

# ConsistentNeRF: Enhancing Neural Radiance Fields with 3D Consistency for Sparse View Synthesis

Shoukang Hu<sup>1</sup> Kaichen Zhou<sup>2</sup> Kaiyu Li<sup>1</sup> Longhui Yu<sup>3</sup> Lanqing Hong<sup>4</sup> Tianyang Hu<sup>4</sup>  
 Zhenguo Li<sup>4\*</sup> Gim Hee Lee<sup>5\*</sup> Ziwei Liu<sup>1\*</sup>  
<sup>1</sup> Nanyang Technological University <sup>2</sup> University of Oxford <sup>3</sup> Peking University  
<sup>4</sup> Huawei Noah’s Ark Lab <sup>5</sup> National University of Singapore

## Abstract

Neural Radiance Fields (NeRF) has demonstrated remarkable 3D reconstruction capabilities with dense view images. However, its performance significantly deteriorates under sparse view settings. We observe that learning the 3D consistency of pixels among different views is crucial for improving reconstruction quality in such cases. In this paper, we propose ConsistentNeRF, a method that leverages depth information to regularize both multi-view and single-view 3D consistency among pixels. Specifically, ConsistentNeRF employs depth-derived geometry information and a depth-invariant loss to concentrate on pixels that exhibit 3D correspondence and maintain consistent depth relationships. Extensive experiments on recent representative works reveal that our approach can considerably enhance model performance in sparse view conditions, achieving improvements of up to 94% in PSNR, 76% in SSIM, and 31% in LPIPS compared to the vanilla baselines across various benchmarks, including DTU, NeRF Synthetic, and LLFF.

## 1. Introduction

Novel view synthesis is a longstanding challenge in the fields of computer vision and graphics. The objective is to generate photorealistic images from perspectives that were not originally captured [10, 15, 17, 24]. Recently, the employment of coordinate-based representation learning in 3D vision has increased the popularity of novel view synthesis. Neural Radiance Fields (NeRF) [21] serves as a notable example that leverages a coordinate-based neural network and dense proximal views to yield high-quality and realistic outcomes. However, NeRF’s capacity for realistic novel view synthesis is constrained in sparse view settings, due to the insufficiency of supervisory information and the inherent challenges of learning 3D consistency from limited data [8]. This limitation leads to unsatisfactory perfor-

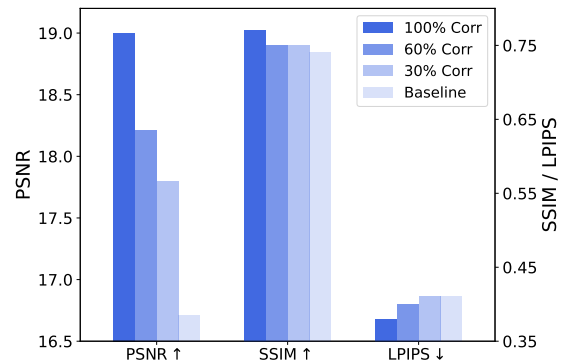


Figure 1. Performance (PSNR $\uparrow$ , SSIM $\uparrow$ , LPIPS $\downarrow$ ) comparison of NeRF with different levels of multi-view 3D consistency information. Using more multi-view 3D consistency constraints leads to better model performance.

mance and restricts the method’s applicability in real-world situations. To address the limitations of NeRF in sparse view settings, researchers have proposed two main strategies. The first strategy involves pre-training NeRF on large-scale datasets containing multiple scenes and subsequently fine-tuning the model [2, 3, 10, 15, 17, 24, 30, 33, 38]. The second strategy introduces additional regularization to optimize NeRF [5, 9, 14, 22, 26, 32, 35]. However, these approaches tend to focus primarily on pixel-level color and depth within a single view, rather than emphasizing both multi-view and single-view 3D consistency. In contrast, existing works dedicated to other 3D tasks, such as depth estimation and scene synthesis, demonstrate that 3D consistency is particularly important for accurate 3D appearance and geometry reconstruction [7, 25, 41].

In the field of 3D reconstruction, there are two types of 3D consistency relationships: multi-view and single-view 3D consistency. Multi-view 3D consistency refers to the correspondence between pixels that result from projecting the same 3D scene point into different views. To achieve this correspondence, the predicted color and depth must

\*Joint last authorship.

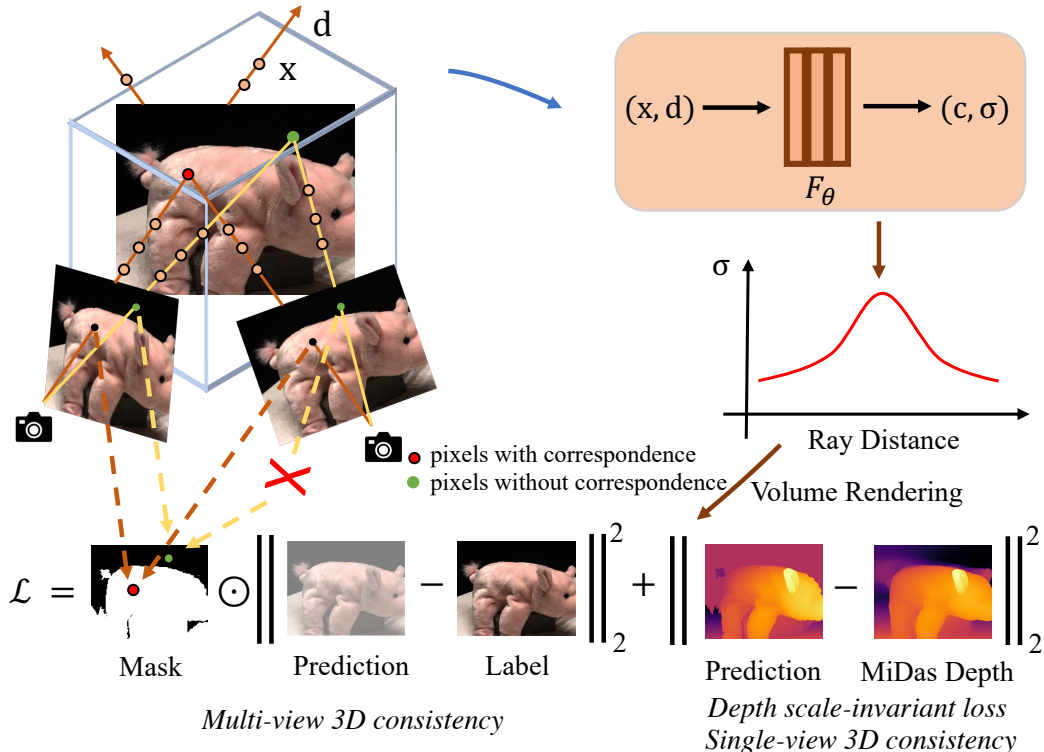


Figure 2. **The demonstration of proposed multi-view and single-view 3D consistency regularization.** We regularize multi-view 3D consistency by utilizing the multi-view depth correspondence among different views to mask pixels satisfying 3D correspondence (the red point) or not (the green point) and construct the loss based on the mask information. We also regularize single-view 3D consistency by constructing a depth scale-invariant loss function based on the monocular depth predicted from state-of-the-art MiDas model.

match and satisfy the homography warping relationship, as shown in Fig.1. Our evaluation demonstrates that including increasing amounts of 3D correspondence information into NeRF optimization improves performance in sparse view settings, highlighting the importance of 3D consistency as discussed in Sec.3.2. Single-view 3D consistency refers to the 3D geometric relationship of pixels within the same view. However, incorporating both multi-view and single-view 3D consistency into NeRF optimization poses a challenging problem.

In this study, we introduce Consistent Neural Radiance Fields (ConsistentNeRF), a solution that explicitly integrates multi-view and single-view 3D consistency to improve performance in sparse view scenarios. Specifically, to direct NeRF optimization towards pixels that fulfill the multi-view correspondence relationship, ConsistentNeRF selects these pixels based on depth-derived geometric information and assigns higher loss weights during training. For single-view consistency, we utilize a depth-invariant loss to extract 3D consistency information from nearby views employing the DPT Large pre-trained model [23]. Our proposed method achieves state-of-the-art results compared to existing approaches, including NeRF [21], DSNeRF [5], Mip-

NeRF [1], InfoNeRF [14], DietNeRF [9], RegNeRF [22], MVSNeRF [2], GeoNeRF [12], and ENeRF [16], across various datasets such as DTU dataset [11], Forward-Facing LLFF dataset [20] and Realistic Synthetic NeRF dataset [21].

The main contributions of this work contain three parts:

1. We introduce ConsistentNeRF, a method that effectively combines multi-view and single-view 3D consistency to improve sparse view synthesis performance.
2. Our approach utilizes depth-derived geometric information and a depth-invariant loss, achieving state-of-the-art results compared to existing methods across various datasets.
3. The significant improvements demonstrated by ConsistentNeRF showcase the effectiveness of the proposed method for enhancing 3D consistency in Neural Radiance Fields.

## 2. Related Works

Two methods were proposed to enhance the generalizability of NeRF with sparse-views: incorporating prior knowledge and introducing additional ground-truth information.

**View Synthesis with Prior Knowledge.** Pre-training neural networks with a large amount of data is a popular approach to incorporate prior knowledge and reduce the need for dense views in rendering novel 3D scenarios. Algorithms such as SSRF [3], GRF [30], Point-NeRF [36], IBRNET [33], PixelNeRF [38], Neural rays [18] and MVNeRF [2] use pre-trained models to extract feature maps from source views, which are then used to form appearance and geometry features for points in target views. Despite their effectiveness in dealing with sparse views, these algorithms still experience a significant decrease in performance when tested on scenarios with dense views or sparse views. ConsistentNeRF, on the other hand, improves the performance of models under sparse view settings without adding to the computational burden by incorporating 3D consistency relationships to regulate the optimization process.

**View Synthesis with Additional Information.** This research introduces additional information to assist the view synthesis process in sparse-view scenarios. DSNeRF [5], GeoNeRF [12] and ENeRF [16] incorporate geometry constraints using ground truth or "free" depth information. CodeNeRF [10], DoubleField [28], ShaRF [24], Improving [4], and DietNeRF [9] introduce object-centric shape or semantic information to build better correspondences among views. RegNeRF [22] and RapNeRF [39] introduce regularization, but none have used cross-view 3D consistency. In this work, the optimization of NeRF is regularized through 3D consistency relationships. Our concurrent work SPARF [31] applies the network mapping to derive the correspondence relationship among different views.

## 3. Method

### 3.1. Background

**Neural Radiance Fields.** The Radiance Field learns a continuous function which takes as input the 3D location  $\mathbf{x}$  and unit direction  $\mathbf{d}$  of each point and predicts the volume density  $\sigma \in [0, \infty)$  and color value  $\mathbf{c} \in [0, 1]^3$ . In NeRF [21], this continuous function is parameterized by a multi-layer perception (MLP) network  $F_\theta : (\gamma(\mathbf{x}), \gamma(\mathbf{d})) \rightarrow (\mathbf{c}, \sigma)$ , where the weight parameters  $\theta$  are optimized to generate the volume density  $\sigma$  and directional emitted color  $\mathbf{c}$ ,  $\gamma$  is the predefined positional embedding applied to  $\mathbf{x}$  and  $\mathbf{d}$ , which maps the inputs to a higher dimensional space.

**Volume Rendering.** Given the Neural Radiance Field (NeRF), the color of any pixel is rendered with principles from classical volume rendering [13] the ray  $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$  cast from the camera origin  $\mathbf{o}$  through the pixel along the unit direction  $\mathbf{d}$ . In volume rendering, the volume density  $\sigma(\mathbf{x})$  can be interpreted as the probability density at an infinitesimal distance at location  $\mathbf{x}$ . With the near and far bounds  $t_n$  and  $t_f$ , the expected color  $\hat{C}_\theta(\mathbf{r})$  of camera ray

$\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$  is defined as

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

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

where  $T(t)$  denotes the accumulated transmittance along the direction  $\mathbf{d}$  from  $t_n$  to  $t$ . In practice, the continuous integral is approximated by using the quadrature rule [19] and reduced to the traditional alpha compositing. The neural radiance field is then optimized by constructing the photometric loss  $\mathcal{L}$  between the rendered pixel color  $\hat{C}_\theta(\mathbf{r})$  and ground truth color  $C(\mathbf{r})$ :

$$\mathcal{L} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{r} \in \mathcal{R}} \|\hat{C}_\theta(\mathbf{r}) - C(\mathbf{r})\|_2^2, \quad (2)$$

where  $\mathcal{R}$  denotes the set of rays, and  $|\mathcal{R}|$  is the number of rays in  $\mathcal{R}$ .

### 3.2. Preliminary: Multi-view Pixel-wise 3D Consistency

In this section, we demonstrate the importance of considering the correspondence, i.e., multi-view 3D consistency, in the optimization process. With no loss of generality, we define  $\mathcal{M}$  to be the set containing pixels satisfying correspondence relationship and  $\mathcal{T}$  to be the correspondence relationship between pixel  $(i, j)$  and  $(m, n) := \mathcal{T}((i, j))$ . The 3D multi-view appearance consistency is defined in Definition 3.1. Similarly, we also define the 3D multi-view geometry consistency and details are shown in Appendix. B. By involving the proposed mask in Sec. 3.3, we select and assign larger loss weights to pixels that satisfy the homography warping relationship between source views and target views, i.e., the correspondence relationship. We compare the performance (PSNR $\uparrow$ , SSIM $\uparrow$ , LPIPS $\downarrow$ ) of assigning larger weights to different portions (30%, 60%, 100%) of pixels satisfying the correspondence relationship in the DTU data set. The baseline is the original NeRF model that treats all pixels equally during the optimization process. As shown in Fig. 1, assigning large weights to more pixels satisfying correspondence leads to better model performance. More details can be found in Appendix C.

#### Definition 3.1 (Multi-view Appearance Consistency)

*The multi-view appearance consistency refers to the color difference between the pixel  $(i, j) \in \mathcal{M}$  (in the left view of Fig. 3) and its corresponding pixel  $(m, n) := \mathcal{T}((i, j))$  (in the right view of Fig. 3) should be smaller than a threshold value  $\epsilon_c$ , i.e.:*

$$\|C_\theta(\mathbf{r}_{ij}) - C_\theta(\mathbf{r}_{mn})\|_2^2 \leq \epsilon_c, \quad (3)$$

where  $C_\theta(\mathbf{r}_{ij})$  and  $C_\theta(\mathbf{r}_{mn})$  are color labels of pixel  $(i, j)$  and  $(m, n)$ .

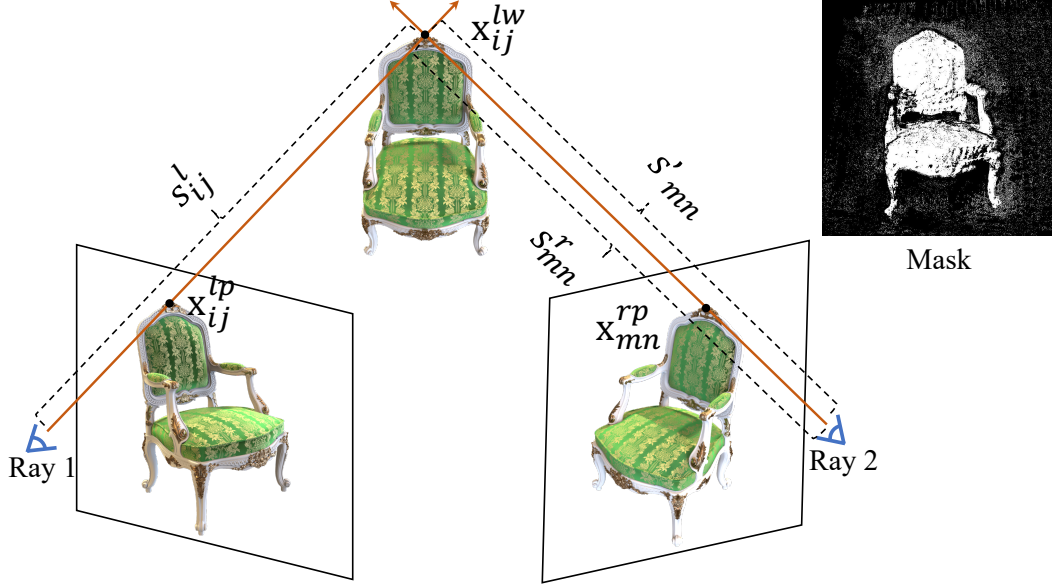


Figure 3. Illustration of deriving Depth-based Mask. We first derive the world coordinate  $\mathbf{x}_{ij}^{lw}$  of pixel  $(i, j)$  in the left view and then project the world coordinate  $\mathbf{x}_{ij}^{lw}$  into the right view, which leads to pixel  $(m, n)$ . If the difference of projected depth  $s'_{mn}$  and depth label  $s_{mn}^r$  is less than a threshold  $\alpha$ , the pixel  $(i, j)$  and  $(m, n)$  are marked as pixels satisfying 3D correspondence. By convention, depth is defined as the coordinate value along the z-axis in the corresponding camera coordinate system.

### 3.3. Neural Radiance Fields with Multi-view 3D Consistency

Based on the importance of 3D correspondence, we propose ConsistentNeRF to enforce NeRF-based algorithms to focus on the 3D correspondence relationship. Given a series of images for the specific scenario, it masks pixels satisfying 3D correspondence relationship between source views and target views. With no loss of generality, we show the derivation of mask in two views. As shown in Fig. 3, ConsistentNeRF samples a bunch of pixels  $\{(i, j)\}$  with coordinates  $\{\mathbf{x}_{ij}^{lp} = [i, j, 1]^T\}$  in the left camera coordinate, where  $l$  denotes the left camera view and  $p$  denotes the pixel coordinate. For each pixel  $(i, j)$ , one camera ray is cast from the camera origin  $\mathbf{o}$  along with the ray direction  $\mathbf{d}$ . With the estimated depth  $s_{ij}^l$  of pixel  $(i, j)$  in the left camera view, the world coordinate of the intersection point  $\mathbf{x}_{ij}^{lw}$  can be derived as

$$\mathbf{x}_{ij}^{lw} = (\mathbf{R}^l)^{-1} \mathbf{K}^{-1} \cdot (s_{ij}^l \cdot \mathbf{x}_{ij}^{lp}), \quad (4)$$

where  $\mathbf{R}^l$  is the world-to-camera transformation matrix of the left camera view,  $\mathbf{K}$  is the camera intrinsic matrix.

To get the pixel coordinate of the intersection point  $\mathbf{x}_{ij}^{lw}$  in the right view, the estimated world coordinate  $\mathbf{x}_{ij}^{lw}$  is transformed into the image plane of the right camera view with the world-to-camera transformation matrix  $\mathbf{R}^r$  and camera intrinsic matrix  $\mathbf{K}$  as follows:

$$s'_{mn} \cdot \mathbf{x}_{mn}^{rc} = \mathbf{K} \mathbf{R}^r \mathbf{x}_{ij}^{lw}, \quad (5)$$

where  $\mathbf{x}_{mn}^{rc} = (m, n, 1)$  is the pixel coordinate by projecting the intersection point  $\mathbf{x}_{ij}^{lw}$  onto the right camera image plane,  $s'_{mn}$  is the estimated depth of the intersection point  $\mathbf{x}_{ij}^{lw}$  in the right camera.

Pixels  $(i, j)$  and  $(m, n)$  are masked as pixels with 3D correspondence relationship when 1) the pixel  $(m, n)$  is not out of the boundary of the right image plane and 2) the transformed depth  $s'_{mn}$  and depth  $s_{mn}^r$  of pixel  $(m, n)$  are sufficiently close. Pixel  $(i, j)$  is regarded as a pixel that does not satisfy 3D correspondence under the sparse view setting and is excluded when it cannot find a pixel that satisfies the above condition in all training views. Following the above derivation, we set a threshold  $\alpha$  to mask pixels with 3D correspondence relationship as follows:

$$|s_{mn}^r - s'_{mn}| < \alpha \rightarrow \text{pixel } (i, j) \in \mathcal{M}, \mathcal{T}((i, j)) = ((m, n)), \quad (6)$$

where  $\mathcal{M}$  is defined to be the set containing masked pixels and  $\mathcal{T}$  defines the correspondence relationship between pixel  $(i, j)$  and  $(m, n) := \mathcal{T}((i, j))$ .

With the derived mask, the loss function is defined as

$$\mathcal{L} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{r} \in \mathcal{R} \cap \mathcal{M}} \|\hat{C}_\theta(\mathbf{r}) - C(\mathbf{r})\|_2^2 + \lambda \sum_{\mathbf{r} \notin \mathcal{R} \cap \mathcal{M}} \|\hat{C}_\theta(\mathbf{r}) - C(\mathbf{r})\|_2^2, \quad (7)$$

where  $\mathcal{R}$  denotes the set of rays, the coefficient  $\lambda \ll 1$  controls the loss ratio of emphasizing the pixels satisfying the correspondence relationship.



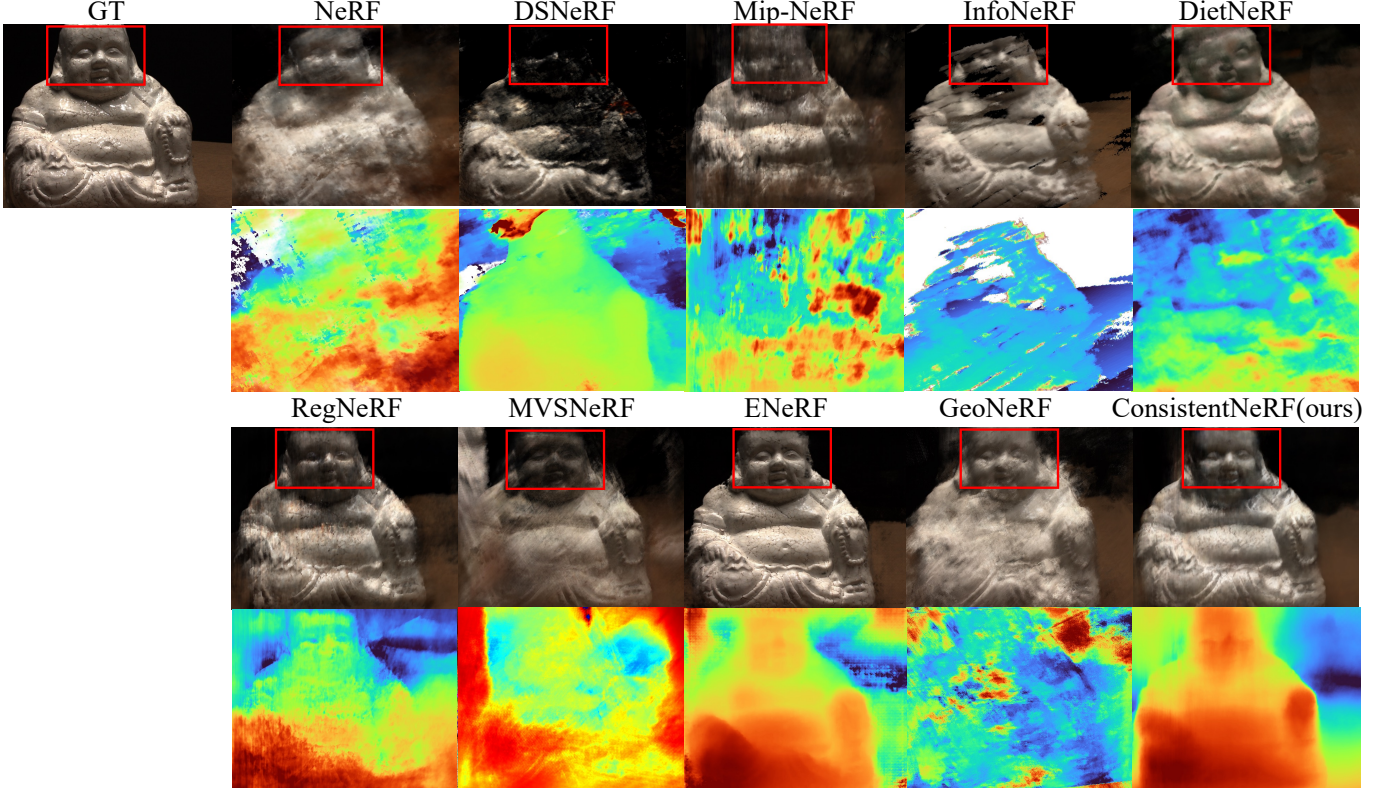


Figure 4. Novel View Synthesis Results on DTU data set with 3 views as input. We observe that the baselines suffer from blur results, while our ConsistentNeRF can produce sharp results with fine-grained details.

Note that according to Definition 3.1, the predicted color difference between pixels  $(i, j)$  and  $(m, n)$  (in the left/right view of Fig. 3) should be smaller than a threshold value  $\epsilon_c$ , i.e.,

$$\|\hat{C}_\theta(\mathbf{r}_{ij}) - \hat{C}_\theta(\mathbf{r}_{mn})\|_2^2 \leq \epsilon_c, \quad (8)$$

where  $\hat{C}_\theta(\mathbf{r}_{ij})$  and  $\hat{C}_\theta(\mathbf{r}_{mn})$  are predicted colors of pixel  $(i, j)$  and  $(m, n) \in \mathcal{M}$ . As shown in Proposition 1, the above loss function, which focuses on the pixels selected by the mask, implicitly emphasizes the appearance consistency in the optimization of NeRF. The proof is provided in Appendix A. We also show that it emphasizes the geometry consistency in the optimization of NeRF (see Appendix B for more details).

**Proposition 1 (Multi-view Appearance Consistency)**

Directly minimizing the above appearance consistency leads to trivial solution  $\hat{C}_\theta(\mathbf{r}_{ij}) = \hat{C}_\theta(\mathbf{r}_{mn}) = 0$ . Focusing on minimizing the errors between predicted color values and their ground truth for pixels included by the mask as in Eqn. (7) would help to emphasize the appearance consistency:

$$\begin{aligned} & \|\hat{C}_\theta(\mathbf{r}_{ij}) - C(\mathbf{r}_{ij})\|_2^2 + \|\hat{C}_\theta(\mathbf{r}_{mn}) - C(\mathbf{r}_{mn})\|_2^2 \\ & \geq \frac{1}{4} \|\hat{C}_\theta(\mathbf{r}_{ij}) - \hat{C}_\theta(\mathbf{r}_{mn})\|_2^2 - \epsilon_c/2. \end{aligned} \quad (9)$$

The above estimated mask locates pixels satisfying 3D correspondence relationship, which enforces NeRF to focus on the optimization of 3D consistency.

**3.4. Neural Radiance Fields with Single-view 3D consistency**

In addition to regularizing appearance and geometry consistency among different views, we also propose to regularize 3D consistency in the same view by using the depth predicted from state-of-the-art monocular depth estimation method MiDaS [23] as additional supervision. Considering that the depth predicted from monocular depth estimation methods can not guarantee the scale-invariant property, we apply the depth-invariant geometry consistency regularization to measure the depth relationships between pixels in the same patch. For predicted depth maps  $s'_{mn}$  and MiDaS depths  $s^r_{mn}$  of pixels in a patch, each with  $N$  pixels indexed by  $(m, n)$ , we apply the scale-invariant mean squared depth error (in log space) defined in [6], i.e.,

$$\begin{aligned} D(s'_{mn}, s^r_{mn}) &= \frac{1}{2N} \sum_{m,n} (\log s'_{mn} - \log s^r_{mn} + \\ & \frac{1}{N} \sum_{m,n} (\log s^r_{mn} - \log s'_{mn}))^2. \end{aligned} \quad (10)$$

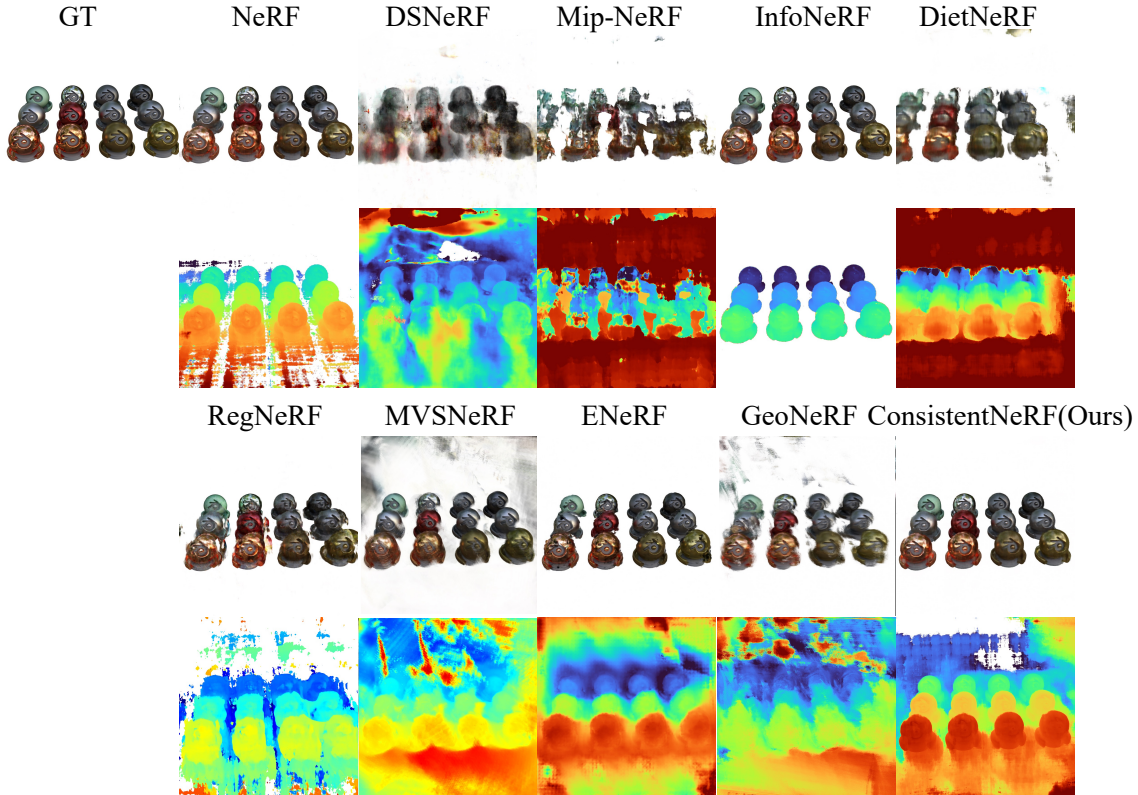


Figure 5. Novel View Synthesis Results on NeRF Synthetic data set with 3 views as input. We observe that the baselines suffer from blur results, while our ConsistentNeRF can produce sharp results with fine-grained details.

For any depth prediction  $s'_{mn}$ ,  $e^{\frac{1}{N} \sum_{m,n} (\log s^r_{mn} - \log s'_{mn})^2}$  is the scale that best aligns it to MiDas depth. Our intuition behind the above idea is that although the scale of the predicted depth from MiDas models is not accurate, the local structure (relative relationship) of predicted depth in a small patch contains a relatively accurate 3D consistency relationship, which can be used to regularize NeRF’s optimization.

## 4. Experiments

### Datasets.

We evaluate the proposed method on three diverse datasets, namely the real-world multi-view DTU dataset [11], Forward-Facing LLFF dataset [20] and Realistic Synthetic NeRF dataset [21]. Specifically, we follow PixelNeRF [38] to split the DTU dataset into 88 training scenes and 16 testing scenes. We utilize the 88 training scenes to pre-train the IBNet [33] and MVSNeRF [2] models. For each testing scene across the three datasets, we follow MVSNeRF to select three views from 20 nearby views as training views, and four views as testing views. In accordance with prior NeRF techniques, we evaluate all the methods on the DTU dataset with object masks applied to the rendered and ground truth images.

**Evaluation Metrics.** For performance comparison, we report the mean of peak signal-to-noise ratio (PSNR) [27], structural similarity index (SSIM) [34] and Learned Perceptual Image Patch Similarity (LPIPS) perceptual metric [40].

**Implementation Details.** We compare our method with NeRF based methods, including NeRF [21], DSNeRF [5], Mip-NeRF [1], InfoNeRF [14], DietNeRF [9], RegNeRF [22], MVSNeRF [2], GeoNeRF [12], and ENeRF [16]. For all NeRF [21] based methods which do not require pre-training, we directly train the model from scratch for each target scene. In our experiments, we use the depth extracted from a pre-trained MVSNeRF [2] to derive the mask. The depth also serves as the supervision for DSNeRF [5] and our methods for fair comparisons. For MiDas Depth, we use the DPT Large pre-trained model to derive the monocular depth information [23]. All mentioned methods (NeRF, DSNeRF, ConsistentNeRF) are trained with 50,000 iterations. For multi-view 3D consistency constraint, the threshold  $\alpha$  is set to be 0.1 and  $\lambda$  is set to be 0.1 on DTU, LLFF and NeRF Synthetic data set. We run each method with four random seeds and report the mean results. More implementation details are provided in Appendix D.

**Initialization for Stable Optimization.** During our experiments, we observe that NeRF is prone to a catastrophic



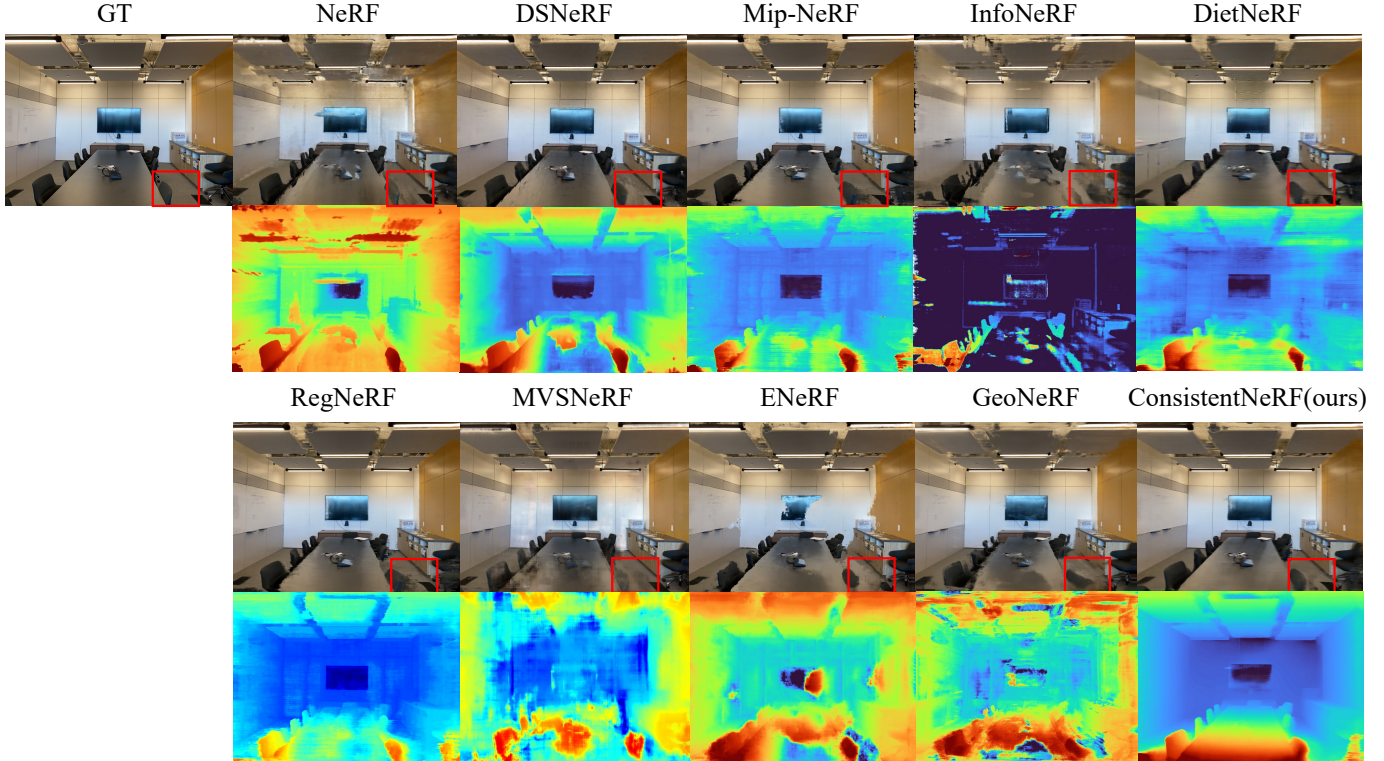


Figure 6. Novel View Synthesis Results on LLFF data set with 3 views as input. We observe that the baselines suffer from blur results, while our ConsistentNeRF can produce sharp results with fine-grained details.

Table 1. Performance (PSNR, SSIM and LPIPS) comparison among state-of-the-art NeRF methods on DTU, NeRF Synthetic and Forward-Facing data sets.  $\uparrow$  means the larger is better;  $\downarrow$  means the smaller is better.

Method	Setting	Pretrain	Real Data (DTU)			Synthetic Data (NeRF)			Forward-Facing (LLFF)		
			PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
NeRF [21]	3-view	$\times$	11.40	0.50	0.49	14.59	0.82	0.29	12.52	0.34	0.60
DSNeRF [5]		$\times$	11.80	0.52	0.49	15.13	0.82	0.30	13.10	0.35	0.62
Mip-NeRF [1]		$\times$	15.87	0.73	0.42	16.52	0.80	0.28	20.19	0.71	0.47
InfoNeRF [14]		$\times$	17.54	0.62	0.44	14.51	0.75	0.30	16.78	0.47	0.56
DietNeRF [9]		$\times$	12.94	0.42	0.64	17.55	0.77	0.28	19.84	0.58	0.51
RegNeRF [22]		$\times$	21.57	0.84	<b>0.31</b>	17.39	0.82	0.26	20.36	0.72	0.45
MVSNeRF [2]		$\checkmark$	19.17	0.80	0.34	15.12	0.82	0.29	18.99	0.68	<b>0.41</b>
GeoNeRF [12]		$\checkmark$	16.51	0.56	0.43	17.67	0.73	0.33	17.76	0.50	0.49
ENeRF [16]		$\checkmark$	18.65	0.83	0.40	18.14	<b>0.83</b>	<b>0.20</b>	20.30	<b>0.75</b>	0.45
ConsistentNeRF (Ours)		$\times$	<b>22.14</b>	<b>0.88</b>	0.34	<b>19.63</b>	<b>0.83</b>	<b>0.20</b>	<b>21.77</b>	0.73	0.43

failure at the initialization stage in which MLP emits negative values before the ReLU activation. In this case, all predicted  $\sigma$  values are zero, and gradients back-propagated from the loss function to MLP parameters are zero, leading to the failure of the optimization. To address the above failure, Mip-NeRF [1] proposes to use the softplus function to stabilize the optimization. However, we observe that NeRF overfits to training views by using the softplus function in

the sparse view setting. In this paper, we propose to modify the initialization of bias parameters in the MLP to guarantee both stable optimization and good generalization ability. During our experiments, we find that initializing the value of bias parameters in MLP using a uniform distribution between 0 and 1 leads to acceptable results. The comparison results are reported in Appendix E.

Table 2. Ablation study on ablating two consistency regularizations on the LLFF data set with 3 training views as input. For performance (PSNR, SSIM and LPIPS) comparison,  $\uparrow$  means the larger is better;  $\downarrow$  means the smaller is better.

Method	Forward-Facing (LLFF)		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
ConsistentNeRF	21.77	0.73	0.43
w/o Single-view Consistency	20.75	0.73	0.44
w/o Multi-view Consistency	20.85	0.73	0.44
w/o Single-view and Multi-view Consistency	20.36	0.72	0.45

#### 4.1. View Synthesis Results

In the experiment, we evaluate the performance achieved by the above-mentioned NeRF models under sparse view settings and compare them with our proposed ConsistentNeRF. Quantitative results are shown in Tab. 1. For 3 input view settings, our proposed ConsistentNeRF could largely improve the performance of the original NeRF, e.g., 70% relative PSNR improvement is achieved on the DTU data set. Besides, when compared with DSNeRF which directly introduces depth constrain, our ConsistentNeRF could bring larger performance improvement through regularizing the optimization with 3D consistency relationship. When further compared with NeRF-based methods with additional regularization, like Mip-NeRF [1], InfoNeRF [14], DietNeRF [9], RegNeRF [22], ConsistentNeRF consistently shows better performance. Quantitative results in Fig. 4, Fig. 5 and Fig. 6 also support the above claim.

We also compare ConsistentNeRF with MVSNerF [2], GeoNeRF [12] and ENeRF [16], which require the pre-training and per-scene optimization. As shown in Tab. 1, MVSNerF, GeoNeRF and ENeRF produce better results than the vanilla NeRF in the 3 view setting. However, we still observe some inconsistent results when the testing view is far from the training views. For example, as shown in Fig. 4, Fig. 5 and Fig. 6, these methods produce images with blur results and poor lighting, while our proposed ConsistentNeRF can predict more sharp results and the correct lighting effect of the pixels in the target view using the multi-view and single-view 3D consistency constraint.

#### 4.2. Ablation Study

As shown in Tab. 2, we ablate the performance of multi-view and single-view 3D consistency regularization introduced in Sec. 3.3. With either multi-view 3D consistency regularization or single-view 3D consistency regularization, ConsistentNeRF consistently outperforms the baseline model in all metrics. Adding both multi-view and single-view 3D consistency regularization leads to the best performance.

#### 5. Limitation

One limitation of our paper is that we adopt a pre-trained MVSNerF to derive the mask information for multi-view 3D consistency regularization. In real-world applications, it is hard to derive the mask information from a pre-trained NeRF model like MVSNerF. In the future work, one potential direction is to apply flow model [29] which directly utilizes RGB information to derive the correspondence relationship among pixels in different views. The other potential direction is to extend our framework to RGBD settings to derive the correspondence relationship among pixels in different views using the depth information from RGBD tensors. In addition, similar to most NeRF-based methods, our proposed optimization can not render images with high quality when the target view is far from source views as 3D correspondence relationship is hard to utilize in this case.

#### 6. Conclusion

In this paper, we target to the challenging sparse view synthesis problem and proposed ConsistentNeRF, which enhances Neural Radiance Fields with 3D Consistency. To build correspondences among pixels in different views, we propose a mask-based loss that locates the pixels with 3D consistency, instead of treating all pixels equally in the training objective. Moreover, we adopt a depth consistency regularization among pixels in the same patch to regularize the 3D consistency among pixels in the same view. Our experimental results demonstrate that our proposed methods significantly improve the performance of representative NeRF methods with sparse view settings and can bring larger performance improvement than previous depth-based methods. These promising results suggest that consistency-based NeRF is an important direction for rendering images with both correct geometry and fine-grained details. In conclusion, our proposed methods offer a new and effective solution to the challenging problem of sparse view synthesis and have promising potential for future applications in various fields.



## References

- [1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. [2](#), [6](#), [7](#), [8](#), [12](#)
- [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. *arXiv preprint arXiv:2103.15595*, 2021. [1](#), [2](#), [3](#), [6](#), [7](#), [8](#)
- [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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7911–7920, 2021. [1](#), [3](#)
- [4] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6260–6269, 2022. [3](#)
- [5] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. *arXiv preprint arXiv:2107.02791*, 2021. [1](#), [2](#), [3](#), [6](#), [7](#)
- [6] David Eigen, Christian Puhresch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in Neural Information Processing Systems*, 27, 2014. [5](#)
- [7] Clément Godard, Oisín Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3828–3838, 2019. [1](#)
- [8] Yi-Hua Huang, Yue He, Yu-Jie Yuan, Yu-Kun Lai, and Lin Gao. Stylizednerf: consistent 3d scene stylization as stylized nerf via 2d-3d mutual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18342–18352, 2022. [1](#)
- [9] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5885–5894, 2021. [1](#), [2](#), [3](#), [6](#), [7](#), [8](#)
- [10] Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12949–12958, 2021. [1](#), [3](#)
- [11] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413, 2014. [2](#), [6](#)
- [12] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. Geonerf: Generalizing nerf with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18365–18375, 2022. [2](#), [3](#), [6](#), [7](#), [8](#)
- [13] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH Computer Graphics*, 18(3):165–174, 1984. [3](#)
- [14] Mijeong Kim, Seonguk Seo, and Bohyung Han. Infonerf: Ray entropy minimization for few-shot neural volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12912–12921, 2022. [1](#), [2](#), [6](#), [7](#), [8](#)
- [15] Jiaxin Li, Zijian Feng, Qi She, Henghui Ding, Changhu Wang, and Gim Hee Lee. Mine: Towards continuous depth mpi with nerf for novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12578–12588, 2021. [1](#)
- [16] Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. [2](#), [3](#), [6](#), [7](#), [8](#)
- [17] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. *arXiv preprint arXiv:2107.13421*, 2021. [1](#)
- [18] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7824–7833, 2022. [3](#)
- [19] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. [3](#)
- [20] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics*, 38(4):1–14, 2019. [2](#), [6](#)
- [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 *European Conference on Computer Vision*, pages 405–421. Springer, 2020. [1](#), [2](#), [3](#), [6](#), [7](#)
- [22] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. *arXiv preprint arXiv:2112.00724*, 2021. [1](#), [2](#), [3](#), [6](#), [7](#), [8](#)
- [23] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1623–1637, 2020. [2](#), [5](#), [6](#)
- [24] Konstantinos Rematas, Ricardo Martin-Brualla, and Vittorio Ferrari. Sharf: Shape-conditioned radiance fields from a single view. *arXiv preprint arXiv:2102.08860*, 2021. [1](#), [3](#)
- [25] Chris Rockwell, David F Fouhey, and Justin Johnson. Pixel-synth: Generating a 3d-consistent experience from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14104–14113, 2021. [1](#)
- [26] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for

- neural radiance fields from sparse input views. *arXiv preprint arXiv:2112.03288*, 2021. 1
- [27] Umme Sara, Morium Akter, and Mohammad Shorif Uddin. Image quality assessment through fsim, ssim, mse and psnr—a comparative study. *Journal of Computer and Communications*, 7(3):8–18, 2019. 6
- [28] Ruizhi Shao, Hongwen Zhang, He Zhang, Mingjia Chen, Yan-Pei Cao, Tao Yu, and Yebin Liu. Doublefield: Bridging the neural surface and radiance fields for high-fidelity human reconstruction and rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15872–15882, 2022. 3
- [29] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision*, pages 402–419, 2020. 8
- [30] Alex Trevithick and Bo Yang. Grf: Learning a general radiance field for 3d representation and rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15182–15192, 2021. 1, 3
- [31] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. *arXiv preprint arXiv:2211.11738*, 2022. 3
- [32] Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. *arXiv preprint arXiv:2303.16196*, 2023. 1
- [33] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 1, 3, 6
- [34] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6
- [35] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5610–5619, 2021. 1
- [36] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5438–5448, 2022. 3
- [37] Lin Yen-Chen. Nerf-pytorch. <https://github.com/yenchenlin/nerf-pytorch/>, 2020. 12
- [38] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 1, 3, 6
- [39] Jian Zhang, Yuanqing Zhang, Huan Fu, Xiaowei Zhou, Bowen Cai, Jinchu Huang, Rongfei Jia, Binqiang Zhao, and Xing Tang. Ray priors through reprojection: Improving neural radiance fields for novel view extrapolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18376–18386, 2022. 3
- [40] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 6
- [41] Kaichen Zhou, Lanqing Hong, Changhao Chen, Hang Xu, Chaoqiang Ye, Qingyong Hu, and Zhenguo Li. Devnet: Self-supervised monocular depth learning via density volume construction. *arXiv preprint arXiv:2209.06351*, 2022. 1

## Appendix

### A. Multi-view 3D Appearance Consistency

#### Definition A.1 (Multi-view Appearance Consistency)

The multi-view appearance consistency refers to the color difference between the pixel  $(i, j) \in \mathcal{M}$  (in the left view of Fig. 3) and its corresponding pixel  $(m, n) := \mathcal{T}((i, j))$  (in the right view of Fig. 3) should be smaller than a threshold value  $\epsilon_c$ , i.e.:

$$\|C_\theta(\mathbf{r}_{ij}) - C_\theta(\mathbf{r}_{mn})\|_2^2 \leq \epsilon_c, \quad (11)$$

where  $C_\theta(\mathbf{r}_{ij})$  and  $C_\theta(\mathbf{r}_{mn})$  are color labels of pixel  $(i, j)$  and  $(m, n)$ .

#### Definition A.2 (Consistency of Estimated Appearance)

The multi-view 3D consistency of estimated appearance refers to the predicted color difference between pixel  $(i, j) \in \mathcal{M}$  and pixel  $(m, n) := \mathcal{T}((i, j))$  (in the left/right view of Fig. 3) should be smaller than a threshold value  $\epsilon_c$ , i.e.:

$$\|\hat{C}_\theta(\mathbf{r}_{ij}) - \hat{C}_\theta(\mathbf{r}_{mn})\|_2^2 \leq \epsilon_c, \quad (12)$$

where  $\hat{C}_\theta(\mathbf{r}_{ij})$  and  $\hat{C}_\theta(\mathbf{r}_{mn})$  are predicted color of pixel  $(i, j)$  and  $(m, n)$ .

#### Proposition 2 (Multi-view Appearance Consistency)

Directly minimizing appearance consistency in Definition A.2 leads to trivial solution  $\hat{C}_\theta(\mathbf{r}_{ij}) = \hat{C}_\theta(\mathbf{r}_{mn}) = 0$ . Focusing on minimizing the errors between predicted color values and their ground truth for pixels included by the Hard-Mask as in Eqn. (7) would help to emphasize the appearance consistency:

$$\begin{aligned} & \|\hat{C}_\theta(\mathbf{r}_{ij}) - C(\mathbf{r}_{ij})\|_2^2 + \|\hat{C}_\theta(\mathbf{r}_{mn}) - C(\mathbf{r}_{mn})\|_2^2 \\ & \geq \frac{1}{4} \|\hat{C}_\theta(\mathbf{r}_{ij}) - \hat{C}_\theta(\mathbf{r}_{mn})\|_2^2 - \epsilon_c/2. \end{aligned}$$

Proof:

$$\begin{aligned} & 2\|\hat{C}_\theta(\mathbf{r}_{ij}) - C(\mathbf{r}_{ij})\|_2^2 + 2\|\hat{C}_\theta(\mathbf{r}_{mn}) - C(\mathbf{r}_{mn})\|_2^2 \\ & \geq \|(\hat{C}_\theta(\mathbf{r}_{ij}) - \hat{C}_\theta(\mathbf{r}_{mn})) + (C(\mathbf{r}_{mn}) - C(\mathbf{r}_{ij}))\|_2^2 \\ & \geq \frac{1}{2} \|\hat{C}_\theta(\mathbf{r}_{ij}) - \hat{C}_\theta(\mathbf{r}_{mn})\|_2^2 - \|C(\mathbf{r}_{mn}) - C(\mathbf{r}_{ij})\|_2^2 \end{aligned}$$

The first inequality follows from the fact that for two vectors  $\mathbf{a}, \mathbf{b}$ ,

$$2\|\mathbf{a}\|_2^2 + 2\|\mathbf{b}\|_2^2 - \|\mathbf{a} + \mathbf{b}\|_2^2 = \|\mathbf{a} - \mathbf{b}\|_2^2 \geq 0.$$

The second inequality is due to the fact that

$$2\|\mathbf{a} + \mathbf{b}\|_2^2 - (\|\mathbf{a}\|_2^2 - 2\|\mathbf{b}\|_2^2) = \|\mathbf{a} + 2\mathbf{b}\|_2^2 \geq 0.$$

### B. Multi-view 3D Geometry Consistency

#### Definition B.1 (Multi-view Geometry Consistency)

The geometry consistency refers to the depth difference between the depth of pixel  $(m, n) \in \mathcal{M}$  in the right camera view and the depth generated by warping its corresponding pixel  $(i, j) := \mathcal{T}((m, n))$  from left camera to the right camera should be smaller than a threshold value  $\epsilon_s$ , i.e.:

$$\|s_{mn}^r - s'_{mn}\|_2^2 \leq \epsilon_s, \quad (13)$$

where  $s_{mn}^r$  is the depth for pixel  $(m, n)$  and  $s'_{mn}$  is the projected depth from left camera pixel  $(i, j)$ .

#### Definition B.2 (Consistency of Estimated Geometry)

The consistency of estimated geometry refers to the predicted depth difference between the depth of pixel  $(m, n) \in \mathcal{M}$  in the right camera view and the predicted depth generated by warping its corresponding pixel  $(i, j) := \mathcal{T}((m, n))$  from left camera to the right camera should be smaller than a threshold value  $\epsilon_s$ , i.e.:

$$\|\hat{s}_\theta(\mathbf{r}_{mn}) - \hat{s}'_\theta(\mathbf{r}_{mn})\|_2^2 \leq \epsilon_s, \quad (14)$$

where  $\hat{s}_\theta(\mathbf{r}_{mn})$  is the predicted depth for pixel  $(m, n)$  and  $\hat{s}'_\theta(\mathbf{r}_{mn})$  is the projected depth from left camera pixel  $(i, j)$ .

#### Proposition 3 (Multi-view Geometry Consistency)

Similar to multi-view Appearance Consistency Regularization, focusing on optimizing the error between predicted depth value and its ground truth for pixels included by Hard-Mask as in Eqn. (7) would help to emphasize the geometry consistency:

$$\begin{aligned} & \|\hat{s}_\theta(\mathbf{r}_{mn}) - s_{mn}^r\|_2^2 + \|\hat{s}'_\theta(\mathbf{r}_{mn}) - s'_{mn}\|_2^2 \\ & \geq \frac{1}{4} \|\hat{s}_\theta(\mathbf{r}_{mn}) - \hat{s}'_\theta(\mathbf{r}_{mn})\|_2^2 - \epsilon_s/2, \end{aligned} \quad (15)$$

Proof:

$$\begin{aligned} & 2\|\hat{s}_\theta(\mathbf{r}_{mn}) - s_{mn}^r\|_2^2 + 2\|\hat{s}'_\theta(\mathbf{r}_{mn}) - s'_{mn}\|_2^2 \\ & \geq \|(\hat{s}_\theta(\mathbf{r}_{mn}) - \hat{s}'_\theta(\mathbf{r}_{mn})) + (s_{mn}^r - s'_{mn})\|_2^2 \\ & \geq \frac{1}{2} \|\hat{s}_\theta(\mathbf{r}_{mn}) - \hat{s}'_\theta(\mathbf{r}_{mn})\|_2^2 - \|s_{mn}^r - s'_{mn}\|_2^2 \end{aligned}$$

The first inequality follows from the fact that for two vectors  $\mathbf{a}, \mathbf{b}$ ,

$$2\|\mathbf{a}\|_2^2 + 2\|\mathbf{b}\|_2^2 - \|\mathbf{a} + \mathbf{b}\|_2^2 = \|\mathbf{a} - \mathbf{b}\|_2^2 \geq 0.$$

The second inequality is due to the fact that

$$2\|\mathbf{a} + \mathbf{b}\|_2^2 - (\|\mathbf{a}\|_2^2 - 2\|\mathbf{b}\|_2^2) = \|\mathbf{a} + 2\mathbf{b}\|_2^2 \geq 0.$$

## C. Preliminary Study

By utilizing the homography warping relationship, we locate pixels satisfying 3D correspondence relationship. Based on the masked pixels among training views, we find the respective 3D points and randomly sample different portions (30%, 60%, 100%) of 3D points for the purpose of emphasizing the 3D correspondence. We conduct each experiment using 4 random seeds and report the mean results.

## D. Implementation Details

All our models are trained on the NVIDIA Tesla V100 Volta GPU cards. The NeRF-based models are implemented based on the code from [37]. For MVSNet, we follow the released code and checkpoint to pre-train and finetune the models. For Mask introduced in Sec. 3.3, we generate the mask information for each training image based on the correspondence among pixels in all training views.

## E. Solutions to Avoid Degenerate Results in NeRF

As mentioned in Sec. 3.3, NeRF is prone to a catastrophic failure at the initialization stage in which MLP emits negative values before the ReLU activation. To address this issue, Mip-NeRF [1] proposed to use the softplus function to yield a stable optimization process. However, we observe that NeRF overfits training views by using the softplus function in the sparse view setting. One possible reason could be that the predicted alpha value of sampled points should be sparse and dropping small values with ReLU activation could effectively improve the generalization ability. Based on the above consideration, we instead propose to modify the initialization of bias parameters in the MLP to guarantee both stable optimization and good generalization ability. As shown in Tab. 3, our proposed initialization effectively improves the performance of NeRF and avoid the degenerate results when compared with SoftPlus activation and the original NeRF setting.

Table 3. Performance (PSNR, SSIM and LPIPS) comparison between SoftPlus and our proposed stable initialization to avoid degenerate results in NeRF on the DTU data set with 3 training views as input.  $\uparrow$  means the larger is better;  $\downarrow$  means the smaller is better.

Method	Real Data (DTU)		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
ReLU	11.40	0.50	0.49
SoftPlus	14.26	0.68	0.45
Stable Initialization	16.91	0.73	0.41