Supplementary Materials: View-consistent Object Removal in Radiance Fields

Anonymous Authors

1 DEPTH-BASED OCCLUSION CORRECTION

To get the set of $\{\mathbf{D}_{prior}\}$ used in Occlusion Correction, we have the following steps:

(1) Apply depth estimation method (i.e. Depth Anything[6]) on the original training images to get a set of depth $\{D\}$.
(2) As introduced in section 4.1, we align the $\{D\}$ with the sparse depth $\{D_{col}\}$ generated by Colmap to get a set of aligned depth $\{D_{ali}\}$.
(3) Utilize LaMa [4] to inpaint $\{D_{ali}\}$, and finally get a set of depth prior $\{\mathbf{D}_{prior}\}$.

Fig. 1 demonstrates the process to get depth prior. Since the estimated depth prior is not completely accurate, it can only be used as a reference. Therefore, we add a fault tolerance factor ϵ (as indicated in section 4.2, alg. 1) to loosen the constrain of depth prior. During projection, a point will be accepted if its depth is less than the prior or its depth is greater than the prior by a margin within the range of ϵ. A video result illustrating the effectiveness of our proposed depth-based occlusion correction is shown in the attached video.

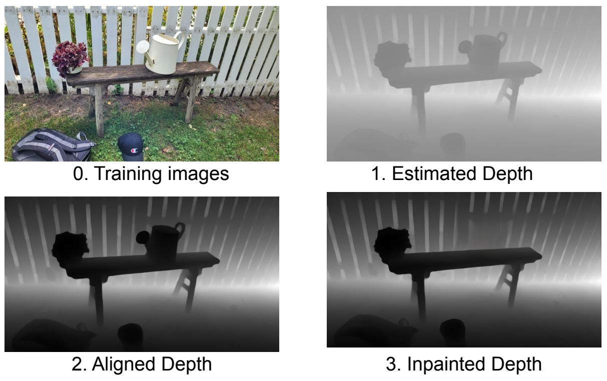

Figure 1: An illustration of the procedure to get depth prior.

2 ADDITIONAL RESULTS ON RADIANCE FIELD INPAINTING

Additional Radiance Field inpainting results of different scenes are shown in Fig. 2. However, we strongly recommend the reviewers to watch the demo video attached in the supplementary material, as figures alone do not fully capture the effectiveness of our proposed method. In the experiment, we observed that NeRFiller [5] (CVPR'24) experienced a ghost effect near the camera in our test scenes, leading to visually strange results. We followed the instructions and settings from their official GitHub repository[1], yet the issue persisted. Moreover, a similar ghost effect is noticeable in the demo video on their official website[2], suggesting this might be an inherent issue (possibily caused by nerfstudio). Despite the odd appearance caused by the ghost effect, the scene reconstruction is quite good, so we can still use them to compare with our approaches. We find that NeRFiller does maintain 3D consistency, but the quality of the inpainted content is not as good as ours.

[1]https://github.com/ethanweber/nerfiller

	LoFTR[3] Threshold				
Methods	**0.95**	**0.9**	**0.85**	**0.8**	**0.75**
SPIn-NeRF[1]	154.03	202.58	237.32	265.81	290.74
LaMask	105.79	147.32	177.78	203.0	225.91
ORNeRF[7]	34.48	52.41	67.16	80.50	931.82
NeRFiller[5]	201.34	258.18	295.89	325.77	350.87
Ours-GS	283.52	338.47	374.07	401.86	425.0
Ours-NeRF	**319.04**	**374.29**	**409.85**	**437.25**	**460.34**

Table 1: Number of correspondence measured by LoFTR, with different confidence level

3 INPAINTING CONSISTENCY.

In the main paper, we compared various baseline methods and our approachs with regard to the the number of matchings output by LoFTR [3] (with confidence level 0.95). Since the 0.95 threshold may be a little bit high, we decided to loosen the constrain and conduct the experiment again, additional quantitative results are shown in Table 1. The visualization of the matching results is also provided in Fig. 3, with LoFTR confidence level 0.9.

In addition, we demonstrated the matching results output by SuperGlue[2] in the video attached in the supplementary material. We followed the default setting to set the threshold for SuperGlue at 0.2, and made no further adjustments. This decision was based on the fact that SuperGlue is not as effective as LoFTR. Setting a higher threshold would result in no keypoint matchings within the inpainted area of the baseline methods.

4 MULTI-VIEW SEGMENTATION

We provide additional results of our proposed multi-view segmentation methods in Fig.4. Besides, we showcase a dynamic result of our segmentation method in the attached video. Thanks to the nature of our method, it has the capability to incorporate regions lacking semantic meaning (e.g. shadows), which may further improve the inpainting quality.

REFERENCES

[1] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G. Derpanis, Jonathan Kelly, Marcus A. Brubaker, Igor Gilitschenski, and Alex Levinshtein. 2023. SPIn-NeRF: Multiview Segmentation and Perceptual Inpainting with Neural Radiance Fields. In *CVPR*.

[2]https://ethanweber.me/nerfiller/

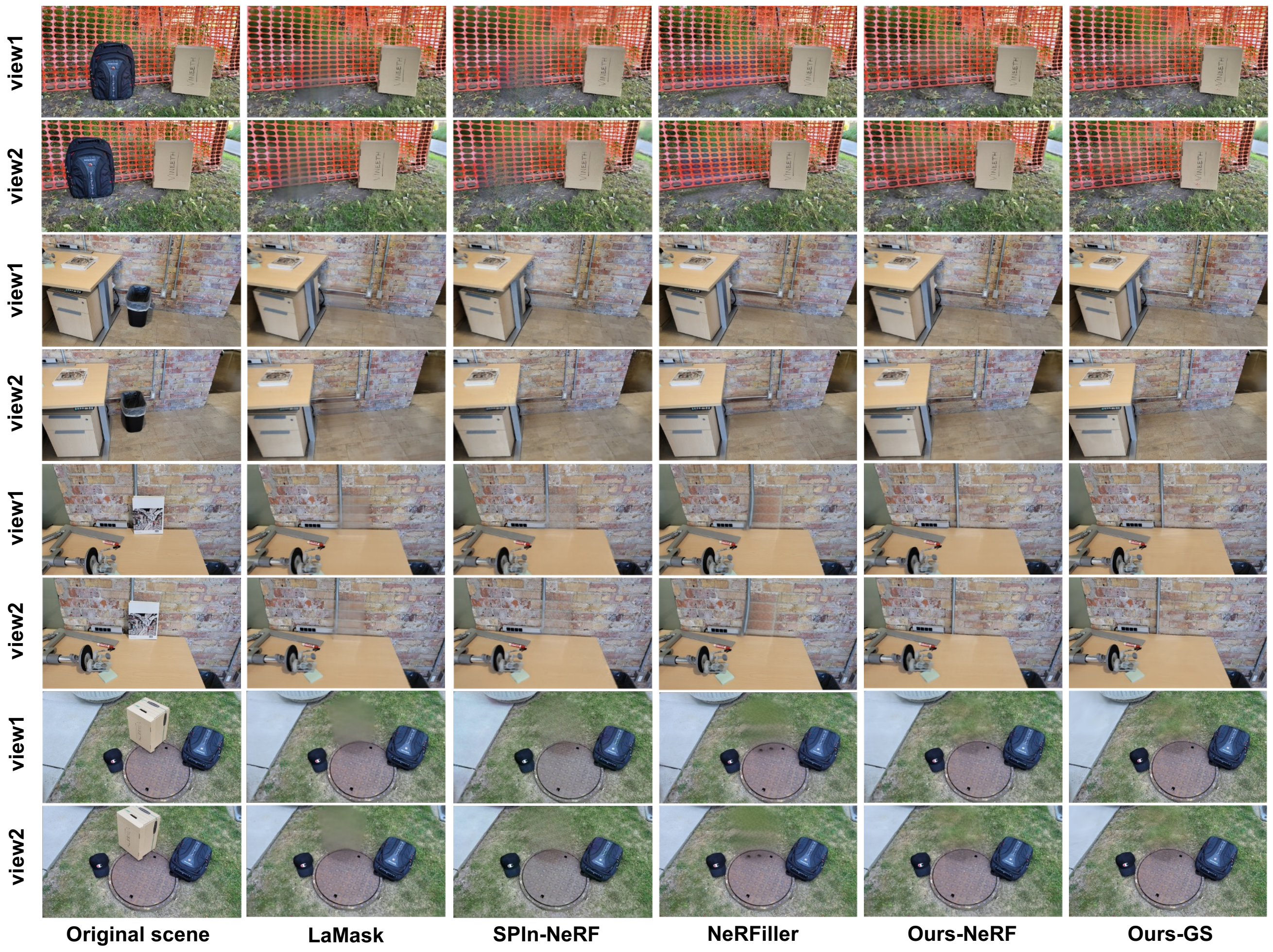

Figure 2: Additional examples of qualitative results.

[2] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. SuperGlue: Learning Feature Matching with Graph Neural Networks. In *CVPR*. https://arxiv.org/abs/1911.11763

[3] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. 2021. LoFTR: Detector-Free Local Feature Matching with Transformers. *CVPR* (2021).

[4] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. 2021. Resolution-robust Large Mask Inpainting with Fourier Convolutions. *arXiv preprint arXiv:2109.07161* (2021).

[5] Ethan Weber, Aleksander Holynski, Varun Jampani, Saurabh Saxena, Noah Snavely, Abhishek Kar, and Angjoo Kanazawa. 2024. NeRFiller: Completing Scenes via Generative 3D Inpainting. In *CVPR*.

[6] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. 2024. Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data. In *CVPR*.

[7] Youtan Yin, Zhoujie Fu, Fan Yang, and Guosheng Lin. 2023. OR-NeRF: Object Removing from 3D Scenes Guided by Multiview Segmentation with Neural Radiance Fields. arXiv:2305.10503 [cs.CV]

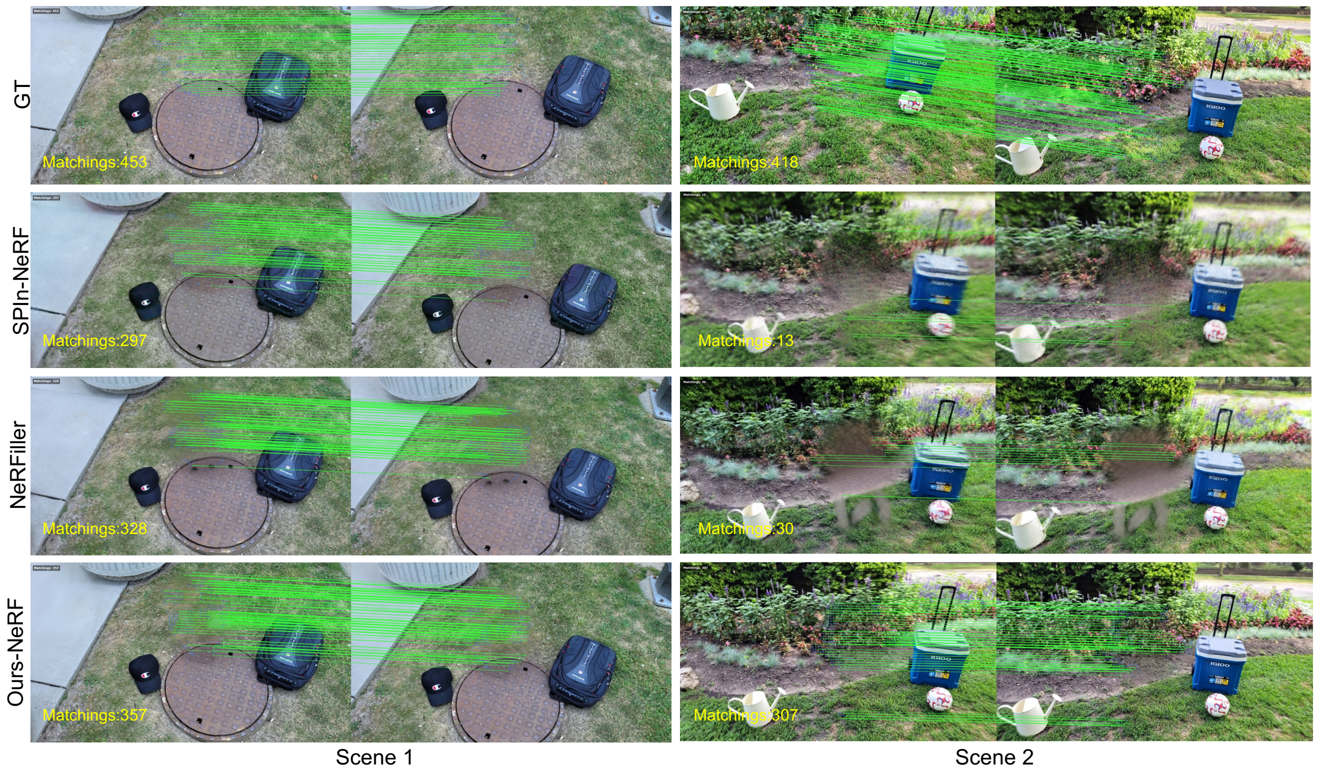

Figure 3: Additional examples of matching results between rendered image pairs.

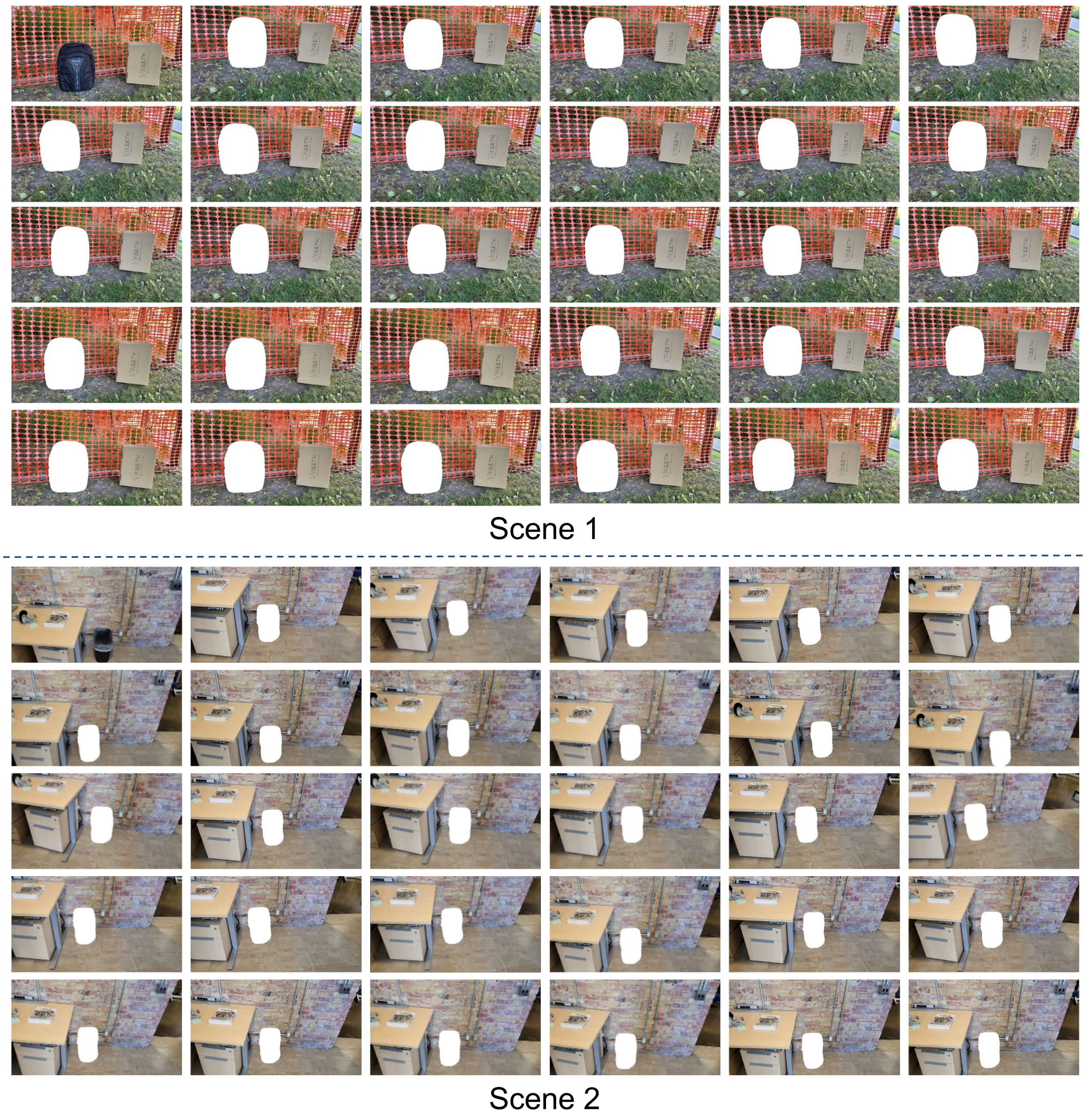

Figure 4: Additional examples of our proposed multi-view segmentation.