ColNeRF: Collaboration for Generalizable Sparse Input Neural Radiance Field

Zhangkai Ni[1], Peiqi Yang[1], Wenhan Yang[2], Hanli Wang[1*], Lin Ma[3], Sam Kwong[4]

[1] Department of Computer Science and Technology, Tongji University, China
[2] Peng Cheng Laboratory, China
[3] Meituan, China
[4] Department of Computer Science, City University of Hong Kong, Hong Kong
{zkni, 2233007, hanliwang}@tongji.edu.cn, yangwh@pcl.ac.cn, forest.linma@gmail.com, cssamk@cityu.edu.hk

Abstract

Neural Radiance Fields (NeRF) have demonstrated impressive potential in synthesizing novel views from dense input, however, their effectiveness is challenged when dealing with sparse input. Existing approaches that incorporate additional depth or semantic supervision can alleviate this issue to an extent. However, the process of supervision collection is not only costly but also potentially inaccurate. In our work, we introduce a novel model: the Collaborative Neural Radiance Fields (ColNeRF) designed to work with sparse input. The collaboration in ColNeRF includes the cooperation among sparse input source images and the cooperation among the output of the NeRF. Through this, we construct a novel collaborative module that aligns information from various views and meanwhile imposes self-supervised constraints to ensure multi-view consistency in both geometry and appearance. A Collaborative Cross-View Volume Integration module (CCVI) is proposed to capture complex occlusions and implicitly infer the spatial location of objects. Moreover, we introduce self-supervision of target rays projected in multiple directions to ensure geometric and color consistency in adjacent regions. Benefiting from the collaboration at the input and output ends, ColNeRF is capable of capturing richer and more generalized scene representation, thereby facilitating higher-quality results of the novel view synthesis. Our extensive experimental results demonstrate that ColNeRF outperforms state-of-the-art sparse input generalizable NeRF methods. Furthermore, our approach exhibits superiority in fine-tuning towards adapting to new scenes, achieving competitive performance compared to per-scene optimized NeRF-based methods while significantly reducing computational costs. Our code is available at: https://github.com/eezkni/ColNeRF.

Introduction

Novel view synthesis aims to generate new view images of a scene based on a set of source images (Zhu, Xie, and Fang 2018). A prominent technique in this field is the Neural Radiance Field (NeRF) (Mildenhall et al. 2021), which learns an implicit neural representation of the scene. NeRF takes a 5D vector as input, comprising a 3D location $\mathrm{x} = (x, y, z)$ and a 2D viewing direction $\mathrm{d} = (\theta, \phi)$ for each point, and estimates the corresponding radiance value (c, σ). The RGB value of a target pixel is then rendered by accumulating radiance from N sampled points along the target ray.

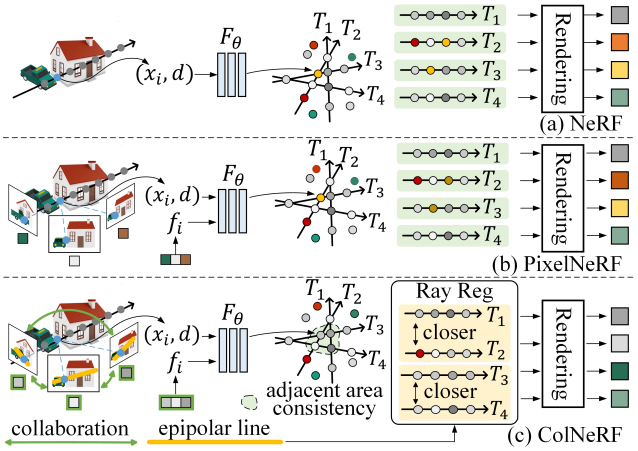


Figure 1: Comparing previous approaches (a), (b) with our method (c): Previous approaches heavily depend on a learned neural radiance field (F_θ) for synthesis. However, these approaches result in undesirable outcomes with limited utilization of source view features and their interrelationships. ColNeRF leverages collaboration at both input and output ends, providing richer supervision for training F_θ.

The excellent performance of NeRF comes with a price, i.e. with a large amount of high-quality input source images used for training. However, acquiring a substantial number of RGB images along with their corresponding accurate camera parameters necessitates a complex process of calibrating. In real-world scenarios, not only is this difficult to execute, but the accuracy of the results obtained is questionable. In scenarios where input images are limited, the novel view results generated by NeRF are degraded due to the lack of dense supervision. Moreover, the optimization of NeRF is typically conducted independently for each scene, resulting in notable time inefficiency. Significant research efforts have been put into addressing these issues. An intuitive strategy to improve geometric accuracy of sparse input NeRF is by supervising the generated density values σ for sampled points. However, obtaining ground truth for all these points is unfeasible. An alternative strategy involves incorporating

*Corresponding author

auxiliary supervisory information during training, such as depth for geometry (Deng et al. 2022; Wang et al. 2023) or semantic cues for appearance (Jain, Tancik, and Abbeel 2021). However, these supervisory signals themselves might be inaccurate, which limits the potential effectiveness of this route. Our work also aims to enhance the generalization ability of NeRF, by training a model that can infer across different scenes with sparse source views. This allows us to flexibly handle situations where training data for certain scenes are limited, while maintaining photo-realistic rendering results. Approaches like MVSNeRF (Chen et al. 2021) and PixelNeRF (Yu et al. 2021) have demonstrated improved generalization capabilities by pre-training their models on a diverse multi-view image dataset with various scenes. PixelNeRF integrates pixel features from source views to enhance network capabilities, but inaccuracies arise due to inconsistent correspondence between 2D pixels in source images and the queried 3D location. Therefore, these methods inevitably lead to imprecise modeling, they fail to maintain consistency in geometry and appearance across various views, as they do not properly consider the correlation and cooperation among different viewpoints.

To address these challenges, we propose a generalizable sparse input neural radiance field (ColNeRF), a novel approach that gets rid of the need for additional supervision, constructing the precise and generalized model with the consideration of the collaboration among input source views. Specifically, this method involves the extraction of feature volumes from source images using a pre-trained encoder, followed by the application of cross-view volume fusion to adaptively integrate these features. The exact spatial locations of relevant patches can be determined by matching and reprojecting them into 3D space using camera parameters. This spatial transformation is achieved implicitly through an attention mechanism, serving as a trainable aggregation function that selectively emphasizes important features within source views (Ni et al. 2020b). This mechanism also corrects features of occluded regions by incorporating information from corresponding parts in alternative viewpoints. Furthermore, the collaboration also pays attention to the output end, where the constraint is enforced in both geometry and appearance reconstruction. For geometric regularization, we adopt a self-supervised approach (Ni et al. 2020a) that aims to minimize discrepancies between predicted depths of adjacent target rays. For appearance regularization, we leverage the insight that the most relevant regions within the source views for each target ray should ideally align with their corresponding epipolar lines. It is notable that we train a single model with potent generalization capabilities applicable to all scenes. In summary, our main contributions can be summarized as follows:

- We propose ColNeRF to integrate multi-view compensation and consistency into NeRF at input/output ends, making ColNeRF outperform other generalizable NeRF methods with sparse input, and comparable to scene-specific NeRF approaches with reduced complexity.
- We introduce self-supervised ray regularization to effectively enforce multi-view consistency for effective model guidance, which leads to more accurate geometry and appearance reconstruction.
- ColNeRF achieves superior performance over state-of-the-art generalizable NeRF methods in sparse scenarios and offers efficient adaptability to new scenes via fine-tuning, showcasing comparable results to scene-specific NeRF approaches with reduced computational burden.

Related Works

Preliminary of NeRF. NeRF generated novel view images through an implicit 5D neural radiation field construction process denoted as $\mathrm{F}(\gamma(\mathrm{x}), \gamma(\mathrm{d})) = (c, \sigma)$, where $\gamma(\cdot)$ signifies the position encoding procedure (Mildenhall et al. 2021), $\mathrm{x} = (x, y, z)$ denotes a 3D locationand and $\mathrm{d} = (\theta, \phi)$ denotes a 2D viewing direction. The output c represents RGB values and σ denotes volume density, which can be understood as the probability of a ray terminating at a given particle. The volumetric radiation field produces 2D images via pixel-wise rendering:

$$\mathrm{C}(\mathrm{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathrm{r}(t))\mathrm{c}(\mathrm{r}(t), \mathrm{d})dt, \tag{1}$$

where $T(t) = \exp(-\int_{t_n}^{t} \sigma(\mathrm{r}(s))ds)$, representing accumulated transmittance indicating ray traversal probability from t_n to t without encountering particles. Here, $\mathrm{c}(\mathrm{r}(t), \mathrm{d})$ and $\sigma(\mathrm{r}(t))$ denote color and volume density at the sampled point along ray r at distance t. The radiance field optimization involves minimizing mean squared error between rendered and ground truth colors:

$$\mathcal{L}_{rec} = \sum_{\mathrm{r}\in\mathrm{R}(\mathrm{P})} \left\| \hat{\mathrm{C}}(\mathrm{r}) - \mathrm{C}(\mathrm{r}) \right\|_2^2, \tag{2}$$

where $\mathrm{R}(\mathrm{P})$ is the set of all camera rays of target pose P.

Sparse Input NeRF. Researchers that pursue accurate reconstructed results with a reduced number of input views (*i.e.* sparse input) have garnered significant attention (Chen et al. 2023a; Xu, Zhong, and Neumann 2022). The challenge of sparse input 3D reconstruction arises from the complex task of maintaining consistency in both geometric shape and appearance. Historically, approaches predominantly relied on additional depth or semantics-based supervisory cues to infer occluded regions. A pioneering effort by PixelNeRF first integrated pixel features of source images into vanilla NeRF that only use position information. Additionally, IBRNet (Wang et al. 2021), SRF (Chibane et al. 2021) and MatchNeRF (Chen et al. 2023b) contributed to scene reconstruction through the feature alignment of projected points from diverse perspectives. Besides, researchers have also explored novel forms of explicit 3D representation established from sparse images (Fang et al. 2023), such as voxel mesh (Maturana and Scherer 2015; Sun, Sun, and Chen 2022; Huang et al. 2019; Deng et al. 2021), multiplane images (MPI) (Li et al. 2021; Fontaine et al. 2022), or layered depth images (LDI) (Tulsiani, Tucker, and Snavely 2018; Shih et al. 2020). To address the challenge of inaccurate geometric information under sparse input settings, regularization methods targeting volume density have

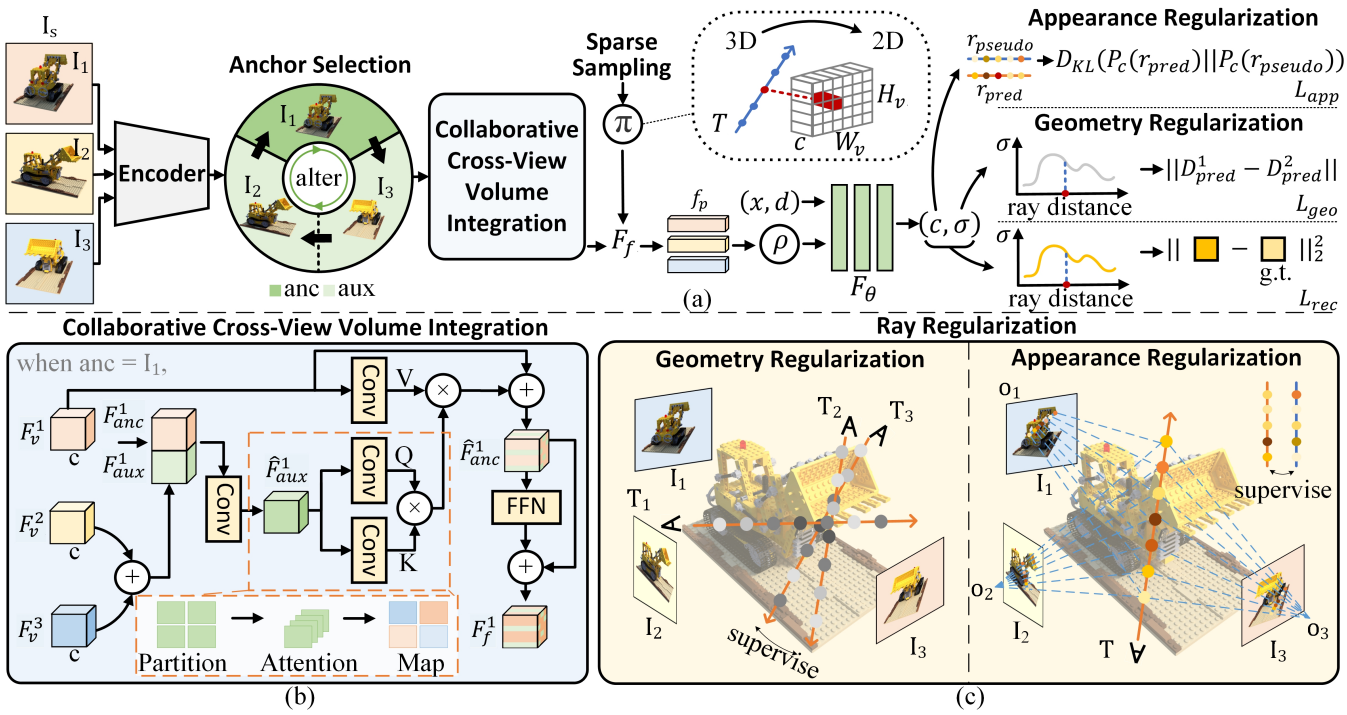


Figure 2: The architecture of proposed ColNeRF. (a): The overview pipeline. (b) and (c): Collaborative Cross-View Volume Integration (CCVI) and Ray Regularization, *i.e.* collaborative input fusion and output constraint. ColNeRF consists of four key steps: 1) Feature volumes are extracted and processed with Anchor Selection (anc: anchor; aux: auxiliary) and Collaborative Cross-View Volume Integration (CCVI) to yield affined feature volumes F_f. 2) Points are sampled and projected (π) onto F_f to derive local features f_p. f_p are subsequently averaged (ρ) and fed into MLP F_θ, alongside (x, d), for predicting radiance values (c, σ). 3) Ray Regularization is employed for predicted output alone for each target ray, encompassing both geometric and appearance aspects. 4) Volume rendering produces final RGB values for pixels in novel views.

been introduced (Somraj and Soundararajan 2023; Somraj, Karanayil, and Soundararajan 2023). For instance, Lombardi et al. (Lombardi et al. 2019) enforced zero volume density for the near camera plane using masks, while InfoNeRF (Kim, Seo, and Han 2022) narrows the distribution of σ within the front and back halves of the same ray, which is more suitable for cases where objects are located in the middle of a scene. RegNeRF (Niemeyer et al. 2022) applies depth constraints on sampled image patches, which is evidently unsuitable for a cross-scene training strategy. An innovative approach taken by FreeNeRF (Yang, Pavone, and Wang 2023) involved the regularization of position encoding frequency for 5D inputs, yielding noteworthy outcomes.

Collaborative Neural Radiance Fields

Motivation

Given a limited set of source images along with the corresponding camera extrinsics $\{(\mathrm{I}_i \in \mathbb{R}^{H \times W \times 3}, \mathrm{P}_i \in \mathbb{R}^{3 \times 4})\}$, we aim to address the following two issues:

- **Limited Effectiveness.** When the input is sparse, incorporating auxiliary supervision such as depth or semantic cues can improve NeRF's performance to an extent. However, these guidance might be not reliable and difficult to obtain, which reduces the effectiveness.
- **Limited Generalization.** As most methods take the one-scene-one-model paradigm. Although pre-training on diverse scenes can improves models' generalization ability, previous works have not fully consider the collaborative relationship of different views systematically, which hinders performance improvement.

Our core goal is to develop a collaborative NeRF model with the capacity for **cross-scene generalization** and rendering **high-quality** results with **multi-view consistency** when taking **sparse input**, without utilizing any auxiliary supervision. In the following, we introduce the overall framework of our approach and then detailed our two contributions.

Overview

Our system pipeline is depicted in Fig. 2. At first, we use a pre-trained encoder **ResNet34** (He et al. 2016) to extract feature volumes $\mathrm{F}_v \in \mathbb{R}^{H_v \times W_v \times d}$ from the source views. H_v, W_v, and d respectively represent the height, width, and the channel dimension. Subsequently, we employ a collaborative cross-view attention mechanism to integrate these feature volumes and obtain fused results F_f of N source views.

The next step involves sampling N_r target rays for training. To train a generalized model applicable across a variety

Method	Setting	PSNR ↑			SSIM ↑			LPIPS ↓			Average ↓		
		3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view
DietNeRF (ICCV 2021)	Trained on DTU and Optimized per Scene	10.01	18.70	22.16	0.354	0.668	0.740	0.574	0.336	0.277	0.383	0.149	0.098
DS-NeRF (CVPR 2022)		16.50	20.50	-	0.540	0.730	-	0.480	0.310	-	0.194	0.113	-
InfoNeRF (CVPR 2022)		11.23	-	-	0.445	-	-	0.543	-	-	0.312	-	-
RegNeRF (CVPR 2022)		15.33	19.10	22.30	0.621	0.757	0.823	0.341	**0.233**	**0.184**	0.189	0.118	0.079
FreeNeRF (CVPR 2023)		18.02	22.39	**24.20**	0.680	0.779	**0.833**	**0.318**	0.240	0.187	0.146	0.094	**0.068**
SRF (CVPR 2021)	Trained on DTU and Not Optimized per Scene	15.84	17.77	18.56	0.532	0.616	0.652	0.482	0.401	0.359	0.207	0.162	0.145
MVSNeRF (ICCV 2021)		16.33	18.26	20.32	0.602	0.695	0.735	0.385	0.321	0.280	0.184	0.146	0.114
PixelNeRF (CVPR 2021)		18.74	21.02	22.23	0.618	0.684	0.714	0.401	0.340	0.323	0.142	0.119	0.105
ColNeRF (Ours)		**19.55**	**22.94**	23.93	**0.716**	**0.797**	0.824	0.362	0.317	0.298	**0.129**	**0.090**	0.079

Table 1: Quantitative comparison on DTU. Our model demonstrates superior performance in sparse input synthesizing compared to most existing methods. Our direct baseline is PixelNeRF. For ease of identification, the entries with the best and second-best performances are respectively highlighted in bold and underscored with an underline.

Method	Setting	PSNR ↑			SSIM ↑			LPIPS ↓			Average ↓		
		3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view
DietNeRF (ICCV 2021)	Trained on LLFF and Optimized per Scene	14.94	21.75	24.28	0.370	0.717	0.801	0.496	0.248	0.183	0.240	0.105	0.073
RegNeRF (CVPR 2022)		19.08	23.10	24.86	0.587	0.760	0.820	0.336	0.206	0.161	0.149	0.086	0.067
FreeNeRF (CVPR 2023)		19.63	**23.73**	**25.13**	**0.612**	**0.779**	**0.827**	**0.308**	**0.195**	**0.160**	0.134	**0.075**	**0.064**
SRF ft (CVPR 2021)	Trained on DTU and Not Optimized per Scene	17.07	16.75	17.39	0.436	0.438	0.465	0.529	0.521	0.503	0.203	0.207	0.193
MVSNeRF ft (ICCV 2021)		17.88	19.99	20.47	0.584	0.660	0.695	0.327	0.264	0.244	0.157	0.122	0.111
PixelNeRF ft (CVPR 2021)		16.17	17.03	18.92	0.438	0.473	0.535	0.512	0.477	0.430	0.217	0.196	0.163
ColNeRF ft (Ours)		**20.97**	23.32	23.52	0.587	0.747	0.762	0.447	0.295	0.280	**0.132**	0.088	0.084

Table 2: Quantitative comparison on LLFF. We generalize the pre-trained model to LLFF dataset and conduct 15K, 10K, and 5K fine-tuning iterations for each scene with 3, 6, and 9 views (all fewer than Pixel-NeRF's default 20K fine-tune steps). Although methods like FreeNeRF may produce better results, they train separate models on each scene for 250K iterations. In contrast, our method trains a single model for all scenes and achieves comparable results with much less fine-tuning cost.

of scenes, we adopt a strategy that randomly selects scene and emits rays into it with a scattered pattern. Subsequently, we sample N_p points on each target ray. For these sampled points, their camera parameters enable us to project them onto each source image. We then extract their corresponding pixel features f_p from the affined feature volume F_f of each view using bilinear interpolation, the local feature of 3D point x in the i-th source view is obtained as follows:

$$\mathrm{f}_p^i = \mathrm{Interpolate}(\mathrm{F}_f^i(\Pi(\mathrm{x})) \in \mathbb{R}^d. \tag{3}$$

The local pixel features f_p are then input into the Neural Radiance Field F_θ along with the coordinates x and view direction d to yield the color c and density σ:

$$\mathrm{F}(\gamma(\mathrm{x}), \gamma(\mathrm{d}), \rho(\{\mathrm{f}_p^i\}_{i=1}^N)) = (c, \sigma), \tag{4}$$

where ρ denotes the averaging operation, and N denotes the number of source views.

Finally, as illustrated in Eqn. (1), we employ principles from classical volume rendering to aggregate the final RGB values $\mathrm{C}(\mathrm{r})$. The training loss function of our model includes three parts. One is the reconstruction loss, which is identical to that in Eqn. (2). The remaining two components originate from the ray regularization module:

$$\mathcal{L}_{total} = \mathcal{L}_{rec} + \lambda_1 \mathcal{L}_{geo} + \lambda_2 \mathcal{L}_{app}. \tag{5}$$

The loss weights λ_1 and λ_2 are set to $1e$-4 and $2e$-4 respectively throughout our experiments.

Collaborative Cross-View Volume Integration

Before being fed into the MLP, we enrich sparse information by fusing multi-view source images, which helps identify corresponding regions cross different views. These correlations are used to correct potential biases in source images.

This process enables us to generate N affined feature volumes that capture information from other source views. Each of these volumes can be denoted as $\mathrm{F}_f^i \in \mathbb{R}^{H_v \times W_v \times d}$:

$$\mathrm{F}_f^i = \mathrm{CCVI}(\mathrm{F}_{anc}^i, \mathrm{F}_{aux}^i), \tag{6}$$

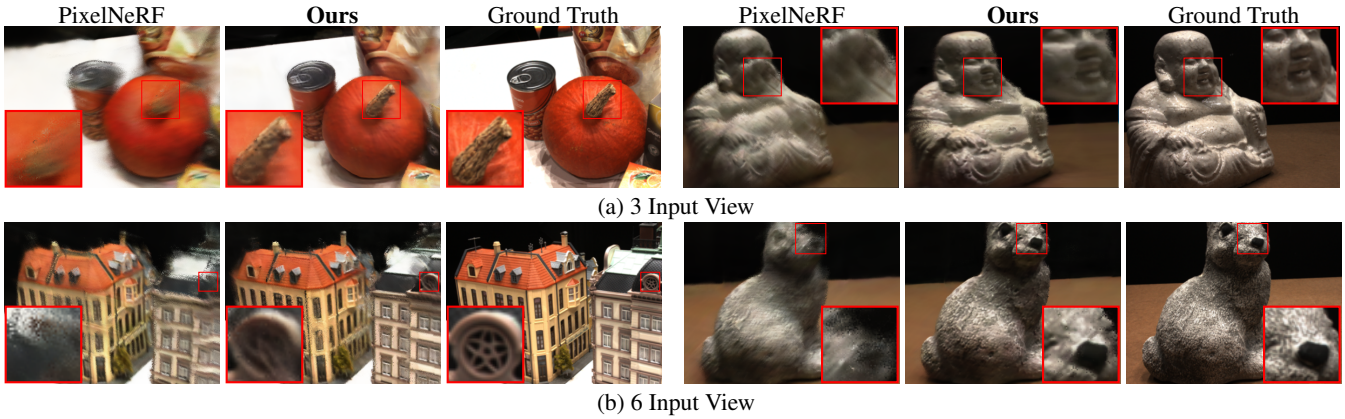
F_{anc}^i represents current anchor feature volume, each source view takes turns as the anchor: $\mathrm{F}_{anc}^i = \mathrm{F}_v^i$, while F_{aux}^i represents the summation of other auxiliary feature volumes:

$$\mathrm{F}_{aux}^i = \sum_{j=1}^{i-1} \mathrm{F}_v^j + \sum_{j=i+1}^{N} \mathrm{F}_v^j. \tag{7}$$

The transformer block in CCVI is computed as:

$$\begin{aligned} \hat{\mathrm{F}}_{anc}^i &= \mathrm{AVGI}(\mathrm{F}_{anc}^i, \hat{\mathrm{F}}_{aux}^i) + \mathrm{F}_{anc}^i, \\ \mathrm{F}_f^i &= \mathrm{FFN}(\hat{\mathrm{F}}_{anc}^i) + \hat{\mathrm{F}}_{anc}^i, \\ \hat{\mathrm{F}}_{aux}^i &= \mathrm{Conv}([\mathrm{F}_{anc}^i, \mathrm{F}_{aux}^i]), \end{aligned} \tag{8}$$

$\mathrm{AVGI}(\cdot)$ denotes Auxiliary Volume Guided Integration, $\mathrm{FFN}(\cdot)$ denotes a Feed-Forward Network, $\mathrm{Conv}(\cdot)$ denotes



(a) 3 Input View

(b) 6 Input View

Figure 3: Qualitative comparison on DTU between PixelNeRF and ColNeRF, we present results under 3 and 6 input views setting. PixelNeRF's direct use of the average pixel feature from each source view often results in blurriness or shape distortion.

a convolutional layer for dimension reduction. $\hat{\mathrm{F}}^i_{aux}$ are used as auxiliary volume for current anchor feature volume F^i_{anc}:

$$\mathrm{AVGI}(\mathrm{F}_{anc}, \hat{\mathrm{F}}_{aux}) = \mathrm{Softmax}\left(\frac{\mathrm{QK}^T}{\sqrt{d_k}}\right)\mathrm{V}, \tag{9}$$

where

$$\begin{aligned}\mathrm{Q} &= \mathrm{W}_Q(\hat{\mathrm{F}}_{aux}),\\ \mathrm{K} &= \mathrm{W}_K(\hat{\mathrm{F}}_{aux}),\\ \mathrm{V} &= \mathrm{W}_V(\mathrm{F}_{anc}),\end{aligned}$$

W_Q,W_K and W_V are learnable transformations, d_k is the feature channel dimension of Q and K. Each view is processed individually to obtain fused feature volumes$\{\mathrm{F}^i_f\}^N_{i=1}$.

We believe this process also implicitly ensures geometric consistency across multiple views. We chose attention mechanism as the method to integrate features, unlike previous approaches that focus on pixel-level features from different perspectives (T et al. 2023), our strategy integrates at the patch level.

Ray Regularization

Geometry Regularization. Our goal is to constrain the prediction of each points' density and improve the model's resilience to variations in view direction. Unlike InfoNeRF and RegNeRF, we opted for a more versatile ray regularization approach that better suits cross-scene training strategies. We employ a collaborative mutual-supervision for neighboring rays, pairing N_{pairs} of the closest rays together and minimize the L1 Loss of the predicted depth for each pairs:

$$\begin{aligned}\mathcal{L}_{geo} &= \sum_{i=1}^{N_{pairs}} \mathrm{M}(\mathrm{r}_i) \odot (D(\mathrm{r}_i) - D(\hat{\mathrm{r}}_i)),\\ D(\mathrm{r}_i) &= \int_{t_n}^{t_f} T(t)\sigma(\mathrm{r}_i(t))tdt,\\ \mathrm{M}(\mathrm{r}_i) &= \begin{cases} 0 & \text{if } \mathrm{Q}(\mathrm{r}_i) < \tau \text{ or } \mathrm{Q}(\hat{\mathrm{r}}_i) < \tau \\ 1 & \text{otherwise} \end{cases},\end{aligned} \tag{10}$$

r_i and $\hat{\mathrm{r}}_i$ refer to the two neighboring target rays that are paired together. $D(\cdot)$ denotes the predicted depth of sampled rays. We employ a mask to exclude certain adjacent ray pairs that not need to be regularized. For instance, a pair that one ray hits the edge of an object while its corresponding ray does not hit anything. Applying geometric constraints to such pairs may introduce foggy artifacts. $\mathrm{Q}(\cdot) = \sum_{i=1}^{N} 1 - \exp(-\sigma_i\delta_i)$, represents the cumulative ray density, here i refers to the i-th sampled point on a ray. We set $\tau = 0.1$ in our experiments.

Appearance Regularization. Epipolar plane is a plane determined in space by a spatial point and the optical centers of two distinct cameras, while the epipolar line arises from the intersection of the epipolar plane and the imaging plane. The RGB labels for each target ray are obtained from the color information along its corresponding epipolar lines on source views. Similar to the extraction of local features, epipolar lines are grabbed using projection and interpolation, their RGB are set as color label for target rays. However, due to occlusion, directly using L_1 loss for constraints can be overly restrictive. To address this, we minimize the KL-divergence between the color distributions P_c of target rays and their corresponding color labels:

$$\begin{aligned}\mathcal{L}_{app} &= \sum_{i=1}^{N_r} D_{KL}\left(P_c(\mathrm{r}_i)||P_c(\hat{\mathrm{r}}_i)\right)\\ &= \sum_{i=1}^{N_r}\sum_{j=1}^{N_p} \mathrm{p_c}(\mathrm{M}'(\mathrm{c}(\mathrm{r}_{i,j})))\log\frac{\mathrm{p_c}(\mathrm{M}'(\mathrm{c}(\mathrm{r_{i,j}})))}{\mathrm{p_c}(\mathrm{M}'(\mathrm{c}(\hat{\mathrm{r}}_{\mathrm{i,j}})))}.\end{aligned} \tag{11}$$

Here, $\hat{\mathrm{r}}_\mathrm{i}$ represents the target ray emitted during training, while r_i signifies the "pseudo-label" ray derived from averaging the color of target ray's corresponding epipolar lines across multiple source views. $\mathrm{p_c}(\hat{\mathrm{r}}_{i,j})$ denotes the probability of j-th point on i-th target ray. A mask is employed to exclude inaccurately projected points from the computation:

$$\mathrm{M}'(\cdot) = \begin{cases} \text{false} & \text{if } \varepsilon(\mathrm{r}_{i,j}) > \varepsilon_{\max} \\ \text{true} & \text{otherwise} \end{cases}, \tag{12}$$

$\varepsilon(\mathrm{r}_{i,j})$ represents the pixel coordinate in the source image obtained by homography transformation from the point $\mathrm{r}_{i,j}$ on the target ray. The term ε_{max} refers to the maximum pixel coordinate in the source image.

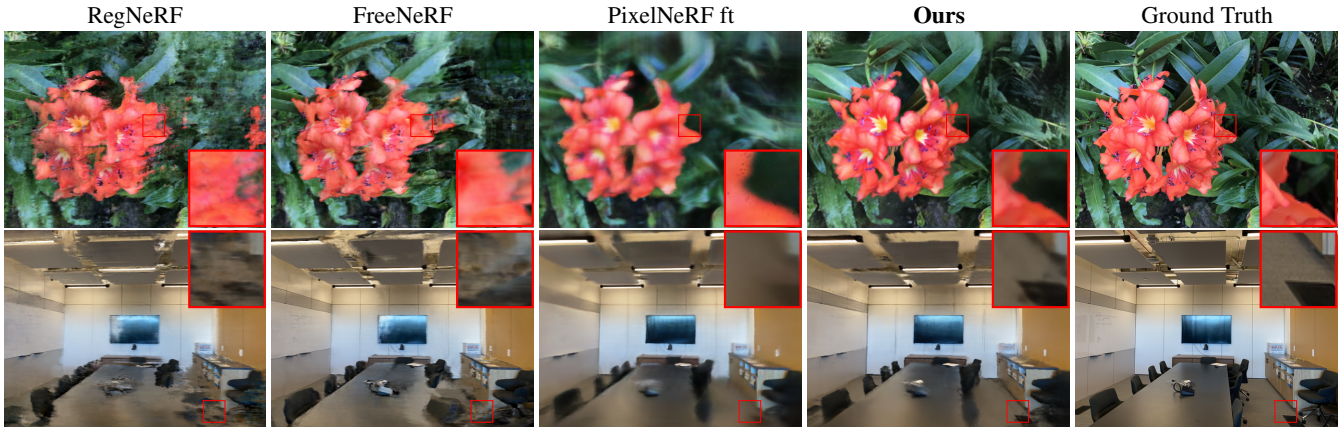


Figure 4: Qualitative results on LLFF under 3 input views setting. FreeNeRF is among the best-performing methods for per-scene optimization, but it exhibits noticeable noise issues due to the inaccurate encoding of high-frequency information. PixelNeRF's results suffer from apparent blurriness when compared to our method.

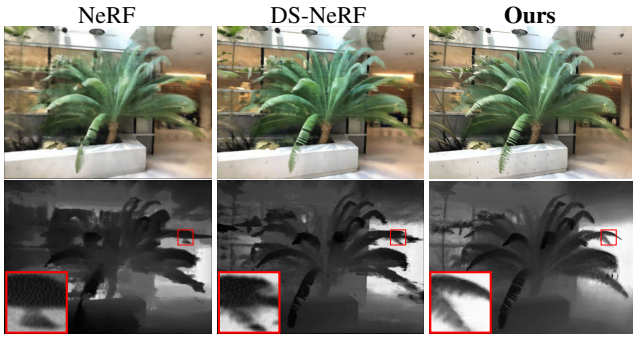


Figure 5: Qualitative results on fern under 5 input views between NeRF, DS-NeRF and ColNeRF (ours). Rendered depth maps reveal that our model achieves more accurate shapes than DS-NeRF, which relies on explicit depth labels.

Experiments

Experimental Setups

Datasets. We evaluate our method on two datasets: DTU (Jensen et al. 2014) and LLFF (Mildenhall et al. 2019). We train on DTU and test the generalization capabilities of the pre-trained model on LLFF. For DTU, we follow the evaluation protocol established by PixelNeRF. For LLFF, we follow the evaluation standards set by NeRF and use it as an out-of-distribution test for conditional models. To evaluate our method's performance with sparse input, we conduct experiments with 3-view, 6-view, and 9-view configurations.

Metrics. We report the mean of PSNR, SSIM (Wang et al. 2004), and the LPIPS perceptual metric (Zhang et al. 2018). To ease comparison, we also report the geometric mean of $\mathrm{MSE} = 10^{-\mathrm{PSNR}/10}$, $\sqrt{1-\mathrm{SSIM}}$, and LPIPS.

Training Details. In line with PixelNeRF, we sample 128 training rays per iteration. To boost controllability, we randomly emit 112 rays and designate the final 16 of them as reference rays. The remaining 16 rays of all 128 rays are sampled from regions adjacent to reference rays. These freshly sampled rays share the same camera parameters and origin with the reference rays, but exhibit an offset of up to 7 pixels on the pixel plane. These last 32 rays are used as paired adjacent rays for geometry ray regularization. For the training of 3-view and 6-view, we set the batch size (BS) to 3, and for the 9-view training, BS is set to 2. We maintain a fixed learning rate of 1e-4 throughout our training process.

Comparing Baselines. To facilitate comparison, we select several state-of-the-art (SOTA) methods that effectively address the challenge of limited input. These include PixelNeRF, SRF, MVSNeRF, DietNeRF, DS-NeRF, InfoNeRF, RegNeRF, and FreeNeRF. The first three, akin to our approach, are pre-trained across various scenes, while the remaining five are optimized for specific scenarios. Given the similarities in the compared methods, datasets, and experiment settings, we directly use the reported results in FreeNeRF and RegNeRF as the basis for our comparison with other methods. The results of DS-NeRF and InfoNeRF were taken from their published papers.

Quantitative Comparisons with SOTA Methods

Comparisons on DTU. Tab. 1 presents the quantitative results on the DTU dataset. Our model outperforms in most experimental settings, with the exception of the 9-view configuration, where it slightly lags behind FreeNeRF.

Comparisons on LLFF. To validate the model's generalization performance, we test our pre-trained model on the LLFF dataset. Following RegNeRF's comparison setup, we conduct extra fine-tuning iterations per scene for each method. The quantitative results of our experiments are presented in Tab. 2. While methods like FreeNeRF, which train separate models for each scene, may yield superior results, it's important to consider the overall performance. The LLFF dataset consists of 8 scenes, and FreeNeRF requires retraining for each scene over 250K iterations. In contrast, our model achieves comparable results with a total of no more than 15K fine-tuning iterations. This underscores our

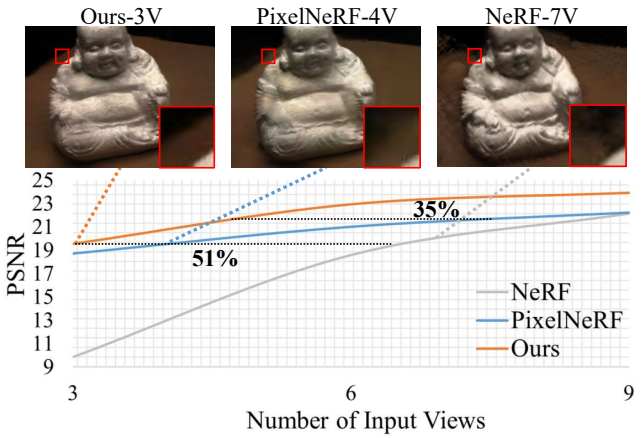


Figure 6: Data Efficiency. In sparse settings, our method requires an average of 51% fewer images than NeRF and an average of 35% fewer images than PixelNeRF to achieve a similar test set performance on DTU.

model's ability to produce realistic results across different scenes with significantly less computational effort.

Qualitative Comparisons with SOTA Methods

Comparisons on DTU. Fig. 3 provides a qualitative comparison between our direct baseline PixelNeRF and our ColNeRF. PixelNeRF exhibits blurriness and shape distortion as it directly feeds the mean of pixel features from all source views into the network, which can introduce negative biases, especially when the projection point falls into occluded regions. In contrast, our model rectifies these errors and implicitly reconstructs the object's geometric shape, leading to improved performance.

Comparisons on LLFF. Fig. 4 presents a comparison of LLFF of our model with FreeNeRF, RegNeRF, and our baseline PixelNeRF. The results from the FreeNeRF and RegNeRF methods are noticeably marred by significant noise, while the PixelNeRF method exhibits a substantial blurring issue. In contrast, our method guarantees accurate and smooth geometric depiction while delivering high-quality rendered images. Further strengthening our claims, Fig. 5 illustrates our model's precise geometric control in comparison to DS-NeRF, another method that incorporates explicit depth supervision. DS-NeRF employs depth labels generated with COLMAP[1] as constraints for rendering. It can be seen that the inaccuracy of these labels distorts DS-NeRF's geometry understanding. Conversely, our model achieves superior geometric reconstruction and multi-view consistency without additional supervision.

Data Efficiency

Our model is designed for novel view synthesis under sparse input setting, to assess the data efficiency of our method, we perform a comparative analysis with NeRF and PixelNeRF using different numbers of input views, depicted in Fig. 6. For sparse inputs, our method necessitates up to 51% fewer input views to attain an equivalent mean PSNR on the test set as that of NeRF, with the disparity being more noticeable for fewer input views. Furthermore, our method delivers performance on par with PixelNeRF, averaging a 35% reduction in the required input views to yield comparable results.

[1]A universal motion structure (SfM) and multi-view stereo (MVS) pipeline, offering convenient tools for 3D reconstruction.

	VI	$\mathcal{L}_{geo}$	$\mathcal{L}_{app}$	Info	Reg	PSNR↑	SSIM ↑	LPIPS ↓
0						18.74	0.618	0.401
1	✓					19.21	0.698	0.384
2	✓	✓				19.39	<u>0.714</u>	0.375
3	✓		✓			<u>19.48</u>	0.710	<u>0.373</u>
4	✓			✓		19.32	0.707	0.380
5	✓				✓	18.11	0.634	0.475
Full	✓	✓	✓			**19.55**	**0.716**	**0.362**

Table 3: Ablative results of our model designs on 3-view input DTU. VI denotes the cross-view volume integration module. Info and Reg respectively denote the Ray Regularization employed in InfoNeRF and RegNeRF.

Ablation Study

We evaluate the impact of our design choices on the 3-view input DTU dataset in Tab. 3. Adding our collaborative cross-view volume integration (CCVI) results in drastically better performance on all metrics. The Ray Regularization was designed to remove potential artifacts in the rendered results. We observe a slight improvement in the results after adding regularization. "To further demonstrate the effectiveness, we also compared the impact of different ray regularization methods from InfoNeRF, RegNeRF and ColNeRF on the same backbone (ColNeRF w/o RayReg). It can be observed that ColNeRF still outperforms the others.

Limitations and Conclusion

ColNeRF is designed to have a lightweight network structure. Based on this consideration, ColNeRF shoots fewer sampling points per ray and adopts a small dimension of the feature volume as well as average pooling. These factors also lead to some downsides, *i.e.* degrading the reconstruction accuracy with locally smooth renderings. To mitigate these issues, future work can pay attention to various strategies, such as incorporating multi-scale feature volume representations, increasing the utilization of sampling points, or applying frequency regularization constraints. To conclude, we have introduced ColNeRF, a method capable of achieving photorealistic renderings without using any external data. We have effectively integrated collaborative compensation and constraint into NeRF which leads to accurate 3D modeling with color and geometric consistency. This new route provides strong supervision for model training even in the absence of ground truth. Future research may explore faster and more detailed NeRF models.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62201387 and 62371343, in part by the Shanghai Pujiang Program under Grant 22PJ1413300, and in part by the Fundamental Research Funds for the Central Universities.

References

Chen, A.; Xu, Z.; Zhao, F.; Zhang, X.; Xiang, F.; Yu, J.; and Su, H. 2021. MVSNeRF: Fast Generalizable Radiance Field Reconstruction from Multi-View Stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 14124–14133.

Chen, S.; Yan, B.; Sang, X.; Chen, D.; Wang, P.; Guo, X.; Zhong, C.; and Wan, H. 2023a. Bidirectional Optical Flow NeRF: High Accuracy and High Quality under Fewer Views. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 359–368.

Chen, Y.; Xu, H.; Wu, Q.; Zheng, C.; Cham, T.-J.; and Cai, J. 2023b. Explicit Correspondence Matching for Generalizable Neural Radiance Fields. *arXiv preprint arXiv:2304.12294*.

Chibane, J.; Bansal, A.; Lazova, V.; and Pons-Moll, G. 2021. Stereo Radiance Fields (SRF): Learning View Synthesis for Sparse Views of Novel Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7911–7920.

Deng, J.; Shi, S.; Li, P.; Zhou, W.; Zhang, Y.; and Li, H. 2021. Voxel R-CNN: Towards High Performance Voxel-Based 3D Object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 1201–1209.

Deng, K.; Liu, A.; Zhu, J.-Y.; and Ramanan, D. 2022. Depth-Supervised NeRF: Fewer Views and Faster Training for Free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12882–12891.

Fang, S.; Xu, W.; Wang, H.; Yang, Y.; Wang, Y.; and Zhou, S. 2023. One is All: Bridging the Gap between Neural Radiance Fields Architectures with Progressive Volume Distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 597–605.

Fontaine, N. K.; Carpenter, J.; Gross, S.; Leon-Saval, S.; Jung, Y.; Richardson, D. J.; and Amezcua-Correa, R. 2022. Photonic Lanterns, 3-D Waveguides, Multiplane Light Conversion, and Other Components that Enable Space-Division Multiplexing. *Proceedings of the IEEE*, 110: 1821–1834.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Huang, W.; Lai, B.; Xu, W.; and Tu, Z. 2019. 3D Volumetric Modeling with Introspective Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 8481–8488.

Jain, A.; Tancik, M.; and Abbeel, P. 2021. Putting NeRF on a Diet: Semantically Consistent Few-Shot View Synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5885–5894.

Jensen, R.; Dahl, A.; Vogiatzis, G.; Tola, E.; and Aanæs, H. 2014. Large Scale Multi-View Stereopsis Evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 406–413.

Kim, M.; Seo, S.; and Han, B. 2022. InfoNeRF: Ray Entropy Minimization for Few-Shot Neural Volume Rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12912–12921.

Li, J.; Feng, Z.; She, Q.; Ding, H.; Wang, C.; and Lee, G. H. 2021. MINE: Towards Continuous Depth MPI with NeRF for Novel View Synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12578–12588.

Lombardi, S.; Simon, T.; Saragih, J.; Schwartz, G.; Lehrmann, A.; and Sheikh, Y. 2019. Neural Volumes: Learning Dynamic Renderable Volumes from Images. *ACM Transactions on Graphics*, 38(4).

Maturana, D.; and Scherer, S. 2015. VoxNet: A 3D Convolutional Neural Network for Real-Time Object recognition. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 922–928.

Mildenhall, B.; Srinivasan, P. P.; Ortiz-Cayon, R.; Kalantari, N. K.; Ramamoorthi, R.; Ng, R.; and Kar, A. 2019. Local Light Field Fusion: Practical View Synthesis With Prescriptive Sampling Guidelines. *ACM Transactions on Graphics*, 38(4): 1–14.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *Communications of the ACM*, 65(1): 99–106.

Ni, Z.; Yang, W.; Wang, S.; Ma, L.; and Kwong, S. 2020a. Towards Unsupervised Deep Image Enhancement With Generative Adversarial Network. *IEEE Transactions on Image Processing*, 29: 9140–9151.

Ni, Z.; Yang, W.; Wang, S.; Ma, L.; and Kwong, S. 2020b. Unpaired Image Enhancement with Quality-Attention Generative Adversarial Network. In *Proceedings of the 28th ACM International Conference on Multimedia*, 1697–1705.

Niemeyer, M.; Barron, J. T.; Mildenhall, B.; Sajjadi, M. S.; Geiger, A.; and Radwan, N. 2022. RegNeRF: Regularizing Neural Radiance Fields for View Synthesis from Sparse Inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5480–5490.

Shih, M.-L.; Su, S.-Y.; Kopf, J.; and Huang, J.-B. 2020. 3D Photography Using Context-Aware Layered Depth Inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8028–8038.

Somraj, N.; Karanayil, A.; and Soundararajan, R. 2023. SimpleNeRF: Regularizing Sparse Input Neural Radiance Fields with Simpler Solutions. *arXiv preprint arXiv:2309.03955*.

Somraj, N.; and Soundararajan, R. 2023. ViP-NeRF: Visibility Prior for Sparse Input Neural Radiance Fields. In *Proceedings of the ACM Special Interest Group on Computer Graphics and Interactive Techniques*.

Sun, C.; Sun, M.; and Chen, H.-T. 2022. Direct Voxel Grid Optimization: Super-Fast Convergence for Radiance Fields Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5459–5469.

T, M. V.; Wang, P.; Chen, X.; Chen, T.; Venugopalan, S.; and Wang, Z. 2023. Is Attention All That NeRF Needs? In *The Eleventh International Conference on Learning Representations*.

Tulsiani, S.; Tucker, R.; and Snavely, N. 2018. Layer-Structured 3D Scene Inference via View Synthesis. In *Proceedings of the European Conference on Computer Vision*, 302–317.

Wang, G.; Chen, Z.; Loy, C. C.; and Liu, Z. 2023. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. *arXiv preprint arXiv:2303.16196*.

Wang, Q.; Wang, Z.; Genova, K.; Srinivasan, P. P.; Zhou, H.; Barron, J. T.; Martin-Brualla, R.; Snavely, N.; and Funkhouser, T. 2021. IBRNet: Learning Multi-View Image-Based Rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4690–4699.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612.

Xu, Q.; Zhong, Y.; and Neumann, U. 2022. Behind the Curtain: Learning Occluded Shapes for 3D Object Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 2893–2901.

Yang, J.; Pavone, M.; and Wang, Y. 2023. FreeNeRF: Improving Few-Shot Neural Rendering with Free Frequency Regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8254–8263.

Yu, A.; Ye, V.; Tancik, M.; and Kanazawa, A. 2021. PixelNeRF: Neural Radiance Fields from One or Few Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4578–4587.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, 586–595.

Zhu, J.; Xie, J.; and Fang, Y. 2018. Learning Adversarial 3D Model Generation with 2D Image Enhancer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 7615–7622.