Where and How: Mitigating Confusion in Neural Radiance Fields from Sparse Inputs

Yanqi Bao
yq_bao@smail.nju.edu.cn
Nanjing University
Nanjing, Jiangsu, China

Yuxin Li
liyuxin16@smail.nju.edu.cn
Nanjing University
Nanjing, Jiangsu, China

Jing Huo
huojing@nju.edu.cn
Nanjing University
Nanjing, Jiangsu, China

Tianyu Ding
tianyuding@microsoft.com
Microsoft Corporation
Redmond, USA

Xinyue Liang
mf21330051@smail.nju.edu.cn
Nanjing University
Nanjing, Jiangsu, China

Wenbin Li
liwenbin@nju.edu.cn
Nanjing University
Nanjing, Jiangsu, China

Yang Gao
gaoy@nju.edu.cn
Nanjing University
Nanjing, Jiangsu, China

ABSTRACT

Neural Radiance Fields from Sparse inputs (NeRF-S) have shown great potential in synthesizing novel views with a limited number of observed viewpoints. However, due to the inherent limitations of sparse inputs and the gap between non-adjacent views, rendering results often suffer from over-fitting and foggy surfaces, a phenomenon we refer to as "CONFUSION" during volume rendering. In this paper, we analyze the root cause of this confusion and attribute it to two fundamental questions: "WHERE" and "HOW". To this end, we present a novel learning framework, WaH-NeRF, which effectively mitigates confusion by tackling the following challenges: (i) **"WHERE" to Sample?** in NeRF-S—we introduce a Deformable Sampling strategy and a Weight-based Mutual Information Loss to address sample-position confusion arising from the limited number of viewpoints; and (ii) **"HOW" to Predict?** in NeRF-S—we propose a Semi-Supervised NeRF learning Paradigm based on pose perturbation and a Pixel-Patch Correspondence Loss to alleviate prediction confusion caused by the disparity between training and testing viewpoints. By integrating our proposed modules and loss functions, WaH-NeRF outperforms previous methods under the NeRF-S setting. Code is available https://github.com/bbbbby-99/WaH-NeRF.

CCS CONCEPTS

• **Computing methodologies** → **Rendering**; **Image-based rendering**.

KEYWORDS

Neural Radiance Field from Sparse Inputs; Volume Rendering; Semi-Supervised Learning

1 INTRODUCTION

The field of 3D modeling and scene synthesis has experienced a significant transformation with the advent of Neural Radiance Fields (NeRF) [19]. By utilizing deep neural networks, NeRF and its variants feature implicit representations of entire 3D scenes, resulting in remarkable progress in generating geometric 3D representations and synthesizing immersive novel views. Unlike traditional Novel View Synthesis methods [10, 25], NeRF exhibits higher efficiency through volume rendering. However, NeRF's ability to generate realistic renderings relies heavily on dense viewing inputs, which can be challenging to collect in real-world applications such as robotic navigation [24] and autonomous driving [27], among others.

In scenarios with insufficient inputs, several works [13, 35] have demonstrated the negative impact on vanilla NeRF performance and introduced *Neural Radiance Fields from Sparse inputs* (NeRF-S) to address this challenge. Unfortunately, sparse viewpoints often fail to provide enough information to generate a complete geometric 3D representation, leading to some derivative works of NeRF [32, 37] as well as traditional NVS methods focusing on adjacent viewpoints. However, this simplification restricts the rendering scope and application scenarios for NeRF, prompting recent research [13, 20] efforts to explore rendering a complete scene under sparse inputs.

For this, recent prominent works [20, 35] propose image-wise (or patch-wise) regularization (as Fig. 1(a)) on geometry and color by adding additional supervision [20] or warping for pseudo-label [35]. Although these approaches alleviate the gap between training and

testing viewpoints and achieve state-of-the-art performance, the additional supervision, such as depth image and pre-trained models, are often not feasible, and the warping can be time-consuming.

In this work, instead of relying on warping or additional supervision on rendered images, we approach this problem by analyzing the fundamental causes of untrustworthy geometry and color predictions during volume rendering. As shown in Fig. 2 (a-b), NeRF-S suffers from drifted sampling points near the surface (as black dashed box) and unstable, incorrect predictions before volume rendering (as black solid box), compared with vanilla NeRF. Intuitively, such uncertainty leads to foggy geometry and color errors in rendered images. Based on the above observations, we speculate that the uncertainty of surface and prediction in NeRF-S arises from the confusion of "WHERE to sample?" and "HOW to predict?", which significantly contributes to rendering collapse when dealing with sparse inputs. Unfortunately, existing works primarily focus on regularizing rendered images after volume rendering and pay little attention to such confusion during the volume rendering process.

To address this issue, we propose the following two modules to mitigate such **confusion** (as Fig. 1(b)):

WHERE to sample in NeRF-S? Due to the lack of necessary information for geometric 3D representation in NeRF-S, we introduce the prior knowledge of surface uniqueness and propose a Deformable Sampling strategy. This strategy employs a learnable offset to adaptively deform sampling positions, forcing density distribution to converge on the target accurate surface. Additionally, by analyzing the correlation between weight and offset, we design a Weight-based Mutual Information Loss to maintain rendering stability during the training phase.

HOW to predict in NeRF-S? To mitigate the over-fitting caused by the gap between training and testing viewpoints, we draw inspiration from consistency regularization in semi-supervised learning and propose a Semi-Supervised NeRF learning Paradigm based on pose perturbation. This paradigm adopts consistency regularization between unseen ray and perturbed rays generated by pose perturbation. To ensure local consistency, we introduce the Pixel-Patch Correspondence Loss (PPC) to alleviate unregistered pixel-wise correspondences from perturbation and improve the smoothness between adjacent rays.

By integrating the aforementioned modules in parallel, we mitigate the uncertainty of surface and prediction, as shown in Fig. 2 (d). Furthermore, compared to baseline, as shown in Fig. 2 (b) and existing work based on rendered image, RegNeRF [20], as shown in Fig. 2 (c), our proposed method answers the confusion of "Where" and "How" better. Its superiority is further demonstrated qualitatively and quantitatively, as detailed in Sec. 4. In summary, we make the following specific contributions in the paper:

- We demonstrate that the confusion of "WHERE to sample?" and "HOW to predict?" in NeRF-S are the main sources of rendering collapse in sparse input scenarios. Based on this insight, we propose WaH-NeRF, aiming to mitigate confusion before volume rendering.
- We propose a Deformable Sampling strategy and a Weight-based Mutual Information Loss to alleviate uncertainty in sample *positions*. Meanwhile, we develop a Semi-Supervised NeRF

(a) Existing work   (b) WaH-NeRF
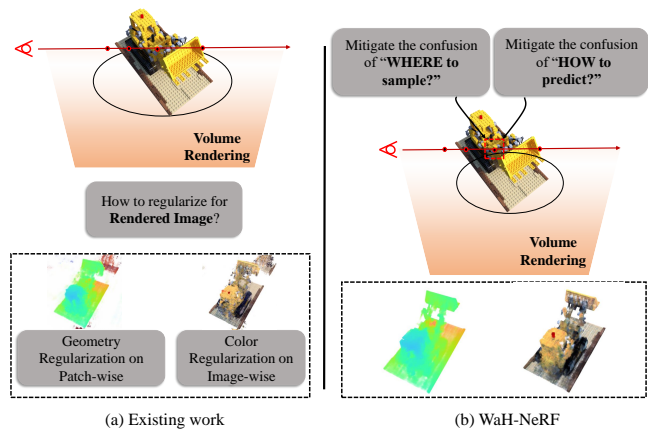
Figure 1: *Different from existing algorithms*: Existing state-of-the-art methods, e.g. RegNeRF [20] as (a), rely on regularization for rendered color and depth images, which ignore the causes of performance degradation before volume rendering. Our work, as shown in (b), analyzes the sample-render process and mitigates the confusion of "WHERE to sample?" and "HOW to predict?" in NeRF-S during volume rendering.

Paradigm based on pose perturbation and a Pixel-Patch Correspondence Loss to mitigate uncertainty in sample *predictions*.
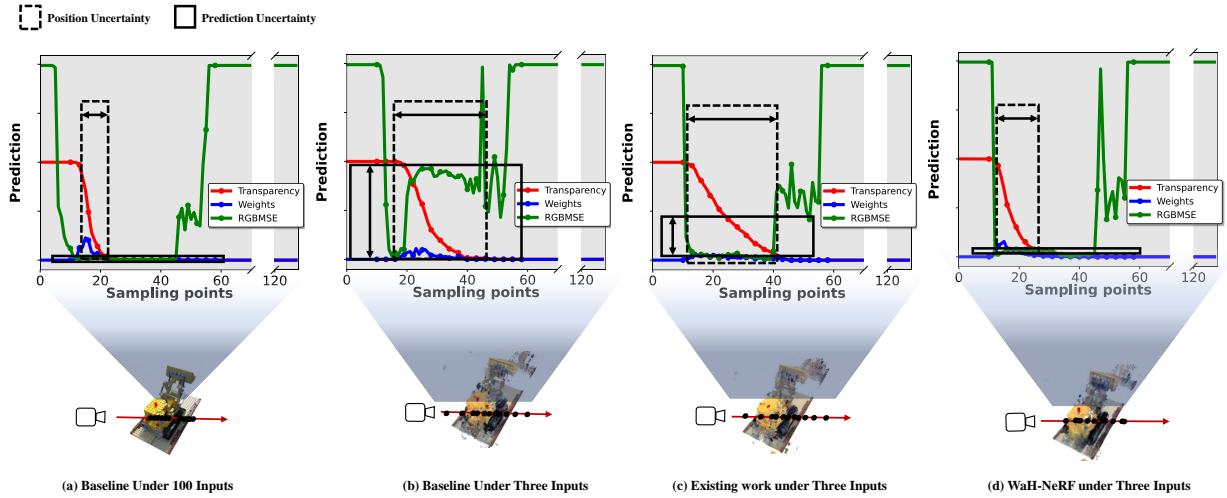- We show that WaH-NeRF achieves state-of-the-art performance compared to existing algorithms through extensive experiments, without any pre-trained models and time-consuming image warping. Our experimental results also demonstrate the effectiveness of these modules and losses in mitigating the "WHERE" and "HOW" confusion, respectively

2 RELATED WORK

2.1 Novel View Synthesis

Novel View Synthesis (NVS) is a fundamental problem in computer vision that has been extensively researched for years [7, 8, 42]. The purpose of NVS is generating images of unseen viewpoints by utilizing multiple images from seen viewpoints. Given multiple seen viewpoints, a straightforward idea is to first reconstruct the 3D scene and then project it onto 2D images under unseen poses [17, 18, 40]. For example, Penner and Zhang [22] directly transformed the images into 3D space with the aid of initial depth images, while more recent works had favored the use of deep neural networks (DNN) [10, 12, 16]. Zhou *et al.* [40] employed a learning framework for MultiPlane Images (MPI) representation, and [25] adopted a transformer-based encoder and decoder to learn a scalable implicit representation with attention architecture.

Recently, single-view input for NVS has gained traction as obtaining images from multiple viewpoints is not always feasible. In this scenario, explicit or implicit 3D representations are often learned from single-view input. [29, 34, 36]. Unlike multi-viewpoint settings, other works synthesised novel views using purely image-to-image transformations without 3D representations [26, 28]. Despite the significant progress made by NVS in both multi-view

(a) Baseline Under 100 Inputs **(b) Baseline Under Three Inputs** **(c) Existing work under Three Inputs** **(d) WaH-NeRF under Three Inputs**

Figure 2: *Uncertainty analysis under sparse inputs.* **We perform an analysis on the 128 sampling predictions of a random ray, which includes the RGBMSE, Weight and Transparency. RGBMSE is the mean squared error loss between the RGB color of each sampling position and the ground truth at corresponding pixel, representing the uncertainty of the color prediction. The Weight and Transparency are intermediate outputs of the network, which approximate the position of the surface. We use the width of the rectangular box to represent the uncertainty (or confusion) about position and prediction. Our results show that WaH-NeRF successfully mitigates the confusion caused by sparse inputs, which is further elaborated in the Appendix.**

and single-view settings, the emergence of Neural Radiance Fields (NeRF) [19] has brought new research perspectives to this field due to its simplicity and state-of-the-art performance. In contrast to most of the aforementioned NVS works that only infer discrete viewpoints, NeRF can render 360-degree continuous viewpoints without explicit representations.

2.2 Neural Radiance Field from Sparse Inputs

NeRF [19] has shown a remarkable ability to produce photorealistic renderings of unseen viewpoints under the supervision of images from numerous input views. However, its performance drastically decreases when the number of input views is limited. Recent works addressed this challenge by regularizing rendered color or depth images. DSNeRF [9] and DDNeRF [23] advocated the introduction of depth supervision by structure-from-motion (sfm) to compensate for the lack of information in sparse views, while DietNeRF [11] and sinNeRF [35] adopted the consistency of semantic information to constrain the optimization process of NeRF. However, sfm struggles to function with large viewpoint gaps, and obtaining additional semantic supervision is often difficult. To address this, InfoNeRF [13] minimized ray entropy for adjacent rays via KL-divergence to avoid introducing other information sources. Additionally, some studies focused on adding unseen-viewpoints-constraints to endow NeRF with prior information using geometry-smooth [20] or image-warping [35] techniques. RegNeRF [20] proposed to regularize the color predictions at unseen viewpoints by a pre-trained flow model and demonstrated that mip-NeRF [1] produced more satisfactory results than vanilla NeRF under sparse view settings.

Similar to NeRF-S, other works have focused on exploring the relationship between the target viewpoints and few reference viewpoints [3, 5, 15, 32, 37], called generalizable NeRF. Despite achieving

photorealistic renderings and fast generalization capabilities [31], the assumption that the reference views and target view are adjacent-viewpoints is overly idealized and often requires pre-training before fine-tuning. In this work, we primarily focus on rendering a complete scene under sparse inputs.

In light of the above discussion, we utilize mip-NeRF as the baseline (see Sec. 3 for details) in this work. Instead of constraining results after volume rendering, our approach delves into the root cause of NeRF's confusion from sparse inputs and employs a sampling point self-alignment strategy along with a semi-supervised paradigm to overcome this challenge during volume rendering. Additionally, we follow the experimental settings of InfoNeRF [13], rendering a complete scene without pre-training model and mainstream image-warping techniques that are time-consuming during the training phase.

3 METHOD

3.1 Background

3.1.1 NeRF. Neural Radiance Fields (NeRF) map the 3D location $\mathbf{x} \in \mathbb{R}^3$ and viewing direction $\mathbf{d} \in \mathbb{S}^2$ of the spatial point sampled on the ray to the view-invariant volume density $\sigma \in [0, \infty)$ and view-dependent color $c \in [0, 1]^3$ of the point using learnable Multi-Layer Perceptrons (MLPs).

$$f_\theta \left(f_\psi \left(\psi \left(\mathbf{x} \right) \right), \kappa \left(\mathbf{d} \right) \right) = \left(\sigma, c \right), \tag{1}$$

where f_θ and f_ψ are MLPs, the learnable mapping functions, and ψ, κ are the positional encoding for $\mathbf{x}, \mathbf{d}$, respectively.

3.1.2 mip-NeRF. Unlike NeRF's ray sampling, mip-NeRF [1] introduces conical frustum sampling to enhance the receptive field of NeRF. Integrated positional encoding is adopted to replace ψ, κ
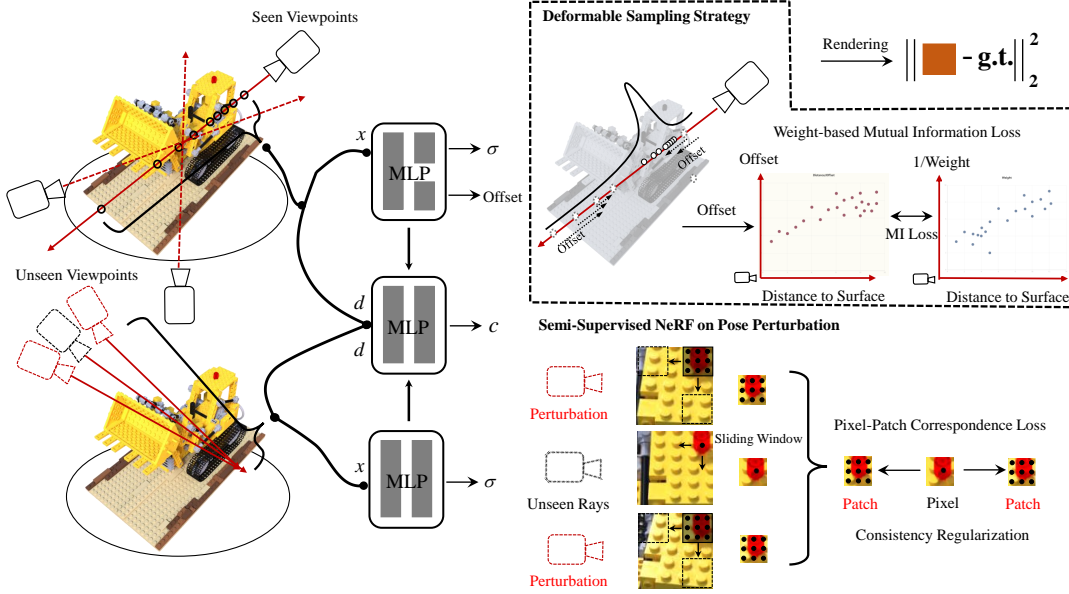
Figure 3: *Overall architecture of WaH-NeRF*. Firstly, for seen viewpoints, we propose a Deformable Sampling strategy and compute an offset for each sampling conical frustum, supervised by a Weight-based Mutual Information Loss. In addition, we adopt the MSE Loss with the ground truth to further constrain the training process. For unseen viewpoints, after perturbing the camera pose of these viewpoints, we use the Pixel-Patch Correspondence Loss to regulate consistency with the unseen ray for semi-supervised learning. Finally, the two modules are integrated together to mitigate the confusion of "Where" and "How".

in conical frustums with a multivariate Gaussian. In mip-NeRF, the context of the points is taken into account during the volume rendering for each pixel, compensating for the instability when providing sparse viewpoints. Therefore, mip-NeRF serves as the baseline in our work.

3.1.3 Volume Rendering. Once the c_i and σ_i values for each point on the rays or cones are obtained using Eq. (1), the pixel color value $\hat{C}$ of the rendered image is approximated by an integral, which is replaced by the quadrature along the rays or cones.

$$\hat{C}(r) = \sum_{i=1}^{M} \left(\sum_{j=1}^{i-1} \sigma_j \delta_j \right) (1 - e^{-\sigma_i \delta_i}) c_i, \tag{2}$$

where M is the number of sampling points, δ_i states the distance between the ith point and its adjacent point, and r represents a ray or cone from an origin.

$$Loss_{MSE} = \sum_{r \in R} \left\| \hat{C}(r) - C_{GT}(r) \right\|_2^2. \tag{3}$$

Eq. (3) provides a Mean Square Error (MSE) loss to supervise the entire rendering process, where R is a set containing all rays or cones and $C_{GT}(r)$ represents the ground truth color value for $\hat{C}(r)$. It is worth noting that NeRF uses a combination of coarse-sampling and fine-resampling for quadrature with non-sharing parameters, whereas mip-NeRF shares the training parameters in two stages.

3.2 Overview

In the following, we introduce WaH-NeRF, a novel training strategy aimed at mitigating the confusion discussed in Sec. 1 and analyzed

in Sec. 4. To explore the context of the cone, a patch-based strategy is used throughout the training process for both seen and unseen viewpoint-inputs. We then design a Deformable Sampling strategy (in Sec. 3.3) and Semi-Supervised NeRF based on Pose perturbation (in Sec. 3.4) to address the confusion of "WHERE to sample?" and "HOW to predict?" in NeRF-S, as shown in Fig. 3. To accurately supervise these two modules, we adopt a Weight-based Mutual Information Loss and a Pixel-Patch Correspondence Loss, which form two parallel branches during training.

3.3 Deformable Sampling Strategy

Recall that NeRF [19] introduces a coarse-to-fine hierarchical sampling strategy, which achieves satisfactory results for rendering. However, this strategy is flawed when handling sparse inputs for the following reasons: 1) the difference between the training and testing viewpoints leads to overfitting with a fixed sampling strategy, and 2) the uncertainty of the surface resulting from coarse sampling makes accurate fine-resampling under sparse input challenging. In general, this sampling drift creates confusion about **"Where to sample?"**, turning the task of predicting target surface and color into an ill-posed problem for NeRF-S.

Deformable Sampling strategy. To determine **"Where to sample?"**, we propose a Deformable Sampling strategy inspired by deformable convolution [6]. Instead of using a fixed sampling aggregation strategy [2], our method predicts a position-dependent and view-invariant offset for each conical frustum to adaptively aggregate sampling-points to the target surface. Based on this offset, we deform the position of the sampling conical frustum along

the sampling cone before volume rendering, encouraging more frustums to approach the accurate surface. Specifically, we predict the adaptive offset using an additional MLP layer with intermediate features $F_\psi(\mathbf{x}) \in \mathbb{R}^{M \times d}$ from f_ψ in Eq. (1), where d represents feature dimension:

$$Offset = \omega_0 \times F_\psi(\mathbf{x}) + b_0, \tag{4}$$

where $Offset \in \mathbb{R}^{M \times 1}$, and ω_0 is the weight and b_0 is the bias of MLP layer. In order to prevent rendering collapse during training, both ω_0 and b_0 are initialized to 10^{-7}. Subsequently, the original sampled conical frustum shifts by the predicted $Offset$ along the cone for adaptive sampling, which is utilized for volume rendering as in Eq. (2). It should be noted that the Deformable Sampling strategy not applied to unseen perspectives to ensure training stability.

In summary, Deformable Sampling strategy can be regarded as a more general sampling strategy, and vanilla NeRF [19] is a special case where $Offset$ is fixed to 0.

Weight-based Mutual Information Loss. However, training ω_0 and b_0 from scratch is sub-optimal in practice since low-weight sample conical frustums struggle to offset towards the surface. To address this, we explore the correlation between the predicted $Offset$ and $weight$ (w) and use it as prior knowledge for additional supervision. Here, w approximates the probability of the surface in NeRF [19], as shown in Eq. (5). Intuitively, when the sampled conical frustums are far from the target surface, w decreases and the $Offset$ increases. Therefore, the predicted $Offset$ has a negative correlation with the non-zero part of w.

$$w_i = \left(\sum_{j=1}^{i-1} \sigma_j \delta_j \right) (1 - e^{-\sigma_i \delta_i}). \tag{5}$$

Based on this analysis, we design a Weight-based Mutual Information Loss, a natural information-theoretic measure of the variables independence. Specifically, the mutual information of $Offset$ and $\frac{1}{w}$ is expressed as

$$I\left(\frac{1}{w+\varepsilon}; Offset\right) = H\left(\frac{1}{w+\varepsilon}\right) - H\left(\frac{1}{w+\varepsilon} | Offset\right)$$
$$= H(Offset) - H\left(Offset | \frac{1}{w+\varepsilon}\right), \tag{6}$$

where ε is set to 5×10^{-4}, and H denotes entropy. The correlation between $Offset$ and $\frac{1}{w}$ increases through minimizing the Weight-based Mutual Information Loss. Inspired by [39], we further express $Loss_{MI}$ by taking into account the correlation between sampling conical frustums,

$$Loss_{MI} = - \sum_{r \in R} I\left(\frac{1}{w+\varepsilon}; O\right)_{mask}, \tag{7}$$

where $mask$ is used to remove positions where w is close to zero. The masked positions are ignored for $Loss_{MI}$, as the offset is meaningless for the positions too far from the surface. In addition, we introduce a distortion loss $Loss_{dist}$ [2] to further constrain the offset conical frustums approaching target surface. In summary, L_{Offset} be expressed as

$$Loss_{Offset} = \lambda Loss_{MI} + Loss_{dist}, \tag{8}$$

where λ is a hyperparameter for weight of $Loss_{MI}$, to 5×10^{-3}.
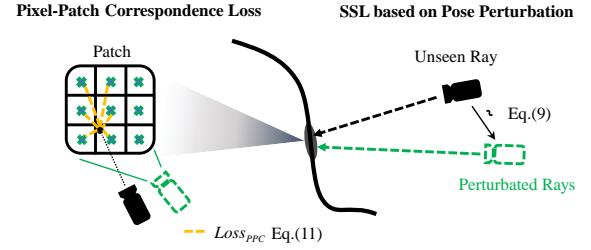
Figure 4: *Semi-Supervised NeRF based on pose perturbation.* We perturb the camera position to augment limited data, and employ a Pixel-Patch Correspondence Loss as a local consistency regularization to supervise unseen ray.

3.4 Semi-Supervised NeRF on Pose Perturbation

Addressing the confusion of "WHERE to sample?" in NeRF-S may be helpful, but accurate sampling predictions remain a challenge under sparse views due to overfitting and insufficient information. Therefore, it is crucial to address the confusion of **"HOW to predict?"** in NeRF-S. Intuitively, introducing additional supervision, as proposed in [35] and [20], could alleviate this confusion.

In image processing, when labeled data is scarce and a vast amount of unlabeled data is available, the semi-supervised paradigm [30, 41] can improve performance by effectively utilizing unlabeled data for supervision. Inspired by semi-supervised learning (SSL), researchers have treated seen viewpoints with accurate camera poses as labeled data, while considering unseen viewpoints as unlabeled data. By generating pseudo-label images of the unseen perspectives, they provide additional supervision for NeRF. However, the existing method for generating these pseudo-labels, which relies on image warping [35], is time-consuming and limited to small viewing changes.

Semi-Supervised NeRF. We propose a Semi-Supervised NeRF Paradigm based on pose perturbation to assist prediction without image warping, as shown in Fig. 4. Thanks to NeRF's implicit representation, it can seamlessly generate continuous views. Building on this advantage, our semi-supervised paradigm employs a straightforward scheme that uses consistency regularization to explore the smoothness between the unseen ray and perturbed rays.

Pose Perturbation. The smooth perturbation of the NeRF for unseen rays is a non-trivial discussion. Aug-NeRF [4] adopted perturbations into three distinct levels, including input coordinate, intermediate feature and pre-rendering output for data augmentations. However, for the semi-supervised paradigm, such local perturbations fail to reflect the smoothness of the rendering process. Instead, we propose a global perturbation method by slightly moving or rotating the camera pose and leveraging the local correlation between adjacent poses to constrain the network training. Due to the minor perturbation, there is theoretical consistency between the original unseen ray and the perturbed unseen rays. The consistency regularization not only ensures the network's robustness against discrepancies between seen and unseen viewpoints, but also improves the local smoothness of the adjacent rendered rays.

Specifically, unseen viewpoints are randomly captured on a fixed-radius sphere oriented towards the spherical render target. It is well known that any position on an arbitrary sphere can be determined by three variables: the radius F controlling the sphere's size, and the azimuth angle φ and polar angle Γ defining the position on the sphere. Based on this prior knowledge, three levels of perturbation are performed on the camera pose, F, φ and Γ, which prove to be significant in our ablation experiments. We define random perturbation as

$$\hat{F} = F + \tau_F; \hat{\varphi} = \varphi + \tau_\varphi; \hat{\Gamma} = \Gamma + \tau_\Gamma, \tag{9}$$

where τ_F, τ_φ, and τ_Γ are perturbation values. Based on the unseen-perturbed viewpoints pair, we sample the corresponding unseen-perturbed rays pair. Subsequently, the perturbed rays will be input to the NeRF-network simultaneously with the corresponding original unseen ray.

Pixel-Patch Correspondence Loss. A straightforward idea for supervision in our semi-supervised paradigm is to constrain the consistency of the rendered results between unseen ray and perturbation rays at the pixel level,

$$Loss_{SSL} = \sum_{p=0}^{P} \sum_{r \in R} \left\| \hat{C}_{unseen}(r) - \hat{C}^p_{perturbation}(r) \right\|_2^2, \tag{10}$$

where P indicates the amount of perturbation for an unseen ray. Despite the global similarity of the rendered images, vanilla MSE loss struggles to achieve pixel-wise (local) correspondence for consistency regularization, due to the mismatches caused by minor camera pose perturbations.

To explore local consistency, the Pixel-Patch Correspondence Loss (PPC) is introduced to alleviate unregistered pixel-wise correspondences from perturbation and enhance patch-wise smoothness. Thanks to the patch-wise training paradigm, Pixel-Patch Correspondence loss is easily accessible. We employ the average MSE loss between each unseen ray and all rays of the corresponding patch around the perturbed rays,

$$Loss_{PPC} = \sum_{p=0}^{P} \sum_{r \in R} \sum_{i=0}^{PS \times PS} \left\| \hat{C}_{unseen}(r) - \right.$$
$$\left. \hat{C}^p_{perturbation(i)}(r) \right\|_2^2 \times \frac{1}{PS \times PS}, \tag{11}$$

where $\hat{C}^p_{perturbation(i)}(r)$ is the ith pixel (ray) in the perturbed patch corresponding r, and PS is the patch size. It is worth noting that the consistency regularization not only applies to color value predictions, but also to depth predictions. Therefore, the overall Pixel-Patch Correspondence Loss is defined as the sum of the RGB and depth losses, i.e., $Loss_{PPC} = Loss_{PPC_{RGB}} + Loss_{PPC_D}$. This is demonstrated in the Appendix ablation experiment section.

3.5 Loss Function

Given a set of N seen viewpoints and their associated images, our work combines the Deformable Sampling strategy with the Semi-Supervised NeRF Paradigm to iteratively optimize the learnable parameters θ by evaluating the rendered images and calculating the loss function.

$$Loss = Loss_{MSE} + \mu Loss_{Offset} + \nu Loss_{PPC} + Loss_{smooth}. \tag{12}$$
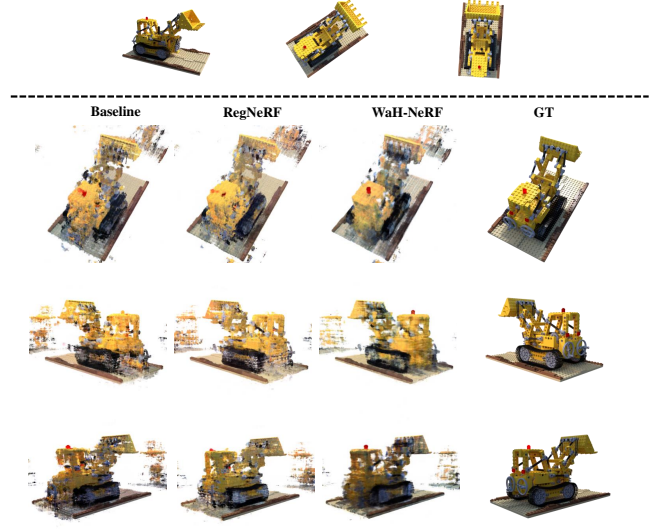
Figure 5: *Qualitative results from lego scene.* **We choose some testing viewpoints that are significantly different from the training viewpoints under our setting.**

As in [20], $Loss_{smooth}$ is introduced to improve the smoothness of depth predictions for unseen viewpoints, which has proven effective in NeRF-S. Hyper-parameters μ and ν are used to weight the loss terms. We define $\mu = 0.5$ and $\nu = 0.5$, whose superiority is proved in the Appendix. In practice, each item in Eq. (12) includes two stages of loss: coarse-sampling loss and fine-resampling loss. During training, we set the coefficient of the coarse-sampling loss to 0.1.

4 EXPERIMENTS

4.1 Datasets

In order to verify the superiority of WaH-NeRF and the effectiveness of each module, we conduct comparative and ablation experiments on the Realistic Synthetic 360° [19] and LLFF [19] datasets.
Realistic Synthetic 360° is a classic dataset of NeRF, which is widely used in NeRF and its derivative works. It contains 8 synthetic objects with viewpoints sampled on the upper hemisphere and coverages 360°. Each scene has 100 views for training and 200 views for testing, with all images at 800×800 resolution. In our task setting, due to the wide range of viewpoints, we use this dataset as the main benchmark in our work.
LLFF consists of 8 complex scenes, each containing 20-62 images with 1008×756 resolution. Compared with Realistic Synthetic 360°, the real-world dataset LLFF is more complex in real-world but the viewpoints are relatively close to each other.

4.2 Implementation and Evaluation

Implementation details. We implement our code on top of the Pytorch [21] mip-NeRF [1]. We optimize using Adam [14] with an

---

[1]https://github.com/bebeal/mip-NeRF-pytorch

Table 1: Mean PSNR and standard deviations of every scene on the Realistic Synthetic 360°dataset under three inputs.

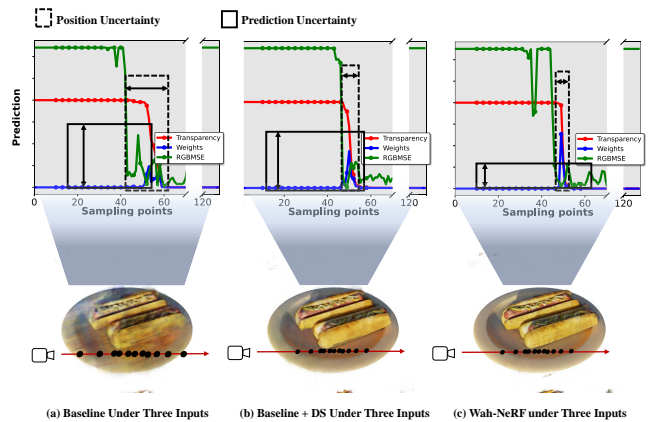| Method	Lego	Chair	Drums	Ficus	Hotdog	Materials	Mic	Ship
NeRF, 100views	32.54	33.00	25.01	30.13	36.18	29.62	32.91	28.65
NeRF [19], 3views	$8.63_{\pm1.21}$	$8.45_{\pm0.33}$	$7.28_{\pm0.51}$	$10.23_{\pm1.73}$	$7.89_{\pm0.42}$	$9.27_{\pm2.10}$	$11.52_{\pm1.98}$	$6.58_{\pm0.75}$
mip-NeRF [1], 3views	$17.23_{\pm2.84}$	$17.31_{\pm1.94}$	$14.33_{\pm0.85}$	$18.47_{\pm2.28}$	$17.35_{\pm1.64}$	$15.51_{\pm2.83}$	$18.02_{\pm2.17}$	$15.33_{\pm3.31}$
IBRNet [32] (CVPR 2021)	$12.60_{\pm1.53}$	$12.33_{\pm0.94}$	$9.89_{\pm1.22}$	$15.09_{\pm3.21}$	$11.18_{\pm2.48}$	$10.47_{\pm3.05}$	$14.77_{\pm1.58}$	$13.24_{\pm3.09}$
pixelNeRF [37] (CVPR 2021)	$13.22_{\pm3.17}$	$11.26_{\pm2.75}$	$11.44_{\pm1.65}$	$15.59_{\pm2.99}$	$12.41_{\pm3.88}$	$10.32_{\pm4.36}$	$15.00_{\pm2.86}$	$13.15_{\pm3.26}$
DietNeRF [11] (ICCV 2021)	$14.98_{\pm2.81}$	$15.11_{\pm2.25}$	$12.43_{\pm0.97}$	$16.39_{\pm3.10}$	$15.21_{\pm3.19}$	$13.48_{\pm2.64}$	$14.59_{\pm2.81}$	$14.41_{\pm3.22}$
InfoNeRF [13] (CVPR 2022)	$16.59_{\pm2.45}$	$16.88_{\pm2.04}$	$12.16_{\pm0.86}$	$18.20_{\pm2.04}$	$17.42_{\pm3.22}$	$14.77_{\pm2.98}$	$16.73_{\pm2.94}$	$14.99_{\pm3.05}$
RegNeRF [20] (CVPR 2022)	$17.82_{\pm2.09}$	$17.96_{\pm1.77}$	$14.20_{\pm0.91}$	$19.01_{\pm1.74}$	$17.98_{\pm3.69}$	$15.35_{\pm3.21}$	$18.11_{\pm2.03}$	$15.82_{\pm2.79}$
WaH-NeRF (ours)	$\mathbf{18.78}_{\pm1.98}$	$\mathbf{18.48}_{\pm1.96}$	$\mathbf{15.39}_{\pm0.96}$	$\mathbf{19.52}_{\pm1.70}$	$\mathbf{19.01}_{\pm3.23}$	$\mathbf{16.84}_{\pm2.45}$	$\mathbf{18.95}_{\pm1.75}$	$\mathbf{16.20}_{\pm2.79}$
WaH-NeRF† (ours)	$20.27_{\pm1.90}$	$20.53_{\pm1.69}$	$16.77_{\pm0.87}$	$21.83_{\pm1.40}$	$22.71_{\pm2.66}$	$18.92_{\pm2.11}$	$20.18_{\pm1.68}$	$17.91_{\pm2.63}$

Table 2: Mean PSNR, SSIM and LPIPS in all scenes for quantitative evaluations and component analysis for Realistic Synthetic 360° benchmark under three inputs. DS and SSN represent Deformable Sampling strategy and Semi-Supervised NeRF Paradigm, respectively.

Method	DS	SSN	PSNR ↑	SSIM ↑	LPIPS ↓
NeRF,100views			31.01	0.947	0.081
NeRF,3views			8.73	0.445	0.580
mip-NeRF,3views			16.69	0.708	0.283
IBRNet			12.44	0.542	0.403
pixelNeRF			12.80	0.587	0.328
DietNeRF			14.58	0.608	0.317
InfoNeRF			15.97	0.647	0.291
RegNeRF			17.03	0.720	0.278
baseline			16.72	0.711	0.290
WaH-NeRF	✓		17.51	0.768	0.246
WaH-NeRF		✓	17.69	0.773	0.237
WaH-NeRF (ours)	✓	✓	**17.90**	**0.776**	**0.230**
WaH-NeRF† (ours)	✓	✓	**19.89**	**0.809**	**0.177**

(a) Baseline Under Three Inputs (b) Baseline + DS Under Three Inputs (c) Wah-NeRF under Three Inputs

Figure 6: Qualitatively demonstrate that the proposed module can effectively alleviate uncertainty of surface position and prediction at hotdog scene.

exponential learning rate decay from 10^{-3} to 5×10^{-5}, decaying exponentially by a factor of 10 at every 2500 iterations. For the Realistic Synthetic 360°, we train for a total of 20,000 iterations and save the model every 5,000 iterations. We set the patch size of the seen and unseen viewpoints to be 8x8, and the batch size to 16, which means that both input 1024 rays, $|R| = 1024$. Following the experimental setup of InfoNeRF [13], under $N = 3$ setting, we randomly select three viewpoints for training, and all comparison experiments use the same three viewpoints to ensure fairness. Our experiments are conducted with a GeForce RTX 3090 GPU.

Metrics. We employ the standard image quality metrics, including Peak Signal to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) [33], to evaluate rendering quality for novel viewpoints. Additionally, we introduce learned perceptual image patch similarity (LPIPS) [38] as a perceptual metric. For all metrics, we calculate the mean and standard deviation for comparative experiments.

Baseline. We use the Pytorch reimplementation of mip-NeRF [1] as the baseline network, which has been shown to outperform vanilla NeRF in our experiments. We further integrate a patch-based training strategy into the mip-NeRF as the baseline. In the comparative experiments, we evaluate our method against vanilla NeRF [19]

and mip-NeRF [1] under sparse inputs, and the state-of-the-art models under our setting, including InfoNeRF [13], DietNeRF [11], DSNeRF [9][2] and RegNeRF [20]. Additionally, we compare our approach against some recent works, namely IBRNet [32] and pixelNeRF [37] on the adjacent-viewpoints setting, which has been shown to outperform traditional NVS methods. For a detailed description about these, see Sec. 2 and Appendix.

4.3 Experiment and Analysis

Quantitative evaluations results. Table 1 and Table 2 present the performance comparison between our proposed WaH-NeRF and the leading methods on the Realistic Synthetic 360°dataset. It can be observed that WaH-NeRF achieves state-of-the-art performance under three inputs ($N = 3$) for each scene. Specifically, summarizing from Table 2, WaH-NeRF surpasses the previous state-of-the-art method (i.e., RegNeRF) by **0.87** (PSNR), **0.046** (SSIM) and **0.048** (LPIPS). Additionally, compared with the vanilla NeRF and mip-NeRF, our WaH-NeRF achieves significant improvement, i.e.,**1.21** (PSNR), **0.068** (SSIM) and **0.053** (LPIPS) for mip-NeRF under our

---

[2]Due to the use of sfm and COLMAP, DSNeRF is only used for comparative experiments on the LLFF.

Table 3: Mean PSNRs, SSIM and LPIPS in all scenes for loss function ablation experiment on Realistic Synthetic 360° benchmark from three inputs.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
w/o L_{PPC}	17.73	0.773	0.239
w/o L_{Offset}	17.30	0.742	0.265
WaHNeRF (ours)	**17.90**	**0.776**	**0.230**

settings. Moreover, we also conduct experiments under different numbers of inputs in the following section. These results demonstrate the effectiveness and superiority of our proposed WaH-NeRF over the existing state-of-the-art methods. Here, WaH-NeRF† represents the WaH-NeRF's rendering performance after masking the background. This shows that WaH-NeRF can achieve satisfactory performance for the foreground object with only three inputs.

Qualitative evaluation results. Fig. 5 reports qualitative results rendered by WaH-NeRF on the lego scene, providing additional evidence of its effectiveness. Intuitively, compared with the state-of-the-art RegNeRF [20] and the baseline, WaH-NeRF shows a clear advantage in mitigating confusion and achieving better rendering results for surface and color. More quantitative results in other scenes can be found in the Appendix.

Component analysis. Our proposed WaH-NeRF consists of two key components: the Deformable Sampling strategy and the Semi-Supervised NeRF on pose perturbation. We validate the effectiveness of each component and present the results in Table 2. It can be summarized that the Semi-Supervised NeRF Paradigm plays the important role in the rendering performance improvement while the Deformable Sampling strategy is indispensable. By combining both Semi-Supervised NeRF Paradigm and Deformable Sampling strategy, WaH-NeRF achieves a substantial improvement of **1.18** (PSNR), **0.065** (SSIM), and **0.060** (LPIPS) over the baseline model.

To further explain the benefits of the proposed components during volume rendering, as shown in Fig. 6, we qualitatively demonstrate how the Deformable Sampling strategy and the Semi-Supervised NeRF Paradigm help to alleviate the confusion of "WHERE to sample?" and "HOW to predict?" in NeRF-S. Additional analyses and quantitative results are available in the Appendix.

Loss functions analysis. L_{Offset} and L_{PPC} are important components in our loss function. In Table 3, the significant contribution of both losses to the network is evident, with L_{Offset} being particularly noteworthy. This could be attributed to the difficulty of training the $Offset$ from scratch without the aid of L_{Offset}.

Quantitative and qualitative results for LLFF. Table 4 shows the quantitative comparison between the previous state-of-the-art methods and WaH-NeRF on the LLFF dataset, where FT representing Fine-Tuning on per scene [20]. Despite the LLFF dataset being more challenging due to its larger scene size and the presence of only nearby viewpoints, WaH-NeRF also attains state-of-the-art results. This demonstrates the superiority of method regardless of the task scenario. Fig. 7 showcase some qualitative results rendered by WaH-NeRF on the room scene, providing further evidence of its effectiveness. Additional quantitative results for other LLFF scenes are available in the Appendix.
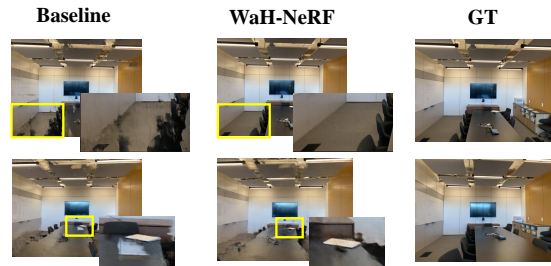
Figure 7: Some qualitative results of the baseline and WaH-NeRF under LLFF dataset (room scene) in the 3-Inputs setting.

Table 4: Mean PSNRs, SSIM and LPIPS in all scenes for quantitative evaluations for LLFF benchmark.

Method	N	Setting	PSNR ↑	SSIM ↑	LPIPS ↓
NeRF	100		26.50	0.811	0.250
pixelNeRF	3	Trained on	16.17	0.438	0.512
MVSNeRF	3	DTU and FT	17.88	0.584	0.327
mip-NeRF	3		14.62	0.351	0.495
DietNeRF	3		14.94	0.370	0.496
InfoNeRF	3	Optimized	14.37	0.349	0.457
DSNeRF	3	per Scene	18.51	0.558	0.338
RegNeRF	3		19.08	0.587	0.336
WaH-NeRF	3		**19.23**	**0.587**	**0.312**

5 CONCLUSION

In this paper, we propose a novel training paradigm for Neural Radiance Fields from sparse inputs, termd as WaH-NeRF. We observed that NeRF's performance significantly declines as the number of input views decreases. Our analysis attribute this phenomenon to the confusion of "WHERE to sample?" and "HOW to predict?" in NeRF-S. To address this confusion, we introduce a Deformable Sampling strategy and a Semi-Supervised NeRF learning Paradigm based on pose perturbation, leveraging the Pixel-Patch Correspondence Loss and the Wight-based Mutual Information Loss in the training process. Extensive experiments on two NeRF benchmarks demonstrate that our proposed WaH-NeRF achieves state-of-the-art performance under sparse input settings, as supported by both quantitative and qualitative results. We believe that the confusion analysis for NeRF-S and the semi-supervised NeRF based on consistency regularization will inspire future research in this field.

REFERENCES

[1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. 2021. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5855–5864.

[2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. 2022. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5470–5479.

[3] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. 2021. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14124–14133.

[4] Tianlong Chen, Peihao Wang, Zhiwen Fan, and Zhangyang Wang. 2022. Aug-NeRF: Training Stronger Neural Radiance Fields With Triple-Level Physically-Grounded Augmentations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15191–15202.

[5] Julian Chibane, Aayush Bansal, Verica Lazova, and Gerard Pons-Moll. 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.

[6] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 764–773.

[7] Paul Debevec, Yizhou Yu, and George Borshukov. 1998. Efficient view-dependent image-based rendering with projective texture-mapping. In *Eurographics Workshop on Rendering Techniques*. Springer, 105–116.

[8] Paul E Debevec, Camillo J Taylor, and Jitendra Malik. 1996. Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 11–20.

[9] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. 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.

[10] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. 2019. Deepview: View synthesis with learned gradient descent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2367–2376.

[11] Ajay Jain, Matthew Tancik, and Pieter Abbeel. 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.

[12] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. 2016. Learning-based view synthesis for light field cameras. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 1–10.

[13] Mijeong Kim, Seonguk Seo, and Bohyung Han. 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.

[14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[15] Jonáš Kulhánek, Erik Derner, Torsten Sattler, and Robert Babuška. 2022. ViewFormer: NeRF-free Neural Rendering from Few Images Using Transformers. *arXiv preprint arXiv:2203.10157* (2022).

[16] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. 2019. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751* (2019).

[17] Ricardo Martin-Brualla, Rohit Pandey, Shuoran Yang, Pavel Pidlypenskyi, Jonathan Taylor, Julien Valentin, Sameh Khamis, Philip Davidson, Anastasia Tkach, Peter Lincoln, et al. 2018. Lookingood: Enhancing performance capture with real-time neural re-rendering. *arXiv preprint arXiv:1811.05029* (2018).

[18] Moustafa Meshry, Dan B Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. 2019. Neural rerendering in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6878–6887.

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

[20] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. 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.

[21] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).

[22] Eric Penner and Li Zhang. 2017. Soft 3D reconstruction for view synthesis. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 1–11.

[23] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. 2022. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12892–12901.

[24] Antoni Rosinol, John J Leonard, and Luca Carlone. 2022. NeRF-SLAM: Real-Time Dense Monocular SLAM with Neural Radiance Fields. *arXiv preprint arXiv:2210.13641* (2022).

[25] Mehdi SM Sajjadi, Henning Meyer, Etienne Pot, Urs Bergmann, Klaus Greff, Noha Radwan, Suhani Vora, Mario Lučić, Daniel Duckworth, Alexey Dosovitskiy, et al. 2022. Scene representation transformer: Geometry-free novel view synthesis through set-latent scene representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6229–6238.

[26] Shao-Hua Sun, Minyoung Huh, Yuan-Hong Liao, Ning Zhang, and Joseph J Lim. 2018. Multi-view to novel view: Synthesizing novel views with self-learned confidence. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 155–171.

[27] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. 2022. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8248–8258.

[28] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. 2016. Multi-view 3d models from single images with a convolutional network. In *European Conference on Computer Vision*. Springer, 322–337.

[29] Richard Tucker and Noah Snavely. 2020. Single-view view synthesis with multi-plane images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 551–560.

[30] Jesper E Van Engelen and Holger H Hoos. 2020. A survey on semi-supervised learning. *Machine Learning* 109, 2 (2020), 373–440.

[31] Peihao Wang, Xuxi Chen, Tianlong Chen, Subhashini Venugopalan, Zhangyang Wang, et al. 2022. Is Attention All NeRF Needs? *arXiv preprint arXiv:2207.13298* (2022).

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

[33] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.

[34] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. 2020. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7467–7477.

[35] Dejia Xu, Yifan Jiang, Peihao Wang, Zhiwen Fan, Humphrey Shi, and Zhangyang Wang. 2022. SinNeRF: Training Neural Radiance Fields on Complex Scenes from a Single Image. *arXiv preprint arXiv:2204.00928* (2022).

[36] Jae Shin Yoon, Kihwan Kim, Orazio Gallo, Hyun Soo Park, and Jan Kautz. 2020. Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5336–5345.

[37] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 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.

[38] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 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.

[39] Shuai Zhao, Yang Wang, Zheng Yang, and Deng Cai. 2019. Region mutual information loss for semantic segmentation. *Advances in Neural Information Processing Systems* 32 (2019).

[40] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. 2018. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817* (2018).

[41] Xiaojin Jerry Zhu. 2005. Semi-supervised learning literature survey. (2005).

[42] C Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. 2004. High-quality video view interpolation using a layered representation. *ACM transactions on graphics (TOG)* 23, 3 (2004), 600–608.