

StructNeRF: Neural Radiance Fields for Indoor Scenes with Structural Hints

Zheng Chen, Chen Wang, *Student Member, IEEE*, Yuan-Chen Guo, Song-Hai Zhang, *Member, IEEE*,

Abstract—Neural Radiance Fields (NeRF) achieve photo-realistic view synthesis with densely captured input images. However, the geometry of NeRF is extremely under-constrained given sparse views, resulting in significant degradation of novel view synthesis quality. Inspired by self-supervised depth estimation methods, we propose StructNeRF, a solution to novel view synthesis for indoor scenes with sparse inputs. StructNeRF leverages the structural hints naturally embedded in multi-view inputs to handle the unconstrained geometry issue in NeRF. Specifically, it tackles the texture and non-texture regions respectively: a patch-based multi-view consistent photometric loss is proposed to constrain the geometry of textured regions; for non-textured ones, we explicitly restrict them to be 3D consistent planes. Through the dense self-supervised depth constraints, our method improves both the geometry and the view synthesis performance of NeRF without any additional training on external data. Extensive experiments on several real-world datasets demonstrate that StructNeRF surpasses state-of-the-art methods for indoor scenes with sparse inputs both quantitatively and qualitatively.

Index Terms—neural radiance fields, novel view synthesis, neural rendering.

1 INTRODUCTION

NOVEL view synthesis (NVS) for indoor scenes plays an important role in VR and AR applications, such as virtual navigation through buildings, tourist sites, and game environments. However, people often have to devote extensive efforts to collecting and processing large amounts of input data in order to produce satisfying results [11], [21], [25]. It remains to be a problem how to synthesize photo-realistic novel views given limited indoor images [5], [15], [22], [25], [35]. Recently, Neural Radiance Fields (NeRF) [21] emerges as a promising technique for NVS. NeRF uses a continuous multi-layer perceptron (MLP) to encode the radiance and density of a 3D scene and then synthesizes novel views through differentiable volumetric rendering. It achieves photo-realistic results even when representing some scenes with complicated geometry and appearance. Nevertheless, sparse indoor scene inputs bring several innate challenges to NeRF. Firstly, reconstructing the geometry and appearance of objects or scenes becomes an ill-posed problem with insufficient inputs. Even though NeRF can well fit the training images at the pixel level, the geometry is indeed inaccurate and leads to unsatisfying renderings at test viewpoints [5]. The necessity of "inside-out" view capture for indoor scene images exaggerates this issue [10]. Compared with "outside-in" viewing scenarios for outdoor scenes or standalone objects, adjacent views would have less overlap with each other given the same number of images [25]. Secondly, indoor scenes contain many textureless regions such as walls, floors, tables, and ceilings, making it hard for NeRF to find enough cross-view 3D

correspondences.

Several recent studies [5], [25], [33] leverage depth priors to improve the performance of NeRF in novel view synthesis. DSNeRF [5] adopts the sparse depth point cloud from COLMAP [26] to directly constrain the depth rendered by NeRF. However, the depth from Structure-from-Motion (SfM) is both sparse and noisy. Dense Depth Priors [25] further utilizes a depth completion network to predict dense depth maps, which are then used to guide the sampling and depth prediction of NeRF. However, the depth completion network introduces view inconsistency and generalization issues. To overcome these problems, we present StructNeRF, a technique that takes inspiration from recent self-supervised depth estimation methods [14], [36] and incorporates structural hints naturally contained in multi-view inputs, which turns into easy-to-adapt regularizations for NeRF geometry without any additional networks or data. StructNeRF considers the huge differences between textured and textureless regions and tackles them separately. Inspired by the insight of NeRF++ [37], we notice that the ability of NeRF to model the appearance of view-dependent effects leads to the ambiguity between its 3D shape and radiance (shape-radiance ambiguity). To reduce this ambiguity, we ensure that the same 3D region in different views is view-consistent by leveraging a patch-based multi-view consistent photometric loss based on depth warping. The resulting depth constraints are therefore dense and view-consistent. Patch-based photometric loss works well for textured regions, but it fails to discriminate non-textured regions that are common in indoor scenes, such as floor, walls, tables and ceiling. At the same time, we notice these regions are almost planes, so we further restrict them to be planar. To be more specific, we segment each input view into superpixels and group them as plane priors (Most superpixels are planes as shown in Fig. 4). Then the co-planar constraint [14] is applied to constrain the depth of those regions. Addition-

• Z. Chen , C. Wang, Y. Guo and S. Zhang are with the BNRist, Department of Computer Science and Technology, Tsinghua University, Beijing, China, 100084.
E-mail: chenz20@mails.tsinghua.edu.cn, cw.chenwang@outlook.com, guoyc19@mails.tsinghua.edu.cn, shz@tsinghua.edu.cn

Manuscript received April 19, 2005; revised August 26, 2015.

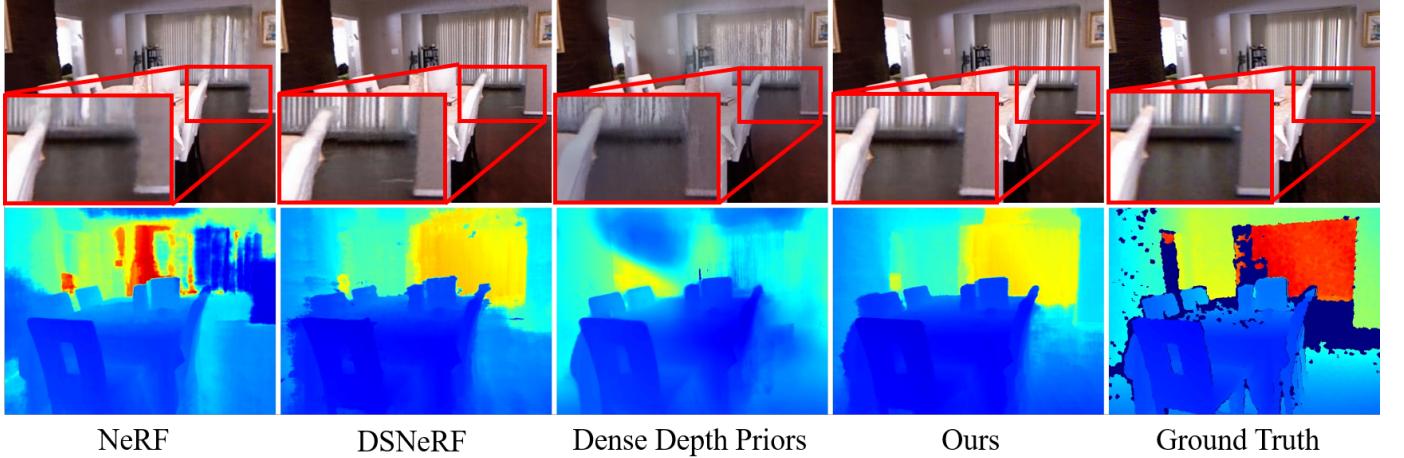


Fig. 1: Qualitative comparison of novel view synthesis and depth estimation given only a sparse set of indoor images. StructNeRF demonstrates superior performance over state-of-the-arts. We propose two structural hints: patch-based multi-view consistency at textured regions and planar consistency at non-textured regions, significantly improving the synthesis and geometry of radiance fields without any additional data or networks.

ally, we apply warm-up training to reduce the negative impact of noisy sparse point clouds from COLMAP, which helps StructNeRF to better utilize the sparse depth information. We evaluate our method on three indoor datasets: ScanNet [4], NYUv2 [28], SUN3D [34], using only sparse inputs. Results show that our proposed structural hints and training strategy together enable StructNeRF outperform existing per-scene training methods (only use data of target scene without additional training). Compared to methods that utilizes external data to train their depth estimation networks (we call data-driven methods), our method still shows comparable performance on their pretrained datasets and surpass them on other ones. A demonstration of the comparisons can be found in Fig. 1.

The contributions of our paper can be summarized as the following:

- 1) By leveraging the patch-based multi-view consistent photometric loss, StructNeRF obtains dense and view-consistent depth constraints, without the need for depth completion network trained on external data.
- 2) We re-project points in textureless regions into the 3D space and enforce them to be planes with the plane consistency loss. Therefore, the reconstructed planes are more flat and the rendering quality is also improved.
- 3) With the simple yet effective warm-up training strategy minimizing the noises coming from SfM reconstructed point clouds, we achieve state-of-the-art novel view synthesis performance given limited input views.

2 RELATED WORK

In this section, we briefly review Neural Radiance Field with sparse inputs and self-supervised depth estimation.

2.1 Neural Radiance Fields with Sparse Inputs

Based on implicit neural representations, Neural Radiance Fields (NeRF) [21] encoded 3D scenes into a continuous multi-layer perceptron (MLP) and achieved photorealistic

novel view synthesis. A growing number of NeRF extensions then emerged, e.g., reconstructing without camera poses [18], [32], modelling non-rigid scenes [23], [24], unbounded scenes [38], handling reflections [9], [29] and super-resolution [30]. When the scene is observed by sparse views, NeRF would however estimate a wrong density distribution, which is specifically reflected as some artifacts in the rendering process, such as "floaters". Here we give a detailed review of NeRF-based methods in both object-level and scene-level when the inputs are sparse.

Given sparse object-level views, several recent works [2], [3], [12], [13], [35] synthesized novel views using a pre-training with an optional per-scene optimization strategy. The pretrained network is however not suitable for indoor scenes due to the domain gap. Other methods impose regularizations on NeRF geometry, for example, RegNeRF [22] samples unobserved camera poses and regularizes patches rendered from those views with a depth smoothness loss and a trained normalizing flow model respectively. InfoNeRF [15] utilizes regularization based on information theory to improve view synthesis. However, both InfoNeRF [22] and RegNeRF [22] do not guarantee multi-view consistency. And all these object-level approaches never take into account the characteristics of the indoor scenes that we mentioned in Sec.1.

With regard to scene-level, recent studies like NerfingMVS [33], DSNeRF [5] and Dense Depth Priors [25] proposed to introduce depth priors to resolve the unconstrained geometry problem in NeRF from different aspects. Without additional network or training, DSNeRF [5] utilizes the sparse depth information from COLMAP [26] directly to constrain the depth rendered. NerfingMVS [33] instead trains a monocular depth estimation network to get scene-specific depth priors for guiding NeRF sampling. Similarly, Dense Depth Priors [25] leverages a pretrained depth completion network to predict dense depth maps for each view individually, which are then used to both supervise the rendered depth and guide NeRF sampling. However, there

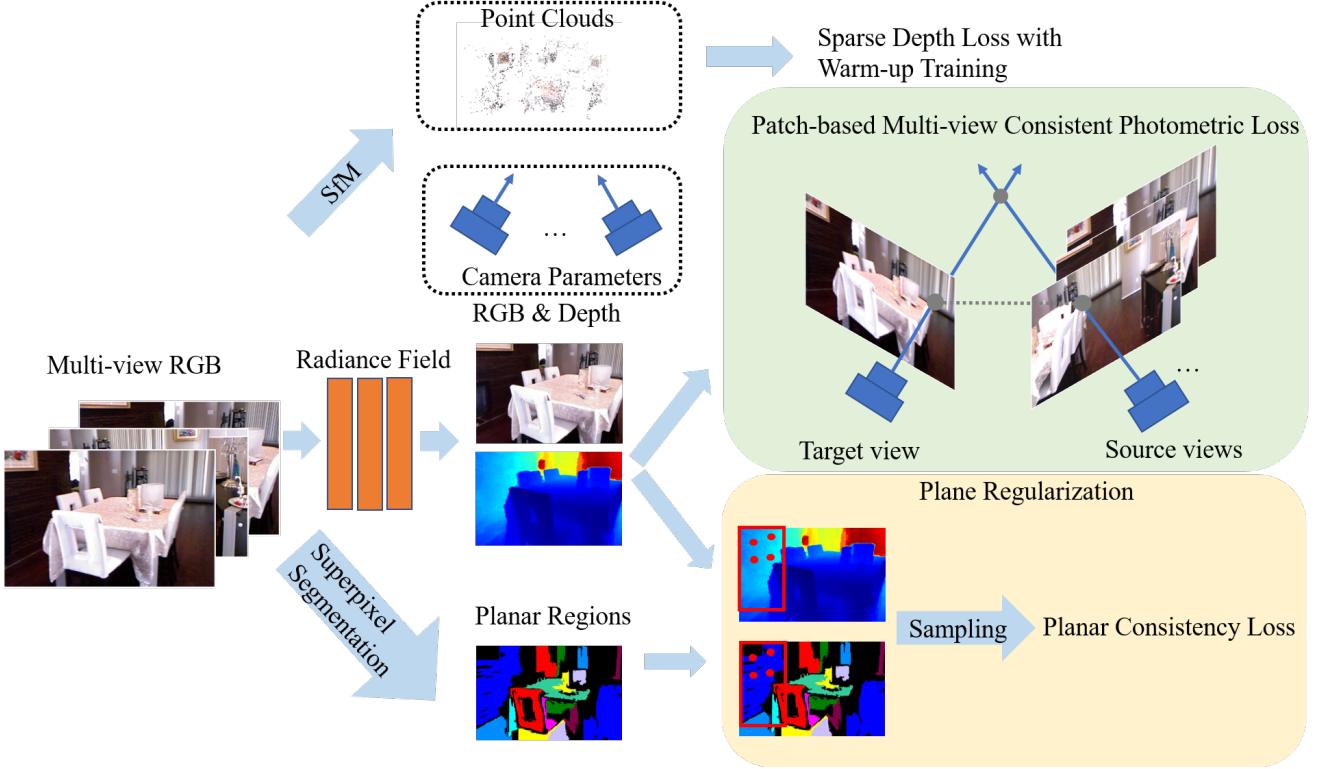


Fig. 2: We exploit the inherent structural hints in sparse views to improve the performance of NeRF in novel view synthesis. Firstly we utilize the structure-from-motion to obtain camera parameters and sparse point clouds. The sparse point clouds are used to constrain the depth of NeRF at keypoints featuring a warm-up training scheme. For regions with rich textures, we utilize the patch-based photometric loss for self-contained dense depth supervision. We also propose a planar consistency loss to regularize the depth of non-texture regions with the assistance of superpixel segmentation.

are two obvious problems in Dense Depth Priors. Firstly, the depth completion network is not view-consistent because each view is processed individually. Secondly, it also suffers from generalization issues as it relies on labeled training data such as ScanNet [4].

Compared with previous methods, StructNeRF leverages patch-based multi-view consistent depth loss to obtain dense supervision for NeRF without any depth completion network and additional data (unlike Dense Depth Priors [25]). Besides, we are the first to utilize a 3D planar consistency loss to further improve the quality of view synthesis in texture-less regions, which is also complementary to the multi-view patch-match loss.

2.2 Self-supervised Depth Estimation

Self-supervised depth estimation methods are proposed to ease the demand for large-scale labeled training data. SfMLearner [41] is a pioneering work that supervises the geometry estimations from a depth estimation network by photometric loss. To solve the issue of dynamic objects, optical flow methods are used to compensate for the moving pixels. Semantic masks provided by pretrained semantic segmentation models are also utilized to handle dynamic objects [20]. The approaches do not get satisfactory results in indoor scenes because they do not take into account the non-texture regions.

MovingIndoor [40] is the first self-supervised depth estimation approach focusing on indoor scenes. The authors

propose to use the sparse flow via matching with SURF [1] to initialize the optical flow estimation network, SFNet. In the training process, sparse flows are propagated from textured regions to non-textured regions through iterations and finally transformed into dense flows, which are then used to supervise the depth estimation network. P²Net [36] leveraged a patch-based multi-view consistency photometric error to constrain the depths. Other methods also adopt structural regularities such as co-planar constraints to improve depth estimations [14], [17], [36].

Motivated by these self-supervised indoor depth estimation approaches, we propose to utilize the structural hints naturally embedded in indoor scenes to constrain the depth of NeRFs, *i.e.*, the patch-based multi-view consistency loss from [36] and planar consistency loss from [14]. However, previous work mainly focus on the depth estimation task, we are the first to introduce these priors in NeRF and demonstrate that they can significantly resolve the unconstrained NeRF geometry issue and enable higher quality view synthesis.

3 METHOD

StructNeRF facilitates indoor novel view synthesis given only sparse input images, the framework of which is shown in Fig. 2. Firstly, we obtain sparse point clouds and camera parameters from Structure-from-Motion (SfM). We then incorporate self-supervised depth estimation methods into the

optimization of NeRF by imposing patch-based multi-view consistent photometric loss (Sec. 3.2) and planar consistency loss (Sec. 3.3). Lastly, we observe that while point clouds from SfM could serve as sparse depth priors for NeRF, it suffers from noisy estimation, for which we adopt a warm-up training strategy to gradually decay its contribution to the entire optimization (Sec. 3.4). Before introducing our method, we briefly revisit NeRF [21] in Sec. 3.1.

3.1 Preliminaries

Neural Radiance Fields (NeRF) represents a scene as a continuous neural volume using a multi layer perceptron (MLP) $f_\theta : (\mathbf{x}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma)$ with θ as the learnable parameters, where $\mathbf{x} \in \mathbb{R}^3$ and \mathbf{d} denotes a 3D position and the view direction, $\sigma \in \mathbb{R}$ and $\mathbf{c} \in \mathbb{R}^3$ the corresponding density and radiance.

NeRF is an emission-only model, which means the color of a pixel only depends on the radiance along a ray with no other lighting factors. Therefore, according to volume rendering, the color along the camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ that shots from the camera center \mathbf{o} in direction \mathbf{d} can be approximated by numerical quadrature:

$$\hat{\mathbf{C}}(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(t)\mathbf{c}(t)dt, \quad (1)$$

where $T(t) = \exp(-\int_{t_n}^t \sigma(s)ds)$.

NeRF is optimized by sampling random rays from all training images and minimizing the rendered and ground truth pixel color in L2 norm:

$$L_{\text{Color}} = \sum \|\hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r})\|_2^2 \quad (2)$$

3.2 Patch-based Multi-view Consistency

The ability of NeRF to model the appearance of view-dependent appearance leads to the ambiguity between its 3D shape and radiance [37]. To reduce the shape-radiance ambiguity, we leverage multi-view consistency explicitly to supervise the depth of every pixel for each view.

To begin with, we render the depth of a given pixel with the formulation proposed in the original NeRF paper. The depth of the ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ can thus be calculated as the following:

$$\hat{\mathbf{D}}(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(t)tdt, \quad (3)$$

After we sample points p_i^t from the target view I_t , the original point-based warping process back-projects the extracted points to the source views I_s by,

$$p_i^{t \rightarrow s} = KM^s M^{t^{-1}} (\hat{\mathbf{D}}(p_i) \odot (K^{-1} p_i^t)), \quad (4)$$

where K denotes camera intrinsic parameters. M_s and M_t are the camera extrinsic parameters of the source view I_s and the target view I_t respectively. $\hat{\mathbf{D}}(p_i)$ is the rendered depth at the pixel p_i . \odot represents Hadamard Product.

Nevertheless, point-based representation is not discriminative enough and may cause false matching because many pixels have the same intensity values in an image. Similar to traditional SLAM pipelines [6] and self-supervised depth estimation [36], we define a *support domain* Ω_{p_i} as the local

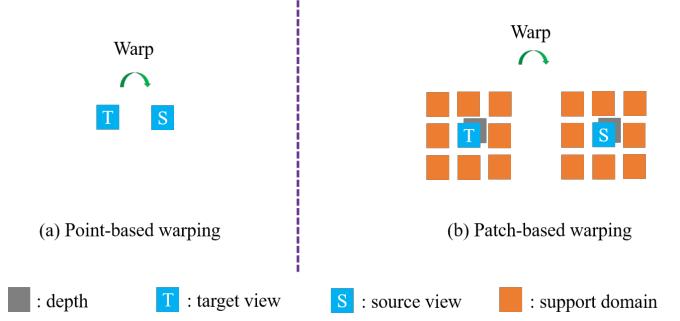


Fig. 3: The point-based and patch-based warping operations. Point-based operation warps pixel-by-pixel and suffers from severe false matching. However, patch-based operation warps a pixel together with its support domain from the target view to the source view, leading to more robust representations in the depth estimation task. This figure is adopted from [36].

window surrounding the sampled point p_i . Photometric loss is calculated over each supported domain instead of an isolated point, with which the depth of NeRF can be more accurate because the sampled points combined with their support domains are more unique (Fig. 3).

In each support domain, the depth of every point is assumed to be the same. We then use the same depth (the depth of the sampled point in the center of each patch), rendered by a neural radiance field, to warp the sampled point together with its support domain to other source views from the target view by,

$$\Omega_{p_i}^{t \rightarrow s} = KM^s M^{t^{-1}} (\hat{\mathbf{D}}(p_i) \odot (K^{-1} \Omega_{p_i}^t)), \quad (5)$$

where $\hat{\mathbf{D}}(p_i)$ denotes the predicted depth of p_i , which is equal to $\hat{\mathbf{D}}(\mathbf{r})$. The ray \mathbf{r} is emitted to the pixel p_i from camera center.

$$\Omega_p = \{(x + x_p, y + y_p), x_p \in \{-N, 0, N\}, y_p \in \{-N, 0, N\}\} \quad (6)$$

With more robust depth warping and cross-view matching enabled by the support domain, like [8], [36], we propose to adopt a photometric loss over the support domain Ω_{p_i} , which is the combination of an L1 loss and a structure similarity loss SSIM [31].

$$L_{\text{SSIM}} = \text{SSIM}(I_t[\Omega_{p_i}^t], I_s[\Omega_{p_i}^{t \rightarrow s}]) \quad (7)$$

$$L_{\text{L1}} = \|I_t[\Omega_{p_i}^t] - I_s[\Omega_{p_i}^{t \rightarrow s}]\|_1 \quad (8)$$

$$L_{\text{ph}} = \alpha L_{\text{SSIM}} + (1 - \alpha) L_{\text{L1}}, \quad (9)$$

where *support domain* Ω_{p_i} is defined as the local window surrounding the sampled point p_i . $I_t[\Omega_{p_i}^t]$ defines the pixel values at $\Omega_{p_i}^t$ in the target view I_t via a bilinear interpolation. $\Omega_{p_i}^{t \rightarrow s}$ defines the region after warping the support domain $\Omega_{p_i}^t$ from the target view I_t to the source view I_s . And α is a weighting factor that is set to 0.85 empirically. By definition, L_{ph} is patch-based and multi-view consistent.

Dense depth constraints are proved to be more beneficial to the geometry of neural radiance fields [25]. Therefore, unlike P²Net [36], we sample points directly from the whole image instead of the keypoints [6]. Our experiments also show that dense sampling results in better performance than that of sampling from keypoints (See Sec. 4.5). More importantly, in contrast to Dense Depth Priors [25], we achieve dense depth constraints free of any depth completion network which relies on external dataset training and have potential generalization problem.

3.3 Planar regularization with Superpixels

Although patch-based photometric loss works well for textured regions, it fails to discriminate non-textured regions that are common in indoor scenes, such as floor, walls, tables and ceiling. We further observe that those non-textured regions are mostly planar. Therefore, how to inform StructNeRF of the planar constraints of a scene is the core concern.

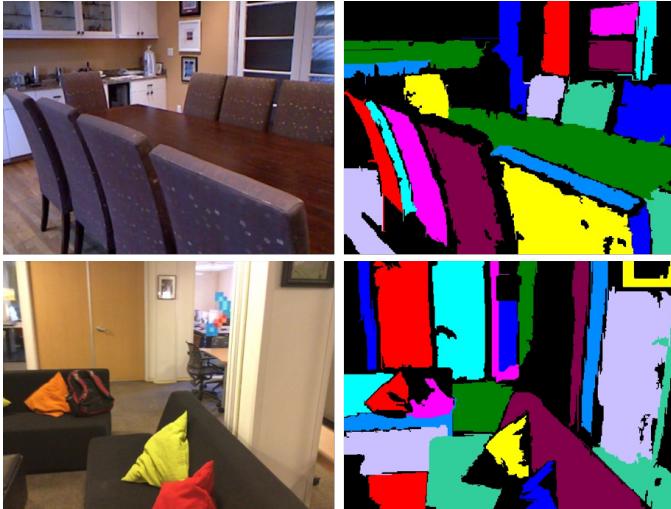


Fig. 4: Superpixel extraction (right) of two indoor images (left), colors represent different regions. We can see that most extracted regions are planes.

Inspired by self-supervised depth estimation [14], [36], we aim to first identify 2D planes in input images by adopting the Felzenszwalb superpixel segmentation algorithm [7]. Specifically, we extract superpixels from each view and define regions with area larger than a threshold as planes (We set it to be 1000 pixels in our experiments empirically) because those non-textured regions often span over a larger area. Fig. 4 provides examples that most of the segmented regions are planes.

Without specific regularization, NeRFs may fail to preserve the planar properties across different views, i.e. the depth map of planar regions is not flat. We propose to further impose the planar constraint to StructNeRF for non-textured regions using the planar consistency loss [14]. From each plane, we randomly sample 4 pixels, i.e., a, b, c and d . With the rendered depth of StructNeRF, we then transform them to 3D points A, B, C , and D in the camera coordinate with the following equation:

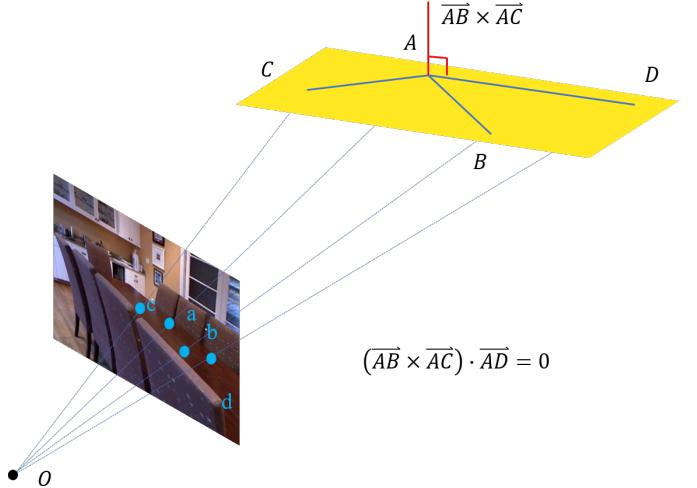


Fig. 5: The camestration process. We first re-project points a, b, c, d in a 2D plane to the 3D coordinates, then enforce the cross product of \overrightarrow{AB} and \overrightarrow{AC} to be perpendicular to \overrightarrow{AD} . This figure is inspired by [36].

$$P = \hat{\mathbf{D}}(p_i) \odot (K^{-1}p_i), p_i \in \{a, b, c, d\}, P \in \{A, B, C, D\}, \quad (10)$$

where K denotes the matrix of camera intrinsic parameters. p_i is the selected pixel and P is the corresponding 3D point.

As shown in Fig. 5, the cross product of \overrightarrow{AB} and \overrightarrow{AC} should be perpendicular to the plane where A, B, C and D is located. Therefore, \overrightarrow{AD} should be perpendicular to $\overrightarrow{AB} \times \overrightarrow{AC}$. The planar consistency loss is computed by,

$$L_{pc} = \frac{1}{N_p} \sum_{i=1}^{N_p} \left| \overrightarrow{A_iB_i} \times \overrightarrow{A_iC_i} \cdot \overrightarrow{A_iD_i} \right|, \quad (11)$$

where N_p denotes the number of 4-point sets we randomly select from planes.

As shown in the experiments, StructNeRF achieves better performances both in terms of depth estimation and view synthesis for planar regions with the proposed plane regularization.

3.4 Training

As introduced in DS-NeRF [5], we also leverage the depth of sparse keypoints extracted by COLMAP [26], [27] to supervise the geometry of the neural radiance field, which is view-consistent in nature.

$$L_{sparse} = \sum_{x_i \in \mathbf{X}_j} w_i \left| \hat{\mathbf{D}}(\mathbf{r}_{ij}) - (\mathbf{M}_j x_i) \cdot [0, 0, 1]^T \right|^2, \quad (12)$$

where the keypoint in camera j is reprojected using camera extrinsic parameters \mathbf{M}_j and then is projected onto its unit camera axis $[0, 0, 1]$. We also introduce a hyperparameter w_i to adaptively adjust the weights of keypoints to reduce the negative influence of unreliable keypoints, determined by the reprojected error from COLMAP estimation. Supposing a 3D keypoint x_i is visible in camera j , the reprojected error e_{ij} is the distance in pixels between the camera coordinates

KM_jx_i and detected 2D keypoint in camera j . Thus, the confidence weight of a keypoint can then be measured by the total reprojected error $e_i = \sum_j e_{ij}$,

$$w_i = \exp\left(-\left(\frac{e_i}{\bar{e}}\right)^2\right), \quad (13)$$

where \bar{e} denotes the mean absolute error of all the keypoints in a scene.

We optimize the depth of neural radiance field by the weighted combination of patch-based multi-view consistency photometric loss, planar consistency loss and sparse depth loss from COLMAP as follows:

$$L_{\text{Depth}} = \lambda_{ph} L_{ph} + \lambda_{pc} L_{pc} + \lambda_{sparse} L_{sparse} \quad (14)$$

The overall loss function of StructNeRF is therefore:

$$L_{\text{Total}} = L_{\text{Depth}} + L_{\text{Color}}, \quad (15)$$

Warm-up Training. During our training, we found that when we use fixed weights for L_{sparse} over all iterations, the results were disturbed by the inaccurate points (See the noisy floaters in the depth map of the second column in Fig. 14). Naively reducing L_{sparse} would only diminish the benefits of the accurate points, so we adopt a warm-up training strategy. Specifically, we introduce the sparse depth loss item L_{sparse} only in the first half of the training ($\lambda_{sparse} = 0.05$). In the remaining iterations, we set $\lambda_{sparse} = 0$ and let L_{ph} and L_{pc} refine the depths of pixels where noisy points are located. By setting $\lambda_{sparse} = 0$, the noisy point clouds will not have an impact on NeRF anymore in the later training process. With warm-up training, we strike a balance of utilizing the sparse depth priors and avoiding noises of point clouds.

4 EXPERIMENTS

4.1 Evaluation Metrics

We use peak signal-to-noise (PSNR), the structural similarity index measure (SSIM) [31] and the learned perceptual image patch similarity (LPIPS) [39] to measure the quality of synthesized RGB novel views by comparing them with the ground truth. Besides, we also include two other metrics to demonstrate the effectiveness of our reconstructed geometry over previous methods. We use depth root-mean-square error (Depth RMSE) for measuring the predicted depth maps and the ground truth. Also, the Plane Mean Deviation is used to evaluate the flatness of planes for the predicted depth [14]. It is defined the distance of the measured point cloud to the fitted plane. We use the mean deviation to measure the flatness of planes for the predicted depth. Since the depth from NeRF is in a relative scale, different from the absolute ground truth depth from sensors. We therefore align the predicted depth to the ground truth according to conventional practice [42] because the deviation is scale-variant.

4.2 Datasets

To evaluate our method, We train and test our model on three multi-view indoor scene datasets in terms of novel view synthesis: ScanNet [4], NYUv2 [28] and SUN3D [34]. We use only the RGB data for training and the ground truth depth are only used for evaluation.

For each scene of the datasets above, we run COLMAP [26] over no more than 28 frames to obtain the poses and point clouds. We use eight sample scenes from each dataset respectively. Among the sampled frames, 8-th, 16-th and 24-th frames are used as the test views, and the rest are used as the training views.

For ScanNet [4] and SUN3D [34], the image resolution is 468×624 after we down-sample and crop the dark borders from calibration. For NYUv2 [28], the image resolution is 545×415 .

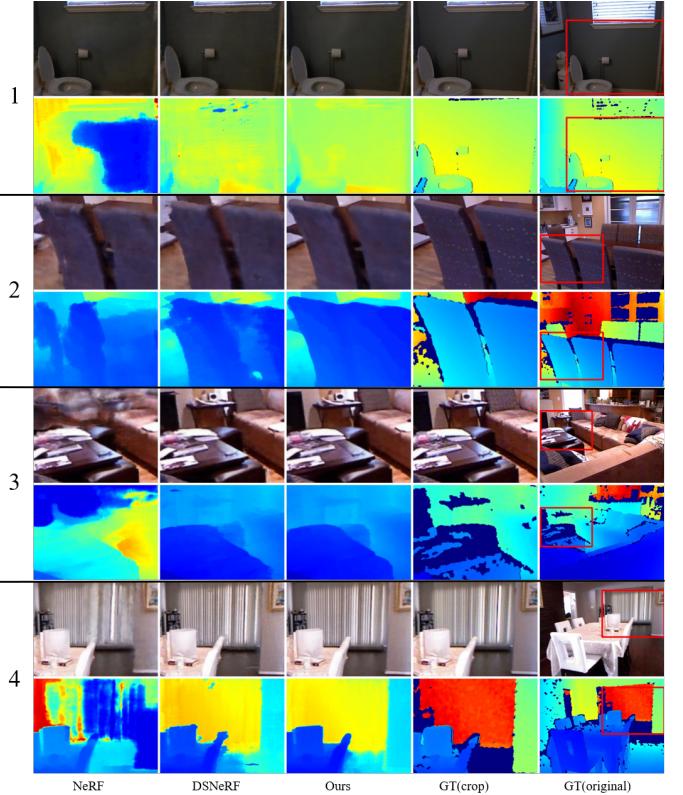


Fig. 6: Comparisons with per-scene optimization method on the test views from NYUv2 scenes. "GT" denotes Ground Truth. "GT (original)" and "GT (crop)" mean the original and cropped ground truth respectively.

4.3 Implementation Details

We set $\lambda_{ph} = 0.025$ and $\lambda_{pc} = 0.025$ and $\lambda_{sparse} = 0.05$ in all the scenes of each dataset. N is set to 2, and we take the previous two frames and the posterior two frames of the target view as the source views, which is the same as [36]. We use the Adam optimizer [16] with learning rate 0.0005 and sample rays in batches of 1024. The radiance field is optimized for 100k iterations. We use the same MLP architecture in all experiments as NeRF [21] for fair comparison.

4.4 Comparison with Existing Methods

We compare StructNeRF with existing methods for novel view synthesis given sparse inputs in the indoor scenes. The baselines include two categories: the methods trained by per-scene optimization without external data and the

Dataset	Method	Additional Training?	PSNR↑	SSIM↑	LPIPS↓	Depth RMSE↓	Plane Mean Dev↓
NYUv2	NeRF [21]	No	24.88	0.7899	0.2304	1.1512	0.0390
	DSNeRF [5]	No	<u>27.33</u>	<u>0.8320</u>	0.1943	<u>0.3604</u>	0.0347
	Dense Depth Priors [25]	Yes	26.34	0.8219	0.1774	0.5640	0.0250
	Ours	No	28.10	0.8561	0.1663	0.3113	<u>0.0301</u>
SUN3D	NeRF [21]	No	17.99	0.6114	0.4451	1.6153	0.0375
	DSNeRF [5]	No	<u>22.26</u>	0.7027	0.3430	<u>0.5761</u>	0.0386
	Dense Depth Priors [25]	Yes	20.91	<u>0.7119</u>	0.3317	0.6866	0.0271
	Ours	No	22.89	0.7627	0.2748	0.5287	<u>0.0327</u>
ScanNet	NeRF [21]	No	23.54	0.8017	0.2779	0.5527	0.0372
	DSNeRF [5]	No	24.25	0.8190	0.2657	0.2500	0.0326
	Dense Depth Priors [25]	Yes	24.84	0.8137	0.2578	0.1929	0.0155
	Ours	No	<u>24.67</u>	<u>0.8308</u>	<u>0.2480</u>	0.2298	0.0270

TABLE 1: Comparisons with other methods. The performance of the thickened is the best. The underlined ranks the second in performance.

Method	Additioanl training?	PSNR↑	SSIM↑	LPIPS↓	Depth RMSE↓	Plane Mean Dev↓
w/o dense-sampling	No	27.18	0.8324	0.1822	0.3246	0.0329
w/o patch	No	28.03	0.8545	0.1671	0.3255	0.0312
w/o warm-up training	No	27.60	0.8456	0.1765	0.3382	0.0329
w/o sparse depth priors	No	27.42	0.8505	0.1734	0.4052	0.0350
w/o patch-match	No	27.56	0.8479	0.1772	0.3236	0.0300
w/o plane regularization	No	27.92	0.8498	0.1705	0.3308	0.0321
full	No	28.10	0.8561	0.1663	0.3113	0.0301

TABLE 2: Ablation studies on NYUv2 datasets. The performance of the thickened is the best.

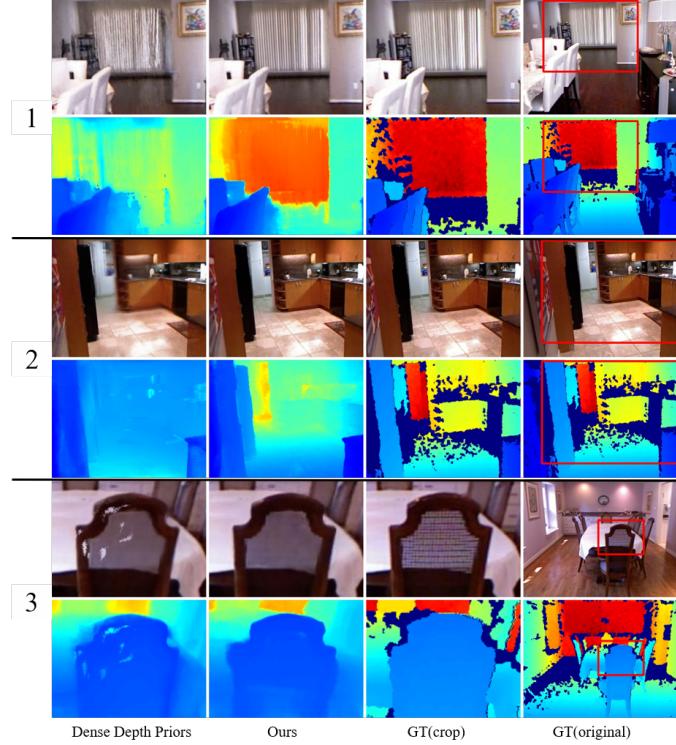


Fig. 7: Comparisons with data-driven method on the test views from NYUv2 scenes. "GT" denotes Ground Truth. "GT(original)" and "GT(crop)" mean the original and cropped ground truth respectively.

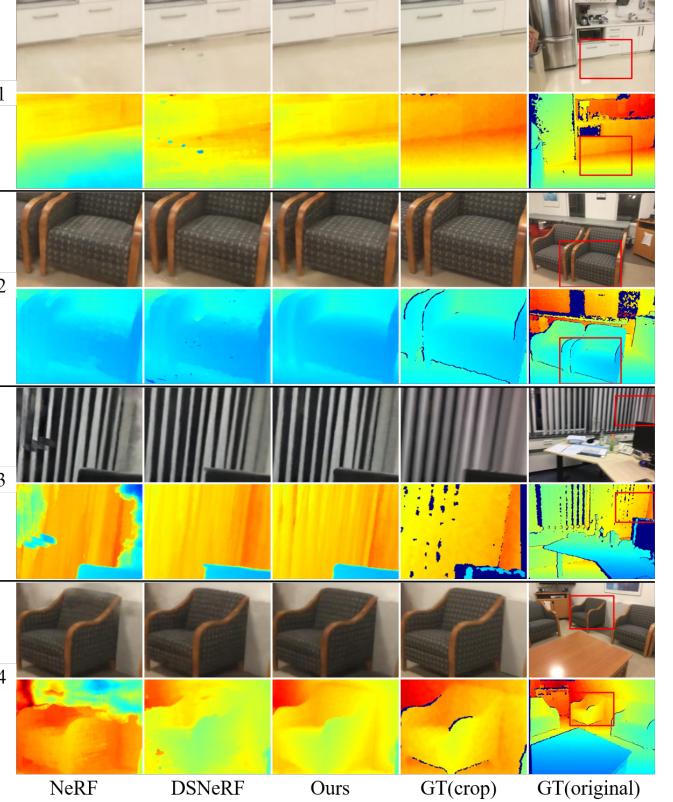


Fig. 8: Comparisons with per-scene optimization method for the test views from ScanNet scenes. "GT" denotes Ground Truth. "GT(original)" and "GT(crop)" mean the original and cropped ground truth respectively.

methods with data-driven depth priors (we call data-driven methods).

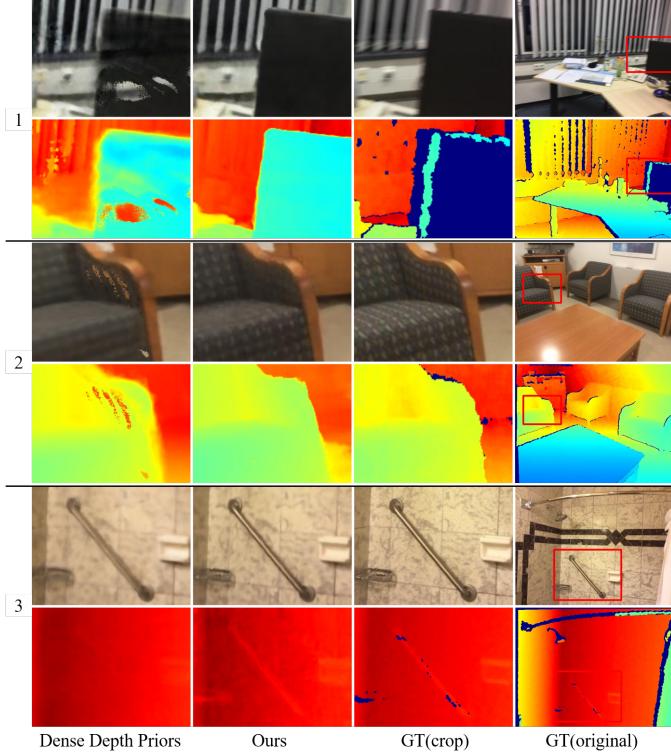


Fig. 9: Comparisons with data-driven method for the test views from ScanNet scenes. "GT" denotes Ground Truth. "GT(original)" and "GT(crop)" mean the original and cropped ground truth respectively.

4.4.1 Comparison with per-scene optimization Methods

Per-scene optimization methods include vanilla NeRF [21] and DSNeRF [5] that uses sparse point clouds from SfM for depth supervision. We ran NeRF, DSNeRF, and StructNeRF on the three datasets, ScanNet, SUN3D and NYU. The quantitative results (as shown in Tab. 1) shows that our method outperforms NeRF and DSNeRF in all the metrics. The visualized results of comparisons on ScanNet and NYUv2 are listed in Fig. 8 and Fig. 6. The visualized results of comparisons on SUN3D can be seen in the appendix. NeRF produces the worst results because its geometry is extremely unconstrained (See Fig. 8 and Fig. 6). DSNeRF has only sparse depth priors from COLMAP, it often produces artifacts in the depth-unconstrained areas. Wrong color and geometry are produced by DSNeRF, e.g., visible in the chairs (Example 2 in Fig 6). In contrast, StructNeRF renders more accurate depths and colors because StructNeRF learns two structural hints which supervise the geometry of NeRF at textured and non-texture regions respectively. Besides, We found that StructNeRF is more robust to the outliers in the sparse depth input in the edge of the window (Example 1 in Fig. 6), while the unnecessary floaters are very obvious for DSNeRF.

4.4.2 Comparison with Data-driven Methods

We also compare our method to the recent work (Dense Depth Priors [25]) that uses the depth completion network to complete depths from sparse depth inputs and then uses them to supervise the geometry of NeRF. Different from

our method, the depth completion network of Dense Depth Priors needs to be pretrained on a large-scale indoor dataset ScanNet in a supervised way. The completed dense depths for new scenes are pre-calculated by the depth completion network and used to supervise the depth of NeRF when training NeRF on the new scenes. Like NeRF [21], our method is only optimized on the specific scene without additional training. Note we didn't include a comparison with another relevant work NerfingMVS [33] because it fails on most scenes for reasons we explained in the appendix.

The results (Tab. 1) show that, with the dataset prior, Dense Depth Priors is better than StructNeRF on ScanNet in terms of Depth RMSE. But StructNeRF still has comparable PSNR and even better SSIM and LPIPS compared with Dense Depth Priors. We also find that our method is more view-consistent than Dense Depth Priors (shown in Example 1 and 2 of Fig. 9 because the predicted depths of the depth completion network are not view-consistent. Besides, we found that the depth estimations from data-driven models for Dense Depth Priors shows less detailed depth and color (such as the metal handrail in Example 3 of Fig. 7).

Moreover, the performance of Dense Depth Priors degrades significantly on SUN3D and NYUv2 as shown in Tab. 1, mainly due to the generalization issue of the depth completion network. Fig. 7 shows that our method doesn't suffer from the generalization problem and surpasses Dense Depth Priors in every metric other than the Plane Mean Dev. The visualized results of comparisons on SUN3D can be found in the appendix.

4.4.3 Implementation Details of Baselines

DSNeRF The depth loss weight was set to 0.1 in all the scenes of each dataset as recommended in [5].

Dense Depth Priors The depth loss weight was set to 0.008 in all the scenes of each dataset as recommended in [25]. The depth completion network is pretrained on ScanNet [4]. We use the official released weights from Github¹ to predict the dense depth maps in all of our experiments. The radiance field of Dense Depth Priors is trained on each scene with the supervision of completed depth maps.

4.5 Ablation Study

We conduct ablation studies on NYUv2 to further validate the effectiveness of the different components in StructNeRF for view synthesis and depth estimation. The quantitative results can be found in Table 2.

Patch-match Omitting patch-match leads to inaccurate depth and color in high-frequency areas. The ability of NeRF to model the appearance of view-dependent appearance leads to the ambiguity between its 3D shape and radiance [37]. With multi-view consistency, patch-match improves the geometry of textured regions and reduces the artifacts in the edges such as the edges of the billboard shown in Fig. 10. When it is omitted, the black edge appears in a wrong place and its shape is wrongly estimated because it lacks the corresponding geometry constraints. L_{sparse} only provides the depth constraints at sparse keypoints

1. https://github.com/barbararoessle/dense_depth_priors_nerf

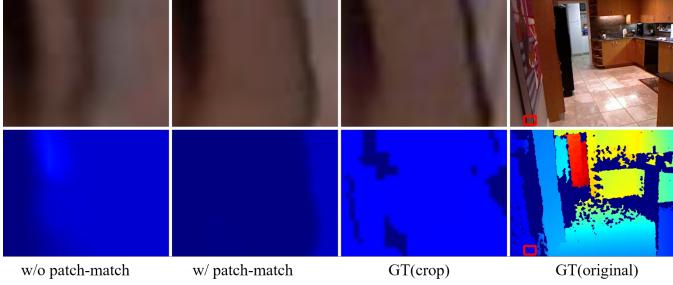


Fig. 10: w/o patchmatch and w/ patchmatch.

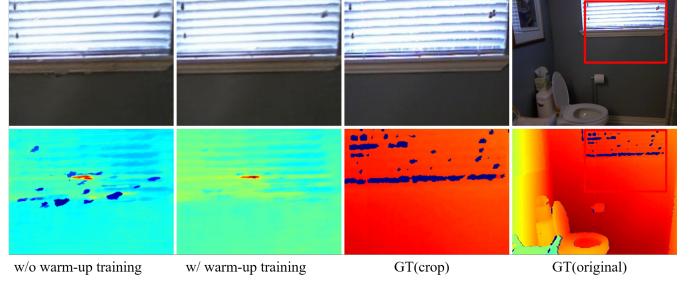


Fig. 14: w/o warm-up training and w/ warm-up training.

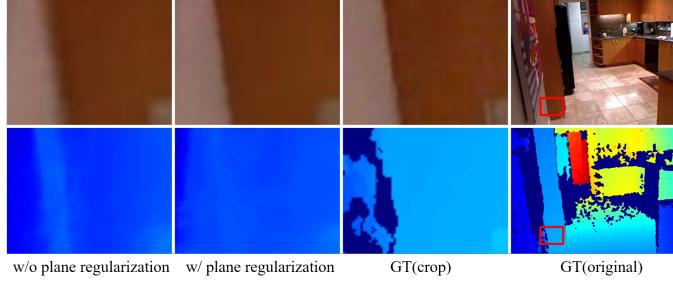


Fig. 11: w/o plane regularization and w/ plane regularization.

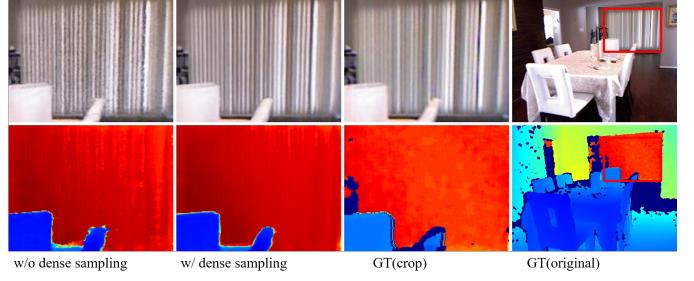


Fig. 15: w/o dense sampling and w/ dense sampling.

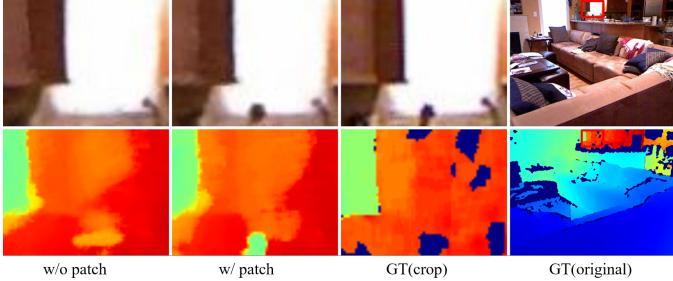


Fig. 12: w/o patch and w/ patch.

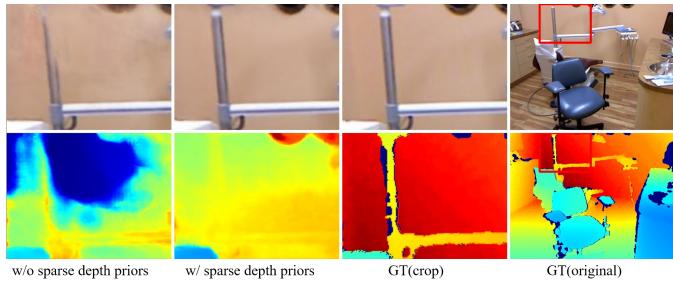


Fig. 13: w/o sparse and w/ sparse.

and L_{pc} at non-textured regions. Patch-based multi-view consistency is necessary for the textured regions other than keypoints.

Plane Regularization Removing plane regularization causes the geometry of textureless regions becomes less constrained, leading to less sharp edges in RGB, as shown in Fig. 11. As a result, the Depth RMSE and Plane Mean Dev degrades. In detail, after we remove the plane regularization, the left white area is mistaken for two planes and the colored edge between the white and brown area

is less clear. Since the white area corresponds to a plane in StructNeRF represented by a superpixel, our proposed planar consistency loss enforces flat geometry in this region, which reduces the artifacts in both the color and depth. It makes up for the insufficiency of patch-match in the non-texture regions.

Patch-based vs. Point-based Photometric Loss We replace the patch-based multi-view consistency photometric loss with the point-based one. It can be easily seen from Fig. 12 that patch-based loss leads to a robust rendering quality (the tiny bottle disappears in the w/o patch setting).

Sparse Depth Priors Excluding the sparse depth priors, we observe that NeRF is more likely to fall into the local optimal as shown in Fig. 13. Therefore, although sparse depth priors from COLMAP contain many noises, our experiments show that they are still indispensable for NeRF with sparse views.

Warm-up Training Omitting the warm-up training strategy and using the same λ_{sparse} across all iterations cause the noises of point clouds to be much more obvious in RGB and depth maps (See the floater artifacts in Fig. 14). The artifacts are perfectly removed with the warm-up strategy.

Dense-sampling In this experiment, we sample at the keypoints extracted by [6] in patch-match to supervise the depth of NeRF. We observe that sparse keypoint sampling leads to under-constrained depth and worse rendering results (as shown in Fig. 15). Dense sampling instead helps patch-match supervise the geometry of most regions.

5 CONCLUSION AND FUTURE WORK

This paper proposes StructNeRF, neural radiance field with self-supervised depth constraints for indoor scene novel view synthesis with sparse input views. We are the first to apply structural hints from multi-view inputs to NeRF for

view synthesis and geometry estimation, specifically, patch-match and plane regularization to constrain the depth of textured and textureless regions respectively. In this way, it learns a view-consistent geometry with dense depth constraints. Most importantly, we doesn't have the generalization problem which occurred in data-driven methods, e.g., Dense Depth Priors [25]. Besides, we adopt a warm-up training strategy to reduce the influence of noisy point clouds from Structure-from-Motion. StructNeRF outperforms state-of-the-arts without additional training [5], [21] both in depth estimation and novel view synthesis. In terms of comparison to data-driven methods, *i.e.*, Dense Depth Priors [25], it still achieves comparable performance on the pretrained dataset (ScanNet) and superior performance on other datasets (SUN3D and NYUv2). StructNeRF raises the upper bound of rendering quality of NeRF without external data given sparse input views. Our work also motivates future research to further exploit the structural hints in multi-view inputs for view synthesis and other related tasks.

Limitations of StructNeRF include limited view-dependent effects since the surfaces are observed by only a few input views, which also happens in other baselines [5], [21], [25]. Also, our method in plane reconstruction is still slightly inferior to supervised data-driven methods, albeit we already surpassed per-scene optimization ones significantly. In the future, We will consider how to model the view-dependent effect in the sparse input setting. We would also investigate how to incorporate the rich priors of indoor datasets, possibly with a more generalized NeRF trained across large-scale datasets.

APPENDIX A ADDITIONAL RESULTS

More visualized results for comparisons on SUN3D can be found in Fig 16 and Fig 17.

A.1 Comparisons with NerfingMVS [33]

NerfingMVS [33] used an error map to guide NeRF sampling. For each input view, 3D points would be back-projected to the remaining views according to the depth prior. Then, NerfingMVS [33] calculated the depth reprojection error and defined error map as the mean of the top-4 smallest errors.

This procedure works fine in original NerfingMVS [33] data, which focus on local region depth estimation. However, when applying it to room-scale inputs, 3D points of projection from other views often falls behind the camera. The calculation of near and far planes of such scenes would result in negative values, and the far boundary even appears in front of the near boundary, leading to invalid sampling range [25].

We use the official code of NerfingMVS² and find that it failed on all the scenes in ScanNet and Sun3D, only success on three scenes in NYUv2: bathroom_0046, kitchen_0026a and living_room_0033. The quantitative and qualitative comparisons on these scenes are shown in Tab. 3 and Fig. 18 respectively.

2. <https://github.com/weiyithu/NerfingMVS>

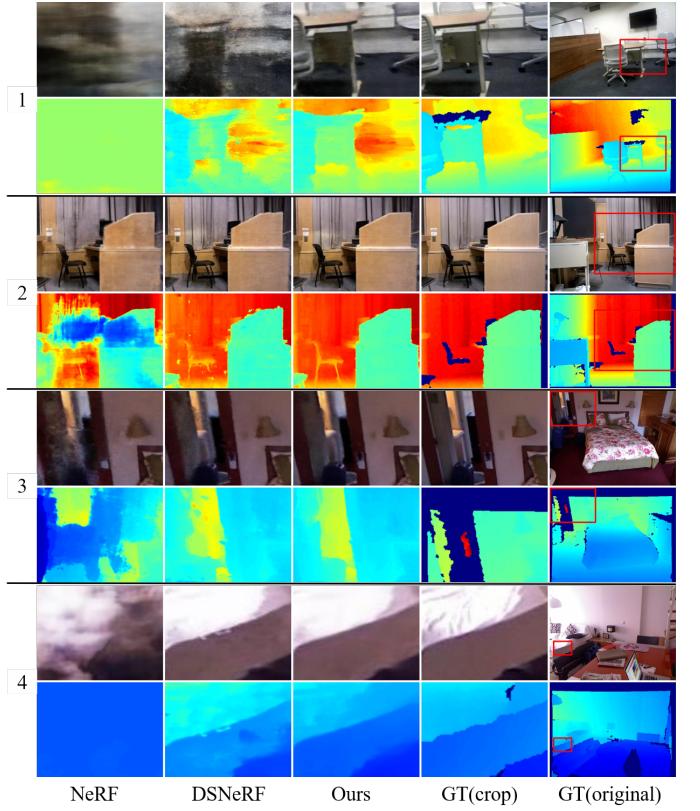


Fig. 16: Comparisons with method without pretraining on SUN3D.

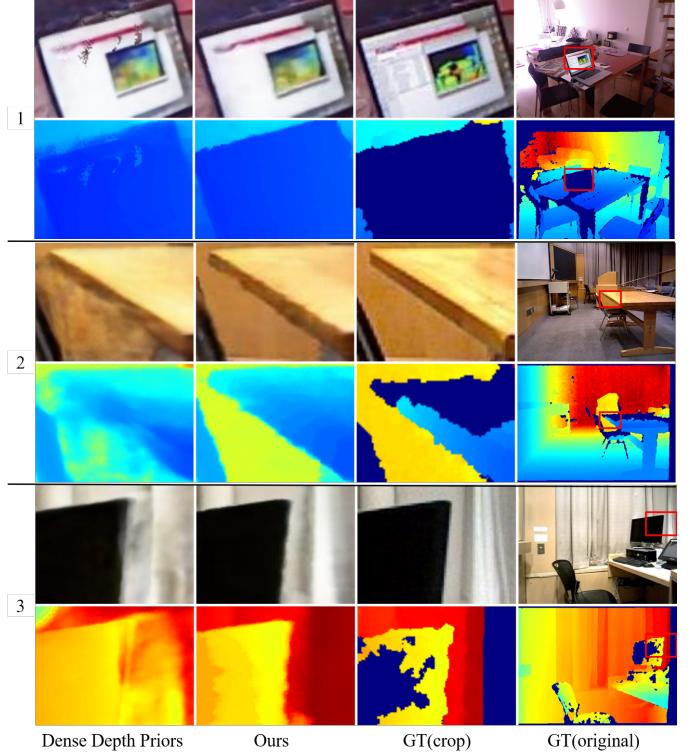


Fig. 17: Comparisons with method with pretraining on SUN3D.

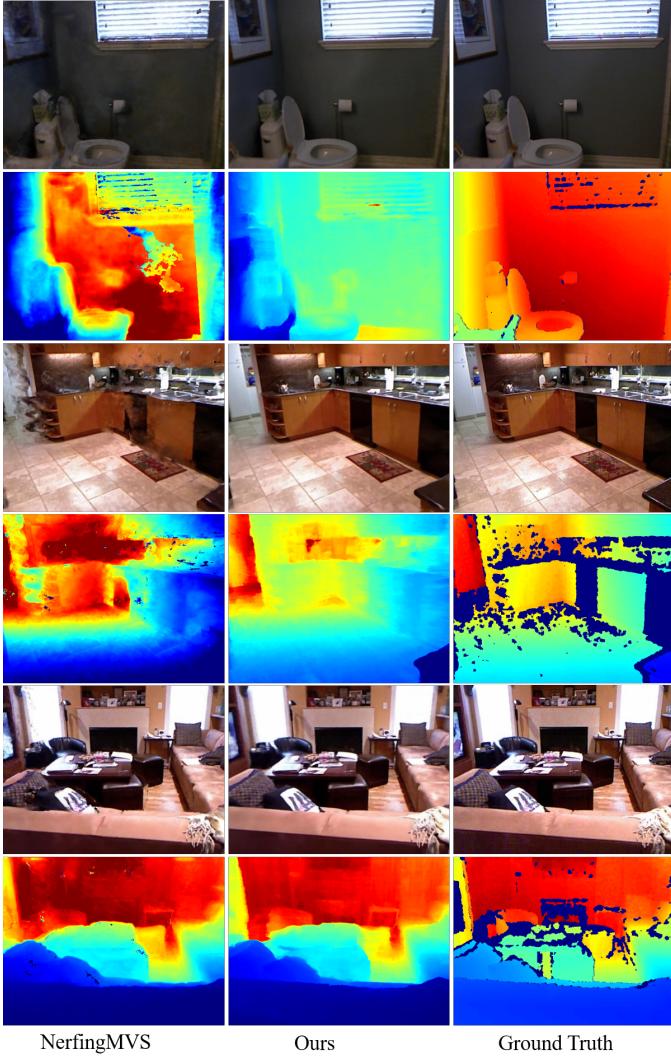


Fig. 18: Comparisons with NerfingMVS.

We can find that StructNeRF is significantly better than NerfingMVS in novel view synthesis and depth estimation. Only the predicted planes of NerfingMVS are slightly more flat than ours, but with the cost of an additional depth estimation network pretrained on a large amount of external indoor data. Such a network also learns strong plane priors like Dense Depth Priors [25].

APPENDIX B DATASETS DETAILS

We take one frame every 20 frames from each video in the following datasets evenly for training and evaluation.

B.1 NYUv2

The following eight scenes are used for evaluation:

- bathroom_0046
- dentist_office_0001
- dining_room_0004
- dining_room_0007
- dining_room_0016
- kitchen_0026a

- living_room_0003
- living_room_0033

B.2 ScanNet

The following eight scenes are used for evaluation:

- scene0419_00
- scene0449_00
- scene0477_00
- scene0577_00
- scene0753_00
- scene0758_00
- scene0776_00
- scene0781_00

B.3 SUN3D

The following eight scenes are used for evaluation:

- home_ac/home_ac_scan2_2012_aug_22
- hotel_stb/scan2
- mit_3_133/classroom_3133_nov_6_2012_scan1_erika
- mit_gym_z_squash/gym_z_squash_scan1_oct_26_2012_erika
- mit_lab_koch/lab_koch_bench_nov_2_2012_scan1_erika
- mit_w20_athena/sc_athena_oct_29_2012_scan1_erika
- mit_w85_4/4_2
- mit_w85_5/5_1

APPENDIX C LATENT CODE

We also experimented on the per-frame latent code to deal with the illumination change across input images. According to NeRF-W [19], latent codes can be used to control the view-specific appearance and is helpful for more consistent color prediction across the scene. Following NeRF-W, we set the dimension of latent code to 48 and concatenate it with view directions for input. Unfortunately, the latent codes in testing frames are never known and we set them to zero by default. The results after using the latent code are listed in Table 4.

In fact, the colors of the rendered images do not have to be exactly the same as that of the test views. In order to compensate for the gap, we also report an additional set of PSNR, SSIM and LPIPS, which is obtained by setting the latent codes of the testing views to the averaged ones of the two nearest training views (two adjacent frames). However, in real-world applications, it is hard to know which two training frames are the most suitable for testing views and the left/right image split evaluation procedure in NeRF-W [19] is also impractical. Therefore, the reported values can only serve as the upper bound of performance (listed in parentheses). In the main text of our paper, we do not use the latent code for fair comparison.

REFERENCES

- [1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: speeded up robust features. In *Computer Vision - ECCV 2006, 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part I*, pages 404–417, 2006.

Method	Need Pretraining?	PSNR↑	SSIM↑	LPIPS↓	Depth RMSE↓	Plane Mean Dev↓
NerfingMVS [33]	Yes	20.64	0.7348	0.2643	0.2190	0.0229
Ours	No	27.95	0.8569	0.1498	0.1960	0.0263

TABLE 3: Comparisons with NerfingMVS on NYUv2.

Dataset	Method	Need Pretraining?	PSNR↑	SSIM↑	LPIPS↓	Depth RMSE↓	Plane Mean Dev↓
NYUv2	Ours	No	28.10	0.8561	0.1663	0.3113	0.0301
	Ours+latent code	No	24.03(28.31)	0.8255(0.8575)	0.1824(0.1621)	0.3136	0.0315
SUN3D	Ours	No	22.88	0.7627	0.2748	0.5292	0.0327
	Ours+latent code	No	18.90(24.65)	0.7281(0.7914)	0.2993(0.2516)	0.4696	0.0318
Scannet	Ours	No	24.67	0.8308	0.2481	0.2298	0.0270
	Ours+latent code	No	23.73(25.92)	0.8276(0.8420)	0.2547(0.2340)	0.2465	0.0264

TABLE 4: The quantitative results after adding the per-camera latent code to our method. Values in the parentheses are obtained by setting the latent code of the testing frames to the averaged latent code values of the two most adjacent frames.

- [2] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 14104–14113, 2021.
- [3] Julian Chibane, Aayush Bansal, Verica Lazova, and Gerard Pons-Moll. Stereo radiance fields (SRF): learning view synthesis for sparse views of novel scenes. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 7911–7920, 2021.
- [4] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2432–2443, 2017.
- [5] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. *CoRR*, abs/2107.02791, 2021.
- [6] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *CoRR*, abs/1607.02565, 2016.
- [7] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *Int. J. Comput. Vis.*, 59(2):167–181, 2004.
- [8] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 3827–3837, 2019.
- [9] Yuan-Chen Guo, Di Kang, Linchao Bao, Yu He, and Song-Hai Zhang. Nerfren: Neural radiance fields with reflections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18409–18418, 2022.
- [10] Peter Hedman, Tobias Ritschel, George Drettakis, and Gabriel Brostow. Scalable inside-out image-based rendering. *ACM Transactions on Graphics (TOG)*, 35(6):1–11, 2016.
- [11] Peter Hedman, Tobias Ritschel, George Drettakis, and Gabriel J. Brostow. Scalable inside-out image-based rendering. *ACM Trans. Graph.*, 35(6):231:1–231:11, 2016.
- [12] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 5865–5874, 2021.
- [13] Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 12929–12938, 2021.
- [14] Hualie Jiang, Laiyan Ding, Junjie Hu, and Rui Huang. Plnet: Plane and line priors for unsupervised indoor depth estimation. In *International Conference on 3D Vision, 3DV 2021, London, United Kingdom, December 1-3, 2021*, pages 741–750, 2021.
- [15] Mijeong Kim, Seonguk Seo, and Bohyung Han. Infonerf: Ray entropy minimization for few-shot neural volume rendering. *CoRR*, abs/2112.15399, 2021.
- [16] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.
- [17] Boying Li, Yuan Huang, Zeyu Liu, Danping Zou, and Wenxian Yu. Structdepth: Leveraging the structural regularities for self-supervised indoor depth estimation. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 12643–12653, 2021.
- [18] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. *arXiv preprint arXiv:2104.06405*, 2021.
- [19] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 7210–7219, 2021.
- [20] Yue Meng, Yongxi Lu, Aman Raj, Samuel Sunarjo, Rui Guo, Tara Javidi, Gaurav Bansal, and Dinesh Bharadia. Signet: Semantic instance aided unsupervised 3d geometry perception. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 9810–9820, 2019.
- [21] 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. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, pages 405–421, 2020.
- [22] Michael Niemeyer, Jonathan T. Barron, Ben Mildenhall, Mehdi S. M. Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. *CoRR*, abs/2112.00724, 2021.
- [23] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021.
- [24] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021.
- [25] Barbara Roessle, Jonathan T. Barron, Ben Mildenhall, Pratul P. Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. *CoRR*, abs/2112.03288, 2021.
- [26] Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 4104–4113, 2016.
- [27] 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 Confer-*

- ence, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, pages 501–518, 2016.
- [28] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V*, pages 746–760, 2012.
- [29] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. *arXiv preprint arXiv:2112.03907*, 2021.
- [30] Chen Wang, Xian Wu, Yuan-Chen Guo, Song-Hai Zhang, Yu-Wing Tai, and Shi-Min Hu. Nerf-sr: High-quality neural radiance fields using super-sampling. *arXiv preprint arXiv:2112.01759*, 2021.
- [31] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, 2004.
- [32] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf-: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021.
- [33] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 5590–5599, 2021.
- [34] Jianxiong Xiao, Andrew Owens, and Antonio Torralba. SUN3D: A database of big spaces reconstructed using sfm and object labels. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 1625–1632, 2013.
- [35] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 4578–4587, 2021.
- [36] Zehao Yu, Lei Jin, and Shenghua Gao. P²net: Patch-match and plane-regularization for unsupervised indoor depth estimation. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXIV*, pages 206–222, 2020.
- [37] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *CoRR*, abs/2010.07492, 2020.
- [38] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.
- [39] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 586–595, 2018.
- [40] Junsheng Zhou, Yuwang Wang, Kaihuai Qin, and Wenjun Zeng. Moving indoor: Unsupervised video depth learning in challenging environments. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 8617–8626, 2019.
- [41] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017.
- [42] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6612–6619, 2017.