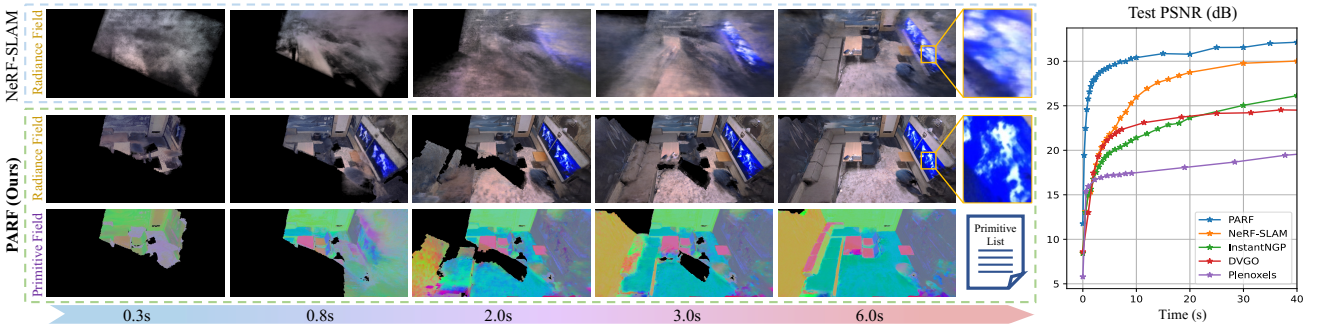
PARF: Primitive-Aware Radiance Fusion for Indoor Scene Novel View Synthesis

Haiyang Ying[1], Baowei Jiang[1], Jinzhi Zhang[1], Di Xu[2], Tao Yu[1†], Qionghai Dai[1], Lu Fang[1†]

[1]Tsinghua University, [2]Huawei Cloud



(a) Incremental Radiance Field Reconstruction Performance Comparison (b) Converge Speed Evaluation

Figure 1: Performance comparison with the state-of-the-art radiance field reconstruction methods on Replica dataset. With the proposed hybrid representation and primitive-aware fusion framework, our method PARF enjoys significantly faster convergence and high-quality rendering for indoor scene novel view synthesis. In (a) the incremental reconstruction setting, we assume a SLAM system with the tracking speed of 10fps.

Abstract

This paper proposes a method for fast scene radiance field reconstruction with strong novel view synthesis performance and convenient scene editing functionality. The key idea is to fully utilize semantic parsing and primitive extraction for constraining and accelerating the radiance field reconstruction process. To fulfill this goal, a primitive-aware hybrid rendering strategy was proposed to enjoy the best of both volumetric and primitive rendering. We further contribute a reconstruction pipeline conducts primitive parsing and radiance field learning iteratively for each input frame which successfully fuses semantic, primitive, and radiance information into a single framework. Extensive evaluations demonstrate the fast reconstruction ability, high rendering quality, and convenient editing functionality of our method.

1. Introduction

Indoor scene 3D reconstruction and novel view synthesis (NVS) is a long-lasting classical topic in the field of computer vision for decades, which is widely used in virtual reality, robot perception, and visualization. Classic indoor scene reconstruction methods [22, 23] focus on geometric registration and fusion [23] based on feature matching, bundle adjustment [6], and multi-view stereo [37] algorithms. However, these methods rely on discrete point clouds or voxels for scene representation, which results in high memory overhead and limited ability to describe scene details, making it challenging to achieve realistic NVS effects.

The emergence of implicit continuous representations based on neural networks has revolutionized 3D vision tasks. NeRF [18] represents the density and color fields of the scene using an implicit representation. Coupled with volume rendering techniques [19, 8, 13], NeRF achieves a simple but effective pipeline for end-to-end radiance field reconstruction. NeRF not only enables realistic novel view synthesis, but also facilitates 3D structure, material, and appearance recovery. However, NeRF-based methods tend to fit a diffused density field to the ground truth geometry surface for achieving view-dependent volume rendering effects, which may not be suitable for view extrapolation due to the lack of a sharp geometry constraint. Although incorporating depth information can constrain the learning of implicit geometry field, generating accurate samples for view extrapolation and achieving faster convergence remains challenging as NeRF requires relatively re-

†The corresponding authors are Lu Fang (fanglu@tsinghua.edu.cn, http://www.luvision.net/) and Tao Yu (ytrock@tsinghua.edu.cn).

dundant sampling around the surface for volume rendering [29]. Additionally, training NeRF in a manner of pixel-independent strategy neglects the global geometric consistency of the whole scene, which introduces noise and artifacts in the final reconstruction.

To overcome this challenge, primitive-based methods such as NeurMiPs [16] use global plane prior extracted from traditional primitive detection methods [3, 11, 30]. These methods typically use a fixed number of planes to fit the reconstructed point cloud obtained from other methods [1, 15, 28]. This global structure prior effectively regularizes the implicit density field in the planar regions, thereby improving view extrapolation performance. However, for regions that are difficult to describe with planes, such as curved surfaces and thin structures, the boundary of the fitted plane suffers from obvious discontinuity artifacts.

In this paper, we aim to establish an incremental radiance field reconstruction pipeline based on NeRF and semantic parsing for much higher performance, no matter view interpolation or extrapolation, with an order of magnitude fewer training iterations than SOTA methods. Our key innovation is a divide-and-conquer strategy that makes the representation ultra-simple in global primitive regions while keeping it complex in non-plane local details.

In light of this, we propose **P**rimitive-**A**ware **R**adiance **F**usion, named **PARF**, for indoor scene novel view synthesis. Our key idea is that: Indoor scene always contains many planar regions, and by leveraging the global primitive prior of planar regions and the local implicit representation for non-planar regions, we can achieve much better performance with strong semantic guidance. However, representing, fusing, and training both primitive and non-primitive representation in the same radiance field from sequential RGB-D inputs in real-time is non-trivial. In order to solve the problems above, PARF proposes a hybrid representation that uses discrete semantic volume as a medium to integrate planar semantics into the continuous and implicit scene radiance field. This allows for a primitive-aware sampling process in volume rendering, resulting in improved efficiency and quality. Additionally, PARF dynamically maintains a global scene plane representation and can fuse and differentiate planar regions through dynamic fusion and adaptive update. This enables efficient and noise-robust optimization of the radiance field, as well as direct semantic editing capabilities. Overall, PARF successfully incorporates semantic parsing and primitive merging into a radiance fusion framework, enabling efficient training, high-quality rendering, and semantic editing.

The contributions of PARF can be summarized as:

- We propose PARF, a novel hybrid scene representation to decompose the radiance field into primitive-based and volume-based components in a unified form.
- We contribute an incremental reconstruction framework for primitive-aware radiance fusion, which effectively leverages the benefits of semantic parsing, primitive merging, and neural representation for indoor scene reconstruction.
- Extensive evaluations demonstrate that our method enjoys fast convergence, robust view extrapolation performance, and convenient scene editing ability.

2. Related Works

2.1. Neural Implicit Rendering and Fusion

Neural Radiance Field [20, 18, 41, 35, 17, 21] is an approach that utilizes coordinate-based MLP as implicit scene representations to continuously encode scene geometry, which achieves high-quality and view-dependent appearance modeling. Signed Distance Field is also an implicit representation that is beneficial to model a continuous geometric surface [25, 2, 44, 10, 39].

Towards indoor scene fusion, NeuralRGBD [2] models and optimizes the scene geometry as a continuous SDF function and achieves high completeness though the training time is quite long. NICE-SLAM [44] proposes a dense SLAM system that optimizes a hierarchical representation with pre-trained geometric priors which, enables detailed reconstruction on large indoor scenes. NeuralRecon [34] establishes a learning-based TSDF fusion module based on GRU to guide the network to fuse features from previous fragments in real time.

For indoor scene rendering, NeRFusion [42] applies a pre-trained fusion model for real-time RGB radiance fusion for novel view synthesis. Based on InstantNGP [21], NeRF-SLAM [29] create a NeRF-based SLAM system with extra depth as supervision signal to achieve real-time radiance field reconstruction.

However, a limitation of volumetric representations is that the optimization is applied on the integral of the radiance field without sufficient prior. This can lead to biased and inconsistent geometry and therefore results in bad view extrapolation and slow convergence speed. Though prior-based methods like [24, 12] uses strong regularization on visual patches show satisfactory results under sparse-view setting, the performance gap still remains between the controlled scenes with structured observations and real-world scenes with unstructured captures.

2.2. Primitive based Rendering and Fusion

Structural scene prior has been proven to be beneficial in neural rendering and fusion [26, 16, 10, 4, 36, 38].

ManhattanSDF [10] uses the Manhattan prior to constrain the normal of an implicit SDF field, which highly relies on known semantics of scene partition and the Manhattan frame. Further work [26] employs self-supervision

of depth and normals through the Manhattan prior and volumetric rendering without the Manhattan frame. But these two works are still limited to the Manhattan assumption, which is not sufficient for modeling unordered planes in 3D space.

To handle planes with free poses, PlanarRecon [36] proposes to fusion bounded planes in an incremental manner based on NeuralRecon [34] but suffers from incomplete fusion results. NeurMiPs [16] uses SFM point cloud to decompose the scene into optimizable planar experts, which benefits from fast planar rendering and optimization. However, purely plane-based modeling may lead to difficulty in complex scene modeling, especially when observation is insufficient.

From the view of rendering efficiency, MobileNeRF [4] decomposes the scene into a set of polygons with textures representing binary opacities and feature vectors. However, since the triangle primitive is quite small, it still suffers from overfitting and cannot handle view extrapolation.

3. Representation

We present a novel primitive-aware hybrid representation to model the scene in a hybrid manner. Based on a primitive-aware semantic volume, the scene can be divided into volume-based and primitive-based regions automatically. Both of dense volume rendering and primitive-based rendering can be applied via a unified representation. In this section, we will first recap the NeRF-based volume rendering in Sec. 3.1, and introduce the primitive-based rendering method in Sec. 3.2. Then the core idea, primitive-aware hybrid representation, will be introduced in detail in Sec. 3.3.

3.1. Volume-based Rendering

We utilize the radiance field [20] as the basis of our representation. More specifically, given the position $\mathbf{x}_i \in \mathbb{R}^3$ and the view direction $\mathbf{d}_i \in \mathbb{R}^2$, an MLP network $F_{\mathbf{\Theta}}$ will act as a decoder and output the per-point attributes:

$$\sigma_i = F_{\mathbf{\Theta}}(\gamma(\mathbf{x}_i)), \quad \mathbf{c}_i = F_{\mathbf{\Theta}}(\gamma(\mathbf{x}_i), SH(\mathbf{d}_i)), \tag{1}$$

where $\sigma_i \in \mathbb{R}$ is the view-independent density and $\mathbf{c}_i \in \mathbb{R}^3$ is the view-dependent RGB color. $\gamma(\cdot)$ and $SH(\cdot)$ are positional encoding functions based on multi-resolution hashing [21] and spherical harmonics respectively. In order to model the semantic information of the space additionally, inspired by semanticNeRF [43], we add a semantic head to the MLP $F_{\mathbf{\Theta}}$ and get the per-point semantic information $\mathbf{s}_i \in \mathbb{R}^4$: $\mathbf{s}_i = F_{\mathbf{\Theta}}(\gamma(\mathbf{x}_i))$, where $\mathbf{s}_i = (\mathbf{n}_p, d_p)$ indicates the primitive the queried point $\mathbf{x}_i$ is located on. We define each primitive as a 3D plane which will be introduced in Sec. 3.2. Instead of predicting the discrete object class labels [43], our semantic logits $\mathbf{s}_i$ are continuous and indicate geometric-level semantic information of the scene.

Then color and density will be integrated along the ray to get the rendered pixel color $\mathbf{c}(\mathbf{r})$.

$$\mathbf{c}(\mathbf{r}) = \sum_{i=1}^{N} T_i \alpha_i \mathbf{c}_i, \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_i) \tag{2}$$

where $\alpha_i = 1 - exp(-\sigma_i \delta_i)$ is the opacity and $\delta_i = r_{i+1} - r_i$ is the distance between adjacent samples. Besides RGB color, Eq. 2 can also be used to render depth and semantic values as:

$$d(\mathbf{r}) = \sum_{i=1}^{N} T_i \alpha_i r_i, \quad \mathbf{s}(\mathbf{r}) = \sum_{i=1}^{N} T_i \alpha_i \mathbf{s}_i. \tag{3}$$

3.2. Primitive-based Rendering

Since dense volume rendering suffers from expensive sampling and ambiguous geometry around the ground truth surface, geometric primitive-based rendering may be an alternative choice. We define each primitive as a 3D plane $\mathbf{p} = \{\mathbf{n}_p, d_p\}$, where $\mathbf{n}_p \in \mathbb{R}^3$ is the plane normal and $d_p \in \mathbb{R}_+$ is the distance from the origin point to the plane, Each primitive $\mathbf{p} = \{\mathbf{n}_p, d_p\}$ holds $\mathbf{n}_p \cdot \mathbf{x} = d_p$ for the point $\mathbf{x}$ located on it.

Given a ray $\mathbf{r} = \{\mathbf{o}, \mathbf{d}\}$ and a primitive $\mathbf{p} = \{\mathbf{n}_p, d_p\}$, the ray-primitive intersection point can be calculated analytically:

$$\mathbf{x} = \mathbf{o} + \frac{\mathbf{d}_p - \mathbf{o} \cdot \mathbf{n}_p}{\mathbf{d} \cdot \mathbf{n}_p} \mathbf{d}. \tag{4}$$

We model primitives as colored and translucent planes so the same rendering method (Eq. 1-Eq. 3) can be applied. However, the primitive intersections are often sparse and unevenly spaced, so $\delta_i = r_{i+1} - r_i$ is unreasonable in primitive-based rendering. To solve this, we assume each plane shares an equal and fixed thickness, i.e., $\delta_i = \delta_p$.

The primitive-based rendering is as follows: shooting a ray $\mathbf{r} = \{\mathbf{o}, \mathbf{d}\}$ from the camera optical center to the space, computing all intersections with existing primitives $\mathbf{P}_G$, sorting intersections $\{\mathbf{x}_i\}$ by the distance from ray origin, sending intersection positions and ray directions into the MLP $F_{\mathbf{\Theta}}$ and executing the volume rendering as Eq. 1 and Eq. 2 with fixed thickness $\delta_i = \delta_p$.

Though plane-based methods are very efficient [4, 16], pure primitive-based modeling may still lead to wrong geometry and blurry rendering results when observations are limited. On the other hand, volume-based rendering can relieve the problem by dense sampling for regions with complex geometry. In light of this, a hybrid representation that combines both primitive and non-primitive based rendering may help improve the fidelity of scene modeling and rendering results.

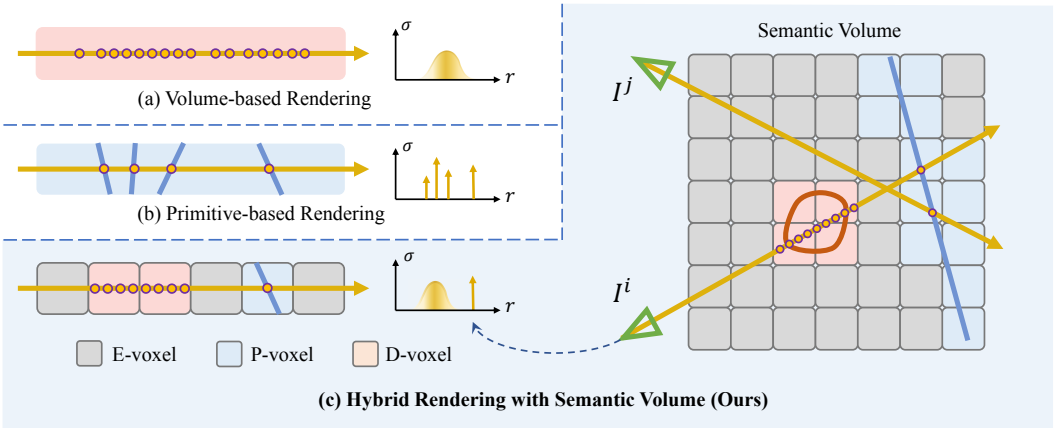


Figure 2: Representation. By discretizing the scene with a semantic volume, the proposed (c) **primitive-aware hybrid rendering** enables highly efficient sampling and rendering in the mixture of both (a) volumetric and (b) primitive rendering.

3.3. Primitive-aware hybrid representation

Taking advantage of both volume rendering and primitive-based rendering, we design a hybrid voxel-based representation to achieve complex scene modeling and fast inference simultaneously, which also helps lift up convergence speed.

The basic idea is to create an indicator to help tell apart primitive and non-primitive regions. Specifically, to represent a scene in a hybrid manner, we establish a dense semantic volume $\mathbf{V_s}$ accompanied by a list of primitive parameters $\mathbf{P}_G$ to describe the global semantic information of each point in the scene. Each voxel of the volume contains an integer label $v_i \in \mathbb{Z}, (v_i \geq -1)$ indicating the type of the voxel. Different sampling and rendering strategies are used for different types of voxels. We first apply ray marching in the semantic volume $\mathbf{V_s}$ to sample points, and a hybrid rendering method is utilized to render per-pixel color, depth, and semantic values to form the rendered images.

Ray Marching. To render a ray, we apply ray marching in the semantic volume to sample points for rendering. At each marching step, we determine which voxel the current point belongs to. Then the semantic label of the current voxel is checked by the semantic volume, and the label determines the sample operation we will execute. We define the following three kinds of voxel to guide the sampling:

E-voxel holds $v_i = -1$, which means the voxel is empty, and the marching process will skip this voxel without sampling.

D-voxel holds $v_i = 0$, which means the voxel is occupied. Samples in this voxel will be dense and evenly spaced.

P-voxel has $v_i \geq 1$, which means the voxel is also occupied, but we apply primitive-based sampling and rendering in these voxels. Each $v_i \geq 1$ corresponds to one primitive in the maintained parameter list $\mathbf{P}_G$. We extract the parameter $\mathbf{p} = \{\mathbf{n}_p, d_p\}$ of the indicated primitive v_i from $\mathbf{P}_G$. Then the ray-plane intersection point is calculated with Eq. 4, and its coordinate is saved for later rendering. After that, we set the coordinates of the next marching point as a point at a fixed distance ψ behind the plane and continue the marching process until the sample point moves outside the semantic volume $\mathbf{V_s}$ or the ray has reached the maximum number of sampling steps. The fixed distance ψ is set as the diagonal size of one voxel along the plane normal.

Hybrid Rendering. After ray marching, we have gathered sampled point set $\{\mathbf{x}\}$ from the traversed D-voxels and P-voxels. Then the point set is sent into MLP F_Θ to infer color and density. When calculating the opacity α_i of each point, we choose the thickness as $\delta_i = r_{i+1} - r_i$ for points sampled in D-voxels and $\delta_i = \delta_p$ for points sampled in P-voxels (δ_p is set to 1.0 in all the experiments). Then the pixel color can be rendered by Eq. 2

One advantage of this hybrid semantic volume is that parametric parameters and boundaries of primitives are encoded into the scene in a unified manner, which means no extra plane parameterization is needed for each primitive.

4. Primitive-aware radiance fusion

Given a posed RGB-D sequence as input, we reconstruct a primitive-aware volumetric field for novel view synthesis in an incremental manner. We apply a plane detection algorithm for each input depth image to estimate plane parameters and merge them into the global plane list. After that, the new primitives will be fused into the semantic volume $\mathbf{V_s}$. Finally, the MLP F_Θ and the semantic volume $\mathbf{V_s}$ will be optimized together via the proposed hybrid rendering.

4.1. Parametric primitive extraction

Though depth sensors may give noisy observation, regions with continuous depth values provide strong prior for the existence of smooth surfaces. This prior is especially

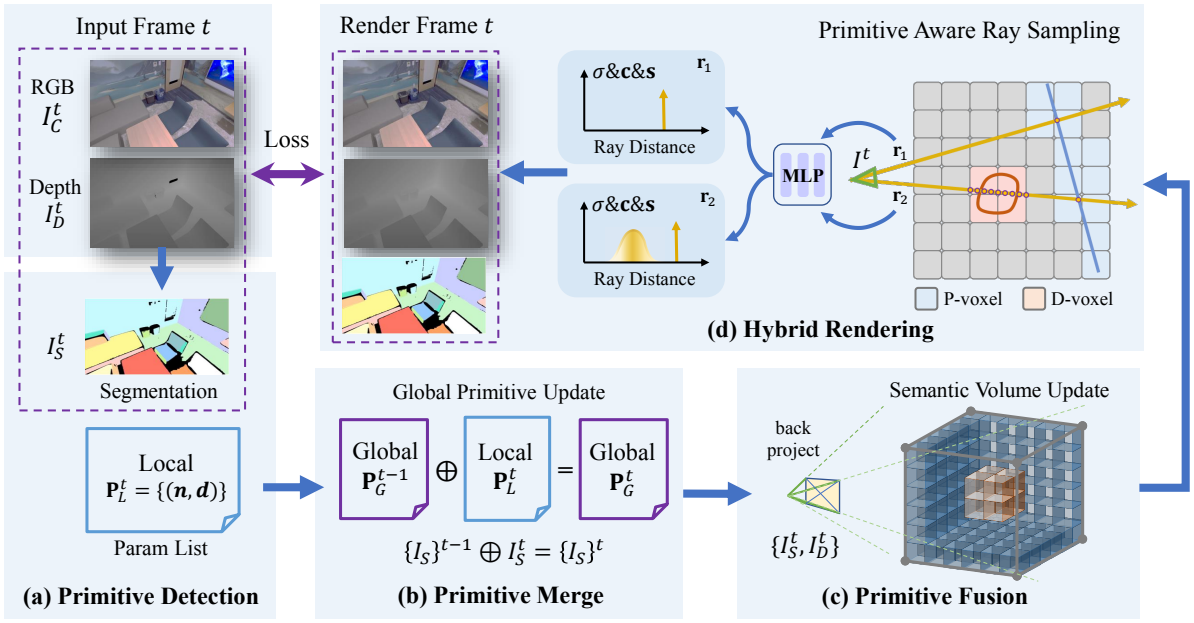


Figure 3: Framework of PARF. For each input RGB-D frame, we apply plane detection to get semantic image I_S^t and a local parameter list $\mathbf{P}_L^t$, which are then merged into the global primitive list. Then the depth frame I_D^t and the updated semantic frame I_S^t are fused into the semantic volume $\mathbf{V_s}$ to update the global representation. Through hybrid rendering, the color, depth, and semantic images can be rendered and supervised with input images and detected primitive images.

valuable for recovering texture-less regions, which is difficult for NeRF [20] and MVS [31] methods. We choose plane as a similar prior to depict the low-level semantics of the scene. Inspired by TSDF Fusion, we propose to detect and fuse the semantic information into the semantic volume in an incremental manner.

Primitive detection. Given depth frame I_D^t at time t, a real-time plane detection method CAPE [27] is utilized to detect planes and get a parameter list $\mathbf{P}_L^t$ as well as a semantic image I_S^t, where each pixel with non-zero value corresponds to a plane in the list $\mathbf{P}_L^t$. For each plane in $\mathbf{P}_L^t$, we validate its flatness by back-projecting the pixels to 3D space and calculating the mean point-to-plane error. If the error exceeds the threshold ϵ_1, the plane will be refused, and the corresponding pixel value on I_S^t will be set to zero.

Primitive merge. After plane detection, the detected planes $\mathbf{P}_L^t$ are compared and merged into the global plane list $\mathbf{P}_G^{t-1}$ to get $\mathbf{P}_G^t$. For each plane $\mathbf{p}_j^t \in \mathbf{P}_L^t, j \in [1, J]$, the distance between $\mathbf{p}_j^t$ and each plane $\mathbf{p}_m \in \mathbf{P}_G^{t-1}, m \in [1, M]$ is calculated as:

$$dist(\mathbf{p}_j^t, \mathbf{p}_m) = \left| d_j^t \mathbf{n}_j^t - d_m \mathbf{n}_m \right|. \tag{5}$$

If all the distances are larger than threshold ϵ_2, plane $\mathbf{p}_j^t$ will be added to $\mathbf{P}_G^{t-1}$ as a new plane. Otherwise, the pixels in I_S^t correspond to plane $\mathbf{p}_j^t$ will be replaced with the index of the closest plane $\arg\min_m dist(\mathbf{p}_j^t, \mathbf{p}_m)$.

After merging $\mathbf{P}_L^t$ into the global list $\mathbf{P}_G^{t-1}$, we apply PCA to evaluate the normal $\tilde{\mathbf{n}}_m$ of each plane $\mathbf{p}_m \in \mathbf{P}_G^t$ by sampling points in the last t semantic index images. If $|\tilde{\mathbf{n}}_m - \mathbf{n}_m| > \epsilon_3$, then plane $\mathbf{p}_m$ will be removed from $\mathbf{P}_G^t$. In our experiments, the threshold values are $\epsilon_1 = 0.005$, $\epsilon_2 = 0.01$, and $\epsilon_3 = 0.1$.

4.2. Primitive fusion in semantic volume

Inspired by TSDF Fusion based reconstruction, we further fuse the current semantic frame I_S^t and the depth frame I_D^t into the semantic volume for hybrid rendering.

At beginning of the fusion, we assign all voxels in semantic volume $\mathbf{V_s}$ as E-voxels ($v_i = -1$). When a new frame comes, we project all the grid points $\{x_i\}$ (center points of voxels) onto the current frame t to get the pixel coordinates $\{u_i\}$ and the observed depth values $\{D^t(x_i)\}$.

If the projected semantic value $I_S^t(u_i) = 0$, which indicates non-primitive, then we apply a bilateral truncated band B_1 to threshold the valid grid points according to the depth value $I_D^t(u_i)$ of the projected pixel. The valid voxels will be assigned as D-voxels ($v_i = 0$):

$$V_{v=0}^t = \{v_i | I_D^t(u_i) - B_1 < D^t(x_i) < I_D^t(u_i) + B_1\}. \tag{6}$$

If $I_S^t(u_i) > 0$, the pixel indicates a primitive $m \in [1, M]$. Firstly, we compute the ray-plane intersection with Eq. 4 and get the observed depth values $\{S^t(x_i)\}$ of the intersection point for the following threshold operations. Sec-

ondly, we use a narrower bilateral band B_2 to threshold the voxels and assign plane index m to these voxels as P-voxels. Thirdly, we take plane primitive as a strong regularization of the space, where there should be no occupied voxels between the camera t and the observed plane $\mathbf{p}_m$. So we apply a unilateral truncation band B_1 to assign voxels behind the plane as D-voxels only, and the voxels located before the plane will be set to E-voxels.

$$V^t_{v>0} = \{v_i | I^t_D(u_i) - B_2 < S^t(x_i) < I^t_D(u_i) + B_2\}, \quad (7)$$

$$V^t_{v=0} = \{v_i | I^t_D(u_i) + B_2 < S^t(x_i) < I^t_D(u_i) + B_1\}, \quad (8)$$

$$V^t_{v=-1} = \{v_i | S^t(x_i) < I^t_D(u_i) - B_2\}. \quad (9)$$

The bandwidth $B_1 = 6\psi, B_2 = \psi$, where ψ is the diagonal size of one voxel. Please refer to our supplementary material for more details.

This primitive-based fusion operation helps sparsify the space, which is beneficial for fast convergence. After the parametric semantic fusion, the updated semantic volume $\mathbf{V_s}$ can be used to execute hybrid rendering (Sec. 3.3) and further optimization (Sec. 4.3.1).

4.3. Implementation details

4.3.1 Optimization

During training, we optimize the MLP $F_{\mathbf{\Theta}}$ and the semantic volume $\mathbf{V_s}$ via hybrid volume rendering. For volume rendering, we apply four loss functions: $\mathcal{L}_c = \sum_{\mathbf{r}} \|\mathbf{c}(\mathbf{r}) - \mathbf{c}_{\text{gt}}(\mathbf{r})\|^2_2$, $\mathcal{L}_d = \sum_{\mathbf{r}} \|d(\mathbf{r}) - d_{\text{gt}}(\mathbf{r})\|^2_2$, $\mathcal{L}_s = \sum_{\mathbf{r}} \|\mathbf{s}(\mathbf{r}) - \mathbf{s}_{\text{gt}}(\mathbf{r})\|^2_2$, and $\mathcal{L}_{reg} = \sum_{\mathbf{r}} -o(\mathbf{r}) \log(o(\mathbf{r}))$, where $o(\mathbf{r}) = \sum_{i=1}^{N} T_i \alpha_i$ is the opacity of each ray. $\mathcal{L}_{reg}$ is used to regularize each ray to be completely saturated or unsaturated. The total loss is:

$$\mathcal{L}_{total} = \mathcal{L}_c + \lambda_1 \mathcal{L}_d + \lambda_2 \mathcal{L}_s + \lambda_3 \mathcal{L}_{reg}, \quad (10)$$

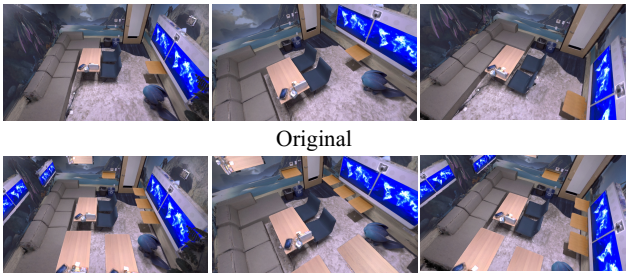
The hyper-parameters are set as $\lambda_1 = 1.0, \lambda_2 = 0.04, \lambda_3 = 0.001$ for all the experiments. With a cosine annealing schedule, the learning rate is set from $1e^{-2}$ to $3e^{-4}$. The ray number of each batch is 8192, and each epoch contains 1000 iterations. We train PARF for 5 epochs for each scene. We apply the same pruning strategy as InstantNGP [21] to prune voxels with low density periodically, which helps sparsify the space and accelerate the optimization speed.

4.3.2 Scene Editing

The hybrid scene representation helps to achieve more convenient scene editing with the following actions.

Primitive Deletion. Since each primitive holds a unique label v_i in the semantic volume $\mathbf{V_s}$, the primitive can be easily removed by setting P-voxels labeled with $v_i \geq 1$ to E-voxels $v_i = -1$.

Primitive Transformation. To transform primitives, We set up an extra editing volume $\mathbf{V_e}$ and a list $\mathbf{T_e}$ to store the editing operations. Non-zero voxel $v^e_i \in \mathbf{V_e}$ indicates an editing operation $\mathbf{t}^e_i \in \mathbf{T_e}$. During ray marching, if the ray arrives a voxel with $v^e_i \geq 1$ in $\mathbf{V_e}$, the current marching point $\{\mathbf{x}, \mathbf{d}\}$ will be transformed to $(\mathbf{x}', \mathbf{d}') = \mathbf{t}^e_i(\mathbf{x}, \mathbf{d})$ according to $\mathbf{t}^e_i$. Then the transformed $(\mathbf{x}', \mathbf{d}')$ will be sent into MLP $F_{\mathbf{\Theta}}$ for attributes inferencing and rendering. The visualization of editing results are shown in Fig. 4.



Original

After editing

Figure 4: Realistic semantic scene editing results.

5. Experiments

In this section, we report the experimental results in detail. We first introduce the experiment settings, then show that PARF achieves high-quality render results and more robust view extrapolation compared to the SOTA methods.

5.1. Datasets

We perform experiments on the following public datasets: one synthetic dataset with ground truth depths and real-world datasets with noisy depths.

Replica [32] consists of 18 scenes scanned and reconstructed from the real world, which can be rendered to RGB-D sequences. We conduct experiments on 8 sequences of Replica following existing method [29]. Specifically, we use 2000 frames with an interval of 10 frames for the training of each scene. Besides, 10 interpolation views and 10 extrapolation views are rendered as ground truth images for the evaluation of each scene.

BundleFusion [6] dataset includes real captured RGB-D sequences from 7 real-world indoor scenes. We obtain camera poses from the BundleFusion algorithm. For each scene, 2000 frames with an interval of 12 or 24 frames from each of 4 scenes (`apt0`, `apt2`, `copyroom`, and `office2`) are used for evaluation.

ScanNet [5] We choose 3 scenes (`scene0012`, `scene0027`, and `scene0457`) from ScanNet dataset for evaluation. For each scene, 2000∼5000 frames with an interval of 10 or 40 frames are chosen and the camera poses are also obtained from the BundleFusion algorithm [6].

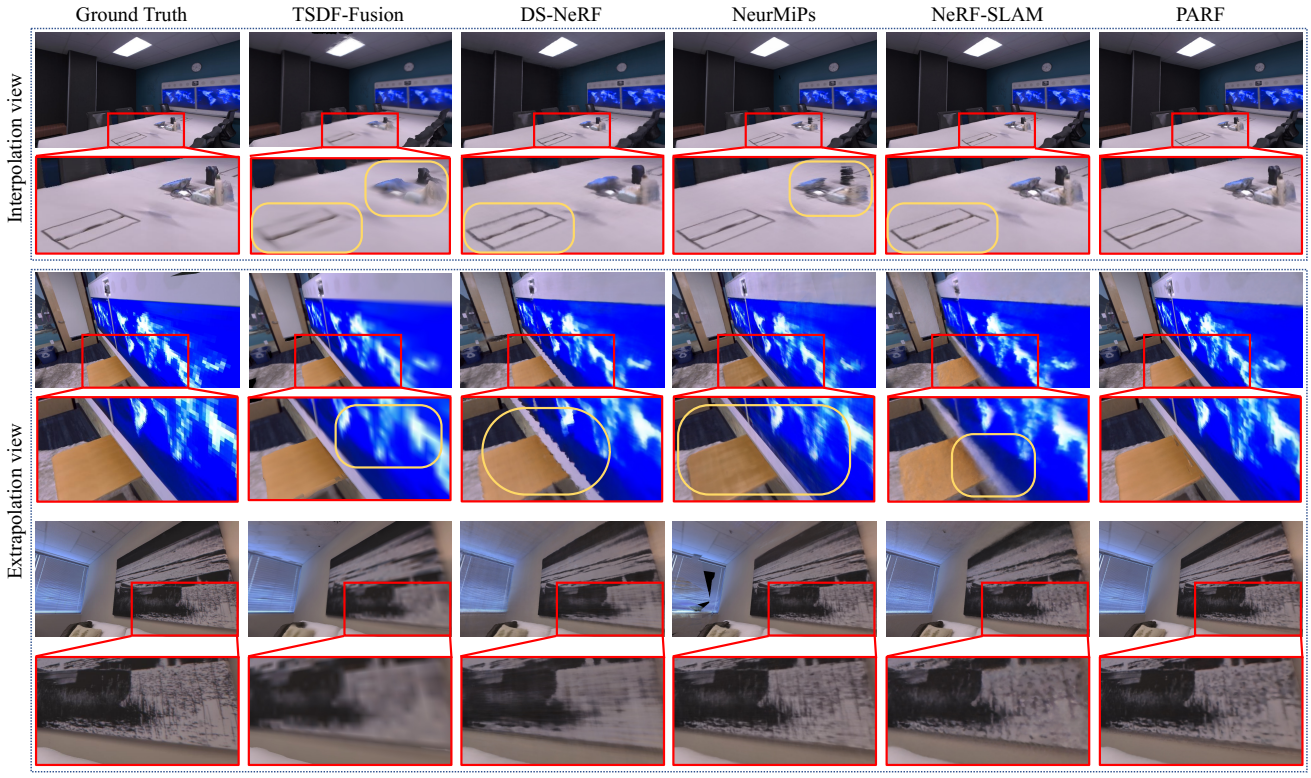


Figure 5: Qualitative comparison on both interpolation and extrapolation views of the Replica dataset. PARF shows high-quality rendering results under both settings, while other methods suffer from blurry patterns, geometry distortion, or floaters.

	Mean			Interpolation			Extrapolation			w/ depth input	training time	render (fps)
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓			
DVGO	21.97	0.781	0.487	27.36	0.835	0.525	16.58	0.727	0.448	0	∼10min	0.4
Plenoxels	27.54	0.860	0.370	31.86	0.903	0.333	23.22	0.832	0.408	0	∼18min	3748
NeRF	30.89	0.890	0.365	32.81	0.904	0.358	28.98	0.879	0.371	0	∼3h	0.03
InstantNGP	31.44	0.892	0.354	34.25	0.917	0.323	28.63	0.867	0.385	0	∼85s	9.25
TSDF Fusion	27.64	0.858	0.379	28.55	0.869	0.371	26.73	0.846	0.387	1	-	-
DSNeRF	31.16	0.887	0.370	32.34	0.895	0.369	29.99	0.878	0.371	1	∼3h	0.03
NeurMiPs	33.29	0.923	0.290	35.07	0.938	0.271	31.52	0.908	0.309	1	∼10h	0.73
NeRF-SLAM	34.50	0.930	0.283	36.94	0.948	0.245	32.07	0.913	0.320	1	∼88s	31.3
PARF (ours)	**35.09**	**0.943**	**0.228**	**37.00**	**0.954**	**0.204**	**33.18**	**0.933**	**0.253**	1	**∼40s**	62.5

Table 1: Quantitative evaluation results on the Replica dataset. Compared to baseline methods, PARF achieves the best performance in all three metrics and shows a significant boost in extrapolation ability.

Since no extra views can be obtained (like the Replica dataset), the BundleFusion and ScanNet datasets are only used to measure the interpolation ability in quantitative evaluation.

5.2. Baselines

We compare our method against the state-of-the-art methods for novel view synthesis, such as NeRF [20], DVGO [33], Plenoxels [9], InstantNGP [21], which do not rely on depth input. Besides, methods like TSDF Fusion [40], DS-NeRF [7], NeurMiPs [16], NeRF-SLAM [29] that need geometry guidance are also compared.

We evaluate the standard **TSDF Fusion [40]** algorithm with the volume size of 512^3 and the truncation band size of 10 voxels. For **DS-NeRF** [7], we use dense depth maps as guidance instead of sparse point clouds to get better performance. **NeurMiPs** [16] models the scene with multiple planar primitives. Since the plane parameters should be optimized with a global point cloud, it is difficult for NeurMiPs to be applied in an incremental reconstruction framework. We prepare the point clouds by fusing multi-view depth maps for the plane initialization stage. During the

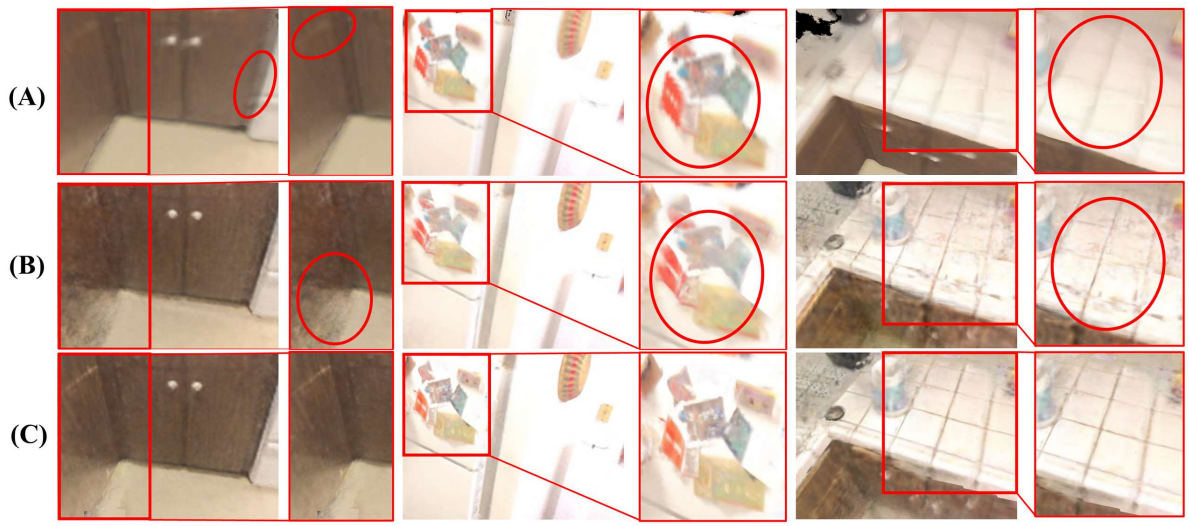


Figure 6: Qualitative comparison of TSDF Fusion(A), NeRF-SLAM(B), and PARF(C) on BundleFusion dataset. The views are rendered from extrapolation views that deviate significantly from the training views. TSDF Fusion and NeRF-SLAM suffer from blurry patterns and floaters, while PARF shows more robust rendering results.

evaluation of the mentioned baseline methods, we strictly follow the official hyper-parameters for a fair comparison. **InstantNGP [21]** is an extension of NeRF that enjoys fast convergence and rendering. We re-implement InstantNGP with pytorch-lightning and customized CUDA kernel functions [14]. As a SLAM system, **NeRF-SLAM** [29] has a mapping stage that incorporates InstantNGP with a depth render loss. Since localization is out of the scope of this paper, only the mapping stage of NeRF-SLAM is evaluated. For a fair comparison with PARF, we follow the same hyper-parameter setting for networks and optimization when training InstantNGP and NeRF-SLAM.

To compare the convergence speed of PARF and NeRF-SLAM, we evaluate both interpolation and extrapolation quality with an appropriate iteration interval for each method, which is shown in Fig. 1(b). Note that we only evaluate the effectiveness of the proposed hybrid representation in Fig. 1(b), so we assume all the observations are available from the start of training. We further evaluate the efficiency under incremental reconstruction setup in Fig. 7.

5.3. Evaluation

In this section, we report the results of quantitative and qualitative experiments on three datasets, which help validate the effectiveness of PARF in terms of rendering quality, convergence speed, as well as incremental reconstruction performance.

Quantitative evaluation. We evaluate the interpolation and extrapolation performance of baselines and PARF on the Replica dataset in Tab. 1. By comparing baselines with and without depth input, it can be found that geometric guidance generally enables better render quality by reducing geometry ambiguity, especially for extrapolation views. By implementing geometric guidance in a hybrid representation, PARF significantly outperforms all the baselines for both interpolation and extrapolation settings. Note that the hybrid representation of PARF helps improve the extrapolation ability by a large margin.

NeurMiPs shows worse performance than PARF because the plane-only representation is relatively hard to optimize and cannot accurately fit complex geometries (like chair legs and flowers). On the contrary, PARF enjoys a sound combination of primitive and non-primitive areas, which is more flexible and robust to the level of detail modeling.

Dataset	Methods	PSNR↑	SSIM↑	LPIPS↓
ScanNet	InstantNGP	21.04	0.685	0.530
	NeRF-SLAM	23.28	0.716	0.490
	PARF	**23.93**	**0.753**	**0.474**
BF	InstantNGP	21.42	0.724	0.460
	NeRF-SLAM	24.98	0.749	0.394
	PARF	**25.82**	**0.760**	**0.363**

Table 2: Quantitative evaluation on ScanNet and BundleFusion(BF) datasets.

From Tab. 2, we find that even the real-world datasets include noisy depth maps and imperfect poses, PARF still consistently outperforms InstantNGP and NeRF-SLAM on all metrics under the interpolation views, which proves the effectiveness of the proposed hybrid representation and primitive-aware fusion framework.

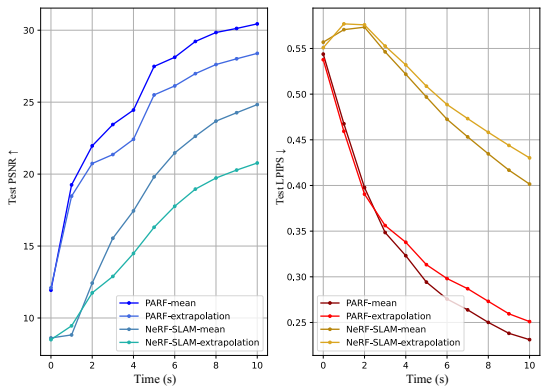


Figure 7: Incremental reconstruction evaluation.

Qualitative evaluation. Fig. 5 and Fig. 6 show the visual comparison on different datasets. The rendering details shows that our method achieves the highest rendering quality and robustness. In both figures, TSDF Fusion shows blurry rendering results because of the simple multi-view color averaging strategy. Because of the insufficient scene regularization, DS-NeRF and NeRF-SLAM shows floaters in textureless regions and extrapolation views, which results in an apparent performance drop. In Fig. 5, NeurMiPs suffers from distorted rendering results in the regions of complex geometry. Besides, NeurMips is very sensitive to plane initialization, which often results in holes due to the difficulty of plane parameter optimization. By comparison, the qualitative performance of PARF is more stable and accurate due to the advantage of the primitive-aware scene sensation and the hybrid representation.

Speed Analysis. The **convergence speed** evaluation results are shown in Fig. 1(b). With the help of the effective combination with primitive-based rendering, PARF enjoys the highest converge speed compared to all the learning-based baselines that claimed for fast convergence, including Plenoxels, DVGO, InstantNGP, and NeRF-SLAM.

The **incremental reconstruction** performance is also evaluated. By comparing to NeRF-SLAM in Fig. 1(a) and Fig. 7, PARF enjoys much faster convergence and can enable on-the-fly radiance fusion.

5.4. Ablation Study

Observation sparsity. Primitives provide vital prior for scene perception, which enables robust performance even when the observation is relatively sparse. We evaluate PARF and NeRF-SLAM with different input sparsity on `office0` of Replica. The sparsity n means we take one of every n images from the original sequence (2000 frames) as input. The results are shown in Tab. 3, which demonstrate the robust performance of PARF even with very limited views (< 20 views).

Sampling Strategy. We conducted an ablation study to validate the effectiveness of our primitive guided sampling strategy. Since depth maps are available, a more straightforward way is to sample with depth values. Specifically, if a ray holds a valid depth value on the depth image, we only sample the points located on and behind the depth value during training. However, depth values from sensors inevitably contain noise, which is harmful to direct depth guidance. We add Gaussian noise with mean $0cm$ and sigma $100cm$ to depth images. The experimental results on Replica `office0` demonstrate the robustness of the hybrid sampling strategy of PARF, while direct depth guidance shows massive quality degradation given noisy depth input.

Methods	Metrics	Sparsity			
		20	60	100	140
NeRF-SLAM	PSNR ↑	32.09	31.27	28.09	25.68
	SSIM ↑	0.919	0.908	0.879	0.831
	LPIPS ↓	0.274	0.287	0.316	0.366
PARF	PSNR ↑	**33.18**	**32.67**	**31.53**	**29.76**
	SSIM ↑	**0.934**	**0.923**	**0.911**	**0.905**
	LPIPS ↓	**0.192**	**0.201**	**0.204**	**0.214**

Table 3: Ablation study of observation sparsity.

Guidance	PSNR↑	SSIM↑	LPIPS↓
depth (w/o noise)	29.19	0.881	0.305
depth (w/ noise)	20.09	0.698	0.539
primitive (w/o noise)	**33.18**	**0.934**	**0.192**
primitive (w/ noise)	**33.00**	**0.933**	**0.196**

Table 4: Ablation study of sampling strategy on Replica.

6. Conclusion

In this paper, we introduce PARF, a **P**rimitive-**A**ware **R**adiance **F**usion method for indoor scene radiance field reconstruction and editing. By combining volumetric and primitive rendering in a hybrid neural representation, we successfully merge semantic parsing, primitive extraction, and radiance fusion into a single framework. PARF achieves significant improvement in convergence speed, strong view extrapolation performance, and realistic semantic editing effects simultaneously. Since the discrete semantic volume may lead to jagged primitive boundaries for novel view synthesis, future work includes combining the semantic information in a more compact manner and adding more kinds of primitives for more effective reconstruction.

Acknowledgements This work is supported in part by Natural Science Foundation of China (NSFC) under contract No.62125106, 61860206003, 62088102, 62171255, in part by Ministry of Science and Technology of China under contract No. 2021ZD0109901, in part by Tsinghua-Toyota Joint Research Fund.

References

[1] Marco Attene and Giuseppe Patané. Hierarchical structure recovery of point-sampled surfaces. *Computer Graphics Forum*, 29, 2010. 2

[2] 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

[3] Peter Carr, Yaser Sheikh, and Iain Matthews. Monocular object detection using 3d geometric primitives. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part I*, ECCV'12, page 864–878, Berlin, Heidelberg, 2012. Springer-Verlag. 2

[4] Zhiqin Chen, Thomas Funkhouser, Peter Hedman, and Andrea Tagliasacchi. Mobilenerf: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. *arXiv preprint arXiv:2208.00277*, 2022. 2, 3

[5] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 6

[6] Angela Dai, Matthias Nießner, Michael Zollöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface re-integration. *ACM Transactions on Graphics 2017 (TOG)*, 2017. 1, 6

[7] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022. 7

[8] Klaus Engel, Martin Kraus, and Thomas Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Workshop on Graphics Hardware*, HWWS '01, page 9–16, New York, NY, USA, 2001. Association for Computing Machinery. 1

[9] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5501–5510, 2022. 7

[10] Haoyu Guo, Sida Peng, Haotong Lin, Qianqian Wang, Guofeng Zhang, Hujun Bao, and Xiaowei Zhou. Neural 3d scene reconstruction with the manhattan-world assumption. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5511–5520, 2022. 2

[11] Dirk Holz, Stefan Holzer, Radu Bogdan Rusu, and Sven Behnke. Real-time plane segmentation using rgb-d cameras. In *Robot Soccer World Cup*, 2012. 2

[12] Mijeong Kim, Seonguk Seo, and Bohyung Han. Infonerf: Ray entropy minimization for few-shot neural volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12912–12921, 2022. 2

[13] J. Kniss, S. Premoze, C. Hansen, P. Shirley, and A. McPherson. A model for volume lighting and modeling. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):150–162, 2003. 1

[14] kwea123. ngp_pl. https://github.com/kwea123/ngp_pl, 2022. 8

[15] Florent Lafarge and Clément Mallet. Creating large-scale city models from 3d-point clouds: A robust approach with hybrid representation. *International Journal of Computer Vision*, 99:69–85, 2012. 2

[16] Zhi-Hao Lin, Wei-Chiu Ma, Hao-Yu Hsu, Yu-Chiang Frank Wang, and Shenlong Wang. Neurmips: Neural mixture of planar experts for view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15702–15712, 2022. 2, 3, 7

[17] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[18] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. *arXiv preprint arXiv:2008.02268*, 2020. 1, 2

[19] N. Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. 1

[20] 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. *arXiv preprint arXiv:2003.08934*, 2020. 2, 3, 5, 7

[21] 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. 2, 3, 6, 7, 8

[22] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015. 1

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

[24] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha 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*, pages 5480–5490, 2022. 2

[25] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[26] Nikola Popovic, Danda Pani Paudel, and Luc Van Gool. Neural radiance fields for manhattan scenes with unknown manhattan frame. *arXiv preprint arXiv:2212.01331*, 2022. 2

[27] Pedro F Proença and Yang Gao. Fast cylinder and plane extraction from depth cameras for visual odometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6813–6820. IEEE, 2018. 5

[28] Tahir Rabbani, Sander Dijkman, Frank van den Heuvel, and George Vosselman. An integrated approach for modelling and global registration of point clouds. *Isprs Journal of Photogrammetry and Remote Sensing*, 61:355–370, 2007. 2

[29] 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, 6, 7, 8

[30] Ruwen Schnabel, Roland Wahl, and R. Klein. Efficient ransac for point-cloud shape detection. *Computer Graphics Forum*, 26, 2007. 2

[31] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 501–518. Springer, 2016. 5

[32] 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. 6

[33] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5459–5469, 2022. 7

[34] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021. 2, 3

[35] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 2

[36] Yiming Xie, Matheus Gadelha, Fengting Yang, Xiaowei Zhou, and Huaizu Jiang. Planarrecon: Real-time 3d plane detection and reconstruction from posed monocular videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6219–6228, 2022. 2, 3

[37] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018. 1

[38] Haiyang Ying, Jinzhi Zhang, Yuzhe Chen, Zheng Cao, Jing Xiao, Ruqi Huang, and Lu Fang. Parsemvs: Learning primitive-aware surface representations for sparse multi-view stereopsis. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6113–6124, 2022. 2

[39] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *arXiv preprint arXiv:2206.00665*, 2022. 2

[40] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *CVPR*, 2017. 7

[41] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 2

[42] Xiaoshuai Zhang, Sai Bi, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Nerfusion: Fusing radiance fields for large-scale scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5449–5458, 2022. 2

[43] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15838–15847, 2021. 3

[44] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12786–12796, 2022. 2