Adaptive and Temporally Consistent Gaussian Surfels for Multi-view Dynamic Reconstruction
–Supplementary Material–

Decai Chen[1,2] Brianne Oberson[1,3] Ingo Feldmann[1]
Oliver Schreer[1] Anna Hilsmann[1] Peter Eisert[1,2]
[1]Fraunhofer HHI [2]Humboldt University of Berlin [3]Technical University of Berlin
{first}.{last}@hhi.fraunhofer.de

1. Implementation Details

In all our experiments, training is conducted on a GPU server equipped with an AMD EPYC 9654 CPU and an NVIDIA RTX 6000 Ada GPU, utilizing the Adam optimizer [4], PyTorch 2.3.1 [5], and CUDA 11.8. For each dynamic scene, we begin with static reconstruction using Gaussian surfels [2] for the first frame, obtaining a surfel-based Gaussian representation from a sparse point cloud generated by COLMAP [6]. For each subsequent frame, we initialize the scene from the previous frame and apply our coarse-to-fine training approach, with 200 iterations for the coarse stage and 800 iterations for the fine stage. Training takes 31.7 seconds per frame on the NHR dataset [8] and 37.5 seconds per frame on the DNA-Rendering dataset [1].

In the coarse stage, the learning rate for the Neural Transformation Cache is set to 0.002. In the fine stage, our unified, adaptive densification of Gaussians starts at iteration 230 and ends at iteration 600, with a densification interval of 30 iterations. Additionally, the Gaussian opacity reset interval is set to 200 iterations. We set the spherical harmonics degree to 1 for the NHR dataset and 2 for the DNA-Rendering dataset, as the latter contains more non-Lambertian objects. All other hyperparameters are kept consistent with 3DGS [3].

For the loss function, we set λ_o to 0.01 and λ_m to 0.1. Additionally, we gradually increase λ_m from 0.01 to 0.11, while linearly decaying λ_t from 0.04 to 0.02.

2. Additional Dataset Details

For the DNA-Rendering dataset [1], we evaluate our method on five widely used sequences: 0008_01, 0012_11, 0013_01, 0013_03, and 0013_09, with images downsampled by a factor of 2 and cropped to focus on the foreground region. Following 4K4D [9], we select views 11, 25, 37, and 57 as testing views, with the remaining views used for training. For all scenes in the NHR dataset [8], we reserve views 18, 28, 37, and 46 for evaluation, while the rest serve as the training set.

3. Additional Ablation Study

In this section, we quantitatively evaluate the effectiveness of our method in enhancing temporal consistency. Specifically, we render dynamic mesh sequences from a fixed testing view and calculate SSIM, PSNR, and LPIPS between consecutive frames. Temporal consistency is then measured by averaging these metrics across the entire sequence, with higher scores indicating greater similarity between consecutive frames. Since the scene movement remains consistent for the same rendering view, more similar images across frames suggest higher temporal consistency. As shown in Tab. 1, our curvature-based temporal consistency (TC) module significantly improves smoothness across frames. Additionally, a qualitative evaluation of temporal consistency is provided in the supplementary video.

Method	PSNR↑	SSIM↑	LPIPS↓
w/o GD + w/o TC	29.268	0.946	0.0145
w/o GD	29.569	0.9507	0.0129
w/o TC	29.271	0.9469	0.0145
Ours Full	29.589	0.9514	0.0129

Table 1. Ablation study on the temporal consistency of rendered mesh videos on the NHR dataset.

4. More Results

Free-Viewpoint Rendering. In Tab. 2 and Tab. 3, we provide a detailed per-scene quantitative comparison of our rendering results against various baselines on both the DNA-Rendering and NHR datasets. Additionally, as shown in Fig. 1, our method consistently achieves photo-realistic rendering with fine-grained details.

Surface Reconstruction. We include further qualitative comparisons of dynamic surface geometry on the DNA-Rendering and NHR datasets in Fig. 2. Our method reconstructs high-quality surface meshes across various complex dynamic scenes.

5. Supplementary Video

The supplementary video includes the following:

- Additional ablation study on the impact of temporal consistency loss on dynamic surface meshes.
- A comparison between our method and NeuS2 [7] on dynamic surface meshes.
- Additional results showcasing free-viewpoint renderings of both color images and surface meshes.

6. Potential Societal Impact

While AT-GS advances dynamic surface reconstruction and novel view synthesis, its deployment carries potential negative societal impacts. When combined with generative technology, it could be misused to create hyper-realistic deepfakes or synthetic media, leading to disinformation, privacy breaches, and security risks. The high-fidelity reconstruction capabilities may also be exploited for intrusive surveillance, further raising privacy concerns. Additionally, although more efficient than some methods, the computational demands of AT-GS could contribute to environmental impact due to energy consumption, especially at scale. It is essential for researchers to remain vigilant and prioritize ethical use, alongside exploring safeguards to mitigate these risks.

References

[1] Wei Cheng, Ruixiang Chen, Wanqi Yin, Siming Fan, Keyu Chen, Honglin He, Huiwen Luo, Zhongang Cai, Jingbo Wang, Yang Gao, Zhengming Yu, Zhengyu Lin, Daxuan Ren, Lei Yang, Ziwei Liu, Chen Change Loy, Chen Qian, Wayne Wu, Dahua Lin, Bo Dai, and Kwan-Yee Lin. Dna-rendering: A diverse neural actor repository for high-fidelity human-centric rendering. *arXiv preprint*, arXiv:2307.10173, 2023. 1, 5

[2] Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality surface reconstruction using gaussian surfels. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 1

[3] 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

[4] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014. 1

[5] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 1

[6] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[7] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3295–3306, 2023. 2

[8] Minye Wu, Yuehao Wang, Qiang Hu, and Jingyi Yu. Multi-view neural human rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1682–1691, 2020. 1, 5

[9] 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. 1

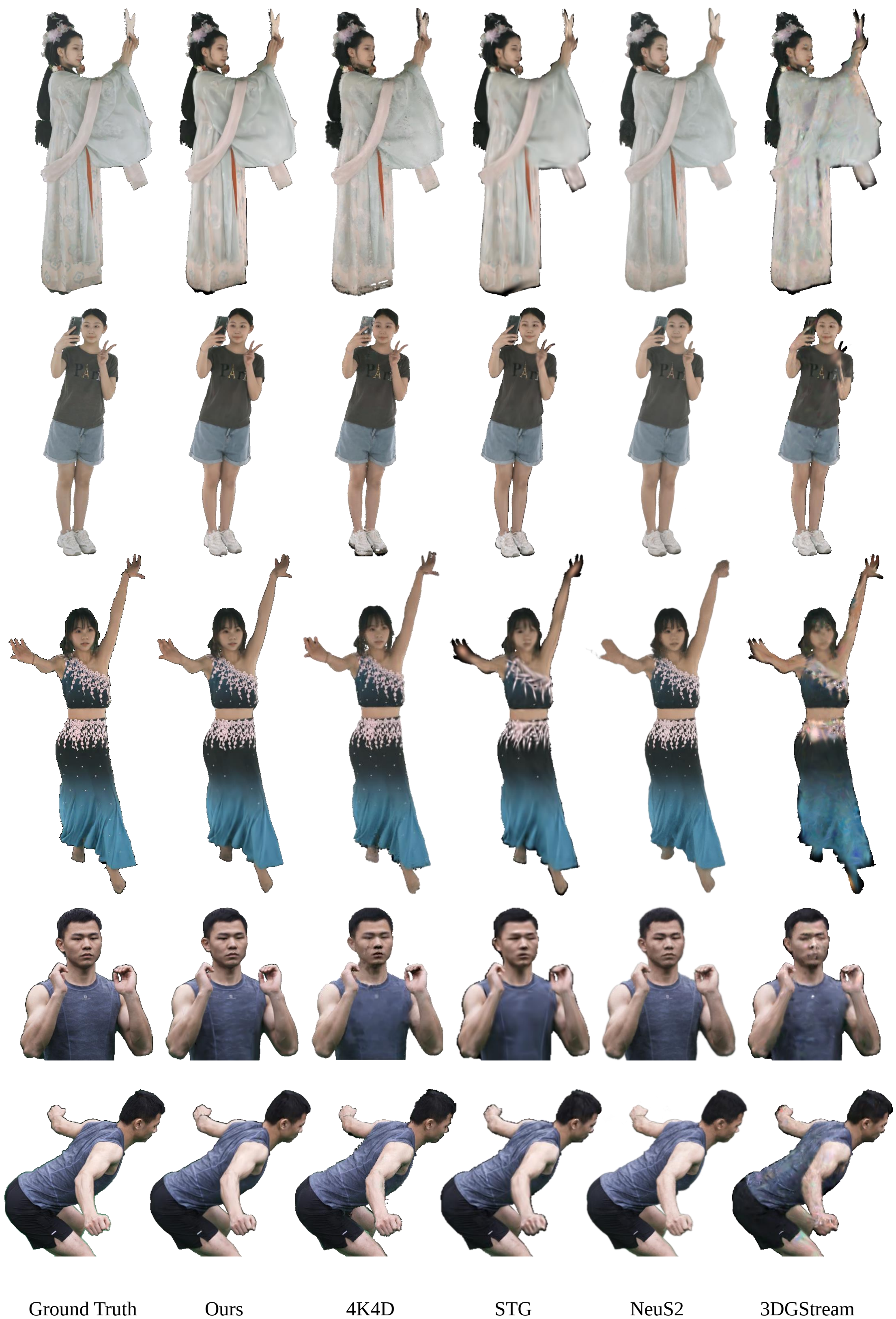

Figure 1. Additional qualitative comparison of novel view synthesis on the DNA-Rendering and NHR datasets.

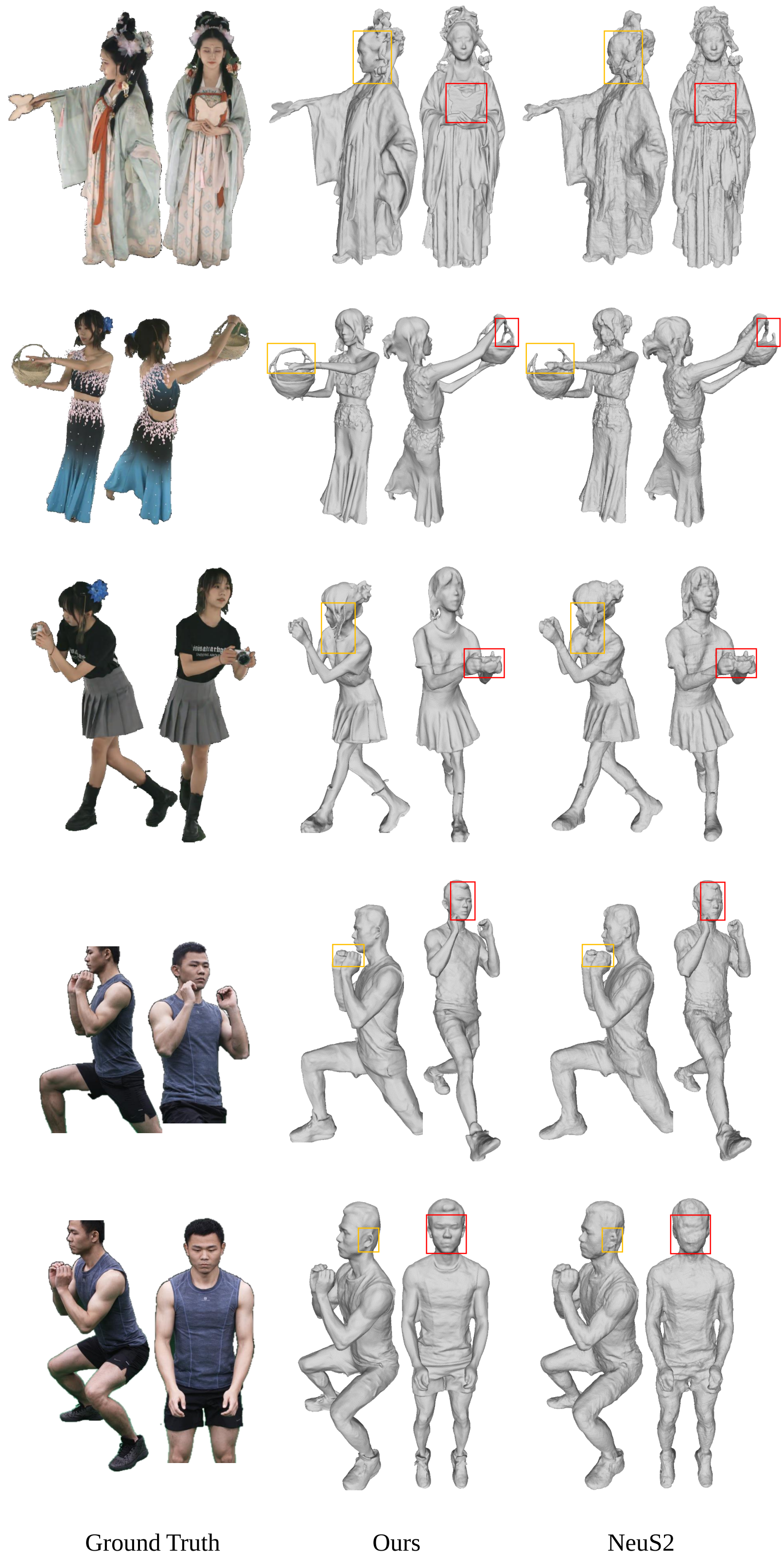

Figure 2. Additional comparison of surface reconstruction on the DNA-Rendering and NHR datasets.

Type	Method	0008_01			0012_11			0013_01		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Holistic	4K4D	31.36	0.974	0.047	35.81	0.990	0.018	34.52	0.987	0.021
	STG	24.08	0.944	0.068	33.55	0.986	0.023	25.47	0.957	0.047
Incremental	NeuS2	30.24	0.980	0.054	35.54	0.992	0.023	33.33	0.987	0.030
	3DGStream	27.46	0.960	0.075	33.88	0.986	0.033	29.14	0.969	0.047
	Ours	32.07	0.980	0.039	37.03	0.992	0.018	35.46	0.988	0.022
Type	**Method**	**0013_03**			**0013_09**			**Average**		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Holistic	4K4D	34.41	0.986	0.022	36.48	0.989	0.020	34.52	0.985	0.025
	STG	27.49	0.965	0.037	31.84	0.977	0.031	28.49	0.966	0.041
Incremental	NeuS2	33.60	0.987	0.029	36.27	0.990	0.025	33.80	0.987	0.032
	3DGStream	29.78	0.972	0.045	33.63	0.982	0.037	30.78	0.974	0.047
	Ours	35.43	0.988	0.020	37.19	0.990	0.020	35.44	0.988	0.024

Table 2. Per-scene quantitative results on the DNA-Rendering dataset [1]. The best values are highlighted in **red**, and the second-best values in **yellow**. Our method achieves the highest rendering quality compared to all other baselines.

Method	sport_1			sport_2			sport_3			basketball			**Average**		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
4K4D	33.37	0.975	0.026	34.57	0.968	0.052	34.19	0.968	0.051	32.49	0.977	0.027	33.65	0.972	0.039
STG	28.65	0.952	0.068	29.88	0.958	0.065	26.34	0.940	0.084	27.35	0.949	0.080	28.05	0.949	0.074
NeuS2	33.53	0.975	0.038	33.62	0.971	0.047	33.35	0.972	0.044	31.66	0.970	0.057	33.04	0.972	0.047
3DGStream	31.73	0.960	0.070	31.12	0.955	0.082	30.86	0.954	0.083	29.08	0.951	0.096	30.70	0.955	0.083
Ours	33.64	0.974	0.046	34.42	0.973	0.056	34.14	0.974	0.052	31.99	0.972	0.060	33.55	0.973	0.054

Table 3. Per-scene quantitative results on the NHR dataset [8].