mathbf{p}}_{j\to i} - \mathbf{p}_i\|_2 ,$$

where $\mathbf{p}_i$ is the target SfM observation and $\hat{\mathbf{p}}_{j\to i}$ its source-view reprojection. Lower is better; all methods use identical tracks and color scales.

C Comparison Protocol

We add V2G to GenFusion, ViewCrafter, and GSFixer without changing their pipelines or optimization settings. All methods use the same sparse views, poses, reconstructed-video trajectories, frame-pair poses, and test views. We adapt ViewCrafter to iterative 3DGS optimization using the given camera poses.

D Additional Qualitative Results

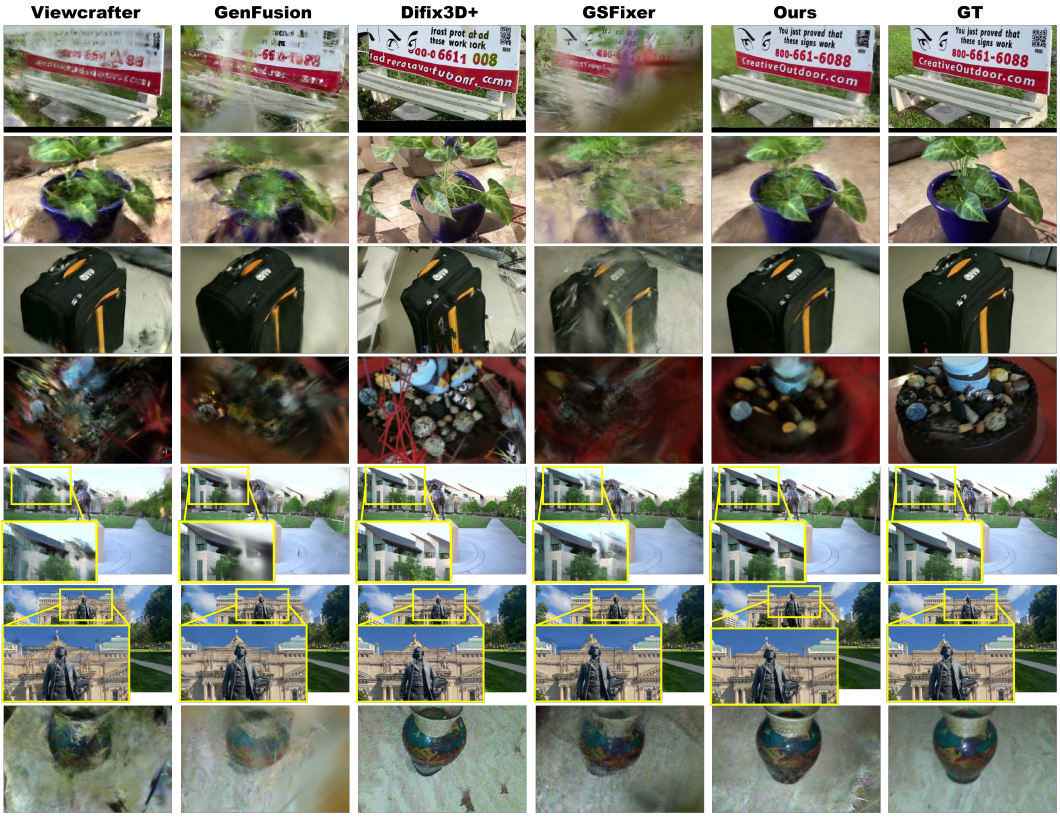


Figure 9: Additional reconstruction comparisons among ViewCrafter, GenFusion, Difix3D+, GSFixer, our method, and ground truth.

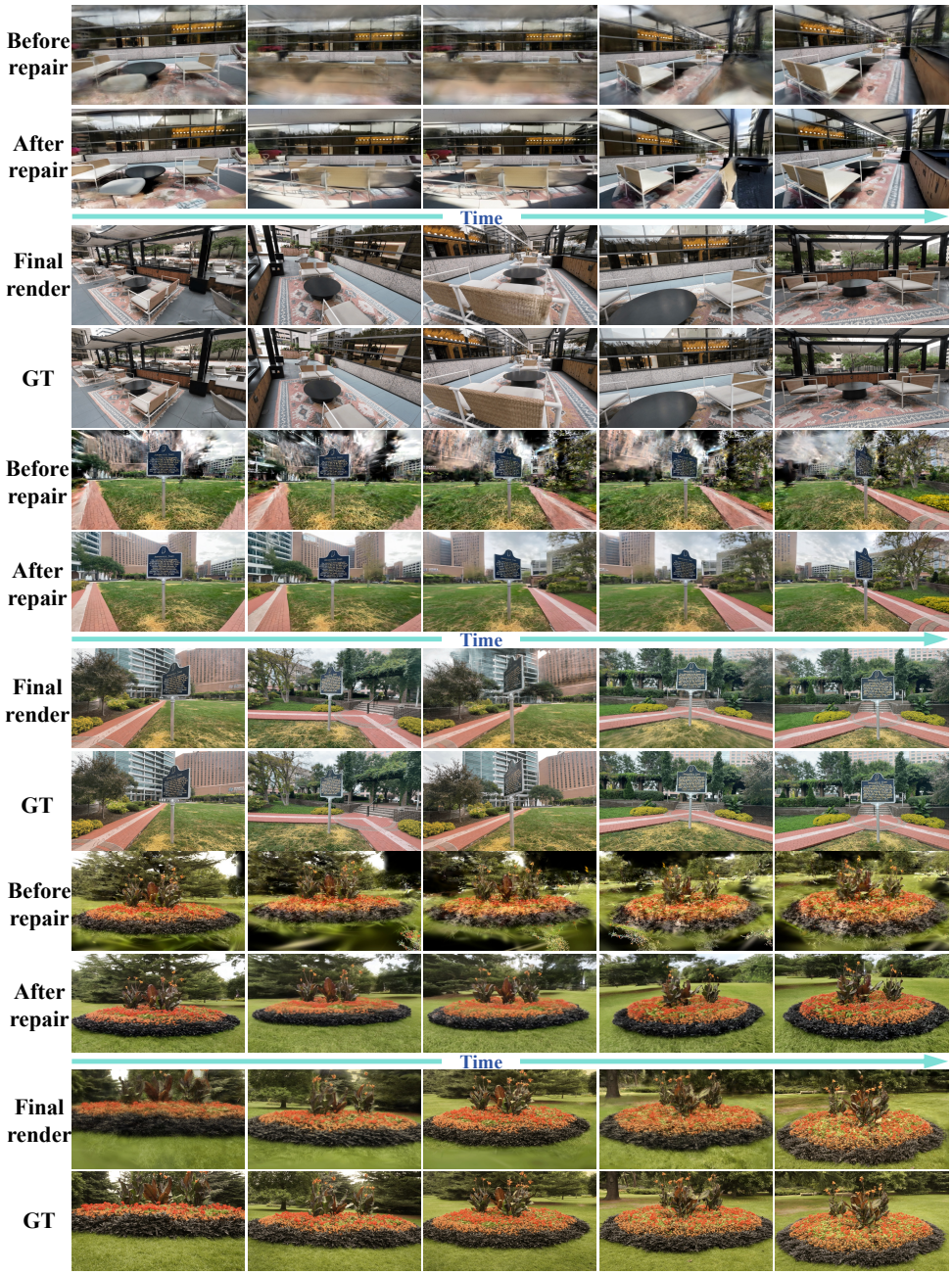


Figure 10: Additional results across datasets. Each row group shows the rendered video before and after restoration, the final reconstruction, and ground truth.

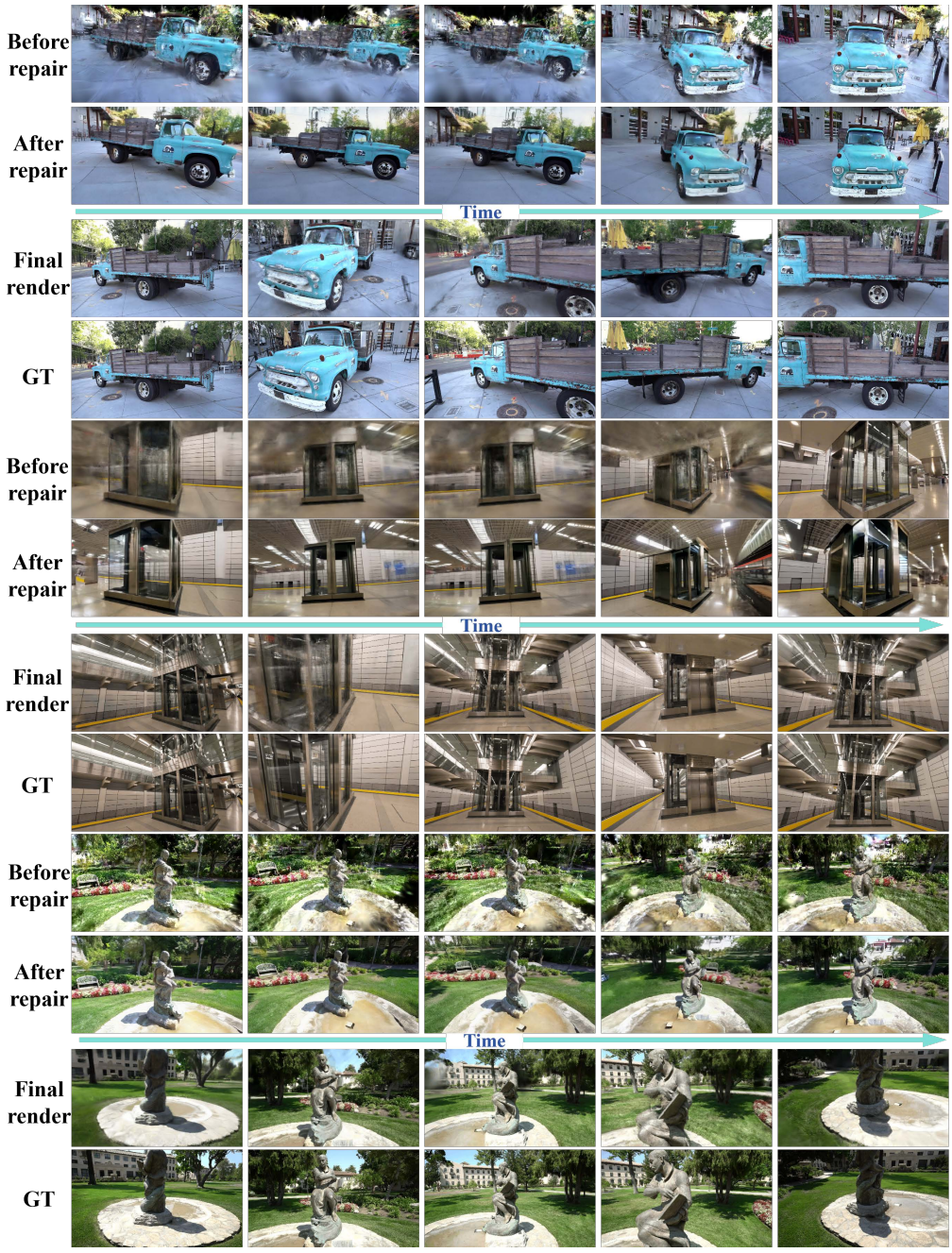


Figure 11: Additional results across datasets. Each row group shows the rendered video before and after restoration, the final reconstruction, and ground truth.



Figure 12: Additional results across datasets. Each row group shows the rendered video before and after restoration, the final reconstruction, and ground truth.

E Additional Ablations

E.1 Geometry F-score Metrics

We follow the Tanks and Temples (T&T) protocol (Knapitsch et al., 2017), including official cropping and alignment, three point-to-point ICP iterations, and voxel downsampling at $\tau/2$. For predicted and reference point sets $\hat{X}$ and X, bidirectional nearest-neighbor matching gives

$$P = \frac{1}{|\hat{X}|} \sum_{\hat{x} \in \hat{X}} \mathbf{1}\left[\min_{x \in X} \|\hat{x} - x\|_2 < \tau\right], \qquad R = \frac{1}{|X|} \sum_{x \in X} \mathbf{1}\left[\min_{\hat{x} \in \hat{X}} \|x - \hat{x}\|_2 < \tau\right].$$

$$F = \frac{2PR}{P+R}.$$

We use $\tau = 1\,\mathrm{cm}$ for Barn, $5\,\mathrm{mm}$ for Caterpillar, $3\,\mathrm{mm}$ for Ignatius, and $5\,\mathrm{mm}$ for Truck, and report $100F$.

Surface F-score. At each checkpoint, we render normalized COLMAP views, fuse valid depths into a TSDF (Curless & Levoy, 1996), and use its zero-crossing surface points as $\hat{X}$.

Official Gaussian-center F-score. Here, $\hat{X}$ comprises the reconstructed Gaussian centers, without rendering or TSDF fusion; we apply the same matching protocol.

E.2 V2G Ablation over Training

We compare the full model with its variant without $\mathcal{L}_{\mathrm{V2G}}$ at 7k, 17k, 27k, and 30k iterations on the four T&T scenes with ground-truth point clouds. Figures 13 and 14 show higher final Surface and Gaussian-center F-scores on every scene, indicating gains in both reconstructed surfaces and Gaussian-center geometry.

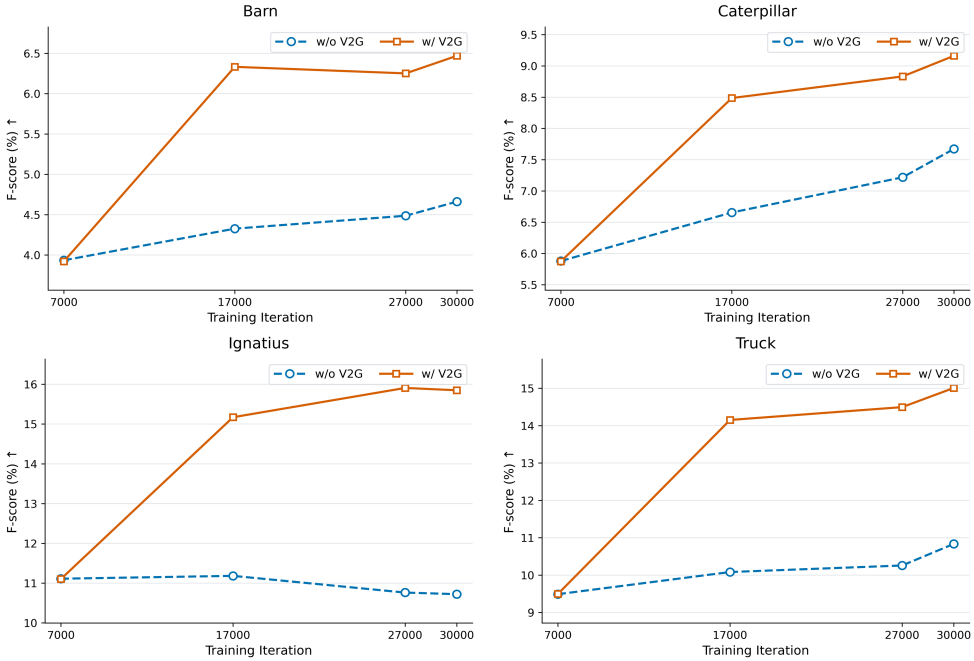


Figure 13: **V2G ablation over training: Surface F-score.** We compare our complete model with and without V2G at four training checkpoints on Barn, Caterpillar, Ignatius, and Truck.

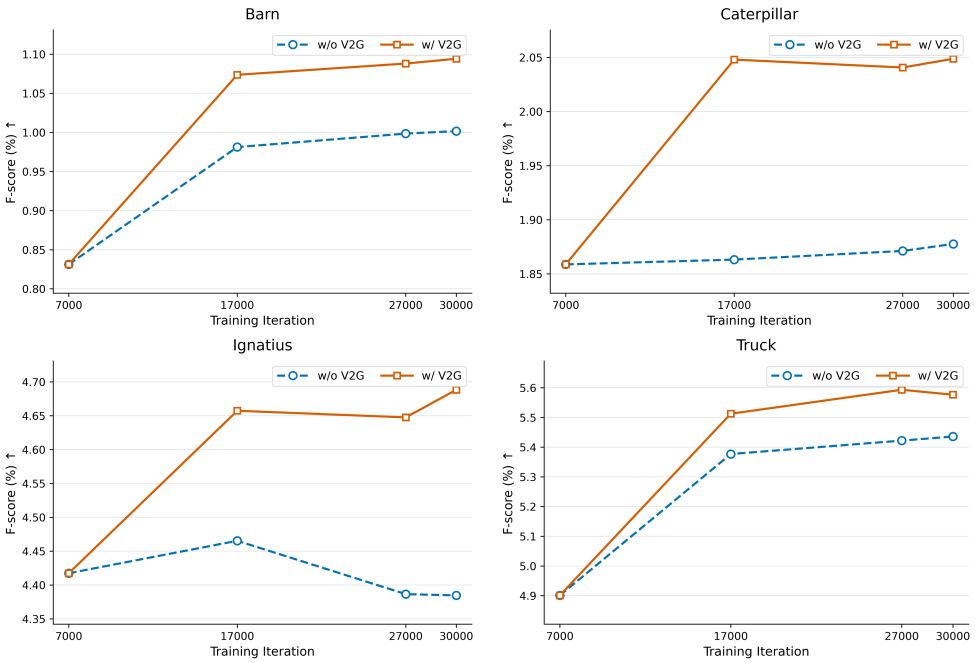


Figure 14: **V2G ablation over training: official Gaussian-center F-score.** We compare our complete model with and without V2G at four training checkpoints on Barn, Caterpillar, Ignatius, and Truck.

E.3 Gaussian-Center Gradient Ablation

We isolate center-directed V2G supervision by blocking only $\partial \mathcal{L}_{\text{V2G}}/\partial \boldsymbol{\mu}$; all other gradients and objectives are unchanged, so reconstruction losses still train $\boldsymbol{\mu}$. We evaluate the official Gaussian-center F-score at 7k, 17k, 27k, and 30k iterations. At 30k, full V2G raises the four-scene mean from 3.268 to 3.352 and improves all three rendering metrics (Table 5), supporting the benefit of direct center supervision.

Table 5: Ablation of V2G supervision to Gaussian centers. F-score is evaluated at 30k iterations from Gaussian centers using the official T&T protocol; rendering metrics are averaged over the same four scenes. The last two rows form the controlled center-gradient comparison.

Method	**Official Gaussian-center F-score (%) ↑**					**Rendering**		
	Barn	**Caterpillar**	**Ignatius**	**Truck**	**Mean**	**PSNR↑**	**SSIM↑**	**LPIPS↓**
Ours w/o V2G	1.002	1.878	4.385	5.436	3.175	15.48	0.496	0.474
Ours (V2G; stop-grad. on $\boldsymbol{\mu}$)	1.082	**2.086**	4.454	5.450	3.268	15.42	0.498	0.471
Ours	**1.094**	2.049	**4.688**	**5.577**	**3.352**	**15.64**	**0.502**	**0.468**

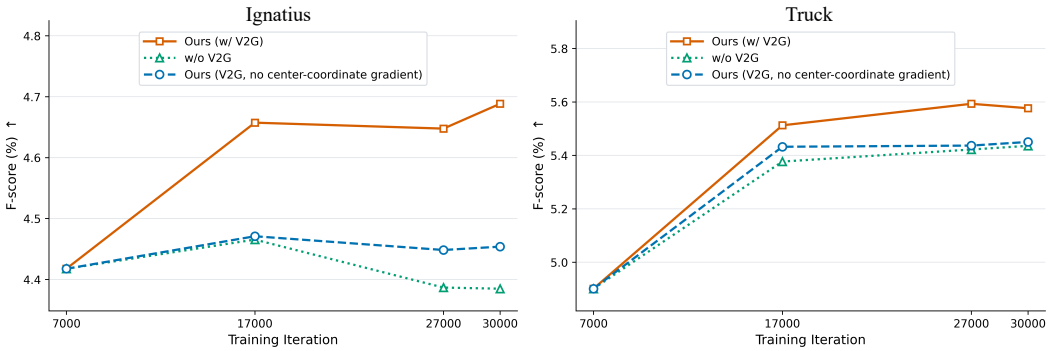


Figure 15: **V2G center-gradient ablation on T&T.** Official Gaussian-center F-score at four checkpoints on Ignatius and Truck. The stop-gradient variant blocks only $\partial \mathcal{L}_{\mathrm{V2G}}/\partial \boldsymbol{\mu}$; other objectives still train the centers.

E.4 Reliability-Mask Ablation

The reliability weight in Eq. (3) combines teacher-flow reliability $\mathbf{M}^T(\mathbf{C}^T)^\gamma$ with geometry reprojection mask $\mathbf{M}^G$. The former filters invalid warps, excessive motion, and large cycle errors and weights retained matches by confidence; the latter filters occluded or geometrically inconsistent matches using source and target depths. We ablate either term by setting it to one while retaining the finite-value, boundary, positive-depth, and projection-support checks needed to define flow residuals.

Table 6: Ablation of the V2G reliability factors. Rendering metrics are averaged over the four Tanks and Temples scenes with ground-truth point clouds.

Teacher-flow reliability	Geometry reprojection mask	PSNR↑	SSIM↑	LPIPS↓
✓	✓	**15.643**	**0.502**	**0.468**
×	✓	15.442	0.499	0.469
✓	×	15.454	0.498	0.470
×	×	15.480	0.499	**0.468**

The full design gives the best PSNR and SSIM and ties the best LPIPS. Removing teacher-flow reliability or geometry masking lowers mean PSNR by 0.201 and 0.189 dB, respectively, supporting their complementary roles.

F Additional Quantitative Results

We report per-scene rendering results and scene-wise averages for the complete Bi-FlowGS model across all benchmarks and input-view settings.

F.1 Mip-NeRF 360

Table 7: Per-scene rendering results on Mip-NeRF 360 under the 3-, 6-, and 9-view settings.

(a) 3 input views

Scene	ViewCrafter			GenFusion			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
bicycle	13.52	0.198	0.632	14.69	0.245	0.640	16.29	0.286	0.609	15.99	0.290	0.598
bonsai	11.63	0.346	0.592	14.47	0.432	0.538	13.52	0.434	0.554	14.79	0.466	0.522
counter	13.20	0.336	0.543	14.93	0.466	0.517	15.18	0.468	0.501	15.45	0.492	0.489
flowers	11.08	0.158	0.674	12.58	0.200	0.694	13.54	0.208	0.655	13.97	0.217	0.630
garden	14.32	0.248	0.559	15.95	0.286	0.575	17.60	0.327	0.537	18.13	0.347	0.532
kitchen	14.51	0.326	0.540	16.30	0.424	0.532	16.31	0.419	0.508	16.78	0.446	0.481
room	13.66	0.424	0.488	16.31	0.569	0.430	15.61	0.542	0.437	16.05	0.571	0.422
stump	15.54	0.249	0.602	16.43	0.288	0.622	17.94	0.327	0.606	17.92	0.325	0.592
treehill	12.49	0.248	0.646	13.81	0.303	0.643	14.87	0.318	0.628	15.64	0.340	0.602
Average	**13.33**	**0.281**	**0.586**	**15.05**	**0.357**	**0.577**	**15.65**	**0.370**	**0.559**	**16.08**	**0.388**	**0.541**

(b) 6 input views

Scene	ViewCrafter			GenFusion			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
bicycle	14.55	0.218	0.571	16.15	0.289	0.573	17.44	0.304	0.545	16.96	0.314	0.541
bonsai	15.24	0.487	0.472	16.76	0.551	0.432	17.40	0.555	0.436	18.48	0.586	0.420
counter	15.42	0.453	0.455	17.00	0.542	0.423	17.18	0.536	0.421	17.45	0.557	0.409
flowers	12.77	0.188	0.609	13.31	0.219	0.633	13.98	0.216	0.592	14.85	0.237	0.577
garden	16.75	0.370	0.443	18.59	0.392	0.452	19.03	0.403	0.429	19.15	0.424	0.430
kitchen	17.05	0.485	0.403	18.49	0.555	0.391	18.71	0.563	0.364	19.23	0.568	0.362
room	14.92	0.509	0.438	17.74	0.631	0.387	16.95	0.611	0.393	16.84	0.614	0.394
stump	16.37	0.286	0.560	17.33	0.308	0.572	18.95	0.337	0.543	18.95	0.342	0.540
treehill	13.81	0.264	0.574	15.78	0.352	0.581	16.59	0.346	0.549	17.37	0.374	0.536
Average	**15.21**	**0.362**	**0.503**	**16.80**	**0.427**	**0.494**	**17.36**	**0.430**	**0.475**	**17.70**	**0.446**	**0.468**

(c) 9 input views

Scene	ViewCrafter			GenFusion			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
bicycle	15.14	0.267	0.530	16.33	0.309	0.545	18.06	0.327	0.508	18.02	0.339	0.508
bonsai	17.82	0.607	0.376	18.94	0.658	0.342	19.37	0.657	0.356	20.01	0.672	0.343
counter	16.68	0.525	0.399	18.01	0.606	0.374	19.02	0.606	0.360	19.37	0.628	0.351
flowers	13.52	0.216	0.565	14.44	0.254	0.594	14.78	0.242	0.533	15.74	0.267	0.526
garden	18.54	0.447	0.378	19.85	0.471	0.396	20.16	0.480	0.364	20.38	0.493	0.369
kitchen	19.25	0.569	0.338	20.71	0.640	0.315	20.60	0.634	0.310	20.98	0.631	0.304
room	17.78	0.634	0.356	19.56	0.705	0.314	18.86	0.669	0.336	19.47	0.692	0.321
stump	18.28	0.345	0.507	19.03	0.372	0.526	19.75	0.382	0.496	19.69	0.392	0.494
treehill	14.85	0.319	0.530	16.46	0.384	0.569	17.54	0.365	0.509	17.90	0.398	0.501
Average	**16.87**	**0.436**	**0.442**	**18.15**	**0.489**	**0.442**	**18.68**	**0.485**	**0.419**	**19.06**	**0.501**	**0.413**

F.2 Plug-and-Play V2G Per-Scene Results

We evaluate plug-and-play V2G independently of the full Bi-FlowGS pipeline by augmenting GenFusion, ViewCrafter, and GSFixer without changing their reconstruction or optimization settings. Tables 8–10 report rendering and sparse-depth metrics on nine Mip-NeRF 360 scenes under 3, 6, and 9 input views. Rendering averages are scene means matching the main paper; geometry metrics pool shared COLMAP/SfM points because dense ground truth is unavailable.

Table 8: Per-scene rendering and sparse-depth geometry results for the plug-and-play V2G evaluation on Mip-NeRF 360 with 3 input views. V2G is applied to GenFusion, ViewCrafter, and GSFixer; "Ours" denotes the complete Bi-FlowGS model. The "#Pts" column gives the retained shared SfM points used for geometry evaluation.

Scene	Method	Rendering			Geometry								
		PSNR↑	SSIM↑	LPIPS↓	#Pts	AbsRel↓	SqRel↓	RMSE↓	RMSE-log↓	MedianRel↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
Bicycle	ViewCrafter+V2G	13.44	0.193	0.621	1318	0.1832	0.1103	0.4419	0.2917	0.1229	0.6419	0.8566	0.9651
	GenFusion+V2G	14.56	0.235	0.627	1318	0.0435	0.0113	0.1330	0.0865	0.0229	0.9712	0.9924	0.9992
	GSFixer+V2G	16.18	0.278	0.590	1318	0.1304	0.0676	0.3511	0.2596	0.0708	0.7671	0.8915	0.9598
	Ours	15.99	0.290	0.598	1318	0.0571	0.0138	0.1405	0.0908	0.0309	0.9704	0.9954	0.9992
Bonsai	ViewCrafter+V2G	11.94	0.340	0.586	16303	0.1179	0.0562	0.2442	0.2017	0.0505	0.8481	0.9489	0.9795
	GenFusion+V2G	14.48	0.434	0.525	16303	0.0164	0.0073	0.0879	0.0513	0.0085	0.9956	0.9975	0.9980
	GSFixer+V2G	13.71	0.432	0.549	16303	0.0500	0.0098	0.1046	0.1092	0.0201	0.9583	0.9836	0.9966
	Ours	14.79	0.466	0.522	16303	0.0193	0.0036	0.0606	0.0493	0.0104	0.9934	0.9967	0.9988
Counter	ViewCrafter+V2G	13.17	0.343	0.542	13082	0.0480	0.0157	0.1644	0.1290	0.0145	0.9502	0.9831	0.9895
	GenFusion+V2G	15.28	0.471	0.510	13082	0.0207	0.0088	0.1325	0.0938	0.0076	0.9842	0.9927	0.9945
	GSFixer+V2G	15.46	0.476	0.494	13082	0.0700	0.0195	0.1489	0.1929	0.0158	0.8886	0.9262	0.9725
	Ours	15.45	0.492	0.489	13082	0.0559	0.0139	0.1044	0.1736	0.0111	0.9079	0.9389	0.9710
Flowers	ViewCrafter+V2G	11.11	0.161	0.680	380	0.1032	0.0364	0.2411	0.2114	0.0522	0.8605	0.9500	0.9816
	GenFusion+V2G	12.93	0.207	0.705	380	0.0872	0.0441	0.2156	0.2001	0.0311	0.8974	0.9579	0.9816
	GSFixer+V2G	13.62	0.204	0.628	380	0.0866	0.0943	0.4186	0.1688	0.0315	0.9263	0.9579	0.9816
	Ours	13.97	0.217	0.630	380	0.1316	0.3086	0.7179	0.2309	0.0291	0.9289	0.9447	0.9658
Garden	ViewCrafter+V2G	14.25	0.251	0.557	11356	0.0853	0.0534	0.3309	0.1705	0.0319	0.8790	0.9595	0.9887
	GenFusion+V2G	16.18	0.292	0.568	11356	0.0321	0.0088	0.1035	0.0801	0.0125	0.9768	0.9881	0.9989
	GSFixer+V2G	17.58	0.325	0.527	11356	0.0295	0.0044	0.0913	0.0495	0.0162	0.9945	0.9989	0.9999
	Ours	18.13	0.347	0.532	11356	0.0228	0.0029	0.0743	0.0401	0.0127	0.9963	0.9996	0.9998
Kitchen	ViewCrafter+V2G	14.63	0.339	0.531	27846	0.0400	0.0070	0.0955	0.0943	0.0160	0.9688	0.9914	0.9974
	GenFusion+V2G	16.32	0.426	0.524	27846	0.0200	0.0071	0.0979	0.1153	0.0057	0.9814	0.9866	0.9900
	GSFixer+V2G	16.29	0.419	0.493	27846	0.0504	0.0126	0.1132	0.1505	0.0105	0.9061	0.9712	0.9939
	Ours	16.78	0.446	0.481	27846	0.0218	0.0055	0.0788	0.1100	0.0075	0.9792	0.9860	0.9884
Room	ViewCrafter+V2G	13.86	0.438	0.479	24295	0.0714	0.1034	0.4817	0.1893	0.0145	0.8880	0.9348	0.9849
	GenFusion+V2G	16.42	0.569	0.425	24295	0.0162	0.0023	0.0615	0.0355	0.0071	0.9956	0.9992	1.0000
	GSFixer+V2G	15.93	0.557	0.426	24295	0.0307	0.0076	0.1161	0.0786	0.0103	0.9708	0.9923	0.9998
	Ours	16.05	0.571	0.422	24295	0.0373	0.0094	0.1242	0.0876	0.0094	0.9443	0.9980	0.9998
Stump	ViewCrafter+V2G	15.34	0.248	0.612	2917	0.2006	0.0842	0.3299	0.3393	0.1461	0.6250	0.8426	0.9071
	GenFusion+V2G	16.69	0.301	0.635	2917	0.0827	0.0163	0.1441	0.1318	0.0659	0.9164	0.9856	1.0000
	GSFixer+V2G	17.94	0.320	0.587	2917	0.0265	0.0038	0.0695	0.0681	0.0128	0.9815	0.9935	1.0000
	Ours	17.92	0.325	0.592	2917	0.0299	0.0042	0.0673	0.0713	0.0125	0.9619	0.9969	1.0000
treehill	ViewCrafter+V2G	12.62	0.255	0.641	5259	0.2373	0.1054	0.3658	0.4584	0.1945	0.5092	0.7385	0.8810
	GenFusion+V2G	13.94	0.303	0.649	5259	0.0380	0.0071	0.1026	0.0912	0.0156	0.9682	0.9882	0.9970
	GSFixer+V2G	15.26	0.323	0.607	5259	0.0523	0.0117	0.1287	0.1160	0.0221	0.9049	0.9899	0.9990
	Ours	15.64	0.340	0.602	5259	0.0294	0.0040	0.0742	0.0648	0.0128	0.9696	0.9998	1.0000

Table 9: Per-scene rendering and sparse-depth geometry results for the plug-and-play V2G evaluation on Mip-NeRF 360 with 6 input views. The "#Pts" column gives the retained shared SfM points used for geometry evaluation.

Scene	Method	Rendering			Geometry								
		PSNR↑	SSIM↑	LPIPS↓	#Pts	AbsRel↓	SqRel↓	RMSE↓	RMSE-log↓	MedianRel↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
Bicycle	ViewCrafter+V2G	14.60	0.226	0.567	3035	0.1004	0.0572	0.3162	0.2569	0.0315	0.8329	0.9193	0.9545
	GenFusion+V2G	16.19	0.292	0.576	3035	0.0310	0.0104	0.1194	0.0696	0.0113	0.9885	0.9970	0.9980
	GSFixer+V2G	17.51	0.305	0.534	3035	0.0429	0.0120	0.1435	0.0954	0.0177	0.9671	0.9898	0.9957
	Ours	16.96	0.314	0.541	3035	0.0296	0.0064	0.0973	0.0600	0.0122	0.9862	0.9987	0.9993
Bonsai	ViewCrafter+V2G	15.28	0.497	0.465	28448	0.0183	0.0101	0.1524	0.0594	0.0063	0.9864	0.9959	0.9987
	GenFusion+V2G	17.15	0.561	0.428	28448	0.0127	0.0062	0.0792	0.0476	0.0063	0.9964	0.9976	0.9982
	GSFixer+V2G	17.65	0.563	0.434	28448	0.0469	0.0144	0.1660	0.1077	0.0215	0.9550	0.9915	0.9939
	Ours	18.48	0.586	0.420	28448	0.0167	0.0045	0.0796	0.0472	0.0084	0.9914	0.9976	0.9989
Counter	ViewCrafter+V2G	15.40	0.461	0.448	21785	0.0218	0.0073	0.1270	0.0586	0.0070	0.9878	0.9972	0.9987
	GenFusion+V2G	17.03	0.542	0.425	21785	0.0156	0.0027	0.0669	0.0396	0.0061	0.9947	0.9993	0.9996
	GSFixer+V2G	17.23	0.537	0.420	21785	0.0615	0.0115	0.1192	0.1116	0.0241	0.9100	0.9969	0.9995
	Ours	17.45	0.557	0.409	21785	0.0182	0.0031	0.0593	0.0485	0.0063	0.9897	0.9989	0.9995
Flowers	ViewCrafter+V2G	13.01	0.195	0.606	947	0.1022	0.0534	0.2889	0.2773	0.0272	0.8427	0.9134	0.9567
	GenFusion+V2G	13.57	0.224	0.639	947	0.0444	0.0204	0.1289	0.1004	0.0189	0.9799	0.9937	0.9947
	GSFixer+V2G	14.15	0.217	0.575	947	0.0783	0.0710	0.3196	0.1561	0.0208	0.9356	0.9747	0.9852
	Ours	14.85	0.237	0.577	947	0.0595	0.0635	0.2587	0.1369	0.0171	0.9736	0.9810	0.9894
Garden	ViewCrafter+V2G	16.70	0.366	0.442	21170	0.0418	0.0193	0.2010	0.1010	0.0128	0.9452	0.9854	0.9991
	GenFusion+V2G	18.63	0.396	0.450	21170	0.0121	0.0011	0.0451	0.0269	0.0068	0.9983	0.9995	0.9999
	GSFixer+V2G	18.90	0.407	0.423	21170	0.0367	0.0081	0.1132	0.0845	0.0127	0.9625	0.9997	0.9999
	Ours	19.15	0.424	0.430	21170	0.0184	0.0028	0.0740	0.0415	0.0089	0.9941	0.9980	1.0000
Kitchen	ViewCrafter+V2G	17.06	0.492	0.396	54085	0.0113	0.0011	0.0345	0.0425	0.0041	0.9947	0.9977	0.9987
	GenFusion+V2G	18.66	0.562	0.385	54085	0.0303	0.0138	0.1322	0.1439	0.0036	0.9448	0.9451	0.9943
	GSFixer+V2G	18.64	0.566	0.363	54085	0.0176	0.0058	0.0735	0.1306	0.0040	0.9818	0.9856	0.9865
	Ours	19.23	0.568	0.362	54085	0.0121	0.0015	0.0364	0.0540	0.0040	0.9884	0.9935	0.9997
Room	ViewCrafter+V2G	15.09	0.518	0.431	30456	0.0530	0.0174	0.1816	0.1168	0.0148	0.9337	0.9843	0.9984
	GenFusion+V2G	17.84	0.630	0.386	30456	0.0116	0.0014	0.0475	0.0268	0.0059	0.9969	0.9998	1.0000
	GSFixer+V2G	16.97	0.608	0.390	30456	0.0185	0.0023	0.0617	0.0401	0.0083	0.9949	0.9997	0.9999
	Ours	16.84	0.614	0.394	30456	0.0277	0.0065	0.1095	0.0591	0.0098	0.9849	0.9990	0.9997
Stump	ViewCrafter+V2G	17.08	0.295	0.555	4482	0.1032	0.0381	0.2239	0.2274	0.0343	0.8030	0.9248	0.9659
	GenFusion+V2G	17.89	0.318	0.571	4482	0.0246	0.0074	0.0854	0.1479	0.0077	0.9817	0.9844	0.9851
	GSFixer+V2G	19.15	0.337	0.532	4482	0.0198	0.0027	0.0705	0.0404	0.0108	0.9958	0.9991	0.9998
	Ours	18.95	0.342	0.540	4482	0.0215	0.0032	0.0729	0.0525	0.0097	0.9877	0.9973	1.0000
treehill	ViewCrafter+V2G	13.86	0.271	0.568	11218	0.0676	0.0195	0.1689	0.1641	0.0337	0.9240	0.9663	0.9816
	GenFusion+V2G	15.54	0.354	0.589	11218	0.0148	0.0016	0.0566	0.0332	0.0077	0.9951	0.9996	0.9999
	GSFixer+V2G	16.70	0.349	0.542	11218	0.0293	0.0032	0.0683	0.0546	0.0153	0.9856	0.9996	0.9999
	Ours	17.37	0.374	0.536	11218	0.0171	0.0016	0.0524	0.0333	0.0097	0.9971	0.9996	0.9997

Table 10: Per-scene rendering and sparse-depth geometry results for the plug-and-play V2G evaluation on Mip-NeRF 360 with 9 input views. The "#Pts" column gives the retained shared SfM points used for geometry evaluation.

Scene	Method	Rendering			Geometry								
		PSNR↑	SSIM↑	LPIPS↓	#Pts	AbsRel↓	SqRel↓	RMSE↓	RMSE-log↓	MedianRel↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
Bicycle	ViewCrafter+V2G	15.19	0.272	0.530	4984	0.0841	0.0287	0.2052	0.1843	0.0332	0.8826	0.9482	0.9858
	GenFusion+V2G	16.48	0.313	0.548	4984	0.0281	0.0085	0.1287	0.0687	0.0107	0.9811	0.9952	0.9990
	GSFixer+V2G	18.11	0.327	0.500	4984	0.0315	0.0049	0.0901	0.0543	0.0154	0.9928	0.9992	0.9996
	Ours	18.02	0.339	0.508	4984	0.0303	0.0050	0.0884	0.0590	0.0135	0.9858	0.9980	0.9998
Bonsai	ViewCrafter+V2G	17.71	0.610	0.369	42744	0.0145	0.0091	0.1467	0.0507	0.0050	0.9915	0.9962	0.9987
	GenFusion+V2G	19.05	0.661	0.342	42744	0.0104	0.0051	0.0726	0.0431	0.0049	0.9967	0.9979	0.9986
	GSFixer+V2G	19.47	0.658	0.352	42744	0.0196	0.0058	0.1083	0.0512	0.0082	0.9927	0.9976	0.9992
	Ours	20.01	0.672	0.343	42744	0.0164	0.0045	0.0788	0.0435	0.0083	0.9965	0.9981	0.9990
Counter	ViewCrafter+V2G	17.11	0.533	0.387	29081	0.0160	0.0058	0.1153	0.0483	0.0058	0.9920	0.9974	0.9992
	GenFusion+V2G	18.10	0.603	0.368	29081	0.0133	0.0026	0.0713	0.0390	0.0052	0.9959	0.9989	0.9995
	GSFixer+V2G	19.05	0.606	0.357	29081	0.0166	0.0026	0.0685	0.0426	0.0066	0.9939	0.9985	0.9995
	Ours	19.37	0.628	0.351	29081	0.0160	0.0029	0.0627	0.0434	0.0062	0.9966	0.9985	0.9993
Flowers	ViewCrafter+V2G	13.78	0.224	0.548	2410	0.0658	0.0288	0.2637	0.1470	0.0210	0.8954	0.9693	0.9946
	GenFusion+V2G	14.57	0.251	0.606	2410	0.0357	0.0109	0.1416	0.0868	0.0150	0.9759	0.9929	0.9954
	GSFixer+V2G	14.89	0.243	0.530	2410	0.0384	0.0394	0.2716	0.0905	0.0143	0.9784	0.9909	0.9963
	Ours	15.74	0.267	0.526	2410	0.0321	0.0163	0.1810	0.0730	0.0130	0.9842	0.9938	0.9967
Garden	ViewCrafter+V2G	18.60	0.453	0.373	27230	0.0201	0.0155	0.2044	0.0525	0.0071	0.9921	0.9975	0.9992
	GenFusion+V2G	19.84	0.473	0.394	27230	0.0092	0.0007	0.0381	0.0198	0.0051	0.9990	0.9997	1.0000
	GSFixer+V2G	20.15	0.481	0.364	27230	0.0193	0.0040	0.0947	0.0383	0.0089	0.9958	0.9992	0.9999
	Ours	20.38	0.493	0.369	27230	0.0242	0.0075	0.1100	0.0900	0.0074	0.9757	0.9846	0.9926
Kitchen	ViewCrafter+V2G	19.33	0.576	0.335	63142	0.0206	0.0054	0.0812	0.0972	0.0040	0.9795	0.9882	0.9938
	GenFusion+V2G	20.79	0.645	0.311	63142	0.0056	0.0003	0.0188	0.0136	0.0034	0.9997	0.9999	0.9999
	GSFixer+V2G	20.48	0.634	0.309	63142	0.0067	0.0003	0.0227	0.0153	0.0037	0.9995	0.9998	1.0000
	Ours	20.98	0.631	0.304	63142	0.0057	0.0002	0.0161	0.0123	0.0034	0.9997	0.9999	1.0000
Room	ViewCrafter+V2G	17.82	0.640	0.348	43187	0.0488	0.0429	0.3056	0.2114	0.0079	0.9390	0.9645	0.9727
	GenFusion+V2G	19.66	0.699	0.312	43187	0.0100	0.0014	0.0504	0.0260	0.0047	0.9969	0.9998	1.0000
	GSFixer+V2G	19.16	0.677	0.331	43187	0.0240	0.0054	0.0988	0.0634	0.0080	0.9759	0.9964	1.0000
	Ours	19.47	0.692	0.321	43187	0.0221	0.0034	0.0768	0.0487	0.0081	0.9939	0.9995	0.9999
Stump	ViewCrafter+V2G	18.39	0.350	0.502	6174	0.0727	0.0243	0.1743	0.1935	0.0137	0.8633	0.9279	0.9759
	GenFusion+V2G	19.00	0.376	0.529	6174	0.0422	0.0174	0.1567	0.1562	0.0056	0.9231	0.9546	0.9841
	GSFixer+V2G	19.86	0.381	0.490	6174	0.0204	0.0019	0.0536	0.0371	0.0099	0.9976	1.0000	1.0000
	Ours	19.69	0.392	0.494	6174	0.0142	0.0014	0.0476	0.0288	0.0066	0.9981	0.9995	1.0000
treehill	ViewCrafter+V2G	15.58	0.331	0.522	13873	0.0286	0.0058	0.0998	0.0742	0.0099	0.9677	0.9964	0.9996
	GenFusion+V2G	16.72	0.384	0.566	13873	0.0239	0.0030	0.0590	0.0500	0.0099	0.9890	0.9993	0.9999
	GSFixer+V2G	17.59	0.371	0.503	13873	0.0268	0.0024	0.0631	0.0447	0.0150	0.9960	0.9997	1.0000
	Ours	17.90	0.398	0.501	13873	0.0164	0.0024	0.0636	0.0289	0.0099	0.9994	0.9999	0.9999

F.3 TANKS AND TEMPLES

Table 11: Per-scene rendering results on Tanks and Temples under the 4-, 6-, and 9-view settings.

(a) 4 input views

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
barn	12.78	0.447	0.538	12.90	0.474	0.576	13.45	0.442	0.485	15.01	0.492	0.514	15.46	0.519	0.500
caterpillar	12.17	0.376	0.558	12.35	0.408	0.548	13.09	0.379	0.511	13.67	0.414	0.537	13.93	0.437	0.514
family	11.42	0.449	0.554	12.46	0.521	0.535	13.33	0.463	0.493	13.06	0.523	0.511	13.57	0.554	0.493
horse	12.91	0.612	0.451	14.36	0.672	0.411	13.39	0.600	0.405	14.30	0.644	0.417	14.29	0.662	0.400
ignatius	12.28	0.346	0.566	12.88	0.374	0.582	12.87	0.322	0.520	13.43	0.376	0.542	14.19	0.407	0.526
truck	12.72	0.484	0.528	13.07	0.530	0.535	14.53	0.503	0.474	14.35	0.542	0.498	14.78	0.567	0.485
Average	**12.38**	**0.452**	**0.533**	**13.00**	**0.497**	**0.531**	**13.44**	**0.451**	**0.481**	**13.97**	**0.498**	**0.503**	**14.37**	**0.525**	**0.486**

(b) 6 input views

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
barn	13.69	0.462	0.511	14.08	0.501	0.537	14.15	0.456	0.468	15.41	0.485	0.502	16.13	0.517	0.479
caterpillar	13.41	0.410	0.514	13.43	0.448	0.502	13.98	0.396	0.476	14.90	0.433	0.495	15.11	0.459	0.478
family	13.13	0.517	0.470	14.94	0.599	0.450	15.24	0.497	0.440	15.87	0.586	0.424	16.43	0.614	0.407
horse	15.83	0.668	0.375	16.09	0.713	0.357	15.16	0.624	0.362	16.76	0.691	0.346	16.56	0.709	0.344
ignatius	13.51	0.380	0.508	13.77	0.410	0.536	13.84	0.345	0.491	14.44	0.403	0.485	15.04	0.432	0.476
truck	14.43	0.537	0.473	15.06	0.583	0.483	15.55	0.519	0.443	15.42	0.568	0.454	16.30	0.598	0.440
Average	**14.00**	**0.496**	**0.475**	**14.56**	**0.543**	**0.478**	**14.65**	**0.473**	**0.446**	**15.47**	**0.528**	**0.451**	**15.93**	**0.555**	**0.437**

(c) 9 input views

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
barn	14.92	0.482	0.481	15.75	0.529	0.508	14.78	0.480	0.448	16.22	0.494	0.473	16.60	0.530	0.463
caterpillar	15.24	0.456	0.459	15.54	0.499	0.469	15.44	0.419	0.440	16.16	0.469	0.446	16.47	0.492	0.439
family	17.19	0.615	0.374	17.20	0.666	0.389	16.67	0.517	0.401	17.44	0.641	0.352	18.24	0.669	0.337
horse	17.24	0.701	0.331	16.83	0.742	0.334	16.00	0.638	0.338	17.83	0.720	0.310	17.38	0.730	0.316
ignatius	14.81	0.421	0.449	15.31	0.465	0.491	14.79	0.366	0.464	15.67	0.443	0.433	16.15	0.468	0.427
truck	15.61	0.581	0.427	15.98	0.618	0.446	16.08	0.530	0.421	16.42	0.599	0.417	16.72	0.622	0.408
Average	**15.83**	**0.542**	**0.420**	**16.10**	**0.587**	**0.439**	**15.62**	**0.492**	**0.419**	**16.62**	**0.561**	**0.405**	**16.93**	**0.585**	**0.398**

F.4 CO3D

Table 12: Per-scene rendering results on CO3D with 3 input views. Scene identifiers retain the category and first two sequence components.

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
bench-167-18175	11.97	0.345	0.590	12.70	0.435	0.564	12.19	0.363	0.539	13.34	0.430	0.566	14.37	0.470	0.502
bench-185-19987	12.89	0.270	0.584	15.21	0.375	0.589	13.20	0.220	0.552	14.86	0.386	0.600	14.64	0.365	0.571
cake-201-21573	16.84	0.505	0.521	16.60	0.598	0.475	15.16	0.497	0.594	17.67	0.614	0.472	17.75	0.618	0.451
cake-201-21631	11.74	0.450	0.552	13.60	0.633	0.528	12.50	0.502	0.575	12.67	0.575	0.523	13.00	0.592	0.517
donut-198-21296	15.43	0.566	0.485	16.76	0.678	0.425	17.08	0.617	0.448	16.31	0.667	0.432	17.64	0.681	0.374
donut-290-30761	12.09	0.430	0.588	13.01	0.585	0.527	12.30	0.472	0.605	13.14	0.594	0.527	13.12	0.594	0.503
hydrant-134-15451	14.47	0.296	0.558	14.07	0.362	0.621	14.29	0.259	0.559	14.88	0.385	0.583	16.97	0.395	0.519
hydrant-147-16374	14.10	0.543	0.480	15.28	0.592	0.482	15.28	0.523	0.467	15.05	0.561	0.465	16.40	0.633	0.439
plant-188-20319	11.66	0.393	0.592	12.52	0.517	0.570	11.50	0.399	0.612	12.88	0.526	0.563	14.21	0.550	0.524
plant-188-20360	13.64	0.441	0.614	14.56	0.548	0.580	13.90	0.430	0.612	15.54	0.565	0.541	15.58	0.568	0.531
suitcase-197-21239	14.33	0.420	0.548	16.08	0.537	0.512	14.66	0.406	0.515	15.54	0.513	0.524	17.51	0.561	0.474
suitcase-247-26503	12.78	0.534	0.475	15.88	0.680	0.348	13.03	0.530	0.528	13.96	0.617	0.426	17.42	0.727	0.346
teddybear-118-13848	15.66	0.629	0.505	16.44	0.722	0.484	16.95	0.652	0.502	17.83	0.705	0.466	18.48	0.743	0.446
teddybear-159-17485	15.17	0.580	0.475	15.88	0.645	0.465	16.08	0.594	0.478	15.32	0.603	0.439	16.58	0.618	0.424
vase-101-11754	16.42	0.724	0.532	18.61	0.826	0.484	17.18	0.760	0.577	15.50	0.744	0.491	13.81	0.713	0.463
vase-118-13872	14.33	0.552	0.598	16.58	0.708	0.539	16.06	0.575	0.563	16.19	0.655	0.529	17.21	0.693	0.506
Average	**13.97**	**0.480**	**0.543**	**15.24**	**0.590**	**0.512**	**14.46**	**0.487**	**0.545**	**15.04**	**0.571**	**0.509**	**15.92**	**0.595**	**0.474**

Table 13: Per-scene rendering results on CO3D with 6 input views.

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
bench-167-18175	14.42	0.411	0.487	14.65	0.455	0.505	13.82	0.388	0.454	15.24	0.448	0.495	17.34	0.528	0.392
bench-185-19987	15.57	0.323	0.544	16.09	0.346	0.552	15.23	0.254	0.481	16.12	0.348	0.548	17.02	0.373	0.479
cake-201-21573	20.63	0.662	0.394	18.42	0.638	0.447	18.69	0.619	0.504	20.38	0.674	0.408	22.86	0.731	0.337
cake-201-21631	14.36	0.579	0.487	14.81	0.655	0.493	15.20	0.619	0.481	16.15	0.664	0.458	15.91	0.663	0.424
donut-198-21296	19.12	0.674	0.383	18.99	0.695	0.378	18.94	0.674	0.403	19.07	0.703	0.395	20.32	0.724	0.341
donut-290-30761	12.42	0.484	0.561	13.81	0.627	0.524	13.14	0.554	0.561	13.59	0.594	0.528	14.17	0.622	0.483
hydrant-134-15451	16.28	0.348	0.522	15.60	0.356	0.573	15.38	0.291	0.494	15.57	0.370	0.562	18.80	0.403	0.459
hydrant-147-16374	17.12	0.596	0.424	17.17	0.640	0.460	16.68	0.581	0.417	18.07	0.615	0.416	18.82	0.662	0.380
plant-188-20319	13.20	0.432	0.555	14.08	0.540	0.534	13.07	0.446	0.558	14.02	0.509	0.534	16.17	0.594	0.451
plant-188-20360	16.05	0.538	0.517	16.79	0.604	0.498	16.55	0.535	0.508	17.25	0.577	0.488	19.08	0.635	0.429
suitcase-197-21239	18.12	0.530	0.460	18.02	0.554	0.493	17.98	0.478	0.453	17.21	0.513	0.483	20.58	0.583	0.424
suitcase-247-26503	17.68	0.703	0.363	17.94	0.755	0.329	16.33	0.670	0.446	19.25	0.735	0.354	21.79	0.814	0.277
teddybear-118-13848	22.14	0.765	0.398	23.07	0.791	0.406	22.45	0.741	0.404	23.10	0.763	0.395	25.91	0.824	0.336
teddybear-159-17485	20.99	0.727	0.320	18.74	0.675	0.384	19.50	0.678	0.360	20.63	0.714	0.330	21.04	0.727	0.305
vase-101-11754	19.43	0.784	0.447	20.54	0.815	0.428	20.14	0.775	0.491	18.78	0.773	0.461	19.16	0.798	0.398
vase-118-13872	19.26	0.657	0.490	19.51	0.719	0.510	19.68	0.617	0.511	19.37	0.679	0.480	20.31	0.712	0.454
Average	**17.30**	**0.576**	**0.459**	**17.39**	**0.616**	**0.470**	**17.05**	**0.558**	**0.470**	**17.74**	**0.605**	**0.458**	**19.33**	**0.650**	**0.398**

Table 14: Per-scene rendering results on CO3D with 9 input views.

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
bench-167-18175	16.20	0.466	0.431	15.60	0.483	0.474	14.63	0.422	0.422	15.49	0.463	0.478	18.92	0.561	0.342
bench-185-19987	17.21	0.355	0.484	17.20	0.373	0.512	16.27	0.295	0.439	16.61	0.361	0.508	18.39	0.403	0.432
cake-201-21573	23.53	0.757	0.308	21.98	0.732	0.344	21.55	0.699	0.438	23.37	0.747	0.335	24.91	0.781	0.275
cake-201-21631	16.62	0.665	0.415	17.45	0.710	0.404	16.41	0.663	0.439	16.74	0.697	0.417	17.90	0.715	0.378
donut-198-21296	21.41	0.723	0.329	21.55	0.733	0.349	21.87	0.735	0.340	20.58	0.730	0.351	21.25	0.753	0.304
donut-290-30761	13.11	0.533	0.527	14.16	0.627	0.488	13.47	0.565	0.541	14.01	0.597	0.510	15.32	0.649	0.457
hydrant-134-15451	17.08	0.375	0.487	16.25	0.357	0.546	16.08	0.306	0.466	16.43	0.385	0.539	19.04	0.406	0.434
hydrant-147-16374	19.86	0.655	0.354	19.66	0.691	0.404	18.38	0.624	0.355	19.82	0.643	0.363	20.69	0.677	0.335
plant-188-20319	14.87	0.525	0.505	14.70	0.549	0.500	13.50	0.485	0.540	14.40	0.540	0.514	17.45	0.636	0.419
plant-188-20360	19.66	0.654	0.426	18.45	0.658	0.454	18.98	0.616	0.439	19.22	0.630	0.442	21.30	0.699	0.375
suitcase-197-21239	20.85	0.584	0.400	20.86	0.625	0.429	19.39	0.517	0.415	19.28	0.558	0.426	22.34	0.613	0.375
suitcase-247-26503	21.55	0.822	0.269	22.74	0.852	0.234	18.80	0.760	0.376	21.42	0.799	0.292	22.49	0.839	0.246
teddybear-118-13848	24.79	0.809	0.336	25.08	0.832	0.368	24.80	0.778	0.360	23.91	0.781	0.358	26.59	0.846	0.291
teddybear-159-17485	23.02	0.776	0.276	21.41	0.738	0.333	21.67	0.739	0.313	22.64	0.759	0.289	22.94	0.775	0.275
vase-101-11754	19.63	0.797	0.454	22.20	0.840	0.400	20.31	0.783	0.503	19.10	0.783	0.458	20.89	0.829	0.372
vase-118-13872	20.24	0.682	0.468	22.00	0.753	0.473	20.32	0.639	0.499	20.53	0.679	0.464	22.08	0.727	0.434
Average	**19.35**	**0.636**	**0.404**	**19.45**	**0.660**	**0.420**	**18.53**	**0.602**	**0.430**	**18.97**	**0.634**	**0.422**	**20.78**	**0.682**	**0.359**

F.5 DL3DV-BENCHMARK

Table 15: Per-scene rendering results on DL3DV-Benchmark with 3 input views.

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
073f5a9b98	10.59	0.161	0.617	11.33	0.222	0.683	11.09	0.146	0.602	12.58	0.237	0.632	13.17	0.257	0.620
093ef327b4	13.08	0.415	0.545	15.59	0.575	0.479	14.90	0.508	0.478	16.14	0.578	0.474	17.08	0.602	0.453
0bfdd020cf	13.27	0.334	0.552	15.65	0.463	0.517	15.12	0.382	0.456	16.34	0.457	0.487	17.24	0.496	0.453
1264931635	12.74	0.205	0.581	14.24	0.265	0.613	13.15	0.186	0.525	14.58	0.269	0.569	15.08	0.297	0.554
183dd248f6	14.59	0.491	0.508	16.64	0.620	0.441	16.91	0.573	0.461	17.09	0.621	0.456	17.47	0.653	0.425
41036716da	13.89	0.502	0.502	15.57	0.612	0.443	15.85	0.579	0.452	16.01	0.607	0.443	17.22	0.672	0.396
484c0aca40	12.72	0.511	0.527	13.98	0.603	0.467	13.52	0.566	0.492	14.24	0.602	0.463	14.90	0.643	0.439
493816813d	12.12	0.250	0.548	13.96	0.338	0.533	13.60	0.273	0.478	14.70	0.342	0.521	15.43	0.378	0.503
4ae797d07b	12.10	0.339	0.524	14.44	0.476	0.474	13.70	0.416	0.459	14.38	0.455	0.472	15.19	0.510	0.441
54bf355ca7	14.47	0.501	0.478	15.94	0.597	0.427	15.50	0.545	0.424	16.40	0.588	0.405	16.83	0.627	0.381
599ca3e04c	12.34	0.405	0.557	14.38	0.550	0.483	14.24	0.474	0.496	14.54	0.528	0.498	14.60	0.551	0.471
5a69d1027b	12.59	0.237	0.542	14.39	0.314	0.562	14.00	0.243	0.480	15.32	0.326	0.502	16.38	0.367	0.466
66fd66cbed	14.05	0.581	0.500	15.67	0.675	0.437	14.61	0.618	0.478	15.57	0.666	0.438	16.18	0.693	0.403
6d22162561	11.22	0.375	0.606	13.14	0.492	0.570	12.84	0.431	0.526	14.67	0.517	0.547	15.35	0.542	0.517
6e11e7f4fe	15.02	0.690	0.513	16.91	0.784	0.428	18.99	0.769	0.442	16.87	0.784	0.433	17.05	0.803	0.393
7da3db9905	12.80	0.422	0.544	14.43	0.545	0.479	13.75	0.477	0.451	13.67	0.504	0.521	15.00	0.559	0.457
9c8c0e0fad	11.62	0.204	0.627	13.05	0.287	0.664	11.66	0.174	0.573	14.13	0.293	0.592	14.93	0.311	0.564
a62c330f54	10.94	0.318	0.616	12.07	0.420	0.596	11.73	0.356	0.592	13.19	0.459	0.579	13.83	0.491	0.540
Average	**12.79**	**0.385**	**0.549**	**14.52**	**0.491**	**0.516**	**14.17**	**0.429**	**0.492**	**15.02**	**0.491**	**0.502**	**15.72**	**0.525**	**0.471**

Table 16: Per-scene rendering results on DL3DV-Benchmark with 6 input views.

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
073f5a9b98	11.99	0.192	0.577	12.26	0.228	0.642	12.44	0.177	0.561	13.51	0.244	0.590	13.82	0.261	0.565
093ef327b4	16.26	0.567	0.434	17.83	0.666	0.391	17.42	0.625	0.374	17.14	0.627	0.410	18.57	0.683	0.362
0bfdd020cf	16.19	0.449	0.437	17.83	0.535	0.417	17.38	0.465	0.367	18.16	0.511	0.400	18.46	0.542	0.383
1264931635	15.05	0.292	0.477	16.27	0.339	0.499	15.13	0.272	0.439	15.92	0.319	0.463	16.48	0.355	0.452
183dd248f6	18.89	0.648	0.390	19.77	0.704	0.358	19.78	0.673	0.360	19.85	0.700	0.364	19.99	0.720	0.343
41036716da	17.10	0.635	0.387	18.60	0.713	0.338	18.07	0.685	0.358	18.34	0.704	0.350	18.99	0.734	0.319
484c0aca40	14.89	0.616	0.428	15.41	0.655	0.406	15.08	0.632	0.395	15.77	0.653	0.404	16.17	0.676	0.385
493816813d	15.44	0.396	0.414	16.45	0.473	0.385	15.71	0.410	0.366	16.71	0.453	0.385	17.31	0.500	0.364
4ae797d07b	15.89	0.508	0.395	17.50	0.598	0.354	16.78	0.555	0.346	17.78	0.595	0.347	18.28	0.623	0.328
54bf355ca7	17.92	0.612	0.364	19.19	0.683	0.317	18.31	0.646	0.319	19.23	0.670	0.306	19.79	0.700	0.283
599ca3e04c	15.14	0.519	0.470	16.37	0.606	0.424	16.32	0.568	0.416	16.13	0.577	0.429	16.63	0.616	0.402
5a69d1027b	16.08	0.368	0.416	16.42	0.414	0.436	16.21	0.364	0.377	17.41	0.418	0.380	17.96	0.455	0.360
66fd66cbed	17.75	0.704	0.364	18.75	0.763	0.308	18.21	0.731	0.322	18.46	0.748	0.319	19.36	0.773	0.293
6d22162561	14.74	0.488	0.510	15.93	0.565	0.493	15.35	0.500	0.437	16.79	0.548	0.474	17.76	0.581	0.439
6e11e7f4fe	19.20	0.794	0.413	21.65	0.849	0.343	22.24	0.821	0.345	20.09	0.833	0.362	20.00	0.849	0.324
7da3db9905	15.53	0.556	0.425	17.27	0.656	0.367	15.94	0.593	0.341	16.48	0.608	0.394	17.72	0.649	0.361
9c8c0e0fad	14.35	0.290	0.506	15.70	0.354	0.512	14.49	0.269	0.450	16.14	0.345	0.463	16.60	0.377	0.438
a62c330f54	15.15	0.502	0.452	16.30	0.581	0.423	15.67	0.525	0.421	16.56	0.569	0.412	16.94	0.595	0.394
Average	**15.97**	**0.508**	**0.437**	**17.20**	**0.577**	**0.412**	**16.70**	**0.528**	**0.389**	**17.25**	**0.562**	**0.403**	**17.82**	**0.594**	**0.377**

Table 17: Per-scene rendering results on DL3DV-Benchmark with 9 input views.

Scene	ViewCrafter			GenFusion			Difix3D+			GSFixer			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
073f5a9b98	13.36	0.249	0.531	13.96	0.283	0.578	13.67	0.225	0.520	14.64	0.282	0.538	14.88	0.297	0.518
093ef327b4	18.95	0.684	0.348	20.04	0.739	0.328	19.72	0.711	0.307	20.17	0.732	0.315	20.37	0.749	0.303
0bfdd020cf	17.77	0.502	0.391	18.96	0.572	0.386	18.04	0.499	0.340	18.97	0.539	0.362	19.24	0.569	0.348
1264931635	16.74	0.370	0.406	17.39	0.404	0.444	16.40	0.339	0.391	17.25	0.386	0.399	17.68	0.416	0.392
183dd248f6	20.98	0.715	0.334	21.79	0.760	0.312	21.35	0.725	0.312	21.52	0.747	0.313	21.91	0.769	0.293
41036716da	18.48	0.709	0.327	19.60	0.765	0.291	19.14	0.733	0.314	19.36	0.750	0.304	19.91	0.770	0.282
484c0aca40	16.95	0.684	0.361	17.00	0.701	0.359	16.82	0.683	0.343	17.18	0.691	0.360	17.87	0.719	0.337
493816813d	17.65	0.542	0.311	18.33	0.594	0.293	17.51	0.529	0.294	18.40	0.579	0.291	18.98	0.610	0.277
4ae797d07b	17.86	0.604	0.329	18.99	0.663	0.304	18.24	0.623	0.297	18.80	0.648	0.301	19.42	0.681	0.283
54bf355ca7	19.35	0.668	0.309	20.41	0.726	0.272	19.74	0.693	0.275	20.31	0.705	0.268	20.91	0.734	0.245
599ca3e04c	16.74	0.575	0.415	18.07	0.663	0.356	17.49	0.614	0.370	17.74	0.626	0.372	18.21	0.658	0.345
5a69d1027b	17.20	0.433	0.359	17.82	0.479	0.380	17.48	0.427	0.331	18.11	0.471	0.333	18.46	0.497	0.321
66fd66cbed	19.22	0.760	0.306	20.65	0.812	0.255	20.02	0.786	0.266	19.62	0.785	0.279	20.73	0.814	0.251
6d22162561	17.66	0.569	0.427	17.94	0.618	0.430	17.73	0.560	0.368	18.49	0.593	0.408	19.39	0.626	0.377
6e11e7f4fe	22.52	0.847	0.329	24.34	0.885	0.288	24.83	0.864	0.287	22.54	0.872	0.292	22.55	0.885	0.267
7da3db9905	17.84	0.640	0.357	18.66	0.699	0.332	17.58	0.650	0.297	17.25	0.654	0.346	18.79	0.691	0.317
9c8c0e0fad	15.91	0.365	0.429	16.84	0.410	0.452	16.01	0.345	0.388	17.03	0.397	0.405	17.30	0.427	0.388
a62c330f54	17.06	0.576	0.381	18.00	0.640	0.357	17.74	0.600	0.351	17.84	0.617	0.358	18.18	0.638	0.339
Average	**17.90**	**0.583**	**0.370**	**18.82**	**0.634**	**0.357**	**18.30**	**0.589**	**0.336**	**18.62**	**0.615**	**0.347**	**19.15**	**0.642**	**0.327**