

Bayes’ Rays: Uncertainty Quantification for Neural Radiance Fields

Supplementary Material

A. Discussion on Geometric Uncertainty Evaluation Metrics

Bayes’ Rays introduces a unique perspective by emphasizing the computation of geometric variance in volumetric points rather than directly measuring pixel-space color or depth variance. This approach proves advantageous for tasks such as NeRF clean-up. To show the effectiveness of our geometric uncertainty, we correlate it with depth error, a valuable geometric signal with available ground-truth maps. It is important to note that we are computing the variance of possible deformation vectors on the reconstructed geometry in 3D space, which differs from the scalar depth distribution in pixel-space. Moreover, our geometric variance is only valid up to a scale factor, as it estimates the geometric variance at the reconstructed scene’s scale rather than the ground truth.

Given these factors, we avoid using the negative log likelihood (NLL) as a reliable metric for our estimated geometric uncertainty. Using NLL with Gaussian likelihood assumes equivalence between our estimated geometric distribution and the pixel-space depth distribution at the ground-truth scale. It also assumes a Gaussian distribution for the predicted depth, which is frequently inaccurate, especially when employing a density regularizer (e.g., Laplacian prior) during NeRF training.

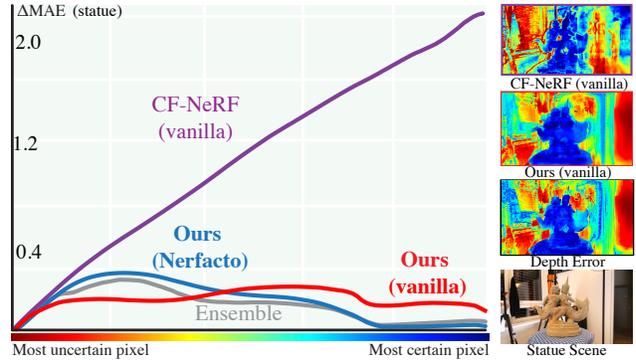
We opt for Area Under Sparsification Error (AUSE) [2, 3] as a more reliable measure for capturing the correlation between estimated geometric uncertainty and depth error in our application. Unlike other metrics, AUSE is not influenced by scene scale or absolute variance values. Instead, it reflects how well the estimated uncertainty describes depth error by comparing pixel depth errors and uncertainties relatively. It is defined as the difference between the area under the curve representing accumulated depth errors when pixels are sorted by uncertainty (A^u), and the area under the curve representing accumulated depth errors when pixels are sorted by depth error (A^d):

$$AUSE = A^u - A^d,$$

$$A^u = \frac{1}{2N} \sum_{i=0}^{N-1} (E_i^u + E_{i+1}^u), \quad A^d = \frac{1}{2N} \sum_{i=0}^{N-1} (E_i^d + E_{i+1}^d)$$

Here, N represents the number of error bins. E_i^u denotes the accumulated normalized absolute depth error of the i th bin when all pixel depths are sorted decreasingly with respect to their uncertainty value, and E_i^d represents the same metric but when the pixels are sorted by depth error decreasingly.

Note that in scenarios where NeRF fails completely in



	CF-NeRF (Vanilla)	Ours (Vanilla)	Ours (Nerfacto)	Ensemble (Nerfacto)
africa	0.35	<u>0.31</u>	0.27	0.18
basket	0.31	0.28	0.28	0.24
statue	0.46	<u>0.18</u>	0.17	0.15
torch	0.97	<u>0.34</u>	0.22	0.19
#0000	0.59	<u>0.35</u>	0.28	0.28
#0079	0.43	<u>0.33</u>	0.35	0.36
#0158	0.55	<u>0.42</u>	0.20	0.19
#0316	0.54	<u>0.38</u>	0.29	0.26

Figure 1. Bayes’ Rays outperforms CF-NeRF when both applied to vanilla NeRF architecture.

learning the scene (resulting in infinite error everywhere), the AUSE metric loses its informative value as the ordering of pixels based on their error becomes meaningless.

B. Comparison of Uncertainty Estimation on Vanilla NeRF

Unlike CF-NeRF [1], our post-hoc, architecture-agnostic method can easily leverage advancements in newer NeRF architectures. To ensure a fair comparison, we provide a comparison between CF-NeRF [1] and Bayes’ Rays when both are using vanilla NeRF as their underlying NeRF architecture, in Figure 1.

C. Application in Next Best View Planning

Epistemic uncertainty can be utilized to enable optimal planning of subsequent views when capturing a scene, as a way of gathering maximum information with a minimal number of views. We validate our derived uncertainty by demonstrating its effectiveness in this application. We mirrored ActiveNeRF [5] settings, conducting experiments with 3 stages of view selection, adding four new images at each stage (a total of 16 images). We compared results on NeRF synthetic dataset scenes (‘hotdog’, ‘lego’, ‘chair’, and ‘drums’) to

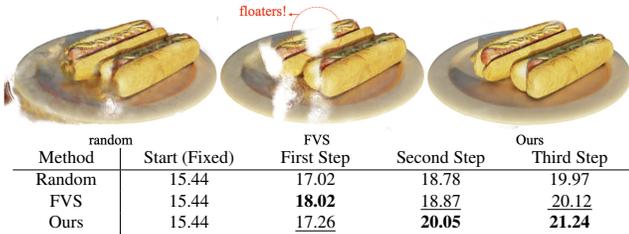


Figure 2. Bayes’ Rays can be applied to next best view planning, outperforming random and farthest view sampling (FVS) strategies. The results after third step of sampling are depicted.

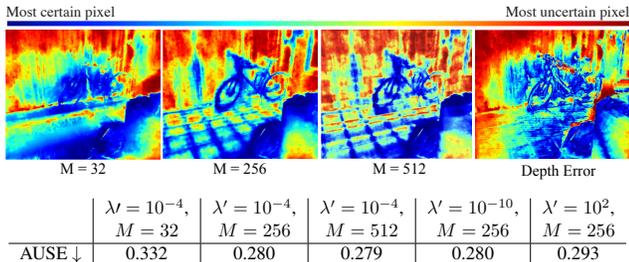


Figure 3. Ablation Study on choice of M and λ on a real scene, shows little sensitivity to choice of $\lambda = \frac{\lambda'}{M^3}$ and diminishing returns for high grid resolution.

baselines from [5]. Even a simple metric like pixel-space average uncertainty from Bayes’ Rays outperforms random and heuristic-based methods, such as farthest view sampling (Figure 2).

D. NeRF Clean-Up Per-Scene Results

We provide per-scene NeRF Clean-up results in Figures 5 to 16 in terms of PSNR, SSIM, LPIPS and Coverage for 10 evenly spaced thresholds set between 0 and 1.

E. Ablation Study on Real Data

We further ablated our choice of λ and grid resolution M on the #0000 scene from the ScanNet dataset as a real-world scene (Figure 3). We compared the AUSE metric across different hyperparameter values in our method. The results revealed minimal sensitivity to changes in λ , with significant shifts in the AUSE metric only happening with large changes in λ . Analyzing the effect of grid resolution on real data echoed the findings depicted in Figure 5 of the main paper. Lower resolution grids tend to underestimate uncertainty, while results stabilize at higher resolutions, indicating diminishing returns.

F. Orthogonality to Aleatoric Uncertainty

One limitation of our approach is its inability to detect aleatoric uncertainty, which refers to uncertainty stemming from inherent noise in the data. A clear example of such noise includes the presence of distractors or transient objects within the training data. As depicted in Figure 4, Bayes’

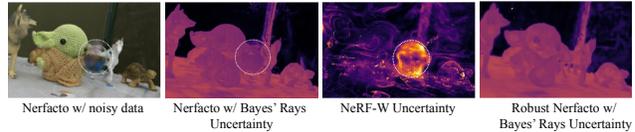


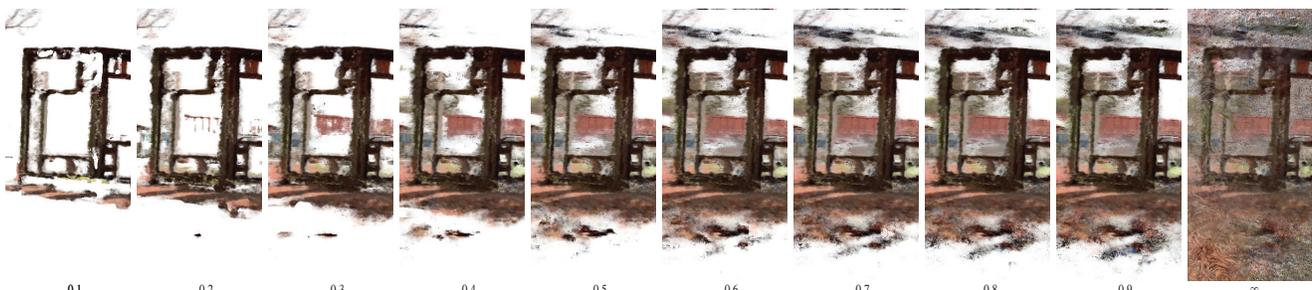
Figure 4. Bayes’ Rays and aleatoric uncertainty detection methods act as complementary.

Rays primarily discerns epistemic uncertainty, which is distinct from the uncertainty identified by aleatoric uncertainty detection techniques (such as NeRF-W [4, eq. 13] and ActiveNeRF [5, eq. 11]). While Bayes’ Rays identifies uncertainty arising from geometric ambiguities (e.g., the flat, textureless table), these methods pinpoint uncertainty arising from the noisy distractors (e.g. the cupcake), therefore complementing each other. Furthermore, our analysis demonstrates that utilizing a method robust to aleatoric noise, such as RobustNeRF [6], preserves the output of Bayes’ Rays, highlighting their orthogonality.



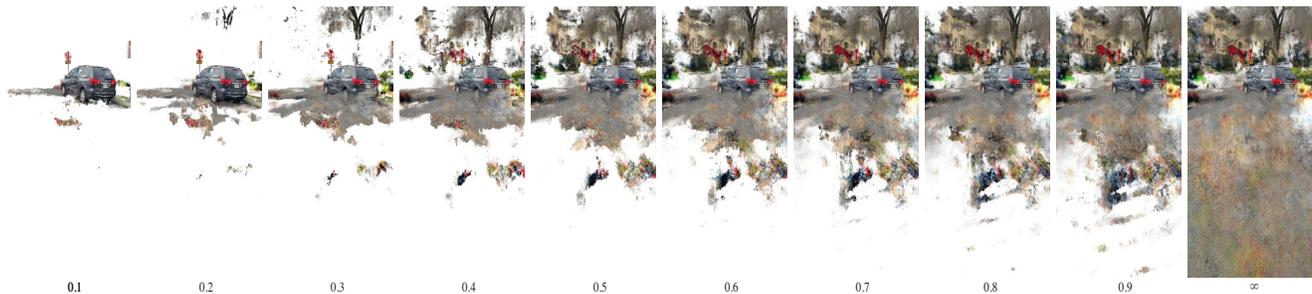
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	14.72	16.64	18.49	18.37	18.24	18.19	18.19	18.20	18.20	18.19
SSIM \uparrow	0.59	0.59	0.62	0.62	0.62	0.62	0.62	0.62	0.62	0.62
LPIPS \downarrow	0.33	0.29	0.26	0.27	0.28	0.28	0.28	0.28	0.29	0.29
Coverage \uparrow	39.49	64.52	79.95	83.58	84.12	84.16	84.16	84.17	84.17	84.19

Figure 5. Aloe Scene



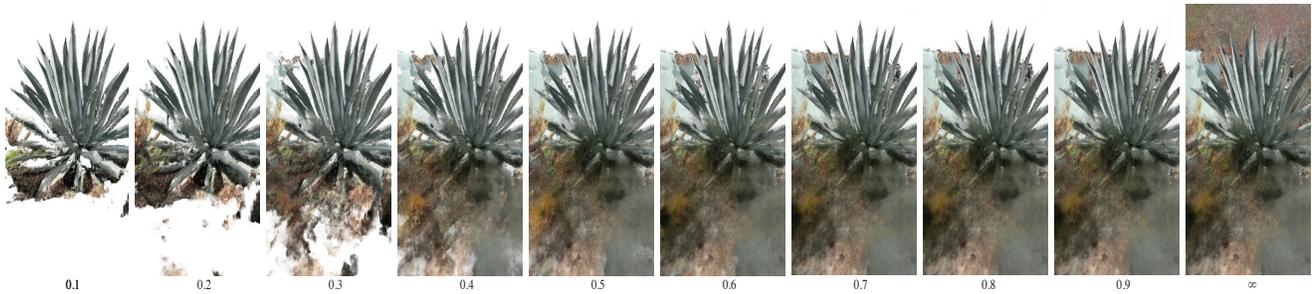
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	13.94	14.18	14.20	14.23	14.24	14.24	14.24	14.23	14.23	14.22
SSIM \uparrow	0.39	0.40	0.40	0.39	0.38	0.38	0.384	0.38	0.38	0.37
LPIPS \downarrow	0.37	0.37	0.37	0.38	0.38	0.39	0.39	0.39	0.39	0.39
Coverage \uparrow	59.99	73.05	80.29	83.039	83.97	84.35	84.51	84.65	84.86	86.36

Figure 6. Art Scene



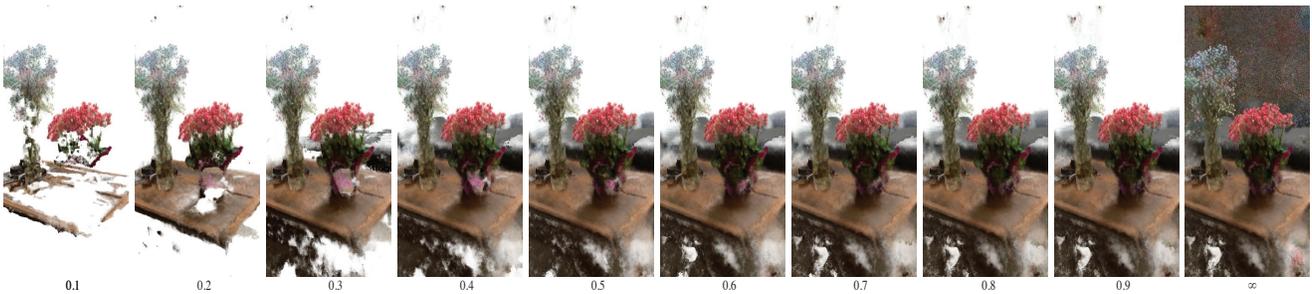
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	15.79	16.29	16.15	15.95	15.84	15.83	15.82	15.77	15.76	15.98
SSIM \uparrow	0.49	0.51	0.49	0.47	0.46	0.45	0.45	0.45	0.45	0.44
LPIPS \downarrow	0.29	0.30	0.33	0.35	0.37	0.38	0.39	0.39	0.40	0.43
Coverage \uparrow	35.91	46.59	60.53	68.27	70.65	72.03	73.01	74.01	75.29	80.17

Figure 7. Car Scene



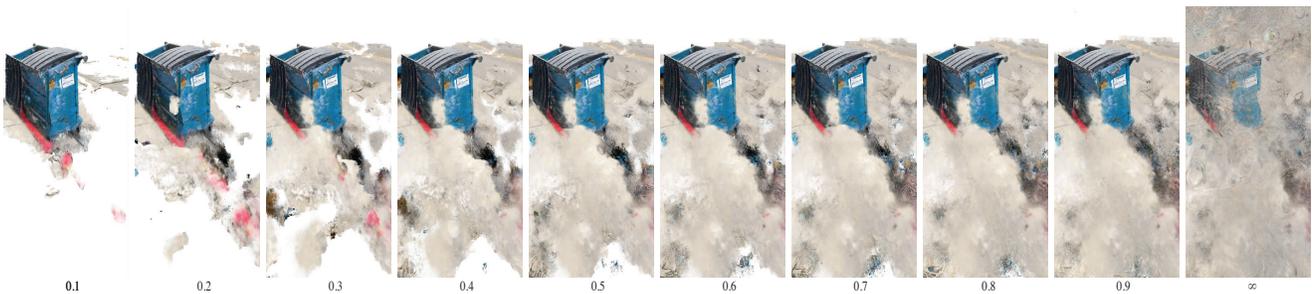
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	15.85	15.68	15.07	14.30	14.14	14.06	13.98	13.92	13.86	13.80
SSIM \uparrow	0.54	0.51	0.47	0.44	0.43	0.42	0.42	0.42	0.42	0.40
LPIPS \downarrow	0.25	0.27	0.32	0.39	0.42	0.43	0.44	0.44	0.45	0.48
Coverage \uparrow	65.18	72.07	83.88	92.58	94.79	95.56	95.92	96.20	96.40	97.05

Figure 8. Century Scene



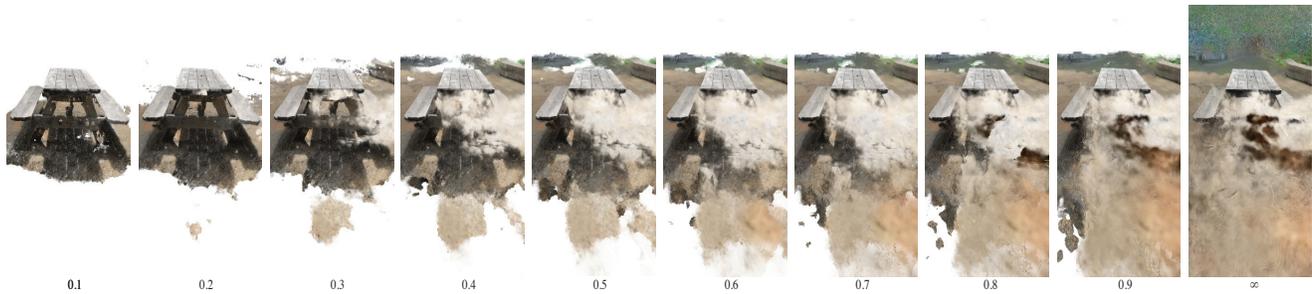
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	15.90	15.21	15.97	15.34	15.18	15.14	15.13	15.11	15.09	14.95
SSIM \uparrow	0.58	0.49	0.48	0.46	0.45	0.45	0.45	0.45	0.45	0.44
LPIPS \downarrow	0.21	0.31	0.37	0.43	0.44	0.45	0.45	0.45	0.45	0.46
Coverage \uparrow	32.67	54.85	77.76	89.26	92.50	93.21	93.34	93.33	93.30	93.22

Figure 9. Flowers Scene



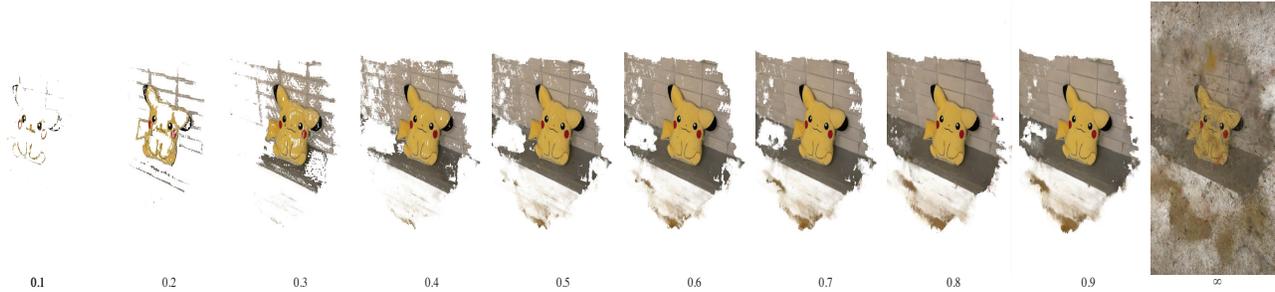
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	15.77	15.09	14.49	14.36	14.41	14.48	14.53	14.58	14.65	14.64
SSIM \uparrow	0.50	0.47	0.439	0.42	0.42	0.41	0.41	0.41	0.41	0.37
LPIPS \downarrow	0.33	0.38	0.43	0.46	0.48	0.48	0.49	0.49	0.50	0.54
Coverage \uparrow	66.32	79.76	87.41	91.87	94.49	96.19	97.24	98.14	99.16	99.99

Figure 10. Garbage Scene



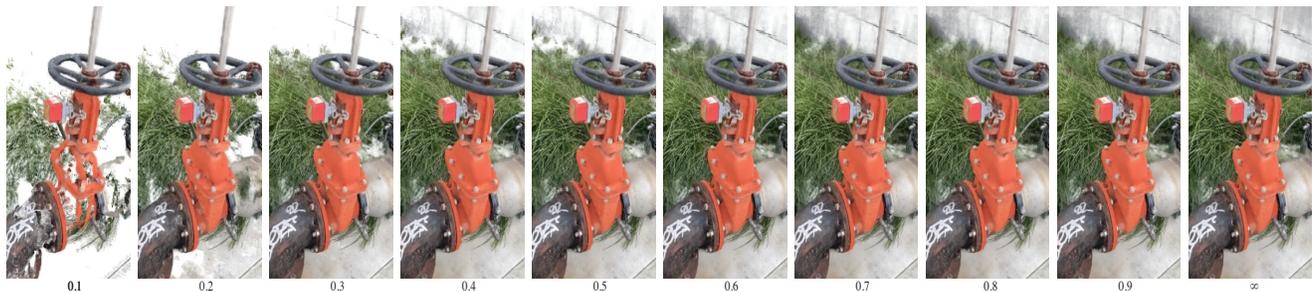
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	16.34	16.01	15.75	15.69	15.71	15.75	15.75	15.78	15.82	15.90
SSIM \uparrow	0.48	0.45	0.38	0.35	0.34	0.34	0.33	0.33	0.33	0.31
LPIPS \downarrow	0.31	0.32	0.39	0.44	0.47	0.48	0.49	0.50	0.50	0.52
Coverage \uparrow	48.94	65.67	83.68	87.92	89.77	90.68	91.27	92.00	93.25	95.66

Figure 11. Picnic Scene



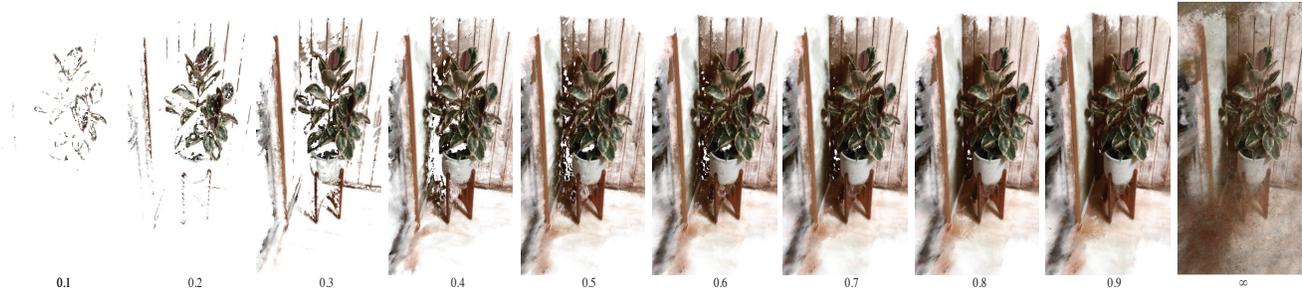
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	20.16	20.727	25.09	25.77	26.07	26.19	26.26	26.26	26.22	20.00
SSIM \uparrow	0.91	0.91	0.90	0.90	0.90	0.89	0.89	0.89	0.89	0.66
LPIPS \downarrow	0.025	0.04	0.04	0.05	0.06	0.07	0.08	0.08	0.09	0.34
Coverage \uparrow	2.80	14.06	39.65	59.55	65.92	68.14	69.29	70.13	71.25	92.22

Figure 12. Pikachu Scene



Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	15.90	18.07	19.28	19.45	19.52	19.52	19.53	19.53	19.53	19.53
SSIM \uparrow	0.61	0.57	0.57	0.57	0.57	0.57	0.57	0.57	0.57	0.56
LPIPS \downarrow	0.21	0.25	0.28	0.28	0.29	0.29	0.29	0.29	0.29	0.29
Coverage \uparrow	42.59	72.50	83.49	89.06	90.55	90.85	90.96	91.03	91.11	91.36

Figure 13. Pipe Scene



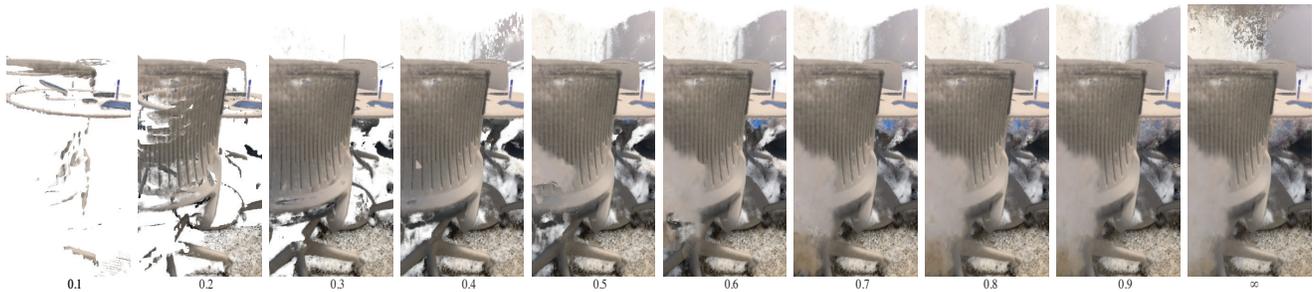
Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	17.32	19.83	20.71	21.07	20.97	20.86	20.79	20.76	20.74	16.87
SSIM \uparrow	0.85	0.80	0.79	0.74	0.72	0.72	0.72	0.71	0.71	0.62
LPIPS \downarrow	0.07	0.10	0.11	0.14	0.16	0.16	0.17	0.17	0.17	0.31
Coverage \uparrow	3.21	10.94	20.34	30.95	34.97	35.94	36.53	37.08	37.80	66.59

Figure 14. Plant Scene



Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	18.45	18.73	17.88	19.68	20.06	20.18	20.20	20.20	20.21	20.21
SSIM \uparrow	0.53	0.55	0.64	0.69	0.71	0.71	0.71	0.71	0.71	0.71
LPIPS \downarrow	0.16	0.19	0.16	0.25	0.24	0.24	0.24	0.24	0.24	0.24
Coverage \uparrow	15.89	26.73	57.10	86.52	91.17	91.55	91.59	91.59	91.60	91.60

Figure 15. Roses Scene



Scene\Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	∞
PSNR \uparrow	14.77	16.27	18.63	19.12	18.67	18.20	17.94	17.74	17.61	17.57
SSIM \uparrow	0.71	0.70	0.70	0.72	0.71	0.70	0.69	0.69	0.69	0.69
LPIPS \downarrow	0.14	0.16	0.23	0.24	0.27	0.28	0.29	0.30	0.31	0.31
Coverage \uparrow	14.95	44.86	66.83	77.12	81.19	82.44	83.05	83.47	83.73	83.97

Figure 16. Table Scene

References

- [1] Conditional-flow nerf: Accurate 3d modelling with reliable uncertainty quantification. *ECCV*, 2022. [1](#)
- [2] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. *ICCV*, 2021. [1](#)
- [3] Eddy Ilg, Özgün Çiçek, Silvio Galesso, Aaron Klein, Osama Makansi, Frank Hutter, and Thomas Brox. Uncertainty estimates and multi-hypotheses networks for optical flow. *ECCV*, 2018. [1](#)
- [4] Ricardo Martin-Brualla, Noha Radwan, Mehdi Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural radiance fields for unconstrained photo collections. *CVPR*, 2020. [2](#)
- [5] Xuran Pan, Zihang Lai, Shiji Song, and Gao Huang. Activenerf: Learning where to see with uncertainty estimation. *ECCV*, 2022. [1](#), [2](#)
- [6] Sara Sabour, Suhani Vora, Daniel Duckworth, Ivan Krasin, David J. Fleet, and Andrea Tagliasacchi. Robustnerf: Ignoring distractors with robust losses. *CVPR*, 2023. [2](#)