2022 The 22nd International Conference on Control, Automation and Systems (ICCAS 2022)
BEXCO, Busan, Korea, Nov. 27~Dec. 01, 2022

Visual-Inertial Odometry Priors for Bundle-Adjusting Neural Radiance Fields

Hyunjin Kim, Minkyeong Song, Daekyeong Lee, and Pyojin Kim*

Department of Mechanical Systems Engineering
Sookmyung Women's University, Seoul, South Korea
{yunjingong12,smk615,swbubl,pjinkim}@sookmyung.ac.kr

Abstract: We present bundle-adjusting Neural Radiance Fields (BARF) with motion priors. Neural Radiance Field (NeRF) has opened up tremendous potential for neural volume rendering and 3D scene representations in recognition of their ability to synthesize photo-realistic novel views. BARF mitigates NeRF's reliance on accurate 6-DoF camera poses, enabling scene learning with inaccurate camera poses. However, initializing estimates far from an optimal solution, such as BARF, can easily fall into local minima. We utilize Visual-Inertial Odometry Motion Priors to the BARF, which jointly optimizes 3D scene representations and camera poses, providing higher accuracy in view synthesis and a more stable motion estimate. The proposed method achieves results that outperform original BARF in real-world data, demonstrating the effectiveness of motion priors to knowledge use.

Keywords: Neural Radiance Fields, View Synthesis, Neural Rendering, Visual-Inertial Odometry (VIO)

1. INTRODUCTION

Image-based map construction is one of the fundamental goals in computer vision, robotics, AR and VR applications, and autonomous driving. Recently, coordinate-based neural representations [1], [2], [3] have attracted increasingly significant attention in this field. In particular, Neural Radiance Fields (NeRF) [4], which aim to render novel viewpoints of a scene given RGB images and corresponding 6-DoF camera poses, have gained popularity owing to their simplicity and powerful view synthesis ability.

NeRF [4] learns 3D scene representation from images and encodes the entire volume space as a continuous function parameterized by Multi-Layer Perceptron (MLP). Although NeRF has tremendous potential for view synthesis and 3D scene representation; however, NeRF requires highly accurate 6-DoF camera positions, which is an unrealistic condition. NeRF uses off-the-shelf SfM algorithms such as COLMAP [5] to generate accurate camera poses corresponding to given images.

Several works have proposed models to overcome these limitations. Bundle-Adjusting Neural Radiance Fields (BARF) [6] and NeRF−−[7] introduce methods for estimating camera poses and training NeRF simultaneously, enabling novel view synthesis from imperfect or even unknown camera poses. BARF and NeRF−− initialize all camera frames with identity matrices on real-world scenes. Although these methods can jointly optimize camera poses, NeRF network is sensitive to initialization. Therefore, using the initial pose of real-world data as an identity may result in optimized parameters being trapped in local minima, not being an optimal solution, and degrading view synthesis quality.

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2021R1F1A1061397). This work was supported by Korea Foundation for Women In Science, Engineering and Technology (WISET) grant funded by the Ministry of Science and ICT(MSIT) under the team research program for female engineering students. (No. WISET-2022-065). * Corresponding author

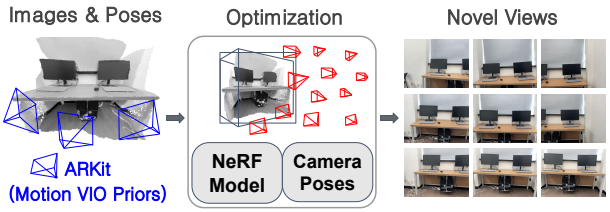


Fig. 1. We propose a BARF-based framework using motion priors obtained from Apple ARKit as initial camera poses.

To solve the problems mentioned, we propose NeRF network and pose optimization technique using motion priors. We obtain the motion priors from commercial libraries such as Apple ARKit [8], Apple's visual-inertial odometry (VIO) algorithm. We use them as initial pose values for BARF (Fig. 1). There are two advantages to using a good initial guess as an initial value for the optimization process. First, the initialization of a good initial guess prevents you from falling into the local minima. Second, the initialization of a good initial guess enables fast convergence. We demonstrate better and faster novel view synthesis results and more stable pose estimation results, outperforming the baseline BARF train with the identity matrix. Additionally, we compare camera pose estimation results with the ground-truth poses obtained with the motion capture system, OptiTrack.

In summary, our contributions are as follows:

- We exploit good motion priors obtained from ARKit as an initial value to jointly optimize view synthesis and camera poses for complex scenes.
- We explore that the proposed method can successfully recover high-fidelity view synthesis and poses for complex scenes using good initials.

2. RELATED WORK

In contrast to traditional representations such as point clouds, mesh, or voxels in 3D scene representations, coordinate-based neural representations [2], [3] can more

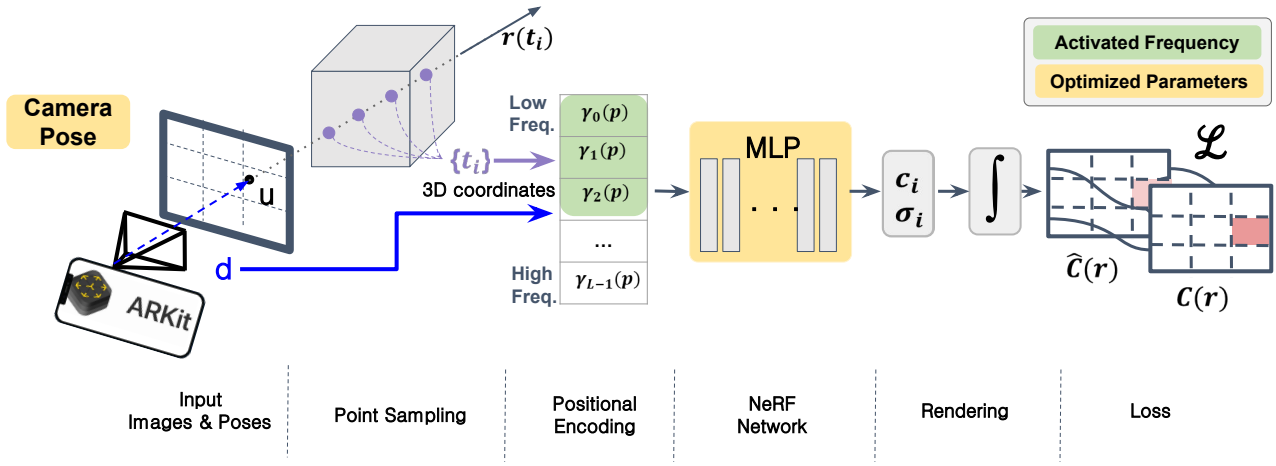


Fig. 2. An overview of BARF with motion priors from ARKit. We jointly optimize the camera 6-DoF poses and the NeRF network. We set the motion priors from ARKit to the initial value of the camera poses and sample discrete points $\{t_i\}$ along the direction $\mathbf{d}$ of the camera ray to render the color of pixel $\mathbf{u}$ on the image. Sampled 3D points feed to NeRF through the smooth Positional Encoding γ and obtain $\widehat{C}$ through volume rendering. Since the entire pipeline is differentiable, we can jointly optimize 6-DoF poses and MLP together.

efficiently represent the complex scene and memorize 3D information in space. This representation is widely used in 3D reconstruction and novel view synthesis [4], [9], [10].

Neural Radiance Fields (NeRF) [4] propose a differentiable rendering method that learns neural representations from input images. It represents the underlying 3D scene using coordinate-based multi-layer perceptrons (MLP). NeRF can produce unprecedented levels of realistic view synthesis in challenging scenes, and the simplicity and superior performance of the network have opened up enormous potential for the presentation of the 3D scene. However, this method relies heavily on accurate camera poses. Although there are some tricky conditions that it is possible only in a small range of forward-facing scenes and takes a long optimization time, several studies have reduced the dependence on pre-calculated pose information.

iMAP [11] is a real-time SLAM system that can efficiently store 3D information in space based on NeRF MLP. iNeRF [12] proposes a novel pose estimation method inverting the NeRF model given a well-trained NeRF network. NeRF−− [7], Self-Calibrating Neural Radiance Fields [13], and Bundle-Adjusting Neural Radiance Fields (BARF) [6] jointly optimize camera poses estimations and scene representations. These methods train the NeRF network without camera poses; there is no need for camera pose pre-calculation through the SfM algorithm. However, pose optimization is limited to generally forward-facing scenes. GNeRF [14] achieves the NeRF reconstruction problem in the unknown camera poses by integrating Generative Adversarial Networks (GAN) and NeRF, and pose optimization is also possible in a wide range of 360-degree captures beyond forward-facing. BARF conducts pose optimization by applying a smooth mask using a weight proportional to the optimization progress in positional encoding. BARF initializes the poses as an identity rigid body transformation matrix for real-world scenes. Initializing with identity without using appropriate initial values may be stuck into suboptimal rather than optimal solutions in optimization problems. Here we propose a method to train the BARF by setting motion priors from ARKit to the initial value of the poses.

3. PROPOSED METHOD

We build upon BARF, incorporating motion priors from ARKit. Fig. 2 shows an overview of our method. Our goal is to improve rendering quality while accelerating rendering by applying self-captured data to BARF using motion priors from ARKit. We briefly cover Neural Radiance Fields [4] and BARF [6].

3.1 Neural Radiance Fields and BARF

NeRF [4] encodes 3D scenes with continuous neural network function F parameterized by a multi-layer perceptron (MLP) that maps input 3D location to RGB color and volume density. The key is that because this function is differentiable, MLP can optimize by minimizing the photometric error between the ground-truth images and the synthesis images. $F_\Theta : (\mathbf{x}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma)$ maps location $\mathbf{x} \in \mathrm{R}^3$ in 3D space and viewing direction $\mathbf{d} \in \mathrm{R}^2$ to radiance color $\mathbf{c} \in \mathrm{R}^3$ and volume density $\sigma \in \mathrm{R}$. Θ indicates MLP networks weight. This architecture ensures a non-Lambertian effect because it considers viewing directions $\mathbf{d}$. To render the color of the pixel $\mathbf{u} \in \mathrm{R}^2$ on the image, casting ray $r(t) = o + td$ passing through pixel $\mathbf{u}$ in direction $\mathbf{d}$ from camera center $\mathbf{o}$ along the camera ray. Given near and far bounds in the ray, sampling discrete N points $\{t_i\}_{i=1}^N$ along the ray $r(t)$ with lengths $\delta_i = t_{i+1} - t_i$. Each sampled 3D point is fed into the MLP. The expected color $\hat{C}(r)$ of $r(t)$ is is :

$$\hat{C}(r) = \sum_{i=1}^{N} T_i(1 - exp(-\sigma_i\delta_i))c_i, \qquad (1)$$

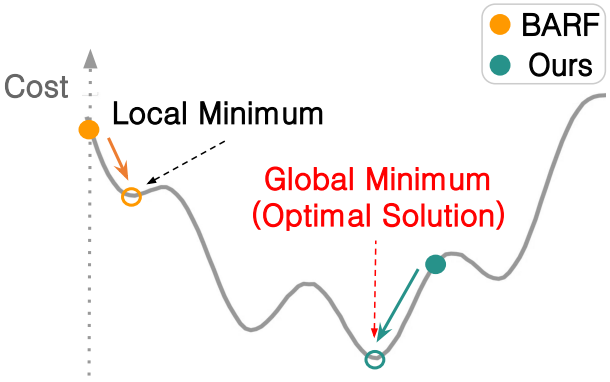


Fig. 3. We can reach the optimal solution by performing optimization with good initial camera poses. On the other hand, because BARF initializes to identity, it can easily fall into local minima.

where

$$T_i = exp\left(-\sum_{j=1}^{i-1}\sigma_i\delta_i\right) \tag{2}$$

In contrast to the original NeRF, BARF jointly optimizes camera poses and rendering results by setting the initial camera pose as identity transformation. So BARF is interested in finding 3D scene representations and solving accurate camera poses simultaneously. In other words, BARF parameterizes the camera pose $\mathbf{p}$, minimizes the photometric error in input pixels given $\{I_i\}_{i=1}^{M}$, and optimizes the camera poses $\{p_i\}_{i=1}^{M}$ corresponding to each image.

$$\min_{p_1,\dots p_M,\Theta}\sum_{i=1}^{M}\sum_{u}\left\|\hat{I}(u;p_i,\Theta)-I_i(u)\right\|_2^2 \tag{3}$$

The camera pose p is as a transformation matrix $[R|t]$ in SE (3), where $R \in SO(3)$ indicates the camera rotation and $t \in R^3$ indicates the translation. As mentioned earlier, BARF initializes all camera rotation R as identity matrix and translation t as zero vectors, i.e., the input camera frames are located in origin and look at the -z-axis. However, we incorporate motion priors into BARF to initialize with poses acquired through ARKit.

3.2 Positional Encoding

Positional Encoding is an essential component that makes NeRF high-fidelity synthesis possible. Positional Encoding $\gamma(p)$ maps the input pose $\mathbf{p}$ to the higher dimensional space before being fed to the MLP network. The encoding function with L frequency bases is defined as:

$$\gamma(p) = [p, \gamma_0(p), \gamma_1(p), ..., \gamma_L - 1(p)] \tag{4}$$

Applying naively Positional Encoding to BARF makes joint optimization difficult to update effectively and consistently. Therefore, BARF presents a simple but effective strategy to control various frequency components, applying a smooth mask to positional encoding via $\alpha \in [0, L]$ proportional to the optimization process. The weight w_k applied to each k-th frequency encoding component is:

Scene	Camera pose optimization			
	Rot (°) ↓		Trans (m)↓	
	BARF	Ours	BARF	Ours
Fan	9.022	**0.683**	0.117	**0.008**
Truck	7.731	**0.380**	0.55	**0.002**

Table 1. Quantitative results of optimized camera poses. We report the difference between ground truth poses obtained from OptiTrack and optimized camera poses.

$$w_k(\alpha) = \begin{cases} 0 & if\ \alpha < k \\ \frac{1-cos((\alpha-k)\pi)}{2} & if\ 0 \le \alpha - k < 1 \\ 1 & if\ \alpha - k \ge 1 \end{cases} \tag{5}$$

The k-th frequency component of γ_k is defined as:

$$\gamma_k(\mathbf{p};\alpha) = w_k(\alpha)\cdot[cos(2^k\pi\mathbf{p}), sin(2^k\pi\mathbf{p})] \tag{6}$$

Thus the Jacobian of γ_k becomes

$$\frac{\partial\gamma_k(\mathbf{p};\alpha)}{\partial\mathbf{p}} = w_k(\alpha)\cdot 2^k\pi\cdot[-sin(2^k\pi\mathbf{p}), cos(2^k\pi\mathbf{p})] \tag{7}$$

This w_k enables only low-frequency components at the start of the optimization process, allowing them to generate smooth signals. Then the higher frequency components are gradually activated for high-fidelity view synthesis.

3.3 BARF with Motion Priors

The BARF initializes all camera frames to the origin, i.e., the rotation and translation movement are zero. However, we apply a good initial guess from Apple ARKit to the optimization process to quickly converge the BARF and significantly improve the rendering quality. ARKit is one of the most accurate and stable VIO algorithms [15], so we obtain the ARKit 6-Dof trajectory with the custom iOS app [16].

The nonlinear optimization process is susceptible to suboptimal solutions if poorly initialized (Fig. 3). In the nonlinear optimization process, BARF, which uses the identity matrix as the initial guess, may fall into the local minima. Our model, with a good initial guess obtained from ARKit, can optimize scene representations and poses effectively without falling into local minima during the optimization process [17]. We can also draw results of higher rendering quality and enable stable pose estimation.

4. EXPERIMENTS

We evaluate the proposed method by collecting training data from various forward-facing indoor scenes, where the camera poses are known through ARKit. ARKit is Apple's software framework that includes a

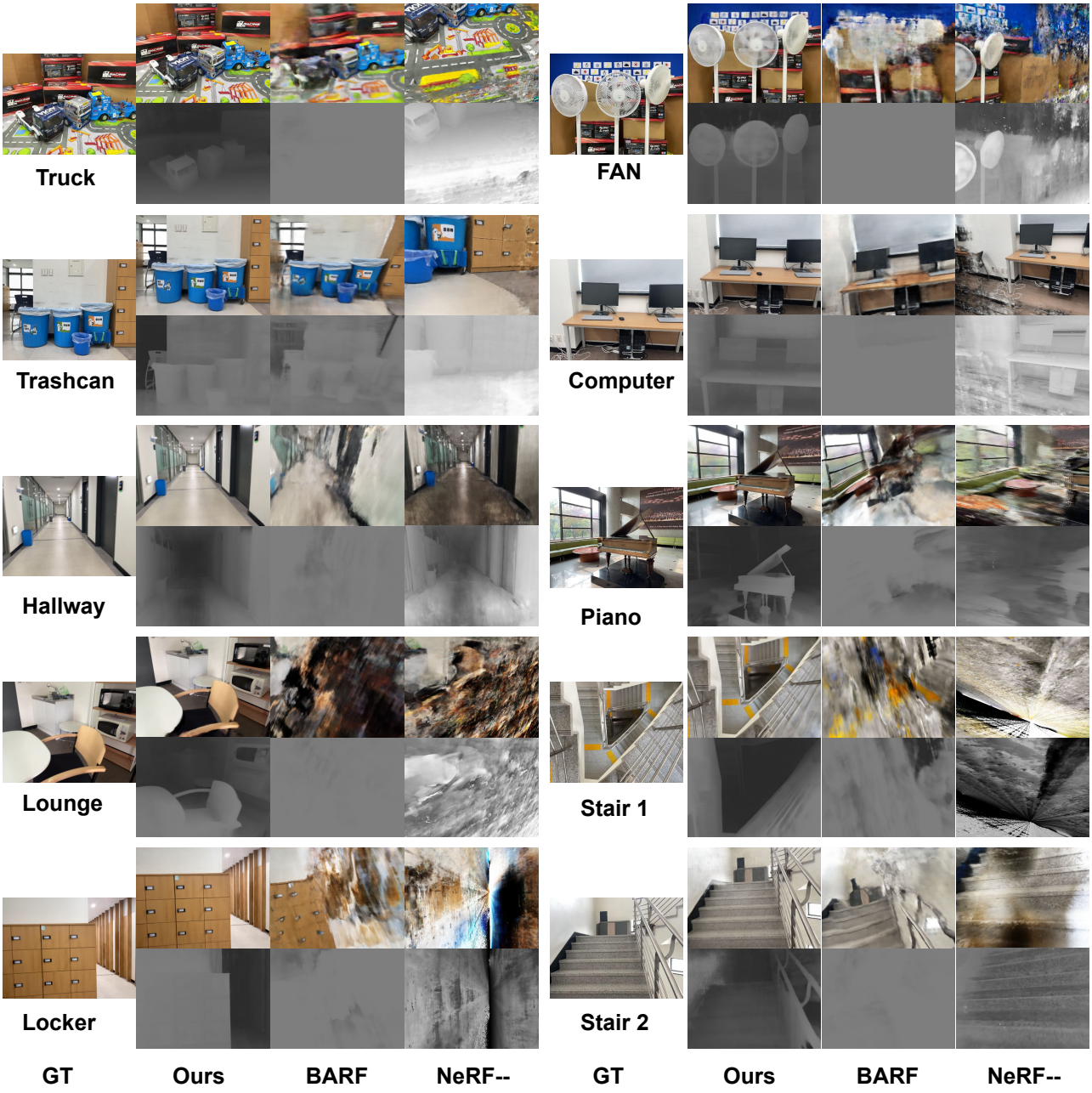


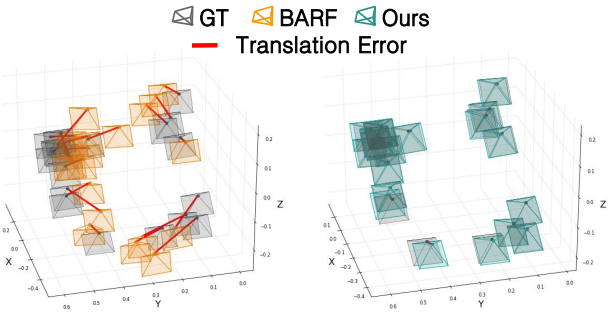
Fig. 4. Comparison of test-sets views for scenes from real-world data acquired with ARKit. We visualize the ground-truth (GT) image, the synthesized image, and the predicted depth map. Our method can effectively optimize scene representations and poses compared to the BARF [6] and NeRF−− [7], allowing us to recover details of an object such as the Piano data. BARF cannot incur high-fidelity reconstructions and depth estimations owing to incorrect pose convergence. View rendering is only possible in places with slight camera movements, such as the Computer or Trashcan data. Similarly, as in the Lounge and Stair1 data, NeRF−− also diverges to poor synthesis and depth estimation quality as the capture range increases.

VIO algorithm to produce Apple's augmented reality apps. We use the custom iOS app [16] to collect custom data consisting of ARKit 6-Dof camera poses and RGB images using iPhone 12 Pro Max running iOS 14.7.1. with LiDAR sensors. We also evaluate the pose estimation results using the pose information obtained from OptiTrack, the motion capture system, as a ground truth. We compare the proposed method with the original BARF [6] and NeRF−− [7] in which all camera frames are initialized with the identity matrix.

We measure quantitative performance regarding optimized pose error and view synthesis quality. We follow evaluation methods employed in the BARF, and the de facto standard, widely used for evaluating view synthesis results in the NeRF series. We report quantitative results based on the mean of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Map (SSIM) [18], and the Learned Perceptual Image Patch Similarity (LPIPS) [19] perceptual metric to evaluate im-

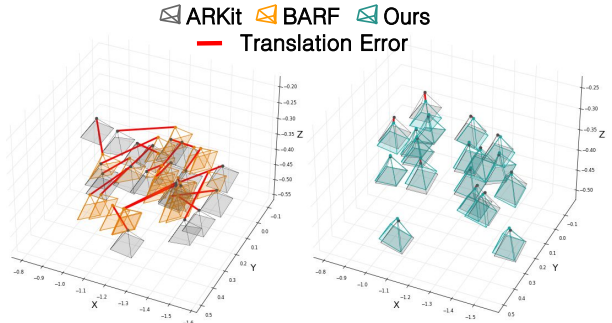
Scene	Camera pose optimization				View synthesis quality								
	Rot (°) ↓		Trans (m) ↓		PSNR ↑			SSIM ↑			LPIPS ↓		
	BARF	Ours	BARF	Ours	NeRF−−	BARF	Ours	NeRF−−	BARF	Ours	NeRF−−	BARF	Ours
Transhcan	15.2	**2.274**	0.196	**0.029**	8.58	12.37	**23.61**	0.4	0.53	**0.81**	0.8	0.71	**0.25**
Computer	9.613	**1.411**	0.086	**0.008**	8.19	16.19	**28.24**	0.35	0.65	**0.9**	0.73	0.52	**0.13**
Stair2	27.589	**7.61**	0.630	**0.18**	10.28	13.28	**15.08**	0.23	0.31	**0.32**	0.77	0.84	**0.62**
Hallway	13.946	**4.561**	0.136	**0.016**	14.47	11.12	**22.67**	0.56	0.52	**0.81**	0.55	0.77	**0.25**
Piano	21.301	**4.452**	0.480	**0.064**	9.11	9.03	**18.41**	0.2	0.26	**0.56**	0.74	0.88	**0.35**
Lounge	51.081	**4.99**	0.478	**0.055**	9.21	7.58	**29.38**	0.47	0.35	**0.93**	0.79	0.88	**0.11**
Stair1	21.865	**3.405**	0.120	**0.016**	9.0	11.56	**20.55**	0.22	0.24	**0.49**	0.79	0.87	**0.43**
Locker	32.938	**1.17**	0.265	**0.017**	8.26	12.69	**29.06**	0.21	0.52	**0.9**	0.95	0.84	**0.12**
Mean	24.191	**3.734**	0.298	**0.048**	9.637	11.727	**23.375**	0.33	0.422	**0.715**	0.765	0.788	**0.282**

Table 2. Quantitative comparison between our model and BARF and NeRF−− on real-world data acquired with ARKit. Since ground-truth poses are not accessible in general indoor scenes, the pose accuracy was evaluated by calculating the difference between optimized poses and poses acquired through ARKit. Our method can optimize for high-fidelity view synthesis, and exact camera poses.



(a) BARF (b) Ours

Fig. 5. Visual comparison of optimized camera poses and OptiTrack GT poses in the Fan scene. The result of our method (right) demonstrates consistency with the OptiTrack data and successful camera pose estimation, but the result from BARF (left) provides inaccurate camera poses.



(a) BARF (b) Ours

Fig. 6. Visual comparison of optimized camera poses and ARKit poses in the Piano scene. Since we cannot obtain ground truth from general indoor data, we calculate the error from the estimated poses using motion priors from ARKit as GT. Our method (right) agrees with ARKit poses, while BARF cannot successfully estimate poses.

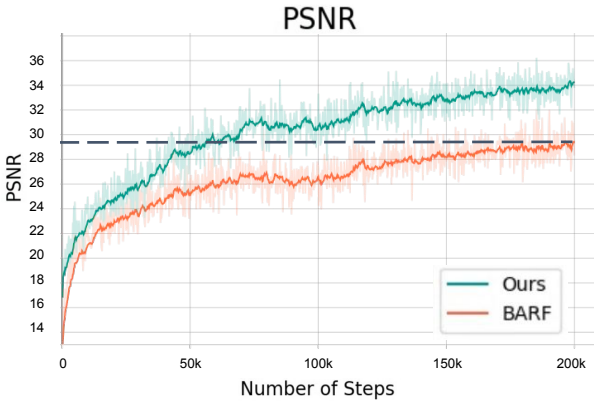


Fig. 7. Quantitative results for PSNR in the Lounge scenes. The proposed method makes convergence of nonlinear optimization (MLP and camera poses) faster than BARF .

age rendering quality and report the average rotation and translation error of the estimated poses. Since true camera poses are not accessible in general indoor scenes except OptiTrack environments, we evaluate the accuracy by calculating the difference between optimized poses and poses acquired through ARKit.

We visualize the result in Fig. 4 and report quantitative results in Table 2. Our method produces stable motion estimates similar to ground truth poses and achieves high-fidelity view synthesis, and is superior on all metrics by comparing BARF and NeRF−− initialized with the identity matrix (Table 2).

Fig. 5 and Table 1 show that the camera poses recovered from the proposed method are consistent with ground truth acquired from OptiTrack, demonstrating the localization ability initialized with ARKit proposed by our method.

Our approach also demonstrates high fidelity view rendering and depth map results (Fig. 4). We can recover 3D scene representations more robustly and accurately than BARF and NERF−−; however, BARF converges to the wrong camera poses when the movement increases and shows poor results in the learned scene (Fig. 6). This em-

phasizes the importance of using a good motion priors for BARF as initial values.

Figure. 7 shows that our method accelerates the origin BARF for the Lounge data. The proposed method for the same PSNR value reaches 60K, whereas the original BARF reaches 200K.

5. CONCLUSION

We integrate motion priors from ARKit, the VIO algorithm, in the BARF for stable and fast optimization. We avoid local minima in the nonlinear optimization process using a good initial guess and hence can effectively optimize scene representations, and 6-Dof camera poses. Our experiments with motion priors demonstrate overwhelming performance by overcoming the limitations of slow optimization and rendering of BARF and inconsistent rendering results on real-world data. Using a proper priors motion as the initial value results in higher accuracy and more stable motion estimates. Thorough analyses of results for our methods support the effectiveness of using the motion priors knowledge. We hope to inspire future work in pose estimation and scene reconstruction of the neural radiance field, especially in how much the initial guess influences the optimization process according to the neural radiance field model.

REFERENCES

[1] Z. Chen and H. Zhang, "Learning implicit fields for generative shape modeling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5939–5948.

[2] M. Michalkiewicz, J. K. Pontes, D. Jack, M. Baktashmotlagh, and A. Eriksson, "Implicit surface representations as layers in neural networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4743–4752.

[3] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 165–174.

[4] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *European conference on computer vision*. Springer, 2020, pp. 405–421.

[5] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.

[6] C.-H. Lin, W.-C. Ma, A. Torralba, and S. Lucey, "Barf: Bundle-adjusting neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5741–5751.

[7] Z. Wang, S. Wu, W. Xie, M. Chen, and V. A. Prisacariu, "Nerf–: Neural radiance fields without known camera parameters," *arXiv preprint arXiv:2102.07064*, 2021.

[8] "Apple ARKit," https://developer.apple.com/documentation/arkit/.

[9] M. Tancik, V. Casser, X. Yan, S. Pradhan, B. Mildenhall, P. P. Srinivasan, J. T. Barron, and H. Kretzschmar, "Block-nerf: Scalable large scene neural view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 8248–8258.

[10] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. Sajjadi, A. Geiger, and N. Radwan, "Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5480–5490.

[11] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison, "imap: Implicit mapping and positioning in real-time," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6229–6238.

[12] L. Yen-Chen, P. Florence, J. T. Barron, A. Rodriguez, P. Isola, and T.-Y. Lin, "inerf: Inverting neural radiance fields for pose estimation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1323–1330.

[13] Y. Jeong, S. Ahn, C. Choy, A. Anandkumar, M. Cho, and J. Park, "Self-calibrating neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5846–5854.

[14] Q. Meng, A. Chen, H. Luo, M. Wu, H. Su, L. Xu, X. He, and J. Yu, "Gnerf: Gan-based neural radiance field without posed camera," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6351–6361.

[15] "An Empirical Evaluation of Four Off-the-Shelf Proprietary Visual-Inertial Odometry Systems," https://arxiv.org/abs/2207.06780/.

[16] "ios logger," https://github.com/hyunJIN7/ios_logger/.

[17] L. Carlone, R. Tron, K. Daniilidis, and F. Dellaert, "Initialization techniques for 3d slam: a survey on rotation estimation and its use in pose graph optimization," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 4597–4604.

[18] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[19] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.