Adaptive Shells for Efficient Neural Radiance Field Rendering Supplementary Material

In this supplementary material, we provide additional results (Section 1) and implementation details (Section 2).

1 ADDITIONAL RESULTS

We provide additional qualitative results on *MipNeRF360* data set (Figure 1) and *DTU* data set (Figure 2, 3), as well as per-scene quantitative numbers for all data sets (Tables 1–11).

2 IMPLEMENTATION DETAILS

Level set evolution. To perform the level set evolution, we extract the initial SDF values on a 512^3 grid and separately dilate and erode the zero level set for 50 iterations with the timestep as 0.1. For dilation, we use $\beta_d = 1$ and the density threshold σ_{min} is set to 0.01 for all data sets. These values were determined empirically such that the dilated level set sufficiently covers thin structures. We set the erosion hyperparameters to $\beta_e = 0.001, v_{\text{max}} = 100$. The evolution process takes approximately 2 seconds which is negligible compared to the other steps of our training pipeline.

Narrow-band rendering. When sampling query points within the shells, we set the maximum number of samples per interval as $N_{max} = 16$, and the maximum cap for depth peeling $dp_{max} = 20$. For *Shelly*, *DTU* and *MipNeRF360* data sets, we use single-sample threshold $w_s = 0.02$ and inter-sample spacing $\delta_s = 0.01$. For *NeRF-Synthetic* data set, we use single-sample threshold $w_s = 0.005$ and inter-sample spacing $\delta_s = 0.0025$.

Representing the background. For *DTU* data set, we combine the main volume representation with a spherical background placed at infinity (only dependent on the ray direction). Similar to our main network, the spherical background is represented using a combination of a hash encoding (4 levels, 2D features per level) and a small MLP such that $\mathbf{c} = \text{NN}_\theta^{\text{bckg}}([\Psi(\mathbf{d})])$. All the rays that completely miss the extracted shell obtain the color from the background, which requires a single sample evaluation.

When training on *MipNeRF360* data set, we follow prior works [Yariv et al. 2023] and extend our method with the scene contraction proposed in [Barron et al. 2022]. Specifically, we map the scene outside the unit sphere into a sphere with radius 2 using the scene contraction function

$$\text{contract}(\mathbf{x}) = \begin{cases} \mathbf{x}, & ||\mathbf{x}|| \leq 1, \\ (2 - \frac{1}{||\mathbf{x}||})\frac{\mathbf{x}}{||\mathbf{x}||}, & ||\mathbf{x}|| > 1. \end{cases} \quad (1)$$

Training details. During training, we linearly warm up the learning rate to 1×10^{-2} in the first 5k iterations, and then exponentially decay it to 1×10^{-4} at the end of training. In all experiments, we use the AdamW optimizer with weight decay as 1e-2. For *Shelly*, *DTU* and *NeRFSynthetic* data sets, we train for a total of 300k iterations, where the first 200k iterations are used for the first stage (full-ray formulation) and the remaining 100k for the second stage (finetuning within the narrow band). We use the batch-size of 4096 rays. For each scene, the training takes 8h on a single A100 GPU. For larger *MipNeRF360* scenes, we increase the full-ray training to 500k iterations. We adopt the progressive training scheme [Li et al. 2023; Wang et al. 2022], where we initially enable the 8 coarsest levels of features (with the remaining feature channels set to zero), we then add one level every 5k iterations until reaching the maximum number of levels which equals 14.

Author's address:

REFERENCES

Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. 2021. Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields. *ICCV* (2021).

Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. 2022. Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields. *CVPR* (2022).

Zhiqin Chen, Thomas Funkhouser, Peter Hedman, and Andrea Tagliasacchi. 2023. MobileNeRF: Exploiting the Polygon Rasterization Pipeline for Efficient Neural Field Rendering on Mobile Architectures. In *The Conference on Computer Vision and Pattern Recognition (CVPR).*

Max Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H. Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. 2023. Neuralangelo: High-Fidelity Neural Surface Reconstruction. In *Conference on Computer Vision and Pattern Recognition (CVPR).*

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV.*

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *ACM Trans. Graph.* 41, 4, Article 102 (July 2022), 15 pages. https://doi.org/10.1145/3528223.3530127

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. *NeurIPS* (2021).

Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. 2022. NeuS2: Fast Learning of Neural Implicit Surfaces for Multi-view Reconstruction.

Lior Yariv, Peter Hedman, Christian Reiser, Dor Verbin, Pratul P Srinivasan, Richard Szeliski, Jonathan T Barron, and Ben Mildenhall. 2023. BakedSDF: Meshing Neural SDFs for Real-Time View Synthesis. *arXiv preprint arXiv:2302.14859* (2023).

Table 1. Per-scene quantitative PSNR comparison on *Shelly* data set.

PSNR ↑	NeRF [Mildenhall et al. 2020]	MipNeRF [Barron et al. 2021]	NeuS [Wang et al. 2021]	I-NGP [Müller et al. 2022]	MobileNeRF [Chen et al. 2023]	Ours (full ray)	Ours
Fernvase	31.77	32.54	29.26	33.48	31.38	33.93	36.47
Pug	31.34	32.26	31.38	32.70	31.50	33.60	35.83
Woolly	28.33	29.18	27.90	31.13	31.61	31.61	34.19
Horse	34.02	37.12	30.92	37.67	36.48	39.40	40.57
Khady	29.01	29.88	28.29	29.21	26.84	31.09	31.22
Kitten	33.18	34.54	32.16	35.13	31.93	35.94	37.82
Average	31.28	32.59	29.98	33.22	31.62	34.26	36.02

Table 2. Per-scene quantitative LPIPS comparison on *Shelly* data set.

LPIPS ↓	NeRF [Mildenhall et al. 2020]	MipNeRF [Barron et al. 2021]	NeuS [Wang et al. 2021]	I-NGP [Müller et al. 2022]	MobileNeRF [Chen et al. 2023]	Ours (full ray)	Ours
Fernvase	0.093	0.088	0.094	0.068	0.074	0.065	0.046
Pug	0.198	0.190	0.209	0.156	0.167	0.132	0.093
Woolly	0.241	0.217	0.232	0.181	0.163	0.139	0.089
Horse	0.071	0.064	0.067	0.049	0.057	0.036	0.029
Khady	0.246	0.239	0.251	0.215	0.218	0.185	0.160
Kitten	0.094	0.092	0.097	0.079	0.094	0.066	0.056
Average	0.157	0.148	0.158	0.125	0.129	0.104	0.079

Table 3. Per-scene quantitative SSIM comparison on *Shelly* dataset.

SSIM ↑	NeRF [Mildenhall et al. 2020]	MipNeRF [Barron et al. 2021]	NeuS [Wang et al. 2021]	I-NGP [Müller et al. 2022]	MobileNeRF [Chen et al. 2023]	Ours (full ray)	Ours
Fernvase	0.937	0.940	0.932	0.955	0.944	0.964	0.976
Pug	0.863	0.868	0.865	0.896	0.885	0.910	0.947
Woolly	0.805	0.822	0.803	0.876	0.891	0.896	0.950
Horse	0.975	0.980	0.973	0.985	0.980	0.988	0.992
Khady	0.831	0.835	0.833	0.852	0.823	0.862	0.881
Kitten	0.949	0.954	0.949	0.967	0.942	0.969	0.979
Average	0.893	0.899	0.893	0.922	0.911	0.932	0.954

Table 4. Per-scene sample count of our method on *Shelly* data set.

	Fernvase	Pug	Woolly	Horse	Khady	Kitten	Average
Ours	1.514	1.921	2.040	0.425	3.292	1.228	1.737

Table 5. Per-scene quantitative results on *DTU* data set. We report the PSNR, LPIPS and SSIM results for each scene and compare them with baselines.

	I-NGP [Müller et al. 2022]			Neus [Wang et al. 2021]			Nerf [Mildenhall et al. 2020]			MipNeRF [Barron et al. 2021]			Ours (full ray)			Ours		
Scene	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
24	29.85	0.237	0.871	26.22	0.324	0.787	24.87	0.364	0.751	25.60	0.359	0.763	32.42	0.110	0.931	30.91	0.102	0.934
37	25.05	0.169	0.869	23.63	0.191	0.835	21.31	0.225	0.792	22.97	0.212	0.815	27.46	0.095	0.932	27.38	0.096	0.938
40	28.80	0.291	0.829	26.38	0.344	0.755	24.98	0.363	0.715	25.90	0.343	0.743	31.16	0.161	0.913	32.10	0.150	0.931
55	27.80	0.171	0.923	25.56	0.171	0.893	23.61	0.202	0.846	24.20	0.192	0.865	32.09	0.067	0.974	32.35	0.065	0.975
63	33.06	0.075	0.961	30.51	0.115	0.954	29.94	0.117	0.945	30.41	0.114	0.947	34.94	0.049	0.974	34.10	0.044	0.975
65	33.75	0.091	0.962	30.71	0.108	0.962	29.86	0.118	0.948	30.38	0.114	0.951	34.90	0.069	0.069	35.08	0.066	0.974
69	31.21	0.149	0.946	27.97	0.184	0.938	28.15	0.217	0.913	28.75	0.213	0.916	32.15	0.104	0.961	31.79	0.102	0.960
83	35.28	0.055	0.977	33.57	0.081	0.974	33.13	0.077	0.971	33.29	0.077	0.971	36.75	0.040	0.983	37.12	0.036	0.985
97	28.50	0.140	0.937	26.93	0.144	0.936	26.58	0.168	0.918	26.70	0.172	0.918	29.99	0.086	0.959	29.77	0.093	0.956
105	34.08	0.113	0.948	31.39	0.164	0.932	31.37	0.160	0.927	31.35	0.160	0.926	35.50	0.067	0.967	35.91	0.061	0.970
106	33.31	0.135	0.947	28.80	0.161	0.934	30.63	0.177	0.921	30.92	0.174	0.923	36.29	0.073	0.973	35.81	0.072	0.972
110	29.89	0.118	0.951	28.14	0.145	0.946	28.84	0.140	0.943	28.55	0.140	0.942	32.81	0.073	0.971	33.18	0.071	0.972
114	29.41	0.163	0.925	28.13	0.178	0.921	28.28	0.183	0.907	28.33	0.183	0.908	31.08	0.102	0.955	31.07	0.094	0.956
118	35.23	0.105	0.964	31.60	0.128	0.961	33.42	0.130	0.952	33.23	0.129	0.951	37.67	0.066	0.978	36.71	0.062	0.979
122	35.42	0.074	0.972	34.36	0.090	0.971	32.66	0.102	0.959	32.97	0.098	0.961	37.50	0.047	0.983	37.21	0.044	0.983
Mean	31.38	0.139	0.932	28.93	0.168	0.913	28.51	0.183	0.894	28.90	0.179	0.900	33.51	0.081	0.901	33.37	0.077	0.964

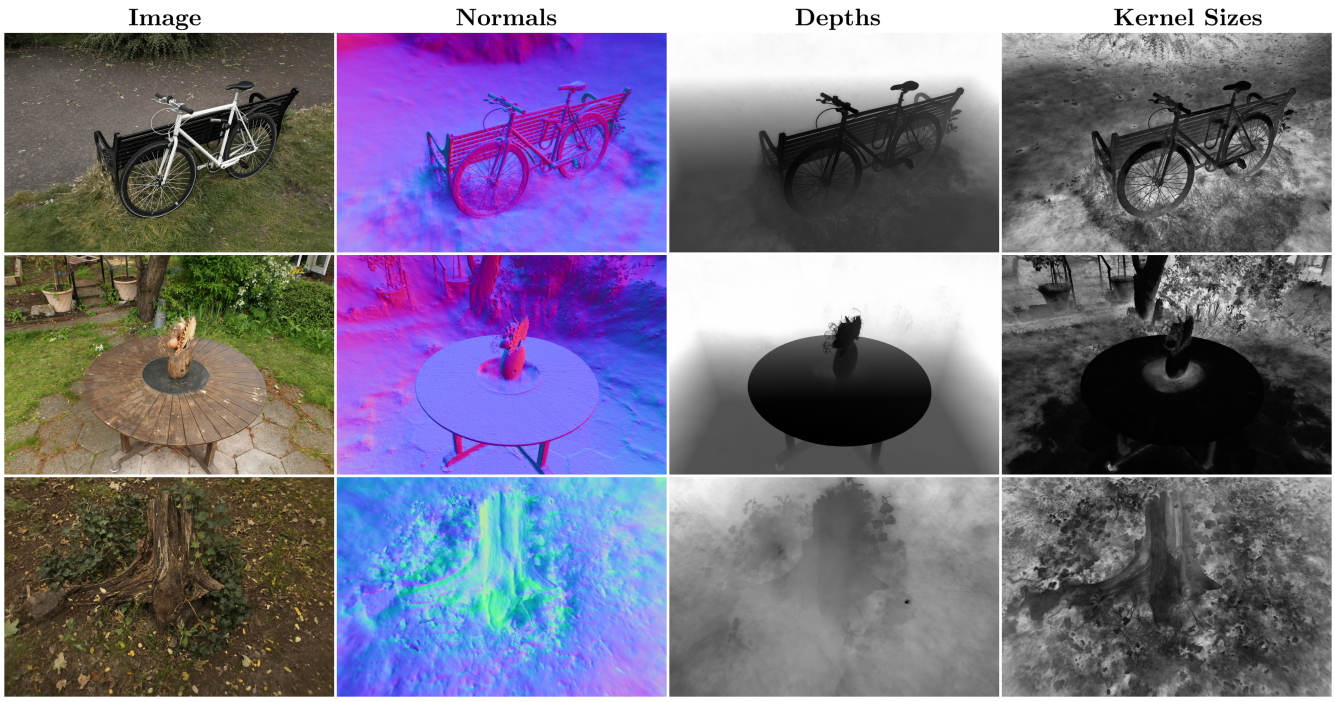

Fig. 1. Qualitative visualization of geometry and kernel size on *MipNeRF360* data set. The kernel size is re-scaled to between 0 and 1. Our method automatically converges to a large kernel size for fuzzy regions such as grass and a small kernel size for sharp surfaces.

Table 6. Per-scene sample count of our method on *DTU* data set.

Scene	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Average
Ours	3.989	5.716	4.085	5.254	3.570	4.385	6.844	4.608	5.882	5.750	4.015	6.492	4.454	3.117	2.910	4.738

Table 7. Per-scene results of our method on *NeRFSynthetic* data set.

Ours (full ray)	Mic	Ficus	Chair	Hotdog	Materials	Drums	Ship	Lego	Average
PSNR	34.46	34.67	35.14	36.17	28.47	25.32	30.27	35.60	32.51
LPIPS	0.012	0.024	0.020	0.029	0.076	0.081	0.121	0.021	0.048
SSIM	0.989	0.985	0.986	0.982	0.941	0.939	0.896	0.981	0.962
Ours	**Mic**	**Ficus**	**Chair**	**Hotdog**	**Materials**	**Drums**	**Ship**	**Lego**	**Average**
PSNR	33.91	33.63	34.94	36.21	27.82	25.19	29.54	33.49	31.84
LPIPS	0.015	0.033	0.023	0.035	0.086	0.086	0.141	0.031	0.056
SSIM	0.988	0.981	0.985	0.981	0.935	0.937	0.877	0.973	0.957
Sample count	1.200	3.097	2.113	3.728	4.235	3.042	6.799	4.035	3.531

Table 8. Per-scene quantitative PSNR comparison on *MipNeRF360* data set.

		NeRF [Mildenhall et al. 2020]	Mip-NeRF [Barron et al. 2021]	Mip-NeRF 360 [Barron et al. 2022]	I-NGP [Müller et al. 2022]	MobileNeRF [Chen et al. 2023]	BakedSDF [Yariv et al. 2023]	Ours (full ray)	Ours
Outdoor	Bicycle	21.76	21.69	24.37	23.67	21.70	22.08	24.07	22.19
	Garden	23.11	23.16	26.98	24.60	23.04	24.53	25.73	25.35
	Stump	21.73	21.21	26.40	23.43	23.96	23.59	23.10	21.96
	Average	22.20	22.02	25.92	23.90	22.90	23.40	24.30	23.17
Indoor	Room	28.56	28.73	31.63	30.16	28.76	28.63	29.61	30.63
	Counter	25.67	25.59	29.55	26.03	24.74	25.63	26.26	25.24
	Kitchen	26.31	26.47	32.23	29.86	26.28	26.88	30.10	28.43
	Bonsai	26.81	27.13	33.46	31.82	23.20	27.67	30.19	32.47
	Average	26.84	26.98	31.72	29.47	25.74	27.20	29.04	29.19

Fig. 2. Full results on the *DTU* data set (part 1).

Fig. 3. Full results on the *DTU* data set (part 2).

Table 9. Per-scene quantitative LPIPS comparison on *MipNeRF360* data set.

		NeRF [Mildenhall et al. 2020]	Mip-NeRF [Barron et al. 2021]	Mip-NeRF 360 [Barron et al. 2022]	I-NGP [Müller et al. 2022]	MobileNeRF [Chen et al. 2023]	BakedSDF [Yariv et al. 2023]	Ours (full ray)	Ours
Outdoor	Bicycle	0.536	0.541	0.301	0.417	0.513	0.394	0.326	0.438
	Garden	0.415	0.422	0.170	0.248	0.396	0.243	0.200	0.247
	Stump	0.551	0.490	0.261	0.441	0.480	0.415	0.421	0.482
	Average	0.501	0.484	0.244	0.369	0.463	0.351	0.316	0.389
Indoor	Room	0.353	0.346	0.211	0.300	0.423	0.310	0.295	0.300
	Counter	0.394	0.390	0.201	0.341	0.476	0.340	0.276	0.395
	Kitchen	0.335	0.336	0.127	0.224	0.393	0.267	0.168	0.219
	Bonsai	0.398	0.370	0.176	0.227	0.522	0.281	0.217	0.225
	Average	0.370	0.361	0.179	0.273	0.453	0.300	0.239	0.285

Table 10. Per-scene quantitative SSIM comparison on *MipNeRF360* data set.

		NeRF [Mildenhall et al. 2020]	Mip-NeRF [Barron et al. 2021]	Mip-NeRF 360 [Barron et al. 2022]	I-NGP [Müller et al. 2022]	MobileNeRF [Chen et al. 2023]	BakedSDF [Yariv et al. 2023]	Ours (full ray)	Ours
Outdoor	Bicycle	0.455	0.454	0.685	0.624	0.426	0.394	0.689	0.544
	Garden	0.546	0.543	0.813	0.708	0.589	0.740	0.814	0.757
	Stump	0.453	0.517	0.744	0.613	0.557	0.597	0.607	0.516
	Average	0.485	0.505	0.747	0.648	0.524	0.577	0.703	0.606
Indoor	Room	0.843	0.851	0.913	0.886	0.836	0.872	0.892	0.895
	Counter	0.775	0.779	0.894	0.826	0.724	0.809	0.872	0.795
	Kitchen	0.749	0.745	0.920	0.869	0.751	0.825	0.904	0.866
	Bonsai	0.792	0.818	0.941	0.925	0.716	0.875	0.933	0.933
	Average	0.790	0.798	0.917	0.877	0.757	0.845	0.900	0.872

Table 11. Per-scene sample count of our method on *MipNeRF360* data set.

	Bicycle	Garden	Stump	Room	Counter	Kitchen	Bonsai	Average
Ours	16.33	14.05	22.81	17.14	22.24	14.49	14.80	17.41