


# PlenopticPoints: Supplemental Material

Florian Hahlbohm 

Moritz Kappel 

Jan-Philipp Tauscher 

Martin Eisemann 

Marcus Magnor 

Institut für Computergraphik, TU Braunschweig, Germany  
 {lastname}@cg.cs.tu-bs.de

## 1. Model Details

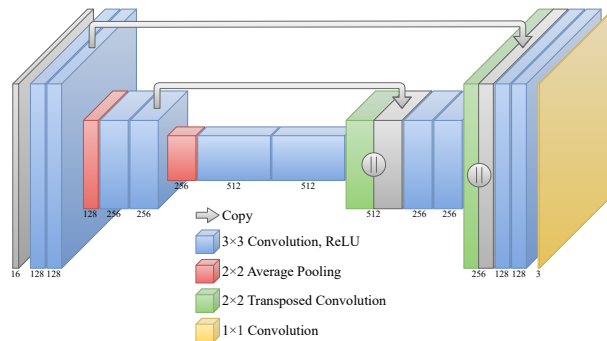
**Pulsar Configuration.** To compute the values for each pixel, Pulsar [LZ21] employs a differentiable blending function that associates a blending weight  $b_i$  with each sphere:

$$b_i = \frac{o_i \cdot d_i \cdot \exp(o_i \cdot \frac{z_i}{\gamma})}{\exp(\frac{\epsilon}{\gamma}) + \sum_k o_k \cdot d_k \cdot \exp(o_k \cdot \frac{z_k}{\gamma})} \quad (1)$$

Where  $o_i$  is the opacity value of each sphere,  $z_i \in [0, 1]$  is the depth value in normalized device space,  $d_i$  is the normalized orthogonal distance of the ray to the sphere center, and  $\gamma$  is a scaling factor used to make the scene more rigorous with respect to depth. During the optimization, we exponentially decay the  $\gamma$  parameter from 0.1 to 0.01 with the final value being used during inference. We choose the same radius  $r = 0.0015$  for every sphere and empirically confirm the validity of this choice. The value originates from the idea that spheres should roughly have the size of a pixel if they are located in the center of a scene with common size. Otherwise, we mostly use Pulsar’s default parameters with the following configurations for improved results. We set the number of spheres per pixel receiving a gradient to 64 and use 0.001 for the parameter defining the maximum allowed difference in color space for each pixel. Regarding the background color, we use a fixed value of 1.0 for each of the 16 output channels.

**U-Net Architecture.** As described in the main paper, we make three configurations to the original U-Net architecture [RFB15]: Three instead of five down-/upsampling layers, no batch-normalization layers, and average instead of maximum pooling when downsampling. For the first modification, we omit the first down-/upsampling layer as our input feature map has 16 channels (in contrast to the single channel used with the original architecture) and thus the direct conversion to 128 channels is a feasible solution to reduce computational requirements without loss of quality. Additionally, we omit the last down-/upsampling layer as we find that it not only increases computational requirements but, most importantly, reduces the network’s generalization capabilities due to overfitting. We visualize the resulting architecture in Figure 1. Moreover, we experiment with using *single convolutions* [ZD23] or *gated convolutions* [YLY\*19, RFS22] instead of the *double convolutions* used by Ronneberger et al. [RFB15] Note that the terms *single convolution*, *gated convolution*, and *double convolution* describe the core architecture of each down-

/upsampling layer. For our method, we found that *single convolutions* [ZD23] are faster but result in lower image quality while *gated convolutions* [YLY\*19, RFS22] are slower and do not improve image quality. We hypothesize the holes in our feature map to have different properties compared to [ZD23] and [RFS22] because of distinct initialization strategies and therefore stick with *double convolutions*.

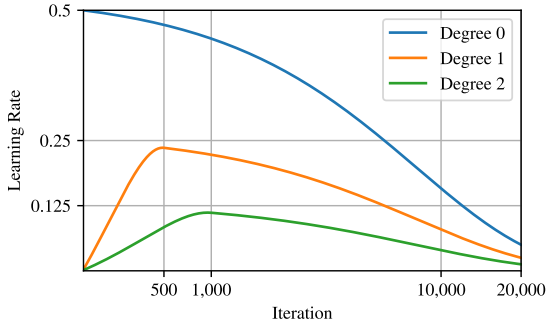


**Figure 1:** The architecture resulting from our modifications to the original U-Net [RFB15]. We use average instead of maximum pooling and omit batch normalization layers while employing shorter contracting and expansive paths.

**Spherical Harmonics Learning Rate Delay.** We use a warm-up strategy with spherical harmonic (SH) degree-specific learning rates and delays. In Figure 2, we visualize the resulting learning rate schedule for the three SH degrees used by our model. Our implementation is based on previous work [BMT\*21].

## 2. Additional Results and Comparisons

We provide per-scene results for the synthetic and real-world scenes to enable in-depth comparisons of PlenopticPoints with relevant baselines. Most importantly, we include results of multiple point-based methods. We omit these in the main paper, as they are unable to match the image quality of state-of-the-art methods for single-object scenes. Instead they achieve different improvements such as faster rendering performance or reduced model size. Furthermore, we include qualitative results for all synthetic and real-



**Figure 2:** Learning rate delay strategy for different degrees of spherical harmonic (SH) functions. Initial learning rates are set to 0.5/0.25/0.125 and exponentially decayed to 0.05/0.025/0.0125 respectively. The respective delay of 0/500/1000 iterations is implemented as described in [BMT\*21].

world scenes we used for evaluation of our method in Figure 4 and Figure 5 respectively.

**Synthetic Experiment.** Table 5, Table 6, and Table 7 show per-scene results for the synthetic dataset by Mildenhall et al. [MST\*20] in terms of PSNR, SSIM [WBSS04], and LPIPS [ZIE\*18]. This includes the per-scene results of JaxNeRF [DBS20], Plenoxels [FKYT\*22], Instant-NGP [MESK22], and TensoRF [CXG\*22], which we computed using the setup described in the main paper. Additionally, we include the results of numerous other recent approaches, for which we provide the source of the reported values in the tables and roughly sort the approaches by their publication date. Note that this collection of results is likely incomplete, but may serve as an overview of approaches evaluated on the synthetic dataset from NeRF [MST\*20]. Among the additions are the original implementation of NeRF [MST\*20], Mip-NeRF [BMT\*21], and Ref-NeRF [VHM\*22], all of which are comparatively slow both with respect to optimization time and frames per second during inference due to their MLP-based implementation. We also include the MLP-based Point-NeRF [XXP\*22], which is faster in terms of optimization time but employs expensive volume rendering and is therefore still comparatively slow during inference. In terms of voxel grid-based methods we add PlenOctrees [YLT\*21], SNeRG [HSM\*21], and DVGO [SSC22]. Moreover, we include results of various point-based methods, for which we omit comparisons in the main paper, as they are unable to match the image quality of state-of-the-art methods. Instead, they make significant contributions with respect to multiple other highly-relevant aspects such as optimization time, model size, and inference frame rate. Namely, these point-based approaches are NPBG [ASK\*20], a method by Zhang et al. [ZBRH22], and SNP [ZD23]. Lastly, we include influential approaches that do not fit the aforementioned categories in Neural Volumes [LSS\*19], Scene Representation Networks [SZW19], LLFF [MSOC\*19], NSVF [LGL\*20], and MobileNeRF [CFHT23].

**Real-World Experiment.** For the sake of completeness and reproducibility, we include an exemplary image and the sequence number for each of the four selected scenes from the CO3D

dataset [RSH\*21] in Figure 3. Per-scene results in terms of PSNR,



**Figure 3:** Exemplary images for the selected scenes from the CO3D dataset [RSH\*21]. We choose four scenes of which each one is challenging for different reasons. CO3D sequence numbers from left to right: 590\_88941\_178115, 250\_26779\_55023, 618\_100125\_200129, and 34\_1479\_4753.

SSIM [WBSS04], and LPIPS [ZIE\*18] are included in Table 1, Table 2, and Table 3. Here, we do not include results of additional baselines apart from the ones used in the main paper. The used setup for each baseline is described in the main paper.

Method	Car	Hydrant	Plant	Teddybear	Mean
JaxNeRF [DBS20]	30.61	<b>37.36</b>	27.91	<b>35.42</b>	32.83
Plenoxels [FKYT*22]	29.41	35.88	19.11	32.06	29.12
Instant-NGP [MESK22]	30.44	34.62	20.74	34.83	30.16
TensoRF [CXG*22]	29.64	37.29	27.53	33.29	31.94
Ours	<b>30.95</b>	36.66	<b>28.61</b>	35.17	<b>32.85</b>

**Table 1:** Per-scene test set PSNRs on the four selected scenes from the CO3D dataset [RSH\*21].

Method	Car	Hydrant	Plant	Teddybear	Mean
JaxNeRF [DBS20]	0.950	<b>0.989</b>	0.941	0.979	0.965
Plenoxels [FKYT*22]	0.941	0.986	0.880	0.968	0.944
Instant-NGP [MESK22]	0.951	0.985	0.881	0.979	0.949
TensoRF [CXG*22]	0.945	0.988	0.936	0.969	0.960
Ours	<b>0.960</b>	<b>0.989</b>	<b>0.952</b>	<b>0.980</b>	<b>0.970</b>

**Table 2:** Per-scene test set SSIMs on the four selected scenes from the CO3D dataset [RSH\*21].

### 3. Point Cloud Shape Analysis

In this section, we provide in-depth analysis of the point clouds generated and used by our method. We show statistics for both the synthetic and real-world scenes in Table 4. As Mildenhall et al. provide the meshes they used to create the Realistic Synthetic 360° dataset [MST\*20], we are able to generate reference point clouds for these scenes. For each scene, we randomly sample  $1024^3$  points on its mesh and round the positions to the vertices of the regular grid resulting from our initialization procedure. We quantitatively compare the initial and final point clouds  $\hat{P}$  to these reference point clouds  $P$  using an intersection over union (IoU) metric:

$$\text{IoU}(P, \hat{P}) = \frac{|P \cap \hat{P}|}{|P \cup \hat{P}|} \quad (2)$$

From these statistics as well as additional qualitative analyses, we can infer multiple things about the geometric accuracy of the point clouds. Firstly, the initial visual hull-based point clouds are

Method	Car	Hydrant	Plant	Teddybear	Mean
JaxNeRF [DBS20]	0.060	0.015	0.116	0.037	0.057
Plenoxels [FKYT*22]	0.074	0.023	0.204	0.072	0.093
Instant-NGP [MESK22]	0.057	0.078	0.266	0.038	0.110
TensoRF [CXG*22]	0.064	0.017	0.126	0.056	0.066
Ours	<b>0.035</b>	<b>0.011</b>	<b>0.068</b>	<b>0.027</b>	<b>0.035</b>

**Table 3:** Per-scene test set LPIPS scores on the four selected scenes from the CO3D dataset [RSH\*21].

far from optimal as shown by the IoU metric (see Table 4). Secondly, our method is able to remove many incorrect points during the optimization. Thirdly, the final point clouds produced by our method are still vastly different from the reference point clouds. Qualitatively, we observe that this is mostly due to points inside of objects not being removed during the optimization. Note that a learned representation does not necessarily work better when given the reference point clouds compared to creating one during its optimization.

## References

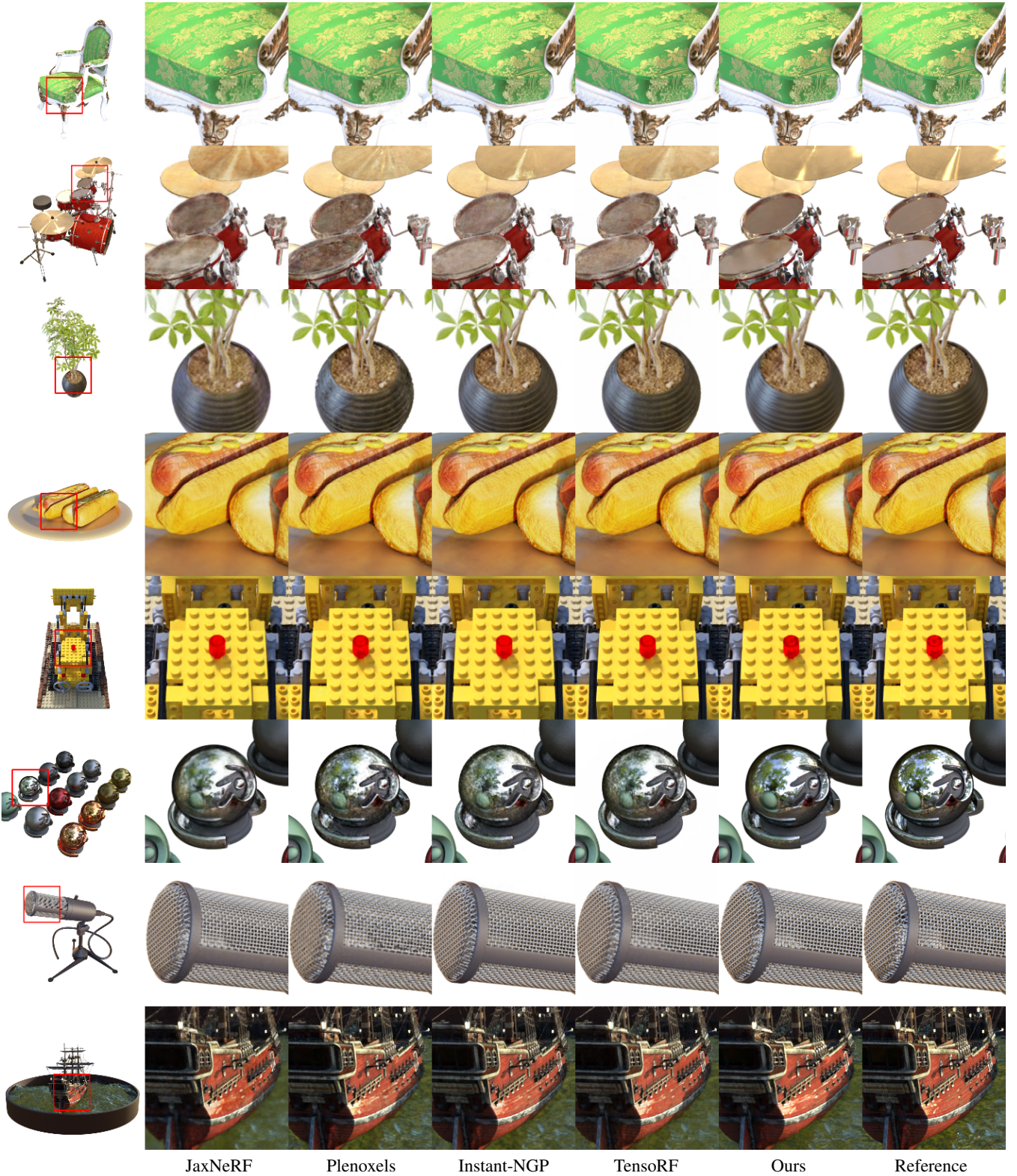
- [ASK\*20] ALIEV K.-A., SEVASTOPOLSKY A., KOLOS M., ULYANOV D., LEMPITSKY V.: Neural point-based graphics. In *ECCV* (2020). doi:10.1007/978-3-030-58542-6\_42. 2, 4, 6, 7
- [BMT\*21] BARRON J. T., MILDENHALL B., TANCİK M., HEDMAN P., MARTIN-BRUALLA R., SRINIVASAN P. P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. *ICCV* (2021). doi:10.1109/ICCV48922.2021.00580. 1, 2, 4, 6, 7
- [CFHT23] CHEN Z., FUNKHOUSER T., HEDMAN P., TAGLIASACCHI A.: Mobilenerf: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. In *CVPR* (2023). doi:10.1109/CVPR52729.2023.01590. 2, 4, 6, 7
- [CXG\*22] CHEN A., XU Z., GEIGER A., YU J., SU H.: TensoRF: Tensorial radiance fields. In *ECCV* (2022). doi:10.1007/978-3-031-19824-3\_20. 2, 3, 4, 6, 7
- [DBS20] DENG B., BARRON J. T., SRINIVASAN P. P.: JaxNeRF: an efficient JAX implementation of NeRF, 2020. URL: <https://github.com/google-research/google-research/tree/master/jaxnerf>. 2, 3, 4, 6, 7
- [FKYT\*22] FRIDOVICH-KEIL S., YU A., TANCİK M., CHEN Q., RECHT B., KANAZAWA A.: Plenoxels: Radiance fields without neural networks. In *CVPR* (2022). doi:10.1109/CVPR52688.2022.00542. 2, 3, 4, 6, 7
- [HSM\*21] HEDMAN P., SRINIVASAN P. P., MILDENHALL B., BARRON J. T., DEBEVEC P.: Baking neural radiance fields for real-time view synthesis. *ICCV* (2021). doi:10.1109/ICCV48922.2021.00582. 2, 4, 6, 7
- [LGL\*20] LIU L., GU J., LIN K. Z., CHUA T.-S., THEOBALT C.: Neural sparse voxel fields. *NeurIPS* (2020). doi:10.48550/arXiv.2007.11571. 2, 4, 6, 7
- [LSS\*19] LOMBARDI S., SIMON T., SARAGIH J., SCHWARTZ G., LEHRMANN A., SHEIKH Y.: Neural volumes: Learning dynamic renderable volumes from images. *ACM TOG* (2019). doi:10.1145/3306346.3323020. 2, 4, 6, 7
- [LZ21] LASSNER C., ZOLLHOFER M.: Pulsar: Efficient sphere-based neural rendering. In *CVPR* (2021). doi:10.1109/CVPR46437.2021.00149. 1
- [MESK22] MÜLLER T., EVANS A., SCHIED C., KELLER A.: Instant neural graphics primitives with a multiresolution hash encoding. *ACM TOG* (2022). doi:10.1145/3528223.3530127. 2, 3, 4, 6, 7
- [MSOC\*19] MILDENHALL B., SRINIVASAN P. P., ORTIZ-CAYON R., KALANTARI N. K., RAMAMOORTHY R., NG R., KAR A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM TOG* (2019). doi:10.1145/3306346.3322980. 2, 4, 6, 7
- [MST\*20] MILDENHALL B., SRINIVASAN P. P., TANCİK M., BARRON J. T., RAMAMOORTHY R., NG R.: NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV* (2020). doi:10.1145/3503250. 2, 4, 5, 6, 7
- [RFB15] RONNEBERGER O., FISCHER P., BROX T.: U-net: Convolutional networks for biomedical image segmentation. In *MICCAI* (2015). doi:10.1007/978-3-319-24574-4\_28. 1
- [RFS22] RÜCKERT D., FRANKE L., STAMMINGER M.: Adop: Approximate differentiable one-pixel point rendering. *ACM TOG* (2022). doi:10.1145/3528223.3530122. 1
- [RSH\*21] REIZENSTEIN J., SHAPOVALOV R., HENZLER P., SBORDONE L., LABATUT P., NOVOTNY D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *ICCV* (2021). doi:10.1109/ICCV48922.2021.01072. 2, 3, 6
- [SSC22] SUN C., SUN M., CHEN H.: Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *CVPR* (2022). doi:10.1109/CVPR52688.2022.00538. 2, 4, 6, 7
- [SZW19] SITZMANN V., ZOLLHÖFER M., WETZSTEIN G.: Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *NeurIPS* (2019). doi:10.48550/arXiv.1906.01618. 2, 4, 6, 7
- [VHM\*22] VERBIN D., HEDMAN P., MILDENHALL B., ZICKLER T., BARRON J. T., SRINIVASAN P. P.: Ref-NeRF: Structured view-dependent appearance for neural radiance fields. *CVPR* (2022). doi:10.1109/CVPR52688.2022.00541. 2, 4, 6, 7
- [WBSS04] WANG Z., BOVIK A. C., SHEIKH H. R., SIMONCELLI E. P.: Image quality assessment: from error visibility to structural similarity. *IEEE TIP* (2004). doi:10.1109/TIP.2003.819861. 2
- [XXP\*22] XU Q., XU Z., PHILIP J., BI S., SHU Z., SUNKAVALLI K., NEUMANN U.: Point-nerf: Point-based neural radiance fields. In *CVPR* (2022). doi:10.1109/CVPR52688.2022.00536. 2, 4, 6, 7
- [YLT\*21] YU A., LI R., TANCİK M., LI H., NG R., KANAZAWA A.: PlenOctrees for real-time rendering of neural radiance fields. In *ICCV* (2021). doi:10.1109/ICCV48922.2021.00570. 2, 4, 6, 7
- [YLY\*19] YU J., LIN Z., YANG J., SHEN X., LU X., HUANG T. S.: Free-form image inpainting with gated convolution. In *ICCV* (2019). doi:10.1109/ICCV.2019.00457. 1
- [ZBRH22] ZHANG Q., BAEK S.-H., RUSINKIEWICZ S., HEIDE F.: Differentiable point-based radiance fields for efficient view synthesis. *SIGGRAPH Asia* (2022). doi:10.1145/3550469.3555413. 2, 4, 6, 7
- [ZD23] ZUO Y., DENG J.: View synthesis with sculpted neural points. In *ICLR* (2023). doi:10.48550/arXiv.2205.05869. 1, 2, 4, 6, 7
- [ZIE\*18] ZHANG R., ISOLA P., EFROS A. A., SHECHTMAN E., WANG O.: The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR* (2018). doi:10.1109/CVPR.2018.00068. 2

Scene	Grid Resolution	Before Optimization		After Optimization		Reference #Points
		#Points	IoU	#Points	IoU	
Chair	(254, 258, 374)	4,056,338	0.101	1,529,012	0.129	420,091
Drums	(483, 370, 319)	4,041,106	0.151	275,909	0.133	609,984
Ficus	(261, 409, 610)	4,043,133	0.130	2,611,687	0.129	527,438
Hotdog	(352, 355, 109)	4,069,466	0.088	686,584	0.068	458,012
Lego	(194, 345, 232)	4,040,533	0.266	676,830	0.266	1,076,451
Materials	(435, 351, 89)	4,064,838	0.236	2,212,395	0.163	961,130
Mic	(517, 512, 493)	3,234,306	0.150	2,790,702	0.115	483,825
Ship	(281, 281, 193)	4,039,149	0.065	479,796	0.158	270,164
Car	(344, 144, 173)	4,038,196		2,277,090		
Hydrant	(165, 354, 204)	4,041,107	N/A	3,817,755	N/A	N/A
Plant	(180, 281, 242)	4,040,173		1,915,470		
Teddybear	(228, 292, 193)	4,044,900		2,800,520		

**Table 4:** We report per-scene statistics regarding the size and shape of the point clouds generated by our method. As we are unable to generate a suitable reference point cloud for the real-world scenes, the IoU metric is not applicable.

Method	Source	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic	Ship	Mean
LLFF [MSOC* 19]	[MST* 20]	28.72	21.13	21.79	31.41	24.54	20.72	27.48	23.22	24.88
NV [LSS* 19]	[MST* 20]	28.33	22.58	24.79	30.71	26.08	24.22	27.78	23.93	26.05
SRN [SZW19]	[MST* 20]	26.96	17.18	20.73	26.81	20.85	18.09	26.85	20.60	22.26
NPBG [ASK* 20]	[XXP* 22]	26.47	21.53	24.60	29.01	24.84	21.58	26.62	21.83	24.56
NeRF [MST* 20]	Publication	33.00	25.01	30.13	36.18	32.54	29.62	32.91	28.65	31.01
JaxNeRF [DBS20]	Ours	34.07	25.03	30.43	36.86	33.27	29.90	34.53	29.35	31.68
NSVF [LGL* 20]	[BMT* 21]	33.19	25.18	31.23	37.14	32.29	32.68	34.27	27.93	31.74
Mip-NeRF [BMT* 21]	Publication	35.14	25.48	33.29	37.48	35.7	30.71	36.51	30.41	33.09
PlenOctrees [YLT* 21]	Publication	34.66	25.31	30.79	36.79	32.95	29.76	33.97	29.42	31.71
SNeRG [HSM* 21]	Publication	33.24	24.57	29.32	34.33	33.82	27.21	32.60	27.97	30.38
Plenoxels [FKYT* 22]	Ours	33.96	25.34	31.83	36.40	34.08	29.13	33.26	29.60	31.70
DVGO [SSC22]	Publication	34.09	25.44	32.78	36.74	34.64	29.57	33.20	29.13	31.95
Ref-NeRF [VHM* 22]	Publication	<b>35.83</b>	25.79	33.91	<b>37.72</b>	36.25	<b>35.41</b>	<b>36.76</b>	30.28	<b>33.99</b>
Point-NeRF [XXP* 22]	Publication	35.40	<b>26.06</b>	<b>36.13</b>	37.30	35.04	29.61	35.95	<b>30.97</b>	33.31
Instant-NGP [MESK22]	Ours	35.75	25.90	33.09	37.35	36.12	29.57	35.13	30.10	32.88
TensorRF [CXG* 22]	Ours	35.67	25.99	33.95	37.30	<b>36.35</b>	30.10	34.53	30.74	33.08
Zhang et al. [ZBRH22]	Publication	32.98	25.53	29.01	34.56	31.33	28.01	33.82	27.01	30.28
MobileNeRF [CFHT23]	Publication	34.09	25.02	30.20	35.46	34.18	26.72	32.48	29.06	30.90
SNP [ZD23]	Publication	30.49	22.78	25.43	33.24	27.94	26.02	28.80	25.07	27.47
Ours	Ours	35.35	25.59	32.68	37.18	33.00	29.27	35.07	28.40	32.07

**Table 5:** Per-scene test set PSNRs on the Realistic Synthetic 360° dataset [MST\* 20].



**Figure 4:** Qualitative results for the Realistic Synthetic 360° dataset [MST\*20]. We select a viewpoint from the test set of each scene and show the output of each baseline. We would like to highlight the more accurately represented drum heads as well as the less blurry reflections on the metal ball.



**Figure 5:** Qualitative results for the CO3D dataset [RSH\*21]. We select a viewpoint from the test set of each scene and show the output of each baseline. Note the cleaner object boundaries of the plant as well as the less blurry reflections on the car.

Method	Source	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic	Ship	Mean
LLFF [MSOC* 19]	[MST* 20]	0.948	0.890	0.896	0.965	0.911	0.890	0.964	0.823	0.911
NV [LSS* 19]	[MST* 20]	0.916	0.873	0.910	0.944	0.880	0.888	0.946	0.784	0.893
SRN [SZW19]	[MST* 20]	0.910	0.766	0.849	0.923	0.809	0.808	0.947	0.757	0.846
NPBG [ASK* 20]	[XXP* 22]	0.939	0.904	0.940	0.964	0.923	0.887	0.959	0.866	0.923
NeRF [MST* 20]	Publication	0.967	0.925	0.964	0.974	0.961	0.949	0.980	0.856	0.947
JaxNeRF [DBS20]	Ours	0.975	0.928	0.968	0.978	0.968	0.952	0.987	0.872	0.954
NSVF [LGL* 20]	[BMT* 21]	0.968	0.931	0.973	0.980	0.960	0.973	0.987	0.854	0.953
Mip-NeRF [BMT* 21]	Publication	0.981	0.932	0.980	0.982	0.978	0.959	0.991	0.882	0.961
PlenOctrees [YLT* 21]	Publication	0.981	0.933	0.970	0.982	0.971	0.955	0.987	0.884	0.958
SNeRG [HSM* 21]	Publication	0.975	0.929	0.967	0.971	0.973	0.938	0.982	0.865	0.950
Plenoxels [FKYT* 22]	Ours	0.977	0.933	0.976	0.980	0.975	0.949	0.985	0.889	0.958
DVGO [SSC22]	Publication	0.977	0.930	0.978	0.980	0.976	0.951	0.983	0.879	0.957
Ref-NeRF [VHM* 22]	Publication	0.984	0.937	0.983	0.984	0.981	<b>0.983</b>	<b>0.992</b>	0.880	<b>0.966</b>
Point-NeRF [XXP* 22]	Publication	0.984	0.935	<b>0.987</b>	0.982	0.978	0.948	0.990	0.892	0.962
Instant-NGP [MESK22]	Ours	<b>0.986</b>	0.941	0.983	0.984	<b>0.983</b>	0.951	0.990	<b>0.898</b>	0.965
TensorRF [CXG* 22]	Ours	0.985	0.938	0.983	0.982	<b>0.983</b>	0.953	0.988	<b>0.898</b>	0.964
Zhang et al. [ZBRH22]	Publication	-	-	-	-	-	-	-	-	0.945
MobileNeRF [CFHT23]	Publication	0.978	0.927	0.965	0.973	0.975	0.913	0.979	0.867	0.947
SNP [ZD23]	Publication	0.962	0.913	0.933	0.977	0.949	0.939	0.972	0.866	0.939
Ours	Ours	<b>0.986</b>	<b>0.947</b>	0.982	<b>0.985</b>	0.975	0.960	0.991	0.891	0.965

**Table 6:** Per-scene test set SSIMs on the Realistic Synthetic 360° dataset [MST\* 20].

Method	Source	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic	Ship	Mean
LLFF [MSOC* 19]	[MST* 20]	0.064	0.126	0.130	0.061	0.110	0.117	0.084	0.218	0.114
NV [LSS* 19]	[MST* 20]	0.109	0.214	0.162	0.109	0.175	0.130	0.107	0.276	0.160
SRN [SZW19]	[MST* 20]	0.106	0.267	0.149	0.100	0.200	0.174	0.063	0.299	0.170
NPBG [ASK* 20]	[XXP* 22]	0.085	0.112	0.078	0.075	0.119	0.134	0.060	0.210	0.109
NeRF [MST* 20]	Publication	0.046	0.091	0.044	0.121	0.050	0.063	0.028	0.206	0.081
JaxNeRF [DBS20]	Ours	0.035	0.085	0.038	0.079	0.040	0.060	0.019	0.185	0.068
NSVF [LGL* 20]	[BMT* 21]	0.043	0.069	0.017	0.025	0.029	<b>0.021</b>	0.010	0.162	0.047
Mip-NeRF [BMT* 21]	Publication	0.021	0.065	0.020	0.027	0.021	0.040	0.009	0.138	0.043
PlenOctrees [YLT* 21]	Publication	0.022	0.076	0.038	0.032	0.034	0.059	0.017	0.144	0.053
SNeRG [HSM* 21]	Publication	0.025	0.061	0.028	0.043	0.022	0.052	0.016	0.156	0.050
Plenoxels [FKYT* 22]	Ours	0.031	0.067	0.027	0.038	0.029	0.057	0.015	0.134	0.050
DVGO [SSC22]	Publication	0.027	0.077	0.024	0.034	0.028	0.058	0.017	0.161	0.053
Ref-NeRF [VHM* 22]	Publication	0.017	0.059	0.019	<b>0.022</b>	<b>0.018</b>	0.022	<b>0.007</b>	0.139	<b>0.038</b>
Point-NeRF [XXP* 22]	Publication	0.023	0.078	0.022	0.037	0.024	0.072	0.014	<b>0.124</b>	0.049
Instant-NGP [MESK22]	Ours	0.020	0.072	0.023	0.032	<b>0.018</b>	0.061	0.013	0.131	0.046
TensorRF [CXG* 22]	Ours	0.026	0.077	0.024	0.035	0.021	0.060	0.022	0.140	0.051
Zhang et al. [ZBRH22]	Publication	-	-	-	-	-	-	-	-	0.078
MobileNeRF [CFHT23]	Publication	0.025	0.077	0.048	0.050	0.025	0.092	0.032	0.145	0.062
SNP [ZD23]	Publication	0.049	0.081	0.050	0.036	0.057	0.072	0.025	0.167	0.067
Ours	Ours	<b>0.016</b>	<b>0.050</b>	<b>0.016</b>	<b>0.022</b>	0.026	0.041	<b>0.007</b>	0.125	<b>0.038</b>

**Table 7:** Per-scene test set LPIPS scores on the Realistic Synthetic 360° dataset [MST\* 20].