DIFF4SPLAT: Repurposing Video Diffusion Models for Dynamic Scene Generation

Supplementary Material

1. Dataset Curation Details

We construct a collection of 130,000 diverse videos. A key challenge is obtaining metric scale for real-world datasets like RealEstate10K, which only provide relative camera poses from COLMAP. Our metric scale estimation procedure, addresses this by anchoring the relative depth predictions from foundation models to a sparse set of metric-scale 3D points. These anchor points are obtained by running Structure-from-Motion (SfM) and then manually scaling the reconstruction using known real-world information, such as the height of the camera or the size of objects in the scene for a small subset of scenes. This provides a robust mechanism for recovering metric scale, which is crucial for training a model with precise camera control. The error of this estimation is typically low, with the scale factor showing a variance of less than 5% across different subsets of anchor points.

2. More Implementation Settings

Reproducibility To facilitate reproducibility, we present our detailed experimental settings and evaluation metrics in Section 2.1. This section provides a comprehensive description of our implementation details. Moreover, **our source code, pre-trained models, and curated dataset will be publicly available upon publication.**

2.1. Video Transformer Denosing Details

Details of Model Inputs The model is conditioned on a single source image and a predefined camera motion trajectory, such as spiral, forward, backward, upward, or downward. Accompanying this, a textual prompt is provided, which can either be automatically generated from the source image using a Multimodal Large Language Model (MLLM) [1] or set to a generic high-fidelity description, for instance, "a scene with 4K ultra HD, surround motion, realistic tone, panoramic shot, wide-angle view, cinematic quality".

Classifier-Free Guidance **C**lassifier-**F**ree **G**uidance (**CFG**) has emerged as a prevalent technique for balancing controllability and sample diversity in diffusion models. However, we observe that its uniform scaling mechanism inadvertently introduces "over-sharpening artifacts" in the final frames of generated orbital sequences. To mitigate this limitation, we introduce a cosine-based dynamic guidance schedule during the sampling of validation videos, formulated as:

$$\gamma(t) = 1 + \gamma_{\text{max}} \cdot \left(\frac{1 - \cos\left(\pi \left(\frac{N-t}{N}\right)^5\right)}{2} \right) \tag{1}$$

where γ_{max} denotes the maximum guidance scale, N represents the total number of inference steps, and t is the current timestep. This adaptive scheduling progressively reduces guidance intensity in later denoising stages, effectively preserving temporal consistency while maintaining sample fidelity. In our experiments, we set the total number of inference steps $N = 30$ and the maximum guidance scale $\gamma_{\text{max}} = 7.5$.

2.2. Deformation Field Generation

To predict the per-Gaussian deformations, our **LDRM** employs a lightweight spatio-temporal network. The network takes as input a latent representation of the scene at a canonical time step, conditioned on a time embedding for the target frame t. The architecture extracts features at multiple spatial resolutions to effectively capture both local and global motion patterns. The final layer of the network is a convolutional layer with a kernel size of 1×1, which projects the high-dimensional features into the final deformation map $\mathcal{D}$. This map has a dimensionality of $K_d = 10$ channels, which directly correspond to the predicted mean displacement (3 channels), rotational delta quaternion (4 channels), and scaling adjustment (3 channels) for each Gaussian primitive. No activation function is applied to the output layers for displacement and scale, allowing for unbounded predictions. The output quaternion components are normalized to ensure they represent a valid rotation.

2.3. Details of Progressive Training Scheme.

Our progressive training scheme's efficacy in decoupling static and dynamic scene components is empirically validated. Initially, the model trains exclusively on static scenes, learning to predict an **identity deformation**. In this stage, positional and scaling offsets ($\Delta\boldsymbol{\mu}_p^t, \Delta\boldsymbol{s}_p^t$) converge to zero, and rotational deformations ($\Delta\boldsymbol{q}_p^t$) approach the identity quaternion, yielding a static representation as canonical Gaussians remain untransformed. Dynamic scenes are introduced in a subsequent fine-tuning stage. This decoupling is enabled by our Gaussian deformation formulation:

$$\boldsymbol{\mu}_p^t := \boldsymbol{\mu}_p^0 + \Delta\boldsymbol{\mu}_p^t, \quad \boldsymbol{q}_p^t := \boldsymbol{q}_p^0 \otimes \Delta\boldsymbol{q}_p^t, \quad \boldsymbol{s}_p^t := \boldsymbol{s}_p^0 + \Delta\boldsymbol{s}_p^t. \tag{2}$$

Table 1. **Training Datasets Statistics.** Overview of the datasets used for training DIFF4SPLAT at scale, highlighting their dynamic nature, multi-camera setups, depth annotations, tracking capabilities, and real-world applicability.

Dataset	*Dynamic?*	*Multi-camera?*	*Depth?*	*Tracking?*	*Real?*	*#Scenes*	*#Frames*
TartanAir [14]	✗	✗	✓	✗	✗	0.4K	0.49M
MatrixCity [7]	✗	✗	✓	✗	✗	4.5K	0.31M
RealEstate10K [19]	✗	✗	✗	✗	✓	70K	6.36M
PointOdyssey [18]	✓	✗	✓	✓	✗	0.1K	0.18M
DynamicReplica [5]	✓	✗	✓	✓	✗	0.5K	0.26M
Spring [8]	✓	✗	✓	✗	✗	0.03K	0.003M
VKITTI2 [3]	✓	✗	✓	✗	✗	0.1K	0.03M
MultiCamVideo [2]	✓	✓	✗	✗	✗	14K	11M
Stereo4D [4]	✓	✗	✓	✓	✓	74K	14.8M

This design inherently separates the prediction of the canonical scene structure $(\boldsymbol{\mu}_p^0, \boldsymbol{q}_p^0, \boldsymbol{s}_p^0)$ from its temporal evolution $(\Delta\boldsymbol{\mu}_p^t, \Delta\boldsymbol{q}_p^t, \Delta\boldsymbol{s}_p^t)$.

2.4. Details of loss function weighting

The loss weights ($\lambda_p = 0.5$, $\lambda_m = 2$) were determined empirically through a series of experiments on a validation set. We started with equal weights and adjusted them to ensure that the model did not prioritize one objective at the expense of others.

3. Evaluation Protocol

To comprehensively evaluate our model, we utilize a suite of established metrics, Specifically:

❶ **Fréchet Video Distance (FVD) and Kernel Video Distance (KVD)** [13]: These metrics evaluate the quality and temporal coherence of generated videos by measuring the distance between the feature distributions of real and generated video sets. Lower scores for both FVD and KVD indicate higher fidelity and better temporal consistency.

❷ **CLIP-Score** [10]: This metric quantifies the semantic similarity between the generated video frames and the input text prompt. It leverages the joint text-image embedding space of the CLIP model, where higher scores signify better alignment between the generated content and the textual description.

❸ **CLIP-Aesthetic** [11]: We use a model built upon CLIP embeddings to predict the aesthetic quality of the generated content. This model is trained on datasets with human aesthetic ratings, and a higher score suggests a more visually pleasing result.

❹ **QA-Quality** [15]: This refers to a Visual Question Answering (VQA)-based evaluation, where a LLaMA2-powered model is employed to assess the logical consistency and objective quality of the generated scenes. The model assigns a score on a range from 0 to 5, where a higher score indicates superior quality.

Table 2. **Capability Comparison.** An explicit 4D representation enables a wide range of functionalities not supported by standard 2D video generation models.

Capability	AC3D (Implicit 3D Models)	Ours (Explicit 4D Repr.)
Novel View Synthesis	✓	✓
Depth Rasterization	✗	✓
Geometry Extraction	✗	✓
Real-time Interaction	✗	✓
Interactive exploration Latency ↓	28000 ms	**6.7 ms** (↓ 99.98%)
Avg. Matches ↑	2489.16	**5114.22** (↑ 105.5%)
Subject Consistency Score ↑	75.64	**88.32** (↑ 16.8%)
Background Consistency Score ↑	75.91	**89.89** (↑ 18.4%)
Cycle-Consistency ↑	20.68 dB	**34.5 dB** (↑ 66.8%)

❺ **Temporal Consistency Metrics (Avg. Matches, Subject Cons. and Bg. Cons.)**: Inspired by Video-bench [9], to specifically measure temporal stability, we use metrics based on dense optical flow or feature matching. Avg. Matches quantifies overall frame-to-frame consistency. Subject Consistency Score and Background Consistency Score measure the stability of the foreground subject and the background, respectively, after performing segmentation. Higher values for these metrics indicate smoother and more coherent videos.

4. More Results

As demonstrated in Table 2, and inspired by prior work such as CAT4D [17], an explicit 3D representation is a critical advantage for applications that demand a concrete understanding of and interaction with the world, including robotics and AR/VR.

Furthermore, 4D consistency is ensured by a training objective calculated from rendering the deformed 3D Gaussian representation from multiple viewpoints and at various timestamps. As shown in Table 2, we generate videos depicting a full 360-degree camera rotation. The resulting scenes exhibit seamless looping, where the final frame aligns perfectly with the first, showing no discernible seams or drift. We quantitatively verify this strong temporal consistency by measuring the similarity between the first and last frames (a.k.a., Cycle-Consistency), achieving a PSNR of 34.5 dB.

5. Feed-forward vs. per-scene optimization

Existing methods that produce explicit 3D outputs, rely on a time-consuming, post-hoc optimization process to reconstruct scenes from generated videos. For instance, DimensionX [12] requires **1.3K GPU hours** to perform scene optimization from a single video. Even state-of-the-art 4D reconstruction algorithms like Mosca [6] require approximately **0.5 hours** to process one input video. The primary motivation of this work is therefore to unify these disparate stages into a single, efficient, feed-forward framework capable of generating a 4D representation in approximately **30 seconds**, achieving 60× acceleration. Our model is designed for efficiency and scalability, enabling dynamic scene reconstruction in a matter of seconds, which is a critical feature for many real-world applications where speed is essential.

Compared to per-scene optimization methods, our proposed approach achieves a substantial reduction in memory consumption during the reconstruction process, decreasing from 80GB to 25GB (a 3.2× reduction) in the same setting. This efficiency gain stems from the elimination of gradient computation requirements. Furthermore, we claim that the two approaches are not mutually exclusive. As explored in recent work like CAT4D [16], efficient, end-to-end models can serve as an excellent initialization for optimization-based methods, significantly accelerating their convergence. This potential synergy further highlights that developing fast, feed-forward models is a valuable research direction.

In summary, considering both the reconstruction and rendering stages (e.g., maximum GPU memory), our approach remains competitive in terms of memory consumption compared to per-scene optimization methods.

6. Additional Ablation Studies

Impact of Gaussian Budget We studied the effect of the number of canonical Gaussians, M, on performance and resource consumption. We tested $M \in \{10K, 50K, 100K\}$. While $100K$ Gaussians offered a marginal improvement in fine details, it increased memory consumption by 75% and inference time by 60%. We found that $M = 50K$ provides the best trade-off between quality and efficiency, as it is sufficient to represent the scenes in our dataset without excessive resource usage.

Impact of Geometric Loss We performed an ablation on the geometric loss components. Removing the geometric loss entirely leads to a noticeable degradation in 3D consistency. Using only the covariance-based term works well, but adding the total variation loss $\mathcal{L}_{TV}$ on the depth maps helps to regularize the geometry and produces smoother surfaces, improving the final visual quality.

Impact of Pose Encoding We experimented with alternative camera pose encodings instead of Plücker embeddings, such as representing the pose as a 12-dimensional vector of the flattened rotation matrix and translation vector. We found that Plücker embeddings provided a 5-10% reduction in relative pose error, likely because they represent 3D lines in a more geometrically natural way, which is beneficial for the ray-based operations implicit in rendering.

Joint vs. Separate Training. We compared our joint training strategy against a "Frozen LDRM" baseline (Tab. 3). Joint training yields a significant improvement in FVD/KVD and consistency metrics. This is because joint training allows the camera-conditioned diffusion model to learn to generate content that is explicitly *compatible* with the LDRM's geometric decoder, effectively avoiding artifacts that a fixed, pre-trained decoder often fails to handle. **(3) Comparison with Two-Stage Baseline.** We implemented the suggested "generate-video-then-reconstruct" baseline: generating a video with our finetuned DiT, followed by a SOTA feed-forward 4DGS reconstructor (4DGT). As shown in Tab. 3, our unified approach significantly outperforms this two-stage baseline in temporal consistency (FVD **210.2** vs 245.3) and efficiency, as our latent reconstruction strategy bypasses the costly overhead of decoding to pixel space. The two-stage method suffers from multi-view inconsistencies in the generated frames, which leads to failure in the downstream reconstruction step.

Table 3. **We benchmark our full model against a two-stage baseline and separate training, reporting FVD/KVD and foreground/background consistency.**

Method	FVD ↓	KVD ↓	Subj. Consist. ↑	Bg. Consist. ↑
Two-Stage (Gen. Video → 4DGT Recon.)	245.3	3.42	0.82	0.83
Ours (Frozen LDRM + Train Diffusion)	231.5	3.10	0.85	0.86
Ours (Joint Training)	**210.2**	**2.32**	**0.88**	**0.90**

7. Limitations

While our method achieves superior performance and efficiency, video generation remains the computational bottleneck. This could be addressed through parallel inference or optimized denoising strategies. Future work will focus on extending temporal coherence modeling and material property prediction.

8. Broader Impact

While DIFF4SPLAT advances 4D generation from single images, it risks misuse for creating deceptive content ("deepfakes"). To ensure responsible deployment, implementing safeguards like watermarking is essential. We aim to advance 3D computer vision and encourage the community to adopt best practices for ethical use.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 1

[2] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, and Di Zhang. Recammaster: Camera-controlled generative rendering from a single video. *arXiv preprint arXiv:2503.11647*, 2025. 2

[3] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 2

[4] Linyi Jin, Richard Tucker, Zhengqi Li, David Fouhey, Noah Snavely, and Aleksander Holynski. Stereo4d: Learning how things move in 3d from internet stereo videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

[5] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[6] Jiahui Lei, Yijia Weng, Adam Harley, Leonidas Guibas, and Kostas Daniilidis. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. *arXiv preprint arXiv:2405.17421*, 2024. 3

[7] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2

[8] Lukas Mehl, Jenny Schmalfuss, Azin Jahedi, Yaroslava Nalivayko, and Andrés Bruhn. Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4981–4991, 2023. 2

[9] Munan Ning, Bin Zhu, Yujia Xie, Bin Lin, Jiaxi Cui, Lu Yuan, Dongdong Chen, and Li Yuan. Video-bench: A comprehensive benchmark and toolkit for evaluating video-based large language models. *arXiv preprint arXiv:2311.16103*, 2023. 2

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[11] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 2022. 2

[12] Wenqiang Sun, Shuo Chen, Fangfu Liu, Zilong Chen, Yueqi Duan, Jun Zhang, and Yikai Wang. Dimensionx: Create any 3d and 4d scenes from a single image with controllable video diffusion. *arXiv preprint arXiv:2411.04928*, 2024. 3

[13] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 2

[14] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *International Conference on Intelligent Robots and Systems (IROS)*, 2020. 2

[15] Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Chunyi Li, Liang Liao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtai Zhai, and Weisi Lin. Q-align: Teaching lmms for visual scoring via discrete text-defined levels. *arXiv preprint arXiv:2312.17090*, 2023. 2

[16] Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. *arXiv preprint arXiv:2411.18613*, 2024. 3

[17] Weiwei Zhang, Jian Sun, and Xiaoou Tang. Cat head detection-how to effectively exploit shape and texture features. In *European conference on computer vision*, 2008. 2

[18] Yang Zheng, Adam W Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2

[19] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *Transactions on Graphics (TOG)*, 2018. 2