HyP-NeRF: Learning Improved NeRF Priors using a HyperNetwork (Supplementary)

Bipasha Sen[*]
MIT CSAIL
bise@mit.edu

Gaurav Singh[*]
IIIT, Hyderabad
gaurav.si[†]

Aditya Agarwal[*]
MIT CSAIL
adityaag@mit.edu

Rohith Agaram
IIIT, Hyderabad
rohith.agaram[†]

K Madhava Krishna
IIIT, Hyderabad
mkrishna@iiit.ac.in

Srinath Sridhar
Brown University
srinath@brown.edu

1 Experiments

		Chairs			Sofa		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
ABO	PixelNeRF [7]	19.00	0.74	0.343	18.49	0.77	0.351
	CodeNeRF [3]	20.51	0.75	0.264	20.38	0.77	0.31
	HyP-NeRF (Ours)	**25.92**	**0.91**	**0.093**	**26.73**	**0.91**	**0.098**
	w/o denoising	24.83	0.87	0.12	25.68	0.87	0.14

Table 1: **Generalization**. Comparison of single-view NeRF generation on the ABO dataset. Metrics are computed on renderings of resolution 128 × 128. HyP-NeRF significantly outperforms PixelNeRF [7] and CodeNeRF [3] on all of the metrics in both object categories.

1.1 Additional Architectural Details

We provide the network architecture in the main paper, Section 4. During training, we use Adam Optimizer, with a learning rate of $1e-3$ with $\beta_1 = 0.9$ and $\beta_2 = 0.99$, along with a lambda LR scheduler[3]. We use the PyTorch implementation InstantNGP[4] and provide the training, inference, and metric computation code in the supplementary.

1.2 Metrics and Additional Comparisons

To the best of our knowledge, we are the first work to perform single-view NeRF generation at a resolution of 512×512. Therefore, we set a benchmark on the ABO dataset against PixelNeRF in the main paper, Table 1. However, it is worth noting that PixelNeRF was originally trained at a resolution of 128×128. Therefore, we also compare with PixelNeRF on ABO at a resolution of 128×128 in the supplementary paper, Table 1 and Figure 1. To do so, we retrain PixelNeRF on 128×128 by downsampling the ground truth datapoints in ABO. Further, we show qualitative results in the main paper on SRN [6] against VisionNeRF [4], FE-NVS [2], CodeNeRF [3], and PixelNeRF at 128×128. In the supplementary paper, we show quantitative results in Table 2. Lastly, in Table 3, we show additional ablation on Denoise and Finetune (see main paper, Section 3.1-Step 2). In this table, we primarily evaluate the geometric consistency before and after the denoising step. As explained in the main paper, Section 4, we use Chamfer's Distance (CD↓) to compute the geometric consistency.

To compute CD, we first train the ground truth NeRFs by optimizing InstantNGP [5] on the multi-view ground truths. Next, we render meshes from InstantNGP's and HyP-NeRF's predicted NeRF

[*]Equal authors (order decided by a coin flip)
[†]@research.iiit.ac.in
[3]https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.LambdaLR.html
[4]https://github.com/ashawkey/torch-ngp

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

		Chairs			Cars		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
SRN	PixelNeRF [7]	23.72	0.90	0.128	23.17	0.89	0.146
	CodeNeRF [3]	23.39	0.87	0.166	22.73	0.89	0.128
	FE-NVS [2]	23.21	**0.92**	**0.077**	22.83	**0.91**	0.099
	VisionNeRF [4]	**24.48**	**0.92**	**0.077**	22.88	0.90	**0.084**
	HyP-NeRF	22.80	0.88	0.13	**23.48**	**0.91**	0.09
	w/o Denoise	21.02	0.87	0.14	21.30	0.88	0.111

Table 2: **Generalization**. Comparison of single-view NeRF generation on the SRN dataset. Metrics are computed on renderings of resolution 128 × 128. Results of all the models (except HyP-NeRF) are taken from the official papers. HyP-NeRF performs comparably with the existing baselines. Note, we do not incorporate the second step of HyP-NeRF, Denoise and Finetune (main paper, Section 3.1).

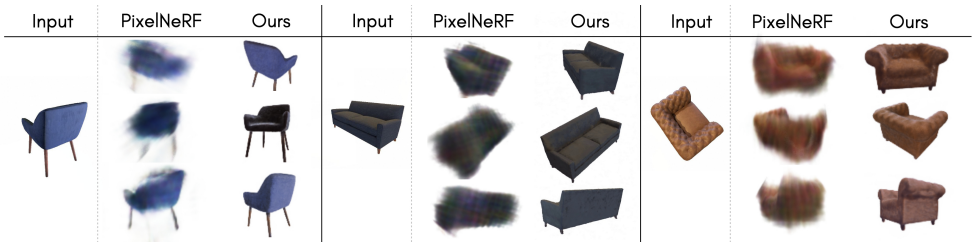

Figure 1: **Qualitative results of single-view NeRF generation** on ABO dataset at a resolution of 128×128. HyP-NeRF preserves fine details even at this low resolution. PixelNeRF improves in quality when compared to 512×512 (see main paper, Figure 4). However, it still struggles to model the fine texture and shape details in the ABO dataset and performs subpar to HyP-NeRF.

Figure 2: **Multiview test-time optimization on SRN Chairs**. HyP-NeRF can perform test-time optimization (see main paper, Section 3.1) with any number of views. In this qualitative result, we start with an instance that did not optimize well through a single pose (because of a challenging viewpoint) and show the improvement in quality of the generated NeRF as we increase the number of views (ie. coverage) for optimization. The header indicate the number of views used for the optimization. As shown, the difference in quality between five and ten poses is insignificant. The rightmost result shows drastic improvement in the render quality from one to three views showcasing the impact of pose on test-time optimization.

using torch-ngp's save_mesh() implementation[5]. From the rendered mesh, we sample 4096 points uniformly and compute CD between both the pointclouds. **We encourage the readers to view the supplementary video for the best experience of the qualitative results.**

1.3 Generalization

To ensure that HyP-NeRF can model novel NeRF instances unseen at the time of training, we rely on the conditional task of "single-view NeRF generation". In the main paper, we show experiments on ABO dataset at 512×512 resolution; in the supplementary paper, we make comparisons on a lower resolution of 128×128 on ABO against PixelNeRF in Table 1 and on SRN against the existing baselines in Table 2. As can be seen, we significantly outperform PixelNeRF on the ABO dataset and perform comparably with the existing baselines on the SRN dataset. **Note** that we do not employ the Denoise and Finetune step (see main paper, Section 3.1) in SRN. However, another reason for our low performance on SRN (when compared to ABO) is the difference in the views adopted in ABO and SRN. ABO renders the 3D structure from 91 viewpoints on the upper icosphere with varying

[5] https://github.com/ashawkey/torch-ngp/blob/main/nerf/utils.py

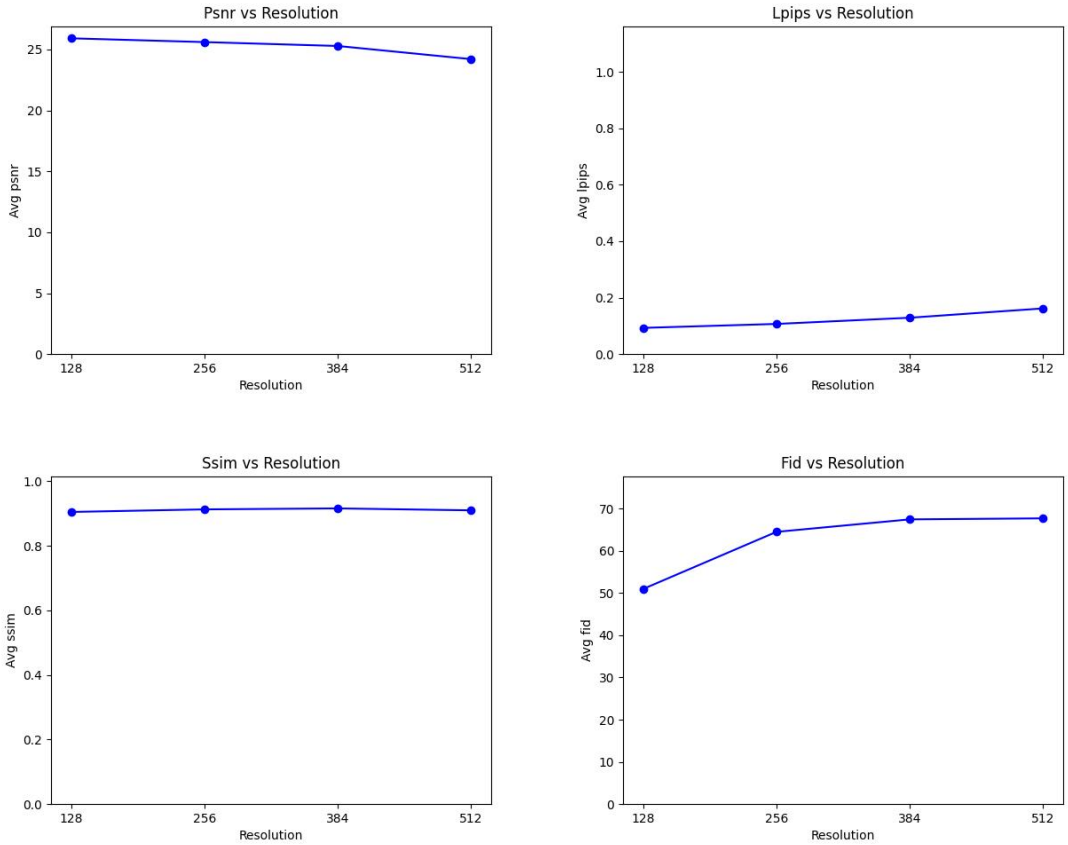

Figure 3: **Performance of HyP-NeRF on multiple resolutions**. As HyP-NeRF operates directly in the NeRF space, it can render the NeRFs in potentially any resolution. In this plot, we showcase HyP-NeRF's performance rendered in different resolutions. As can be seen, the quality does not degrade with the resolution and HyP-NeRF performs well consistently.

azimuth and elevation [1]. SRN, on the other hand, renders the upper along with the lower icosphere. This includes viewpoints from the absolute bottom and top parts of the object providing insufficient context for test-time optimization.

We observe that, in practice, our output quality improves significantly as we increase the number of viewpoints on SRN as shown qualitatively in Figure 2. This indicates that although HyP-NeRF has modeled this particular NeRF instance, it is hard to find the NeRF through single-view optimization suggesting the need for a more robust mechanism to map to HyP-NeRF's prior.

It is also worth noting that, PixelNeRF and VisionNeRF (trained on 16 NVIDIA A100 for 5 days) are designed specifically for the task single-view NeRF generation. Whereas, we aim to train a prior and use this conditional task to validate that our learned prior can model novel instances unseen at the time of training. Further, as can be seen in Table 1, HyP-NeRF significantly outperforms PixelNeRF on the challenging ABO dataset with high-fidelity structure and texture, indicating that HyP-NeRF is capable of modeling datasets made of fine textures and shapes as found in the real-world, that the existing work (PixelNeRF) struggle to train on.

2 Additional Ablations

Impact of Resolution on the Quality: As we operate directly in the NeRF space, we can essentially render the NeRFs in any resolution. In this ablation, we measure the quality of our renderings at different resolutions as shown in Figure 3. To generate the ground truth, we perform interarea downsampling on the ABO datapoints. As expected, HyP-NeRF generates high-quality NeRF in each resolution, and the quality does not degrade with the rendered resolution. As ABO consists

of rendering at a resolution of 512×512, we only make comparisons on the lower resolutions as bicubic upsampling on the ground truth would reduce the quality of the ground truth itself. However, to showcase our quality on higher resolution, we present our rendering at 1024×1024 in the supplementary video, timestamp 01:17.

Geometric Consistency on Denoising: As explained in the main paper, Section 3.1, we perform Denoise and Finetune by first projecting the NeRF into predefined multiview images, followed by performing image-level denoising frame-by-frame. As we only finetune an already multiview and geometrically consistent NeRF, we observed that the finetuning is robust to minor denoised image-level multiview inconsistencies. We showcase this qualitatively in the supplementary video (timestamps 3:05-3:30), in the main paper-Figure 4, and in the supplementary paper-Figure 5 and Figure 6. In this section, we quantitatively evaluate the geometric consistency using the CD metric as defined in Section 1.2. The results are presented in Table 3, and as expected, there is no degradation in the quality between HyP-NeRF's output before and after the Denoise and Finetune process, clearly showcasing that the geometric consistency is not affected even though we only rely on a simple frame-by-frame denoising.

HyP-NeRF	Chairs CD ↓
with D&F	0.0062
without D&F	0.0064

Table 3: **Denoise and Finetune (D&F) ablation (see main paper (Section 3.1 Step 2)**. We evaluate the geometric consistency using CD↓ metric defined in Section 1.2.

3 Qualitative Results

In this section, we show the qualitative results in higher resolution. Figure 4 presents the comparison between InstantNGP, trained on a single instance, against HyP-NeRF trained on thousands of NeRF instances, thereby compressing the instances to a single network (see the main paper, Section 4.2). Figure 5 and Figure 6 present qualitative results on inversion and highlight the difference before and after the Denoise and Finetune step (see main paper, Section 3.1).

Figure 4: **Qualitative comparison on Compression**. We compare against InstantNGP [5], which is trained for a specific instance. On the other hand, HyP-NeRF is trained on thousands of NeRF instances. Despite that, HyP-NeRF has learned to generate the NeRFs and essentially compress them almost losslessly. See the main paper, Section 4.2, for more details.

Figure 5: **Qualitative Results on Generalization**. We perform test-test optimization (see the main paper, Section 3.2) to generate NeRFs from a single input view. Our Denoise and Finetune step (see the main paper, Section 3.1) significantly improves the texture and the edges by making it smooth and even.

Figure 6: **Qualitative Results on Generalization**. We perform test-test optimization (see the main paper, Section 3.2) to generate NeRFs from a single input view. Denoise and Finetune (see the main paper, Section 3.1) improves the quality of the outputs, for example, the legs are clearly more evened out and noiseless in the bottom example. The difference is, however, less drastic in the top example.

References

[1] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, Matthieu Guillaumin, and Jitendra Malik. Abo: Dataset and benchmarks for real-world 3d object understanding. *CVPR*, 2022.

[2] Pengsheng Guo, Miguel Angel Bautista, Alex Colburn, Liang Yang, Daniel Ulbricht, Joshua M Susskind, and Qi Shan. Fast and explicit neural view synthesis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3791–3800, 2022.

[3] Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12949–12958, 2021.

[4] Kai-En Lin, Lin Yen-Chen, Wei-Sheng Lai, Tsung-Yi Lin, Yi-Chang Shih, and Ravi Ramamoorthi. Vision transformer for nerf-based view synthesis from a single input image. In *WACV*, 2023.

[5] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022.

[6] Vincent Sitzmann, Michael Zollhoefer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[7] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelNeRF: Neural radiance fields from one or few images. In *CVPR*, 2021.