From Low to High: Adversarial Training of NeRF for High-Resolution Novel View Synthesis from Single Image

Jiwon Shin
jiwonshin@snu.ac.kr

Gyeong Chan Kim
skykim0609@snu.ac.kr

Jae Gwan Ahn
jk.ahn7@dm.snu.ac.kr

Jihoon Han
joon7092@snu.ac.kr

Abstract

Neural Radiance Field (NeRF) model aims to synthesize images of scenes for unseen views. Recently, multiple works on generalizable NeRF models were proposed to relax the heavy requirement for training the NeRF model, and allow for few-show Novel view synthesis. Although there has been much of thoughtful prior research carried out to fulfill this objective, output of these models tends to become highly blurry when the input images are low-resolution. It is easy to verify that training a model with low-quality images leads to more severe blurring. In this paper, we introduce Adversarial SR-NeRF,a Neural Radiance Field model that can synthesize high quality images of objects for unseen views with only low resolution image as a reference during inference. The model consists of three main parts: convolutional neural networks to encode images, a super-resolution (SR) decoder to learn representative features that map low-resolution image features to high-resolution image features, and a discriminator to determine whether an image is high-resolution.

1. Introduction

Novel view synthesis is a challenging task of synthesizing images from an unseen view given input images. While recent advancements in Neural Radiance Fields (NeRF) have shown promising results in generating novel view images, it requires hundreds of input images to train for a single scene, and lacks the capability to share knowledge across multiple scenes. Furthermore, when the model is inquired to render novel views with higher resolution compared to the observed images, the results typically suffer from blurriness and shortage of details.

The main contribution of our work is a novel approach that enhances the resolution of outputs from the few-shot NeRF model, specifically PixelNeRF [18] trained with low-resolution input by incorporating adversarial training. The motivation behind solving this problem is to obtain high-resolution and sharp renderings for NeRF-generated images trained with low-resolution input, which are blurry and lack fine details. This enhancement can greatly improve the visual quality of the rendered images with few low-quality data available at runtime, making them more suitable for various applications, and even generate images with higher resolution.

Our model builds upon PixelNeRF [18], which takes a single input image and learns a scene prior from a set of multi-view images. PixelNeRF enables the network to perform novel view synthesis in a feed-forward manner from a sparse set of views. This model encodes 3D information using local 2D image features, making the learned representations highly generalizable to unseen scenes after being trained on a lot of scenes. This is a significant advance in the field as it renders from minimal input data. Specifically, we condition NeRF on input images by first computing a fully convolutional image feature grid from the input image. For each query spatial point $\mathbf{x}$ and viewing direction $\mathbf{d}$ of interest in the view coordinate frame, we sample the corresponding image feature via projection and bilinear interpolation.

To extend the capabilities of PixelNeRF, we incorporate a new component to the architecture. On top of pixelNeRF, we train a decoder structure which takes the feature map to generate high resolution image corresponding to the input view. By adding this process, we expect to learn representative features that can generate high resolution output images. Additionally, we adapt an adversarial training framework to enforce the distribution of the rendered high-resolution images to resemble the real images. We expect the proposed approach to outperform baseline methods trained on low-resolution images in ShapeNet [2] dataset with greater detail and reduced blur.

Our main contributions are summarized as follows:

- We present Adversarial SR-NeRF, a Neural Radiance

Field model that can synthesize high quality images of objects for unseen views with only low resolution images during inference.

- Utilizing the SR-Decoder, the model is capable of learning representative features that map low-resolution image features to high-resolution images.
- Addition of the discriminator enables the blurry images of standard NeRF outputs to be reconstructed into more detailed high-resolution image.

2. Related Works

2.1. Novel View Synthesis

Novel view synthesis is the problem of generating new views of a scene or object that were not seen directly from a given set of existing images or data. Since the vast success of deep learning in computer vision, plethora of researches have investigated on novel view synthesis with deep neural networks trained to encode 3D scene representation. Methods such as Deep-SDF [13], implicit occupancy fields [7], and Scene representation networks [15] demonstrated the capability of neural networks to learn useful prior information from training data to reconstruct 3D shapes for given input. Combined with differentiable rendering [6], these methods can render shape and appearance of the scenes.

Within these approaches, Neural Radiance Fields(NeRF) [11] established outstanding results compared to its competitors. NeRF renders detailed and photorealistic images of novel views, by modeling the scene as continuous function that maps volume density and radiance field dependent on position and viewing direction. However, it comes with limitations, such as the requirement of hundreds of images per scene with corresponding camera pose for training and a lack of generalization across multiple scenes, unlike other frameworks.

Researchers have been actively exploring for extensions of NeRF with the aim of enabling it to learn shared priors from training on multiple scenes. These variants of NeRF is referred to as generalizable NeRF [16]. By learning 3D priors coherent across multiple scenes, generalizable NeRF models are also able to render target viewpoint image with much fewer input images for a specific scene compared to the original. CodeNeRF [4] employs auto-decoder framework to learn one dimensional shape and appearance code for a chosen category, and demonstrates novel view synthesis capability. PixelNeRF [18], an advancement over these generalizable NeRF models, uses a fully convolutional network to predict the neural radiance field from one or few images. PixelNeRF learns pixel-aligned feature map which is passed into the NeRF network alongside positional encoding and viewing direction. It not only exhibits high-quality results when the target viewpoint is close to the input view but also generalizes to unseen scenes and object categories, making it a flexible solution for various applications. While PixelNeRF achieves high-quality results when the target viewpoint is close to the input view, its inability to handle occlusion and reliance on local features leads to poor performance when the target viewpoint is far from the input. Vision-NeRF [5] demonstrates notable improvement in unseen views by combining 1D features learned by Vision Transformer(ViT) which encodes the global shape of the scene and 2D feature maps for detailed rendering around the input views.

2.2. NeRF Super Resolution

Vanilla NeRF often faces difficulties in effectively rendering images with higher resolutions beyond those of the input images, resulting in production of blurry views. A recent study, NeRF-SR [17] has explored the combination of NeRF and super-resolution techniques to address this limitation of the original NeRF. However, the model still lacks the ability to share knowledge across multiple training scenes which can reduce the number of required images and largely enhance the quality of rendered output. In this research we aim to develop a framework which renders high resolution image of novel views with single low resolution input image, by combining the cross-scene generalization capability of PixelNeRF [18] and adversarial training framework. By training discriminator model to correctly distinguish between the generated images and real images alongside the NeRF model, we expect the model to synthesize images which closely resemble the true high-resolution images.

2.3. Generative Adversarial Network(GAN)

Generative adversarial networks (GANs), a type of recent generative image models, have demonstrated remarkable capabilities in producing high-resolution and visually appealing images [1, 9]. Recently, diffusion models became the new standard for large scale generative models and considered better method than GANs in generative models [3]. However, compared to diffusion models, GANs not only has faster inference time and more applicable with 3D rendering and unsupervised learning of 3D representation from natural images.

2.4. 3D-Aware Image Synthesis

Recent works exploit generative 3D models for 3D-Aware Image Synthesis. Unlike 2D GANs, 3D GANs utilize a combination of two key elements: a generator network architecture that incorporates a 3D-structure-aware inductive bias and a neural rendering engine designed to produce consistent results from different viewpoints. HoloGAN [12] and GRAF [14] are the few early works that attempts to integrate NeRFs and GAN. It employs a low-

dimensional 3D feature along with a trainable 3D-to-2D projection. However, the learned projections in HoloGAN can result in intertwined latents, such as object identity and viewpoint, especially when dealing with high-resolution images. Improvements then made by GRAF [14], it excels in generating controllable images with high resolutions, this representation is limited to single-object scenes, and its performance declines when applied to more complex, real-world images.

3. Methodology

In this section, we describe the network architecture for the proposed method. The overview of the model is presented in figure 1. The model includes two modules which are common in generalizable NeRF models [5, 18] which are; an image encoder which extracts pixel-aligned feature map from the source image, and the Multi-layer perceptron(MLP) which combines the projected feature map and positional encoding and outputs the radiance field($\mathbf{c}, \boldsymbol{\sigma}$). On top of that, we connect two networks in training phase to allow the NeRF model to render detailed high-resolution output from low-resolution source images; a Super-Resolution(SR) decoder which impose the input feature map to retain information to render the corresponding high-resolution image, and a discriminator for the rendered output distribution to closely resemble the distribution of ground truth high resolution images. Details of each modules will be explained further in the following subsections.

3.1. Model

3.1.1 Image Encoder

Given a single 32 $\times$ 32 low resolution image, $\mathbf{I}_i^L$ as input, we start with an encoder-decoder fashioned structure to extract image features. The model to encode the image can be either the image encoder in PixelNeRF or VisionNeRF. [5]. When utilizing PixelNerF [18] image encoder, the input image is encoded by using ResNet34 backbone pretrained on ImageNet, providing features $\mathbf{W}$. On the other hand, when using VisionNeRF, the input image is divided into N = 8 $\times$ 8 patches P, then flattening the patches, we obtain image tokens. Additionally attaching positional embedding e, we pass them to a transformer encoder which outputs a latent feature $\mathbf{f}$, which represents the global information of the input image. Utilizing the convolutional decoder from VisionNeRF, we decode $\mathbf{f}$ into multi-level feature maps $\mathbf{W}_G$. In order to create a global and local aware representation vector $\mathbf{W}$, we extract 2D CNN features, $\mathbf{W}_L$, that contains local information of the image, and then fuse it with $\mathbf{W}_G$.

3.1.2 Volume Renderer

The volume renderer takes the value of pixel-aligned feature map as the input to the network on top of direction and positional encoding. Given pixel-aligned feature map $\mathbf{W}$, the target view direction d_c, and the positional encoding, γ, we output the color $\mathbf{c}$ and density σ using the NeRF Multi-Layer Perceptron (MLP) as follows:

$$(\sigma, c) = MLP(\gamma(x_c); d_c; W(\pi(\mathbf{x})))$$

For a ray $\mathbf{r}$, rendering is performed by calculating the following integral for radiance field over points on the ray as written below:

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

where $\pi(x)$ refers to the reprojection of the position $\mathbf{x}$ to the reference image plane as explained in [18]. In implementation, the integral is replaced with summation on finite sampled points along the ray. We train the model to minimize rendering loss $\mathcal{L}_{re}(\mathbf{I}_i^H, \hat{\mathbf{I}}_t^H)3$ which is computed by comparing the ground truth image, $\mathbf{I}_i^H$ and the volume rendered output, namely $\hat{\mathbf{I}}_t$.

Rendering high-resolution images directly from the model trained with low-resolution images typically fails to capture details required for HR images and tends to be blurry. Therefore, we would like to train the feature map to capture information to render such detail. To this end, we attach two additional networks; super-resolution decoder (SR) decoder and discriminator, to enable the model to learn from high resolution images. These networks are employed only during the training phase, and omitted during the inference phase.

3.1.3 Super-Resolution(SR) Decoder

As mentioned in section 3.1.1, we extract features from low-resolution images (32x32) using the image encoder. It is important to ensure the model understands how to map features from a low-resolution image space to high-resolution image space. Therefore, we introduce the (Super Resolution) SR-decoder, which uses three deconvolution layers to reproduce the high-resolution image of an object from an unseen view. Letting $\mathbf{W}_i$ be the pixel-aligned feature map for the reference image, and the SR-decoder, be $F_{\Theta_{sr}}$, the predicted high-resolution image is formed as $\hat{\mathbf{I}}_{sr}^H = F_{\Theta_{sr}}(\mathbf{W}_i)$. Then, given the high-resolution reference image, $\mathbf{I}_i^H$ as input, the SR-decoder is optimized by minimizing the loss which is constructed as follows.

$$\mathcal{L}_{sr}(\mathbf{I}_i^H, \hat{\mathbf{I}}_{sr}^H) = ||\mathbf{I}_i^H - \hat{\mathbf{I}}_{sr}^H||_1 \quad (1)$$

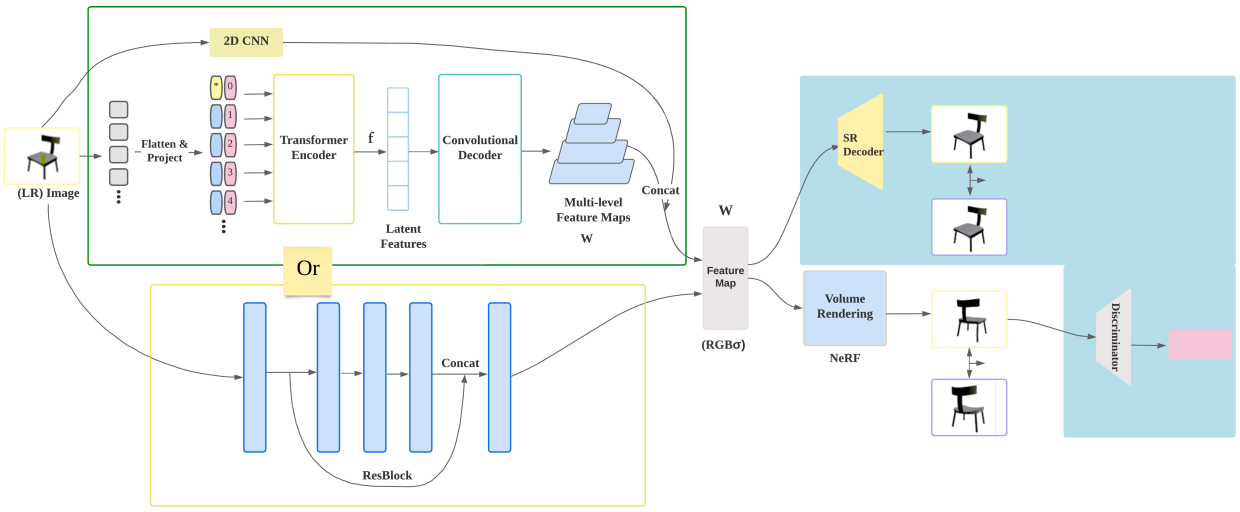

Figure 1. An overview of our single image-based High-resolution novel view synthesis framework. Blue shaded region(SR decoder and Discriminator) is the part which is only used during training to enhance the quality of high-resolution output.

The SR decoder module takes feature map of the PixelNeRF/VisionNeRF as input, and reconstruct the HR image at the input view. SR decoder is trained to minimize the difference between super-resolution result and the ground truth high resolution image. By adding Super-Resolution loss(2), we expect the network to learn the mapping from low resolution to high resolution domain.

3.1.4 Discriminator

We would like to ensure that the rendered target view image closely resemble the real image taken at the viewpoint. To fulfill this requirement, we adopt adversarial training framework. A discriminator network D_ϕ takes either rendered HR output $\hat{\mathbf{I}}_t$ or true HR image, and classifies whether it is real or synthesized image.

However, rendering a full image for single step training of the discriminator is to time consuming. Thus, to reduce the time required for training, we adopt a similar ray sampling technique as GRAF [14] to generate regular grid of pixel patches for both input to the rendering, and filter out pixels of GT High resolution images. Unlike GRAF, our goal is to impose output image of the renderer to achieve quality of detail which is exhibited in the GT HR images. Thus, we sample regular grid of small patches(8×8) so that the sampled pixels in ground truth images retain the desired detail. The pixel sampling strategy is displayed in figure 2.

The whole framework is trained by optimizing the combined loss function which is the sum of SR reconstruction loss(2), the photometric loss(3), and the adversarial loss(4). Throughout the training, We expect the generator network to synthesize a high resolution image from another viewpoint that can fool the discriminator.

Specific details about losses are outlined on section 3.2.

In the inference stage, given a single image as input, it follows the exact same PixelNeRF and VisionNeRF inference process. Simply passing the learned $\mathbf{W}$, target viewpoint, d_c, and the positional encoding, γ to the NeRF-MLP, and then processing volume rendering, we obtain the rendered high resolution image corresponding to the target view.

3.2. Training

We assume that the high resolution image corresponding to the low resolution input is available at training time. For each image pair $\{\mathbf{I}_i^L, \mathbf{I}_i^H\}$, The forward pass output of image super-resolution is compared with the ground truth HR image. The image reconstruction loss, $\mathcal{L}_r(\mathbf{I}_i^H, \hat{\mathbf{I}}_{sr}^H)$ penalize the difference between output of the super-resolution and the ground truth.

$$\mathcal{L}_{sr}(\mathbf{I}_i^H, \hat{\mathbf{I}}_{sr}^H) = ||\mathbf{I}_i^H - \hat{\mathbf{I}}_{sr}^H||_1 \tag{2}$$

We also demand for the rendered target view to be consistent with the ground truth image at the target viewpoint. We employ L2 norm loss for rendering loss function.

$$\mathcal{L}_{re}(\mathbf{I}_t^H, \hat{\mathbf{I}}_t^H) = \sum_{\mathbf{r} \in \text{Rays}} ||\hat{\mathbf{C}}_t(\mathbf{r}) - \mathbf{C}_t(\mathbf{r})||_2^2 \tag{3}$$

Finally, we train the convolutional discriminator D_ϕ to improve its ability to distinguish between real and rendered image at target views. To this end, we adopt non-saturating GAN loss with R1- regularization [10] loss.

$$\mathcal{L}_d(\theta, \phi) = \mathbb{E}\left[f\left(D_\phi\left(F_\theta((\mathbf{W}, T_t)\right)\right)\right] + \mathbb{E}\left[f\left(-D_\phi\left(F_\theta((\mathbf{W}, T_t)\right)\right) + \lambda \left\|\nabla D_\phi\left(F_\theta((\mathbf{W}, T_t)\right)\right\|^2\right] \tag{4}$$

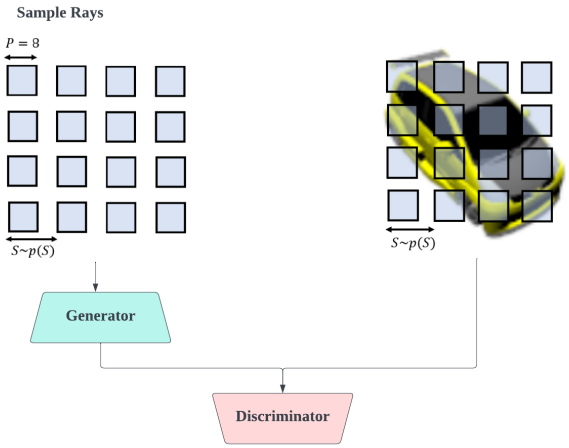

Figure 2. Ray sampling method utilized in the proposed work. Unlike in GRAF [14], we sample grid of patches with fixed size(=8) to capture the detail exhibited in GT high-resolution images

The combined loss function is formulated as the weighted sum of the aforementioned individual loss, and the volume rendering network, image reconstruction network and discriminator network is trained in an end-to-end fashion.

4. Experiment

4.1. Dataset and Comparison

We train and evaluate the model with the Shapenet [2] dataset. Each of the dataset consists of images of objects taken from multiple views and camera pose annotations. For generating low resolution images, we process the image by passing through a Gaussian Kernel and downsampling by a rate of 4.

We tried to compare the method's performance with PixelNeRF [18] and VisionNeRF [5] trained with low resolution images obtained by downsampling the images in the dataset at various ratios, 1, 2, and 4.

We planned to perform ablation studies by examining the effect of each loss functions in the proposed framework. We would compare the proposed method trained by optimizing combined loss function $\lambda_{sr}\mathcal{L}_{sr} + \lambda_{re}\mathcal{L}_{re} + \lambda_{d}\mathcal{L}_{d}$, with the models trained without adversarial loss($\mathcal{L}_{d}$) and super-resolution loss($\mathcal{L}_{sr}$) respectively.

4.2. Experimental Settings

We implement all experiments on top of PixelNeRF [18] using PyTorch 1.7.1. We train pairs of low-resolution and high-resolution images of the ShapeNet [2] SRN cars dataset. We set the batch size to be 4 and the number of epochs to be 400000 with a learning rate of 0.0001, which follows the same setting as PixelNeRF [18]. The experimental GPU consists of an NVIDIA A10-12Q, with 12GB VRAM and NVIDIA V100, with 32GB VRAM.

5. Future Works

In our plan, we are missing specifics about how to accelerate training and inference time for the implied NeRF model. Since projecting ray points at each pixel of the image takes plenty of time, investigating a way to speed up the training and inference of NeRF seems to be ideal. Another enhancement we can make later on is to use the well known Swin Transformer [8] for image encoding rather than the Vision Transformer. Then we expect faster inference times because Vision Transformer requires quadratic computational effort on the input image while Swin Transformer operates linearly, allowing for model scalability with relatively less computational resources.

6. Conclusion

To address the problem of regular NeRF models providing blurry images when trained with low resolution images, we propose, Adversarial SR-NeRF, a model that can synthesize high quality images of objects for unseen views with only low resolution images during inference. The model consists of three main parts: convolutional neural networks to encode images, a super-resolution (SR) decoder to learn representative features that map low-resolution image features to high-resolution image features, and a discriminator to determine whether an image is high-resolution. The architecture of the model is carefully designed, but unfortunately, due to lack of computational equipment and time, the results of the code could not be verified. We look forward to see the performance of the high-resolution view synthesis model to improve further once the discussed limitations are resolved.

References

[1] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis, 2019. 2

[2] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 1, 5

[3] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 8780–8794. Curran Associates, Inc., 2021. 2

[4] Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *Proceed-*

ings of the IEEE/CVF International Conference on Computer Vision, pages 12949–12958, 2021. 2

[5] Kai-En Lin, Yen-Chen Lin, Wei-Sheng Lai, Tsung-Yi Lin, Yi-Chang Shih, and Ravi Ramamoorthi. Vision transformer for nerf-based view synthesis from a single input image. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 806–815, 2023. 2, 3, 5

[6] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7708–7717, 2019. 2

[7] Shichen Liu, Shunsuke Saito, Weikai Chen, and Hao Li. Learning to infer implicit surfaces without 3d supervision. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[8] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *CoRR*, 2021. 5

[9] Youssef A. Mejjati, Christian Richardt, James Tompkin, Darren Cosker, and Kwang In Kim. Unsupervised attention-guided image to image translation, 2018. 2

[10] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018. 4

[11] 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. 2

[12] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images, 2019. 2

[13] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 2

[14] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis, 2021. 2, 3, 4, 5

[15] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[16] Alex Trevithick and Bo Yang. Grf: Learning a general radiance field for 3d representation and rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15182–15192, 2021. 2

[17] Chen Wang, Xian Wu, Yuan-Chen Guo, Song-Hai Zhang, Yu-Wing Tai, and Shi-Min Hu. Nerf-sr: High quality neural radiance fields using supersampling. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6445–6454, 2022. 2

[18] 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, 3, 5