Benchmarking Implicit Neural Representation and Geometric Rendering in Real-Time RGB-D SLAM

Tongyan Hua[1] Lin Wang[1,2*]
[1]AI Thrust, HKUST(GZ) [2]Dept. of CSE, HKUST
t.hua.msc@outlook.com, linwang@ust.hk

Project Page: https://vlis2022.github.io/nerf-slam-benchmark/

Abstract

Implicit neural representation (INR), in combination with geometric rendering, has recently been employed in real-time dense RGB-D SLAM. Despite active research endeavors being made, there lacks a unified protocol for fair evaluation, impeding the evolution of this area. In this work, we establish, to our knowledge, the ***first*** *open-source benchmark framework to evaluate the performance of a wide spectrum of commonly used INRs and rendering functions for mapping and localization. The goal of our benchmark is to 1) gain an intuition of how different INRs and rendering functions impact mapping and localization and 2) establish a unified evaluation protocol w.r.t. the design choices that may impact the mapping and localization. With the framework, we conduct a large suite of experiments, offering various insights in choosing the INRs and geometric rendering functions: for example, the dense feature grid outperforms other INRs (*e.g. *tri-plane and hash grid), even when geometric and color features are jointly encoded for memory efficiency. To extend the findings into the practical scenario, a hybrid encoding strategy is proposed to bring the best of the accuracy and completion from the grid-based and decomposition-based INRs. We further propose explicit hybrid encoding for high-fidelity dense grid mapping to comply with the RGB-D SLAM system that puts the premise on robustness and computation efficiency.*

1. Introduction

Simultaneous Localization and Mapping (SLAM) is a pivotal task in 3D computer vision, with the goal of estimating the position and orientation of a sensor, while concurrently building a map of the surrounding scene. For the real-time dense visual SLAM system, a large number of methods have been proposed, predominantly based on the RGB-D cameras [8, 23, 32, 36, 42, 43, 45].

Neural Radiance Field (NeRF) is an emerging technique based on the *Implicit Neural Representation (INR)*, in combination with *geometric rendering* for novel view synthesis [28]. It employs a Multilayer Perceptron (MLP) to map a 3D point (*i.e.*, spatial location along viewing direction) to density and color. Various NeRF variants have emerged since, featuring unique representations such as hash-grid [29] and tri-plane [4], or focusing on mapping the 3D point to different geometric properties, such as the sur-

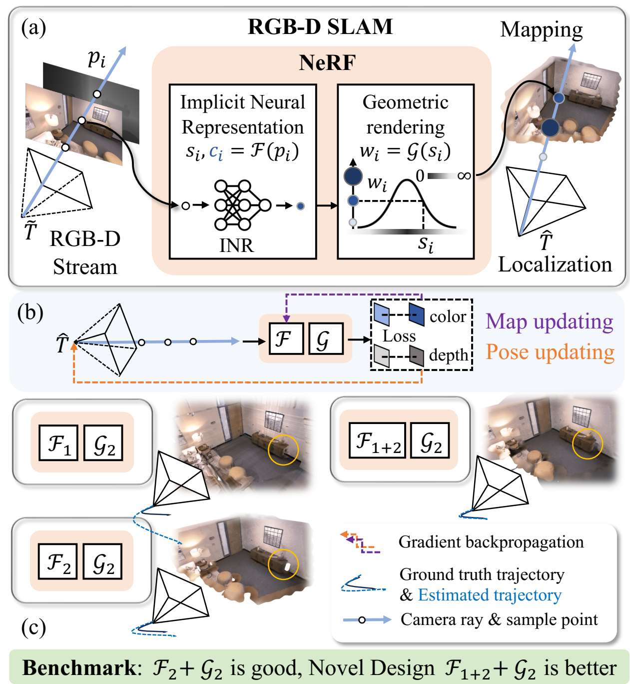

Figure 1. (a) We establish a novel benchmark to evaluate different elements of NeRF, narrowly defined as a combination of INR function $\mathcal{F}$ and geometric rendering function $\mathcal{G}$, under the unified RGB-D SLAM paradigm. (b) Rendering Loss guides the online updating of the pose from $\tilde{T}$ to $\hat{T}$, and parameter of $\mathcal{F}$. (c) A toy example illustrates the impact of various combinations of $\mathcal{F}$ and $\mathcal{G}$: $\mathcal{F}_2$ surpasses $\mathcal{F}_1$ in trajectory estimation and reconstruction fidelity but compromising completeness, inspire new designs that bring the benefits of $\mathcal{F}_1$ and $\mathcal{F}_2$ to form a hybrid encoding $\mathcal{F}_{1+2}$.

*Corresponding author.

face [7, 33, 41, 47, 51], while developing corresponding geometric rendering strategies. NeRF effectively models the intrinsic structure of a specific scene, capturing its geometry in a compact yet expressive manner that aligns closely with observed locations. As a result, it inherently incorporates the corresponding observation positions. This feature facilitates the deduction of the camera poses directly from the trained NeRF [2, 5, 15, 22, 48].

This has inspired active research endeavors, integrating NeRF with RGB-D SLAM, demonstrated by the increasing number of publications [6, 13, 17, 27, 34, 38, 40, 46, 50, 52, 54]. In general, these methods can be classified into: 1) NeRF-centric methods, where NeRFs are used for both scene reconstruction and pose estimation [17, 38, 40, 46, 54], as depicted in Fig. 1(a) and (b), and 2) SLAM-centric methods, where the location is provided by other SLAM systems [6, 27, 34, 50, 52]. Although tremendous efforts have been made to reconstruct high-fidelity scenes and improve pose estimation, several limitations persist with the active progress of research:

L1: The Absence of a Unified and Comprehensive Benchmark Framework. This hampers the comparison of different NeRFs within the RGB-D SLAM system. State-of-the-art (SOTA) NeRF-SLAM methods [17, 35, 38, 40, 46, 54] usually exhibit a variety of strategies regarding components other than the INR and rendering, *e.g.*, how training data is selected (refer to as keyframe selection in SLAM [54]), This makes it hardly possible to directly compare systems and capture the actual progress stemming from the NeRFs' design. Consequently, it is imperative to understand the individual characteristics of NeRF when designing the SLAM system for varied purposes.

L2: Lack of Assessment of NeRF Component Variations on SLAM Performance. As depicted in Fig. 1(c), NeRF, defined by $\mathcal{F}$ and $\mathcal{G}$. has many variants that can significantly affect the performance of SLAM systems. The network architecture choice of $\mathcal{F}$, for example, is vital for learning scene representations accurately and efficiently. Some architectures, *e.g.*, joint color and geometry encoding of Tri-plane, can quickly converge but may lose important details. This influences the precision of mapping and tracking within NeRF-enhanced SLAM systems. Moreover, the rendering methods used to integrate geometry and color information along rays are also critical, since the quality of rendered pixels directly influences the pose and map updating, as Fig. 1(b). While high-accuracy methods [33, 41, 47, 51] improve the rendering quality, they generally increase computational demands and may not perform well with less accurately estimated poses.

In this paper, we establish, to our knowledge, the **first** open-source benchmark framework to evaluate the performance of a wide spectrum of commonly used $\mathcal{F}$ and $\mathcal{G}$ for examining their effectiveness of mapping and localization. The major contributions are summarized as follows:

C1: Comprehensive Evaluation of NeRFs within a Unified RGB-D SLAM Framework. We propose a novel RGB-D SLAM benchmark framework, featuring a unified evaluation protocol to assess different NeRF components effectively. We unfold our benchmark from a NeRF-centric paradigm. It covers five major variables that categorizing into two categories in Fig. 2, *i.e.*, the unified SLAM framework (including uniformed implemented sampling, training, and keyframe selection) and the NeRFs as a combination of $\mathcal{G}$ and $\mathcal{F}$. Our main objective is to investigate how the NeRFs influence the SLAM performance under uniformly controlled configurations (Sec. 3.2) in two established scenarios: lab and practical scenarios (Sec. 3.4).

C2: Pivotal Insights and Derived New Designs. We reveal the significant differences in the performance of INRs in RGB-D SLAM problems attributable to their structural paradigms. Specifically, hybrid representations that simplify the 4D feature space, such as hash grid and tri-plane, often require separate encoding of geometry and appearance to achieve optimal performance, whereas representations of complete forms, such as dense grid and pure MLP, do not necessitate this. We also discovered that, in datasets with complete trajectory loops, *e.g.* Replica dataset [37], grid-based INRs (hash grid and dense grid) show better performance, while decomposition-based INRs (tri-plane and factorization) exhibit superior efficacy under random, loop-free trajectories, *e.g.* the sequences in [1]. This phenomenon inspired us to propose a novel blending of grid-based and decomposition-based methods.

C3: Bags of Engineering Tricks for the Extension to Mapping. Our research demonstrates that a strategic blend of dense grid representation, initially introduced in [54], and appropriate geometric rendering functions can not only surpass more recently proposed sparse alternatives [40], but also surpass its application in earlier SLAM systems [54]. This finding suggests that our new SLAM framework successfully capitalizes on a suite of the latest engineering techniques. Our benchmark also indicates that minimal feature dimensions are sufficient for achieving relatively high-quality mapping and tracking. This enables us to achieve extremely fine spatial partitioning without a substantial increase in memory usage. Based on this, we have effectively re-engineered the dense grid – a scene representation known for its high memory demand but exceptional leaderboard performance – for real-time high-fidelity mapping.

2. Related Works

NeRF-centric approaches stem from NeRF's inherent potential for pose estimation [2, 5, 15, 22, 48]. The pioneering study [38] showcased that the foundational NeRF model [28] could act as the sole representation for concurrent localization and mapping. This spurred subse-

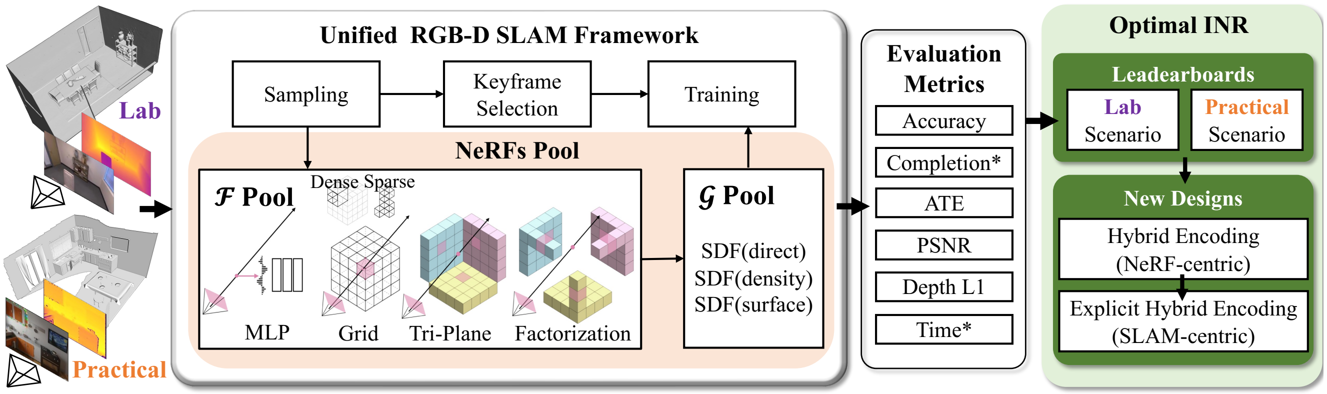

Figure 2. The proposed pipeline for NeRF-SLAM benchmark. The asterisk * indicates the existing two values for evaluation.

quent research that illustrated the advantages of hybrid representations [17, 40, 46, 55] over singular MLP structures. Predominantly, dense SLAM methods have relied on RGB-D inputs to accelerate convergence on sampling distributions. Yet, recent efforts have ventured into dense SLAM with RGB-only inputs, where multi-level dense feature grids have yielded impressive results, whether an external depth estimator is used [54] or not [20]. Another development is the shift in volume density representation from occupancy [20, 35, 38, 54] to Signed Distance Fields (SDF) [17, 40, 46, 55], with the latter demonstrating rapid convergence and superior reconstruction quality. Furthermore, these methods have laid the groundwork for semantic tasks, enhancing scene understanding and enabling real-time reconstruction [12, 18], and have even shown promise in style transfer applications [49].

SLAM-centric methods were developed to leverage NeRF as an external module within a self-content SLAM system, aimed at achieving the robustness similar to well-acknowledged visual SLAM systems [3, 11, 19, 30, 31]. Initial adoption of hash-based sparse parametric encoding [29] was favored for its memory efficiency and quick convergence [6, 34]. Some recent studies have begun to re-explore the superiority of purely MLP-based spatial representations [21, 25]. The potential for dense reconstruction and effective point cloud compression has led to a focus on neural implicit mapping using posed RGB-D observations [10, 16, 24, 53], and even solely from posed RGB inputs [13] in contexts such as robotics and autonomous driving. Further, some research has expanded these methods for practical applications, addressing challenges in large-scale mapping and multi-robot mapping fusion [26, 39, 44].

3. NeRF-SLAM Benchmark

Problem Formulation. Given a stream of synchronized RGB-D input frames I, D_t at timestamp t, the color and depth of a pixel x are represented by c_x and d_x, respectively. Along the camera ray passing through pixel x, we sample N points, each associated with a specific sample p_i at a distance d_i. A learnable neural implicit function $F(\cdot)$ is then employed to predict the appearance c_i and geometric properties g_i of each sample:

$$(c_i, g_i) = F(p_i) \tag{1}$$

To determine the weight w_i for each sample along a ray, we employ a geometric rendering function $G(\cdot)$:

$$w_i = G(g_i) \tag{2}$$

The color and depth can be estimated as:

$$\tilde{c}_x = \sum_{i=1}^{N} w_i \cdot c_i, \tilde{d}_x = \sum_{i=1}^{N} w_i \cdot d_i \tag{3}$$

We formulate NeRF-SLAM as a continuous online learning task. The training data, *i.e.*, the sampled rays through the pixel x, are cached, for the continuous optimization of $F(\cdot)$ and camera pose T_t. This process adheres to the fundamental photometric e_x^p and geometric e_x^g constraints:

$$e_x^p = \tilde{c}_x - c_x, e_x^g = \tilde{d}_x - d_x \tag{4}$$

3.1. NeRFs for RGB-D SLAM

To address the RGB-D SLAM problem, our objective is to formulate the neural implicit function $F(\cdot)$ and the geometric function $G(\cdot)$. As such, they can efficiently approximate the true density distribution along a camera ray, thereby providing an accurate estimation of w_i:

$$w_i = G(F_{geo}(p_i)), F \in \mathcal{F}, G \in \mathcal{G} \tag{5}$$

Denoting learnable geometric and appearance implicit neural functions $\{F_{\text{geo}}, F_{\text{app}}\} \in F$, consolidating choices from prominent baselines [17, 38, 40, 46, 52, 54], we firstly unified the formulation of F_{geo} in Tab. 1, laying the groundwork for defining $G(\cdot)$.

A key benefit of employing the SDF over alternative representations, such as occupancy grids, lies in its capability to leverage per-point losses across all samples and losses from the rendered image. This contributes to the model's fast convergence, a finding validated by [17, 40]. Despite occupancy grids' efficacy across various domains, particularly in robotics, this paper focuses on the high-fidelity reconstruction and accelerated convergence offered

$\mathcal{F}$	$F_{geo} \in \mathcal{F}$
MLP^1	$(g_i, h_i) = MLP(\gamma(p_i))$.
Grid^2	$(g_i, h_i) = MLP(\Phi^*_{dense}(p_i), \gamma(p_i))$. $(g_i, h_i) = MLP(\Phi^*_{hash}(p_i), \gamma(p_i))$.
Decomposition^2	$(g_i, h_i) = MLP(\Phi^*_{tri}(p_i), \gamma(p_i))$, $(g_i, h_i) = MLP(\Phi^*_{fac}(p_i), \gamma(p_i))$.
$\mathcal{G}$	$G \in \mathcal{G}$
SDF (direct)^4	$w_i = sig(\frac{g_i}{tr}) \cdot sig(-\frac{g_i}{tr})$.
SDF (density)^5	$w_i = exp(-\sum_{k=1}^{i-1} \sigma_i)(1 - exp(-\sigma_i))$, $\sigma_i = \beta \cdot sig(-\beta \cdot g_i)$.
SDF (surface)^6	$w_i = \alpha_i \prod_{j=1}^{i-1}(1 - \alpha_j)$, $\alpha_i = \max\left(\frac{sig(g_i) - sig(g_{i+1})}{sig(g_i)}, 0\right)$.

Table 1. The selection of $F_{geo}(\cdot)$ and $G(\cdot)$. * denotes the existence of multi-resolution spatial splits, and $\gamma(\cdot)$ represents the positional encoding function, and sig refers to sigmoid function. $\mathcal{F}^1$ represents a pure multi-layer perceptron [38]; $dense$ and $hash$ in $\mathcal{F}^2$ refers to the dense [54] and hash [40] feature grid encoding, respectively; tri and fac in $\mathcal{F}^3$ denote tri-plane [17] and factorization [14] encoding, respectively. tr in $\mathcal{G}^4$ stands for truncation of SDF [40, 46]; the β in $\mathcal{G}^5$ stands a learnable parameter [17], and the $\mathcal{G}^6$ is originate from [41]and adopted by [52].

by SDFs. This focus aligns with that of the recent SOTA methods [17, 55]. Thus, discussions on the occupancy are excluded, with g_i interchangeably referred to as s_i. In the following, we delineate three structural paradigms for the appearance functions corresponding to each $\mathcal{F}$:

$$c_i = \begin{cases} MLP(\Phi^*(p_i), \gamma(p_i)), & \text{if coupled(base)} \\ MLP(\Phi^*(p_i), \gamma(p_i), h_i), & \text{if coupled} \\ MLP(\phi^*(p_i), \gamma(p_i)). & \text{if decoupled} \end{cases} \tag{6}$$

The distinction between coupled and decoupled structures hinges on the encoding of color: it is either independently encoded by ϕ or jointly encoded with geometry by Φ. We further categorize the coupled structure into two types based on the presence of channeled geometric features.

In examining hybrid representations that incorporate structural elements beyond the standard MLP, we observe a diversity of configurations w.r.t. the resolution levels and feature dimensions. For example, some studies assign high-dimensional features to spatial partitions, *e.g.*, 32 dimensions as in [17, 54] and 16 in [46], alongside relatively low-resolution levels (3 in [54], 2 in [17], and 1 in [46]). Conversely, other studies utilize extremely low-dimensional features (2 dimensions in [20, 40]) while significantly increasing the number of multi-resolution levels (6 in [20] and 16 in [40]). Also, the settings for resolutions vary widely: the finest granularity for appearance embeddings is set at 16cm in [54] and 3cm in [17], with the broadest level at 200cm in [54] and 24cm in [17]. In this work, we ensure a fair comparison of these representations by controlling the feature dimensions and the number of resolution levels. This minimizes the computational overhead (*i.e.*, opting for 2 resolution levels and 2 feature dimensions in constructing the leaderboard). *Detailed visual descriptions for the impacts of feature dimension are available in the supplmat.*

While the sampling strategy is an ineligible component in NeRF, this paper does *not consider* it a variable for examination, due to the substantial simplification of the sampling process, afforded by the incorporation of additional depth input. Therefore, we leverage the off-the-shelf sampling strategy for a unified SLAM framework in the following section, based on the validated techniques proposed in the latest research [38, 54, 55].

3.2. Unified Evaluation Protocol

In this section, we describe the proposed unified SLAM framework for evaluating the possible combinations, detailed in Tab. 1 and Eq. (6). The schematic diagram in Fig. 2 delineates the three primary components of our framework, namely sampling, keyframe management, and training.

Training is continuously conducted in real-time to optimize objectives that include common photometric and geometric losses in accordance with Eq. (4), as well as SDF loss and free space suppressing loss that are used in previous SDF-based INR-SLAM [17, 40]. Detailed formulations are available in the *supplmat*.

Sampling. For pixels with available depth, we utilize stratified sampling, dividing the samples into two categories: surface and free space. Surface samples are densely placed around the ground truth depth to accurately capture the surface details within the truncation range. Free space samples, on the other hand, are evenly distributed along the ray. In cases where ground truth depth is absent, we evenly allocate samples. This method highlights the fine details of surfaces, making the use of an $L2$ loss more effective than the $L1$ loss traditionally used in earlier studies [38, 46, 54].

Keyframe Selection. Enhanced pose estimation and consistent global reconstruction can be achieved with a global bundle adjustment strategy, as suggested in [40, 52]. For better storage and retrieval efficiency, we follow the approach in [40], selectively caching only key sampled rays from keyframes for optimization purposes.

3.3. Evaluation Metrics

For overall performance, We evaluate the final reconstruction quality using three established metrics that are commonly used in INR-based RGB-D SLAM [17, 38, 40, 54]: Accuracy(*Acc.*[cm]), Completion(*Comp.*[cm]), and Completion Ratio that gauges the proportion of extracted meshes that of Completion value smaller than 5cm (*Comp.*[%]). For the sampled points on the reconstructed mesh and the ground truth mesh, the Accuracy and Completion metrics are calculated by determining the average distance from the former to the latter, and from the latter to the former, respectively. Prior to metric calculation, mesh culling is conducted in line with the procedure outlined in [1]. The overall accuracy of the camera trajectory is quantified by the Root Mean Square Error (*ATE*[cm] *RMSE*).

For procedural performance, we assess the Peak Signal-to-Noise Ratio (*PSNR*[db]) and L1 term of the estimated depth (*Depth L1*[cm]) throughout the entire sequence by comparing their mean values, since SLAM additionally emphasizes the continual estimation performance, unlike static 3D reconstruction tasks. The efficiency of mapping and tracking is evaluated by the average update time per input frame over the whole sequence.

3.4. Scenario Settings

As illustrated in Fig. 2, our goal is to incrementally unveil the capabilities of INRs for RGB-D SLAM by implementing a unified evaluation protocol. Thus, it is possible to identify the best choices for subsequent refinements. To accommodate the extensive range of evaluation metrics and inherent complexity of SLAM systems, we establish two distinct scenarios, each with a specific evaluative emphasis, enhancing the modularity and clarity of our analysis:

Lab Scenario. This scenario aims to gain comprehensive insights from a detailed leaderboard, see Tab. 2, based on the synthetic Replica dataset [37]. The initial evaluation centers on the structural paradigm of INRs, with the best-performed one being adopted for subsequent benchmark analysis. Then, we construct the main leaderboard using eight synthetic sequences from the Replica dataset. Each sequence includes ground truth trajectories, guaranteeing thorough coverage of the entire room.

Practical Scenario. In contrast to simulated datasets, real-world data are subject to noise. Also, camera trajectories often only capture the scenes partially, posing distinct challenges for INR-based RGB-D SLAM systems. For this practical scenario, we evaluate seven sequences from the synthetic NeuralRGBD dataset [1], which are crafted to mimic the noise and artifacts characteristic of real-world depth sensors. Intentionally, the camera trajectories within these experiments were designed to scan only portions of the scenes. The leaderboard results are shown in Tab. 4. Notably, the metrics of completeness and accuracy typically used for full-scene reconstruction are less definitive in the context of partial observations. Nonetheless, prior research suggests that increased completeness correlates with improved pose estimation. Therefore, in this scenario, the reconstruction quality (*Acc.* and *Comp.*) act only as indicators for the SLAM performance. We instead place significant emphasis on performance metrics such as *ATE*, *Depth L1*, and *PSNR*.

Note that, due to substantial variations in sequence lengths, it introduces considerable variability in average frame processing time. Therefore, we remove the tracking and mapping speed metrics from the practical scenario leaderboard for more transparent comparisons. The full records of time measurement can be found in the *supplmat*.

3.5. Optimal INR Designs

We concentrate on the **NeRF-centric** paradigm to showcase two distinct scenarios in our leaderboard. It provides critical insights for developing an optimal INR. *We will introduce our new design in conjunction with the discussion in the experimental section.*

From the **SLAM-centric** viewpoint, NeRF shows promise as a dual-purpose tool for localization and mapping. However, its effectiveness is somewhat limited in applications, such as robot navigation. This domain necessitates a SLAM system that is not only *robust* but also capable of *rapid convergence*, particularly in environments with ambiguously defined scene boundaries. To address these limitations, we propose to *integrate the insights from the NeRF-centric leaderboard into SLAM-centric paradigms*. We provide a qualitative evaluation (see Fig. 5) on Scannet Dataset [9], demonstrating its adaptability and enhanced performance within the context of SLAM-centric methods.

4. Experiments and New Designs

In this section, we firstly observe the impact of various combinations of NeRF components on SLAM system performance under a Lab scenario. Subsequently, we examine whether the system's performance alters in a practical scenario. Finally, based on the analysis of these empirical observations, which are displayed as leaderboards in Sec. 4.1, we propose new NeRF designs suitable for different SLAM scenarios in Sec. 4.2. Please refer to the *supplmat* for implementation specifications, including parameters and platform configurations. Note that 'Tri-plane' and 'Factorization' are denoted as Tri. and Fact. in this section, respectively.

4.1. Benchmark Leaderboard

Lab Scenario. We begin by examining the impact of structural paradigms, which are commonly presumed but not often analyzed within the current NeRF-SLAM paradigm. In line with Eq. (6), we list the results in Tab. 3, focusing on the *initialization phase*, where INRs are trained using a single posed RGB-D frame. It can be seen that coupling geometric and appearance features boosts depth estimation within a limited number of iterations for MLP, dense, and sparse representations, suggesting a faster SDF convergence. MLP notably shows about a 25% reduction in *Depth L1* loss when adopting a coupled structure. However, the coupled paradigm seems to compromise color rendering in dense (0.91dB ↓) and Sparse (1.97dB ↓) methods. On the contrary, decomposition methods (Tri. and Fact.) benefit from decoupled structures, showing both enhanced geometric (0.08cm and 0.12cm ↓ in *Depth L1* loss) and appearance

$\mathcal{G}$	$\mathcal{F}$	From results				From processes			
		Acc.[cm]↓	Comp.[cm]↓	Comp.[%]↑	ATE[cm]↓	PSNR[db]↑	Depth L1[cm]↓	Tracking[ms]↓	Mapping[ms]↓
SDF(direct)	MLP	11.95	8.36	76.82	14.41	24.20	3.12	293	421
	Dense[1]	**1.65**	5.62	**83.93**	**1.37**	27.88	**1.50**	288	**286**
	Sparse[3]	1.76	5.66	83.61	1.40	**28.23**	**1.65**	**197**	**300**
	Tri.	1.69	5.64	83.60	1.42	27.52	1.80	352	820
	Fact.	1.69	5.60	83.69	1.50	27.52	1.74	419	911
SDF(density)	MLP	9.64	9.92	72.22	24.58	23.51	7.01	250	419
	Dense[2]	**1.60**	**5.58**	**84.01**	**1.31**	27.77	4.42	253	585
	Sparse[4]	1.69	5.65	83.72	1.40	**28.10**	4.45	**207**	310
	Tri.[5]	1.80	**5.59**	83.82	1.49	27.55	4.51	376	829
	Fact.	1.75	5.60	83.73	1.55	27.54	4.48	414	897
SDF(surface)	MLP	30.44	24.28	20.21	44.68	17.13	98.24	342	585
	Dense	32.83	20.71	42.28	135.39	16.21	176.15	418	669
	Sparse	48.22	25.73	30.73	176.68	16.49	174.94	258	359
	Tri.	30.28	18.06	41.77	87.71	16.05	180.68	490	897
	Fact.	30.75	20.45	31.31	85.63	16.43	168.81	584	1009

Table 2. **The leaderboard of lab scenario**. Text in **bold** indicates the best performance, while text in **blue bold** denotes the second best. $^{1-5}$ indicated the top 5 combinations according to the counting of performance ranking.

$\mathcal{F}$	Structure	PSNR[db] ↑	Depth L1[cm] ↓	Time[s] ↓
MLP	Coupled(base)	23.03	2.19	**65.59**
	Coupled	**23.76**	**1.68**	82.00
	Decoupled	6.37	2.26	106.7
Dense	**Coupled(base)**	30.21	**0.83**	**47.55**
	Coupled	29.66	0.84	93.00
	Decoupled	**31.12**	0.85	85.76
Sparse	Coupled(base)	28.28	0.97	49.31
	Coupled	28.45	**0.92**	70.27
	Decoupled	**30.42**	0.99	**32.72**
Tri.	Coupled(base)	26.41	1.17	**61.78**
	Coupled	25.45	1.16	120.40
	Decoupled	**27.95**	**1.08**	83.22
Fact.	Coupled(base)	26.34	1.19	121.10
	Coupled	25.58	1.28	127.40
	Decoupled	**28.36**	**1.07**	**85.83**

Table 3. **Impact of network architectures at initialization**: performance comparison on Replica Room0 sequence, where Time[s] indicates the total processing time in seconds. All outcomes correspond to the geometric function SDF (direct).

(1.54dB and 2.02dB ↑) accuracy compared to their coupled counterparts.

The structure with the most top-ranked instances (highlighted in **bold**) in Tab. 3 is selected as the optimal choice for each INR ($\mathcal{F}$) and is used to construct the following leaderboards. We assess the collective performance of $\mathcal{F}$ and $\mathcal{G}$ within the unified SLAM framework, with results for the Lab scenario detailed in Tab. 2.

Overall, hybrid representations (*i.e.*, $\mathcal{F}$ other than MLP) demonstrate markedly superior color and geometry estimation performance, with decomposition methods (Tri. and Fact.) generally lagging behind grid-based ones (dense and sparse) in terms of processing speed, shown in both Tab. 3 and Tab. 2. Despite using a coupled network structure that halves the total optimization parameters, *the dense grid representation excels*. It achieves overall six top and three second-place rankings. This is *followed by sparse encoding*, which secures two top spots and four second-place, with the Tri. claiming one second-place spot. It is worth noting that the dense grid's concatenated 4D features consistently yield excellent SDF results (with first and second rankings in both *Acc.* and *Depth L1*) and hold their own in RGB quality (marginally lower than the 8D feature-encoded sparse grid) across the SLAM process while maintaining the quickest mapping speed.

$\mathcal{G}$	$\mathcal{F}$	Indicators				Targets	
		Acc.↓ [cm]	Comp.↓ [cm]	Comp.↑ [%]	ATE↓ [cm]	PSNR↑ [db]	Depth L1↓ [cm]
SDF(direct)	MLP	4.37	5.16	79.22	3.69	22.56	3.56
	Dense	2.69	4.69	83.45	1.96	25.15	**1.70**
	Sparse	2.84	4.81	82.64	2.12	**25.23**	1.84
	Tri.	2.29	**4.42**	**84.01**	1.89	24.67	1.85
	Fact.	2.62	**4.47**	83.54	1.94	24.69	1.87
	Hybrid	2.40	4.64	83.48	**1.86**	**25.25**	**1.68**
SDF(density)	MLP	3.98	5.12	78.82	3.87	22.56	4.80
	Dense	2.70	4.72	83.27	**1.87**	25.04	4.40
	Sparse	2.84	4.71	82.79	1.96	25.08	4.46
	Tri.	**2.12**	4.62	**83.90**	1.90	24.62	4.42
	Fact.	**2.11**	4.45	**84.01**	2.01	24.63	4.41
	Hybrid	2.34	4.71	83.25	1.91	25.05	4.40

Table 4. **The leaderboard of practical scenario**. Black bold text indicates top performance, blue marks second place, with an additional underline denotes third rank for the targeted metric. Note that the hybrid denotes our proposed encoding strategy that synergizes the strengths of dense grid and tri-plane.

Notably, the neural implicit surfaces [41] rendering method shows significantly inferior performance than its more naive counterparts [1], which also shares the unbiased approximation but directly maps SDF to weighting factors rather through volume rendering. This finding might suggest the delicate formulation for occlusion in offline 3D volume rendering might be sensitive to relatively poorly estimated poses in the online SLAM.

Practical Scenario. When adopting synthetic data with noisy depth and incomplete scene coverage, decomposi-

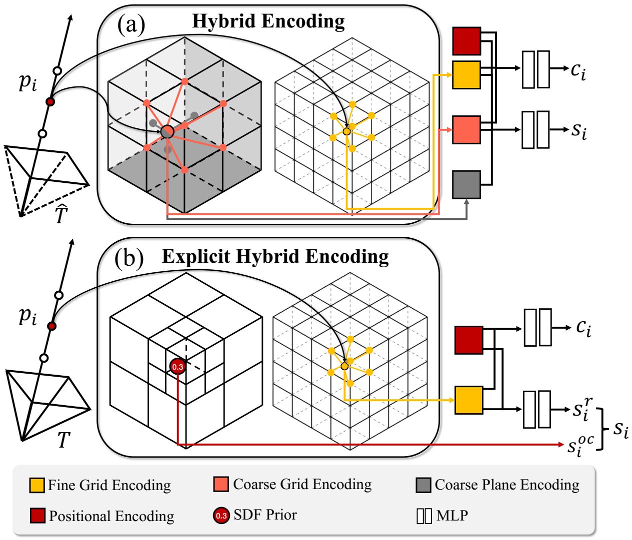

Figure 3. **Illustration of new designs.** For hybrid encoding, a point p_i is (a) encoded using feature planes and a feature grid at a coarse level, and exclusively by a feature grid at a fine level. In contrast, for explicit hybrid encoding, p_i is (b) solely encoded with an optimizable fine-level feature grid and decoded by MLP into an SDF residual s_i^r and color c_i. This residual is then combined with the SDF prior stored in an explicit octree s_i^{oc} to derive the inferred SDF value s_i.

tion methods excel in achieving high geometric accuracy (*Acc.*) and scene completion (*Comp.*) Tab. 4. This verifies our statement in Sec. 3.4. The decomposition of 3D implicit spaces into mutually orthogonal 2D (additional 1D for the factorization method) results in planar geometries akin to real-world indoor environments, extending even beyond the observed view frustum. In these instances, more points might align closely with the true spatial geometry.

Yet, this increase in scene completion doesn't directly improve targeting performance metrics. Specifically, *dense representation still performs better* in position accuracy (*ATE*) and depth estimation (*Depth L1*), compared to the decomposition methods, especially when combined with SDF(direct). In the realm of color estimation, dense grid exhibit only a marginal drop in accuracy (0.08dB↓) compared to their sparse counterparts. This trend is consistent with the Lab scenario leaderboard Tab. 3, where both grid-based methods notably surpass the performance of decomposition methods in *ATE*, *Depth L1* and *PSNR*.

4.2. New Designs for Encoding

Hybrid Encoding. NeRF-centric approaches estimate camera trajectories by freezing the parameters of $\mathcal{F}$ and setting 6-DOF poses as optimizable parameters. Therefore, effective modelling of spatial geometry and color is beneficial for pose estimation. Decomposition-based methods encode scenes in ways that echo the real three-dimensional world, particularly within artificial indoor settings. This characteristic strengthens their capacity to project unobserved regions accurately, allowing for sturdier pose estimates even when faced with occasional inaccuracies in observed segments of the scene. However, their precision falls short of grid-based methods, leading us to consider combining the strengths of both approaches. For this reason, we introduce *hybrid encoding, a strategy blending the comprehensive spatial depiction of decomposition with the precision of grid-based methods*, as depicted in Fig. 3 (a). This strategy complies with our leaderboard protocol, with both feature dimensions and resolution levels set at two. At the coarse resolution level, a sample point undergoes both bilinear and trilinear interpolation within the respective feature plane and grid. At the fine resolution level, interpolation is exclusively performed using a feature grid. Outputs from both scales are then combined to form the input for MLP decoders for estimating raw color and geometry.

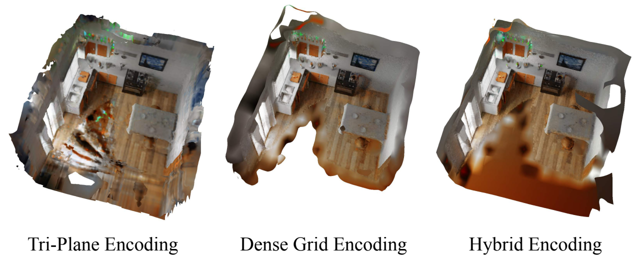

Figure 4. Reconstruction of '*morning apartment*' sequence on the NeuralRGBD dataset, Our hybrid encoding strategy brings the best of two worlds.

The quantitative and qualitative results are shown in Tab. 4 and Fig. 4 for the dense grid feature encoding. It is revealed that hybrid encoding achieves superior trajectory estimation accuracy and reconstruction fidelity, both in color and depth. This is mainly due to the enhanced completeness augmented by plane-based representation, as confirmed by various indicators, *i.e.*, increased performance for *Acc.* and *Comp.*, and visual demonstration in Fig. 4.

Notably, our benchmark strictly controls feature dimensions and spatial resolution for the implicit encoding. However, optimal performance in such a controlled comparisons may not always align with real-world application priorities. For example, a minor compromise in accuracy (*e.g.* a 1mm decrease in trajectory estimation accuracy) can lead to substantial gains in memory efficiency. Hence, *hybrid encoding combining tri-plane and hash grid feature encoding* might be a preferable alternative to the tri-plane and dense combination. Such trade-off is discussed in the *supplmat*.

Explicit Hybrid Encoding. SLAM-centric methods utilize external trackers to ensure robust pose estimation and maintain system stability over time. Our evaluations show that dense grids, along with their hybrid encoding with tri-plane, excel in implicit scene encoding across two prominent leaderboards. To evolve these techniques for NeRF-

Dataset Mesh (With Dataset Pose) | NICE-SLAM (With Dataset Pose) | Explicit Hybrid Encoding (With Dataset Pose)

Figure 5. Qualitative evaluation of explicit hybrid encoding on '*scene0000*' sequence of ScanNet Dataset. Both NICE-SLAM and Ours run on the posed RGB-D stream to simulate an externally provided tracker.

	Lab		Practical	
	NICE-SLAM [54]	Ours	CO-SLAM [40]	Ours
Depth L1↓	3.53	**1.50**	3.02	**1.68**
Acc.↓	2.85	**1.65**	2.95	**2.40**
Comp.↓	**3.00**	5.62	**2.96**	4.64
Comp.%↑	**89.33**	83.93	**86.88**	83.48
ATE↓	1.95	**1.37**	-	**1.86**
Res×Dim↓	3×32	**2×2**	16×2	**2×2**

Table 5. Quantitative comparison with the SOTA method, Co-SLAM, featuring similar implicit scene representations. In the Lab scenario, both NICE-SLAM and our approach use dense grid representations, whereas, in the practical scenario, Co-SLAM and Our method are both in the spirit of hybrid encoding.

based mapping, enhancements in computational and memory efficiency are necessary. Dense grids exhibit cubic scaling with resolution ($O(n^3)$), while octrees show logarithmic scaling ($O(log(n))$) in sparse environments, leaving room for memory optimization. Drawing inspiration from [46, 53], we propose an explicit hybrid encoding, see Fig. 3(b), to *substitute the coarse-level feature grid with an octree structure and simplifying the encoding process by using a single-level dense grid*. To reduce the complexity of simultaneously encoding color and geometry for real-time applications using only two-dimensional features, we adopt the SDF residual optimization strategy detailed in [16].

Our explicit hybrid encoding method's capability for high-fidelity online map updating is qualitatively showcased in Fig. 5. When tested on sequence *'scene0169'* with a posed RGB-D stream on our hardware platform, explicit hybrid encoding achieves map updating at approximately 1 Hz, compared to around 0.25 Hz for NICE-SLAM. More records are available in *supplmat*.

Discussion. For context, we include comparisons with existing methods in Tab. 5. The top performances, *i.e.*, dense grid encoding with SDF(density) rendering in the Lab scenarios, and hybrid encoding with SDF(direct) rendering in the Practical scenarios showcase higher reconstruction and pose estimation accuracy, even with generally lower completeness and notably fewer total feature dimensions (Res×Dim). For a visual representation of these findings, readers are directed to the *supplmat*. In this benchmark, discussions concentrate on achieving optimal performance, rigorously measured against specific metrics under strictly regulated variables for fairness comparison. Nonetheless, the appeal of computational efficiency often takes precedence over the pursuit of complete comparative fairness. Please refer to the *supplmat* for such trade-offs.

5. Conclusion and Future Work

We proposed an open-source benchmark to evaluate INRs for RGB-D SLAM, filling a crucial gap in standardizing performance assessments. We demonstrated the superior efficacy of dense grid representations and introduced a hybrid encoding strategy that marries precision with efficiency. Our work not only guided the selection of INR components but also advanced practical SLAM applications.

Future Work. SLAM systems are intricate and are often tailored to specific environments. Consequently, it is not feasible to declare a universally superior representation within the scope of this paper. For example, in scenarios with orthogonal geometry and muted colors, such as an office corridor, planar representations may stand as more suitable alternatives to the dense grids. We advise that future research should expand to more diverse scenes.

Acknowledgement. This paper is supported by the National Natural Science Foundation of China (NSF) under Grant No. NSFC22FYT45 and the Guangzhou City, University and Enterprise Joint Fund under Grant No.SL2022A03J01278.

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6290–6301, 2022. 2, 5, 6

[2] Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4160–4169, 2023. 2

[3] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021. 3

[4] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022. 1

[5] Yu Chen and Gim Hee Lee. Dbarf: Deep bundle-adjusting generalizable neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24–34, 2023. 2

[6] Chi-Ming Chung, Yang-Che Tseng, Ya-Ching Hsu, Xiang-Qian Shi, Yun-Hung Hua, Jia-Fong Yeh, Wen-Chin Chen, Yi-Ting Chen, and Winston H Hsu. Orbeez-slam: A real-time monocular visual slam with orb features and nerf-realized mapping. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9400–9406. IEEE, 2023. 2, 3

[7] Forrester Cole, Kyle Genova, Avneesh Sud, Daniel Vlasic, and Zhoutong Zhang. Differentiable surface rendering via non-differentiable sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6088–6097, 2021. 2

[8] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion. *ACM Transactions on Graphics (TOG)*, 36:1 – 18, 2016. 1

[9] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 5

[10] Junyuan Deng, Qi Wu, Xieyuanli Chen, Songpengcheng Xia, Zhen Sun, Guoqing Liu, Wenxian Yu, and Ling Pei. Nerf-loam: Neural implicit representation for large-scale incremental lidar odometry and mapping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8218–8227, 2023. 3

[11] Jakob J. Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *European Conference on Computer Vision*, 2014. 3

[12] Yasaman Haghighi, Suryansh Kumar, Jean Philippe Thiran, and Luc Van Gool. Neural implicit dense semantic slam. *arXiv preprint arXiv:2304.14560*, 2023. 3

[13] Tongyan Hua, Haotian Bai, Zidong Cao, and Lin Wang. Fmapping: Factorized efficient neural field mapping for real-time dense rgb slam. *arXiv preprint arXiv:2306.00579*, 2023. 2, 3

[14] Tongyan Hua, Haotian Bai, Zidong Cao, Ming Liu, Dacheng Tao, and Lin Wang. Hi-map: Hierarchical factorized radiance field for high-fidelity monocular dense mapping. *arXiv preprint arXiv:2401.03203*, 2024. 4

[15] Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Anima Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5846–5854, 2021. 2

[16] Chenxing Jiang, Han-Qi Zhang, Peize Liu, Zehuan Yu, Hui Cheng, Boyu Zhou, and Shaojie Shen. H$_{2}$-mapping: Real-time dense mapping using hierarchical hybrid representation. *IEEE Robotics and Automation Letters*, 8:6787–6794, 2023. 3, 8

[17] Mohammad Mahdi Johari, Camilla Carta, and François Fleuret. ESLAM: efficient dense SLAM system based on hybrid representation of signed distance fields. *CoRR*, abs/2211.11704, 2022. 2, 3, 4

[18] Xin Kong, Shikun Liu, Marwan Taher, and Andrew J Davison. vmap: Vectorised object mapping for neural field slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 952–961, 2023. 3

[19] Stefan Leutenegger, Simon Lynen, Michael Bosse, Roland Y. Siegwart, and Paul Timothy Furgale. Keyframe-based visual–inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*, 34:314 – 334, 2015. 3

[20] Heng Li, Xiaodong Gu, Weihao Yuan, Zilong Dong, Ping Tan, et al. Dense rgb slam with neural implicit maps. In *The Eleventh International Conference on Learning Representations*, 2022. 3, 4

[21] Mingrui Li, Jiaming He, Yangyang Wang, and Hongyu Wang. End-to-end rgb-d slam with multi-mlps dense neural implicit representations. *IEEE Robotics and Automation Letters*, 2023. 3

[22] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5741–5751, 2021. 2

[23] Haomin Liu, Chen Li, Guojun Chen, Guofeng Zhang, Michael Kaess, and Hujun Bao. Robust keyframe-based dense slam with an rgb-d camera. *ArXiv*, abs/1711.05166, 2017. 1

[24] Jianheng Liu and Haoyao Chen. Towards real-time scalable dense mapping using robot-centric implicit representation. *arXiv preprint arXiv:2306.10472*, 2023. 3

[25] Shaofan Liu and Jianke Zhu. Efficient map fusion for multiple implicit slam agents. *IEEE Transactions on Intelligent Vehicles*, 2023. 3

[26] Shaofan Liu and Jianke Zhu. Efficient map fusion for multiple implicit slam agents. *IEEE Transactions on Intelligent Vehicles*, 2023. 3

[27] Hidenobu Matsuki, Keisuke Tateno, Michael Niemeyer, and Federic Tombari. Newton: Neural view-centric mapping for on-the-fly large-scale slam. *arXiv preprint arXiv:2303.13654*, 2023. 2

[28] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2

[29] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 1, 3

[30] Raul Mur-Artal and Juan D. Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33:1255–1262, 2016. 3

[31] Raul Mur-Artal, José M. M. Montiel, and Juan D. Tardós. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31:1147–1163, 2015. 3

[32] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew William Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, 2011. 1

[33] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5569–5579, 2021. 2

[34] Antoni Rosinol, John J Leonard, and Luca Carlone. Nerf-slam: Real-time dense monocular slam with neural radiance fields. *arXiv preprint arXiv:2210.13641*, 2022. 2, 3

[35] Erik Sandström, Yue Li, Luc Van Gool, and Martin R Oswald. Point-slam: Dense neural point cloud-based slam. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18433–18444, 2023. 2, 3

[36] Thomas Schöps, Torsten Sattler, and Marc Pollefeys. Bad slam: Bundle adjusted direct rgb-d slam. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 134–144, 2019. 1

[37] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 2, 5

[38] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J. Davison. imap: Implicit mapping and positioning in real-time. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 6209–6218. IEEE, 2021. 2, 3, 4

[39] Yijie Tang, Jiazhao Zhang, Zhinan Yu, He Wang, and Kai Xu. Mips-fusion: Multi-implicit-submaps for scalable and robust online neural rgb-d reconstruction. *arXiv preprint arXiv:2308.08741*, 2023. 3

[40] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13293–13302, 2023. 2, 3, 4, 8

[41] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021. 2, 4, 6

[42] Thomas Whelan, Michael Kaess, Hordur Johannsson, Maurice Fallon, John J Leonard, and John McDonald. Real-time large-scale dense rgb-d slam with volumetric fusion. *The International Journal of Robotics Research*, 34(4-5):598–626, 2015. 1

[43] Thomas Whelan, Renato F. Salas-Moreno, Ben Glocker, Andrew J. Davison, and Stefan Leutenegger. Elasticfusion: Real-time dense slam and light source estimation. *The International Journal of Robotics Research*, 35:1697 – 1716, 2016. 1

[44] Beichen Xiang, Yuxin Sun, Zhongqu Xie, Xiaolong Yang, and Yulin Wang. Nisb-map: Scalable mapping with neural implicit spatial block. *IEEE Robotics and Automation Letters*, 2023. 3

[45] Zhixin Yan, Mao Ye, and Liu Ren. Dense visual slam with probabilistic surfel map. *IEEE Transactions on Visualization and Computer Graphics*, 23:2389–2398, 2017. 1

[46] Xingrui Yang, Hai Li, Hongjia Zhai, Yuhang Ming, Yuqian Liu, and Guofeng Zhang. Vox-fusion: Dense tracking and mapping with voxel-based neural implicit representation. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 499–507. IEEE, 2022. 2, 3, 4, 8

[47] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *ArXiv*, abs/2106.12052, 2021. 2

[48] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1323–1330. IEEE, 2021. 2

[49] Yijun Yuan and Andreas Nuechter. Uni-fusion: Universal continuous mapping. *arXiv preprint arXiv:2303.12678*, 2023. 3

[50] Wei Zhang, Tiecheng Sun, Sen Wang, Qing Cheng, and Norbert Haala. Hi-slam: Monocular real-time dense mapping with hybrid implicit fields. *arXiv preprint arXiv:2310.04787*, 2023. 2

[51] Yongqiang Zhang, Zhipeng Hu, Haoqian Wu, Minda Zhao, Lincheng Li, Zhengxia Zou, and Changjie Fan. Towards unbiased volume rendering of neural implicit surfaces with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4359–4368, 2023. 2

[52] Youmin Zhang, Fabio Tosi, Stefano Mattoccia, and Matteo Poggi. Go-slam: Global optimization for consistent 3d instant reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3727–3737, 2023. 2, 3, 4

[53] Xingguang Zhong, Yue Pan, Jens Behley, and Cyrill Stachniss. Shine-mapping: Large-scale 3d mapping using sparse hierarchical implicit neural representations. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8371–8377. IEEE, 2023. 3, 8

[54] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc Pollefeys. NICE-SLAM: neural implicit scalable encoding for SLAM. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 12776–12786. IEEE, 2022. 2, 3, 4, 8

[55] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, Martin R Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. *arXiv preprint arXiv:2302.03594*, 2023. 3, 4