

418 A Technical Appendices and Supplementary Material

419 A.1 Experiments and visualizations

420 A.1.1 Ablation study

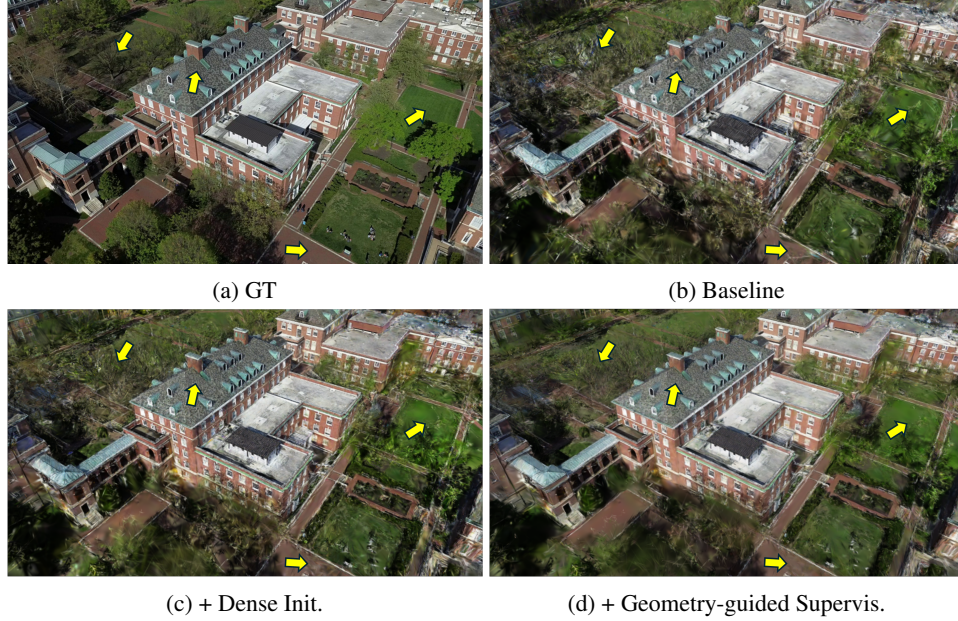


Figure 5: Visualizations of ablation study

421 As illustrated in Fig. 5, the baseline exhibits noisy scene elements and holes in the building’s rooftop
 422 due to insufficient geometric support. With our dense initialization, complete surfaces are rendered.
 423 Our multi-view geometry-guided supervision regularizes the scene appearance and suppresses residual
 424 artifacts.

425 A.1.2 Initialization comparisons

Table 5: Experiments on the effectiveness of different initializations. The metrics are reported as the average on the Sparse Mip-NeRF 360 dataset.

Method	PSNR↑	SSIM↑	LPIPS↓	DSIM↓
sparse init.	20.83	0.627	0.267	0.109
image-level alignment	21.06	0.631	0.253	0.094
semantic alignment	21.96	0.690	0.216	0.080
DUST3R [31]	19.89	0.585	0.270	0.118

426 In this section, we compare different initialization strategies, namely the sparse SfM point cloud,
 427 dense initialization with image-level alignment as introduced in Section 3.1, our proposed dense
 428 initialization with semantic alignment, and DUST3R [31] point cloud. In Figure 6, the sparse SfM
 429 point cloud contains only a few thousand points and covers a small fraction of the scene. Even
 430 though initialization with image-level alignment is much denser, it also introduces more errors in the
 431 point cloud, leading to noisy structures. In contrast, our method, which favors more accurate local
 432 alignments, achieves cleaner and semantically meaningful scene components.

433 Quantitatively (Table 5), initialization from image-level alignment offers only marginal benefit
 434 compared to the baseline, as misplaced Gaussians that are not pruned or densified correctly can
 435 produce noisy structures, as shown in Fig. 7. DUST3R is a two-view pointmap estimator. When the
 436 number of images is greater than two, it aggregates all pairwise pointmap predictions into a very
 437 dense point cloud, usually millions of points. To utilize DUST3R points, we align them to the SfM

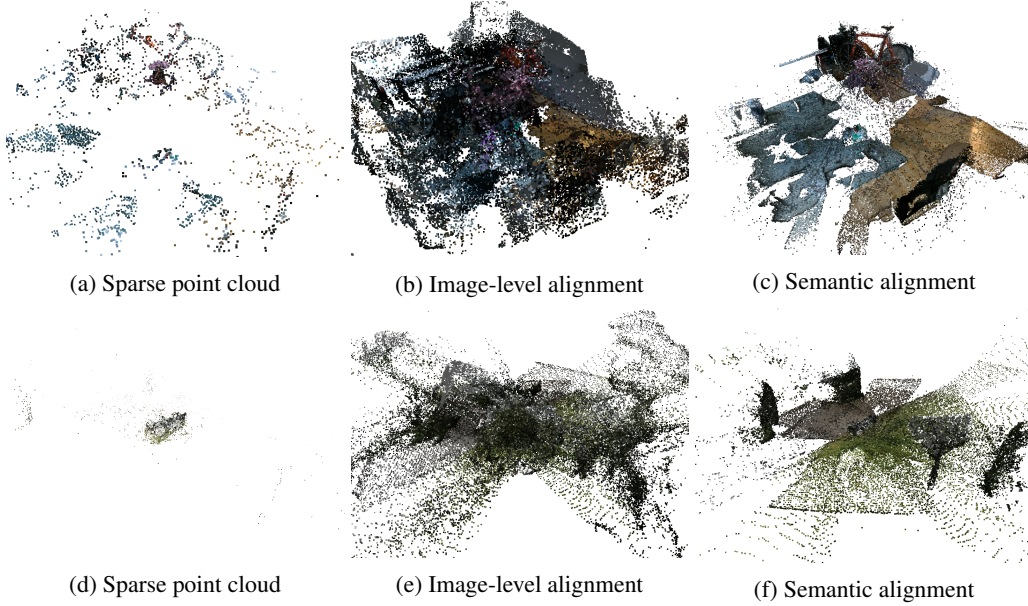


Figure 6: Visualizations of different point cloud initializations.

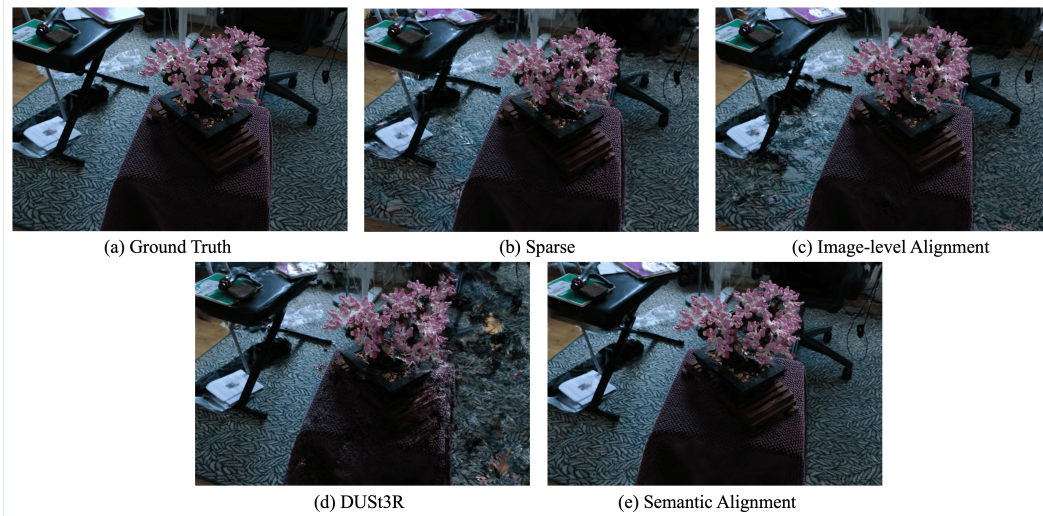


Figure 7: Visualizations of rendering with different point cloud initializations.

points based on corresponding pixels using Procrustes Alignment [32]. While the output of DUST3R is visually pleasing, it still suffers from the two-view depth ambiguities, often leading to incorrect distances between objects. As shown in Fig. 7, it produces ghosting artifacts due to the strong initialization bias. Notably, our approach improves the PSNR by 1.13, SSIM by 0.063, LPIPS by 0.051, and DSIM by 0.029. In addition, the time complexity of DUST3R to run N images is $\mathcal{O}(N^2)$ compared to $\mathcal{O}(N)$ for the monocular depth estimator, which makes it harder to scale to more images. This analysis highlights the importance of semantic depth alignment, which guides 3DGS to converge to a better scene reconstruction.

A.1.3 Dense init. for other in-the-wild methods

In this section, we investigate the performance of other in-the-wild methods using our proposed dense initialization. Based on Table 6, on one hand, all methods achieve significantly better metrics compared to their original baseline. For example, GS-W gains 0.9 dB in PSNR, 0.054 in SSIM,

450 and reduces 0.11 in LPIPS and 0.069 in DSIM. This experiment confirms that our initialization is a
 451 drop-in enhancement for any 3DGS-based pipeline, requiring only a few lines of code change for
 452 switching the initialized point cloud. On the other hand, the improved performance of other methods
 453 is still inferior to MS-GS by a large margin, validating the effectiveness of our appearance modules
 454 and multi-view geometry-guided supervision in this challenging setting.

Table 6: Experiments on the effectiveness of our dense initialization applied to other methods for multi-appearance synthesis. The metrics are reported as the average on the Sparse unbounded drone dataset.

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	DSIM \downarrow
GS-W [12]	17.33	0.491	0.487	0.279
+ dense init.	18.23	0.545	0.377	0.210
Wild-GS [13]	14.13	0.345	0.547	0.487
+ dense init.	14.35	0.395	0.544	0.443
WildGaussians [17]	15.60	0.388	0.546	0.428
+ dense init.	16.50	0.449	0.482	0.316
Ours	19.87	0.580	0.322	0.096

455 A.2 Semantic alignment algorithm

Algorithm 1: Semantic Masks Prediction

Input : Image I_n , a set of visible 2D SfM points \mathcal{X} on I_n , segmentation model \mathcal{S} , threshold TH_{sfm} , threshold TH_{IoU} .

Output : Final set of masks M_{final} .

```

1 Def append_mask( $M_i, M_{final}, TH_{IoU}$ ):
2   merged = False;
3   for  $M \in M_{final}$  do
4     if  $M_i \cap M > TH_{IoU}$  then
5        $M = M \cup M_i$ ;                                /* Merge the masks */
6       merged = True;
7       break;
8   end
9   end
10  if not merged then
11     $M_i \rightarrow M_{final}$ ;                                /* Append the mask to set */
12  end
13   $M_{final} = \emptyset$ ;
14  while  $\mathcal{X}$  is not empty do
15     $x_i \sim \mathcal{X}$ ;                                          /* Sample a point */
16     $M_i = \mathcal{S}(x_i, I_n)$ ;                                /* Prompt a mask */
17     $x_{m,i} = \mathcal{X} \cap M_i$ ;                                /* Find points within the mask */
18    if  $|x_{m,i}| > TH_{sfm}$  then
19      append_mask( $M_i, M_{final}, TH_{IoU}$ );                /* Enough points */
20    else
21       $M_i = \mathcal{S}(x_{m,i}, I_n)$ ;                            /* Re-prompt with points within the mask */
22       $x_{m,i} = \mathcal{X} \cap M_i$ ;
23      if  $|x_{m,i}| > TH_{sfm}$  then
24        append_mask( $M_i, M_{final}, TH_{IoU}$ );
25      else
26        continue;
27      end
28    end
29    Exclude  $x_{m,i}$  from  $\mathcal{X}$ ;                                /* Remove points from set */
30 end

```

The iterative refinement algorithm is detailed in Algorithm 1. This is an automatic process to find the semantic masks anchored by SfM points, which are back-projected individually to form a dense point cloud for 3DGS initialization.

A.3 Appearance embedding initialization

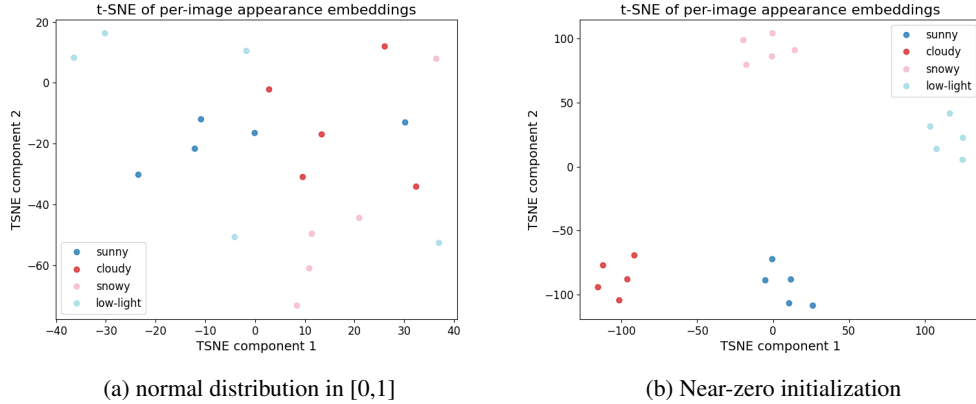


Figure 8: t-SNE visualizations of per-image appearance embeddings after training with different initializations

Appearance embeddings are typically initialized with a normal distribution in $[0, 1]$ [33, 14]. We find that this initialization introduces view-specific biases. Instead, we initialize them near zero, i.e., uniform distribution in $[-1 \times 10^{-4}, 1 \times 10^{-4}]$, which shows improved metrics and yields meaningful clusters after training, as shown in Fig. 8. We attribute this result to the near-zero initialization: it delays the expressive power of the per-image appearance embeddings, minimally influencing the MLP training in the early stages, so the network first learns a shared color basis and later allocates capacity to disentangle appearances.

A.4 Implementation details

We develop MS-GS based on the 3DGS implementation from NeRFStudio, called Splatfacto [34]. The baseline introduced in our ablation study Section 4.3 uses the same Splatfacto model. In Semantic Depth Alignment, the minimum number of SfM points threshold within a valid mask is 10. The intersection of two masks for merging is 0.7. The appearance MLP consists of 3 layers of 64 hidden units. The embedding sizes for the Gaussian feature and per-image appearance embeddings are 16 and 32, respectively. Virtual views are generated by interpolating toward one of the $k=4$ nearest training cameras. We use features extracted from blocks 3 and 4 of VGG-16 [25, 35, 36, 37] for feature loss at different resolutions and receptive fields. We set $\lambda_I = 0.8$, $\lambda_{\text{pix}} = 1.0$, and $\lambda_{\text{feat}} = 0.04$. The total number of training iterations is 16,500, with the geometry-guided supervision enabled after 15,000 iterations. The same hyperparameters are maintained throughout the experiments. Results are obtained using an NVIDIA RTX A5500 GPU.

A.5 In-the-wild evaluation

Sparse-view and multi-appearance registration is challenging because of limited overlap and view inconsistency; Fewer reliable feature matches result in suboptimal pose estimation and point triangulation. Sparse-view methods [27, 3, 4, 6, 7, 8, 5, 9] commonly assume ground-truth camera poses, i.e., calibration from dense views. However, only training views should be available in an in-the-wild setting. 3DGS-based methods rely heavily on the SfM point cloud, further necessitating the separation of training and testing views from the registration stage. A previous approach [11] has tried to perform re-triangulation based on known train poses, but does not account for pose inaccuracy. Therefore, we propose a coordinate alignment method, illustrated in Fig. 9, to disentangle train and test images in registration while maintaining them in the same coordinate system.

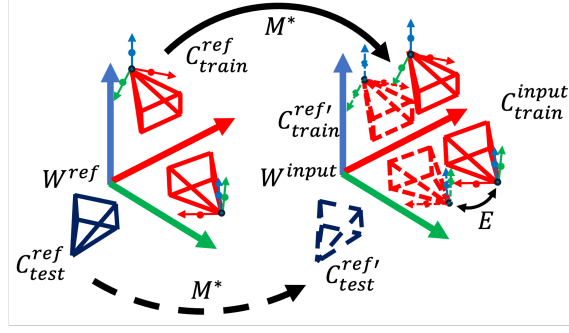


Figure 9: Illustration of Coordinate Alignment. We first compute the transformation M^* between train cameras in two coordinate systems $C_{\text{train}}^{\text{ref}}$ and $C_{\text{train}}^{\text{input}}$; each camera corresponds to 4 points: one position and three rotation points, displayed as small black, red, green, and blue points in the figure. The transformed $C_{\text{train}}^{\text{ref}}$ is denoted as $C_{\text{train}}^{\text{ref}'}$ in dashed lines, which is used to compute the error E between $C_{\text{train}}^{\text{input}}$. Finally, M^* transforms test camera poses $C_{\text{test}}^{\text{ref}}$ to $C_{\text{test}}^{\text{ref}'}$ in the input coordinate system.

489 A.5.1 Coordinate alignment

490 In coordinate alignment, we perform two registrations: 1. train-only images to get train cameras
 491 in the input coordinate system $C_{\text{train}}^{\text{input}}$ 2. train-test images to get train cameras and test cameras in
 492 the reference coordinate system, $C_{\text{train}}^{\text{ref}}$ and $C_{\text{test}}^{\text{ref}}$. A transformation M^* is computed between $C_{\text{train}}^{\text{input}}$
 493 and $C_{\text{train}}^{\text{ref}}$ using Procrustes Alignment [32] to transform test cameras $C_{\text{test}}^{\text{ref}}$ to the input coordinate
 494 system $C_{\text{test}}^{\text{ref}'}$. Conventionally, only camera positions/centers are considered during alignment. To
 495 leverage the camera rotation information, we sample additional 3 points along the rotation axis for
 496 each camera with the camera positions to achieve more accurate alignment. Formally, the point cloud
 497 of each camera $P_{\text{cam}} \in \mathbb{R}^{4 \times 3}$ is:

$$\begin{aligned} R &= [r_x, r_y, r_z], \\ s &= \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}, \\ P_{\text{cam}} &= \{T, T + sr_x, T + sr_y, T + sr_z\}, \end{aligned} \quad (9)$$

498 where $R \in \mathbb{R}^{3 \times 3}$, $T \in \mathbb{R}^{1 \times 3}$, and $s \in \mathbb{R}^{1 \times 3}$ are camera rotation matrix, translation, and scale
 499 approximated by per-dimension standard deviation σ . As a result, we use $C_{\text{train}}^{\text{input}}$, $C_{\text{test}}^{\text{ref}'}$, and points
 500 triangulated from $C_{\text{train}}^{\text{input}}$ for 3DGS training and evaluation.

501 As shown in Table 7, our rotation-aware alignment reduces the rotation error E_R , in degrees, by
 502 more than 10 times and the position error E_T , in an arbitrary unit as in the SfM, by 4 times. This
 503 improvement results in accurately aligned test cameras and, consequently, more reliable evaluations.

Table 7: Experiments on the effectiveness of our rotation-aware camera alignment. The metrics are reported as the average on the Sparse Unbounded Drone dataset.

Method	$E_R(\text{med})$	$E_R(\mu)$	$E_T(\text{med})$	$E_T(\mu)$
w/o rotation points	0.791	0.793	0.0397	0.0377
ours	0.063	0.066	0.0067	0.0085

504 A.5.2 Evaluation metrics

505 We evaluate the novel view rendering quality based on the image and perceptual metrics, including
 506 PSNR, SSIM [28], and LPIPS [29]. We also propose to use DreamSim (DSIM) [30] as an additional
 507 metric, which is an ensemble method of different perceptual metrics [29, 38, 39, 40, 41, 42] fine-tuned
 508 for human visual perspectives.

509 Our coordinate alignment method is accurate but not perfect, leaving small residual pose shifts.
 510 However, this slight pixel offset should not reflect a significant difference in metrics, dominating

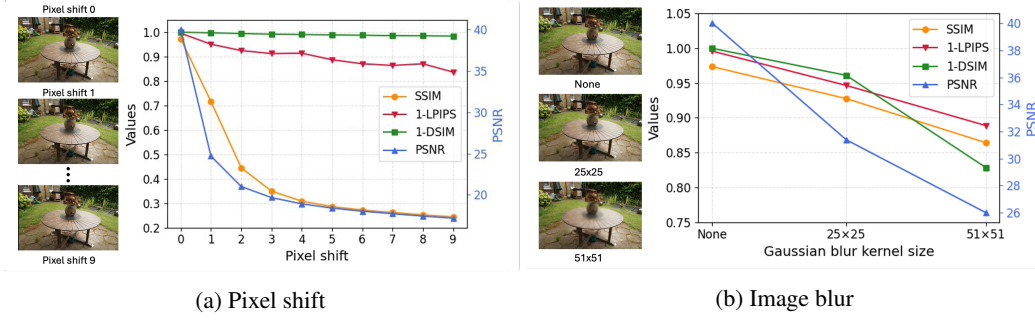


Figure 10: Evaluation of DSIM as metric

the quality assessment. As Fig. 10a shows, PSNR and SSIM drop steeply with a few pixel offsets, whereas DSIM remains almost flat. When images are dissimilar, where we add a blob of Gaussian blur at different kernel sizes in Fig. 10b to simulate semi-transparent Gaussians, DSIM shows a consistent decline as other metrics. This analysis indicates that DSIM is an appropriate metric for in-the-wild evaluations: it avoids over-penalising inevitable alignment errors while still capturing real perceptual degradation.

References

- [1] 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.
- [2] 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.
- [3] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5885–5894, 2021.
- [4] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490, 2022.
- [5] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022.
- [6] Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9065–9076, 2023.
- [7] Jiawei Yang, Marco Pavone, and Yue Wang. Freenerf: Improving few-shot neural rendering with free frequency regularization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8254–8263, 2023.
- [8] Mijeong Kim, Seonguk Seo, and Bohyung Han. Infonerf: Ray entropy minimization for few-shot neural volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12912–12921, 2022.
- [9] Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20775–20785, 2024.
- [10] Haolin Xiong, Sairisheek Muttukuru, Rishi Upadhyay, Pradyumna Chari, and Achuta Kadambi. Sparsegs: Real-time 360 $\{\deg\}$ sparse view synthesis using gaussian splatting. *arXiv preprint arXiv:2312.00206*, 2023.
- [11] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. In *European conference on computer vision*, pages 145–163. Springer, 2025.
- [12] Dongbin Zhang, Chuming Wang, Weitao Wang, Peihao Li, Minghan Qin, and Haoqian Wang. Gaussian in the wild: 3d gaussian splatting for unconstrained image collections. *arXiv preprint arXiv:2403.15704*, 2024.
- [13] 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.
- [14] 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.
- [15] Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12943–12952, 2022.

- [16] Yifan Yang, Shuhai Zhang, Zixiong Huang, Yubin Zhang, and Minghui 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.
- [17] Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024.
- [18] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3), 2022.
- [19] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9492–9502, 2024.
- [20] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024.
- [21] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.
- [22] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006.
- [23] Jaeyoung Chung, Jeongtaek Oh, and Kyoung Mu Lee. Depth-regularized optimization for 3d gaussian splatting in few-shot images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 811–820, 2024.
- [24] Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 245–254, 1985.
- [25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [26] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mipnerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022.
- [27] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4578–4587, 2021.
- [28] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612, 2004.
- [29] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.
- [30] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023.
- [31] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024.
- [32] John C Gower. Generalized procrustes analysis. *Psychometrika*, 40:33–51, 1975.

- 386 [33] Chen Quei-An. Nerf pl: a pytorch-lightning implementation of nerf. URL <https://github.com/kwea123/nerfpl>, 5, 2020.
- 387
- 388 [34] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander
- 389 Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, et al. Nerfstudio: A modular frame-
- 390 work for neural radiance field development. In *ACM SIGGRAPH 2023 Conference Proceedings*,
- 391 pages 1–12, 2023.
- 392 [35] Yuechen Zhang, Zexin He, Jinbo Xing, Xufeng Yao, and Jiaya Jia. Ref-npr: Reference-based
- 393 non-photorealistic radiance fields for controllable scene stylization. In *Proceedings of the*
- 394 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4242–4251, 2023.
- 395 [36] Kai Zhang, Nick Kolkin, Sai Bi, Fujun Luan, Zexiang Xu, Eli Shechtman, and Noah Snavely.
- 396 Arf: Artistic radiance fields. In *European Conference on Computer Vision*, pages 717–733.
- 397 Springer, 2022.
- 398 [37] Yiqun Mei, Jiacong Xu, and Vishal Patel. Regs: Reference-based controllable scene stylization
- 399 with gaussian splatting. *Advances in Neural Information Processing Systems*, 37:4035–4061,
- 400 2024.
- 401 [38] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. Image quality assessment: Unifying
- 402 structure and texture similarity. *IEEE transactions on pattern analysis and machine intelligence*,
- 403 44(5):2567–2581, 2020.
- 404 [39] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski,
- 405 and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings*
- 406 *of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.
- 407 [40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal,
- 408 Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual
- 409 models from natural language supervision. In *International conference on machine learning*,
- 410 pages 8748–8763. PmLR, 2021.
- 411 [41] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade
- 412 Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws
- 413 for contrastive language-image learning. In *Proceedings of the IEEE/CVF conference on*
- 414 *computer vision and pattern recognition*, pages 2818–2829, 2023.
- 415 [42] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked
- 416 autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on*
- 417 *computer vision and pattern recognition*, pages 16000–16009, 2022.