# RPBG: Towards Robust Neural Point-based Graphics in the Wild – Supplementary Material

## A Further Discussions

By extensive experiments, we have demonstrated the promising potential of point-based methods for NVS, especially the great generalizability and robustness of RPBG on varying scenes, with perceptually satisfactory rendering results. We make a visual comparison on *Museum* of T&T dataset [13] in Fig. 1 to demonstrate the varying sources of rendering artifacts and thus further discuss the fundamental differences between RF-based and point-based methods.

The privilege of adopting triangulated point clouds as the scene representation has been partially discussed in the main paper, as the points have contained all the verified co-visibility information across images. Besides, the point-based representation enables certain editing of the target scene, which is more difficult for RF-based methods. Please refer to Appendix C for the cases of scene editing.

We also consider the convolutional patch-wise rendering scheme by RPBG plays an important role in achieving perceptually good renderings. A similar ideology is explored in [29] by enforcing structural supervision on a group of rendered pixels.

As the framed area in Fig. 1(a) shows, the rendering noise is mainly caused by under-representation of RF, which is further due to the sparsity of input views (lack of ray intersections). RF-based methods aim to represent the target scene loyally, where each inquiry is supposed to be a frank reflection of local optical properties. In this way, RF-based methods render an image in pixels without considering the context information, establishing better pixel-to-pixel correspondence (thus higher PSNR).

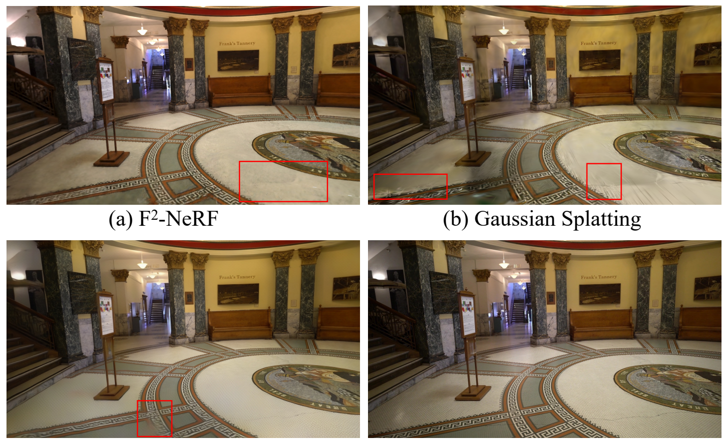
We would like to in particular mention a series of RF-based methods, *e.g.*, MVS-NeRF [7], DS-NeRF [8], Point-NeRF [30], DDP [19], which incorporate geometric prior information for optimizing NeRFs. They either adopt more explicit 3D proxies [7,30] than RFs, or enforce supervision on the rendered depth [8,19] to accelerate reconstruction or handle sparse views. However, their rendering scheme is still RF-based volume rendering, leaving the relevant drawbacks remain. For the readers' information, we also include the evaluation of DS-NeRF [8] in Tab. 2.

## B Dense Triangulation

We here elaborate the details of the dense triangulation procedure to obtain the point clouds.

Recall that the NVS datasets consist of images and corresponding camera parameters (intrinsics and extrinsics). To ensure the alignment between the poses and the reconstructed point clouds, we first triangulate sparse SIFT [14] points with COLMAP [21], where we only optimize the 3D coordinates, leaving the camera parameters frozen.

Based on the sparse triangulation, we follow the view selection strategy in [31], and choose 4 neighboring images with the best co-visibility for each image. Then we estimate a depth map for each image, by aid of the top-4 neighboring images, with



(a) F²-NeRF (b) Gaussian Splatting

(c) Ours (d) Ground Truth

**Fig. 1:** Comparison on *Museum* of T&T dataset [13] to showcase the typically different sources of noise due to the fundamental differences between different types of methods. Zoom in for best view.

AA-RMVSNet [28]. The per-view depth maps are filtered and fused to obtain the final 3D point cloud. We select AA-RMVSNet for its high memory-efficiency that allows a large batch size and RPBG is supposed to work fine with other off-the-self MVS methods.

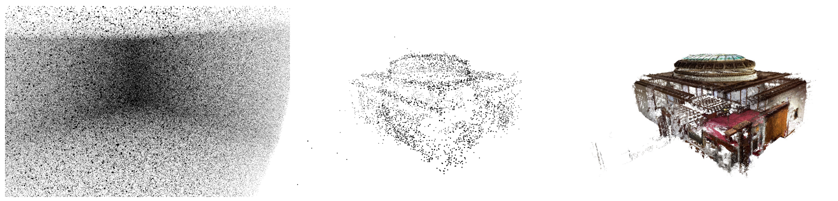
For the scene of *Building*, which consists of the most images among all the datasets (1940 images) we apply for quantitative experiments, the reconstruction can be done within one hour. With better engineering optimized algorithms, *e.g.*, OpenMVS [5], the point cloud densification can be even faster.

Note that we leverage a point cloud augmentation strategy to relax the requirements of triangulated points. More details will be covered in Appendix C.

# C Point Cloud

In addition to the ablation study, we provide some further analysis and results relevant to the point-based proxy RPBG adopts, including the effectiveness of the point cloud augmentation strategy, the analysis of RPBG applied with random initialized points, RPBG's additional properties of automatic handling dynamic objects and scene editing.

*Point Cloud Augmentation* The detailed augmentation steps are as Algorithm 1. On the scene of *Church*, we study the impact of such strategy applied on sparsely triangulated points multiple times. As is shown in Tab. 1, the first round of sampling and pruning brings the largest performance gain, and a larger gain is observed when applying to the SfM-initialized triangulation, which is much sparser compared to the MVS-initialized one. Note that the augmentation is optional and thus not performed on well triangulated scenes for the sake of time only.



(a) Random Points (b) Pruned Points (c) Reference MVS Reconstruction

**Fig. 2:** The randomly initialized point cloud can be pruned to the coarse scene geometry. The example is *Courtroom* from T&T dataset [13].

**Algorithm 1** Point Cloud Augmentation

**Input:** The point cloud $\{X\}$ to be augmented
**for** a given number of times **do**
    Sample one existing point $X = (x, y, z)$ randomly from $\{X\}$
    Form a 3D Gaussian distribution $G$ with its mean value $\mu = X$
    Sample a new point $X'$ from $G$
**end for**
Train RPBG with the point cloud $\{X\} \cup \{X'\}$
**for** each $X_i$ in $\{X'\}$ **do**
    Retrieve its neural texture $\mathbf{T}(X_i)$
    Approximate its pseudo density $\sigma_i = \sum |\mathbf{T}(X_i)|$
    **if** $\sigma_i < \sigma_{\text{threshold}}$ **then**
        Discard $X_i$ from $\{X'\}$
    **end if**
**end for**
**Output:** The augmented point cloud $\{X\} \cup \{X'\}$

*Random Point Cloud* Since we have demonstrated in the main paper that RPBG is able to perform re-rendering even with a randomly initialized point cloud taken as input. Empirically, we find that by applying the spatial pruning strategy to the random point cloud (by thresholding point-wise $\sigma > 180$ in this case), the point cloud shrinks to a shape similar to the actual geometry, as is illustrated in Fig. 2(b). It suggests that when neurally re-rendering, the network is able to implicitly verify the occupancy of each rasterized point and if a point is observed with poor multi-view consistency, it is more likely to be considered as an invalid point. The attempt of pruning random points is considered as an extreme case explaining how the point cloud augmentation strategy of RPBG manages to alleviate the problem of patchy or erroneous triangulation.

*Dynamic Objects* The RF-based methods are sensitive to dynamic objects and require either data pre-processing, *e.g.*, masking by manual labeling and semantic segmentation, or modeling of such ambiguity or uncertainty [20], to aid the RF's optimization. As for RPBG, the robustness against transient objects is trivially achieved since they are typically not reconstructed in SfM or MVS for not satisfying the static scene assumption. By experiments, we discover that RPBG is robust to such dynamic objects and able to automatically such objects in the training views when re-rendering (Fig. 3), which

**Table 1:** Quantitative metrics when applying the point cloud augmentation strategy on both the sparsely and the densely triangulated points for different times.

| *#Iters* | *SfM Init.* | | | *MVS Init.* | | |
|---|---|---|---|---|---|---|
| | PSNR↑ | SSIM↑ | LPIPS↓ | PSNR↑ | SSIM↑ | LPIPS↓ |
| 0 | 21.41 | 0.750 | 0.318 | 23.16 | 0.809 | 0.243 |
| 1 | 21.86 | 0.759 | 0.302 | 23.33 | 0.818 | 0.239 |
| 2 | 21.85 | 0.761 | 0.303 | 23.36 | 0.814 | 0.241 |
| $\varDelta_{0\rightarrow 2}$ | +0.45 | +0.011 | -0.015 | +0.20 | +0.005 | -0.002 |

(a) Re-rendering (b) Training View (c) Rasterized Points

**Fig. 3:** Automatic removal of dynamic objects with RPBG on self-collected data. The points of the dynamic objects are not triangulated for they do not meet the static scene assumption.

suggests that multi-view consistency is implicitly enforced during training and the renderer tends to restore the most consensual re-rendering.

*Scene Editing* As RPBG is a point-based pipeline, where an explicit 3D geometry is adopted for re-rendering, similar to previous point-based alternatives [1, 35], it allows certain scene editing and manipulation. We give two examples in Fig. 4. By removing the points, along with the point-bounded features, RPBG manages to re-render the edited scene, yet with some artifacts observed. It is because in RPBG, we enhance the context exchange among rasterized points by DAC, where each point does not solely represent its local optical property. Besides, it is also observed that FFC [10] may lead to repetitive artifacts at incomplete regions, as also can be found in the inpainted images by LaMa [22].

## D More Quantitative Results

*Traditional Reconstruction* For a more comprehensive comparison, we also include OpenMVS [5], as a traditional pipeline [2, 11, 25, 26] that reconstructs textured mesh

(a) Before Editing (b) After Editing

**Fig. 4:** Scene editing with RPBG on *DayaTemple* of GigaMVS dataset [33] and *Ballroom* of T&T dataset [13].

**Table 2:** Additional quantitative results on *Auditorium*, *Ballroom*, and *Courtroom* of T&T dataset [13]. The scores of $F^2$-NeRF [27], NPBG [1], Gaussian Splatting [12] and RPBG are provided for reference. PSNR↑/SSIM↑/LPIPS↓

| **Method** | ***Auditorium*** | ***Ballroom*** | ***Courtroom*** |
|---|---|---|---|
| OpenMVS [5] | 16.81/0.688/0.404 | 14.69/0.336/0.486 | 14.92/0.472/0.430 |
| DS-NeRF [8] | 16.29/0.542/0.612 | 14.74/0.668/0.570 | 14.62/0.491/0.616 |
| $F^2$-NeRF [27] | 20.36/0.843/0.329 | 22.21/0.706/0.328 | 20.13/0.672/0.425 |
| NPBG [1] | 22.05/0.814/0.375 | 21.04/0.681/0.330 | 20.99/0.681/0.386 |
| Gaussian Splatting [12] | 23.82/0.868/0.288 | 22.96/0.769/0.227 | 22.43/0.765/0.278 |
| **Ours** | 25.08/0.888/0.245 | 23.36/0.782/0.217 | 23.22/0.781/0.249 |

models to get rendered at arbitrary novel views. We compare the results on *Auditorium*, *Ballroom*, and *Courtroom* (Tab. 2) as they are inside-out scenes to avoid the negative impact of background.

*Geometry-bounded NeRF* In RPBG, the scene parameterization relies on the sparse/dense triangulation which incorporates estimated depth maps by SfM/MVS. To analogize this parameterization from the perspective of NeRF, we also evaluate DS-NeRF [8] on the aforementioned inside-out scenes in Tab. 2.

*Densely Captured Dataset* Though RPBG targets more generic scenes with casual settings, for the readers' information, we also evaluate RPBG with a densely captured dataset, NeRF-360 dataset [4], which is considered as ideal for training NVS, in Tab. 3. Note that mip-NeRF-360 [4] is particularly designed for such cases and takes about 6× time for training.

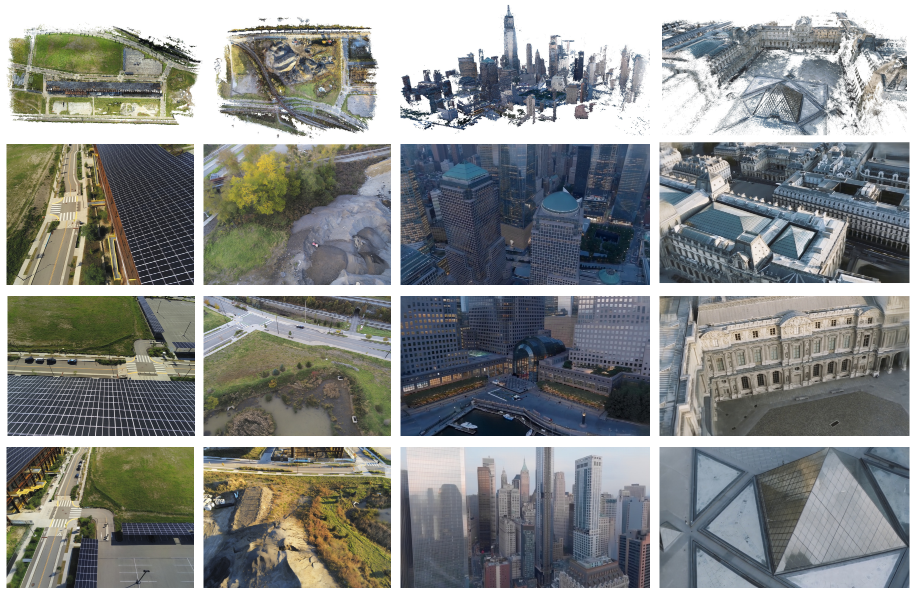



**Fig. 5:** Results of RPBG on aerial scenes, *i.e.*, Mill19 dataset [24] and OMMO dataset [15].

**Table 3:** Additional quantitative results on NeRF-360 dataset [4]. The provided methods [3, 4, 34] are typical unbounded NeRF variants.

| **Method** | GPU Hours | ***NeRF-360 outdoor*** PSNR↑ | SSIM↑ | LPIPS↓ | ***NeRF-360 indoor*** PSNR↑ | SSIM↑ | LPIPS↓ |
|---|---|---|---|---|---|---|---|
| mip-NeRF [3] | 22 | 22.65 | 0.505 | 0.484 | 26.98 | 0.798 | 0.360 |
| NeRF++ [34] | 66 | 23.77 | 0.585 | 0.401 | 28.05 | 0.836 | 0.309 |
| mip-NeRF-360 [4] | 48 | 25.92 | 0.747 | 0.244 | 31.72 | 0.917 | 0.180 |
| **Ours** | 8 | 24.72 | 0.709 | 0.252 | 28.76 | 0.898 | 0.140 |

*ScanNet++ Benchmark* We also evaluate RPBG on the public benchmark of ScanNet++ [32] (Novel View Synthesis on DSLR Images). ScanNet++ contains a wide variety of indoor scenes that are challenging for novel view synthesis for glossy and reflective materials and unseen poses captured independently of the training trajectory. The results are shown in Tab. 4. Note that the scores are all retrieved from the leaderboard. RPBG outperforms all the baselines listed by the benchmark, with a particular good perceptual quality (LPIPS).

## E More Qualitative Results

*Mill19 and OMMO Results* For the scenes in Mill19 [24] and OMMO [15], we provide the triangulated points and visualized re-renderings in Fig. 5. Since RPBG represents

**Table 4:** Benchmarking results on ScanNet++ [32]. The scores are retrieved by the evaluation system of the public benchmark.

| **Method** | PSNR↑ | SSIM↑ | LPIPS↓ |
|---|---|---|---|
| Nerfacto [23] | 24.05 | 0.861 | 0.342 |
| Instant-NGP [17] | 23.81 | 0.859 | 0.375 |
| Gaussian Splatting [12] | 23.89 | 0.871 | 0.319 |
| **Ours** | 24.36 | 0.873 | 0.280 |

the scene appearance with point-bounded features, it relieves users from partitioning large-scale data into smaller chunks, revealing the great scalability. Besides, the DAC module is well suited to capture periodic structures, which are common in human-made environments [22].

*ETH-MS Results* We also test RPBG's capability of handling super-large-scale scenes on ETH-MS dataset [9], which is for visual localization in AR applications. Its mapping set is captured by the 6-camera rig of a NavVis M6 mobile scanner, and contains 4914 images captured at the HG building of the campus of ETH Zurich, both in the main halls and on the sidewalk. The dataset is extremely challenging for NVS as its observations are very sparse and it exhibits many self-similarities and symmetric structures. The triangulated dense point cloud as well as three novel views absent in the training set is demonstrated in Fig. 6. Note that, RPBG also adopts the exactly identical settings without any partition of data. RPBG achieves visually pleasing results even when the scene is extremely complicated, indicating that our re-rendering is robust to point sparsity and occlusion.

## F Use of Existing Assets

We here list all the existing assets used in this manuscript and would like to sincerely appreciate the maintainers of these open-source projects:

- NeRF [16], NeRF++ [34], and TensoRF [6]: `https://github.com/ashawkey/torch-ngp`
- Mega-NeRF and Mill19 Dataset [24]: `https://github.com/cmusatyalab/mega-nerf`
- $F^2$-NeRF and Free Dataset [27]: `https://github.com/Totoro97/f2-nerf`
- NPBG [1] and NPBG++ [18]: `https://github.com/rakhimovv/npbgpp`
- Gaussian Splatting [12]: `https://github.com/graphdeco-inria/gaussian-splatting`
- COLMAP [21]: `https://colmap.github.io`
- OpenMVS [5]: `https://github.com/cdcseacave/openMVS`
- AA-RMVSNet [28]: `https://github.com/QT-Zhu/AA-RMVSNet`
- Tanks and Temples Benchmark [13]: `https://www.tanksandtemples.org`
- OMMO Dataset [15]: `https://ommo.luchongshan.com`
- GigaMVS Benchmark [33]: `https://www.gigavision.cn`

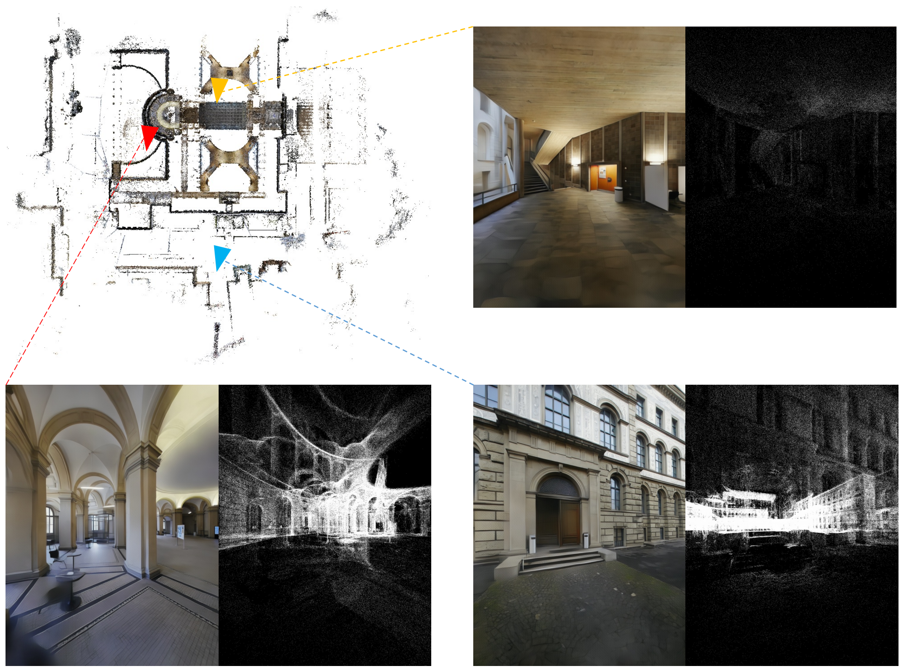


**Fig. 6:** Results of RPBG on ETH-MS dataset [9]. The location and orientation of the sampled cameras are marked with different colors in the densely triangulated point cloud respectively.

- ScanNet++ Benchmark [32]: `https://kaldir.vc.in.tum.de/scannetpp/benchmark/nvs`
- ETH-MS Dataset [9]: `https://github.com/cvg/visloc-iccv2021`

## References


1. Aliev, K.A., Sevastopolsky, A., Kolos, M., Ulyanov, D., Lempitsky, V.: Neural point-based graphics. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16. pp. 696–712. Springer (2020) 4, 5, 7
2. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: A randomized correspondence algorithm for structural image editing. ACM Trans. Graph. **28**(3), 24 (2009) 4
3. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5855–5864 (2021) 6
4. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5470–5479 (2022) 5, 6
5. Cernea, D.: OpenMVS: Multi-view stereo reconstruction library (2020), https://cdcseacave.github.io/openMVS 2, 4, 5, 7
6. Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H.: Tensorf: Tensorial radiance fields. In: European Conference on Computer Vision. pp. 333–350. Springer (2022) 7
7. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14124–14133 (2021) 1
8. Deng, K., Liu, A., Zhu, J.Y., Ramanan, D.: Depth-supervised nerf: Fewer views and faster training for free. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12882–12891 (2022) 1, 5
9. ETH Zurich Computer Vision Group and Microsoft Mixed Reality & AI Lab Zurich: The ETH-Microsoft Localization Dataset. https://github.com/cvg/visloc-iccv2021 (2021) 7, 8
10. Fuoli, D., Van Gool, L., Timofte, R.: Fourier space losses for efficient perceptual image super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2360–2369 (2021) 4
11. Jancosek, M., Pajdla, T.: Exploiting visibility information in surface reconstruction to preserve weakly supported surfaces. International scholarly research notices **2014** (2014) 4
12. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics (TOG) **42**(4), 1–14 (2023) 5, 7
13. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Transactions on Graphics (ToG) **36**(4), 1–13 (2017) 1, 2, 3, 5, 7
14. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision **60**, 91–110 (2004) 1
15. Lu, C., Yin, F., Chen, X., Chen, T., Yu, G., Fan, J.: A large-scale outdoor multi-modal dataset and benchmark for novel view synthesis and implicit scene reconstruction. arXiv preprint arXiv:2301.06782 (2023) 6, 7
16. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European Conference on Computer Vision. pp. 405–421 (2020) 7
17. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Transactions on Graphics (ToG) **41**(4), 1–15 (2022) 7
18. Rakhimov, R., Ardelean, A.T., Lempitsky, V., Burnaev, E.: Npbg++: Accelerating neural point-based graphics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15969–15979 (2022) 7

19. Roessle, B., Barron, J.T., Mildenhall, B., Srinivasan, P.P., Nießner, M.: Dense depth priors for neural radiance fields from sparse input views. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12892–12901 (2022) 1
20. Sabour, S., Vora, S., Duckworth, D., Krasin, I., Fleet, D.J., Tagliasacchi, A.: Robustnerf: Ignoring distractors with robust losses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20626–20636 (2023) 3
21. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016) 1, 7
22. Suvorov, R., Logacheva, E., Mashikhin, A., Remizova, A., Ashukha, A., Silvestrov, A., Kong, N., Goka, H., Park, K., Lempitsky, V.: Resolution-robust large mask inpainting with fourier convolutions. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 2149–2159 (2022) 4, 7
23. Tancik, M., Weber, E., Ng, E., Li, R., Yi, B., Wang, T., Kristoffersen, A., Austin, J., Salahi, K., Ahuja, A., et al.: Nerfstudio: A modular framework for neural radiance field development. In: ACM SIGGRAPH 2023 Conference Proceedings. pp. 1–12 (2023) 7
24. Turki, H., Ramanan, D., Satyanarayanan, M.: Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12922–12931 (2022) 6, 7
25. Vu, H.H., Labatut, P., Pons, J.P., Keriven, R.: High accuracy and visibility-consistent dense multiview stereo. IEEE transactions on pattern analysis and machine intelligence **34**(5), 889–901 (2011) 4
26. Waechter, M., Moehrle, N., Goesele, M.: Let there be color! large-scale texturing of 3d reconstructions. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 836–850. Springer (2014) 4
27. Wang, P., Liu, Y., Chen, Z., Liu, L., Liu, Z., Komura, T., Theobalt, C., Wang, W.: F2-nerf: Fast neural radiance field training with free camera trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4150–4159 (2023) 5, 7
28. Wei, Z., Zhu, Q., Min, C., Chen, Y., Wang, G.: Aa-rmvsnet: Adaptive aggregation recurrent multi-view stereo network. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6187–6196 (2021) 2, 7
29. Xie, Z., Yang, X., Yang, Y., Sun, Q., Jiang, Y., Wang, H., Cai, Y., Sun, M.: S3im: Stochastic structural similarity and its unreasonable effectiveness for neural fields. arXiv preprint arXiv:2308.07032 (2023) 1
30. Xu, Q., Xu, Z., Philip, J., Bi, S., Shu, Z., Sunkavalli, K., Neumann, U.: Point-nerf: Point-based neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5438–5448 (2022) 1
31. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: Mvsnet: Depth inference for unstructured multi-view stereo. In: Proceedings of the European conference on computer vision (ECCV). pp. 767–783 (2018) 1
32. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023) 6, 7, 8
33. Zhang, J., Zhang, J., Mao, S., Ji, M., Wang, G., Chen, Z., Zhang, T., Yuan, X., Dai, Q., Fang, L.: Gigamvs: a benchmark for ultra-large-scale gigapixel-level 3d reconstruction. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(11), 7534–7550 (2021) 5, 7
34. Zhang, K., Riegler, G., Snavely, N., Koltun, V.: Nerf++: Analyzing and improving neural radiance fields. arXiv preprint arXiv:2010.07492 (2020) 6, 7
35. Zhang, Y., Peng, S., Moazeni, A., Li, K.: Papr: Proximity attention point rendering. Advances in Neural Information Processing Systems **36** (2024) 4