Generalizable Patch-Based Neural Rendering Supplementary Material

Mohammed Suhail[1], Carlos Esteves[4], Leonid Sigal[1,2,3], and Ameesh Makadia[4]

[1] University of British Columbia {suhail33,lsigal}@cs.ubc.ca
[2] Vector Institute for AI
[3] Canada CIFAR AI Chair
[4] Google {machc,makadia}@google.com

A Additional Experiments and Results

A.1 Fine-tuning

While our model is focused at generalizing to unseen scenes, for a thorough comparison against previous methods, we follow the protocol from IBRNet [5] and fine-tune our model (for setting 1) on each of the RFF test scenes for 10k iterations. We report the average metrics across all scenes in Table A.1.

Method	PSNR	SSIM	LPIPS
IBRNet [5]	26.73	0.851	0.175
GeoNeRF [2]	26.58	0.856	0.162
Ours	27.66	0.924	0.138

Table A.1. Fine-tuning results on the RFF dataset, setting 1. Our approach not only improves over the baselines on unseen scenes with no re-training as shown in the main paper, but also when allowed to fine-tune for a few iterations on new scenes.

A.2 Number of Reference Views

We train our model with $3, 5, 7, 10$ and 12 reference images to investigate the effect of number of reference views available to view synthesis. The models are trained on forward-facing scenes from LLFF [3] and IBRNet [5]. We summarize the average performance of each variant on the real-forward-facing dataset in Table A.2. Our model benefits from having access to a large number (up to 10) of reference images.

Number of Reference Views	Real-Forward-Facing		
	PSNR	SSIM	LPIPS
3	22.36	0.800	0.286
5	24.33	0.850	0.216
7	25.05	0.860	0.195
10	25.72	0.880	0.175
12	25.69	0.881	0.178

Table A.2. Effect of varying the number of reference view on view-synthesis.

A.3 RGB Prediction

To synthesize a novel view, our model predicts the weights of a linear combination of reference image pixel colors. A common alternative is to combine learned visual features, followed by a learned mapping to color values [4]. To substantiate our argument of better generalization from combining colors instead of features, we train a variant of our model where the output color is predicted from the aggregated features as

$$\mathtt{c} = \mathtt{MLP}\left(\sum_{k=1}^{K} \beta_k f_3^k\right). \tag{1}$$

We evaluate this approach in setting 1 and show that our model, which combines pixels, is indeed superior. Table A.3 shows the results.

Interpolation Method	Real-Forward-Facing		
	PSNR	SSIM	LPIPS
Features	25.08	0.86	0.199
Colors (ours)	25.72	0.88	0.175

Table A.3. Comparison of average performance when using feature versus color interpolation for view-synthesis. Results show that combining the colors of reference views generalizes better than combining visual features.

A.4 DTU to RFF Generalization

We present results on generalization to scenes in the real-forward-facing (RFF) dataset for a model trained only on DTU in Table A.4. While MVSNeRF [1] has a better PSNR and LPIPS performance our method achieves better SSIM scores on average across all scenes in RFF.

Method	Real-Forward-Facing		
	PSNR	SSIM	LPIPS
MVSNeRF	21.93	0.795	0.252
Ours	20.69	0.808	0.281

Table A.4. Generalization results on RFF for a model trained on DTU.

A.5 Timing Statistics

On average, our model trains at around 3.2 steps per second on 32 TPUs with a batch size of 4096. A prior transformer-based neural rendering work, LFNR [4], is slightly faster at 4.2 steps per second on the same hardware with the same batch size. Rendering one image with our model takes around 15 seconds whereas LFNR takes around 10 seconds on the same hardware. While LFNR takes around 16 hours to train, it can only be trained on single scene. Thus, to render the novel views for the 8 scenes in the RFF dataset, it would take at least 128 hours of training plus the inference time. Our model just trains for approximately 24 hours and can be used for inference directly on all the scenes, albeit with a small drop in rendering quality.

B Quatlitative Results

B.1 DTU Comparison

We compare renderings on the DTU test set against MSVNeRF in Fig. B.1. Compared to MVSNeRF, our model produces renderings with sharper boundaries and textures.

References

1. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 14124–14133 (2021) 2
2. Johari, M.M., Lepoittevin, Y., Fleuret, F.: Geonerf: Generalizing nerf with geometry priors. arXiv preprint arXiv:2111.13539 (2021) 1
3. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (TOG) **38**(4), 1–14 (2019) 1
4. Suhail, M., Esteves, C., Sigal, L., Makadia, A.: Light field neural rendering. CoRR (2021), http://arxiv.org/abs/2112.09687v1 2, 3
5. Wang, Q., Wang, Z., Genova, K., Srinivasan, P.P., Zhou, H., Barron, J.T., Martin-Brualla, R., Snavely, N., Funkhouser, T.: Ibrnet: Learning multi-view image-based rendering. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4690–4699 (2021) 1

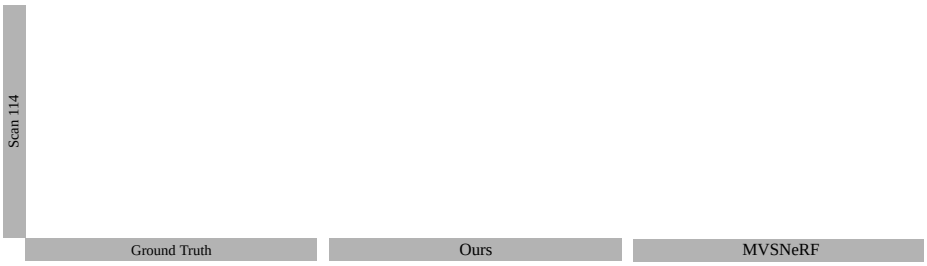

Fig. B.1. Qualitative Results on DTU.