Splatent: Splatting Diffusion Latents for Novel View Synthesis

Supplementary Material

0.1. Introduction

In the supplementary material we provide additional analyses and results to support our main paper:

- **Multi-view inconsistencies in VAE latents:** Extended analysis demonstrating that VAE 3D inconsistencies exist across various VAE architectures, not just the Stable Diffusion VAE.
- **Ablation studies:** Experiments on multi-view attention designs, noise scheduling, and LPIPS loss weight.
- **Additional implementation details:** Additional details for reproducibility.
- **Further quantitative results:** Additional quantitative comparisons.
- **Further qualitative results:** Additional visual comparisons across both optimization-based and feed-forward settings.

0.2. Multi-view inconsistencies in VAE latents

In Sec. 4.2 of our main paper, we show that VAE latent representations include high-frequency components that are inconsistent across viewpoints. These inconsistencies result in latent representations that lack 3D coherence, limiting their direct use for 3D scene reconstruction and novel view synthesis.

To demonstrate that these multi-view inconsistencies are not tied to a specific VAE, we evaluate multiple VAE models, including SD [5], SD-XL [4], and FLUX [2]. Fig. 6 presents the spectral analysis of the latent representations of all VAE models. For comparison, we also show the spectral behavior of rendered RGB images. Across all tested VAEs, latents spectral profile exhibits significantly larger high-frequency components compared to RGB.

Importantly, we observe that all VAEs exhibit similar 3D inconsistencies: when optimizing 3DGS directly in the latent space, the rendered latents consistently lose high-frequency details relative to their encoded features (solid vs. dashed line in Fig. 6). This indicates that the multi-view inconsistency problem is a fundamental limitation of current VAEs rather than an artifact of a specific model. The persistent degradation across different VAE designs further motivates our approach.

We further visualize this effect in Fig. 9 by projecting the latent features channel dimension to 3 (RGB) using PCA. As shown, rendered latent features appear noticeably smoother and lack the fine-grained texture present in ground-truth VAE-encoded latents, visually confirming the loss of high-frequency information during 3D reconstruction.

Table 4. **Multi-view attention ablation.** Comparison of different attention mechanisms for multi-view context. Self-attention follows Difix3D+ [6] by modifying the diffusion architecture. Cross-attention reduces memory by a factor of V but degrades performance. Our grid-based approach achieves superior results without architectural modifications.

Configuration	PSNR ↑	SSIM ↑	LPIPS ↓
Self-attention	21.58	0.690	**0.265**
Cross-attention	21.29	0.683	0.273
Splatent (grid)	**21.94**	**0.692**	**0.265**

0.3. Ablation studies

In addition to evaluating the impact of the number of reference images (Section 5.4), we examine several other components of our method. All ablations follow the experimental protocol described in Section 5.4.

Multi-view attention. In Sec. 4.3 we explain how we arrange the degraded latent with reference latents as a grid to leverage reference views for recovering missing details while preserving the geometry of the rendered latent. We also experiment with different multi-view attention designs. The results of these comparisons are reported in Tab. 4.

First, we follow Difix3D+ [6] by altering the design of attention layers in the diffusion model. In this approach, reference views are concatenated along the token dimension and fused through specialized attention blocks. This yields slightly inferior performance and requires modifications to the original diffusion model architecture, making it less generalizable to other diffusion architectures and future models. In contrast, our grid-based input solution does not necessitate any architectural changes, and yields the best performance.

Alternatively, we experiment with using cross-attention to provide multi-view context. In this design, we apply cross-attention between the degraded input $\hat{z}$ and the other reference views $\{z^i_{\text{ref}}\}$. To maintain the normal behavior of the diffusion model and allow information sharing between similar input features, we preserve the self-attention for each z_{ref}. Importantly, this results in reduced memory consumption. While self-attention requires $\mathcal{O}((VM)^2)$ memory, where V is the number of reference views and $M = h \cdot w$ (h, w are height and width respectively), the cross-attention implementation requires only $\mathcal{O}(VM^2)$, reducing memory consumption by a factor of V. Although this approach slightly degrades performance, it is benefi-

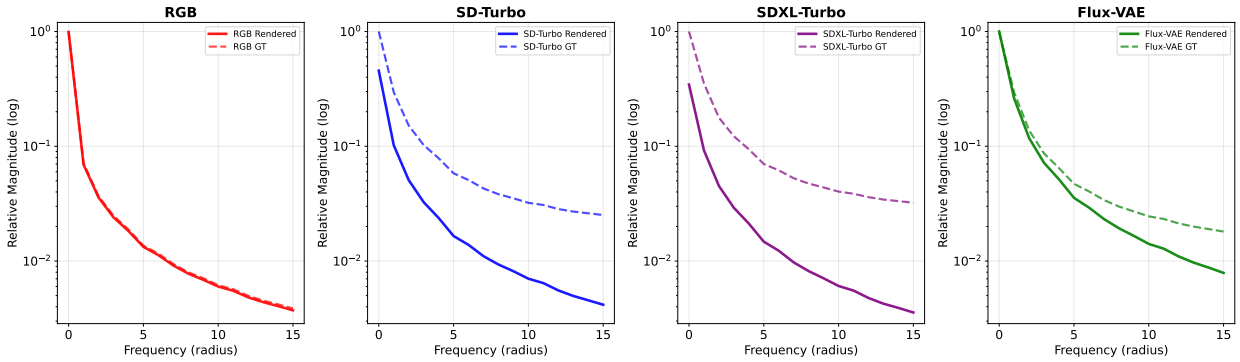

Figure 6. **Spectral analysis across different VAE architectures.** Magnitude spectrum of RGB images and latent features across various VAE models (SD, SDXL, FLUX), normalized to 1 and clipped to radius 15. All VAE models exhibit the same phenomenon: during 3DGS optimization, inconsistent high frequencies average out, leaving only low-frequency components and causing blurry decoded images. This demonstrates that multi-view inconsistency is a fundamental limitation across current VAE models, and not tied to a single model.

cial in memory-constrained scenarios, such as when using a large number of reference views, offering a trade-off between visual fidelity and computational efficiency.

Timestep selection. We also examine the impact of timestep selection on reconstruction quality. The noise scheduling in diffusion models determines the magnitude of changes applied during denoising, with timesteps serving as a control parameter. Our fine-tuned model inherits its behavior from the pre-trained diffusion model, which has learned to apply progressively stronger modifications at higher timesteps. This creates an inherent trade-off: at very low timesteps, the model applies minimal changes, failing to sufficiently refine the rendered latents and recover lost high-frequency details. Conversely, at excessively high timesteps, the model applies overly aggressive modifications that can hallucinate details inconsistent with the underlying 3D representation, effectively "overwriting" the input rather than refining it.

We train our model using 13 different timesteps ($\tau \in \{10, 20, 30, 40, 50, 100, 200, 300, 500, 600, 700, 800, 900\}$) and evaluate them using PSNR, SSIM and LPIPS. As shown in Fig. 7, our method demonstrates robustness across a wide range of intermediate timestep values (approximately $\tau \in [100, 600]$). As we fine-tune the diffusion model, it adapts to the scaling needed to fix the degraded input latent across a wide range of timesteps. Within this range, performance remains stable with negligible differences across metrics. This robustness is attributed to our geometric grounding signal from the rendered latents $\tilde{z}$, which provides rough structural constraints that prevent hallucinations even at moderately high noise levels. Additionally, the multi-view attention mechanism ensures that refinements remain consistent with the reference views $\{z^i_{\text{ref}}\}$. We select $\tau = 300$ as our default value, leading to best results across all metrics.

Table 5. **Weight λ_{LPIPS} loss ablation.** Impact of λ_{LPIPS} on reconstruction quality. Removing LPIPS loss ($\lambda_{\text{LPIPS}} = 0$) significantly degrades perceptual quality. Our method shows robustness across $\lambda_{\text{LPIPS}} \in \{1, 2, 5\}$ with stable PSNR/SSIM and moderate LPIPS variation. We use $\lambda_{\text{LPIPS}} = 2$ for optimal balance.

Configuration	PSNR ↑	SSIM ↑	LPIPS ↓
$\lambda_{\text{LPIPS}} = 0$	21.91	0.686	0.409
$\lambda_{\text{LPIPS}} = 1$	22.02	0.695	0.276
$\lambda_{\text{LPIPS}} = 5$	21.76	0.687	0.257
Splatent ($\lambda_{\text{LPIPS}} = 2$)	21.94	0.692	0.265

Weight λ_{LPIPS} in loss. We conduct an ablation study on the LPIPS loss weight λ_{LPIPS} from Eq 7, to analyze its impact on reconstruction quality. As shown in Table 5, our method demonstrates robustness to the choice of λ_{LPIPS}, with relatively small performance variations across different values. Removing the LPIPS loss ($\lambda_{\text{LPIPS}} = 0$) results in notably worse perceptual quality, while introducing any non-zero weight substantially improves perceptual metrics. Across the range of $\lambda_{\text{LPIPS}} \in \{1, 2, 5\}$, PSNR and SSIM remain stable, while LPIPS varies moderately.

We select $\lambda_{\text{LPIPS}} = 2$ as it provides a balanced performance. However, the consistent performance across different weight values demonstrates that our approach is not overly sensitive to this hyperparameter.

0.4. Feed-forward integration details

Fig. 8 illustrates our integration within the MVSplat360 [1] pipeline, where blue components indicate our modifications. Unlike the original MVSplat360, which directly inputs rendered features to the video diffusion model, Spla-

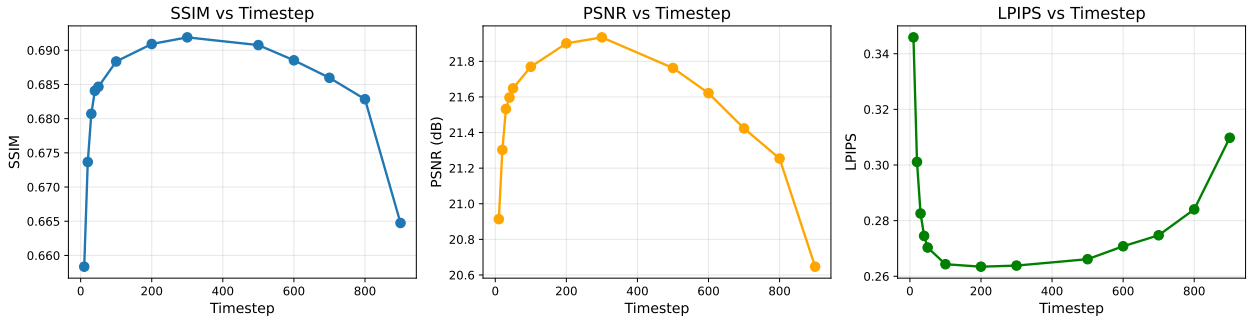

Figure 7. **Timestep Selection.** Our method demonstrates robustness across intermediate timestep values. Very low timesteps fail to sufficiently refine details, while excessively high timesteps cause hallucinations. Performance remains stable within $\tau \in [100, 600]$.

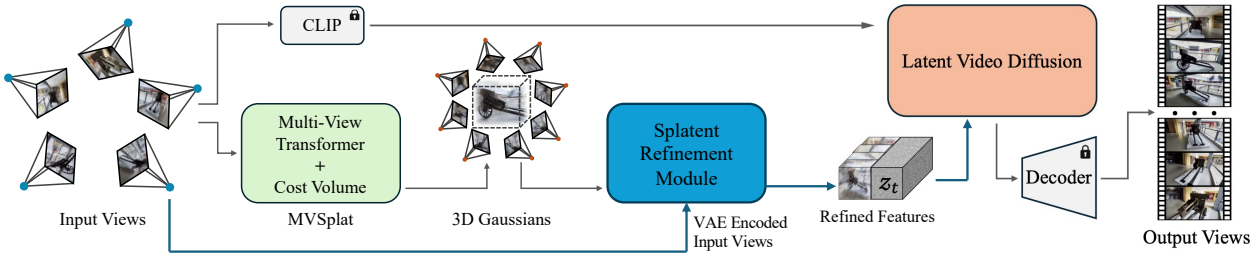

Figure 8. **Splatent integration in MVSplat360.** Our modifications to MVSplat360 are shown in blue. Unlike the original MVSplat360, which directly inputs rendered features to the video diffusion model, we use our Splatent module to refine the rendered latents using reference views. This refinement process recovers accurate high-frequency details lost during 3D reconstruction, and the refined features are then input to the Video Diffusion model.

tent allows for information sharing with reference latents before the Stable Video Diffusion (SVD) refinement stages. Specifically, the 3DGS backbone first renders latent features from the input views. Our Splatent module then refines these rendered latents by injecting high-frequency details from reference views during the single-step diffusion process, recovering fine-grained information lost during 3D reconstruction. Finally, SVD performs temporal refinement across the video sequence. Notably, we use pretrained MVSplat360 weights for the 3DGS backbone and only finetune SVD, demonstrating that Splatent enables seamless integration without requiring retraining of the 3D reconstruction component.

0.5. Additional implementation details.

Our code is publicly available. For comparisons, we use the original repositories and pre-trained models. To construct our dataset, closest cameras where chosen using both position and rotation distance, using $\lambda = 10$:

$$d(\mathbf{c}_1, \mathbf{c}_2) = \underbrace{\|\mathbf{t}_1 - \mathbf{t}_2\|_2}_{\text{position distance}} + \lambda \cdot \underbrace{(1 - |\mathbf{q}_1 \cdot \mathbf{q}_2|)}_{\text{rotation distance}} \quad (1)$$

For the feed-forward setting, we follow MVSplat360's [1] reference view selection, which uses farthest point sampling based solely on camera positions.

Table 6. **Diffusion-based NVS Comparison.** We evaluate performance on Mip-NeRF360 with reference-only diffusion methods.

Setting	Method	PSNR↑	SSIM↑	LPIPS↓
Dense	SEVA [8]	17.36	0.362	**0.360**
	Splatent (Ours)	**20.42**	**0.546**	0.364
Sparse	SEVA [8]	14.47	0.295	0.503
	NVC [3]	13.01	0.24	0.62
	Splatent (Ours)	**16.70**	**0.45**	**0.501**

0.6. Further quantitative results

Diffusion-based Comparison. We evaluate SOTA diffusion-based NVS on Mip-NeRF360 in dense and sparse settings. While reference-only methods hallucinate details, we rely on 3DGS which provides explicit 3D structure, constraining outputs to the actual scene.

This comparison highlights a key distinction: reference-only methods (SEVA, NVC) rely solely on diffusion priors, hallucinating details without geometric grounding. In contrast, relying on 3DGS provides explicit 3D structure, constraining outputs to the actual scene. This results in significantly higher PSNR/SSIM.

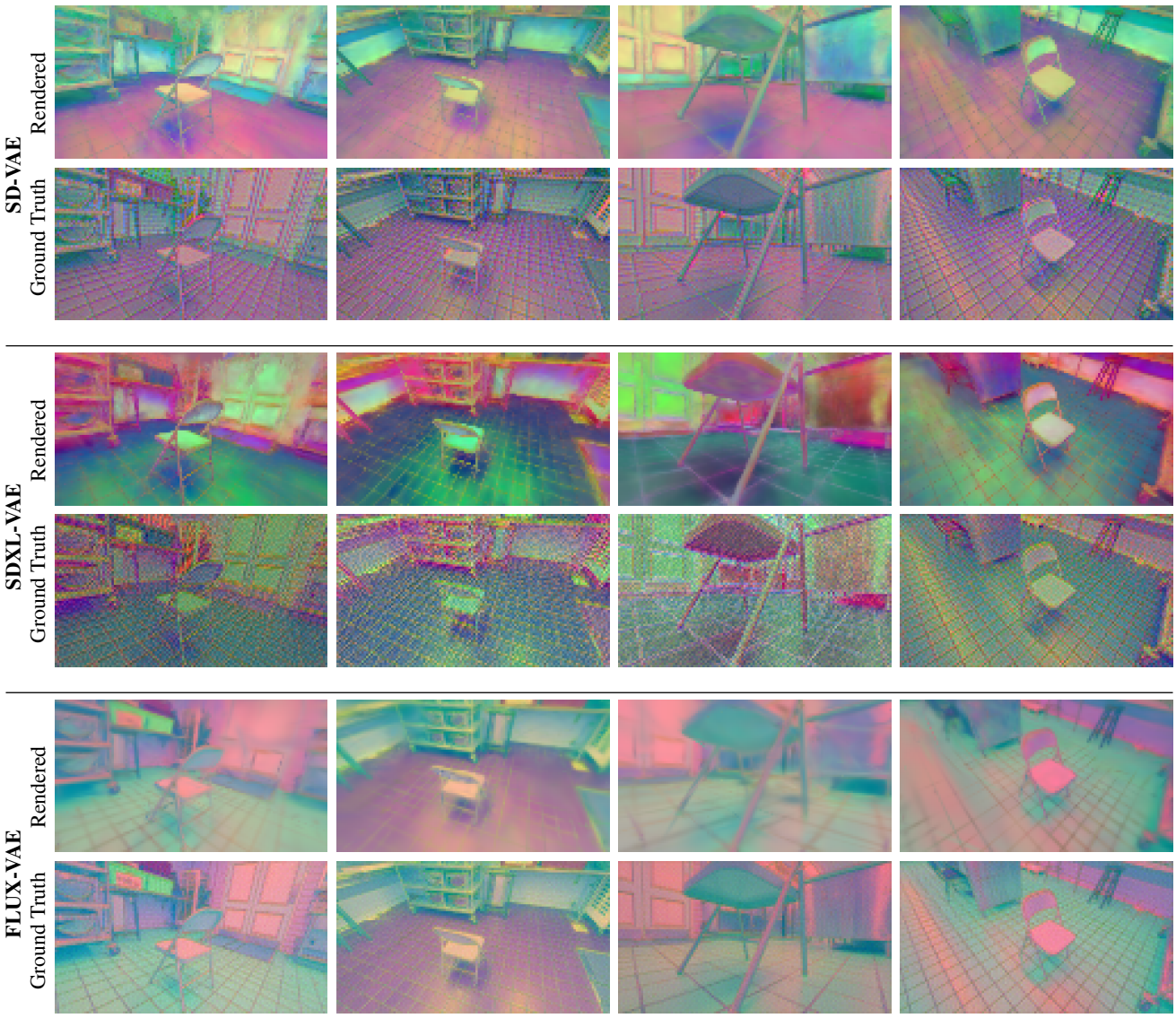

Figure 9. **Visualization of VAE latent inconsistencies across different architectures.** We visualize rendered latent 3D Gaussian splatting versus ground truth VAE-encoded latents across multiple viewpoints and VAE architectures. For each VAE model (SD, SDXL, FLUX), rendered latents consistently lose high-frequency information compared to ground truth latents, appearing smoother and lacking fine-grained texture. This visualization complements Fig. 6, demonstrating that the multi-view inconsistency problem is pervasive across different VAE architectures and manifests as visible loss of detail in the latent space.

Runtime and memory. Our method requires 6.3GB memory and runs in 330 ms (+50 ms decoder), compared to LRF which requires effectively only the decoder time. For MVSplat360 integration, we add 5 seconds to MVSplat360's 80 seconds. All measured on H100 GPU without optimization.

0.7. Further qualitative results

Splatent. We provide additional qualitative comparisons. Fig. 10 shows novel view synthesis results across diverse scenes with varying complexity, lighting conditions, and geometric structures. Consistent with our main findings, Feature-3DGS [9] produces blurry outputs lacking fine-grained details due to VAE latent inconsistencies, while LRF [7] achieves better reconstruction at the cost of losing high-frequency details. Our method consistently recovers sharp textures and fine geometric details by effectively injecting information from reference views while maintaining 3D consistency. These results further demonstrate the robustness and generalizability of our approach across different scene types and viewing conditions.

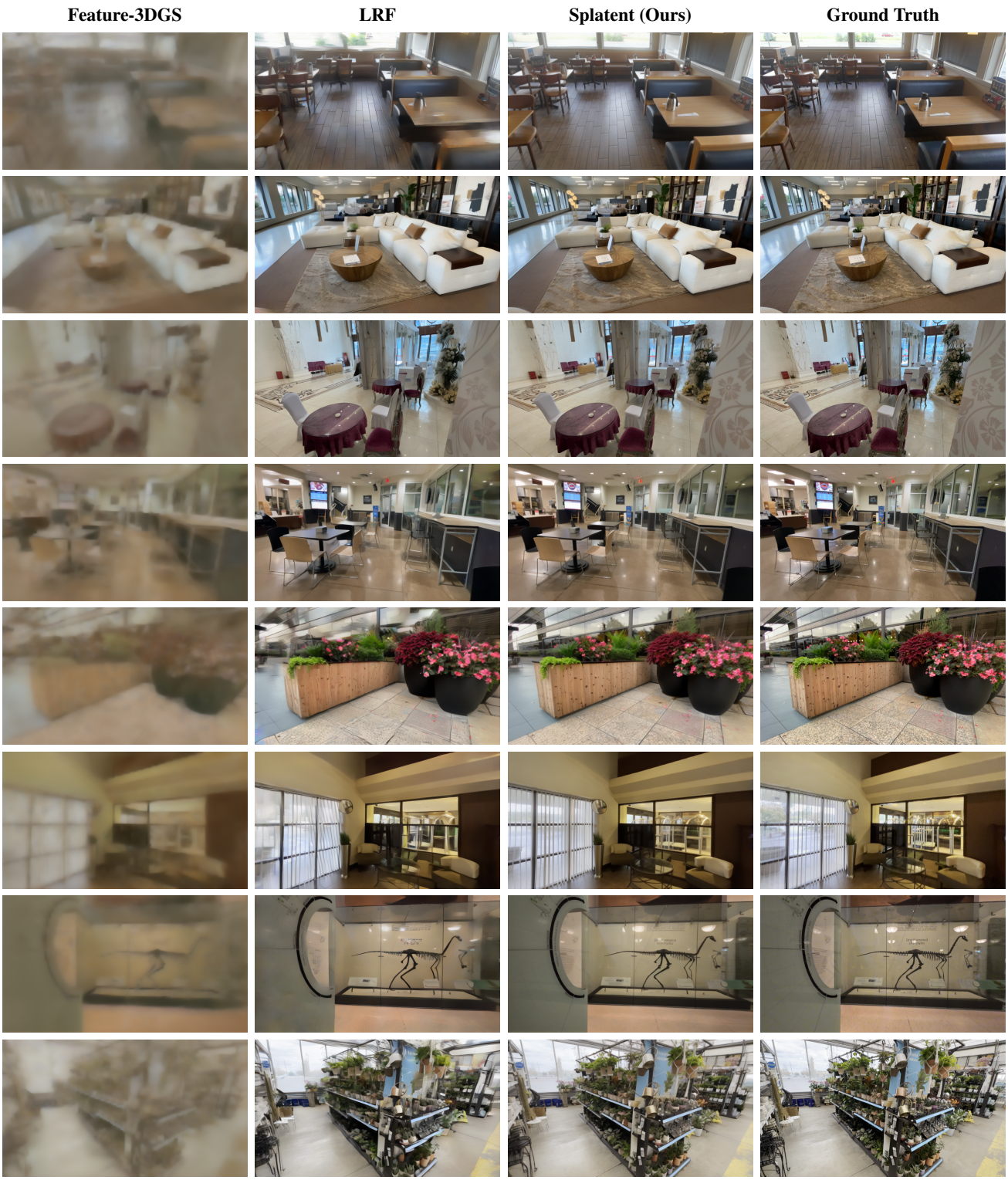

Figure 10. **Additional qualitative comparisons** We compare Splatent to other latent radiance field methods on novel view synthesis reconstruction quality. Feature-3DGS [9] exhibits considerable loss of detail, and LRF [7] improves upon this baseline but still fails to recover fine details. In contrast, Splatent produces sharper and more faithful reconstructions. The scenes are taken from the DL3DV-10K dataset.

Feed-forward integration. We further demonstrate that Splatent can enhance feed-forward latent radiance field methods, as shown in Fig. 11. MVSplat360 achieves satisfactory results but exhibits two key limitations from operating directly in inconsistent latent spaces: geometric hallucinations and lack of fine-grained textural details. By applying Splatent refinement to the rendered latents in MVSplat360, we recover sharp textures and eliminate geometric inconsistencies, yielding more faithful results. Importantly, this demonstrates the generalizability of our approach. Splatent can serve as a refinement module for various latent-based 3D reconstruction methods, enhancing their output quality while preserving their architectural advantages such as feed-forward efficiency.

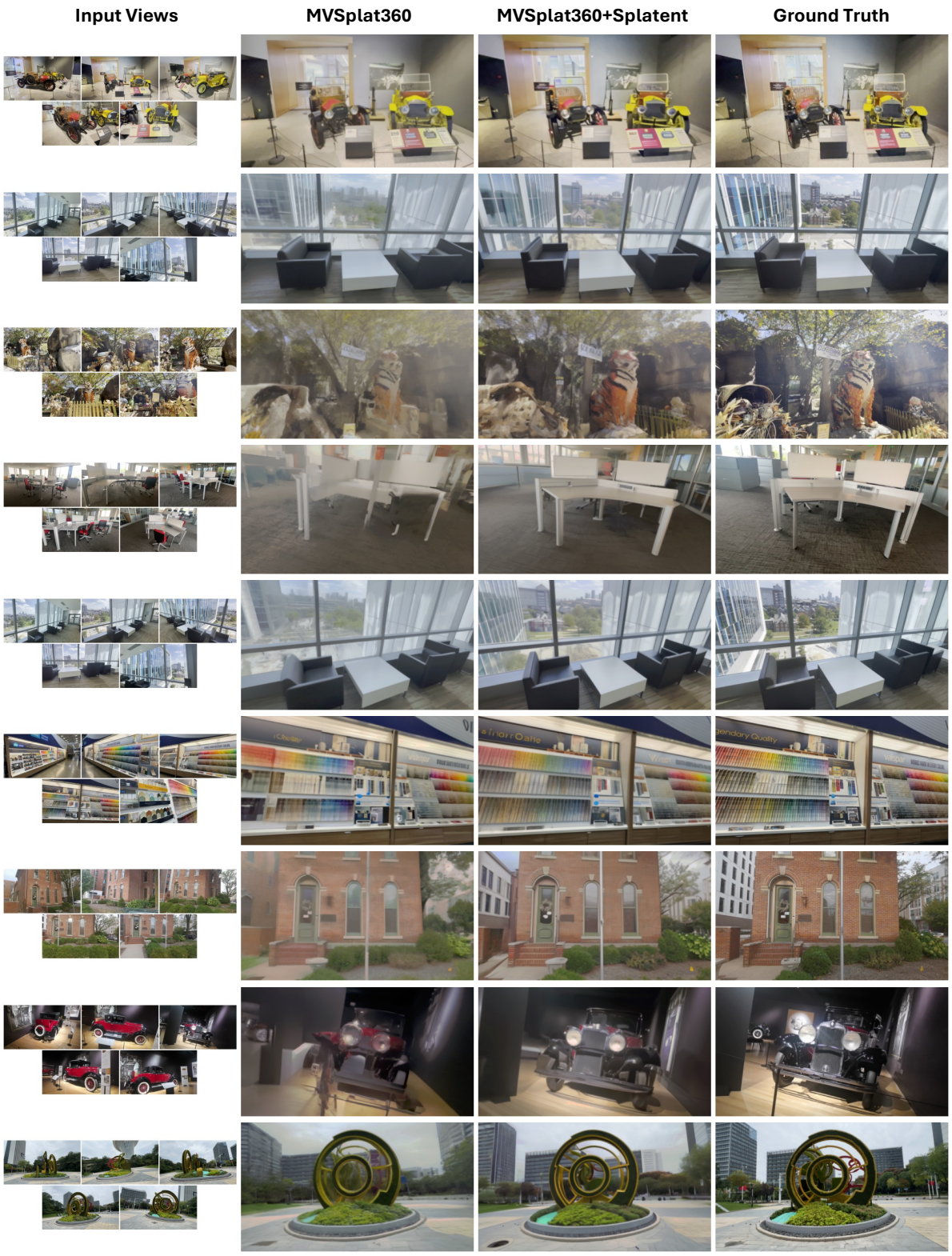

Figure 11. **Additional feed-forward qualitative comparison.** Our Splatent module improves feed-forward latent radiance field methods. MVSplat360 [1] produces hallucinations and lacks fine details. Splatent yields sharper, more faithful reconstructions while preserving the feed-forward efficiency.

References

[1] Yuedong Chen, Chuanxia Zheng, Haofei Xu, Bohan Zhuang, Andrea Vedaldi, Tat-Jen Cham, and Jianfei Cai. Mvsplat360: Feed-forward 360 scene synthesis from sparse views. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. 2, 3, 7

[2] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 kontext: Flow matching for in-context image generation and editing in latent space. *CoRR*, abs/2506.15742, 2025. 1

[3] Lingen Li, Zhaoyang Zhang, Yaowei Li, Jiale Xu, Wenbo Hu, Xiaoyu Li, Weihao Cheng, Jinwei Gu, Tianfan Xue, and Ying Shan. Nvcomposer: Boosting generative novel view synthesis with multiple sparse and unposed images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 777–787. Computer Vision Foundation / IEEE, 2025. 3

[4] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 1

[5] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXXVI*, pages 87–103. Springer, 2024. 1

[6] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 26024–26035. Computer Vision Foundation / IEEE, 2025. 1

[7] Chaoyi Zhou, Xi Liu, Feng Luo, and Siyu Huang. Latent radiance fields with 3d-aware 2d representations. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. 4, 5

[8] Jensen Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12405–12414, 2025. 3

[9] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 21676–21685. IEEE, 2024. 4, 5