Flow-NeRF: Joint Learning of Geometry, Poses, and Dense Flow within Unified Neural Representations

Supplementary Material

1. More details of the optimization losses

Following Nope-NeRF [1], we enforce a depth loss between the rendered depth $\hat{D}(\mathbf{p})$ and the undistorted pre-computed pseudo ground truth depth $D^*(\mathbf{p})$ as follows:

$$L_{depth} = \frac{1}{N} \sum_{\mathbf{p} \in \Omega_N} ||D^*(\mathbf{p}) - \hat{D}(\mathbf{p})||_1 \tag{1}$$

Additionally, we consider a point cloud loss to constrain the relative poses between frame i and frame j:

$$L_{pc} = \sum_{(i,j)} l_{cd}(P_j^*, T_{ji} P_i^*), \tag{2}$$

where P_j^* and P_i^* denote the point clouds of frame j and frame i computed from their undistorted depths D_j^* and D_i^*, respectively; T_{ji} represents the relative poses of the two frames; and l_{cd} denotes the Chamfer Distance between the two point clouds. Additionally, we introduce a photometric warping loss for the entire image, given by the projection of point cloud P_i^* onto frame j:

$$L_{rgb\_s} = \sum_{(i,j)} ||I_i\langle K_i P_i^* \rangle - I_j\langle K_j T_j T_i^{-1} P_i^* \rangle||_1, \tag{3}$$

where $\langle\rangle$ represents the bilinear interpolation operation on the image to acquire the corresponding color. Finally, our loss function is defined as:

$$L_o = L_{rgb} + \lambda_1 L_{flow} + \lambda_2 L_{depth} + \lambda_3 L_{pc} + \lambda_4 L_{rgb\_s}. \tag{4}$$

The geometry and the flow branches of Flow-NeRF are jointly optimized using the overall loss L_o.

2. Verification of feature complementarity between the geometry and the flow branches

Intuitively, given the shared points sampling strategy described in Sec. 3.2 of the main paper, the point features extracted from the canonical space G of the flow branch should have significant overlap with the point features from the geometry branch, as both features represent the same physical scene. We validate this insight by rendering RGB images from the canonical feature extractor $F_{\theta 2}$, resulting in a 4-channel tensor output. The first three channels correspond to RGB values, while the last channel predicts the alpha value σ_2. To visualize what can be learned from the canonical features, we enforce an additional photometric loss between the rendered flow RGB $\hat{\mathbf{C}}_{flow}$ and the ground truth RGB image $\mathbf{C}$ as:

$$L_{rgb\_flow} = \frac{1}{N} \sum_{\mathbf{p} \in \Omega_N} ||\hat{\mathbf{C}}_{flow}(\mathbf{p}) - \mathbf{C}(\mathbf{p})||_1, \tag{5}$$

where Ω is the set of all N pixels sampled from the frame. For the RGB prediction from the geometry branch, the loss function is defined as:

$$L_{rgb} = \frac{1}{N} \sum_{\mathbf{p} \in \Omega_N} ||\hat{\mathbf{C}}(\mathbf{p}) - \mathbf{C}(\mathbf{p})||_1, \tag{6}$$

In Fig. 1, we visualize the results at iteration 20000, an early stage of the training process. It can be observed that within the red box, the RGB rendered from the flow branch and the flow map prediction show clearer details of the desk corner and the chair contour. In contrast, while exhibiting more visually realistic colors, the RGB image of the geometry branch lacks the accuracy of the geometry depicted in the flow RGB. This simple verification experiment demonstrates that the flow branch captures more structured and finer details than the geometry branch. This observation inspires us to develop the flow feature message passing strategy for the geometry branch, which proves effective in improving both the novel-view synthesis and depth prediction.

3. Dataset details

For **Tanks and Temples**, following Nope-NeRF [1], we evaluate novel view synthesis across 8 scenes. We sample 7 images from each 8-frame clip as training views and evaluate the novel view synthesis results on all other views. For the Family scene, we follow the work of Nope-NeRF [1] by sampling every other view and evaluating the novel view synthesis results on the remaining half.

For **ScanNet**, following Nope-NeRF [1], we sample 80-100 images from 4 scenes. We sample 7 training views from each 8-frame clip and evaluate both the novel view synthesis results and the depth estimation results. For data preprocessing, we use the ImageMagick [6] toolbox to downsample all images into half resolution. In addition, for Scene 0079_00, we crop the dark borders by 10 pixels before preprocessing. The details of the selected ScanNet sequences are shown in Tab. 1.

For **Sintel**, which consists of several scenes with ground truth flow available between consecutive frames, we train on 2 scenes: mountain1 and sleeping2. Each scene contains a total of 50 images; we sample every other frame to create

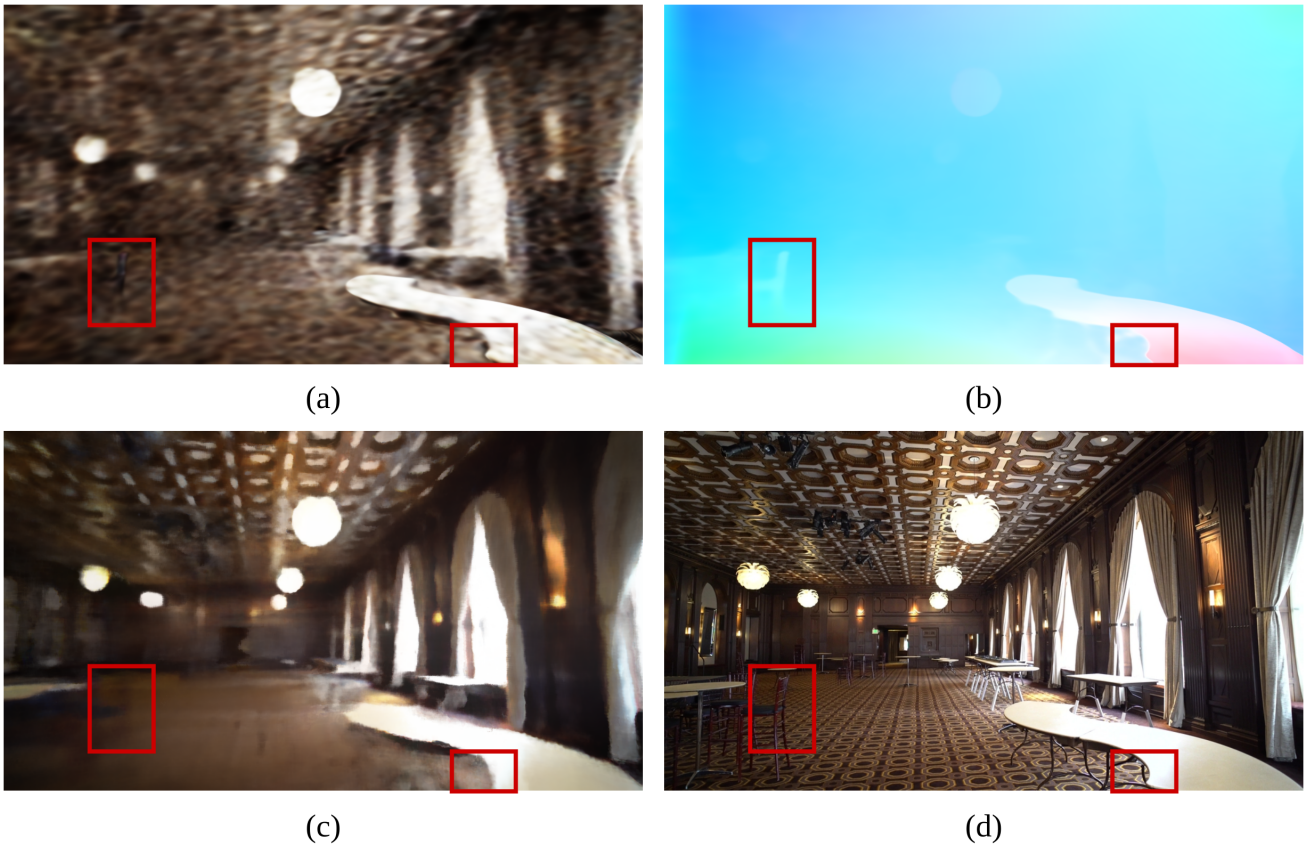


Figure 1. **Complementary features of geometry and flow branch outputs at training iteration 20000:** (a) RGB image rendered from the canonical feature $F_{\theta 2}$; (b) flow prediction; (c) RGB image rendered from NeRF MLP $F_{\theta 1}$; (d) RGB ground truth.

	Scenes	Type	Seq. length	Frame ID	Max. rotation (deg)
ScanNet	0079_00	indoor	90	331-420	54.4
	0418_00	indoor	80	2671-2750	27.5
	0301_00	indoor	100	831-930	43.7
	0431_00	indoor	100	591-690	45.8

Table 1. **Details of the selected ScanNet sequences.**

the training set, leaving the other 25 frames as novel view test frames. For the flow evaluation, we compare our novel-view flow results with the RAFT flow prediction results using the average end-point error (EPE) at non-occluded pixels across different frame intervals, as shown in Fig. 9 of the main paper. RAFT-D in Fig. 9 refers to directly inferring distant frame flows with RAFT, which yields better flow results than RAFT-C (where consecutive RAFT flow predictions are chained to formulate long-range flow). We obtain the ground truth optical flow for long-range flows by chaining consecutive ground truth flow along with their occlusion masks. We calculate the end-point error (epe) as follows:

$$\text{epe} = \frac{1}{M} \sum_{\mathbf{p}_{gt} \in \Omega_M} ||\hat{\mathbf{p}}_{est} - \mathbf{p}_{gt}||_1, \tag{7}$$

where $\hat{\mathbf{p}}_{est}$ and $\mathbf{p}_{gt}$ denote the estimated and the ground truth flow vector, respectively, and Ω_M is the set of all non-occluded pixels.

4. Implementation details

Network structure and learning rate: For the bijective network, following omnimotion [10], we use a simplified Real-NVP [3] but with much fewer layers for the trade-off of training speed. We use 4 affine coupling layers for the network, and each layer contains three 128-dimensional MLPs. We set the initial learning rate of both the pose and the $F_{\theta 2}$ MLP to be 0.0005, the bijective network to be 0.0001, the canonical feature MLP to be 0.0003 and the latent space embedding network to be 0.001. We employ an auto-scheduler to decrease the learning rate for both the network and the pose until the training PSNR does not increase for more than 1000 epochs. We set $\lambda_1 = 0.05$, $\lambda_2 = 0.04$, $\lambda_3 = 1$ and $\lambda_4 = 1$ for all the loss terms.

Point sampling: For simplicity, we discard the hierarchical sampling strategy in the original NeRF, but apply a uniform sampling with perturbation during training to sample m distance values. In all our experiments, we set the near and far bound of $z_n = 0.01$, $z_f = 10$, and $m = 128$.

Pose module: The pose-conditioned latent embedding network is a 256-dimensional 3 layer MLP with Gaber layer, which takes a 6DOF pose vector $[r_1, r_2, r_3, t_1, t_2, t_3]$ as input and outputs a 128-dimensional feature. The latent pose feature in the flow branch serves as an identifier of differ-

Method	PSNR ↑	SSIM ↑	Abs Rel ↓	RMSE ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
LocalRF (Meuleman, 2023)	31.25	0.83	11.148	1.498	0.422	0.564	0.850
CF-3DGS (Yu, 2023)	28.51	0.80	12.360	1.014	0.617	0.819	0.875
Ours	**32.55**	**0.85**	**0.047**	**0.151**	**0.982**	**0.993**	**0.999**

Table 2. **Comparison against SOTA methods on novel-view synthesis and depth estimation across all 4 scenes on the ScanNet dataset.**

Method	PSNR ↑	SSIM ↑	LPIPS ↓
DBARF (Chen, 2023)	22.97	0.73	0.30
CoPoNeRF (Hong, 2024)	21.60	0.67	0.27
Ours	**28.73**	**0.82**	0.29

Table 3. **Comparison against generalizable NeRF methods on novel-view synthesis across all 8 scenes on the Tanks and Temple dataset.**

ent frames. The driving force for pose learning comes from the complementary of RGB, flow, depth, and point cloud matching loss.

0079_00	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
BARF [7]	0.208	0.165	0.588	0.263	0.639	0.896	0.983
NeRFmm [11]	0.494	1.049	1.419	0.534	0.378	0.567	0.765
SC-NeRF [9]	0.360	0.450	0.902	0.396	0.407	0.730	0.908
Nope-NeRF [1]	0.099	0.047	0.335	0.128	0.904	0.995	1.000
Ours	**0.040**	**0.006**	**0.106**	**0.057**	**0.993**	**1.000**	**1.000**

Table 4. **Depth map evaluation on ScanNet 0079_00**.

0418_00	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
BARF [7]	0.718	1.715	1.563	0.630	0.205	0.569	0.769
NeRFmm [11]	0.907	3.650	2.176	0.769	0.240	0.456	0.621
SC-NeRF [9]	0.319	0.441	0.898	0.377	0.456	0.792	0.930
Nope-NeRF [1]	0.152	0.137	0.645	0.185	0.738	0.998	0.997
Ours	**0.034**	**0.010**	**0.118**	**0.070**	**0.984**	**0.995**	**0.998**

Table 5. **Depth map evaluation on ScanNet 0418_00**.

5. Comparison with other SOTA methods

We compare the novel-view synthesis and depth estimation across all 4 scenes on ScanNet, as shown in Tab. 2. Our method is clearly better than LocalRF [8] and CF-3DGS [4]. Since both LocalRF and CF-3DGS employ an incremental manner, they lack global bundle adjustment to correct the depth scale discrepancy among each sub-model/local Gaussian, leading to inconsistent geometry.

We also compare the novel-view synthesis against generalizable NeRF methods DBARF [2] and CoPoNeRF [5] across all 8 scenes on the Tanks and Temple dataset, as shown in Tab. 3. For CoPoNeRF designed for two-view geometry, we select the $i-1$ and $i+1$ frame as the context frames, and query the test frame i in the middle. We largely outperform generalizable NeRF although they have been pretrained on large-scale datasets.

6. Drastic camera motion scenes and pose visualization

We test our method on several scenes of the LLFF dataset which contains irregular and fast camera motion. The visualization of the rendered flow and camera pose can be found in Fig. 4. The visualization results show that our method can render plausible flow and estimate camera poses even under drastic camera motion. We also provide the visualization comparison (see Fig. 5) on 2 challenging scenes, from which the Museum scene has the maximum rotation of 76.2, and the 0079_00 has the maximum rotation of 54.4. Our method performs better than Nope-NeRF when the camera rotation is large.

7. Additional results

We provide several additional visualization results for novel-view synthesis, novel-view depth, and long-range novel-view flow predictions across several scenes, as shown in Fig. 2 and Fig. 3. The qualitative results indicate consistent predictions among the novel-view images, novel-view depths, and novel-view flows, demonstrating that all our optimization objectives are indeed coupled. We also present a further visual comparison of the novel-view images and depth predictions on the Tanks and Temples dataset, as shown in Fig. 6. Compared to the state-of-art method Nope-NeRF [1], our method produces more photo-realistic novel-view images and significantly smoother depth maps with fewer artifacts, clearly validating the effectiveness of the proposed flow-enhanced novel-view synthesis and flow-enhanced geometry.

Besides, we provide the per-scene depth prediction results on the ScanNet dataset. The qualitative results are displayed in Fig. 7, while the quantitative results are shown in Tab. 4, 5, 7 and 8. Both the qualitative and quantitative results demonstrate that our method predicts depth maps significantly better than all other methods.

We also present additional novel-view synthesis and pose estimation results on the Sintel dataset (see Tab. 9 and Tab. 10). Note that the Sintel dataset contains high-resolution images and large camera motion, and our method significantly outperforms Nope-NeRF in both tasks. We also provide the visualization of the depth prediction and novel view synthesis on Sintel (see Fig. 8). Our method can generate depth maps with sharper details and produce more photo-realistic novel-view images.

8. Limitations and future work

Given accurate pixel-wise and frame-wise correspondence prediction, one can easily compute the relative poses for frame pairs using either an analytical pose-solver or performing a motion-only bundle adjustment. We have not yet explored the potential of the predicted novel-view flow in this promising direction.

Rendered RGB Rendered depth Forward/backward long-range novel-view flow

Figure 2. **Additional visualization results on long-range frame flow estimation on the Tanks and Temples dataset**. T+8 and t-8 denote forward and backward flow with a frame interval of 8, respectively.

scenes		Ours			Nope-NeRF [1]			BARF [7]			NeRFmm [11]			SC-NeRF [9]		
		$RPE_t \downarrow$	$RPE_r \downarrow$	ATE$\downarrow$	RPE_t	RPE_r	ATE	RPE_t	RPE_r	ATE	RPE_t	RPE_r	ATE	RPE_t	RPE_r	ATE
Tanks and Temple	Church	0.035	0.093	0.008	0.034	0.008	0.008	0.114	0.038	0.052	0.626	0.127	0.065	0.836	0.187	0.108
	Barn	0.076	0.162	0.008	0.046	0.032	0.004	0.314	0.265	0.050	1.629	0.494	0.159	1.317	0.429	0.157
	Museum	0.125	0.187	0.007	0.207	0.202	0.020	3.442	1.128	0.263	4.134	1.051	0.346	8.339	1.491	0.316
	Family	0.173	0.068	0.009	0.047	0.015	0.001	1.371	0.591	0.115	2.743	0.537	0.120	1.171	0.499	0.142
	Horse	0.181	0.069	0.009	0.179	0.017	0.003	1.333	0.394	0.014	1.349	0.434	0.018	1.366	0.438	0.019
	Ballroom	0.115	0.101	0.008	0.041	0.018	0.002	0.531	0.228	0.018	0.449	0.177	0.031	0.328	0.146	0.012
	Francis	0.177	0.296	0.023	0.057	0.009	0.005	1.321	0.558	0.082	1.647	0.618	0.207	1.233	0.483	0.192
	Ignatius	0.081	0.049	0.006	0.026	0.005	0.002	0.736	0.324	0.029	1.302	0.379	0.041	0.533	0.240	0.085
	mean	0.120	0.128	0.010	0.080	0.038	0.006	1.046	0.441	0.078	1.735	0.477	0.123	1.890	0.489	0.129

Table 6. **Camera pose comparison on the Tanks and Temples dataset**.

0301_00	Abs Rel $\downarrow$	Sq Rel $\downarrow$	RMSE $\downarrow$	RMSE log $\downarrow$	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$
BARF [7]	0.179	0.146	0.502	0.268	0.736	0.883	0.938
NeRFmm [11]	0.444	0.830	1.239	0.481	0.397	0.680	0.845
SC-NeRF [9]	0.383	0.378	0.810	0.452	0.360	0.663	0.846
Nope-NeRF [1]	0.185	0.252	0.711	0.233	0.792	0.918	0.958
Ours	**0.036**	**0.006**	**0.127**	**0.053**	**0.991**	**1.000**	**1.000**

Table 7. **Depth map evaluation on ScanNet 0301_00**.

0431_00	Abs Rel $\downarrow$	Sq Rel $\downarrow$	RMSE $\downarrow$	RMSE log $\downarrow$	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$
BARF [7]	0.398	0.710	1.307	0.444	0.381	0.655	0.847
NeRFmm [11]	0.514	1.354	1.855	0.562	0.250	0.539	0.742
SC-NeRF [9]	0.608	1.300	1.706	0.677	0.225	0.446	0.645
Nope-NeRF [1]	0.127	0.111	0.579	0.160	0.877	0.978	0.994
Ours	**0.078**	**0.028**	**0.251**	**0.107**	**0.960**	**0.978**	**0.999**

Table 8. **Depth map evaluation on ScanNet 0431_00**.

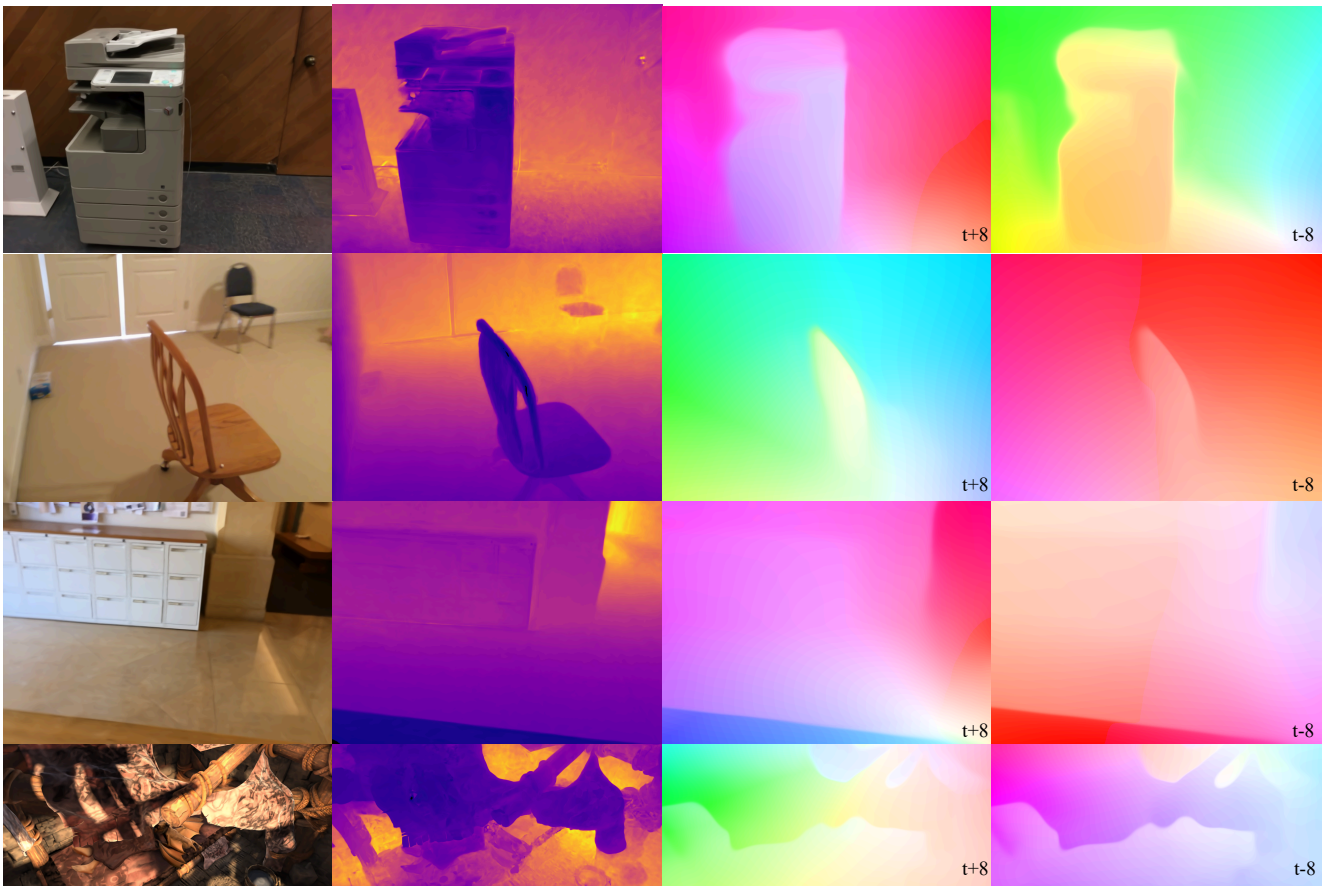


Figure 3. **Visualization on long-range frame flow estimation on the ScanNet and Sintel dataset**. T+8 and t-8 denote forward and backward flow with a frame interval of 8, respectively.

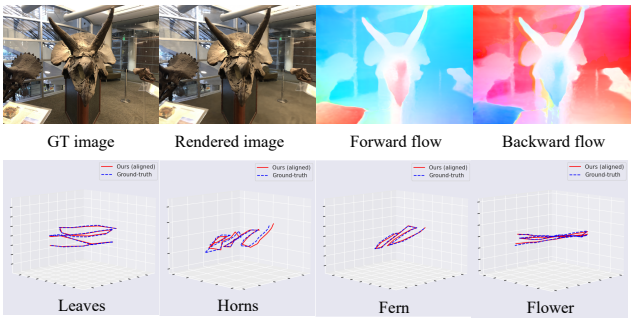


Figure 4. **Visualization results on the LLFF dataset.** Our method can handle large and fast camera motion, and render out plausible flow.

Method	mountain1			sleeping2		
	PSNR↑	SSIM↑	LPIPS↓	PSNR	SSIM	LPIPS
Nope-NeRF [1]	27.24	0.86	0.33	27.93	0.79	0.40
Ours	**31.24**	**0.93**	**0.24**	**31.19**	**0.88**	**0.32**

Table 9. **Novel view synthesis comparison on Sintel**.

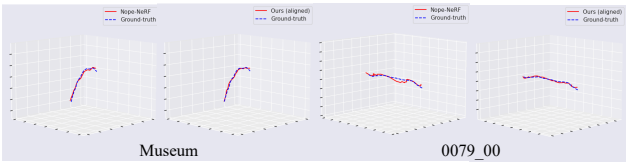


Figure 5. **Visualization of camera pose comparson against Nope-NeRF on the Museum scene and 0079_00 scene.** Our method can estimate more accurate camera poses when large camera rotation exists. Better viewed when zoom in.

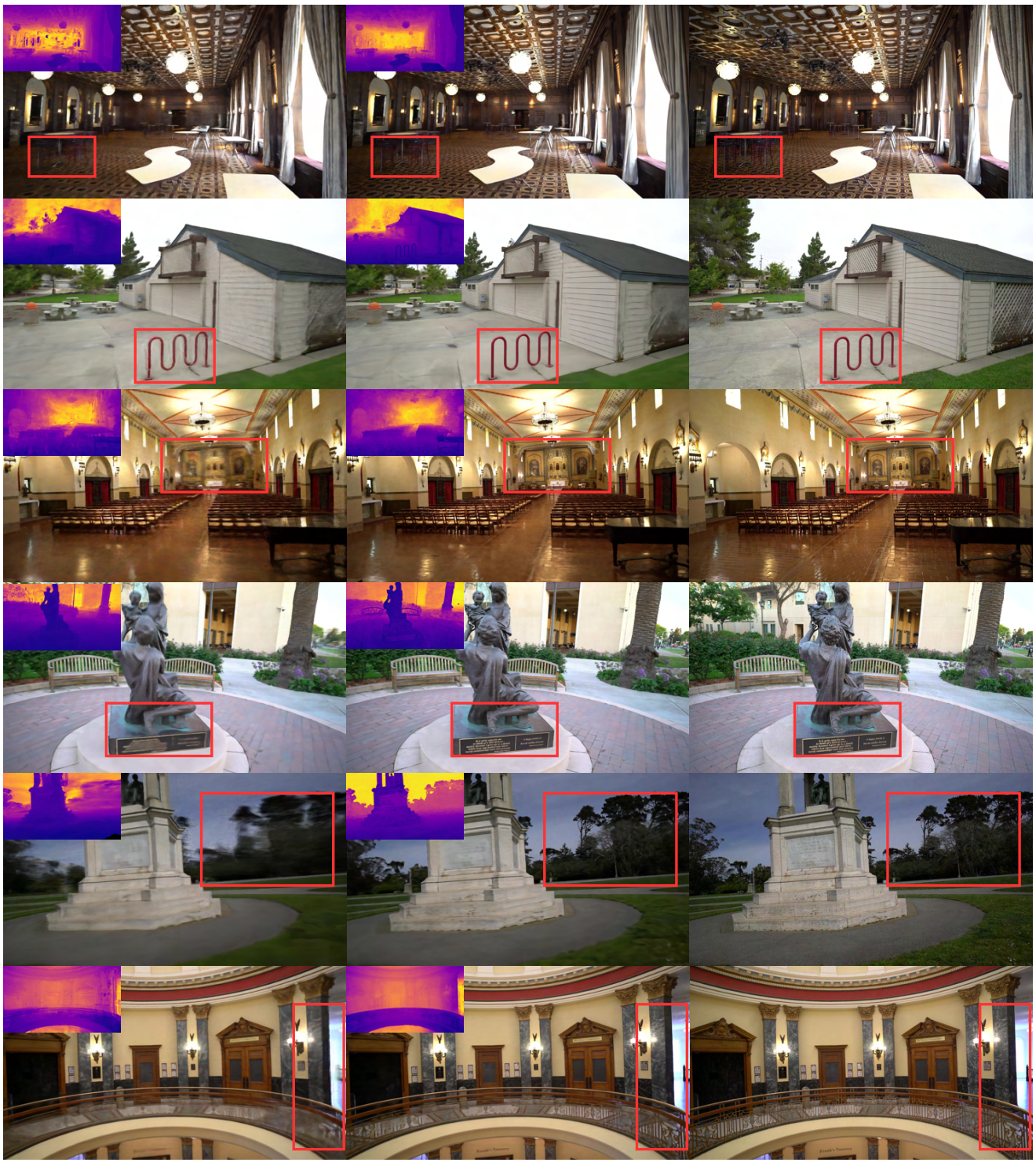
Method	mountain1			sleeping2		
	RPE_t ↓	RPE_r ↓	ATE↓	RPE_t	RPE_r	ATE
Nope-NeRF [1]	1895	0.237	21.57	3.616	0.574	0.005
Ours	**1332**	0.248	**3.501**	**3.566**	**0.537**	**0.002**

Table 10. **Camera pose comparison on Sintel**.

References

[1] Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. In *CVPR*, 2023. 1, 3, 4, 5, 6, 7, 8

[2] Yu Chen and Gim Hee Lee. Dbarf: Deep bundle-adjusting



Nope-NeRF Ours Ground truth

Figure 6. **Qualitative novel-view synthesis and depth estimation results on Tanks and Temples**. Compared with Nope-NeRF [1], our method can produce more photo-realistic novel-view synthesis results, and yield smoother depth maps while preserving more structure details. Better viewed when zoomed in.

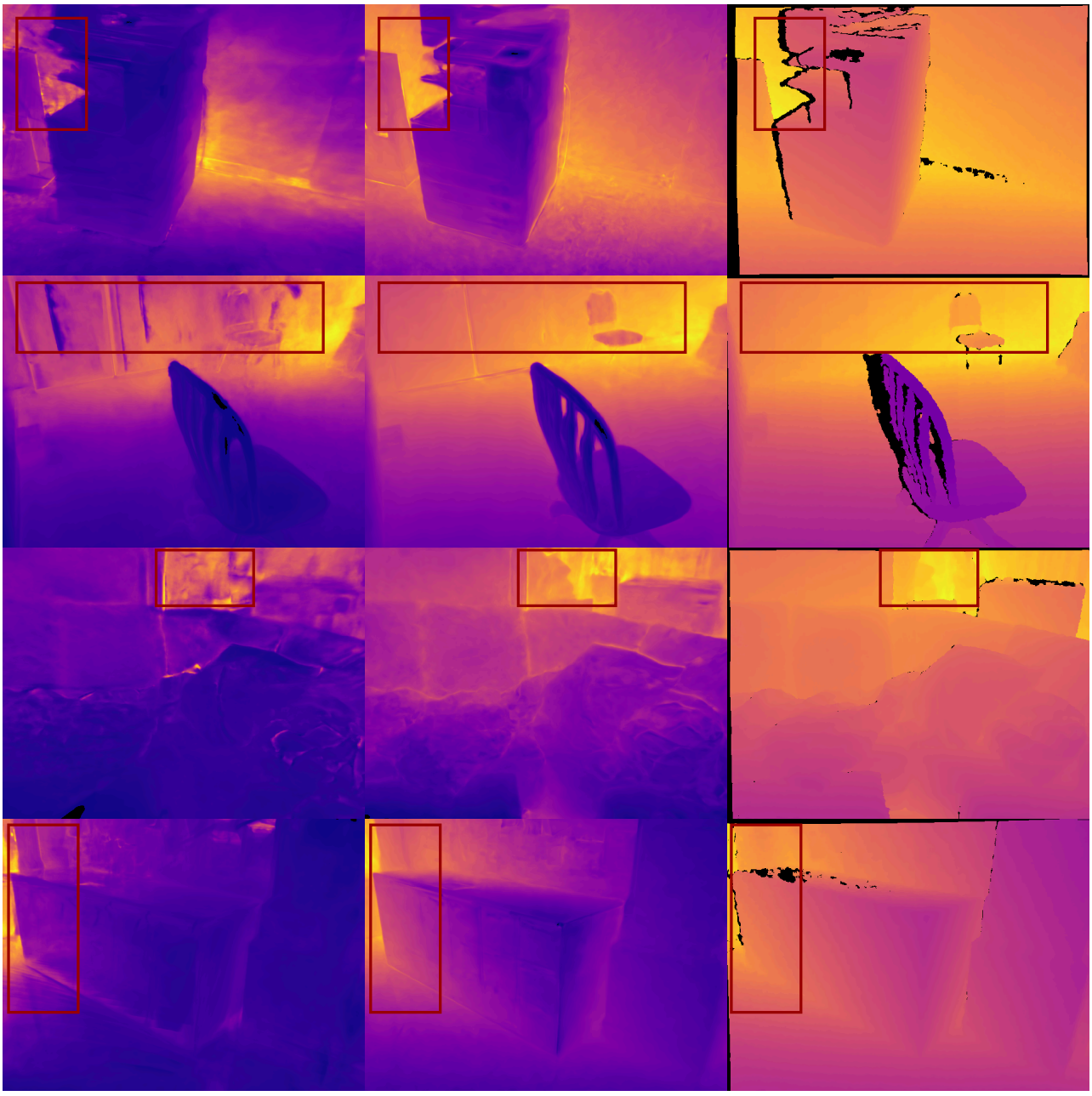


Figure 7. **Qualitative depth prediction comparison on ScanNet**. Compared with Nope-NeRF [1], our method predicts smoother novel-view depth with much fewer artifacts and preserves more structure details.

generalizable neural radiance fields. In *CVPR*, 2023. 3

[3] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016. 2

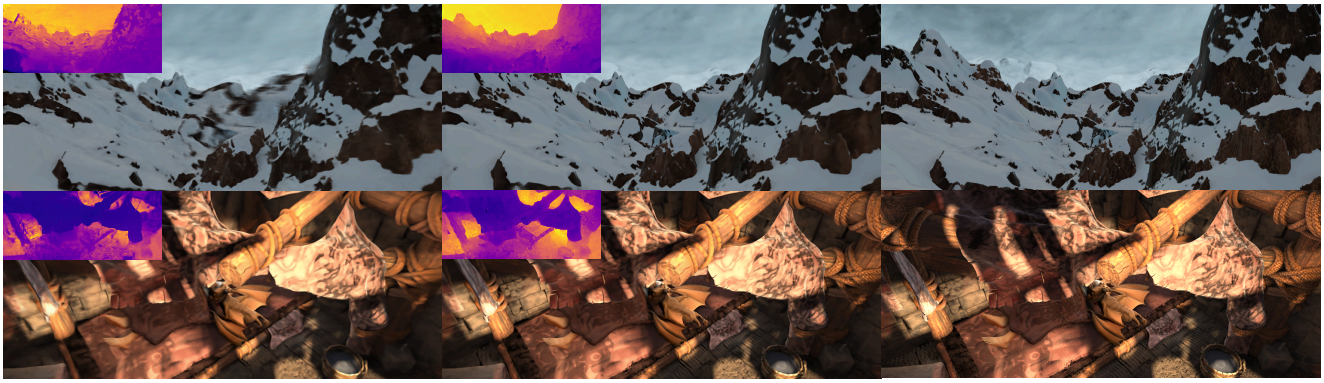
[4] Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A Efros, and Xiaolong Wang. Colmap-free 3d gaussian splatting. *arXiv preprint arXiv:2312.07504*, 2023. 3

[5] Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jiaolong Yang, Seungryong Kim, and Chong Luo. Unifying correspondence, pose and nerf for pose-free novel view synthesis from stereo pairs. *arXiv preprint arXiv:2312.07246*, 2023. 3

[6] ImageMagick Studio LLC. Imagemagick. 1

[7] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *ICCV*, 2021. 3, 4

[8] Andreas Meuleman, Yu-Lun Liu, Chen Gao, Jia-Bin Huang, Changil Kim, Min H Kim, and Johannes Kopf. Progressively optimized local radiance fields for robust view synthesis. In



Nope-NeRF Ours Ground truth

Figure 8. **Novel view synthesis and depth prediction comparison on Sintel**. Compared with Nope-NeRF [1], our method produces more photo-realistic novel-view images and depth maps with sharper details.

CVPR, 2023. 3

[9] Liang Song, Guangming Wang, Jiuming Liu, Zhenyang Fu, Yanzi Miao, et al. Sc-nerf: Self-correcting neural radiance field with sparse views. *arXiv preprint arXiv:2309.05028*, 2023. 3, 4

[10] Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. Tracking everything everywhere all at once. In *ICCV*, 2023. 2

[11] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 3, 4