

Supplementary Material for ForestSplats: Deformable transient field for Gaussian Splatting in the Wild

In this supplementary materials, we provide a detailed description of the more complete experimental settings, along with additional qualitative and quantitative results, including metrics that were not included in the main paper due to the page limitations. Furthermore, we present additional ablation studies, detailed analyses and observations, and provide failure cases. We also discuss potential future directions. The contents are summarized as follows:

- Sec. A: More implementation details
- Sec. B: More experimental results
- Sec. C: More ablation studies
- Sec. D: User study
- Sec. E: Failure cases and Future works

A. More Implementation details

A.1. Architecture

Our method is based on the WildGaussians [5] codebase. For the deformable transient field, we follow Deform3D-GS [19] for modeling transient elements. We train a deformable MLP f_d with the transient field to capture transient elements. Otherwise, we design a color mapping MLP f_c consisting of eight fully connected layers with ReLU activations, 256-dimensional hidden layers, and a 256-dimensional output. We use transient embedding per image $t \in \mathbb{R}^{32}$ for deformable MLP f_d .

A.2. Inference phase

We provide details on inference of ForestSplats in Fig. 1. During inference, we utilize only static field to render static scenes without transient fields. In Photo Tourism dataset, we also utilize appearance features to consider appearance consistency.

A.3. Photo Tourism & NeRF On-the-go

For the Photo Tourism dataset [13], we optimize for 200K iterations to train a representation of the entire scene. For the NeRF On-the-go dataset [10], we optimize for 30K iterations. Most hyperparameter settings follow the default codebase [5]. To represent transient elements and handle appearance changes in the Photo Tourism dataset [13], we leverage an appearance MLP, following WildGaussians [5].

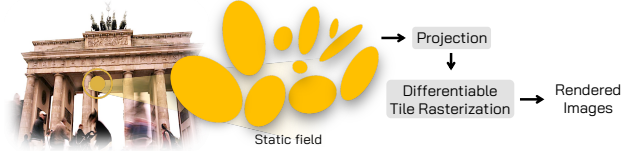


Figure 1. Illustration on inference of ForestSplats.

Scene	Method	Memory	Scene	Method	Memory
Mountain	Desplat [16]	29.03	Fountain	Desplat [16]	37.81
	HybridGS [7]	44.29		HybridGS [7]	57.68
	ForestSplats (Ours)	2.28		ForestSplats (Ours)	2.29
Patio	Desplat [16]	22.28	Spot	Desplat [16]	41.86
	HybridGS [7]	33.99		HybridGS [7]	63.86
	ForestSplats (Ours)	2.26		ForestSplats (Ours)	2.27

Table 1. Comparison of memory usage efficiency. We report CPU Memory usage (MB) for the initial number of transient Gaussians.

Additionally, we introduce a multi-stage training scheme, where \mathcal{L}_{init} is applied up to 80K iterations, followed by \mathcal{L}_{mid} until 120K iterations, and finally, \mathcal{L}_{total} is used for the remaining steps. For the NeRF On-the-go dataset [10], we apply \mathcal{L}_{init} for the first 15K iterations, continue with \mathcal{L}_{mid} until 20K iterations, and finally switch to \mathcal{L}_{total} .

B. More experimental results

We provided additional experiment results to demonstrate a more comprehensive evaluation of our methods on the Photo Tourism [13] and NeRF On-the-go [10] datasets. Furthermore, we provide additional qualitative results to validate the effectiveness of each component in Fig. 14. Moreover, we highly recommend that the reader watch the several videos on the [webpage](#). Our method achieves consistency and high-quality novel-view synthesis.

B.1. More Photo Tourism results

As illustrated in Fig. 8 and Tab. 6, our methods achieve high-quality rendering and show competitive performance compared to existing methods without VFM. Wild-GS and WildGaussians use semantic features from pre-trained VFM to generalize static scenes. In contrast, our method effec-

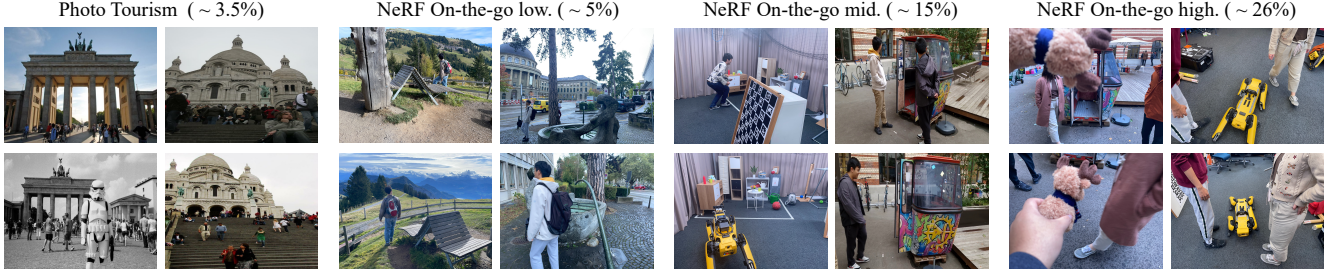


Figure 2. Visualization of sample training images. We present occluders in the Photo Tourism and NeRF On-the-go datasets.

Scene	Method	Memory usage (MB)		Testing FPS
		Static field	Transient field	
Corner	3D-GS [4]	410.70	0	116
	HybridGS [7]	42.20	34.70	160
	ForestSplats (Ours)	58.0	2.27	143
Trevi	GS-W [5]	91.21	0	38
Fountain	ForestSplats (Ours)	30.5	3.39	103

Table 2. Comparison of memory usage (MB) and testing FPS.

Segments	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
10	19.83	0.65	0.324
15	21.47	0.711	0.198
20	22.61	0.781	0.145
25	22.21	0.738	0.199

Table 3. Comparison of the number of superpixels.

tively decomposes transient elements from static scenes without VFM. Furthermore, we gradually interpolate between two appearance embeddings, as shown in Fig. 10. The results indicate the smoothness and consistency of the appearance embedding.

B.2. More NeRF On-the-go results

For NeRF On-the-go dataset [10], we visualize additional results of the remaining scenes due to page limitations as shown in Fig. 9. NeRF On-the-go [10] and WildGaussians [5] tackle transient elements using semantic features from the Vision Foundation model. However, our method addresses transient elements by considering photometric errors and superpixels. The detailed quantitative results demonstrate that ForestSplats outperforms the prior methods in Tab. 7. Specifically, our ForestSplats achieves state-of-the-art performance across five scenes, showing high-quality results.

C. More ablation studies

C.1. Efficiency of deformable transient field

As shown in Tab. 1 and Tab. 2, we emphasize the advantages of our method, particularly in terms of memory usage and

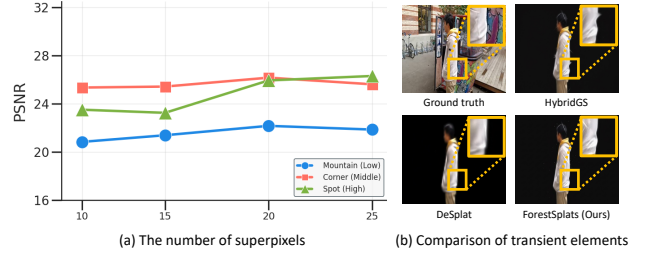


Figure 3. Analysis of superpixel-aware mask. (a) The number of superpixels (b) Qualitative results from rendered transient results.

efficiency. Although ForestSplats requires the Deformable MLP and the color mapping MLP, our method efficiently utilizes memory to represent transient elements. Compared to GS-W [22], our method demonstrates remarkable computational efficiency on the Trevi Fountain scene in Tab. 2.

C.2. Effectiveness of superpixel-aware mask

As shown in Fig. 11, superpixel-aware mask effectively captures the transient elements compared to the existing methods. Furthermore, we demonstrate that our method explicitly decomposes the transient elements from the 2D scenes, as provided in Fig. 12. Moreover, we provide additional results for the multi-stage training scheme in Fig. 4. We also provide detailed results for the number of superpixels in Tab. 3. As shown in Fig. 2, the distribution of transient elements varies between the Photo Tourism and NeRF On-the-go datasets. Therefore, adjusting the number of superpixels for each scene could enhance performance. We experimentally show impressive results on scenes ranging from low to high occlusion, as shown in Fig. 3, which indicates that the performance generally generalizes well. In addition, although our method does not explicitly capture distractors, it shows better qualitative results compared to DeSplat and HybridGS, as shown in Fig. 3.

C.3. Proof of positional gradient

We provide a step-by-step proof that the positional gradient is generally directed towards a region of static elements.

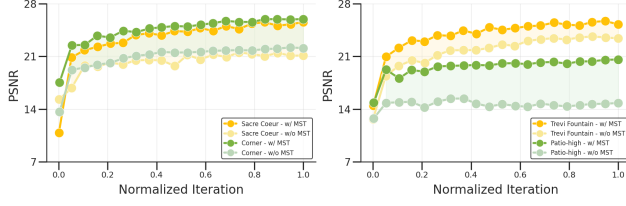


Figure 4. Comparison of training PSNR across with and without a multi-stage training scheme. MST denotes multi-stage training.

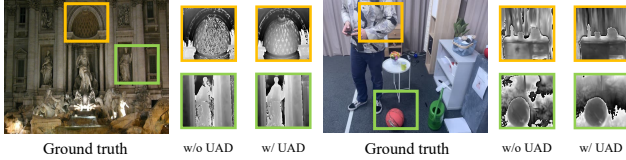


Figure 5. Comparison of depth consistency w/o and w/ UAD.

The gradient is calculated by:

$$\mathcal{G}_{i,x} = \frac{\partial \mathcal{L}}{\partial \mu_{i,x}} = \sum_{k=1}^p \frac{\partial \mathcal{L}_k}{\partial \mu_{i,x}}, \quad (1)$$

where \mathcal{L} denotes the rendering loss from \mathcal{L}_{GS} , p is the number of pixels covered by Gaussian \mathcal{G}_i , and $\mu_{i,x}$ is the pixel-space projection of Gaussian \mathcal{G}_i under viewpoint x . The computation for static Gaussians is as follows:

$$\frac{\partial \mathcal{L}_k^{(static)}}{\partial \mu_{i,x}} \propto (1 - \mathcal{M}_S(x_k)) \frac{\partial \mathcal{L}_k}{\partial \mu_{i,x}}. \quad (2)$$

On the other hand, the positional gradient for transient Gaussians is computed as:

$$\frac{\partial \mathcal{L}_k^{(transient)}}{\partial \mu_{i,x}} \propto \mathcal{M}_S(x_k) \frac{\partial \mathcal{L}_k}{\partial \mu_{i,x}}. \quad (3)$$

Thus, the positional gradient is influenced by the superpixel-aware mask \mathcal{M}_S . In the Photo Tourism [13] and NeRF On-the-Go datasets [10], transient elements typically constitute less than 30%. Consequently, the gradient tends to be directed toward regions of static elements.

C.4. Discussion of uncertainty-aware densification

We further demonstrate that our uncertainty-aware densification (UAD) effectively removes high-uncertainty static Gaussians, thereby enhancing the consistency and reconstruction quality in Fig. 5 and Fig. 13. Specifically, UAD enhances depth consistency compared to the case without UAD, as illustrated in Fig. 5. Furthermore, we performed an additional evaluation using SIFT-based feature matching on images rendered with and without UAD. As shown in Fig. 13, leveraging UAD ensures that feature correspondences remain consistent across different viewpoints by removing high-uncertainty static Gaussians and avoiding generating static Gaussian in the region of transient elements.

Method	UAD (Ours)	PUP	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	static (MB)
ForestSplats	✓		24.11	0.802	0.213	30.5
ForestSplats		✓	22.63	0.747	0.237	59.7

Table 4. Comparison of densification strategies on Trevis Fountain.

DTF	SAM	UAD	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
✓			20.14	0.868	0.178
	✓		22.38	0.861	0.175
	✓	✓	22.62	0.862	0.169
✓	✓		22.82	0.871	0.155
✓	✓	✓	22.75	0.869	0.162
✓	✓	✓	23.84	0.876	0.123

Table 5. Analysis of each module on the Sacre Coeur scene.

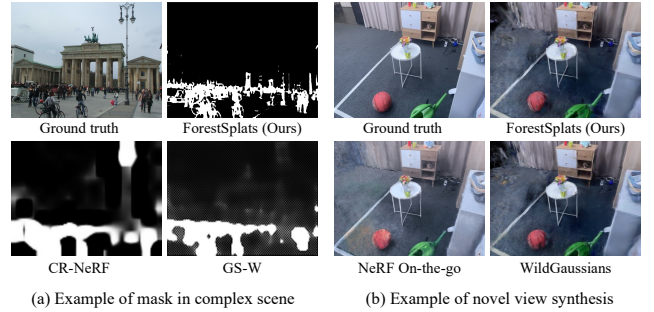


Figure 6. Failure cases of our method. (a) Our method fails to capture fine details in masks within complex scenes. (b) Furthermore, our method fails to generalize to sparse training views.

We also compare PUP [3] with UAD and show that ours achieves more effective performance, as shown in Table 4.

C.5. Analysis of each component

Furthermore, we demonstrate the effect of each module independently in Tab. 5, complementing the progressively additive results presented in the main paper.

D. User study

D.1. User evaluation results

We recruited several researchers as survey participants and asked them to complete a questionnaire, as shown in Fig. 15. The questionnaire includes several questions and rendering results comparing our method with baselines on three different scenes. Baseline methods include WildGaussians [5], GS-W [22], NeRF On-the-go [10]. Each participant was asked to evaluate the rendering videos on a scale from 1 (poor) to 5 (excellent). The average scores for each question are visualized in Fig. 7. The results show that our method achieved the highest scores across all aspects, including transient removal, floater reduction, and consistency quality, demonstrating the effectiveness of our approach.

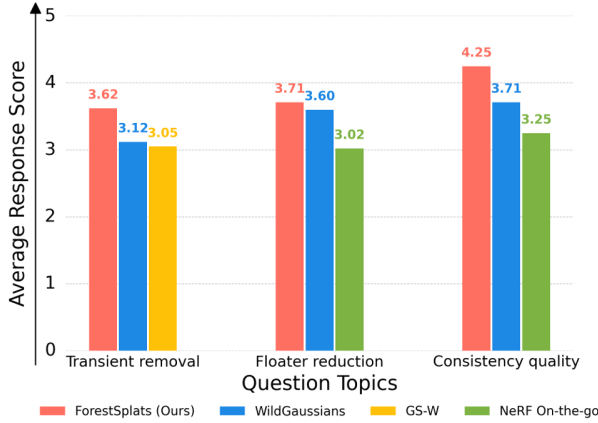


Figure 7. User Study Results: Average response scores on transient removal, floater reduction, and consistency quality.

E. Failure cases and Future works

E.1. Failure cases

Our method shows impressive results in capturing transient elements. However, similar to prior methods, our method struggles to accurately mask transient elements, as shown in Fig. 6-(a) due to complex real-world environments. Furthermore, as shown in Fig. 6-(b), our method often fails to generalize in novel-view synthesis, especially in scenarios with sparse training views.

E.2. Future works

Our method explicitly decomposes a complex scene into static and transient fields. However, transient elements are ephemeral and short-lived in the scenes. As a result, the absence of multi-view consistency eliminates the need to consider depth. Thus, deforming the 2D transient field of HybridGS [7], which consists of Gaussians with nine parameters, improves efficiency and reduces memory usage. Specifically, while we demonstrate the effectiveness of our methods on the Photo Tourism [13] and NeRF On-the-Go [10] datasets, another promising direction is extending our approach to large-scale dynamic scenes such as KITTI-360 [6] and NuScenes [1]. We leave this as future work.



Figure 8. Additional qualitative results from novel-view synthesis on the Photo Tourism dataset.

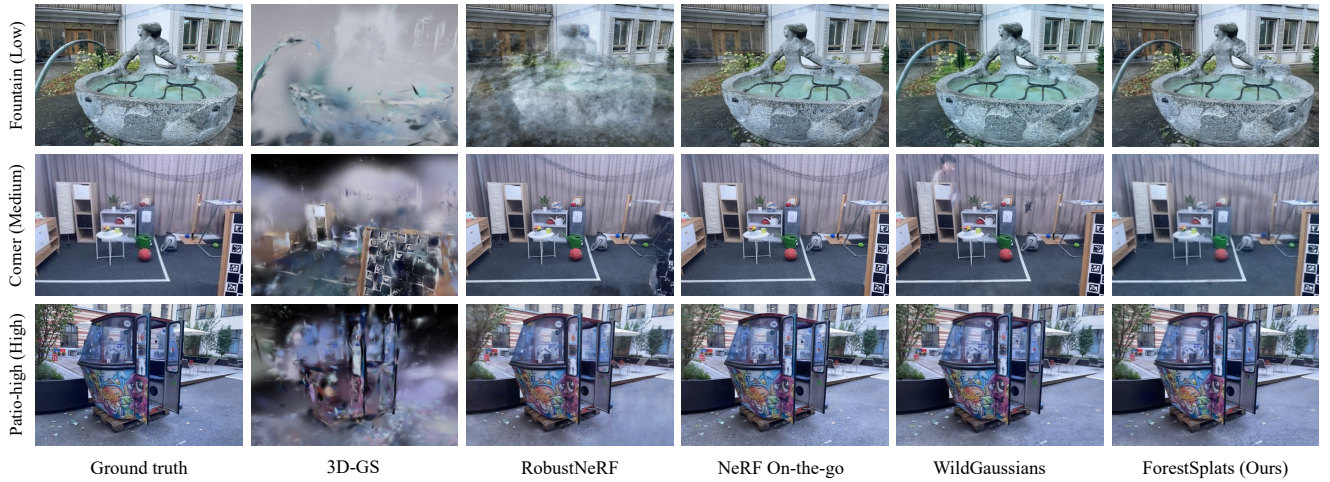


Figure 9. Additional qualitative results from novel-view synthesis on the NeRF On-the-go dataset.

Method	Brandenburg Gate			Sacre Coeur			Trevi Fountain		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3D-GS [4]	19.33	0.884	0.132	17.70	0.845	0.176	17.08	0.714	0.241
NeRF-W [8]	24.17	0.891	0.167	19.20	0.808	0.192	18.97	0.698	0.265
Ha-NeRF [9]	24.04	0.887	0.139	20.02	0.801	0.171	20.18	0.691	0.223
DeSplat [16]	25.04	0.92	0.142	20.14	0.868	0.178	23.31	0.775	0.226
CR-NeRF [18]	26.53	0.900	0.106	22.07	0.823	0.152	21.48	0.712	0.207
RobustNeRF [11]	25.79	0.923	<u>0.094</u>	20.94	0.852	0.137	23.58	0.785	0.170
IE-NeRF [15]	25.33	0.898	0.158	20.37	0.861	0.169	20.76	0.719	0.217
SWAG [2]	26.33	0.929	0.139	21.16	0.86	0.185	23.10	0.815	0.208
WildGaussian [5]	27.77	0.927	0.133	22.56	0.859	0.177	23.63	0.766	0.228
NexusSplats [14]	27.76	0.922	0.141	23.13	0.859	0.174	23.96	0.766	0.240
GS-W [22]	27.96	0.932	0.086	23.24	0.863	0.130	22.91	0.801	0.156
Wild-GS [17]	29.65	<u>0.933</u>	0.095	24.99	0.878	<u>0.127</u>	24.45	<u>0.808</u>	<u>0.162</u>
ForestSplats (Ours)	<u>28.13</u>	0.935	0.118	<u>23.84</u>	<u>0.876</u>	0.123	<u>24.11</u>	0.802	0.213

Table 6. Quantitative results on Photo Tourism dataset. The bold and underlined numbers indicate the best and second-best results.

Method	Low Occlusion						Medium Occlusion						High Occlusion					
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
RobustNeRF [11]	17.54	0.496	0.383	15.65	0.318	0.576	23.04	0.764	0.244	20.39	0.718	0.251	20.65	0.625	0.391	20.54	0.578	0.366
Gaussian Opacity Field [21]	19.86	0.649	0.200	20.19	0.672	0.189	21.15	0.728	0.230	18.31	0.639	0.328	20.18	0.689	0.338	18.31	0.639	0.328
3D-GS [4]	19.40	0.638	0.213	19.96	0.659	<u>0.185</u>	20.90	0.713	0.241	17.48	0.704	0.199	20.77	0.693	0.316	17.29	0.604	0.363
Mip-Splatting [20]	20.70	0.661	0.169	20.37	0.662	0.187	21.53	0.739	0.241	15.58	0.491	0.536	20.03	0.683	0.324	15.58	0.491	0.536
GS-W [22]	19.43	0.596	0.299	20.06	<u>0.723</u>	0.274	22.17	0.793	0.155	19.90	0.681	0.260	17.13	0.608	0.409	19.90	0.681	0.260
NeRF On-the-go [10]	20.46	0.661	0.186	20.79	0.661	0.195	23.74	0.806	0.127	20.88	0.754	<u>0.133</u>	22.80	0.800	0.132	21.57	0.706	0.205
WildGaussians [5]	20.43	0.653	0.255	20.81	0.662	0.215	24.16	0.822	0.045	21.44	0.800	0.138	23.82	0.816	0.138	22.23	0.725	0.206
SpotLessSplats [12]	19.84	0.580	0.294	20.19	0.612	0.258	24.03	0.795	0.258	21.55	<u>0.838</u>	0.065	23.52	0.756	0.185	20.31	0.664	0.259
DeSplat [16]	19.59	0.710	<u>0.170</u>	20.27	0.680	0.170	<u>26.05</u>	<u>0.880</u>	<u>0.090</u>	20.89	0.810	0.110	26.07	0.900	0.090	<u>22.59</u>	0.840	0.120
HybridGS [7]	21.73	0.693	0.284	<u>21.11</u>	0.674	0.252	25.03	0.847	0.151	21.98	0.812	0.169	24.33	0.794	0.196	21.77	0.741	0.211
ForestSplats (Ours)	22.17	0.736	0.235	21.62	0.741	0.198	26.17	0.891	0.136	<u>21.76</u>	0.849	0.134	<u>25.94</u>	<u>0.846</u>	<u>0.109</u>	22.74	<u>0.784</u>	<u>0.129</u>

Table 7. Quantitative results on NeRF On-the-go dataset. The bold and underlined numbers indicate the best and second-best results.



Figure 10. Visualization of appearance interpolation. We gradually interpolate from the source appearance to the target appearance.



Figure 11. Comparison of transient mask quality with existing methods on the Photo Tourism dataset.

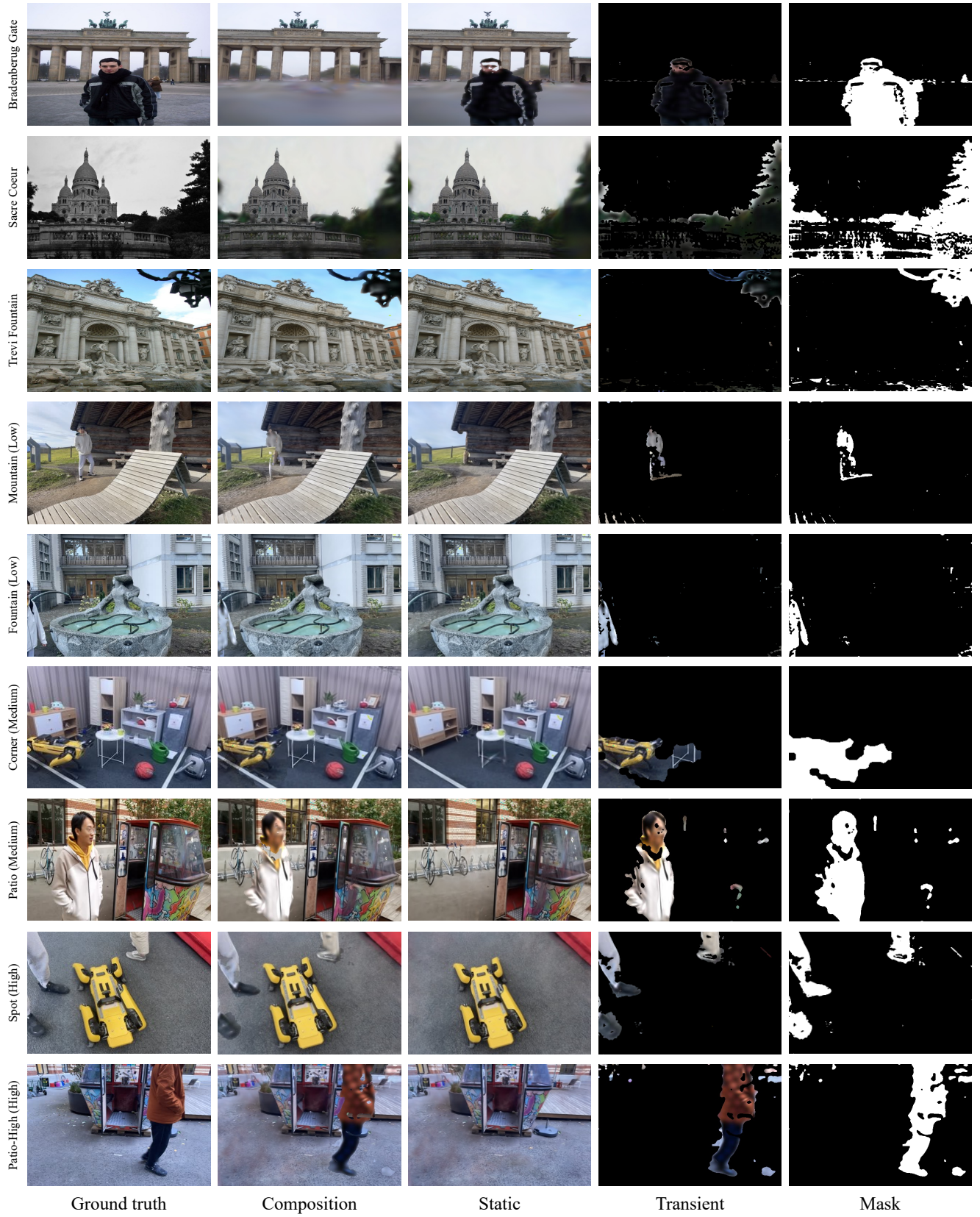


Figure 12. Additional novel-view synthesis results for static and transient elements on the Photo Tourism and NeRF On-the-go datasets.

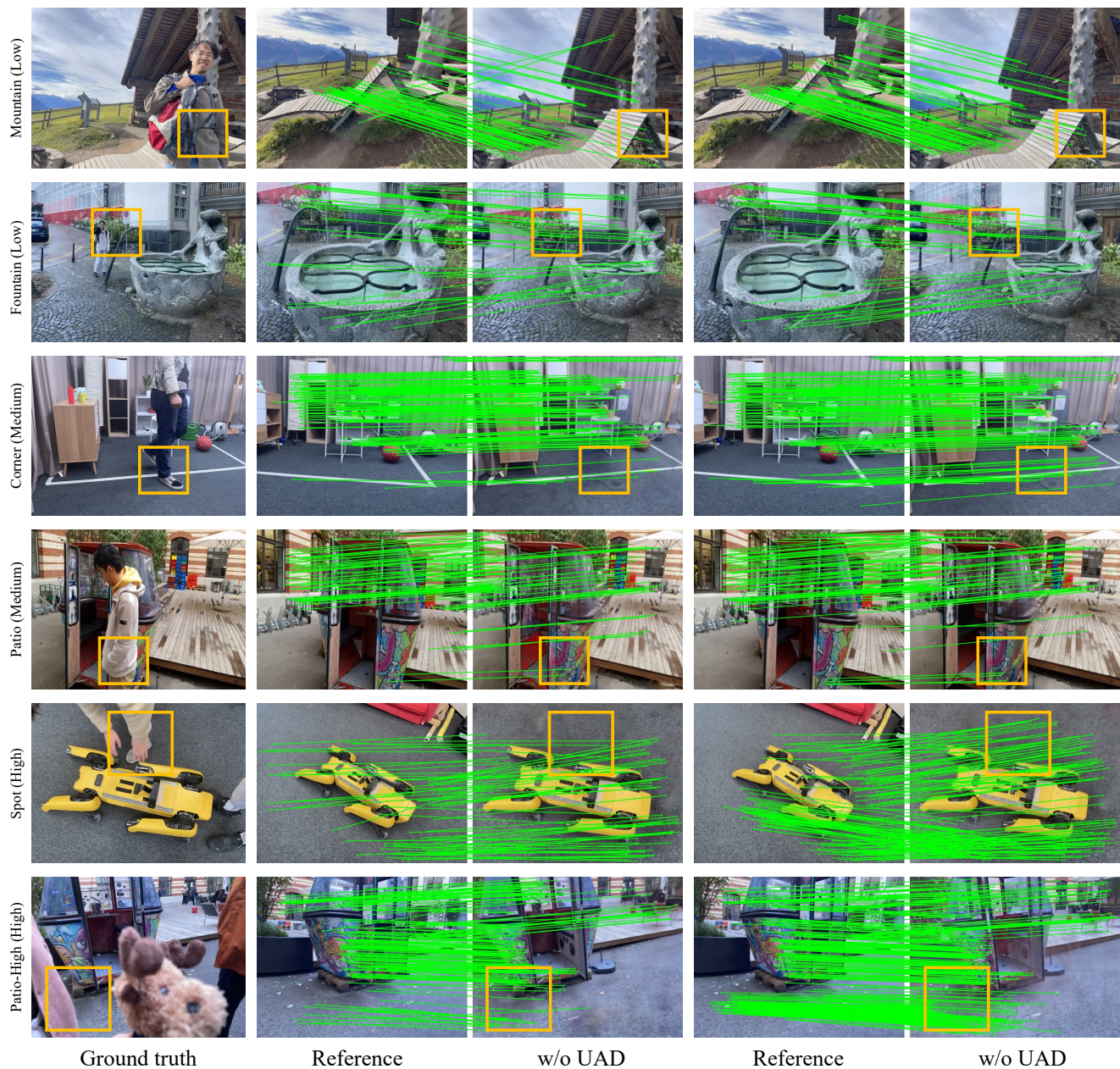


Figure 13. Comparison of image matching results on the NeRF On-the-go dataset with and without UAD across all scenes.

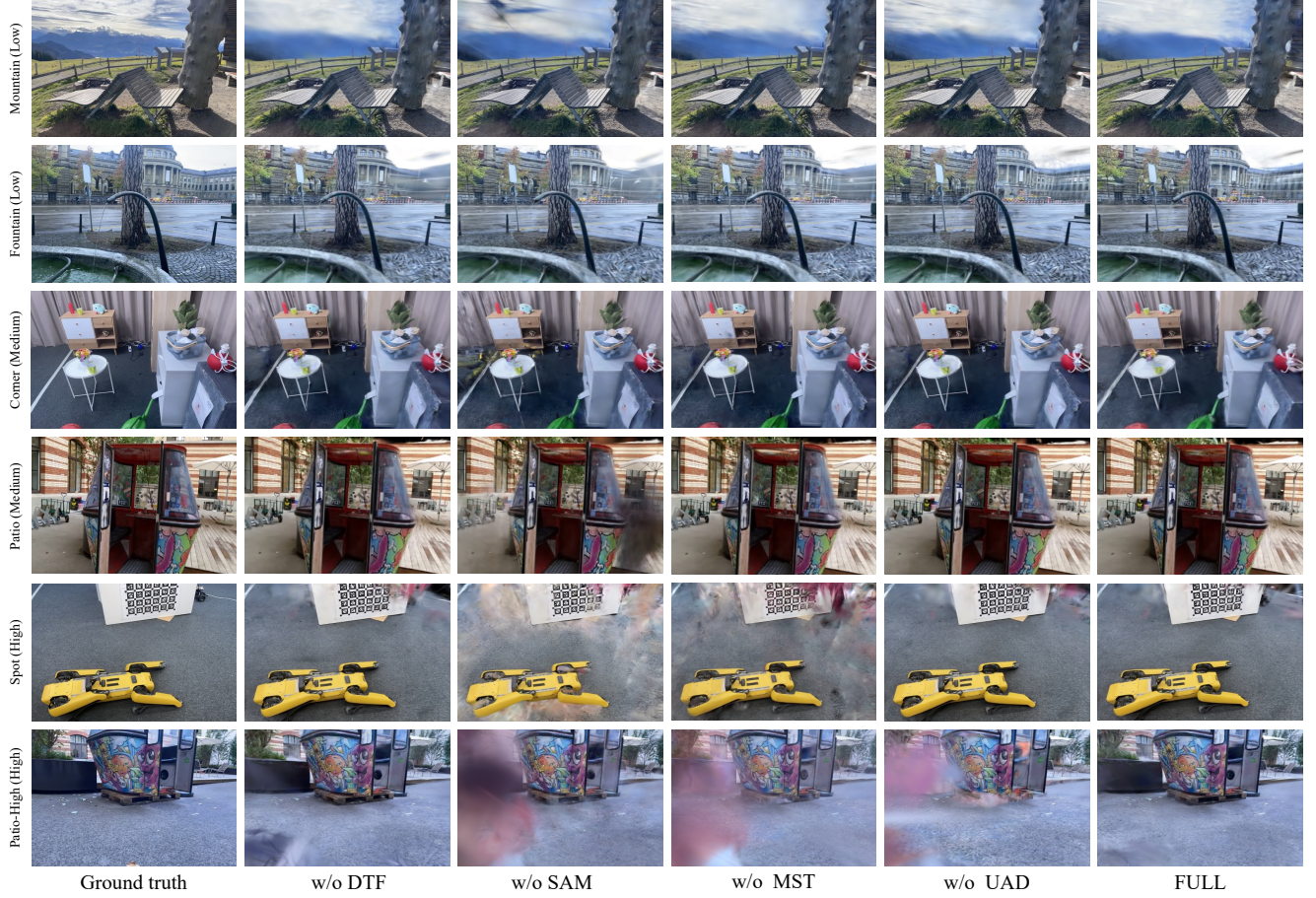


Figure 14. More qualitative results for each component in ForestSplats. DTF, SAM, MST, and UAD denote Deformable MLP, Superpixel-aware mask, Multi-stage training scheme, and Uncertainty-aware densification.

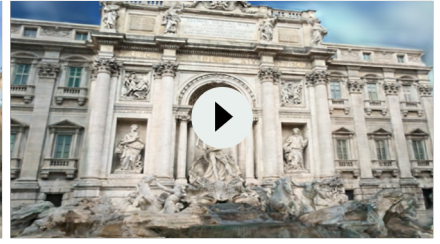
1. Trevi Fountain (Photo Tourism dataset)



(a)



(b)



(c)

2. Patio-high (NeRF On-the-go dataset)



(a)



(b)



(c)

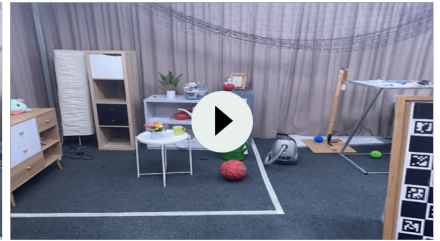
3. Corner (NeRF On-the-go dataset)



(a)



(b)



(c)

Question 1. How naturally do you think the transient distractors were removed?

Question 2. How naturally does the scene appear without the interference of the floaters?

Question 3. How well do the rendering results consistently maintain quality?

Figure 15. Our user study questionnaire. Each participant was shown an upper figure, which is a rendering video of several scenes using different methods. After watching the video, they were asked to answer the question below.

References

- [1] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multi-modal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 4
- [2] Hiba Dahmani, Moussab Bennehar, Nathan Piasco, Luis Roldao, and Dzmitry Tsishkou. Swag: Splatting in the wild images with appearance-conditioned gaussians. In *European Conference on Computer Vision*, pages 325–340. Springer, 2024. 6
- [3] Alex Hanson, Allen Tu, Vasu Singla, Mayuka Jayawardhana, Matthias Zwicker, and Tom Goldstein. Pup 3d-gs: Principled uncertainty pruning for 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5949–5958, 2025. 3
- [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. 2, 6
- [5] Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024. 1, 2, 3, 6
- [6] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3292–3310, 2022. 4
- [7] Jingyu Lin, Jiaqi Gu, Lubin Fan, Bojian Wu, Yujing Lou, Renjie Chen, Ligang Liu, and Jieping Ye. Hybrids: Decoupling transients and statics with 2d and 3d gaussian splatting. *arXiv preprint arXiv:2412.03844*, 2024. 1, 2, 4, 6
- [8] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7210–7219, 2021. 6
- [9] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 6
- [10] Weining Ren, Zihan Zhu, Boyang Sun, Jiaqi Chen, Marc Pollefeys, and Songyou Peng. Nerf on-the-go: Exploiting uncertainty for distractor-free nerfs in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8931–8940, 2024. 1, 2, 3, 4, 6
- [11] Sara Sabour, Suhani Vora, Daniel Duckworth, Ivan Krasin, David J Fleet, and Andrea Tagliasacchi. Robustnerf: Ignoring distractors with robust losses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20626–20636, 2023. 6
- [12] Sara Sabour, Lily Goli, George Kopanas, Mark Matthews, Dmitry Lagun, Leonidas Guibas, Alec Jacobson, David J Fleet, and Andrea Tagliasacchi. Spotlessplats: Ignoring distractors in 3d gaussian splatting. *arXiv preprint arXiv:2406.20055*, 2024. 6
- [13] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006. 1, 3, 4
- [14] Yuzhou Tang, Dejun Xu, Yongjie Hou, Zhenzhong Wang, and Min Jiang. Nexussplats: Efficient 3d gaussian splatting in the wild. *arXiv preprint arXiv:2411.14514*, 2024. 6
- [15] Shuaixian Wang, Haoran Xu, Yaokun Li, Jiwei Chen, and Guang Tan. Ie-nerf: inpainting enhanced neural radiance fields in the wild. *arXiv preprint arXiv:2407.10695*, 2024. 6
- [16] Yihao Wang, Marcus Klasson, Matias Turkulainen, Shuzhe Wang, Juho Kannala, and Arno Solin. Desplat: Decomposed gaussian splatting for distractor-free rendering. *arXiv preprint arXiv:2411.19756*, 2024. 1, 6
- [17] Jiacong Xu, Yiqun Mei, and Vishal M Patel. Wild-gs: Real-time novel view synthesis from unconstrained photo collections. *arXiv preprint arXiv:2406.10373*, 2024. 6
- [18] Yifan Yang, Shuhai Zhang, Zixiong Huang, Yubing Zhang, and Mingkui Tan. Cross-ray neural radiance fields for novel-view synthesis from unconstrained image collections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15901–15911, 2023. 6
- [19] 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
- [20] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19447–19456, 2024. 6
- [21] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient and compact surface reconstruction in unbounded scenes. *arXiv preprint arXiv:2404.10772*, 2024. 6
- [22] Dongbin Zhang, Chuming Wang, Weitao Wang, Peihao Li, Minghan Qin, and Haoqian Wang. Gaussian in the wild: 3d gaussian splatting for unconstrained image collections. In *European Conference on Computer Vision*, pages 341–359. Springer, 2024. 2, 3, 6