LocalDyGS: Multi-view Global Dynamic Scene Modeling via Adaptive Local Implicit Feature Decoupling

Supplementary Material

A. Overview

The supplementary content is organized as follows: Section B outlines the implementation details of our paper, including the workflow, the deactivation of Temporal Gaussians, the choice of activation functions for each parameter, and the method for extracting dynamic objects. Section C presents additional experimental data and results.

B. Implementation details

B.1. Implementation Workflow.

Our algorithm is divided into two main parts: **initialization of the local space** and **the specific rendering process**. The detailed steps are presented in the Algorithm 1 and 2.

In the first stage, we fuse and downsample the Colmap point cloud generated from N frames of images, resulting in a point cloud with a certain level of motion awareness. We assign new parameters, namely *scale* and *static feature*, to this point cloud, initializing each seed and its local space.

In the second stage, the static features of each local space provide static information, while time-varying dynamic residual features represent motion changes within the local space at each moment. A weighted sum of these two features is passed through a specific MLP to decode the attributes of the Temporal Gaussians. Finally, all active Temporal Gaussians generated by each local space are rasterized to produce the rendered image $\mathcal{I}_t$ at the query time.

It is worth noting that, unlike previous methods that model global motion by mapping points from canonical space to deformed spaces at each time step via spatiotemporal structures [14, 15, 18] or by using Fourier techniques and polynomial functions to model point trajectories [3], our method achieves global dynamic modeling by **representing motion within local spaces** using time-varying Temporal Gaussians.

Algorithm 1: Local Space Initialization

Input: Temporal multi-view N frames of images $\mathcal{I}_{i=0}^{N}$
Output: The center, scale and static feature set of local space **C, S, F**$_s$;
Initialization: $\mathbf{F}_s = 0$; **C, S** = $\{\phi\}$; $\mathbf{P} = \{\phi\}$.
for $\mathcal{I}_k$ *in* $\mathcal{I}_{i=0}^{N}$ **do**
  $\mathbf{P}_k = \text{Colmap}(\mathcal{I}_k)$
  $\mathbf{P} = \mathbf{P} + \mathbf{P}_k$
end
P = DownSample(**P**)
C = GetCenter(**P**)
$\mathbf{S} = \text{KNN}(\mathbf{P}, 3)$
return ***C,S,F***$_s$.

Algorithm 2: Rendering Process

Input: The center, scale and static feature set of local space **C, S, F**$_s$; querry time t; view v
Output: Rendered image I_t at time t
Temperal Gaussian set **TG** = $\{\phi\}$
for c^i, s^i, f_s^i *in* ***C, S, F***$_s$ **do**
  f_d^i = DynamicResidualField(c^i, t)
  w_s^i, w_d^i = WeightField (c^i, t)
  $f_w^i = w_s^i \cdot f_s^i + w_d^i \cdot f_d^i$
  $\mathbf{TG}^i = s^i \cdot \text{MLPs}(f_w, v)$
  TG = **TG** + **TG**^{i}
end
TG = Deactivation(**TG**)
I_t = Splatting(**TG**,v)
return I_t.

B.2. Activation functions

We use separate MLPs to predict each parameter of the Temporal Gaussian at each time step. Each MLP is implemented with two linear layers of dimension 128 and uses ReLU as the activation function. Different activation functions are employed for the outputs: a Sigmoid activation function is used for color and opacity; the normalization activation method from [4] is applied for rotation; and the scaling approach follows the technique outlined in [8].

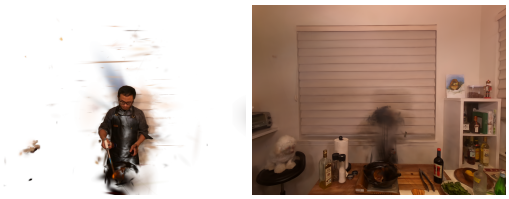
B.3. The difference with some current method.

First, initializing with SfM point clouds from multi-frame images was first introduced by SpaceTimeGS. However, they initialize these points as Gaussian points and model their motion using polynomials. In contrast, we initialize these points as seed points, which generate Gaussian points within their local space, effectively leveraging the geometric advantages of multi-frame point cloud initialization.

Second, while our method draws some inspiration from NeRFPlayer, it is fundamentally different. NeRFPlayer relies on a NeRF-based deformation field, which often struggles with large-scale motion. In contrast, our approach is GS-based and deviates from the conventional deformation field paradigm. Instead, we introduce two novel strategies:

Table 1. **Per-scenes PSNR results on the N3DV dataset [5].** The best and the second best results are denoted by orange and orange.

Method	Coffee Martini	Spinach	Cut Beef	Flame Salmon	Flame Steak	Sear Steak	Mean
MixVoxels [12]	29.36	31.61	31.30	29.92	31.21	31.43	30.80
HexPlane [1]	–	32.04	32.55	29.47	32.08	32.39	31.70
K-Planes [2]	29.99	32.60	31.82	30.44	32.38	32.52	31.63
4DGS [14]	27.34	32.46	32.90	29.20	32.51	32.49	31.15
3DGStream [9]	27.75	33.31	33.21	28.42	34.30	33.01	31.67
SpaceTimeGS [6]	28.61	33.18	33.52	29.48	33.64	33.89	32.05
Real-Time4DGS [17]	28.33	32.93	33.85	29.38	34.03	33.51	32.01
LocalDyGS(Ours)	29.03	33.31	33.67	29.82	34.09	33.77	32.28

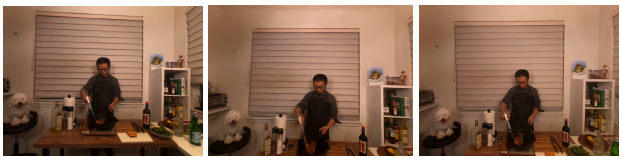


(a) Dynamic Rendering (b) Static Rendering

Figure 1. Our method can achieve dynamic-static separation without explicit supervision. (a) and (b) show the rendering results of dynamic and static Temporal Gaussians, respectively.

(a) Seeds (b) Temporal Gaussians (c) Rendering

Figure 2. (a), (b), and (c) respectively show the seeds, the temporal Gaussians, and the final rendered images.



(a) Gaussian-Flow [3] (b) SPGS [11] (c) Ours

Figure 3. (a), (b), and (c) respectively show the rendering results of Gaussian-Flow,SPGS,Ours.

(1) decomposing the global space into local spaces and (2) generating Temporal-aware Gaussians to model motion within each local space. Meanwhile, our static and dynamic residual features have distinct physical meanings compared to previous works, as shown in Fig. 4 of the main paper.

Experimentally, our method surpasses NeRFPlayer across multiple aspects, including training efficiency, rendering quality, inference speed, and the ability to handle complex scenes.

B.4. Dynamic-static decomposition

Based on this decoupling approach, our method can achieve the separation of dynamic and static seeds without auxiliary information or explicit supervision, as shown in Fig. 1. Additionally, we provide videos (5_Extract_dynamic) in the supplementary materials.

C. Experiments and Results

Table 1 provides a detailed comparison of the results for each scene in the N3DV dataset, while Figure 4 displays all the rendering results rendered by our method. To demonstrate the robustness and generalization of our approach, we also conducted experiments on the ENeRF dataset. The results, shown in Table 2, follow the training policies described in 4k4d [16].

Fig. 2 illustrates the core rendering primitives and the rendered images produced by our method, showcasing both the seed points and the temporal Gaussians.

In Fig. 3, we present a comparison of the rendering quality of our method against other approaches, including the latest state-of-the-art methods, Gaussian-Flow [3] and SPGS [11].

In Fig. 6, we compare our method with the online method 3DGStream on the ENeRF dataset. Our method demonstrates superior subjective performance as well.

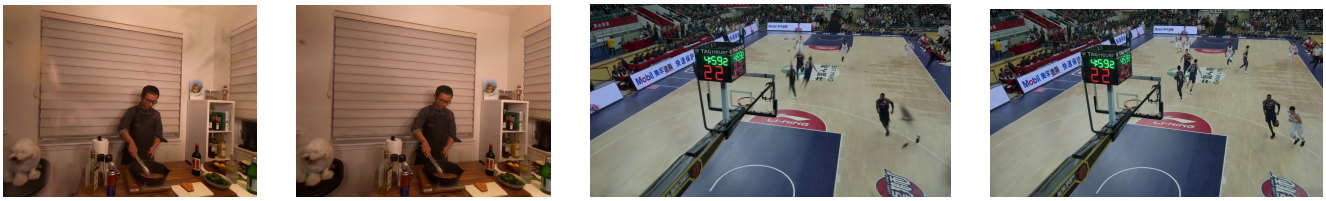
In Fig. 5, we compare our method with 4DGS. The results demonstrate that our local modeling approach produces cleaner renderings with fewer floaters. Additionally, for the more challenging VRU dataset, our method captures motion more faithfully compared to 4DGS, effectively avoiding issues such as the disappearance of players.

To better showcase the effectiveness of our model, we provide several videos for demonstration and comparison.

Table 2. Performance comparison of different methods on ENeRF dataset. The results are derived from 4k4d.

Methods	**PSNR** ↑	**SSIM** ↑	**LPIPS** ↓
ENeRF [7]	25.452	0.809	0.273
IBRNet [13]	24.966	0.929	0.172
KPlanes [2]	21.310	0.735	0.454
4k4d [16]	25.815	0.898	0.147
LocalDyGS(Ours)	26.230	0.923	0.065

Figure 4. The rendering results of our method on the N3DV dataset [5].



(a) 4DGS (b) Ours (c) 4DGS (d) Ours

Figure 5. A comparison between 4DGS [14] and our method on the N3DV [5] and VRU Basketball [10] datasets.

(a) GT (b) Ours (c) 3DGSTream [9]

Figure 6. Qualitative results of *actor 2_3* from the E-NeRF dataset [7].

References

[1] Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 130–141, 2023. 2

[2] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023. 2

[3] Quankai Gao, Qiangeng Xu, Zhe Cao, Ben Mildenhall, Wenchao Ma, Le Chen, Danhang Tang, and Ulrich Neumann. Gaussianflow: Splatting gaussian dynamics for 4d content creation. *arXiv preprint arXiv:2403.12365*, 2024. 1, 2

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

[5] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5521–5531, 2022. 2, 3

[6] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8508–8520, 2024. 2

[7] Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 2, 3

[8] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians

for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20654–20664, 2024. 1

[9] Jiakai Sun, Han Jiao, Guangyuan Li, Zhanjie Zhang, Lei Zhao, and Wei Xing. 3dgstream: On-the-fly training of 3d gaussians for efficient streaming of photo-realistic free-viewpoint videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20675–20685, 2024. 2, 3

[10] VRU. https://anonymous.4open.science/r/vru-sequence/. 2024. 3

[11] Diwen Wan, Ruijie Lu, and Gang Zeng. Superpoint gaussian splatting for real-time high-fidelity dynamic scene reconstruction. *arXiv preprint arXiv:2406.03697*, 2024. 2

[12] Feng Wang, Sinan Tan, Xinghang Li, Zeyue Tian, Yafei Song, and Huaping Liu. Mixed neural voxels for fast multi-view video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19706–19716, 2023. 2

[13] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2021. 2

[14] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20310–20320, 2024. 1, 2, 3

[15] Jiawei Xu, Zexin Fan, Jian Yang, and Jin Xie. Grid4d: 4d decomposed hash encoding for high-fidelity dynamic gaussian splatting. *arXiv preprint arXiv:2410.20815*, 2024. 1

[16] Zhen Xu, Sida Peng, Haotong Lin, Guangzhao He, Jiaming Sun, Yujun Shen, Hujun Bao, and Xiaowei Zhou. 4k4d: Real-time 4d view synthesis at 4k resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20029–20040, 2024. 2

[17] Zeyu Yang, Hongye Yang, Zijie Pan, Xiatian Zhu, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023. 2

[18] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20331–20341, 2024. 1