Supplementary Material for Balanced Spherical Grid for Egocentric View Synthesis

A. Vector-Matrix Decomposition of Balanced Spherical Feature Grids

In this section, we describe the vector-matrix factorization of our balanced spherical feature grids. As mentioned in Sec. 3.1. of the main manuscript, we model the radiance fields as 3D/4D tensors, which map a 3D position vector to volume density σ and appearance feature vector. Inspired by [3], we decompose the 3D/4D tensor into low-rank tensor components.

VM Decomposition VM decomposition or vector-matrix decomposition, proposed by [3], decomposes a 3D tensor $\mathbf{T} \in \mathbb{R}^{I\times J\times K}$ into multiple vectors and matrices:

$$\mathbf{T} = \sum_{n=1}^{N_1} \mathbf{v}_n^1 \otimes \mathbf{M}_n^{2,3} + \sum_{n=1}^{N_2} \mathbf{v}_n^2 \otimes \mathbf{M}_n^{3,1} + \sum_{n=1}^{N_3} \mathbf{v}_n^3 \otimes \mathbf{M}_n^{1,2}, \quad (1)$$

where $\otimes$ denotes outer product, $\mathbf{v}_n^1 \in \mathbb{R}^I$, $\mathbf{v}_n^2 \in \mathbb{R}^J$, $\mathbf{v}_n^3 \in \mathbb{R}^K$, and $\mathbf{M}_n^{2,3} \in \mathbb{R}^{J\times K}$, $\mathbf{M}_n^{3,1} \in \mathbb{R}^{K\times I}$, $\mathbf{M}_n^{1,2} \in \mathbb{R}^{I\times J}$ are vector and matrix factors for three modes of nth component respectively. In general, N_1, N_2, N_3 have different values, but we use the same number of components for each mode for simplicity. i.e. $N_1 = N_2 = N_3 = N$. Then, Eq. (1) can be expressed as

$$\mathbf{T} = \sum_{n=1}^{N} \sum_{m\in\{1,2,3\}} \mathcal{A}_n^m, \quad (2)$$

where $\mathcal{A}_n^1 = \mathbf{v}_n^1 \otimes \mathbf{M}_n^{2,3}$, $\mathcal{A}_n^2 = \mathbf{v}_n^2 \otimes \mathbf{M}_n^{3,1}$, $\mathcal{A}_n^3 = \mathbf{v}_n^3 \otimes \mathbf{M}_n^{1,2}$.

VM Decomposition of Balanced Spherical Feature Grids Our density feature grid $\mathcal{G}_\sigma$ is a 3D tensor of $\mathbb{R}^{2N_r^y \times N_\theta^y \times N_\phi^y}$. The overset grid $\mathcal{G}_\sigma$ is a union of two tensors $\mathcal{G}_\sigma^{\text{Yin}}$ and $\mathcal{G}_\sigma^{\text{Yang}} \in \mathbb{R}^{N_r^y \times N_\theta^y \times N_\phi^y}$. Each 3D tensor is further decomposed into vector and matrix factors using Eq. (2):

$$\begin{aligned}\mathcal{G}_\sigma^y &= \sum_{n=1}^{N_\sigma} \mathbf{v}_{\sigma,n}^{y,R} \otimes \mathbf{M}_{\sigma,n}^{y,\Theta\Phi} + \mathbf{v}_{\sigma,n}^{y,\Theta} \otimes \mathbf{M}_{\sigma,n}^{y,\Phi R} + \mathbf{v}_{\sigma,n}^{y,\Phi} \otimes \mathbf{M}_{\sigma,n}^{y,R\Theta} \\ &= \sum_{n=1}^{N_\sigma} \sum_{m\in R\Theta\Phi} \mathcal{A}_{\sigma,n}^{y,m}, \quad y \in \{\text{Yin, Yang}\}. \end{aligned} \quad (3)$$

In contrast, our appearance gird $\mathcal{G}_a \in \mathbb{R}^{2N_r^y \times N_\theta^y \times N_\phi^y \times C}$ is a 4D tensor which has additional C-dimensional neural appearance features. Since the mode of appearance feature does not need high dimension as spatial modes (R, Θ, Φ), we assign only vector components $\mathbf{b}$ for this mode, instead of matrix components from [3]. Specifically, $\mathcal{G}_a$ also consists of two tensors $\mathcal{G}_a^{\text{Yin}}$ and $\mathcal{G}_a^{\text{Yang}} \in \mathbb{R}^{N_r^y \times N_\theta^y \times N_\phi^y \times C}$ and each are factorized as following:

$$\begin{aligned}\mathcal{G}_a^y =& \sum_{n=1}^{N_a} \mathbf{v}_{a,n}^{y,R} \otimes \mathbf{M}_{a,n}^{y,\Theta\Phi} \otimes \mathbf{b}_{3n-2}^y + \mathbf{v}_{a,n}^{y,\Theta} \otimes \mathbf{M}_{a,n}^{y,\Phi R} \otimes \mathbf{b}_{3n-1}^y \\ &+ \mathbf{v}_{a,n}^{y,\Phi} \otimes \mathbf{M}_{a,n}^{y,R\Theta} \otimes \mathbf{b}_{3n}^y \\ =& \sum_{n=1}^{N_a} \mathcal{A}_{a,n}^{y,R} \otimes \mathbf{b}_{3n-2}^y + \mathcal{A}_{a,n}^{y,\Theta} \otimes \mathbf{b}_{3n-1}^y + \mathcal{A}_{a,n}^{y,\Phi} \otimes \mathbf{b}_{3n}^y. \end{aligned} \quad (4)$$

$\mathbf{B} \in \mathbb{R}^{C\times 6N_a}$ in Fig.4 of the main manuscript is a matrix obtained by stacking all $\mathbf{b}^y$s columnwise. By using $\mathbf{B}$ matrix, we can calculate Eq. (4) with simple matrix multiplication.

Querying Values from Grids In the volume rendering pipeline, the volume density σ and color c are queried from our feature grids along the camera rays:

$$\sigma(\mathbf{x}) = \mathcal{T}(\mathcal{G}_\sigma, \mathbf{x}), \quad c(\mathbf{x}, \mathbf{d}) = f_{\text{MLP}}(\mathcal{T}(\mathcal{G}_a, \mathbf{x}), \mathbf{d}), \quad (5)$$

where $\mathbf{x}$, $\mathbf{d}$ are querying position and viewing direction respectively, and $\mathcal{T}$ is a trilinear interpolation operator, as denoted in Eq. (5) of the main manuscript. Furthermore, we can reduce computational burden by replacing trilinear interpolation with linear/bilinear interpolation of vector/matrix factors.

B. Implementation Details

EgoNeRF is implemented with PyTorch [9] without using any customized CUDA kernels. We will release the code and the dataset publicly upon publication.

B.1. Hyperparameter Setup

In this section, we report the hyperparameter setup used in experiments for EgoNeRF. We use Adam optimizer [5]

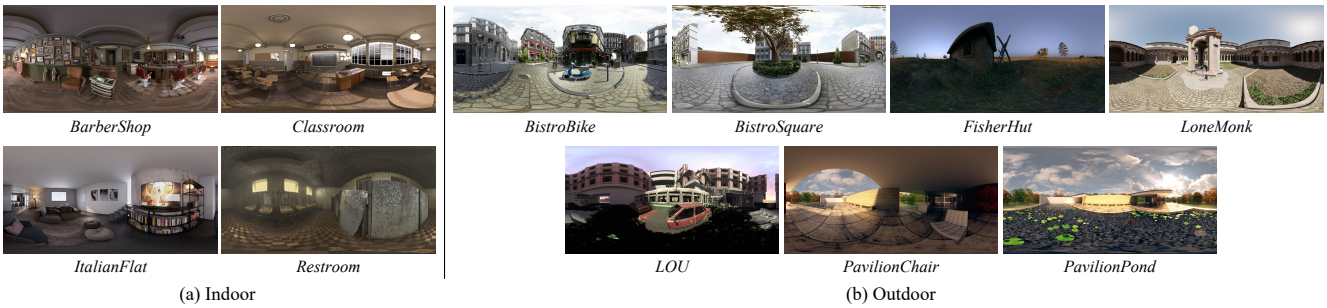


Figure 1. Samples from our synthetic *OmniBlender* dataset.

Figure 2. Samples from our real-world *Ricoh360* dataset.

with a learning rate of 0.02 following [3], and use default values of other hyperparameters of Adam optimizer ($\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-7}$). For all scenes, we use 300^3 voxels for both $\mathcal{G}_\sigma$ and $\mathcal{G}_a$ with $N_r^y : N_\theta^y : N_\phi^y = 1 : \frac{2\sqrt{3}}{3} : 2\sqrt{3}$. The dimension of appearance feature C is 27 and we use two-layer MLP of 128 hidden units for decoding network f_{MLP}. The number of decomposed components $N_\sigma = 16$ and $N_a = 48$. We use the $r_0 = 0.03, 0.05$, and $R_{\text{max}} = 15, 300$ for the indoor and outdoor scenes, respectively. And the size of convolution kernel K for obtaining a coarse grid is 2.

B.2. Dataset Information

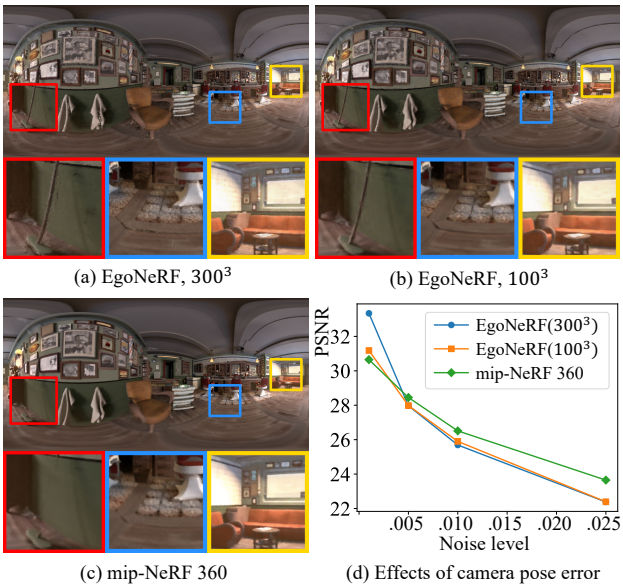
OmniBlender *OmniBlender* is our newly introduced synthetic dataset. OmniBlender contains outward-looking images of 11 challenging large-scale indoor/outdoor environments with various objects and materials. We render equirectangular images along short helix camera trajectory with Blender's Cycles path tracer [4]. All the images have 2000×1000 resolution and we use 25 images for the training set and test set respectively. Sample images from our dataset are presented in Fig. 1. We slightly modify the publicly available 3D models from various sources below:

```
# Indoor
- BarberShop by Blender (CC-BY)
https://www.blender.org/download/demo-files/
- Classroom by Christophe Seux (CC-BY)
https://www.blender.org/download/demo-files/
- ItalianFlat by Flavio Della Tommasa (CC-BY)
https://www.blender.org/download/demo-files/
- Restroom by oldtimer (CC-BY)
https://blendswap.com/blend/14216
# Outdoor
- Bistro by Amazon Lumberyard (CC-BY)
http://developer.nvidia.com/orca/amazon-lumberyard-
    bistro
- FisherHut by DHCG (CC-BY)
https://www.blendswap.com/blend/30099
- LoneMonk by Carlo Bergonzin (CC-BY)
https://www.blender.org/download/demo-files/
- LOU by Andreas Stromberg
https://stromberg.gumroad.com/l/dGeBoS
- Pavilion by eMirage (CC-BY)
https://www.blender.org/download/demo-files/
```

Ricoh360 *Ricoh360* is our newly introduced real-world dataset. The dataset contains short omnidirectional videos captured by rotating a commercial Ricoh Theta V camera attached to a selfie stick. We collect 11 large-scale scenes from various indoor/outdoor environments. After capturing the videos, we estimate camera parameters corresponding to each images by structure from motion library [8]. We use 50 images for training set and test set respectively. Sample images from Ricoh360 dataset are demonstrated in Fig. 2.

C. Analysis on Camera Parameter Error

In this section, we analyze the effects of errors in camera parameters. While EgoNeRF is able to reconstruct precise 3D scenes and synthesize high-quality novel views under perfect camera parameters in synthetic *OmniBlender* dataset, our approach shows a degraded performance when the camera poses have errors in real-world scenes like prior



(a) EgoNeRF, 300^3 (b) EgoNeRF, 100^3 (c) mip-NeRF 360 (d) Effects of camera pose error

Figure 3. Qualitative results under the Gaussian perturbation $\epsilon \sim (0, 0.005)$ in different models (a) EgoNeRF, (b) EgoNeRF with coarser grid (100^3), and (c) MLP-based method mip-NeRF 360 [2]. (d) Quantitative results of injecting different levels of Gaussian noise into camera poses.

works. To further study the effects of the camera pose error, we train EgoNeRF with different grid resolutions (300^3 which is identical to our original setup, and 100^3) and MLP-based approach mip-NeRF 360 [2] under various levels of camera pose errors. We perturb the camera pose by adding Gaussian noises $\epsilon \sim (0, \sigma^2)$ with different levels of variance in the *BarberShop* scene in *OmniBlender*.

As shown in Fig. 3 (d), injecting a higher level of noise reduces the performance across all the methods consistently. EgoNeRF with the default parameter (resolution of 300^3) outperforms other baselines amidst a negligible amount of noise (variance of 0.001), which is coherent with the main results. When the level of noise increases, however, the performance of our model with fine resolution degrades rapidly and reaches a similar level of EgoNeRF with coarse resolution. The MLP-based approach [2] shows better performance in the presence of high level of noise (greater than 0.01).

In Fig. 3 (a) to (c), we visualize the qualitative results in the noise level 0.005 of which PSNR values from different methods are comparable. As shown in the red box of Fig. 3 (a), we observe a noisy artifact nearby the fine structure of the close objects in EgoNeRF. On the other hand, EgoNeRF with coarse resolution does not show such a phenomenon in the red box of Fig. 3 (b). In contrast, EgoNeRF with both fine and coarse resolution does not make the noisy artifact in the far-away regions (yellow box). We hypothesize that if the camera pose noise is non-negligible with relative to the

Method	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM
NeRF [7]	31.01	0.081	0.947	24.85	0.426	0.659
mip-NeRF [1]	32.63	0.047	0.958	25.12	0.414	0.672
mip-NeRF 360 [2]	33.25	0.039	0.962	29.23	0.207	0.844
TensoRF [3]	33.14	0.027	0.963	22.75	0.619	0.558
DVGO [10]	32.80	0.027	0.961	20.67	0.490	0.575
EgoNeRF	31.51	0.037	0.952	25.83	0.320	0.701
	(a) Synthetic-NeRF			(b) mip-NeRF 360		

Table 1. Quantitative results in inward-facing datasets (a) Synthetic-NeRF [7] and (b) mip-NeRF 360 [2].

r_0 (m)	0.01	0.03	0.05*	0.1	0.15	0.2	0.3	0.5
PSNR	32.95	33.70	34.07	34.50	34.39	33.93	33.34	31.80
R_{max} (m)	25	50	100	200	300*	500	1000	
PSNR	28.43	33.96	34.31	34.23	34.07	33.93	33.69	

Table 2. Quantitative results in *BistroBike* (max depth =220) for different hyperparameters. We mark * for the default values.

r (m)	0	0.1	0.2	0.5	1*	1.5	2	3	5
PSNR	35.38	35.47	35.45	35.26	34.74	33.15	31.17	27.36	22.02

Table 3. Out of distribution test. r is the distance between the center of training camera trajectory and position of test view.

grid size, the wrong camera parameters cause the camera ray for fine objects to hit the wrong neighborhood grids, which leads to multiple erroneous reconstructions of fine structures. Since our distance-adaptive balanced spherical feature grid has a small grid size near the center and the grid has a larger volume at far regions, the noisy artifact only appears at the close region. As shown in the blue box in Fig. 3, the MLP-based method shows blurry artifacts amidst the noise in the camera pose in contrast to the noisy artifact in grid-based methods. This may be because MLP output naturally interpolates the values observed in the training set.

D. Inward-facing Dataset

The spherical grid of EgoNeRF aligns nicely with outward-facing scenes, not inward-facing images of typical NeRF settings. We optionally report results from widely-used datasets for novel view synthesis in Tab. 1. EgoNeRF shows comparable results in the Synthetic-NeRF dataset [7], which contains 8 synthetic objects. In mip-NeRF 360 dataset [2], which contains inward-facing objects but has unbounded background scenes, EgoNeRF outperforms other baselines except mip-NeRF 360.

E. Robustness Study

In this section, we analyze the effects of various components to the reconstruction quality. In Tab. 2, we study the effects of hyperparameters. EgoNeRF shows robust performance regardless of the choice of r_0 and R_{max} unless R_{max}

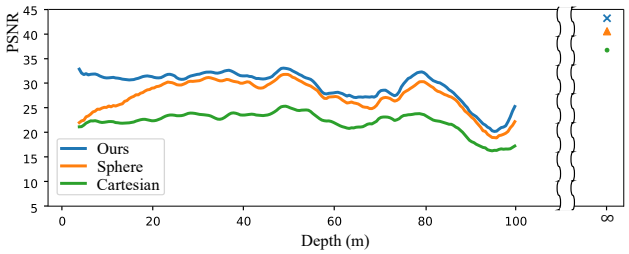


Figure 4. Reconstruction quality according to scene depth.

(a) Rendered result (b) Extracted geometry

Figure 5. We demonstrate the (a) rendered result from novel viewpoint and (b) the extracted geometry from our density grid $\mathcal{G}_\sigma$ in *BistroBike* scene in *OmniBlender* dataset.

is too small compared to the scene size.

Also, we compare the quality of reconstructed images at various depths in Fig. 4. EgoNeRF outperforms regular spherical grid and Cartesian grid, especially in the near region. It supports our claim that Cartesian grid or regular spherical grid has insufficient resolution at nearby regions and is extravagant for far objects.

We further provide the quality of rendering at various distances from the original trajectory ($r = 1$) in Tab. 3. We noticed only minimal quality degradation for $r < 1$ and $1 < r \leq 3$. Only when the viewing position is extremely far ($r \geq 5$), there exists a noticeable performance decrease due to the unseen regions and the unconstrained scene depth.

F. Additional Results

Figure 5 illustrates the geometry obtained from our density feature grid $\mathcal{G}_\sigma$. The explicit mesh is obtained by applying the marching cube algorithm [6]. Our approach is able to reconstruct fine details of large-scale scenes from a very short camera trajectory. We also demonstrate additional rendered results on *OmniBlender* and *Ricoh360* datasets in Fig. 6 and Fig. 7, respectively. The results show that our approach is able to render high-quality images in both large-scale indoor and outdoor scenes from novel viewpoints. We also provide per-scene breakdown for *OmniBlender*, *Ricoh360*, Synthetic-NeRF [7], and mip-NeRF 360 [2] dataset in Tabs. 4 to 17.

References

[1] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5855–5864, October 2021. 3, 9, 10

[2] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5470–5479, June 2022. 3, 4, 5, 6, 7, 8, 9, 10

[3] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 5, 6, 7, 8, 9, 10

[4] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 2

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015. 1

[6] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 4

[7] 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. 3, 4, 5, 6, 7, 8, 9, 10

[8] Pierre Moulon, Pascal Monasse, Romuald Perrot, and Renaud Marlet. OpenMVG: Open multiple view geometry. In *International Workshop on Reproducible Research in Pattern Recognition*, pages 60–74. Springer, 2016. 2

[9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 1

[10] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5459–5469, June 2022. 3, 5, 6, 7, 8, 9, 10

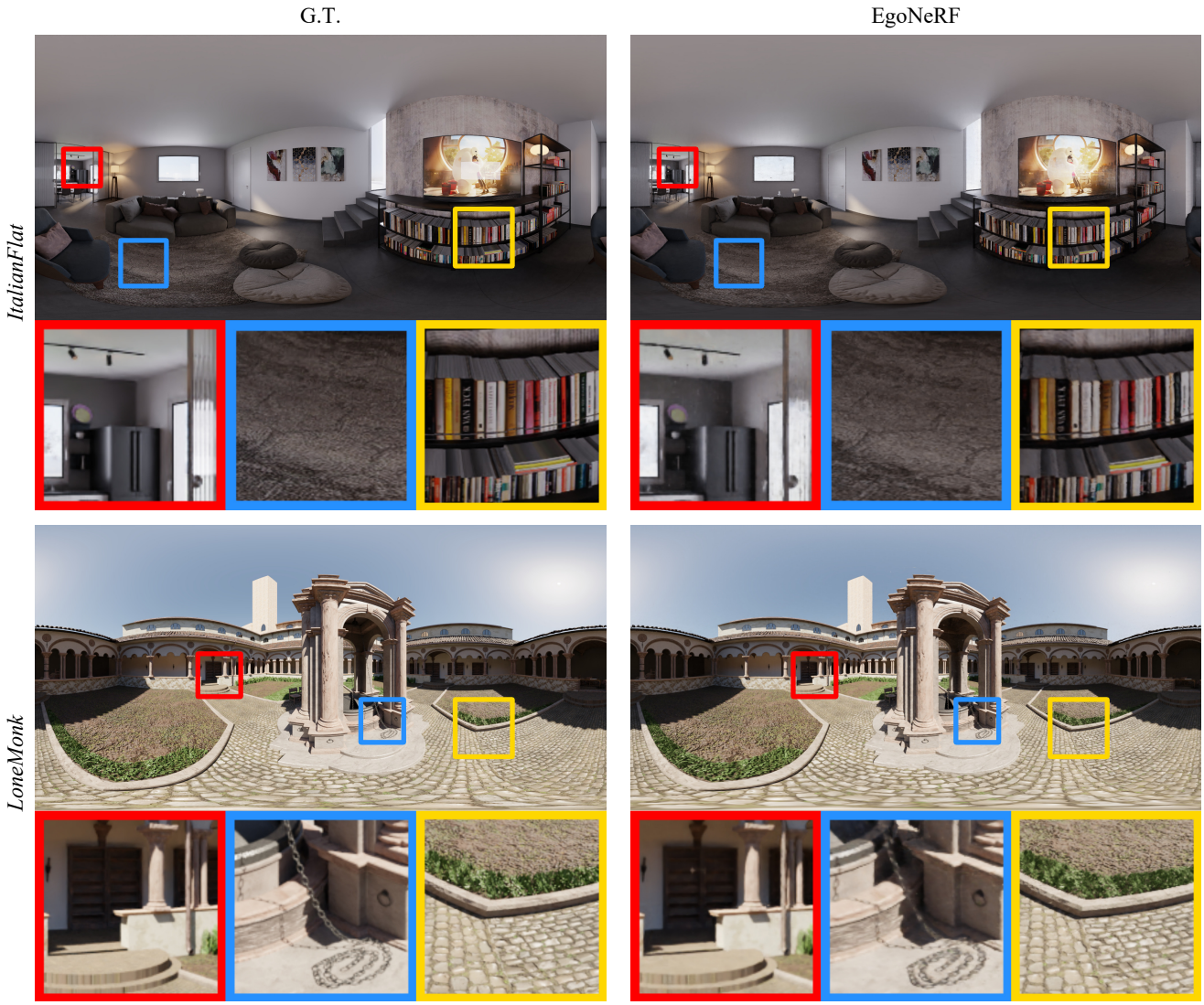


Figure 6. Additional qualitative results in *OmniBlender* dataset.

Step	Method	Indoor				Outdoor						
		BarberShop	*ClassRoom*	*ItalianFlat*	*Restroom*	*BistroBike*	*BistroSquare*	*FisherHut*	*LoneMonk*	*LOU*	*PavilionChair*	*PavilionPond*
5k	NeRF [7]	26.32	24.70	26.31	27.67	20.31	17.80	26.91	21.95	23.57	25.33	20.65
	mip-NeRF 360 [2]	22.65	21.23	24.33	25.83	20.79	18.68	26.99	20.38	21.98	23.72	19.77
	TensoRF [3]	26.03	24.30	27.67	25.62	20.72	18.84	28.52	22.05	26.27	25.26	20.79
	DVGO [10]	22.84	22.31	25.24	26.65	19.24	17.87	27.56	20.30	23.45	24.14	19.34
	EgoNeRF	**31.12**	**26.35**	**29.01**	**29.01**	**29.96**	**23.85**	**29.71**	**28.27**	**30.80**	**28.99**	**23.73**
10k	NeRF [7]	28.01	26.75	27.46	28.41	21.50	18.64	27.90	23.90	25.49	26.05	21.94
	mip-NeRF 360 [2]	28.35	24.50	28.76	28.03	25.27	21.82	29.02	25.18	27.81	26.85	23.03
	TensoRF [3]	27.54	25.22	28.81	26.26	21.74	19.49	28.85	22.96	28.04	26.07	21.47
	DVGO [10]	24.47	23.51	26.42	27.37	20.11	18.31	28.16	21.28	25.08	24.96	19.90
	EgoNeRF	**32.53**	**27.47**	**30.48**	**30.43**	**31.29**	**24.52**	**30.01**	**29.28**	**32.01**	**29.86**	**24.68**
100k	NeRF [7]	33.20	**31.05**	30.17	32.24	25.29	20.85	30.10	28.23	31.43	29.28	24.68
	mip-NeRF 360 [2]	33.30	26.83	32.82	31.54	30.62	24.93	**31.13**	30.22	32.59	30.53	25.39
	TensoRF [3]	30.20	28.91	31.00	26.91	23.55	20.50	29.59	24.64	31.35	27.70	22.43
	DVGO [10]	29.21	26.70	30.14	29.29	22.69	19.87	29.54	23.50	29.75	27.24	21.48
	EgoNeRF	**35.10**	30.37	**33.30**	**33.67**	**34.07**	**25.83**	30.50	**31.53**	**34.03**	**31.67**	**26.29**

Table 4. Per-scene quantitative results in terms of PSNR in *OmniBlender* dataset.

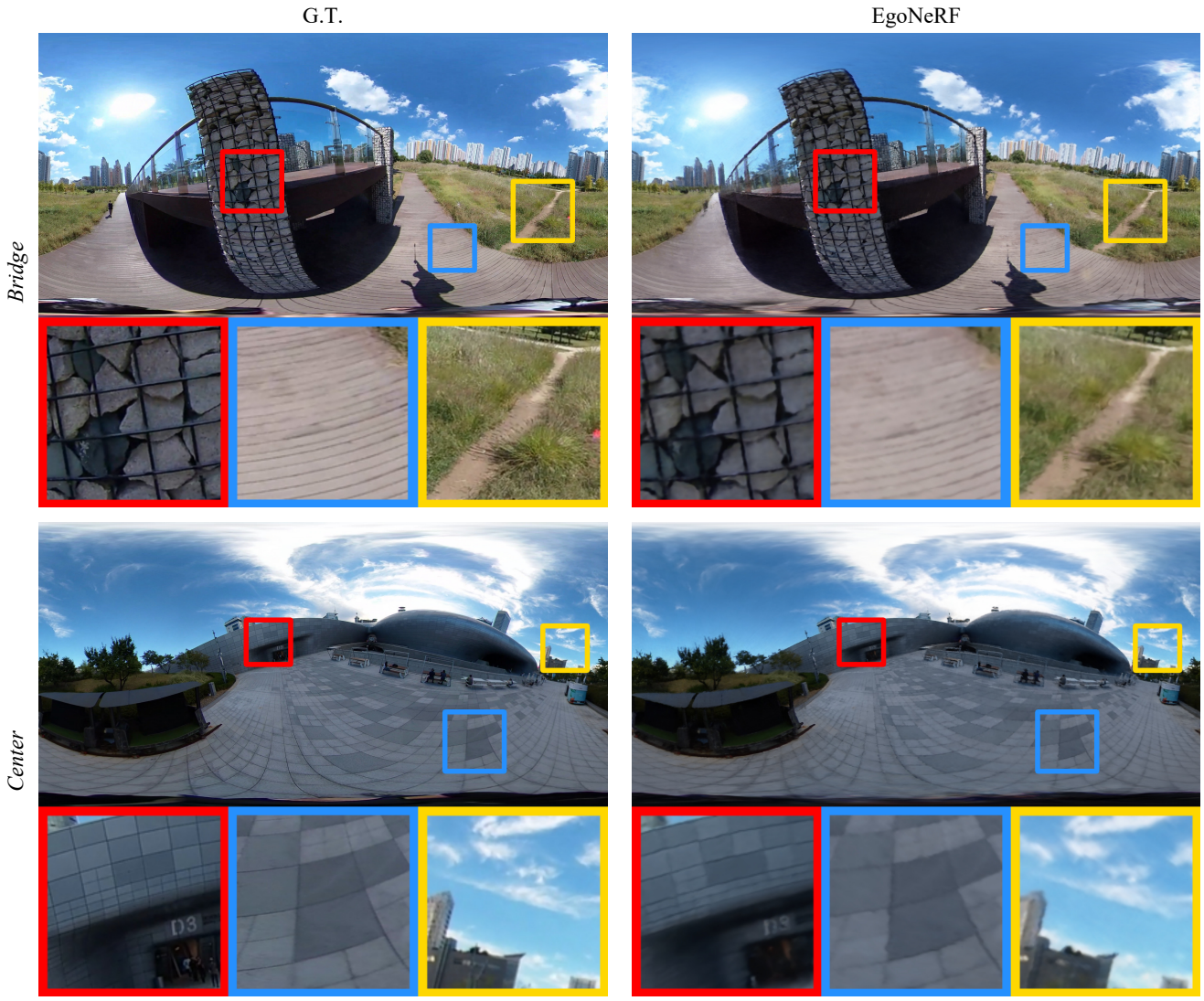


Figure 7. Additional qualitative results in *Ricoh360* dataset.

Step	Method	Indoor				Outdoor						
		BarberShop	*ClassRoom*	*ItalianFlat*	*Restroom*	*BistroBike*	*BistroSquare*	*FisherHut*	*LoneMonk*	*LOU*	*PavilionChair*	*PavilionPond*
5k	NeRF [7]	27.09	25.38	27.30	29.32	21.54	19.69	27.96	23.45	24.49	26.10	22.08
	mip-NeRF 360 [2]	23.40	21.71	25.07	27.46	21.80	20.74	28.10	21.89	22.74	24.78	21.13
	TensoRF [3]	26.88	24.94	28.86	27.03	21.94	21.14	29.99	24.08	27.05	26.47	22.52
	DVGO [10]	23.66	22.94	26.14	28.40	20.40	19.98	28.83	21.98	24.20	25.34	21.29
	EgoNeRF	**32.07**	**26.94**	**30.28**	**30.95**	**30.91**	**26.10**	**31.53**	**30.04**	**31.73**	**29.76**	**25.06**
10k	NeRF [7]	28.80	27.60	28.61	30.20	22.76	20.54	29.12	25.39	26.35	26.87	23.25
	mip-NeRF 360 [2]	29.15	25.01	29.95	29.77	26.10	23.84	30.46	26.74	28.65	27.65	24.16
	TensoRF [3]	28.40	25.86	30.19	27.49	23.07	21.89	30.42	25.23	28.90	27.25	23.21
	DVGO [10]	25.36	24.17	27.45	29.13	21.28	20.57	29.56	23.14	25.86	26.18	21.81
	EgoNeRF	**33.49**	**28.11**	**31.92**	**32.36**	**32.22**	**26.88**	**31.84**	**31.12**	**32.96**	**30.60**	**25.86**
100k	NeRF [7]	34.13	**32.29**	31.86	34.02	26.87	22.98	31.77	29.92	32.41	29.99	25.81
	mip-NeRF 360 [2]	34.20	27.37	34.59	33.48	31.39	27.24	**32.93**	31.86	33.49	31.13	26.38
	TensoRF [3]	31.16	30.00	32.90	28.21	25.04	23.09	31.47	27.01	32.35	29.02	24.31
	DVGO [10]	30.25	27.49	31.85	31.32	24.23	22.51	31.32	26.02	30.73	28.63	23.64
	EgoNeRF	**36.13**	31.30	**35.07**	**35.60**	**35.03**	**28.32**	32.49	**33.42**	**35.10**	**32.48**	**27.46**

Table 5. Per-scene quantitative results in terms of PSNR^{WS} in *OmniBlender* dataset.

Step	Method	Indoor				Outdoor						
		BarberShop	*ClassRoom*	*ItalianFlat*	*Restroom*	*BistroBike*	*BistroSquare*	*FisherHut*	*LoneMonk*	*LOU*	*PavilionChair*	*PavilionPond*
5k	NeRF [7]	0.420	0.560	0.373	0.646	0.658	0.711	0.503	0.448	0.418	0.368	0.561
	mip-NeRF 360 [2]	0.613	0.662	0.470	0.768	0.624	0.573	0.462	0.531	0.495	0.536	0.596
	TensoRF [3]	0.453	0.595	0.328	0.834	0.658	0.639	0.479	0.465	0.294	0.457	0.506
	DVGO [10]	0.605	0.714	0.425	0.788	0.721	0.711	0.499	0.544	0.400	0.529	0.584
	EgoNeRF	**0.178**	**0.370**	**0.249**	**0.444**	**0.102**	**0.160**	**0.310**	**0.145**	**0.116**	**0.137**	**0.196**
10k	NeRF [7]	0.324	0.491	0.333	0.551	0.562	0.678	0.490	0.361	0.345	0.302	0.468
	mip-NeRF 360 [2]	0.346	0.482	0.275	0.544	0.299	0.303	0.418	0.266	0.257	0.297	0.301
	TensoRF [3]	0.345	0.517	0.274	0.738	0.574	0.566	0.467	0.409	0.228	0.374	0.437
	DVGO [10]	0.501	0.637	0.366	0.720	0.674	0.665	0.485	0.485	0.330	0.471	0.515
	EgoNeRF	**0.128**	**0.323**	**0.189**	**0.350**	**0.074**	**0.126**	**0.281**	**0.115**	**0.095**	**0.099**	**0.164**
100k	NeRF [7]	0.133	0.324	0.234	0.269	0.278	0.496	0.357	0.166	0.148	0.139	0.301
	mip-NeRF 360 [2]	0.135	0.321	0.134	0.308	0.092	0.113	0.261	0.107	0.121	0.099	0.150
	TensoRF [3]	0.237	0.410	0.209	0.647	0.468	0.444	0.424	0.299	0.155	0.269	0.347
	DVGO [10]	0.248	0.426	0.220	0.498	0.487	0.508	0.433	0.325	0.180	0.281	0.325
	EgoNeRF	**0.070**	**0.241**	**0.082**	**0.175**	**0.037**	**0.081**	**0.204**	**0.069**	**0.059**	**0.053**	**0.106**

Table 6. Per-scene quantitative results in terms of LPIPS in *OmniBlender* dataset.

Step	Method	Indoor				Outdoor						
		BarberShop	*ClassRoom*	*ItalianFlat*	*Restroom*	*BistroBike*	*BistroSquare*	*FisherHut*	*LoneMonk*	*LOU*	*PavilionChair*	*PavilionPond*
5k	NeRF [7]	0.801	0.698	0.804	0.603	0.554	0.517	0.737	0.644	0.742	0.777	0.587
	mip-NeRF 360 [2]	0.685	0.613	0.752	0.545	0.539	0.528	0.737	0.570	0.678	0.710	0.538
	TensoRF [3]	0.796	0.702	0.828	0.563	0.567	0.551	0.754	0.653	0.827	0.761	0.593
	DVGO [10]	0.733	0.655	0.796	0.572	0.536	0.523	0.744	0.611	0.752	0.745	0.582
	EgoNeRF	**0.905**	**0.766**	**0.849**	**0.694**	**0.906**	**0.829**	**0.777**	**0.875**	**0.898**	**0.881**	**0.738**
10k	NeRF [7]	0.838	0.732	0.820	0.636	0.594	0.532	0.747	0.714	0.786	0.800	0.627
	mip-NeRF 360 [2]	0.845	0.724	0.848	0.637	0.764	0.723	0.768	0.777	0.852	0.809	0.686
	TensoRF [3]	0.839	0.730	0.845	0.592	0.605	0.576	0.759	0.684	0.867	0.776	0.609
	DVGO [10]	0.769	0.682	0.813	0.595	0.553	0.534	0.751	0.637	0.791	0.757	0.592
	EgoNeRF	**0.930**	**0.794**	**0.876**	**0.761**	**0.930**	**0.862**	**0.788**	**0.901**	**0.914**	**0.905**	**0.774**
100k	NeRF [7]	0.927	0.813	0.858	0.809	0.761	0.608	0.779	0.860	0.894	0.885	0.736
	mip-NeRF 360 [2]	0.937	0.803	0.912	0.783	0.924	0.881	**0.813**	0.910	0.924	0.913	0.790
	TensoRF [3]	0.887	0.782	0.871	0.624	0.668	0.608	0.770	0.735	0.906	0.810	0.641
	DVGO [10]	0.874	0.765	0.863	0.688	0.646	0.599	0.769	0.711	0.873	0.802	0.635
	EgoNeRF	**0.960**	**0.843**	**0.930**	**0.873**	**0.962**	**0.909**	0.811	**0.940**	**0.935**	**0.941**	**0.831**

Table 7. Per-scene quantitative results in terms of SSIM in *OmniBlender* dataset.

Step	Method	Indoor				Outdoor						
		BarberShop	*ClassRoom*	*ItalianFlat*	*Restroom*	*BistroBike*	*BistroSquare*	*FisherHut*	*LoneMonk*	*LOU*	*PavilionChair*	*PavilionPond*
5k	NeRF [7]	0.763	0.682	0.796	0.601	0.487	0.479	0.712	0.614	0.695	0.728	0.561
	mip-NeRF 360 [2]	0.616	0.573	0.725	0.540	0.468	0.494	0.712	0.536	0.609	0.659	0.502
	TensoRF [3]	0.759	0.689	0.829	0.556	0.503	0.530	0.737	0.650	0.797	0.720	0.576
	DVGO [10]	0.677	0.627	0.785	0.574	0.462	0.492	0.722	0.592	0.709	0.697	0.559
	EgoNeRF	**0.895**	**0.760**	**0.852**	**0.704**	**0.894**	**0.824**	**0.772**	**0.871**	**0.888**	**0.853**	**0.722**
10k	NeRF [7]	0.809	0.726	0.818	0.641	0.538	0.495	0.725	0.687	0.746	0.756	0.600
	mip-NeRF 360 [2]	0.820	0.709	0.852	0.638	0.720	0.701	0.755	0.759	0.828	0.768	0.657
	TensoRF [3]	0.814	0.724	0.852	0.581	0.552	0.562	0.744	0.693	0.846	0.740	0.598
	DVGO [10]	0.725	0.664	0.808	0.598	0.484	0.509	0.732	0.629	0.759	0.714	0.574
	EgoNeRF	**0.925**	**0.792**	**0.883**	**0.764**	**0.922**	**0.861**	**0.784**	**0.899**	**0.907**	**0.882**	**0.757**
100k	NeRF [7]	0.922	0.821	0.868	0.803	0.746	0.588	0.768	0.849	0.882	0.856	0.712
	mip-NeRF 360 [2]	0.934	0.801	0.919	0.782	0.912	0.876	0.805	0.902	0.917	0.887	0.767
	TensoRF [3]	0.875	0.793	0.885	0.619	0.634	0.608	0.762	0.748	0.896	0.787	0.643
	DVGO [10]	0.862	0.772	0.875	0.703	0.608	0.596	0.759	0.729	0.867	0.778	0.641
	EgoNeRF	**0.960**	**0.848**	**0.937**	**0.870**	**0.959**	**0.913**	**0.813**	**0.939**	**0.933**	**0.927**	**0.823**

Table 8. Per-scene quantitative results in terms of SSIM^{WS} in *OmniBlender* dataset.

Step	Method	*Bricks*	*Bridge*	*BridgeUnder*	*CatTower*	*Center*	*Farm*	*Flower*	*GalleryChair*	*GalleryPillar*	*Garden*	*Poster*
	NeRF [7]	20.05	20.91	21.71	21.65	24.75	19.84	18.91	24.75	24.34	24.06	22.03
	mip-NeRF 360 [2]	20.22	20.87	20.87	22.15	25.53	20.07	19.25	24.96	24.90	25.13	21.34
5k	TensoRF [3]	20.97	21.75	21.92	22.39	26.91	20.89	20.09	25.91	26.02	25.13	23.20
	DVGO [10]	20.19	20.91	20.70	21.77	26.15	20.33	19.60	25.27	25.20	25.00	21.82
	EgoNeRF	**22.44**	**22.86**	**23.99**	**23.49**	**27.88**	**21.83**	**21.31**	**26.98**	**27.29**	**26.31**	**25.29**
	NeRF [7]	20.64	21.48	22.43	22.18	25.81	20.29	19.52	25.60	25.30	24.49	22.79
	mip-NeRF 360 [2]	22.08	22.73	23.37	23.38	27.73	21.66	20.93	27.03	26.97	26.09	25.11
10k	TensoRF [3]	21.62	22.17	22.78	22.80	27.61	21.20	20.63	26.68	26.60	25.57	24.31
	DVGO [10]	20.84	21.64	21.43	22.29	26.92	20.62	20.12	25.55	26.16	25.24	23.07
	EgoNeRF	**22.68**	**22.98**	**24.25**	**23.69**	**28.07**	**21.98**	**21.51**	**27.13**	**27.50**	**26.50**	**25.57**
	NeRF [7]	22.71	23.39	24.82	23.99	28.54	21.68	21.44	27.77	27.43	26.42	25.85
	mip-NeRF 360 [2]	**23.39**	**23.80**	**25.01**	**24.46**	**29.30**	**22.48**	**22.01**	**28.36**	**28.42**	**27.03**	**27.00**
100k	TensoRF [3]	23.08	23.27	24.56	23.84	29.25	22.02	21.72	28.04	28.14	26.47	26.38
	DVGO [10]	22.97	22.94	23.96	23.84	29.21	21.79	21.72	26.49	28.28	26.40	26.24
	EgoNeRF	23.37	23.40	24.94	24.23	28.45	22.23	21.80	27.78	28.02	26.87	26.62

Table 9. Per-scene quantitative results in terms of PSNR in *Ricoh360* dataset.

Step	Method	*Bricks*	*Bridge*	*BridgeUnder*	*CatTower*	*Center*	*Farm*	*Flower*	*GalleryChair*	*GalleryPillar*	*Garden*	*Poster*
	NeRF [7]	21.79	22.68	23.15	23.47	26.71	21.52	20.73	27.14	25.78	25.62	23.41
	mip-NeRF 360 [2]	22.03	22.68	22.60	24.09	27.54	21.88	21.21	27.39	26.35	26.75	22.77
5k	TensoRF [3]	22.95	23.88	23.71	24.45	28.91	22.78	22.10	28.53	27.65	26.84	24.95
	DVGO [10]	22.47	23.23	22.74	23.89	28.30	22.43	21.74	28.31	26.99	26.86	23.49
	EgoNeRF	**24.81**	**25.25**	**25.95**	**25.63**	**30.39**	**23.97**	**23.44**	**29.89**	**29.19**	**28.01**	**27.61**
	NeRF [7]	22.35	23.30	23.87	24.05	27.60	22.00	21.39	27.86	26.75	26.04	24.21
	mip-NeRF 360 [2]	24.10	24.80	25.20	25.36	30.15	23.59	22.91	29.58	28.69	27.65	27.09
10k	TensoRF [3]	23.59	24.33	24.50	24.87	29.59	23.07	22.61	29.18	28.17	27.19	26.17
	DVGO [10]	23.04	23.95	23.43	24.45	29.22	22.79	22.31	28.84	27.99	27.19	24.82
	EgoNeRF	**25.08**	**25.46**	**26.26**	**25.84**	**30.65**	**24.16**	**23.67**	**30.07**	**29.43**	**28.19**	**27.93**
	NeRF [7]	24.42	25.09	26.38	25.90	30.27	23.40	23.38	30.04	29.01	27.87	27.37
	mip-NeRF 360 [2]	25.52	**25.96**	26.90	**26.48**	**31.52**	**24.50**	**24.03**	**30.89**	**30.20**	**28.66**	**29.13**
100k	TensoRF [3]	25.06	25.42	26.26	26.00	31.12	23.97	23.72	30.56	29.81	28.08	28.46
	DVGO [10]	25.23	25.41	25.91	26.20	31.42	24.00	24.00	30.62	30.36	28.38	28.55
	EgoNeRF	**25.74**	25.88	**27.01**	26.41	31.08	24.40	24.01	30.64	30.07	28.53	28.70

Table 10. Per-scene quantitative results in terms of PSNR^{WS} in *Ricoh360* dataset.

Step	Method	*Bricks*	*Bridge*	*BridgeUnder*	*CatTower*	*Center*	*Farm*	*Flower*	*GalleryChair*	*GalleryPillar*	*Garden*	*Poster*
	NeRF [7]	0.587	0.542	0.559	0.631	0.516	0.599	0.725	0.572	0.460	0.577	0.566
	mip-NeRF 360 [2]	0.569	0.524	0.629	0.575	0.472	0.583	0.662	0.518	0.438	0.531	0.599
5k	TensoRF [3]	0.531	0.509	0.591	0.617	0.457	0.537	0.709	0.546	0.408	0.562	0.496
	DVGO [10]	0.547	0.540	0.680	0.636	0.473	0.574	0.714	0.552	0.456	0.574	0.552
	EgoNeRF	**0.317**	**0.330**	**0.309**	**0.395**	**0.251**	**0.342**	**0.434**	**0.336**	**0.238**	**0.371**	**0.323**
	NeRF [7]	0.547	0.505	0.499	0.610	0.484	0.554	0.698	0.538	0.414	0.562	0.509
	mip-NeRF 360 [2]	0.371	0.363	0.390	0.460	0.293	0.366	0.517	0.401	0.275	0.427	0.357
10k	TensoRF [3]	0.469	0.459	0.484	0.575	0.387	0.484	0.653	0.485	0.349	0.529	0.415
	DVGO [10]	0.499	0.495	0.613	0.606	0.437	0.531	0.678	0.522	0.411	0.553	0.478
	EgoNeRF	**0.292**	**0.312**	**0.282**	**0.380**	**0.236**	**0.322**	**0.424**	**0.323**	**0.227**	**0.361**	**0.290**
	NeRF [7]	0.396	0.378	0.292	0.472	0.295	0.404	0.540	0.383	0.288	0.424	0.350
	mip-NeRF 360 [2]	**0.246**	**0.258**	0.238	**0.337**	**0.192**	**0.265**	**0.378**	0.301	**0.180**	**0.312**	**0.239**
100k	TensoRF [3]	0.342	0.360	0.332	0.487	0.279	0.378	0.530	0.385	0.274	0.457	0.314
	DVGO [10]	0.331	0.355	0.365	0.481	0.302	0.375	0.512	0.387	0.271	0.458	0.302
	EgoNeRF	0.254	0.276	**0.231**	0.369	0.207	0.312	0.412	**0.288**	0.206	0.342	0.245

Table 11. Per-scene quantitative results in terms of LPIPS in *Ricoh360* dataset.

Step	Method	*Bricks*	*Bridge*	*BridgeUnder*	*CatTower*	*Center*	*Farm*	*Flower*	*GalleryChair*	*GalleryPillar*	*Garden*	*Poster*
	NeRF [7]	0.577	0.624	0.621	0.606	0.768	0.540	0.512	0.774	0.755	0.646	0.722
	mip-NeRF 360 [2]	0.563	0.605	0.574	0.617	0.748	0.507	0.516	0.758	0.737	0.657	0.667
5k	TensoRF [3]	0.617	0.647	0.631	0.628	0.803	0.571	0.542	0.794	0.785	0.664	0.758
	DVGO [10]	0.609	0.637	0.598	0.620	0.797	0.564	0.532	0.784	0.773	0.660	0.732
	EgoNeRF	**0.707**	**0.704**	**0.746**	**0.673**	**0.844**	**0.641**	**0.606**	**0.829**	**0.830**	**0.706**	**0.818**
	NeRF [7]	0.594	0.634	0.650	0.615	0.783	0.549	0.523	0.783	0.769	0.653	0.742
	mip-NeRF 360 [2]	0.676	0.695	0.723	0.668	0.838	0.626	0.593	0.823	0.821	0.695	0.816
10k	TensoRF [3]	0.639	0.657	0.667	0.640	0.819	0.585	0.557	0.806	0.800	0.672	0.787
	DVGO [10]	0.628	0.650	0.616	0.631	0.809	0.574	0.545	0.793	0.790	0.665	0.762
	EgoNeRF	**0.720**	**0.713**	**0.763**	**0.681**	**0.850**	**0.651**	**0.617**	**0.834**	**0.835**	**0.713**	**0.831**
	NeRF [7]	0.670	0.687	0.755	0.659	0.830	0.607	0.579	0.825	0.815	0.690	0.815
	mip-NeRF 360 [2]	**0.761**	**0.748**	**0.801**	**0.713**	**0.872**	**0.690**	**0.651**	**0.859**	**0.860**	**0.739**	**0.866**
100k	TensoRF [3]	0.701	0.695	0.736	0.665	0.849	0.631	0.595	0.831	0.831	0.692	0.832
	DVGO [10]	0.708	0.696	0.717	0.670	0.849	0.628	0.601	0.826	0.833	0.689	0.832
	EgoNeRF	0.748	0.733	0.791	0.697	0.858	0.665	0.633	0.847	0.846	0.725	0.853

Table 12. Per-scene quantitative results in terms of SSIM in *Ricoh360* dataset.

Step	Method	*Bricks*	*Bridge*	*BridgeUnder*	*CatTower*	*Center*	*Farm*	*Flower*	*GalleryChair*	*GalleryPillar*	*Garden*	*Poster*
	NeRF [7]	0.553	0.583	0.579	0.587	0.754	0.496	0.499	0.766	0.729	0.613	0.697
	mip-NeRF 360 [2]	0.541	0.565	0.533	0.598	0.730	0.467	0.503	0.746	0.706	0.625	0.635
5k	TensoRF [3]	0.604	0.614	0.600	0.616	0.794	0.534	0.535	0.789	0.767	0.635	0.747
	DVGO [10]	0.601	0.605	0.568	0.607	0.788	0.528	0.528	0.782	0.755	0.631	0.717
	EgoNeRF	**0.704**	**0.681**	**0.732**	**0.668**	**0.842**	**0.618**	**0.608**	**0.831**	**0.824**	**0.683**	**0.824**
	NeRF [7]	0.571	0.595	0.611	0.596	0.770	0.506	0.511	0.775	0.744	0.621	0.721
	mip-NeRF 360 [2]	0.665	0.665	0.700	0.657	0.832	0.594	0.588	0.820	0.808	0.668	0.814
10k	TensoRF [3]	0.628	0.627	0.638	0.630	0.811	0.550	0.553	0.804	0.785	0.644	0.783
	DVGO [10]	0.623	0.622	0.590	0.621	0.802	0.542	0.545	0.794	0.776	0.640	0.755
	EgoNeRF	**0.718**	**0.692**	**0.750**	**0.677**	**0.849**	**0.630**	**0.620**	**0.837**	**0.831**	**0.691**	**0.840**
	NeRF [7]	0.649	0.649	0.729	0.643	0.821	0.569	0.571	0.820	0.801	0.660	0.807
	mip-NeRF 360 [2]	**0.754**	**0.727**	**0.788**	**0.706**	**0.869**	**0.668**	**0.651**	**0.859**	**0.857**	**0.716**	**0.873**
100k	TensoRF [3]	0.698	0.672	0.717	0.662	0.845	0.606	0.595	0.832	0.824	0.669	0.839
	DVGO [10]	0.715	0.679	0.705	0.672	0.849	0.609	0.610	0.835	0.833	0.671	0.844
	EgoNeRF	0.747	0.713	0.782	0.693	0.860	0.645	0.637	0.850	0.844	0.704	0.861

Table 13. Per-scene quantitative results in terms of SSIM^{WS} in *Ricoh360* dataset.

	Chair	*Drums*	*Ficus*	*Hotdog*	*Lego*	*Materials*	*Mic*	*Ship*
PSNR	34.15	25.30	31.72	36.19	33.88	28.87	33.09	28.85
LPIPS	0.017	0.062	0.020	0.020	0.014	0.037	0.015	0.111
SSIM	0.977	0.928	0.972	0.978	0.972	0.941	0.982	0.862

Table 14. Per-scene breakdown of EgoNeRF results for Synthetic-NeRF [7] dataset.

Method	*Bicycle*	*Bonsai*	*Counter*	*Garden*	*Kitchen*	*Room*	*Stump*
NeRF [7]	21.76	26.81	25.67	23.11	26.31	28.56	21.73
mip-NeRF [1]	21.69	27.13	25.59	23.16	26.47	28.73	23.10
mip-NeRF 360 [2]	24.37	33.46	29.55	26.98	32.23	31.63	26.40
TensoRF [3]	19.86	24.99	22.58	21.96	22.78	26.13	20.91
DVGO [10]	18.88	16.89	22.54	19.39	21.63	26.89	20.40
EgoNeRF	21.88	28.62	25.82	24.41	26.48	29.68	23.94

Table 15. Per-scene quantitative results in terms of PSNR in mip-NeRF 360 [2] dataset.

Method	*Bicycle*	*Bonsai*	*Counter*	*Garden*	*Kitchen*	*Room*	*Stump*
NeRF [7]	0.536	0.398	0.394	0.415	0.335	0.353	0.551
mip-NeRF [1]	0.541	0.370	0.390	0.422	0.336	0.346	0.490
mip-NeRF 360 [2]	0.301	0.176	0.204	0.170	0.127	0.211	0.261
TensoRF [3]	0.838	0.414	0.578	0.728	0.578	0.456	0.738
DVGO [10]	0.687	0.639	0.405	0.529	0.430	0.331	0.589
EgoNeRF	0.507	0.220	0.319	0.318	0.250	0.273	0.357

Table 16. Per-scene quantitative results in terms of LPIPS in mip-NeRF 360 [2] dataset.

Method	*Bicycle*	*Bonsai*	*Counter*	*Garden*	*Kitchen*	*Room*	*Stump*
NeRF [7]	0.455	0.792	0.775	0.546	0.749	0.843	0.453
mip-NeRF [1]	0.454	0.818	0.779	0.543	0.745	0.851	0.517
mip-NeRF 360 [2]	0.685	0.941	0.894	0.813	0.920	0.913	0.744
TensoRF [3]	0.345	0.754	0.681	0.411	0.552	0.777	0.387
DVGO [10]	0.365	0.553	0.728	0.476	0.627	0.814	0.428
EgoNeRF	0.464	0.847	0.767	0.631	0.765	0.853	0.576

Table 17. Per-scene quantitative results in terms of SSIM in mip-NeRF 360 [2] dataset.