Self-Evolving Neural Radiance Fields

Jaewoo Jung*, Jisang Han*, Jiwon Kang*, Seongchan Kim MinSeop Kwak Seungryong Kim†

KAIST AI

https://cvlab-kaist.github.io/SE-NeRF/

Abstract

*Recently, neural radiance field (NeRF) has shown remarkable performance in novel view synthesis and 3D reconstruction. However, it still requires abundant high-quality images as input, limiting its applicability in real-world scenarios. To overcome this limitation, recent works have focused on training NeRF only with sparse viewpoints by giving additional regularizations. However, due to the under-constrained nature of the task, solely using additional regularization is not enough to prevent the model from overfitting to sparse viewpoints. In this paper, we propose a novel framework, dubbed self-evolving neural radiance fields (**SE-NeRF**), that applies a self-training paradigm to NeRF to address these problems. We formulate few-shot NeRF into a teacher-student framework to guide the network to learn a more robust representation of the scene by training the student with additional pseudo labels generated from the teacher. By distilling ray-level pseudo labels using distinct distillation schemes for reliable and unreliable rays obtained with our novel reliability estimation method, we enable NeRF to learn a more accurate and robust geometry of the 3D scene. We show that applying our self-training framework to existing NeRF models improves the quality of the rendered images and achieves state-of-the-art performance.*

1. Introduction

Novel view synthesis that aims to generate novel views of a 3D scene from given images is one of the essential tasks in computer vision fields. Recently, neural radiance field (NeRF) [21] has shown remarkable performance for this task, modeling highly detailed 3D geometry and specular effects solely from given image information. However, the requirement of abundant high-quality images with accurate poses restricts its application to real-world scenarios, as reducing the input views causes NeRF to undergo severe performance degradation.

Numerous works [14, 16, 23, 36, 42] tried to address this problem, known as few-shot NeRF, whose aim is to robustly optimize NeRF in scenarios where only a few and sparse input images are given. To compensate for the few-shot NeRF's under-constrained nature, they either utilize the prior knowledge of a pre-trained model [14, 42] such as CLIP [27] or 2D CNN [42] or introduce an additional regularization [16, 17, 23], showing compelling results. However, these works show limited success in addressing the fundamental issue of overfitting as NeRF tends to memorize the input known viewpoints instead of understanding the geometry of the scene. In our toy experiment, this behavior is clearly shown in Figure 1, where existing methods (even with regularization [12, 16, 23]) trained with 3-views show a noticeable drop in PSNR even with slight changes of viewpoints.

Unlike these methods, we propose a novel framework that exploits additional ground truth data for viewpoints that were unknown to the few-shot setting. We assume that if we can accurately identify reliable regions in the rendered images at unknown views, the rendered regions can be utilized as additional data achieved with no extra cost. As a pilot study, we compare the rendered images from few-shot NeRF with the ground truth images and verify that there are accurately modeled regions even in *unknown* viewpoints that are *far* from known ones as exemplified in Figure 1. Based on these observations, we formulate the few-shot NeRF task into the self-training framework by considering the rendered images as pseudo labels and training a new NeRF network with confident pseudo labels as additional data.

Expanding upon this idea, we introduce a novel framework, dubbed self-evolving neural radiance fields (**SE-NeRF**), which enables a more robust training of few-shot NeRF in a self-supervised manner. We train the few-shot NeRF under an iterative teacher-student framework, in which pseudo labels for geometry and appearance generated by the teacher NeRF are distilled to the student NeRF,

*These authors contributed equally.

†Corresponding author.

and the trained student serves as the teacher network in the next iteration for progressive improvement. To estimate the reliability of the pseudo labels, we utilize the semantic features of pre-trained convolutional networks to measure the consistency of the pseudo labels within multiple viewpoints. We also apply distinct distillation schemes for reliable and unreliable rays, in which reliable ray labels are directly distilled to the student, while unreliable rays undergo a regularization process to distill more robust geometry.

Our experimental results show that our framework successfully guides NeRF to learn a more robust geometry of the scene in the few-shot NeRF setting without using any external 3D priors or generative models [40]. Also, we show the versatility of our framework, which can be applied to all existing models without changing their structure. We evaluate our approach on synthetic and real-life datasets, achieving state-of-the-art results in multiple settings.

2. Related Work

Neural radiance fields (NeRF). Synthesizing images from novel views of a 3D scene given multi-view images is a long-standing goal of computer vision. Recently, neural radiance fields (NeRF) [21] has achieved great success by optimizing a single MLP that learns to estimate the radiance of the queried coordinates. The MLP learns the density $\sigma \in \mathbb{R}$ and color $\mathbf{c} \in \mathbb{R}^3$ of continuous coordinates $\mathbf{x} \in \mathbb{R}^3$, and is further utilized to explicitly render the volume of the scene using ray marching [15]. Due to it's impressive performance in modeling the 3D scene, various follow-ups [10, 12, 14, 16, 23, 28, 36, 41] adopted NeRF as their baseline model to solve various 3D tasks.

Few-shot NeRF. Although capable of successfully modeling 3D scenes, NeRF requires abundant high-quality images with accurate poses, making it hard to apply in real-world scenarios. Several methods have paved the way to address these issues by showing that NeRF can be successfully trained even when the input images are limited. One approach addresses the problem using prior knowledge from pre-trained local CNNs [9, 17, 42]. PixelNeRF [42], for instance, employs a NeRF conditioned with features extracted by a pre-trained encoder. Another line of research introduces a geometric or depth-based regularization [10, 14, 16, 23, 28, 36]. DietNeRF [14] proposes a semantic consistency loss using CLIP [27] to encourage realistic renderings at novel poses. RegNeRF [23] regularizes the geometry and appearance of patches from unobserved viewpoints. DS-NeRF [10] introduces additional depth supervision leveraging point clouds obtained from COLMAP [30].

Generalized NeRF. Generalized NeRFs attempt to solve the need for per-scene optimization by learning a universal prior knowledge that can be applied across diverse 3D scenes. These models [7, 37, 42] are trained on multiple scenes to capture common spatial and appearance patterns by training a feature predictor. This enables rapid adaptation to new environments with minimal additional input. In the context of training NeRF with sparse inputs, generalized NeRFs and few-shot NeRFs can be seen as orthogonal approaches each improving different aspects, which can be combined to further improve the overall performance. In this work, we follow existing few-shot NeRF works and focus on tackling sparse input optimization in a per-scene manner.

Self-training. Self-training is one of the earliest semi-supervised learning methods [11, 26, 31, 39] mainly used in settings where obtaining sufficient labels is expensive (e.g., Instance segmentation). Self-training exploits the *unlabeled* data by pseudo labeling with a teacher model, which is then *combined* with the labeled data and used in the student training process. We bring self-training to NeRFs by formulating the few-shot NeRF task as a semi-supervised learning task. Our approach can be seen as an analogous method of noisy student [39] that exploits NeRF as the teacher and student model, with teacher-generated *unknown* views as the *unlabeled* data.

Concurrent to our work is Self-NeRF [3], which introduces a new architecture combining Mip-NeRF's [4] cone tracing algorithm and NeRF-W's [19] uncertainty modeling architecture. It iteratively trains the network with rendered and warped images as additional data by guiding the network to learn reliable regions leveraging the uncertainty modeling architecture from Martin-Brualla et al. [19]. Differently, our uncertainty modeling block is designed to infer the reliable regions of the 3D scene regardless of the backbone architecture acting as a plug-and-play module. This makes our framework applicable to any existing NeRF model, enabling extra improvements to all existing works. In addition, we propose a novel knowledge distillation scheme tailored for NeRFs and show that our method is superior to them through quantitative comparison.

3. Preliminaries and Motivation

3.1. Preliminaries

Given a set of training images $\mathcal{S} = \{I_i \,|\, i \in \{1, \dots, N\}\}$, NeRF [21] represents the scene as a continuous function $f(\cdot; \theta)$, a neural network with parameters θ. The network renders images by querying the 3D points $\mathbf{x} \in \mathbb{R}^3$ and view direction $\mathbf{d} \in \mathbb{R}^2$ transformed by a positional encoding $\gamma(\cdot)$ to output a color value $\mathbf{c} \in \mathbb{R}^3$ and a density value $\sigma \in \mathbb{R}$ such that $\{\mathbf{c}, \sigma\} = f\left(\gamma(\mathbf{x}), \gamma(\mathbf{d}); \theta\right)$. The positional encoding transforms the inputs into Fourier features [33] that facilitate learning high-frequency details. Given a ray parameterized as $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$, starting from camera center $\mathbf{o}$ along the direction $\mathbf{d}$, the expected color value $C(\mathbf{r}; \theta)$

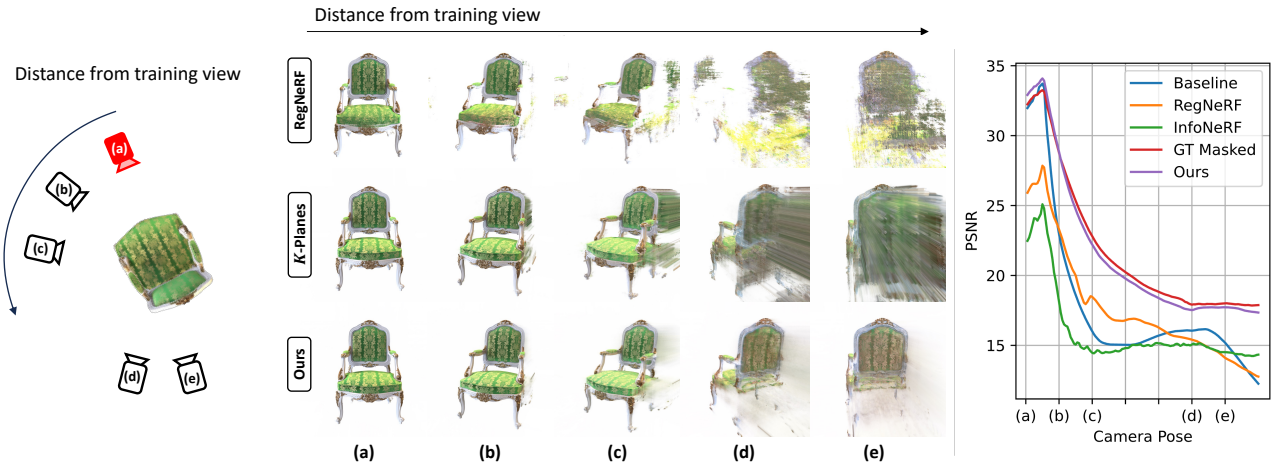


Figure 1. **Toy experiment to verify the robustness of models trained with sparse views. (Left)** The red camera (a) indicates the camera position used for training and cameras from (b-e) are used to verify the robustness of models when the novel viewpoint gets further from the known viewpoint. **(Middle)** For each viewpoint (a-e), we visualize the rendered images by RegNeRF [23], baseline (K-Planes [12]), and **SE-NeRF** (from top to bottom rows). **(Right)** Starting from viewpoint (a), we show the PSNR graph of the rendered images as the viewpoint moves gradually from (a-e). Existing models show extreme PSNR drops, even with slight movements.

along the ray $\mathbf{r}(t)$ from t_n to t_f is rendered as follows:

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

$$T(t) = \exp\left(-\int_{t_n}^{t}\sigma(\mathbf{r}(s);\theta)ds\right), \quad (2)$$

where $T(t)$ is the accumulated transmittance from t_n to t.

To optimize the network $f(\cdot;\theta)$, the photometric loss $\mathcal{L}_{\text{photo}}(\theta)$ enforces the rendered pixel color value $C(\mathbf{r};\theta)$ to be consistent with the ground-truth pixel color value $C^{\text{gt}}(\mathbf{r})$:

$$\mathcal{L}_{\text{photo}}(\theta) = \sum_{\mathbf{r}\in\mathcal{R}} \|C^{\text{gt}}(\mathbf{r}) - C(\mathbf{r};\theta)\|_2^2, \quad (3)$$

where $\mathcal{R}$ is the set of rays of each pixel in the image set $\mathcal{S}$.

3.2. Motivation

Despite its impressive performance, NeRF has the critical drawback of requiring large amounts of posed input images $\mathcal{S}$ for robust scene reconstruction. Naïvely optimizing NeRF in a few-shot setting (e.g., $|\mathcal{S}| < 10$) results in NeRF producing erroneous artifacts and wrong geometry due to the task's under-constrained nature [23].

A closer look reveals important details regarding the nature of the few-shot NeRF optimization. As described by the PSNR graph in Figure 1, all existing methods show a noticeable PSNR drop even with slight viewpoint changes, which indicates the tendency of NeRF to *memorize* the given input views. Such a tendency results in broken geometry that looks perfect in known viewpoints but progressively degenerates as the rendering view gets further away from known views. Although training with additional data directly solves this problem, obtaining high-quality images with accurate poses is extremely expensive. Instead, we notice that although images (rendered from NeRF trained with only sparse viewpoints) contain artifacts and erroneous geometry, there are reliable pixels of the image that are close to ground truth pixels, which can be used as additional data.

To check if using reliable pixels from the rendered images as additional data can help prevent NeRF from overfitting, we conduct an experiment of first optimizing NeRF under the identical few-shot setting. After training a teacher NeRF with three images, we train a new student NeRF with the extended set of images $\mathcal{S}\cup\mathcal{S}^+$ where $\mathcal{S}^+$ is the set of rendered images. To train with only the reliable pixels of $\mathcal{S}^+$, we define a binary reliability mask $M(\mathbf{r})$, which masks out pixels where the difference between the rendered color value $C(\mathbf{r};\theta^{\mathbb{T}})$ and its ground truth color value $C^{\text{gt}}(\mathbf{r})$ is above a predetermined threshold. Training the student NeRF to follow the reliably rendered color values $\{C(\mathbf{r};\theta^{\mathbb{T}}) \,|\, M(\mathbf{r}) = 1\}$ of the teacher can be seen as a weak distillation from the teacher to the student. The new student NeRF is trained with the following loss function:

$$\mathcal{L}_{\text{photo}}(\theta) + \lambda \sum_{\mathbf{r}\in\mathcal{R}^+} M(\mathbf{r})\|C(\mathbf{r};\theta^{\mathbb{T}}) - C(\mathbf{r};\theta)\|_2^2, \quad (4)$$

where $\mathcal{R}^+$ is a set of rays corresponding to each pixel in the rendered image set $\mathcal{S}^+$, and λ denotes the weight parameter.

The result of this experiment, described in "GT Masked" of the PSNR graph in Figure 1, shows that the student trained with K-Planes [12] as the baseline, displays staggering improvement in performance, with *unknown* viewpoints showing higher PSNR values and their rendered geometry remaining highly robust and coherent. This leads us to deduce that a major cause of few-shot NeRF geometry breakdown is its tendency to *memorize* the given sparse viewpoints and that selected distillation of additional reliable rays is crucial to enhance the robustness and coherence of 3D geometry. Based on this observation, our concern now moves on to how to estimate the reliability mask M for the rendered novel images of $\mathcal{S}^+$ to develop a better few-shot NeRF model.

4. Method

4.1. Teacher-student framework

Teacher network optimization. A teacher network is trained naïvely by optimizing the standard NeRF photometric loss where the number of known viewpoints is $|\mathcal{S}| < 10$. During this process, NeRF recovers accurate geometry for certain regions and inaccurate, broken geometry in other regions. The parameters of teacher network $\theta^{\mathbb{T}}$ is optimized with the loss function $\mathcal{L}_{\text{photo}}(\theta)$.

Pseudo labeling with teacher network. By evaluating the teacher NeRF representation $\theta^{\mathbb{T}}$, we can generate per-ray pseudo labels $\{C(\mathbf{r};\theta^{\mathbb{T}})|\ \mathbf{r} \in \mathcal{R}^+\}$ from the rendered images $\mathcal{S}^+$ from unknown viewpoints. To accurately identify and distill the reliable regions of $\mathcal{S}^+$ to the student model, we assess the reliability of every pseudo label in $\mathcal{R}^+$ to acquire a reliability mask $M(\mathbf{r})$ using a novel reliability estimation method we describe in detail in Section 4.2.

Student network optimization. The student network $\theta^{\mathbb{S}}$ is then trained with the extended training set of $\mathcal{S} \cup \mathcal{S}^+$, with the reliability mask M taken into account. In addition to the photometric loss with the initial image set $\mathcal{S}$, the student network is also optimized with a distillation loss that encourages it to follow the robustly reconstructed parts of the teacher model in $\mathcal{S}^+$. In the distillation process, the estimated reliability mask M determines how each ray should be distilled, a process which we explain further in Section 4.3. In summary, student network $\theta^{\mathbb{S}}$ is optimized with the following loss function:

$$\mathcal{L}_{\text{photo}}(\theta) + \lambda \sum_{\mathbf{r}\in\mathcal{R}^+} M(\mathbf{r})\|C(\mathbf{r};\theta^{\mathbb{T}}) - C(\mathbf{r};\theta)\|_2^2, \quad (5)$$

where $C(\mathbf{r};\theta^{\mathbb{T}})$ and $C(\mathbf{r};\theta)$ is the rendered color of the teacher and student model, respectively and λ denotes the weight parameter.

Iterative labeling and training. After the student network is fully optimized, the trained student network becomes the teacher network of the next iteration for another distillation process to a newly initialized NeRF, as described in Figure 2. We achieve improvement of the NeRF's quality and robustness every iteration with the help of the continuously extended dataset.

4.2. Ray reliability estimation

To estimate the reliability of per-ray pseudo labels $\{C(\mathbf{r};\theta^{\mathbb{T}})|\ \mathbf{r} \in \mathcal{R}^+\}$ from the rendered images $\mathcal{S}^+$, we expand upon an important insight that if a ray has accurately recovered a surface location and this location is projected to multiple viewpoints, the semantics of the projected locations should be consistent except for occlusions between viewpoints. This idea has been used in previous works that formulate NeRF for refined surface reconstruction [9], but our work is the first to leverage it for explicitly modeling ray reliability in a self-training setting.

The surface location recovered by a ray $\mathbf{r}$ corresponding to pixel $\mathbf{p}_i$ of the viewpoint i can be projected to another viewpoint j with the extrinsic matrix $R_{i\to j}$, intrinsic matrix K, and the estimated depth D_i from viewpoint i with the following projection equation:

$$\mathbf{p}_{i\to j} \sim K R_{i\to j} D_i(\mathbf{r}) K^{-1} \mathbf{p}_i. \quad (6)$$

Using the projection equation, we can make corresponding pixel pairs between viewpoint i and j such as $(\mathbf{p}_i, \mathbf{p}_j)$ where $\mathbf{p}_j = \mathbf{p}_{i\to j}$. Similarly, if we acquire pixel-level feature maps from viewpoint i and j using a pre-trained 2D CNN, we can make corresponding feature pairs as $(f^i_{\mathbf{p}}, f^j_{\mathbf{p}})$. In our case, by projecting the feature vector of the corresponding pseudo label $\{C(\mathbf{r};\theta^{\mathbb{T}})|\ \mathbf{r} \in \mathcal{R}^+\}$ to all given input viewpoints, we can achieve $|\mathcal{S}|$ feature pairs for every pseudo label. To generate a reliability mask for each ray, if a ray has at least one feature pair whose similarity value is higher than the threshold value $\boldsymbol{\tau}$, it indicates that the feature consistency of the ray's rendered geometry has been confirmed and classify such rays as reliable. Summarized in equation, the binary reliability mask $M(\mathbf{r})$ for the ray $\mathbf{r}$ rendered from viewpoint i can be defined as follows:

$$M(\mathbf{r}) = \min\left\{ \sum_{j\in|\mathcal{S}|} \mathbb{1}\left[\frac{\mathbf{f}^i_{\mathbf{p}} \cdot \mathbf{f}^j_{\mathbf{p}}}{\left\|\mathbf{f}^i_{\mathbf{p}}\right\| \left\|\mathbf{f}^j_{\mathbf{p}}\right\|} > \boldsymbol{\tau} \right], 1 \right\}. \quad (7)$$

To prevent the unreliable rays from being misclassified as reliable, we must carefully choose the threshold $\boldsymbol{\tau}$. Although using a fixed value for the $\boldsymbol{\tau}$ is straightforward, we find that choosing the adequate value is extremely cumbersome as the similarity distribution for each scene varies greatly. Instead, we adopt the adaptive thresholding method, which chooses the threshold by calculating the

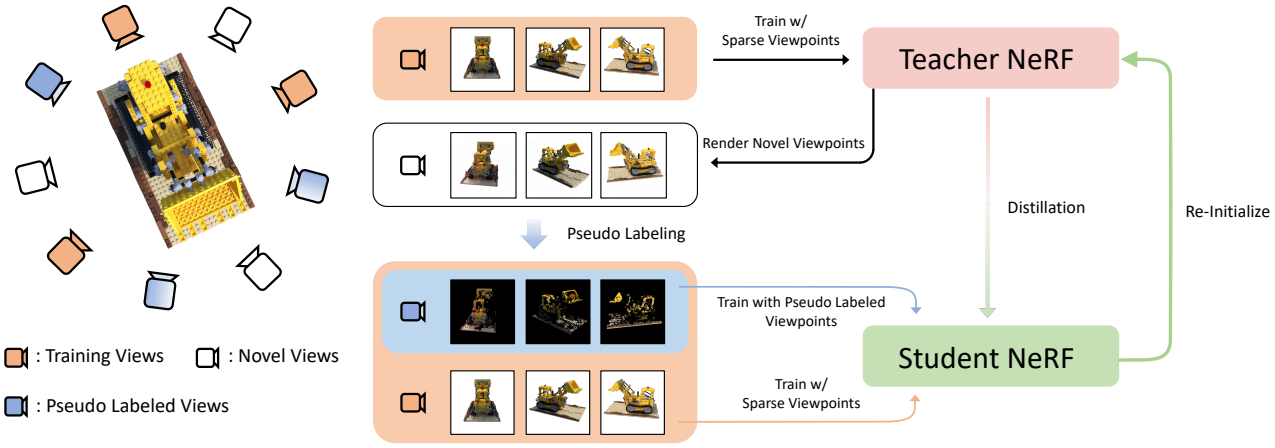


Figure 2. **Illustration of our overall framework for applying self-training to NeRF.** **SE-NeRF** utilizes the self-training framework to distill the knowledge of learned appearance and 3D geometry from teacher to student. The process is done iteratively as the student becomes the new teacher.

$(1-\alpha)^{\text{th}}$ percentile of the similarity distribution where α is a hyperparameter in the range $\alpha \in [0, 1]$. This enables the threshold $\boldsymbol{\tau}$ to be dynamically adjusted to each scene, leading to a better classification of the reliable rays.

4.3. Reliability-based distillation

To guide the student network to learn a more robust representation of the scene, we distill the label information from the teacher to the student with two distinct losses based on the ray's reliability. By remembering the rays evaluated in the teacher network and re-evaluating the same rays in the student network, the geometry and color information of reliable rays is directly distilled into the student network through distillation loss, while the rays classified as unreliable are regularized with nearby reliable rays for improved geometry before applying the distillation loss.

Reliable ray distillation. Since we assume the reliable rays' appearance and geometry have been accurately predicted by the teacher network, we directly distill their rendered color so that the student network faithfully follows the outputs of the teacher for these reliable rays. With the teacher-generated per-ray pseudo labels $\{C(\mathbf{r};\theta^{\mathbb{T}})|\ \mathbf{r} \in \mathcal{R}^{+}\}$ from the rendered images $\mathcal{S}^{+}$ and the estimated reliability mask M, the appearance of a reliable ray is distilled by the reformulated photometric loss $\mathcal{L}_c^{\mathcal{R}}$:

$$\mathcal{L}_c^{\mathcal{R}}(\theta) = \sum_{\mathbf{r}\in\mathcal{R}^{+}} M(\mathbf{r}) \| C(\mathbf{r};\theta^{\mathbb{T}}) - C(\mathbf{r};\theta) \|_2^2. \tag{8}$$

In addition to the photometric loss $\mathcal{L}_c^{\mathcal{R}}$, we follow Deng et al. [10], Roessle et al. [28] of giving the depth-supervision together to NeRF. As the teacher network $\theta^{\mathbb{T}}$ also outputs the density $\sigma(\mathbf{r};\theta^{\mathbb{T}})$ for each of the rays, we distill the density weights of the sampled points of the reliable rays to the student network. Within the same ray, we select an identical number of points randomly sampled from evenly spaced bins along the ray. This allows us to follow the advantages of injecting noise to the student as in Xie et al. [39] as randomly sampling points from each bin induces each corresponding point to have slightly different positions, which acts as an additional noise to the student.

The density distillation is formulated by the geometry distillation loss $\mathcal{L}_g^{\mathcal{R}}$, which is L2 loss between accumulated density values of corresponding points within the teacher and student rays, with teacher rays' density values $\sigma^{\mathbb{T}}$ serving as the pseudo ground truth labels. Therefore, for reliable rays, our distillation loss along the camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ is defined as follows:

$$\mathcal{L}_g^{\mathcal{R}}(\theta) = \sum_{\mathbf{r}\in\mathcal{R}^{+}} \sum_{t,t'\in T} M(\mathbf{r}) \| \sigma(\mathbf{r}(t);\theta^{\mathbb{T}}) - \sigma(\mathbf{r}(t');\theta) \|_2^2, \tag{9}$$

where T refers to the evenly spaced bins from t_n to t_f,and t and t' indicate randomly selected points from each bins.

Unreliable ray distillation. In many semi-supervised methods [26, 39], unreliable labels are ignored to prevent the confirmation bias problem. Similarly, unreliable rays must not be directly distilled as they are assumed to have captured inaccurate geometry. However, stemming from the prior knowledge that depth is smooth above the surface, we propose a novel method for regularizing the unreliable rays with geometric priors of nearby reliable rays, dubbed prior-based distillation.

To distill the knowledge of nearby reliable rays, we calculate a weighted average of nearby reliable rays' density distribution and distill this density to the student. As de-

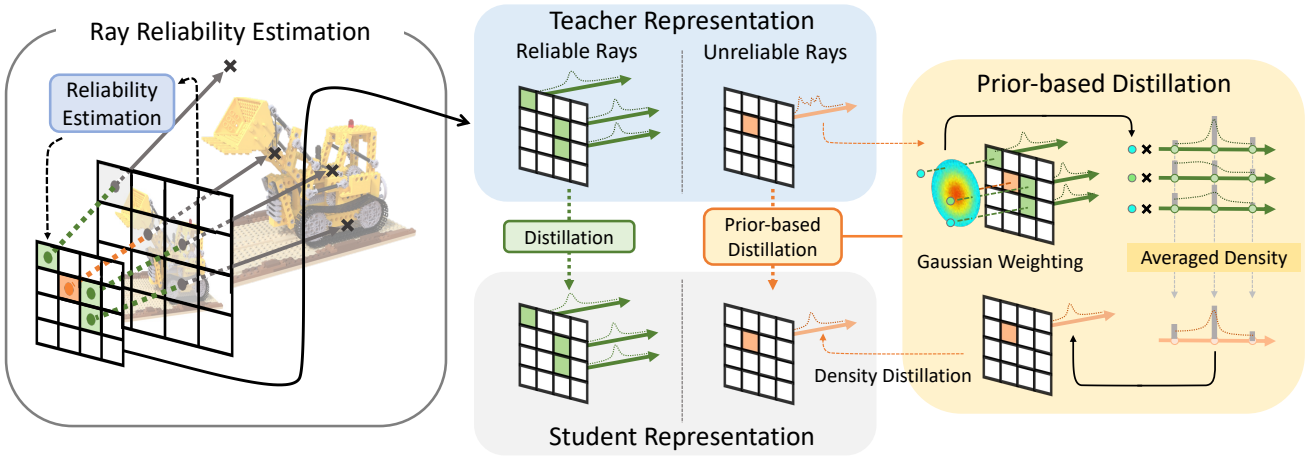


Figure 3. **Distillation of pseudo labels.** After estimating the reliability of the rays from unknown views, we apply distinct distillation schemes for reliable and unreliable rays. Reliable rays are directly distilled to the student while we aggregate the nearby reliable rays to regularize the unreliable rays.

scribed in Figure 3, we apply a Gaussian mask to unreliable ray r to calculate per-ray weights for nearby reliable rays. The intuition behind this design choice is straightforward: the closer a ray is to an unreliable ray, the more likely it is to be that the geometry of the two rays will be similar. Based on these facts, we apply the prior-based geometry distillation loss $\mathcal{L}_g^{\mathcal{P}}$, which is the L2 loss between the weighted-average density $\tilde{\sigma}(\mathbf{r};\theta^{\mathbb{T}})$ and the student density outputs $\sigma(\mathbf{r};\theta)$, is described in the following equation:

$$\mathcal{L}_g^{\mathcal{P}}(\theta) = \sum_{\mathbf{r}\in\mathcal{R}^+} \sum_{t,t'\in T} (1-M(\mathbf{r}))\|\tilde{\sigma}(\mathbf{r}(t);\theta^{\mathbb{T}}) - \sigma(\mathbf{r}(t');\theta)\|_2^2. \tag{10}$$

We apply the prior-based geometry distillation loss to the unreliable rays only when adjacent reliable rays exist. A more detailed explanation can be found in Appendix C.3.

Total distillation loss. Finally, our entire distillation loss function can be formulated as follows:

$$\mathcal{L}_{\text{photo}}(\theta) + \lambda_c^{\mathcal{R}}\mathcal{L}_c^{\mathcal{R}}(\theta) + \lambda_g^{\mathcal{R}}\mathcal{L}_g^{\mathcal{R}}(\theta) + \lambda_g^{\mathcal{P}}\mathcal{L}_g^{\mathcal{P}}(\theta), \tag{11}$$

where $\lambda_c^{\mathcal{R}}$, $\lambda_g^{\mathcal{R}}$, and $\lambda_g^{\mathcal{P}}$ denotes the weight parameters.

5. Experiments

5.1. Setups

Datasets and metrics. We evaluate our methods on NeRF Synthetic [21] and LLFF dataset [20]. For the NeRF Synthetic dataset, we randomly select 4 views in the train set and use 200 images in the test set for evaluation. For LLFF, we chose every 8-th image as the held-out test set and randomly select 3 views for training from the remaining images. In addition, we find that all existing NeRF models' performance on the NeRF Synthetic dataset is largely affected by the randomly selected views. To explore the robustness of our framework and existing methods, we introduce a novel evaluation protocol of training every method with an extreme 3-view setting (NeRF Synthetic Extreme) where all the views are selected from one side of the scene. The selected views can be found in Appendix D. We report PSNR, SSIM [38], LPIPS [45] and geometric average [4] values for qualitative comparison.

Implementation details. As any NeRF representation is viable, we conduct experiments using our framework with NeRF [21], K-Planes [12], and Instant-NGP [22] to demonstrate the versatile applicability of our framework. For our reliability estimation method, we use VGGNet [32], specifically VGG-19, and utilize the first 4 feature layers located before the pooling layers. For efficient training, we train the networks with only a portion of the original number of steps in each iteration, not to exceed the overall training time of each of the baseline models. Specific training details for each model are provided in Appendix B.

Hyper-parameters. We set the adaptive threshold value at $\alpha = 0.15$ for the first iteration. To enable the network to benefit from more reliable rays for each subsequent iteration, we employ a curriculum labeling [6] approach that increases α by 0.05 every iteration. As images rendered from views near the initial inputs include more reliable regions, we progressively increase the range of where the pseudo labels should be generated. We start by selecting views that are inside the range of 10 degrees in terms of ϕ, θ of the initial input and increase range after iterations. For the weights

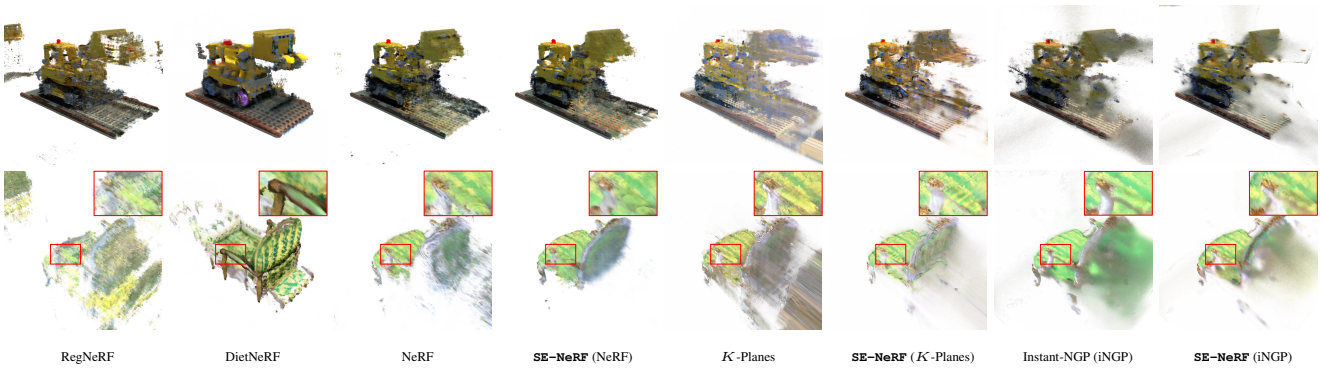


Figure 4. **Qualitative comparison on NeRF Synthetic Extreme.** The results show the rendered images from viewpoints far away from the seen views. A noticeable improvement over existing models regarding artifacts and distortion removal can be observed in **SE-NeRF**.

Methods	NeRF Synthetic Extreme				NeRF Synthetic				LLFF			
	PSNR↑	SSIM↑	LPIPS↓	Avg.↓	PSNR↑	SSIM↑	LPIPS↓	Avg.↓	PSNR↑	SSIM↑	LPIPS↓	Avg.↓
DietNeRF	14.46	0.72	0.28	0.17	15.42	0.73	0.31	0.17	14.94	0.37	0.50	0.23
InfoNeRF	14.62	0.74	0.26	0.17	18.44	0.80	0.22	0.11	13.57	0.33	0.58	0.28
RegNeRF	13.73	0.70	0.30	0.19	13.71	0.79	0.35	0.19	19.08	0.59	0.34	0.15
Self-NeRF †	-	-	-	-	20.66	0.84	0.18	0.09	-	-	-	-
NeRF	14.85	0.73	0.32	0.18	19.38	0.82	0.17	0.10	17.79	0.50	0.44	0.17
SE-NeRF (NeRF)	17.41 (+2.56)	0.78 (+0.05)	0.21 (-0.11)	0.12 (-0.06)	20.66 (+1.28)	0.84 (+0.02)	0.16 (-0.01)	0.08 (-0.02)	18.73 (+0.94)	0.55 (+0.05)	0.41 (-0.03)	0.15 (-0.02)
K-Planes	15.45	0.73	0.28	0.16	17.99	0.82	0.18	0.11	16.40	0.39	0.47	0.20
SE-NeRF (K-Planes)	17.49 (+2.04)	0.78 (+0.05)	0.23 (-0.05)	0.12 (-0.04)	19.93 (+1.94)	0.83 (+0.01)	0.17 (-0.01)	0.09 (-0.02)	16.76 (+0.36)	0.42 (+0.03)	0.45 (-0.02)	0.19 (-0.01)
Instant-NGP (iNGP)	15.87	0.72	0.39	0.17	16.53	0.75	0.33	0.15	16.60	0.45	0.47	0.20
SE-NeRF (iNGP)	17.52 (+1.65)	0.77 (+0.05)	0.32 (-0.07)	0.14 (-0.03)	17.78 (+1.25)	0.77 (+0.02)	0.31 (-0.02)	0.14 (-0.01)	16.93 (+0.33)	0.47 (+0.02)	0.46 (-0.01)	0.19 (-0.01)

Table 1. **Quantitative comparison on NeRF Synthetic and LLFF.** †: Limited comparison due to code not being made publicly available. A comparison on LLFF 2-views can be found in Appendix E.

for our total distillation loss, we use $\lambda_c^{\mathcal{R}} = 1.0$, $\lambda_g^{\mathcal{R}} = 1.0$, and $\lambda_g^{\mathcal{P}} = 0.005$.

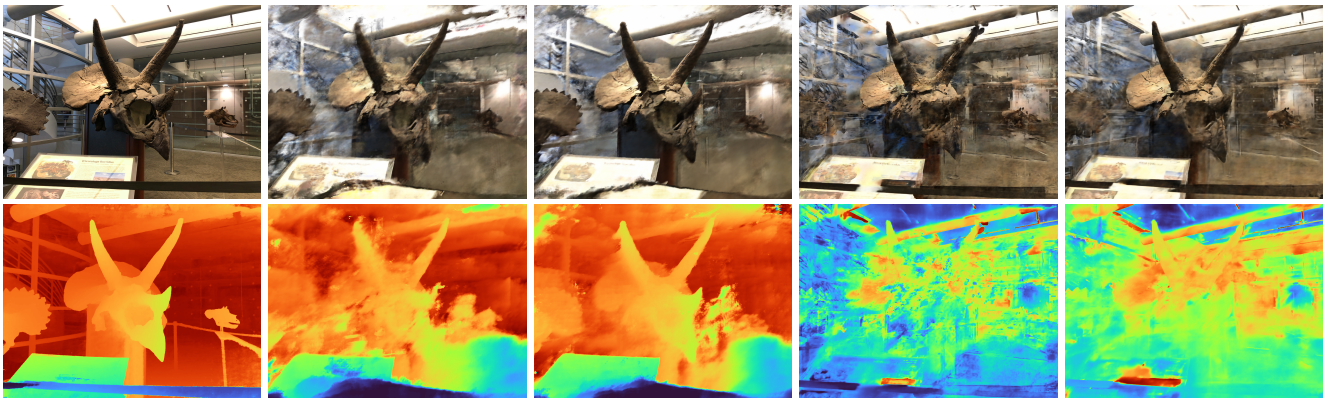
5.2. Comparison

Qualitative comparison. Figure 4 and Figure 5 illustrate the robustness of our model to unknown views, even when the pose differs significantly from the training views. Our model demonstrates robust performance on unknown data, surpassing the baselines. This is particularly evident in the "chair" scene, where all existing methods exhibit severe overfitting to the training views, resulting in heavy artifacts when the pose significantly changes from those used during training. In contrast, **SE-NeRF** maintains the shape of an object even from further views with less distortion, resulting in the least artifacts and misrepresentation.

Quantitative comparison. Table 1 and Table 2 show quantitative comparisons of multiple baseline models with and without applying our framework, and other few-shot NeRF models on NeRF synthetic and LLFF datasets. As shown in Table 1, **SE-NeRF** outperforms previous few-shot NeRF models in the NeRF synthetic Extreme and the conventional 4-view setting. By applying **SE-NeRF**, we observe an general improvement in performance over different methods and different datasets, demonstrating that our framework successfully guides networks of existing methods to learn more robust knowledge of the 3D scene.

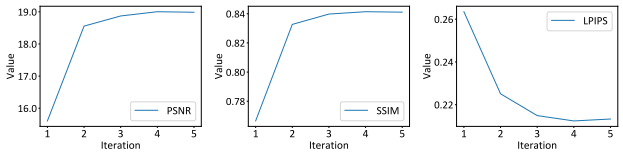
Methods	chair	drums	ficus	hotdog	lego	mater.	ship	mic
NeRF	15.08	11.98	17.16	13.83	16.31	17.31	10.84	16.29
K-Planes	15.61	13.23	18.29	12.45	14.67	16.30	13.35	19.74
Instant-NGP (iNGP)	17.66	12.75	18.44	13.64	14.72	16.83	13.82	19.05
DietNeRF	16.60	8.09	18.32	19.00	11.45	16.97	15.26	10.01
InfoNeRF	15.38	12.48	18.59	19.04	12.27	15.25	7.23	16.76
RegNeRF	15.92	12.09	14.83	14.06	14.86	10.53	11.44	16.12
SE-NeRF (NeRF)	19.96 (+4.88)	14.72 (+2.74)	19.29 (+2.13)	16.06 (+2.23)	16.45 (+0.14)	17.51 (+0.20)	14.20 (+3.36)	21.09 (+4.80)
SE-NeRF (K-Planes)	20.54 (+4.93)	13.38 (+0.15)	18.33 (+0.04)	20.14 (+7.69)	16.65 (+1.98)	17.01 (+0.71)	13.72 (+0.37)	20.13 (+0.39)
SE-NeRF (iNGP)	20.40 (+2.74)	13.34 (+0.59)	19.07 (+0.63)	18.15 (+4.51)	16.41 (+1.69)	17.94 (+1.11)	14.61 (+0.79)	20.23 (+1.18)

Table 2. **Quantitative comparison per-scene on NeRF Synthetic Extreme.**



Ground-truth NeRF **SE-NeRF** (NeRF) K-Planes **SE-NeRF** (K-Planes)

Figure 5. **Qualitative improvement from baselines.**



(a) PSNR ↑ (b) SSIM ↑ (c) LPIPS ↓

Figure 6. **Quantitative improvement from baseline after multiple iterations.**

5.3. Ablation study.

Iterative training. As shown in Figure 6, which presents the quantitative results for each iteration, a significant improvement in performance can be observed after the first iteration. The performance continues to be boosted with each subsequent iteration until the convergence. Based on our experimental analysis, we find that after the simultaneous distillation of reliable rays and regularization of unreliable rays in the first iteration, there is much less additional knowledge to distill to the student in certain scenes which leads to a smaller performance gain from the second iteration. However, although the performance gain in terms of metrics is small, the remaining artifacts and noise in the images continue to disappear after the first iteration, which is important in perceptual image quality.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	Average ↓
K-Planes	14.67	0.68	0.31	0.18
+ Reliable ray distillation	16.15 (+1.48)	0.72 (+0.04)	0.27 (-0.04)	0.15 (-0.03)
+ Unreliable ray distillation	**16.65** (+1.98)	**0.75** (+0.07)	**0.24** (-0.07)	**0.14** (-0.04)

Table 3. **Ray distillation ablation.**

Prior-based ray distillation. In Table 3, we conduct an ablation study on the "lego" scene of the NeRF Synthetic Extreme setting and show that using both reliable and unreliable ray distillation is crucial to guide the network to learn a more robust representation of the scene, showing the highest results in all metrics. This stands in contrast to existing semi-supervised appraoches [1, 39], which typically discard unreliable pseudo labels to prevent the student learning from erroneous information [2]. We show that for NeRFs, the unreliable labels can be further utilized by the prior knowledge that depth within a 3D space exhibits smoothness.

Threshold	PSNR ↑	SSIM ↑	LPIPS ↓	Avg. ↓
Fixed	17.02	0.77	0.25	0.13
Unified	15.95	0.73	0.28	0.16
Adaptive	**17.49**	**0.78**	**0.23**	**0.12**

Table 4. **Thresholding ablation.**

Thresholding. In Table 4, we show the results of **SE-NeRF** trained on NeRF Synthetic Extreme setting with different thresholding strategies. Following traditional semi-supervised approaches [6, 8, 35, 43], we conduct experiments using a fixed threshold, adaptive threshold (ours), and a unified threshold which does not classify psuedo labels as reliable and unreliable but uses the similarity value to decide how much the distillation should be made from the teacher to the student. The adaptive thresholding method resulted in the most performance gain, showing the rationale of our design choice. A more detailed analysis of thresholding strategies is provided in Appendix C.4.

6. Conclusion

In this paper, we present a novel self-training framework called framework self-evolving neural radiance fields (**SE-NeRF**), specifically designed for few-shot NeRF. By employing a teacher-student framework in conjunction with our unique implicit distillation method, which is based on the estimation of ray reliability through feature consistency, we demonstrate that our self-training approach yields a substantial improvement in performance without the need for any 3D priors or modifications to the original architecture.

References

[1] Massih-Reza Amini, Vasilii Feofanov, Loic Pauletto, Emilie Devijver, and Yury Maximov. Self-training: A survey, 2023. 8

[2] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020. 8, 12, 21

[3] Jiayang Bai, Letian Huang, Wen Gong, Jie Guo, and Yanwen Guo. Self-nerf: A self-training pipeline for few-shot neural radiance fields, 2023. 2, 26

[4] 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 (ICCV)*, pages 5855–5864, 2021. 2, 6

[5] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009. 18

[6] Paola Cascante-Bonilla, Fuwen Tan, Yanjun Qi, and Vicente Ordonez. Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6912–6920, 2021. 6, 8, 18

[7] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14124–14133, 2021. 2

[8] Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. *arXiv preprint arXiv:2301.10921*, 2023. 8, 18

[9] 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. 2, 4, 11

[10] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022. 2, 5

[11] S Fralick. Learning to recognize patterns without a teacher. *IEEE Transactions on Information Theory*, 13(1):57–64, 1967. 2

[12] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *CVPR*, 2023. 1, 2, 3, 4, 6, 11, 21, 26

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 11

[14] 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

[15] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*, 18(3):165–174, 1984. 2

[16] 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

[17] Minseop Kwak, Jiuhn Song, and Seungryong Kim. Geconerf: Few-shot neural radiance fields via geometric consistency. *arXiv preprint arXiv:2301.10941*, 2023. 1, 2

[18] Ruilong Li, Hang Gao, Matthew Tancik, and Angjoo Kanazawa. Nerfacc: Efficient sampling accelerates nerfs. *arXiv preprint arXiv:2305.04966*, 2023. 11

[19] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7210–7219, 2021. 2

[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 (TOG)*, 38(4):1–14, 2019. 6, 26

[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. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2, 6, 11, 21

[22] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, 2022. 6, 11

[23] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490, 2022. 1, 2, 3

[24] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 21

[25] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 11

[26] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *Proceedings of the IEEE/CVF conference*

on computer vision and pattern recognition, pages 11557–11568, 2021. 2, 5, 14

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2

[28] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12892–12901, 2022. 2, 5

[29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 11

[30] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 2

[31] H Scudder. Adaptive communication receivers. *IEEE Transactions on Information Theory*, 11(2):167–174, 1965. 2

[32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6, 11

[33] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020. 2

[34] 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. 11

[35] Gokhan Tur, Dilek Hakkani-Tür, and Robert E Schapire. Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, 45(2): 171–186, 2005. 8, 18

[36] 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, 2

[37] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 2, 11

[38] 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

[39] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020. 2, 5, 8, 14

[40] Dejia Xu, Yifan Jiang, Peihao Wang, Zhiwen Fan, Humphrey Shi, and Zhangyang Wang. Sinnerf: Training neural radiance fields on complex scenes from a single image. In *European Conference on Computer Vision*, pages 736–753. Springer, 2022. 2

[41] Jiawei Yang, Marco Pavone, and Yue Wang. Freenerf: Improving few-shot neural rendering with free frequency regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8254–8263, 2023. 2

[42] 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, 2, 11

[43] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419, 2021. 8, 18

[44] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6525–6534, 2021. 11

[45] 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