Adaptive Keyframe Selection for Online Iterative NeRF Construction

Joshua Wilkinson, Jack Naylor, Ryan Griffiths and Donald G. Dansereau

***Abstract*— We propose a method for intelligently selecting images for building neural radiance fields (NeRFs) from the large number of frames available in typical robot-mounted cameras. Our approach iteratively constructs and queries a NeRF to adaptively select informative frames. We demonstrate that our approach maintains high-quality representations with a 78% reduction in input data and reduced training time in single-pass mapping, while preventing unbounded growth of input frames in persistent mapping. We also demonstrate our adaptive approach outperforming non-adaptive spatial and temporal methods in terms of training time and rendering quality. This work is a step towards persistent robotic NeRF-based mapping.**

I. Introduction

Simultaneous localisation and mapping (SLAM) is a critical task in robotics, requiring both accurate representation of the environment and accurate localisation of the robotic platform. In complex environments such as around reflective windows or transparent doors this can be extremely challenging as view-dependent appearance causes conventional approaches to fail. Neural radiance fields (NeRFs) [1] offer an avenue for overcoming this challenge, as these have demonstrated high-fidelity representation of complex visual appearance.

There has been substantial progress in adapting NeRF to robotic mapping [2]–[7]. However a key challenge arises in the volume of imagery available to a mobile robot. A single camera can collect tens or hundreds of frames of video per second, and the total number of frames grows indefinitely over time for long-term deployments. To effectively leverage neural representations it is necessary to curate this stream of imagery, limiting on-board data and compute requirements while maintaining the quality of the resulting models.

In this work we propose a method for adaptively selecting NeRF input frames from a video stream. We achieve this by iteratively constructing a NeRF-based map representation and querying the intermediary model to judge the novelty of new input frames. We show this approach outperforms non-adaptive spatial and temporal frame selection techniques by maintaining high representation quality with fewer input frames and less training time. Our method allows a robot to persistently survey an area without growing an unbounded collection of input imagery.

We envision adaptive frame selection complementing pose and model refinement as well as selective frame removal in persistent robotic NeRF-based mapping systems.

Limitations: we do not address pose estimation in this work, and anticipate incorporating pose refinement as future work.

II. Related Work

Complex appearance like reflection and refraction are not well handled by state-of-the-art SLAM systems [8]–[10]. Both feature-based and direct methods fail around

The authors are with the Australian Centre for Robotics, School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney. donald.dansereau@sydney.edu.au

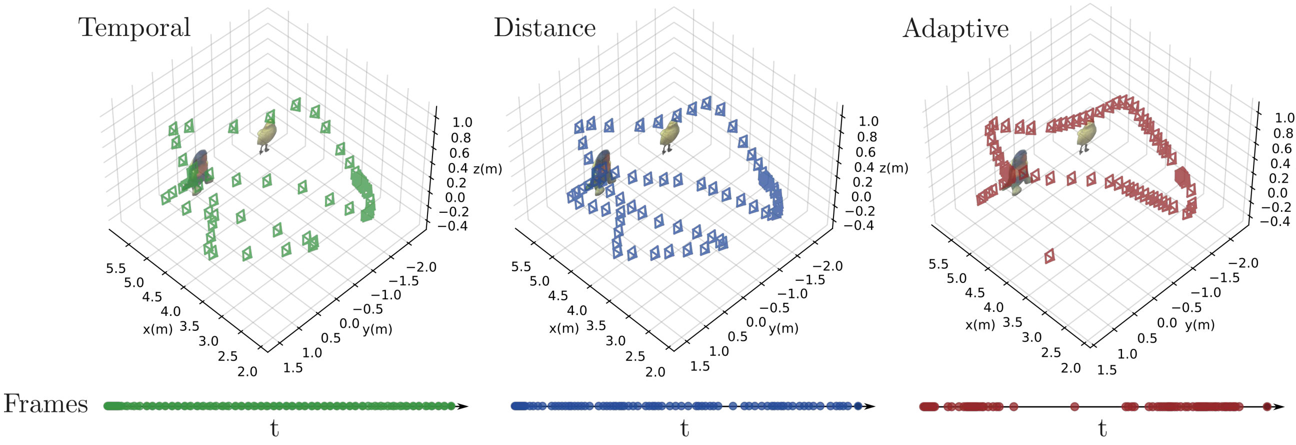

Fig. 1: Visual depiction of the keyframes selected for a typical robotic trajectory (Sequence 35 in the LearnLFOdo dataset). A temporal depiction of selected keyframes is shown at bottom. The proposed adaptive method only incorporates frames covering regions where the scene is poorly represented, while non-adaptive methods incorporate frames throughout the trajectory irrespective of scene content. Our approach yields higher-quality representations with less data and compute and allows robots to operate with full frame rate video and over extended deployments.

view-dependent effects, impacting map fidelity. Neural scene representations [11], [12] have recently gained traction as a method of handling visually complex scenes [8], [13]–[18]. Neural radiance fields in particular [1], [19], [20] have enabled photorealistic representation of reflection, refraction and scattering within a scene. In robotics this has enabled manipulation, localisation, and navigation around visually complex surfaces [21]–[27].

However key challenges remain in leveraging NeRFs in robotics. For real-time operation, prior approaches require depth sensors, or remove the view-dependent portion of the representation [26], [28], limiting their applicability and capability. Others incorporate traditional SLAM front-ends to alleviate some of these constraints [16], however these are limited by the performance of the front-ends which do not generally handle view-dependent appearance. While some approaches to neural odometry address these issues [29], [30], they are not full mapping solutions, and a complete end-to-end robotic neural mapping pipeline is yet to be realised.

NeRF works best while operating with sufficient input imagery [4], [6], [31]–[33], but it becomes computationally prohibitive as the number of input images grows. This is a critical issue given the large number of images available to a mobile robot, and the unbounded nature of images available where robots operate over extended periods.

In this work we propose a representation-driven keyframe selection method for a camera-only SLAM pipeline. In particular, we use the appearance representation of the NeRF as opposed to geometric innovation [16], [26], [28] by leveraging the view synthesis capability of the representation. We select and add frames only to poorly represented regions of the scene. This allows a robot to curate its input data to effectively build NeRF-based maps without being impacted by high input frame rates or long-term accumulation of frames where sites are revisited over time. We believe this work is an important step toward achieving online NeRF-based SLAM.

III. Method

We leverage the view synthesis capabilities of a NeRF to adaptively determine whether new keyframes are required to represent scene regions. We compare our adaptive approach to two non-adaptive baselines that employ temporal and distance based approaches to select new frames. In all cases, we use a common initialisation method to produce constrained NeRFs at the beginning of a trajectory.

A. Keyframe Selection

By leveraging the novel view synthesis capabilities of a NeRF, we develop an adaptive keyframe selection method which balances data efficiency with visual reconstruction quality. We denote the current state of the NeRF by its training dataset K_i and weights Θ_{K_i}. For a new input frame k_{i+1}, we compare the frame with the NeRF's prediction of that frame, computed as peak signal to noise ratio (PSNR). We add the frame to the training set only when the synthesised frame fidelity drops below a threshold γ,

$$K_{i+1} = K_i \cup \{k_{i+1} \,|\, \mathrm{PSNR}(I_{i+1}, \Theta_{K_i}) < \gamma\}. \tag{1}$$

In an incremental construction, this leverages the ability for the NeRF to represent content outside the training set, adding keyframes only when innovative visual information is captured. This results in a set of keyframes which provides a minimal drop in quality of the NeRF while being substantially more data efficient than distance or time based approaches to keyframe selection.

Non-Adaptive Keyframe Algorithms We compare with two naive keyframe selection methods. Firstly, a time-based method in which a new keyframe is selected after a specified amount of time γ_t. This equates to taking every n^{th} frame assuming a constant camera frame rate. The second comparison method uses the odometry information available to robotic platforms. This distance-based approach selects a new keyframe after a given translation γ_d of the robot has occurred. See Fig. 1 for graphical depictions of the frames selected by each method.

B. Iterative NeRF

Due to our operation on a continuous stream of images on a robotic platform, our training approach differs from conventional approaches. After deciding to include an image in the set of keyframes, we retrain a NeRF with the updated keyframe set. We do this to avoid the edge effects of regions of the scene which are seen by few frames, often characterised by floaters. This minimises special treatment required to extend the representation which is not a focus of this work, but is discussed elsewhere [16], [34].

At each step, the adaptive algorithm uses the current NeRF model to predict the view of the camera. Comparing the PSNR between the NeRF render and the frame from the camera allows for an estimation of how well the current training set captures the current view. If the PSNR is poor, the frame is included as a new keyframe.

Initialisation A NeRF with few input views is ill-constrained [4], [6], [32]. We propose a simple solution to initialise the representation at the beginning of a trajectory. We train a NeRF on the first i frames, until the $i+1^{\text{th}}$ frame exceeds our reconstruction threshold γ. We repeat this process until there are sufficient multi-view constraints on the NeRF to converge and produce a high fidelity representation. In this work, we determined that the first $i = 10$ frames produced a sensible representation, and applied the proposed adaptive keyframe selection to each subsequent frame.

IV. Experiments

We benchmark our keyframe selection approach on data captured from a robotic arm, providing ground truth poses. We iteratively train NeRF map representations, progressively incorporating new frames from the stream of images obtained from the robot. We consider both single-pass mapping, in which a scene is covered once, and persistent operation, in which a robot repeatedly covers the same parts of a scene over multiple passes. The latter is representative of long-term

TABLE I: Comparison of keyframe selection algorithms for different sequences in the LearnLFOdo dataset

Method	Metrics	Seq. 17	Seq. 24	Seq. 33	Seq. 35	Seq. 38	Seq. 41	Avg.
Gold Standard	No. Images	192	358	156	303	338	158	250.8
	PSNR (dB) (↑)	27.00	31.38	30.89	29.54	28.32	28.26	29.50
	SSIM (↑)	0.856	0.884	0.896	0.859	0.867	0.850	0.869
Temporal	No. Images	44	74	38	64	71	38	54.8
	Total time (s) (↓)	**1468.03**	3173.75	1163.90	2685.23	2872.21	**1161.67**	2087.46
	PSNR (dB) (↑)	25.51	28.6	**28.12**	27.19	27.13	24.76	27.08
	SSIM (↑)	0.806	**0.869**	0.840	0.820	0.838	0.796	0.828
Distance	No. Images	58	72	28	79	75	64	62.7
	Total time (s) (↓)	1663.47	**1255.58**	907.592	2684.54	2471.47	1662.28	1774.15
	PSNR (dB) (↑)	26.02	28.57	25.26	**27.55**	**27.488**	25.39	26.88
	SSIM (↑)	0.816	0.833	0.801	0.828	0.810	**0.824**	0.819
Adaptive (Ours)	No. Images	65	42	33	80	53	64	56.2
	Total time (s) (↓)	1945.50	1280.67	**772.27**	**2593.03**	**1611.52**	1657.14	**1643.35**
	PSNR (dB) (↑)	**26.32**	**30.2**	27.24	27.12	26.98	**25.55**	**27.50**
	SSIM (↑)	**0.829**	0.833	**0.844**	**0.836**	**0.846**	0.816	**0.834**

robotic deployments in which the robot surveys or maintains a fixed area over an extended duration.

A. Implementation Details

Dataset We use the LearnLFOdo dataset [35] to provide posed images from a robotic platform. This dataset was captured using a UR5e robotic arm providing accurate ground truth pose information and contains 45 separate scenes and camera trajectories. While the dataset was captured using an EPIImaging light field camera, only the centre image is taken so that the camera is effectively monocular.

Architecture and Optimisation As the most tractable NeRF variant for robotics in terms of training time, we use InstantNGP [36]. We note, however, that our approach could be applied to more recent and higher-fidelity NeRF variants [20], [37]. All models are trained on an NVIDIA 4080, utilising the Adam optimiser and published parameters from the original InstantNGP paper [36]. We train for 5000 iterations allowing the representation to converge, before querying for keyframe addition.

Threshold Parameters For the experiments conducted the threshold values for each of the competing methods was kept constant for all sequences within the dataset. To enable a fair comparison of the results, the thresholds were chosen such that each method selected a similar number of images across all experiments. We selected an adaptive reconstruction threshold $\gamma = 24\,\text{dB}$, time threshold $\gamma_t = 0.183\,\text{s}$, corresponding to sampling every 5.5 frames, and distance threshold $\gamma_d = 0.25\,\text{m}$. The single- and multiple-pass experiments used the same threshold values.

TABLE II: Multiple-pass trajectory performance

Method	Metrics	Seq. 13	Seq. 20	Avg.
Temporal	No. Images	104	134	119
	Total time (s) (↓)	3663.89	4694.17	4179.03
	PSNR (dB) (↑)	23.38	27.62	26.00
	SSIM (↑)	0.777	0.844	0.811
Distance	No. Images	138	187	162.5
	Total time (s) (↓)	4936.28	5963.91	5450.10
	PSNR (dB) (↑)	24.05	28.46	26.79
	SSIM (↑)	0.801	0.855	0.828
Adaptive (Ours)	No. Images	**34**	**51**	**42.5**
	Total time (s) (↓)	**1044.87**	**1544.23**	**1294.55**
	PSNR (dB) (↑)	**24.32**	**28.51**	**26.90**
	SSIM (↑)	**0.811**	**0.862**	**0.837**

B. Single-Pass Performance

We evaluated the methods on six single-pass sequences, with numerical results shown in Tab. I. For each sequence we show the number of keyframes selected, the total time taken to complete the incremental reconstruction process, as well as the reconstruction quality in PSNR and structural similarity (SSIM). We include comparison to a model trained using all available input images. Because this is an estimate and not ground truth data it represents a “gold standard”, i.e. an upper bound on performance based on the best available estimate. Note also that this approach is prohibitively slow, taking hours to run, and is thus incompatible with online operation.

Comparing our adaptive method to the non-adaptive baselines we see that the proposed method is able to select more informative keyframes which allows it to not only produce higher quality reconstructions, demonstrated through higher PSNR and SSIM scores, but it does so with less computational time. Compared to the gold standard our method reduces the input frames by 78% while decreasing PSNR by only 2dB, meaning the quality has not dropped significantly despite the large decrease in input data and computation time.

Fig. 1 shows a graphical representation for the keyframes selected along Seq. 35. Here we see that while the time and distance baselines show similar sampling density in time and space, respectively, the proposed adaptive method recognises when the scene is already well represented and samples more densely where doing so is more informative. Our approach adds few frames towards the middle of the trajectory where the scene is already well described by existing frames in the training set.

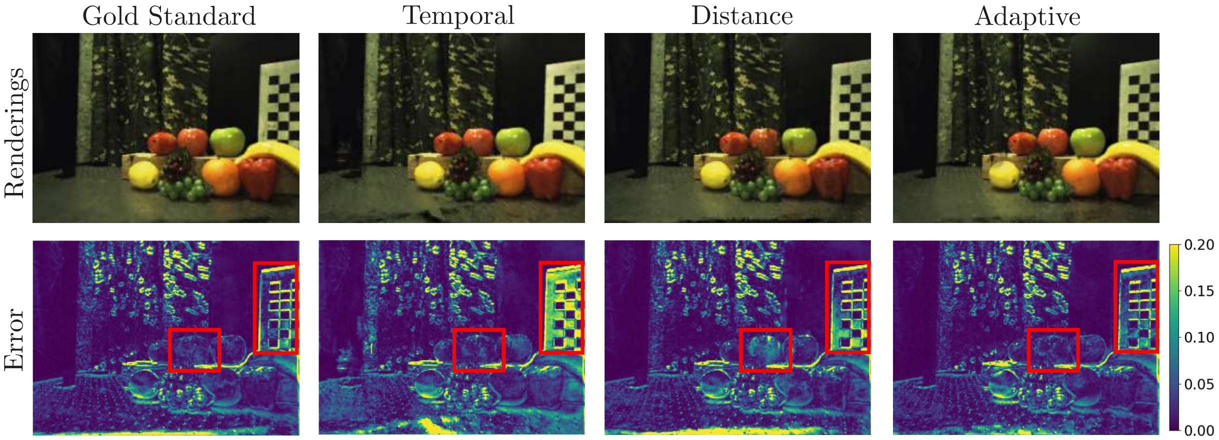

Fig. 2: Reconstruction quality for different keyframe selection methods, taken from Seq. 41. The rendered images along with the error in the rendering compared to the captured image. Areas of high discrepancy between methods are outlined in red.

C. Multiple-Pass Performance

Table II depicts performance for trajectories that revisit parts of the scene multiple times. This is representative of an extended deployment in which a robot completes repeated surveys of an area. The adaptive approach does not add additional frames after the first pass through the environment, recognising that the additional information provided on repeated visits falls below the selected threshold. The non-adaptive baselines are unaware of scene content and so continue to add frames even for well-represented parts of the scene. Compared to the temporal and distance baselines, the adaptive approach uses just 35.7% and 26.2% of the frames respectively, while also resulting in a small improvement to PSNR compared to the naïve approaches. We expect the benefit of the adaptive approach to grow with the duration of the deployment.

D. Qualitative Results

In Figure 2 we show a qualitative comparison between the gold standard approach and the temporal, distance, and adaptive keyframe selection methods. The figure shows mean absolute error in pixel intensity compared with measured frames, for pixel values between 0 and 1.

The figure shows notable improvements in specular regions of the scene like the fruit, which require more views to successfully constrain, as well as planar Lambertian scene regions like the checkerboard. By adaptively selecting frames based on render quality, the proposed method is able to determine regions which require additional input frames to refine appearance. Note that this process leverages the view-dependence of the NeRF, and this would not be possible by employing only depth information or a Lambertian scene assumption.

V. Conclusions

We presented an adaptive technique for selecting NeRF input frames from a video stream by iteratively constructing and querying a NeRF model. Our approach maintains model quality while reducing storage and computation time by as much as 78% in single-pass mapping, and prevents the unbounded growth of input frames in persistent mapping. It outperforms non-adaptive temporal and spatial methods by delivering greater rendering quality in less time.

This work is a step towards complete NeRF-based robotic mapping. We envision as future work employing similar insights to remove outdated frames, or to detect and describe changes in the environment. We further envision incorporating pose refinement as part of the integration of new frames into the model.

References

[1] 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 *ECCV*, 2020.

[2] C.-H. Lin, W.-C. Ma, A. Torralba, and S. Lucey, "BARF: Bundle-adjusting neural radiance fields," in *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[3] 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.

[4] M. Kim, S. Seo, and B. Han, "InfoNeRF: Ray entropy minimization for few-shot neural volume rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 912–12 921.

[5] N. Somraj, A. Karanayil, and R. Soundararajan, "SimpleNeRF: Regularizing sparse input neural radiance fields with simpler solutions," in *SIGGRAPH Asia 2023 Conference Papers*, ser. SA '23. New York, NY, USA: Association for Computing Machinery, 2023.

[6] B. Zhu, Y. Yang, X. Wang, Y. Zheng, and L. Guibas, "VDN-NeRF: Resolving shape-radiance ambiguity via view-dependence normalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 35–45.

[7] K. Park, P. Henzler, B. Mildenhall, J. T. Barron, and R. Martin-Brualla, "CamP: Camera preconditioning for neural radiance fields," *ACM Trans. Graph.*, 2023.

[8] C. Campos, R. Elvira, J. J. Gómez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[9] T. Schops, T. Sattler, and M. Pollefeys, "BAD SLAM: Bundle adjusted direct RGB-D SLAM," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 134–144.

[10] A. Concha, G. Loianno, V. Kumar, and J. Civera, "Visual-inertial direct SLAM," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1331–1338.

[11] A. Chen, Z. Xu, X. Wei, S. Tang, H. Su, and A. Geiger, "Factor Fields: A Unified Framework for Neural Fields and Beyond," *arXiv preprint arXiv:2302.01226*, 2023.

[12] Y. Xie, T. Takikawa, S. Saito, O. Litany, S. Yan, N. Khan, F. Tombari, J. Tompkin, V. Sitzmann, and S. Sridhar, "Neural fields in visual computing and beyond," in *Eurographics STAR*, 2021.

[13] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[14] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[15] A. Rosinol, J. J. Leonard, and L. Carlone, "Probabilistic volumetric fusion for dense monocular SLAM," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 3097–3105.

[16] ——, "NeRF-SLAM: Real-time dense monocular SLAM with neural radiance fields," *arXiv preprint arXiv:2210.13641*, 2022.

[17] K.-N. Lianos, J. L. Schonberger, M. Pollefeys, and T. Sattler, "VSO: Visual semantic odometry," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 234–250.

[18] K. Tateno, F. Tombari, I. Laina, and N. Navab, "CNN-SLAM: Real-time dense monocular slam with learned depth prediction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6243–6252.

[19] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-NeRF 360: Unbounded anti-aliased neural radiance fields," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[20] ——, "Zip-NeRF: Anti-aliased grid-based neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023, pp. 19 697–19 705.

[21] J. Ichnowski, Y. Avigal, J. Kerr, and K. Goldberg, "Dex-NeRF: Using a Neural Radiance Field to Grasp Transparent Objects," in *Conference on Robot Learning*. PMLR, 2022, pp. 526–536.

[22] Q. Zhou, M. Maximov, O. Litany, and L. Leal-Taixé, "The NeRFect match: Exploring NeRF features for visual localization," *arXiv preprint arXiv:2403.09577*, 2024.

[23] 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 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[24] O. Kwon, J. Park, and S. Oh, "Renderable neural radiance map for visual navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9099–9108.

[25] Z. Zhu, S. Peng, V. Larsson, Z. Cui, M. R. Oswald, A. Geiger, and M. Pollefeys, "Nicer-SLAM: Neural implicit scene encoding for RGB SLAM," in *3DV*, 2024.

[26] 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.

[27] J. Ortiz, A. Clegg, J. Dong, E. Sucar, D. Novotny, M. Zollhoefer, and M. Mukadam, "iSDF: Real-time neural signed distance fields for robot perception," in *Robotic Science and Systems*, 2022.

[28] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "NICE-SLAM: Neural Implicit Scalable Encoding for SLAM," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 786–12 796.

[29] R. Griffiths, J. Naylor, and D. G. Dansereau, "NOCaL: Calibration-free semi-supervised learning of odometry and camera intrinsics," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 4056–4062.

[30] J. Naumann, B. Xu, S. Leutenegger, and X. Zuo, "NeRF-VO: Real-time sparse visual odometry with neural radiance fields," *arXiv preprint arXiv:2312.13471*, 2023.

[31] H. Zhu, T. He, X. Li, B. Li, and Z. Chen, "Is vanilla MLP in neural radiance field enough for few-shot view synthesis?" *arXiv preprint arXiv:2403.06092*, 2024.

[32] J. Yang, M. Pavone, and Y. Wang, "FreeNeRF: Improving few-shot neural rendering with free frequency regularization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8254–8263.

[33] K. Deng, A. Liu, J. Zhu, and D. Ramanan, "Depth-supervised NeRF: Fewer views and faster training for free," *CoRR*, vol. abs/2107.02791, 2021.

[34] M. Tancik, V. Casser, X. Yan, S. Pradhan, B. Mildenhall, P. Srinivasan, J. T. Barron, and H. Kretzschmar, "Block-NeRF: Scalable large scene neural view synthesis," *arXiv*, 2022.

[35] S. T. Digumarti, J. Daniel, A. Ravendran, R. Griffiths, and D. G. Dansereau, "Unsupervised learning of depth estimation and visual odometry for sparse light field cameras," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 278–285.

[36] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (ToG)*, vol. 41, no. 4, pp. 1–15, 2022.

[37] D. Verbin, P. Hedman, B. Mildenhall, T. Zickler, J. T. Barron, and P. P. Srinivasan, "Ref-NeRF: Structured View-Dependent Appearance For Neural Radiance Fields," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022, pp. 5481–5490.