Self-calibrating Neural Radiance Fields: Supplementary Materials

1. Implementation Details

1.1. Training NeRF Networks

We use batch size of 1024 rays for NeRF [3], 512 rays for NeRF++ [4]. We initially set the learning rate of NeRF to 0.0005. The learning rate decays exponentially to one-tenth for every 400000 steps. For NeRF++, we initially set the learning rate to 0.0005. The learning rate decays exponentially to one-tenth for every 7500000 steps. As explained in the main paper, we adopt curriculum learning for better stability of training. We extend our learnable camera parameters for every 200K iterations for NeRF experiments. For NeRF++, we have extended our learnable parameters in 500K iterations, 800K iterations, and 1.1M iterations. For NeRF, we use 64 samples for the coarse network and 128 samples for the fine network. In NeRF++ experiments, we have scaled extrinsic noises to 0.01. Especially for tanks and temples [1] dataset, 64 points along a ray are sampled and fed to a coarse network. 128 points along the same ray are sampled and fed to a fine network. For FishEyeNeRF experiments, we have sampled 128 points for the coarse network and 256 for the fine network along the ray. Besides, 1024 rays are used for the FishEyeNeRF experiments for each iteration.

1.2. Projected Ray Distance Evaluation

The projected ray distance measures the deviation of a correspondence pair. However, as it only finds the shortest distance between rays in 3D space, a small change in the direction sometimes leads to a large change in the ray distance. Thus, we threshold ray distance with η and remove pairs above this threshold. We set the η to 5.0 for all the experiments.

2. Calibration with COLMAP initialization

We extend Table 2 in the main paper by conducting experiments in other scenes of the LLFF dataset [2]. Table 1 reports the rendering qualities and projected ray distance of NeRF and our model. Our model shows a consistent improvement from NeRF when learnable camera parameters are initialized by COLMAP camera information.

Table 1: Comparison between NeRF and NeRF + ours when the camera information is initialized with COLMAP. We report PSNR, SSIM, LPIPS, and PRD for training dataset.

scene		PSNR(↑) / SSIM(↑) / LPIPS(↓) / PRD(↓)
Fern	NeRF	30.7 / 0.912 / 0.127 / 2.369
	ours	**31.1 / 0.917 / 0.117 / 0.993**
Flower	NeRF	32.2 / 0.937 / 0.067 / 2.440
	ours	**33.3 / 0.946 / 0.058 / 0.895**
Fortress	NeRF	35.3 / 0.947 / 0.056 / 2.475
	ours	**36.6 / 0.96 / 0.049 / 0.724**
Horns	NeRF	31.6 / 0.931 / 0.116 / 2.499
	ours	**32.2 / 0.932 / 0.114 / 0.907**
Leaves	NeRF	25.3 / 0.874 / 0.149 / 2.709
	ours	**25.9 / 0.886 / 0.136 / 0.854**
Orchids	NeRF	25.6 / 0.864 / 0.151 / 2.417
	ours	**26.4 / 0.881 / 0.134 / 1.173**
Room	NeRF	**39.7 / 0.981 / 0.063** / 2.531
	ours	**39.7 / 0.981 / 0.063 / 0.805**
Trex	NeRF	31.4 / 0.955 / 0.099 / 2.368
	ours	**32.0 / 0.959 / 0.095 / 0.953**

3. Calibration without COLMAP initialization

We also extend Table 1 in the main paper by conducting the experiments in other scenes of the LLFF dataset [2]. Table 2 reports the rendering qualities and projected ray distance metric. NeRF fails to render the scenes reliably; however, our model does. Qualitative results are shown in Figure 4.

4. Ablation Studies

We extend ablation study in our main paper by conducting experiments in the other scenes of LLFF [2] dataset. Table 3 reports the quantitative results of the ablation study.

5. Qualitative Results

We report some qualitative results of experiments in the main paper. Figure 1 compares NeRF++ and our model in a tanks and temples [1] dataset. Figure 2 compares NeRF++ and our model in two fish-eye scenes. Figure 4 visualizes rendered images of our model when no calibrated camera information is provided. Lastly, Figure 5 visualizes the captured non-linear distortion in for all the scenes in LLFF [2] dataset.

Table 2: Comparison between NeRF and NeRF + ours when the camera information is initialized with COLMAP. We report PSNR, SSIM, LPIPS, and PRD for training dataset.

scene		PSNR(↑) / SSIM(↑) / LPIPS(↓) / PRD(↓)
fern	NeRF	16.9 / 0.435 / 0.544 / nan
	ours	**31.2 / 0.918 / 0.117 / 1.020**
flower	NeRF	13.8 / 0.302 / 0.716 / nan
	ours	**33.2 / 0.945 / 0.060 / 0.911**
fortress	NeRF	16.3 / 0.524 / 0.445 / nan
	ours	**35.7 / 0.945 / 0.069 / 0.833**
horns	NeRF	14.8 / 0.390 / 0.634 / nan
	ours	**22.6 / 0.0613 / 0.494 / 1.578**
leaves	NeRF	13.0 / 0.170 / 0.687 / nan
	ours	**25.8 / 0.878 / 0.146 / 0.885**
orchids	NeRF	13.1 / 0.170 / 0.674 / nan
	ours	**24.8 / 0.830 / 0.204 / 1.269**
room	NeRF	18.1 / 0.660 / 0.486 / nan
	ours	**37.5 / 0.967 / 0.103 / 0.852**
trex	NeRF	15.7 / 0.409 / 0.575 / nan
	ours	**31.8 / 0.954 / 0.104 / 1.002**

Figure 1: Error map of rendered images by NeRF++ [4] and our model in tanks and temples [1] dataset. The above and the below maps are generated error maps by NeRF++ and our model, respectively. For each subfigure, PSNR is shown on the upper left.

(a) NeRF++ (b) ours (c) Target Image

Figure 2: Comparison between NeRF++ [4] and our model in FishEyeNeRF dataset. Our model shows much clearer rendering compared to NeRF++.

Table 3: Ablation studies about components of our model. ”IE”, ”OD”, and ”PRD” denote learnable intrinsic and extrinsic parameters, learnable non-linear distortion, and projected ray distance loss, respectively.

scene		PSNR(↑) / SSIM(↑) / LPIPS(↓) / PRD(↓)
Fern	NeRF	25.3 / 0.809 / 0.178 / 1.069
	+ IE	30.2 / 0.907 / 0.127 / **0.988**
	+ IE + OD	30.9 / 0.915 / 0.118 / 0.991
	+ IE + OD + PRD	**31.1 / 0.917 / 0.117** / 0.993
Flower	NeRF	28.1 / 0.879 / 0.104 / 0.989
	+ IE	32.2 / 0.937 / 0.068 / **0.893**
	+ IE + OD	33.1 / 0.944 / 0.060 / **0.893**
	+ IE + OD + PRD	**33.3 / 0.946 / 0.058** / 0.895
Fortress	NeRF	30.5 / 0.866 / 0.096 / 0.856
	+ IE	35.3 / 0.948 / 0.058 / 0.729
	+ IE + OD	36.4 / 0.957 / 0.051 / **0.724**
	+ IE + OD + PRD	**36.6 / 0.960 / 0.049 / 0.724**
Horns	NeRF	27.0 / 0.857 / 0.171 / 0.987
	+ IE	31.2 / 0.921 / 0.128 / **0.907**
	+ IE + OD	32.0 / 0.930 / 0.117 / **0.907**
	+ IE + OD + PRD	**32.2 / 0.932 / 0.114 / 0.907**
Leaves	NeRF	22.0 / 0.787 / 0.193 / 0.951
	+ IE	25.2 / 0.872 / 0.147 / 0.853
	+ IE + OD	25.8 / 0.883 / 0.138 / **0.852**
	+ IE + OD + PRD	**25.9 / 0.886 / 0.136** / 0.854
Orchids	NeRF	22.8 / 0.783 / 0.199 / 1.240
	+ IE	25.7 / 0.866 / 0.147 / **1.170**
	+ IE + OD	26.3 / 0.878 / 0.137 / 1.172
	+ IE + OD + PRD	**26.4 / 0.881 / 0.134** / 1.173
Room	NeRF	31.5 / 0.950 / 0.096 / 0.883
	+ IE	38.3 / 0.978 / 0.070 / 0.806
	+ IE + OD	39.4 / 0.980 / 0.065 / **0.805**
	+ IE + OD + PRD	**39.7 / 0.981 / 0.063 / 0.805**
Trex	NeRF	26.5 / 0.893 / 0.138 / 1.016
	+ IE	31.0 / 0.952 / 0.104 / **0.951**
	+ IE + OD	31.8 / 0.958 / 0.097 / 0.952
	+ IE + OD + PRD	**32.0 / 0.959 / 0.095** / 0.953

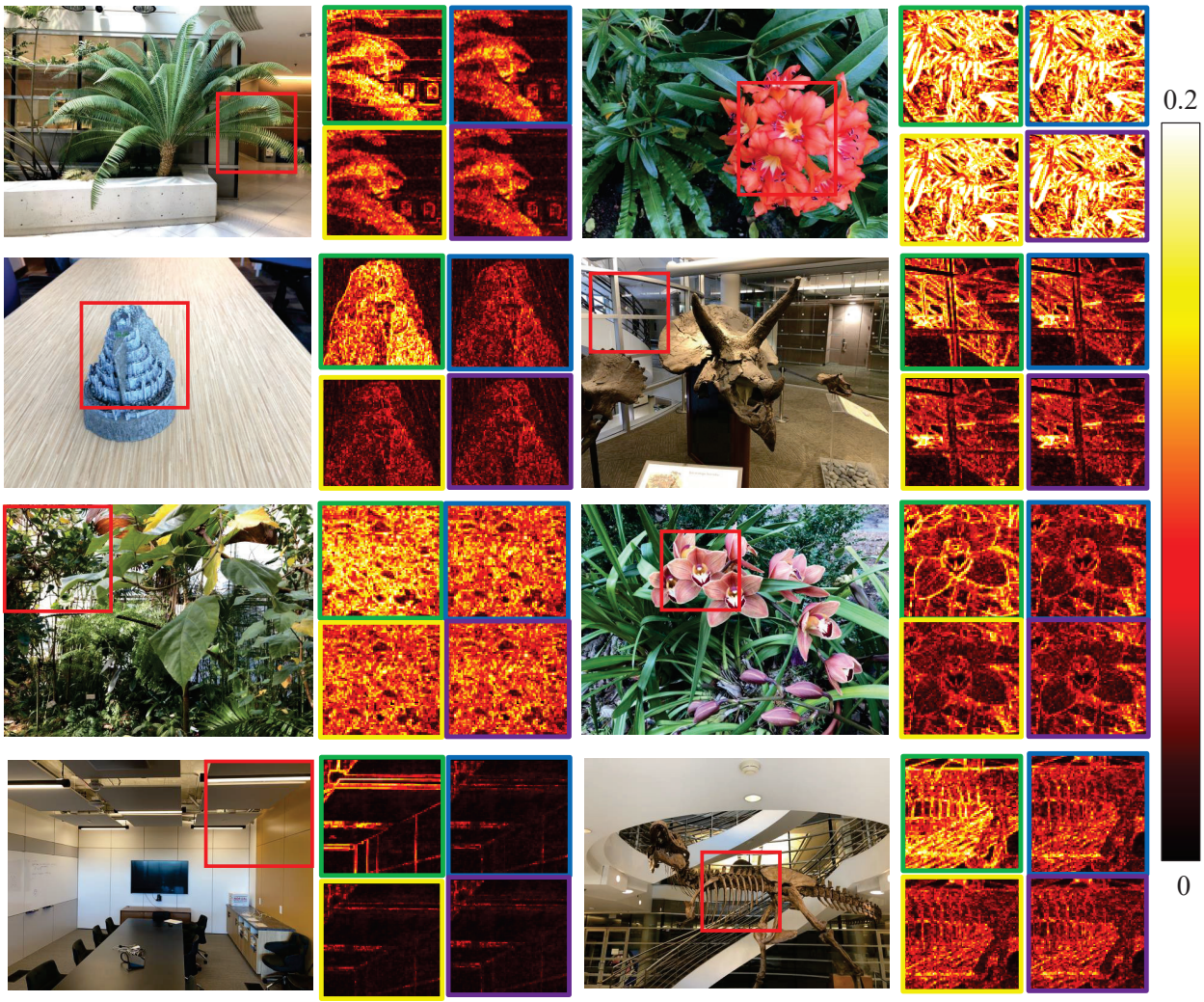

Figure 3: Visualization of rendered images for each phase shown in Table 3. The green, blue, yellow, and purple box are the error map of NeRF, NeRF + IE, NeRF + IE + OD, and NeRF + IE + OD + PRD, respectively.

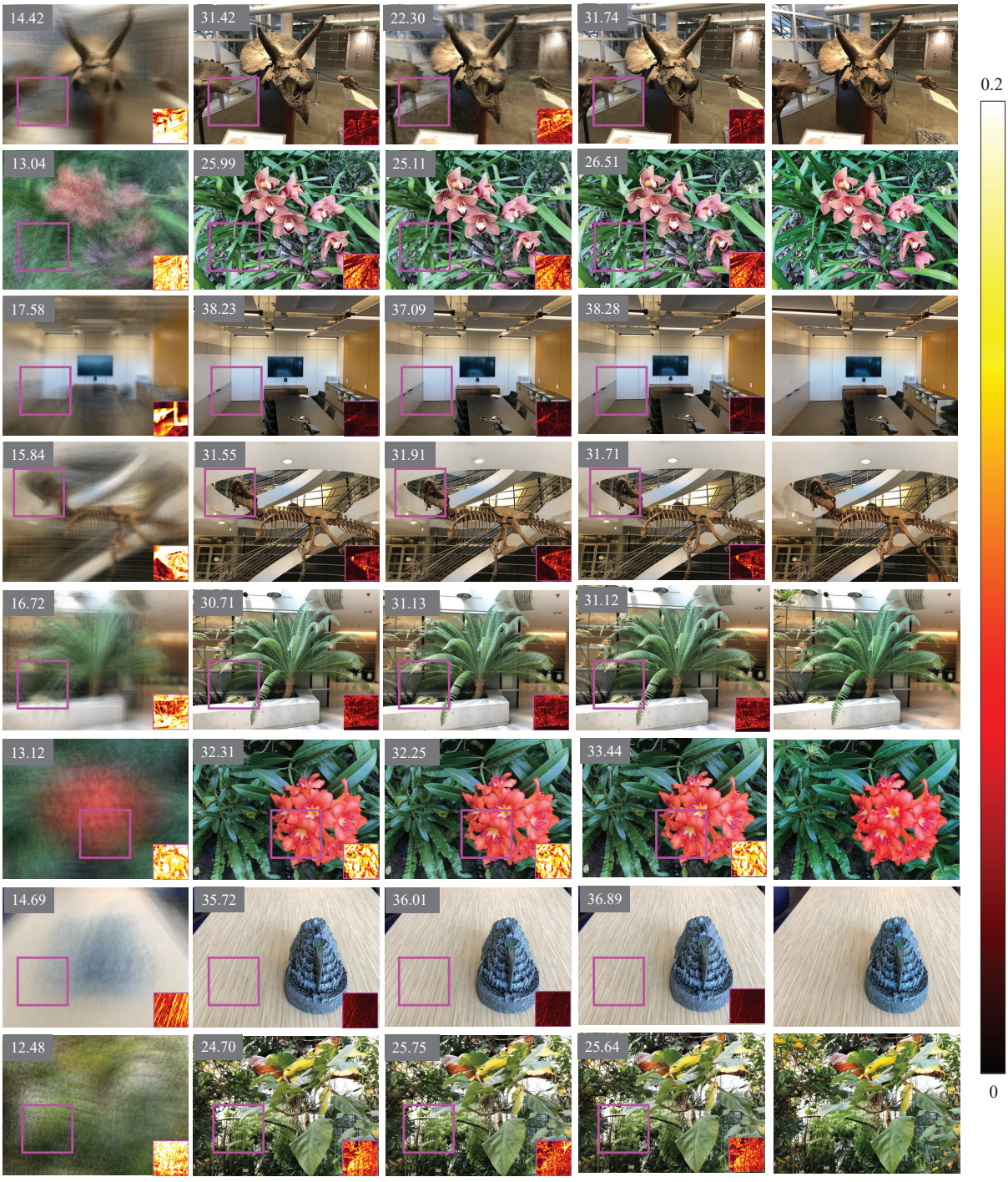

Figure 4: The red block visualizes rendered images without calibrated camera information. Although our model is trained without camera information, our model shows comparable performance with NeRF, trained with COLMAP camera information. The blue block visualizes the rendering of NeRF using COLMAP camera information. For each subfigure, PSNR is shown on the upper left.

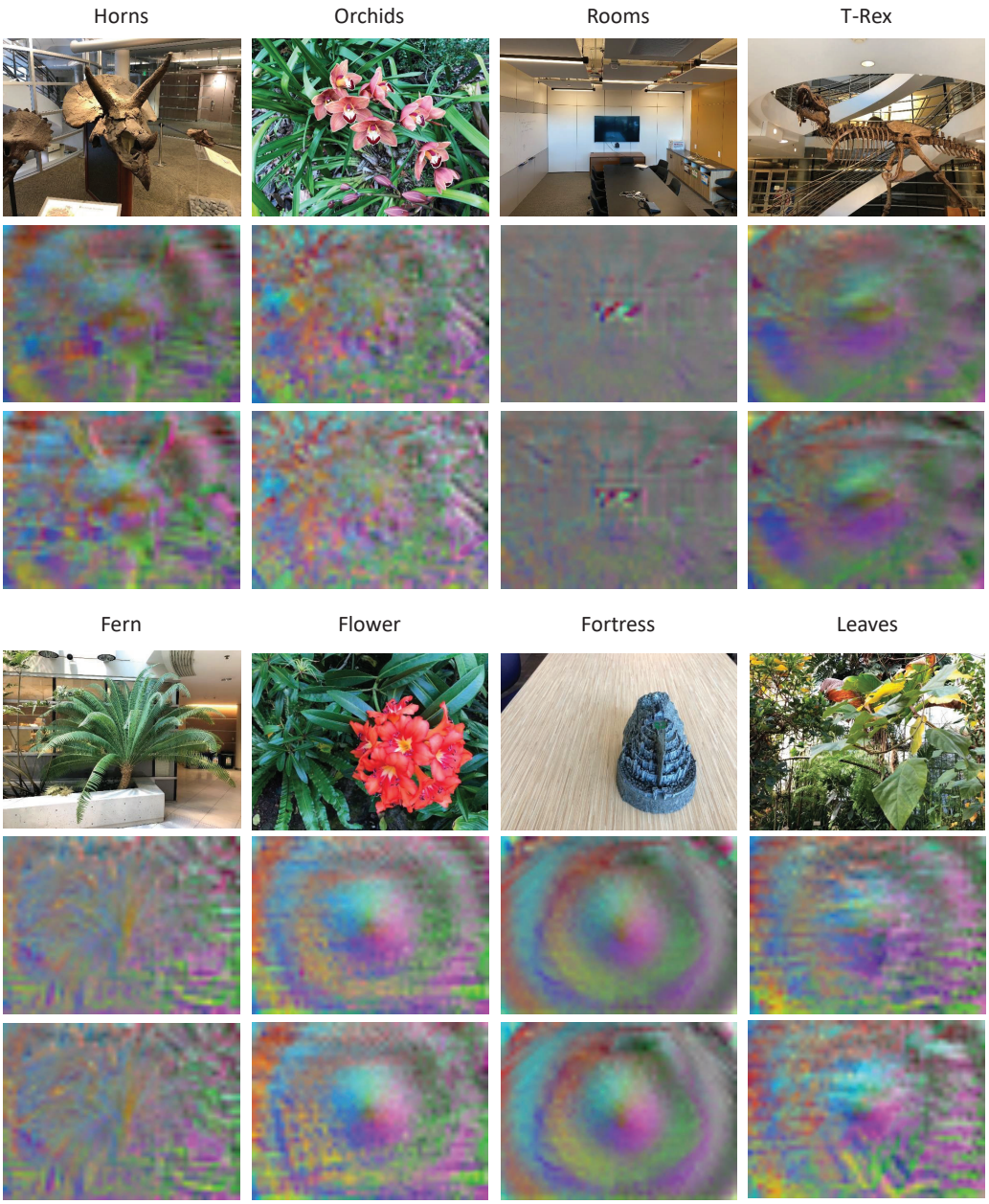

Figure 5: Visualization of captured non-linear distortions of our trained camera model. We have observed circular patterns in all the scenes. The second row and the forth row are captured ray offset distortions, and the third row and the fifth row are captured ray direction distortions.

References

[1] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

[2] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019.

[3] 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. *arXiv preprint arXiv:2003.08934*, 2020.

[4] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.