Supplementary Materials – URS-NeRF: Unordered Rolling Shutter Bundle Adjustment for Neural Radiance Fields

Bo Xu[1,2], Ziao Liu[2], Mengqi Guo[1], Jiancheng Li[2], and Gim Hee Lee[1]

[1] National University of Singapore
[2] Wuhan University
https://boxulibrary.github.io/projects/URS-NeRF/

A Implementation and Training Details

We implement our method in PyTorch, running a total of 25K steps on a computer with Intel i7-9750H@2.6GHz CPU and NVIDIA RTX 4090 GPU. The coarse stage comprises 10K iterations, followed by the fine stage with 15K iterations. The Adam optimizer [2] is used to estimate the weights of the network, embedding pose parameters, and velocity parameters. For Tri-Mip $\mathcal{M}$, the learning rate is set to 2×10^{-3}, while it is 2×10^{-2} for encoding params. We follow a learning rate reduction schedule, decreasing it by $0.6\times$ at $\frac{1}{2}$, $\frac{3}{4}$, $\frac{5}{6}$, and $\frac{9}{10}$ of the total steps, consistent with [4]. The learning rates for camera pose and velocity are set to 2×10^{-3} and 2×10^{-4}, respectively. We reduce the learning rates by $0.6\times$ at $\frac{1}{12}$, $\frac{1}{6}$, $\frac{1}{4}$, and $\frac{1}{3}$ of the total steps. The total training iterations for NeRF [8] and BARF [7] are 200K. The pose corresponding to the first row of each image is assumed as the pose of the frame. The pose accuracy is evaluated by the tool evo [3]. Since DiffSfM [9] cannot synthesize novel images, we first apply the method to restore global shutter images and then use them as input to train Tri-Mip-BA.

B Details on Selected datasets

We use the synthetic datasets WHU-RS [1], and real datasets ZJU-RS [5] to verify the effectiveness of our method. We conduct experiments using 6 sequences from the WHU-RS dataset, comprising two scenarios with each scenario including fast, medium, and slow sequences. To analyze the performance of our method under different camera motion speeds, we select similar scenes and train the model using approximately 100 images for each sequence. For the ZJU-RS dataset, we select 6 sequences (D0, D2, D3, D8, C5, C11) out of the 23 available for reconstruction, all of which are captured using smartphones equipped with rolling shutter cameras. For each sequence, we select 70-100 images for reconstruction. In the main paper, we analyze the performance of different methods on D0 and D2 sequences. In the supplementary materials, more experimental results are reported.

Table S1: Average training and querying time consumption of Traj1-fast scene of WHU-RS dataset in seconds.

	NeRF	BARF	DiffSfM	Tri-MipRF	Tri-MipRF-BA	USB-NeRF-RE	URS-NeRF
Training Time	32314.32	43174.51	1014.29 (+22859.34)	543.29	1010.88	2092.35	1407.25
Querying Time	9.25	9.31	1.04	0.99	0.98	1.30	0.98

Table S2: Quantitative comparisons on the synthetic datasets for novel view synthesis on the WHU-RS dataset. For the fast, medium, and slow modes of the WHU-RS dataset, the average values of each metric are computed from two scenes. For each metric, the best is in **bold** for the unordered datasets and blue for the sequence video datasets.

		WHU-RS-Fast			WHU-RS-Medium			WHU-RS-Slow		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Un-view	NeRF	18.13	0.46	0.72	18.57	0.48	0.72	18.54	0.48	0.71
	BARF	16.37	0.49	0.64	17.54	0.49	0.59	15.34	0.46	0.64
	DiffSfM	25.07	0.80	0.19	27.08	0.85	0.12	27.78	0.87	0.11
	Tri-MipRF	16.35	0.47	0.61	16.80	0.49	0.57	16.94	0.50	0.58
	Tri-MipRF-BA	24.09	0.78	0.19	24.24	0.77	0.15	26.10	0.84	0.10
	USB-NeRF-RE	16.64	0.49	0.61	18.76	0.58	0.51	20.56	0.64	0.43
	URS-NeRF	**27.27**	**0.84**	**0.11**	**28.48**	**0.87**	**0.09**	**29.02**	**0.88**	**0.09**
Seq-view	USB-NeRF-RE	28.93	0.86	0.13	29.61	0.88	0.10	29.85	0.89	0.10
	URS-NeRF	27.56	0.85	0.15	28.82	0.87	0.11	29.21	0.87	0.11

C Supplementary Analysis

C.1 Training Time Analysis

As described in the main paper, since USB-NeRF [6] is based on BARF [7], we re-implement the interpolation method used in USB-NeRF based on Tri-MipRF [4] to maintain consistency of the backbone. We test the training time and querying time of the methods used in the main paper, especially comparing the time consumption between the interpolation method and our method with the same backbone. From Tab. S1, we can observe that the training and querying progress of NeRF and BARF is particularly slow due to the adoption of the coordinate-based MLPs in the network. Due to the Tri-Mip representation in Tri-MipRF [4], Tri-MipRF and its extensions can achieve both high-fidelity anti-aliased renderings and efficient reconstruction. The training and querying speeds have been significantly improved. Specifically, by comparing Tri-MipRF and Tri-MipRF-BA, it can be observed that the training time doubles nearly after introducing bundle adjustment. Subsequently, the computation time of URS-NeRF is longer than Tri-MipRF-BA when estimating additional velocities. Finally, the training and querying time of USB-NeRF-RE is the longest, mainly due to the complex cubic interpolation calculation and the differentiation in the backpropagation process. It is worth noting that the training time of the DiffSfM is similar to Tri-MipRF-BA. However, DiffSfM requires an additional 22,859.34s to eliminate the rolling shutter effect in the images.

C.2 Generality Analysis

We conduct further analysis of the generality of our method. As shown in Fig. S1, our method can reconstruct the scene with only GS images, only RS images, and mixed RS+GS images due to introducing the estimated parameters $\mathbf{v}$ and $\boldsymbol{\omega}$ which are independent of the camera poses. This indicates that our method is unaffected by the order of input images and does not require restricting the types of images, which conforms to the generality of utilizing multi-source data for reconstruction in SfM.

C.3 Additional Experimental Results

We note that we have reported the quantitative comparisons on the training view on WHU-RS dataset in Tab.2 (*cf*. main paper), which indicates the effectiveness

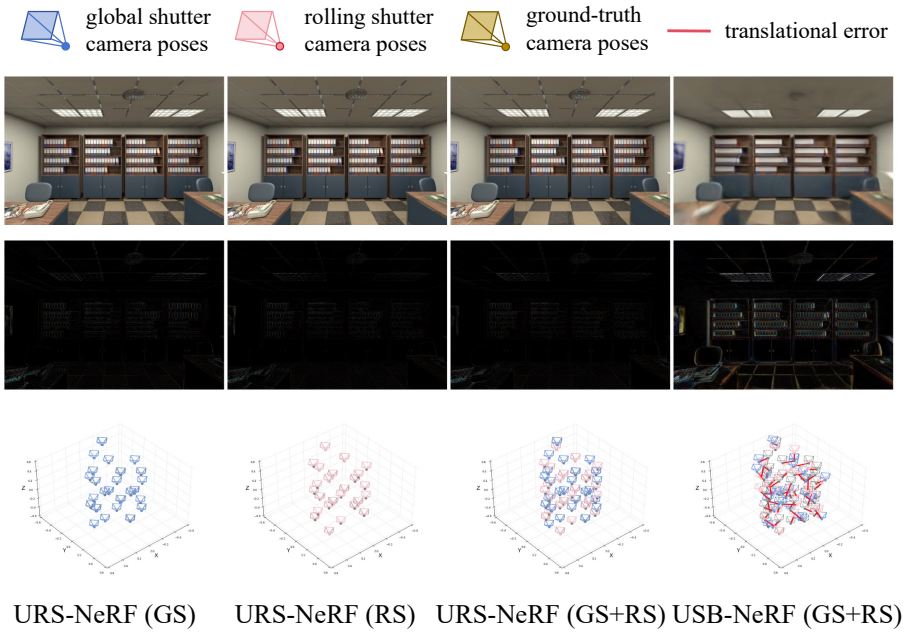

Fig. S1: Given a set of unordered GS/RS images, our model can simultaneously learn the undisturbed 3D scene representation and recover the unordered camera poses with only GS images (1^{th} col), only RS images (2^{th} col) and mixed RS, GS images (3^{th} col). However, USB-NeRF can only take sequential images as input and cannot process hybrid images of RS and GS simultaneously. This limits its practical applications (*e.g*. reconstruction using different types of cameras or utilizing crowdsourced data for reconstruction). The second row presents residual images (the darker the better) that are defined as the absolute difference between the rendered undisturbed images (first row) and ground truth global shutter images.

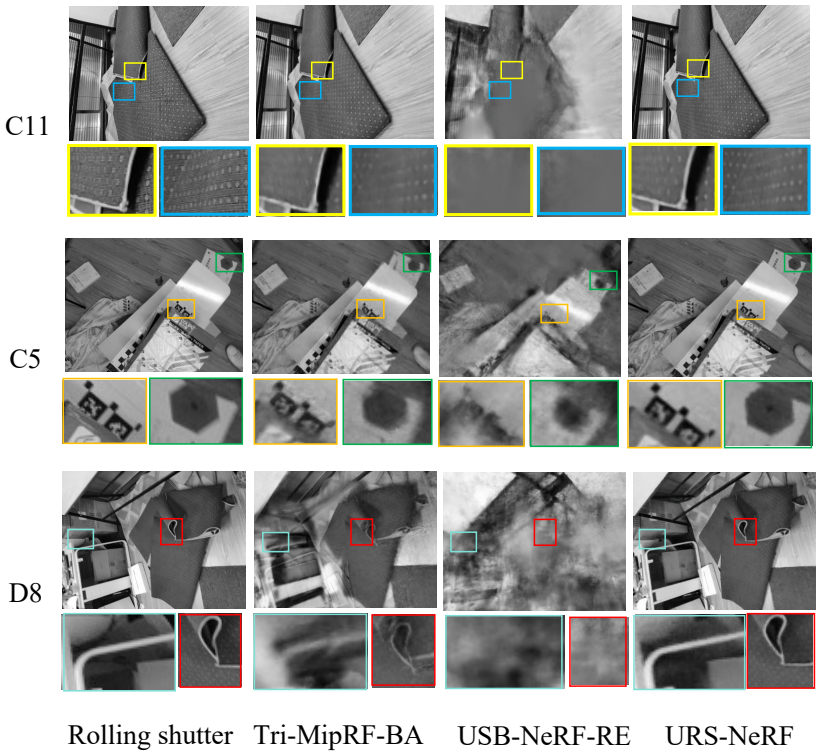

Fig. S2: Qualitative comparisons on ZJU-RS datasets. The detailed and overall images demonstrate that our method achieves better performance compared to other works on unordered images.

of removing the rolling shutter effect on the training view. In tab. S2, we also evaluate the performance of our method against the state-of-the-art methods in terms of novel view synthesis. Some conclusions consistent with the main paper can be obtained. URS-NeRF still outperforms other methods on the unordered datasets. However, USB-NeRF-RE cannot handle unordered input images due to the sequential constraints used in USB-NeRF-RE.

We also conduct additional quantitative and qualitative experimental analyses on the unordered view of real ZJU-RS datasets. Tab. S3 presents the accuracy of trajectory estimation using different methods. Fig. S2 depicts the quality of rendering, while Fig. S3 shows the recovered trajectories compared with the ground truth. These results demonstrate that our method effectively improves the performance of the reconstruction with the images captured by the smartphones.

D Limitation Discussion

As mentioned in the main paper, introducing the estimated parameters $\mathbf{v}$ and $\boldsymbol{\omega}$ which are independent of the camera poses increases the degree-of-freedom of

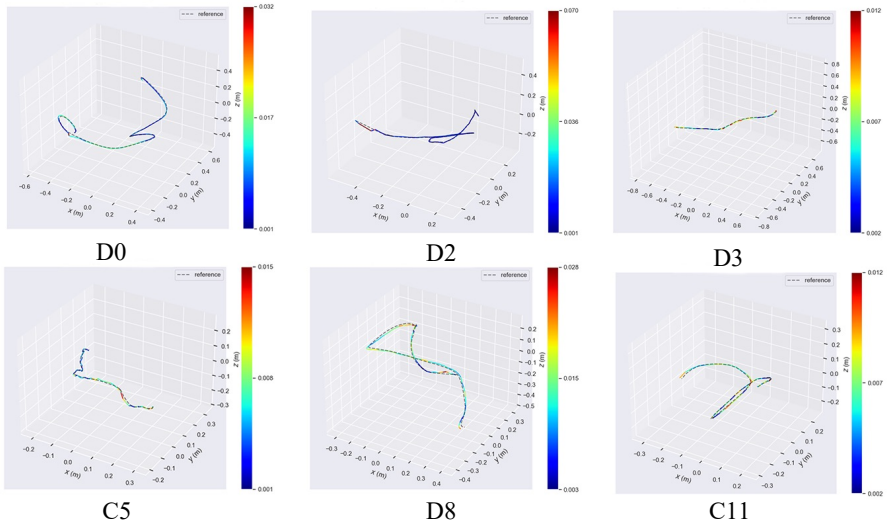

Fig. S3: Comparisons of estimated trajectories of real ZJU-RS datasets. The experimental results demonstrate that our method can recover the motion trajectories of unordered images.

Table S3: Camera pose estimation on the unordered view of ZJU-RS dataset. We evaluate the translation error (m) and rotation error (°). For each metric, the best in **bold**.

	BARF		DiffSfM		Tri-MipRF-BA		USB-NeRF-RE		URS-NeRF	
	Trans	Rot	Trans	Rot	Trans	Rot	Trans	Rot	Trans	Rot
D0	0.047	8.307	0.010	**1.393**	0.012	2.104	0.147	20.80	**0.008**	2.663
D2	0.064	2.999	0.015	**1.075**	0.010	1.916	0.101	10.99	**0.007**	3.081
D3	0.045	11.258	**0.007**	1.398	0.009	3.131	0.195	28.91	**0.007**	**1.334**
D8	0.021	5.058	0.027	**2.274**	0.033	2.264	0.112	7.709	**0.014**	2.668
C5	0.023	5.286	0.010	1.787	0.013	5.866	0.079	11.134	**0.007**	**1.694**
C11	0.033	4.047	0.010	2.032	0.009	1.987	0.081	5.725	**0.007**	**1.000**

the model compared to the interpolation methods. Consequently, the accuracy of our method on sequential data, particularly for intense camera motion, is inferior to the interpolation method used in USB-NeRF. However, the flexibility and generalizability of our method are significant advantages in practical applications. Depending on the specific application scenarios, we can flexibly choose between these two methods.

E Video Comparison

To further demonstrate the advantage of our method, we also present more videos on the web page that demonstrate the ability of our method to recover high-quality global shutter images from the rolling shutter training images and generate images with different degrees of RS effects using the estimated velocity and angular velocity.

References

1. Cao, L., Ling, J., Xiao, X.: The whu rolling shutter visual-inertial dataset. IEEE Access **8**, 50771–50779 (2020)
2. Diederik, P.K.: Adam: A method for stochastic optimization. (No Title) (2014)
3. Grupp, M.: evo: Python package for the evaluation of odometry and slam. https://github.com/MichaelGrupp/evo (2017)
4. Hu, W., Wang, Y., Ma, L., Yang, B., Gao, L., Liu, X., Ma, Y.: Tri-miprf: Tri-mip representation for efficient anti-aliasing neural radiance fields. In: ICCV (2023)
5. Jinyu, L., Bangbang, Y., Danpeng, C., Nan, W., Guofeng, Z., Hujun, B.: Survey and evaluation of monocular visual-inertial slam algorithms for augmented reality. Virtual Reality & Intelligent Hardware **1**(4), 386–410 (2019)
6. Li, M., Wang, P., Zhao, L., Liao, B., Liu, P.: Usb-nerf: Unrolling shutter bundle adjusted neural radiance fields. arXiv preprint arXiv:2310.02687 (2023)
7. Lin, C.H., Ma, W.C., Torralba, A., Lucey, S.: Barf: Bundle-adjusting neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5741–5751 (2021)
8. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
9. Zhuang, B., Cheong, L.F., Hee Lee, G.: Rolling-shutter-aware differential sfm and image rectification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 948–956 (2017)