InNeRF: Learning Interpretable Radiance Fields for Generalizable 3D Scene Representation and Rendering

Dan Wang[1]   Xinrui Cui[2]

[1]University of Copenhagen [2]Univesity of North Texas

dawa@di.ku.dk, xinrui.cui@unt.edu

Abstract

We propose Interpretable Neural Radiance Fields (In-NeRF) for generalizable 3D scene representation and rendering. In contrast to previous image-based rendering, which used two independent working processes of pooling-based fusion and MLP-based rendering, our framework unifies source-view fusion and target-view rendering processes via an end-to-end interpretable Transformer-based network. In-NeRF enables the investigation of deep relationships between the target-rendering view and source views that were previously neglected by pooling-based fusion and fragmented rendering procedures. As a result, InNeRF improves model interpretability by enhancing the shape and appearance consistency of a 3D scene in both the surrounding view space and the ray-cast space. For a query rendering 3D point, InNeRF integrates both its projected 2D pixels from the surrounding source views and its adjacent 3D points along the query ray and simultaneously decodes this information into the query 3D point representation. Experiments show that InNeRF outperforms state-of-the-art image-based neural rendering methods in both scene-agnostic and per-scene finetuning scenarios, especially when there is a considerable disparity between source views and rendering views. The interpretation experiment shows that InNeRF can explain a query rendering process.

1. Introduction

Novel view synthesis is a long-standing open problem concerned with the rendering of unseen views of a 3D scene given a set of observed views [21, 26]. Recent remarkable NeRF research [15, 16, 19, 23, 36] introduces neural radiance field scene representations, which use multi-layer perceptrons (MLPs) to map a continuous 3D location and view direction to its density and color.

However, these models need to optimize a specific 3D representation for each scene, which is time-consuming and does not learn the shared information among scenes. Sub-sequently, to learn prior knowledge in diverse scenes, researchers [6, 28, 31, 35] generalize the radiance field scene representation by incorporating a pooling-based multi-view feature as the conditional input. These prior NeRFs generally contain three basic components: a CNN-based single-view feature extraction module, a pooling-based multi-view fusion module, and an MLP-based NeRF module.

Despite the intrinsic connection between these modules, each module is designed and studied independently, making the overall framework disjointed. This incoherent framework design damages the model interpretability from three aspects: 1) Separating feature extraction of each source view overlooks their relevancy in representing 3D scenes. 2) Pooling-based fusion cannot fully explore the complicated relationship among source views. 3) The MLP network rendering the color and density from a single aggregated feature struggles to decode intricate relationships between observed views and the rendering view. The reason for this framework design is that previous NeRFs are built on MLPs that are incapable of processing an arbitrary number of observed views. Consequently, they need an auxiliary fusion model to aggregate multi-view information, and pooling-based fusion provides such a straightforward technique.

This limitation also impairs the capability of NeRFs to learn a view-consistent 3D scene representation from observed views, especially for the scenario where source views have a more complicated relationship with the target view, e.g. the observed source views are captured at camera poses that are very different from the camera pose of the target view. When camera poses of source views are similar to the rendering view, source views and the target view are distributed in a local region in 3D scene representation space, making it possible to approximate their relationship by a linear function as in previous work [6, 28, 31, 35]. However, as the difference between observed views and the rendering view increases, the correlation becomes more complicated, making it challenging for these approaches to synthesize a realistic novel view. In this scenario, existing MLP-based NeRFs, using a pooling-based function to fuse the multi-view, are insufficient to tackle this challenge.

Therefore, the fundamental issue is how to free the intrinsic interpretability of NeRFs from the previously fragmented frameworks for learning generalizable radiance fields. Meanwhile, previous NeRFs normally learn volume density and radiance of each 3D point independently, ignoring the shape and appearance consistency in a 3D scene. To tackle these unmet needs, we present Interpretable Neural Radiance Fields (InNeRF), an end-to-end Transformer-based architecture that unifies source-view fusion and target-view rendering processes for generalizable 3D scene representation and rendering. In this framework, to enhance local consistency, we design ray-cast space decoders utilizing attention layers to decode a 3D point's latent representation by considering its neighboring 3D points, improving the rendering of color and density.

In the rendering process of a query 3D point, InNeRF is divided into two stages: the first works in the surrounding-view space, integrating information of the projected 2D pixels at the surrounding source views for the query 3D point; and the second works in the ray-cast space, fusing the neighboring 3D points along the query ray into the representation of the query 3D point, as shown in Fig 1. This design provides our model with a comprehensive understanding of the shape and appearance consistency of a 3D scene in both the surrounding-view space and ray-cast spaces. Furthermore, the Transformer-based framework taking advantage of the attention mechanism enables our rendering process to learn in-depth and complicated relationships between source views and the rendering view, which is essential for novel view synthesis. Therefore, InNeRF has improved interpretability and learns a more comprehensive general neural radiance field.

Our contributions can be summarized as follows:
- We propose Interpretable Neural Radiance Fields (InNeRF), a unified Transformer-based framework, to learn deep correlations between observed and rendering views and simultaneously integrate this intricate information into a generalizable neural radiance field.
- InNeRF exploits geometry and appearance consistency of a neural radiance field in both the surrounding view space and the ray-cast space, strengthening its interpretability.
- Experiments show that InNeRF achieves more realistic rendering results than state-of-the-art methods in both scene-agnostic and per-scene fine-tuning settings, especially when source views are captured at camera poses that differ significantly from the rendering view.
- InNeRF explains a query rendering process by utilizing its attention layers. Experiments show that the interpretation of InNeRF is consistent with human perception.

2. Related Work

Novel View Synthesis. The goal of novel view synthesis is to render unseen views of a scene from its multiple observed images. The essence of novel view synthesis is exploring and learning a view-consistent 3D scene representation from a sparse set of input views. The early work focused on modeling 3D shapes by discrete geometric 3D representations, such as mesh surface [10, 11, 22], point cloud [14, 24] and voxel grid [1, 30, 34]. Although explicit 3D geometry-based representations are intuitive, they are discrete and sparse, making them incapable of learning high-resolution renderings with sufficient quality for complex scenes.

More recently, the impressive neural radiance field (NeRF) [21] has shown a solid ability to synthesize novel views by representing continuous scenes as 5D radiance fields in MLPs. Nevertheless, NeRF optimizes each scene representation independently, not exploring the shared information amongst scenes and being time-consuming. Subsequently, researchers proposed models [2, 5, 6, 8, 12, 31, 35], which receive as conditional inputs multiple observed views to learn a general neural radiance field. These methods, such as PixelNeRF [35], MVSNeRF [6], IBRNet [31], are proposed using the divide-and-conquer strategy and have two separate components: a CNN feature extractor for each observed image and an MLP as the NeRF network. However, pooling-based fusion models in these methods barely explore the complex relationship across multiple views for 3D scene understanding. Furthermore, processing each 3D point independently ignores the geometry consistency of a 5D radiance field of a scene.

Here, we propose an encoder-decoder Transformer framework, InNeRF, to represent the neural radiance field scene for novel view synthesis. Compared with the pooling-based fusion in previous work, InNeRF can explore deep relationships among multiple views and aggregate multi-view information into the coordinate-based scene representation by the attention mechanism in a unified network. Meanwhile, InNeRF can learn the consistency of shape and appearance in a scene by considering the corresponding information in the surrounding view space and the ray-cast space.

Transformer. Transformer recently emerged as a promising network framework and has achieved impressive performance in natural language processing [3, 25, 33] and computer vision [4, 7, 9, 13, 18, 37]. The main idea behind this approach is to utilize the multi-head self-attention operation to explore the dependence within input tokens and learn a global feature representation. In the object detection task, DETR [4] presents a new framework combining a 2D CNN with a Transformer and predicts object detection in parallel as a sequence of output tokens. In image classification, ViT [9] demonstrates the impressive ability to learn global contexts in Transformer even without using CNN features [29]. In 3D scene understanding, FlatFormer [18] introduces a new window attention mechanism to optimize the computational efficiency and achieve improved performance in reconstruction.
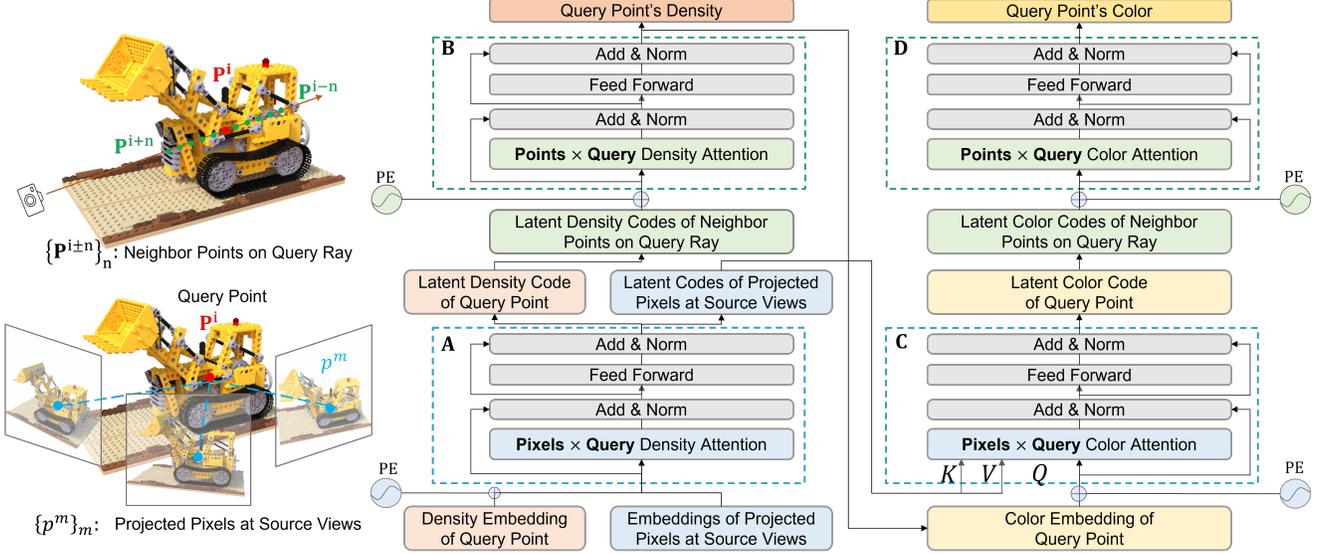
Figure 1. Workflow of the proposed InNeRF. Module **A** is the density decoder in surrounding-view space (Sec. 3.2). Module **B** is the density decoder in ray-cast space (Sec. 3.3). Module **C** is the color decoder in surrounding-view space (Sec. 3.4). Module **D** is the color decoder in ray-cast space (Sec. 3.5).

For novel view synthesis, we introduce an end-to-end Transformer framework to implicitly model the continuous 3D scene as a neural radiance field representation. Our model leverages the advantage of Transformer in exploring deep relationships among observed images to learn a consistent generalizable 3D scene representation.

3. Methodology

3.1. Framework

We propose InNeRF to learn an interpretable generic radiance field representation for novel scenes. Given captured multi-view images $\{\mathbf{I}^m\}_{m=1}^M$ (M source views) of diverse scenes and their camera parameters $\{\mathbf{\Theta}^m\}_{m=1}^M$ (camera poses, intrinsic parameters and scene bounds), InNeRF reconstructs a generic radiance field F_{InNeRF} to learn the prior knowledge:

$$(\sigma, \mathbf{c}) \leftarrow F_{\text{InNeRF}}((x, y, z), \mathbf{d}; \{\mathbf{I}^m, \mathbf{\Theta}^m\}_m) , \quad (1)$$

where (x, y, z) is a 3D point location, $\mathbf{d}$ denotes a unit-length direction of a viewing ray and outputs are a differential volumetric density σ and a directional emitted color $\mathbf{c}$.

As shown in Fig. 1, for rendering a query 3D point on a target-viewing ray, the proposed InNeRF includes two stages: 1) In the surrounding-view space, our $\text{Decoder}_\sigma^{views}$ (Sec. 3.2) and Decoder_c^{views} (Sec. 3.4) fuse source views and query spatial information ((x, y, z), $\mathbf{d}$) into the latent density and color representations for the query point; 2) In the ray-cast space, we use $\text{Decoder}_\sigma^{ray}$ (Sec. 3.3) and Decoder_c^{ray} (Sec. 3.5) to enhance the query density and color

representations by considering neighboring points along the target ray. Finally, we obtain the density and color for the query point on a target-viewing ray.

3.2. Density Decoder in Surrounding-view Space

We first present our density decoder in surrounding-view space ($\text{Decoder}_\sigma^{views}$) decoding the projected pixels at source views into the query latent density code.

For each source view, we first extract its feature volume by a pre-trained view-shared U-Net. A query 3D point (x, y, z) is then projected into each source view $\mathbf{I}^m$ by its camera projection matrix $\mathbf{\Theta}^m$ to extract the corresponding RGB color $\{\mathbf{c}_{src}^m\}_{m=1}^M$ and feature vector $\{\mathbf{f}_{src}^m\}_{m=1}^M$ at the projected 2D pixel $\{\mathbf{p}^m\}_{m=1}^M$ location through bilinear interpolation. In each source view, we also record its viewing direction $\{\mathbf{d}_{src}^m\}_{m=1}^M$ for the projected pixel from the source camera pose. Based on it, we obtain the initial source-view embeddings $\{\mathbf{x}_0^m\}_{m=1}^M$ for source views.

For the query point, $\text{Decoder}_\sigma^{views}$ receives the initial source-view embeddings $\{\mathbf{x}_0^m\}_{m=1}^M$ and the learnable query density embedding $\mathbf{x}_0^\sigma$ as inputs $\mathbf{X}_0$.

$\text{Decoder}_\sigma^{views}$ can be formulated as follows:

$$\mathbf{X}_0 = [\mathbf{x}_0^\sigma; \mathbf{x}_0^1; \mathbf{x}_0^2; \cdots ; \mathbf{x}_0^M] , \quad (2)$$

$$\tilde{\mathbf{X}}_{l+1} = \text{Norm}(\text{Pixels} \times \text{Query}_\sigma(\mathbf{X}_l) + \mathbf{X}_l) , \quad (3)$$

$$\mathbf{X}_{l+1} = \text{Norm}(\text{FFN}(\tilde{\mathbf{X}}_{l+1}) + \tilde{\mathbf{X}}_{l+1}) , \quad (4)$$

where l denotes the index of a basic block ($l = 1, \cdots, L$), "Norm" is a layer normalization function, and "FFN" is a position-wise feed-forward network. At the L-th block, we

can obtain $\mathbf{X}_L = [\mathbf{x}_L^\sigma; \mathbf{x}_L^1; \mathbf{x}_L^2; \cdots; \mathbf{x}_L^M]$. In Decoder$_\sigma^{views}$, we concatenate the embedding $\mathbf{x}_L^\sigma$ and its 3D coordinate location (x, y, z) as the latent density code for the query point.

Pixels×Query Density Attention layers explore deep relationships among source views, defined as follows:

$$\text{Pixels×Query}_\sigma(\mathbf{X}) = \text{MH-Attn}(\mathbf{X}, \mathbf{X}, \mathbf{X}) , \quad (5)$$

where the multi-head attention function is defined as:

$$\text{MH-Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Cat}(\mathbf{A}_1, \cdots, \mathbf{A}_H)\mathbf{W} , \quad (6)$$
$$\text{where } \mathbf{A}_h = \text{Attention}(\mathbf{Q}_h, \mathbf{K}_h, \mathbf{V}_h) ,$$
$$\mathbf{Q}_h = \mathbf{Q}\mathbf{W}_h^q; \mathbf{K}_h = \mathbf{K}\mathbf{W}_h^k; \mathbf{V}_h = \mathbf{V}\mathbf{W}_h^v .$$

Here, $\mathbf{W}_h^q, \mathbf{W}_h^k \in \mathbb{R}^{d_k \times d_h}; \mathbf{W}_h^v \in \mathbb{R}^{d_v \times d_h}$ and $\mathbf{W} \in \mathbb{R}^{Hd_h \times d_k}$ are parameter matrices ($H \times d_h = d_k$ and d_h is the feature dimension in each head). The Attention function is computed by

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V} , \quad (7)$$

Here, N_q queries are stacked in $\mathbf{Q} = [\mathbf{q}_1; \mathbf{q}_2; \cdots; \mathbf{q}_{N_q}] \in \mathbb{R}^{N_q \times d_k}$, a set of N_k key-value pairs are stacked in $\mathbf{K} = [\mathbf{k}_1; \mathbf{k}_2; \cdots; ; \mathbf{k}_{N_k}] \in \mathbb{R}^{N_k \times d_k}$ and $\mathbf{V} = [\mathbf{v}_1; \mathbf{v}_2; \cdots; \mathbf{v}_{N_k}] \in \mathbb{R}^{N_k \times d_v}$, d_k is used as a scalar for normalization. Our Decoder$_\sigma^{views}$ is invariant to permutations of source views and can receive an arbitrary number of source views.

3.3. Density Decoder in Ray-cast Space

The density decoder in ray-cast space (Decoder$_\sigma^{ray}$) decodes the density information of the query point by aggregating the density features of the neighboring 3D points along the target-view ray.

For the query point and neighboring $2n$ points along the target-viewing ray, we denote $[\sigma_0^{i-n}; \cdots \sigma_0^i \cdots; \sigma_0^{i+n}]$ as their initial density representations at the input end of the Decoder$_\sigma^{ray}$, where the query point is denoted as P^i and neighboring $2n$ points are $\{P^{i-n}, \cdots, P^{i-1}, P^{i+1}, \cdots, P^{i+n}\}$. Here, the initial density representation for each 3D point is computed via an FC layer based on the Decoder$_\sigma^{views}$ output for the corresponding point ($\sigma_0 = \text{FC}(\mathbf{x}_L^\sigma \odot (x, y, z))$, where $\odot$ is the concatenation operation). Then positional encodings $\mathbf{E}^{pos}$ are added to density representations of neighboring points to keep their position information in the ray-cast space. Each positional encoding informs each point of its 3D spatial location, which is computed by utilizing sine and cosine functions of different frequencies as [4].

Decoder$_\sigma^{ray}$ is formulated as follows:

$$\mathbf{D}_0 = [\sigma_0^{i-n}; \cdots \sigma_0^i \cdots; \sigma_0^{i+n}] + \mathbf{E}^{pos} , \quad (8)$$
$$\tilde{\mathbf{D}}_{l+1} = \text{Norm}(\text{Points×Query}_\sigma(\mathbf{D}_l) + \mathbf{D}_l) , \quad (9)$$
$$\mathbf{D}_{l+1} = \text{Norm}(\text{FFN}(\tilde{\mathbf{D}}_{l+1}) + \tilde{\mathbf{D}}_{l+1}) , \quad (10)$$

where the Points×Query Density Attention layer is computed as Points×Query$_\sigma$ = MH-Attn($\mathbf{D}, \mathbf{D}, \mathbf{D}$) fusing information of surrounding 3D points on the target-viewing ray. At the end block, the Decoder$_\sigma^{ray}$ outputs the density representation σ_L^i of the query 3D point, and then we use an FC layer to project it to the density value.

3.4. Color Decoder in Surrounding-view Space

The color decoder in surrounding-view space (Decoder$_c^{views}$) decodes the projected pixels' information from source views into the query color representation.

Decoder$_c^{views}$ can be formulated as follows:

$$\tilde{\mathbf{Y}}_{l+1} = \text{Norm}(\text{Pixels×Query}_c(\mathbf{Y}_l, \hat{\mathbf{X}}, \hat{\mathbf{C}}) + \mathbf{Y}_l) , \quad (11)$$
$$\mathbf{Y}_{l+1} = \text{Norm}(\text{FFN}(\tilde{\mathbf{Y}}_{l+1}) + \tilde{\mathbf{Y}}_{l+1}) . \quad (12)$$

In Pixels×Query Color Attention layers, the initial query color embedding is represented as $\mathbf{Y}_0 = \text{FC}(\sigma_L^i) \odot \mathbf{d}_{tgt}$, where σ_L^i is the latent density representation from Decoder$_\sigma^{ray}$ and $\mathbf{d}_{tgt}$ is the target-viewing direction for the query point. Pixels×Query Color Attention layer is calculated as:

$$\text{Pixels×Query}_c(\mathbf{Y}, \hat{\mathbf{X}}, \hat{\mathbf{C}}) = \text{MH-Attn}(\mathbf{Y}, \hat{\mathbf{X}}, \hat{\mathbf{C}}) , \quad (13)$$

where the value is $\hat{\mathbf{C}} = [\gamma(\mathbf{c}_{src}^1); \cdots; \gamma(\mathbf{c}_{src}^M)]$ ($\gamma(\cdot)$ is the embedding function) and the key is $\hat{\mathbf{X}} = [\text{FC}(\mathbf{x}_L^1) \odot \mathbf{d}_{src}^1; \cdots; \text{FC}(\mathbf{x}_L^M) \odot \mathbf{d}_{src}^M]$ representing the projected pixels' representations in source views. The output $\mathbf{Y}_L$ is the latent color code for the query 3D point.

3.5. Color Decoder in Ray-cast Space

The color decoder in ray-cast space (Decoder$_c^{ray}$) learns a query color by fusing latent color codes of adjacent 3D points along the target ray in Points×Query Color Attention layers (Points×Query$_c(\mathbf{Z})$ = MH-Attn($\mathbf{Z}, \mathbf{Z}, \mathbf{Z}$)). By incorporating neighboring points, the submodule learns more local-consistent representations for the target point. Decoder$_c^{ray}$ is represented as:

$$\mathbf{Z}_0 = [\mathbf{z}_0^{i-n}; \cdots \mathbf{z}_0^i \cdots; \mathbf{z}_0^{i+n}] + \mathbf{E}^{pos} , \quad (14)$$
$$\tilde{\mathbf{Z}}_{l+1} = \text{Norm}(\text{Points×Query}_c(\mathbf{Z}_l) + \mathbf{Z}_l) , \quad (15)$$
$$\mathbf{Z}_{l+1} = \text{Norm}(\text{FFN}(\tilde{\mathbf{Z}}_{l+1}) + \tilde{\mathbf{Z}}_{l+1}) . \quad (16)$$

where the query latent color code from Decoder$_c^{views}$ is assigned to the corresponding $\mathbf{z}_0^i$ and likewise for adjacent $2n$ points in ray-cast space.

Table 1. Quantitative comparison of methods on the scene-agnostic setting for the realistic synthetic dataset [21] and the real forward-facing dataset [20].

Dataset	S_i	PSNR ↑						SSIM ↑						LPIPS ↓					
		PixelNeRF	MVSNeRF	IBRNet	GeFu	GNT	**InNeRF**	PixelNeRF	MVSNeRF	IBRNet	GeFu	GNT	**InNeRF**	PixelNeRF	MVSNeRF	IBRNet	GeFu	GNT	**InNeRF**
realistic synthetic	S1	21.20	22.47	25.31	25.16	24.64	**26.45**	0.857	0.874	0.913	0.914	0.906	**0.922**	0.161	0.143	0.104	0.094	**0.088**	0.092
	S2	17.00	18.44	21.80	18.57	12.74	**23.16**	0.732	0.755	0.805	0.806	0.637	**0.842**	0.295	0.286	0.236	0.292	0.396	**0.183**
	S3	15.88	17.43	20.99	16.66	10.66	**22.70**	0.660	0.687	0.749	0.780	0.583	**0.810**	0.355	0.328	0.270	0.327	0.486	**0.211**
	S4	14.67	16.25	19.97	13.30	9.21	**21.72**	0.567	0.687	0.672	0.728	0.564	**0.758**	0.440	0.400	0.322	0.355	0.517	**0.248**
real forward-facing	S1	19.02	20.09	24.96	24.87	24.79	**24.97**	0.651	0.680	0.813	0.807	0.808	**0.816**	0.380	0.347	0.208	0.217	0.213	**0.205**
	S2	16.30	17.68	22.69	20.33	20.01	**22.94**	0.576	0.614	0.749	0.680	0.667	**0.760**	0.459	0.422	0.273	0.294	0.277	**0.260**
	S3	13.56	15.21	20.33	17.04	17.44	**20.81**	0.489	0.543	0.683	0.662	0.534	**0.701**	0.551	0.504	0.340	0.426	0.379	**0.318**

Subsequently, after the Decoder_c^{ray}, we use an FC layer to project the output color embedding $\mathbf{z}_L^i$ to its output predicted color value. Then the predicted density and color of each query point along a ray of the desired virtual camera are put forward to the classical volume rendering. The implementation details of the network and training are described in the supplementary material.

4. Experiments

The proposed approach is evaluated in the following experimental settings:

- Scene-agnostic setting: we train a single scene-agnostic model on a large training dataset that includes various camera setups and scene types. We test its generalization ability to unseen scenes on all test scenes.
- Per-scene fine-tuning setting: our pretrained scene-agnostic model is finetuned on each test scene. We evaluate each finetuned scene-specific model separately.

We train and evaluate our method on a collection of multi-view datasets containing both synthetic data and real data, as in IBRNet [31]. For novel view synthesis, we quantitatively evaluate the rendered image quality based on PSNR, SSIM [32] (higher is better), and LPIPS [38] (lower is better).

4.1. Conditional Source-view Set

Experiments are designed to examine whether the proposed InNeRF can effectively learn a neural radiance field scene representation in scenarios where the variation degree between the conditional source view set and the target rendering view changes. Here, we sample 10 views from the surrounding view set as the conditional source-view set to render a target view. Given the camera pose, we can compute and sort the difference between each surrounding view and the target rendering view.

Based on the sorted order, we construct N_s conditional source-view sets ($\{\mathbf{S}_i\}_{i=1}^{N_s}$) from the surrounding-view set to render each test view. For the real evaluation dataset, there are $N_s = 3$ sets, i.e. top 10 ($\mathbf{S}_1$), middle 10 ($\mathbf{S}_2$), and bottom 10 ($\mathbf{S}_3$) views. For the synthetic evaluation dataset, there are $N_s = 4$ sets which are the top 10 ($\mathbf{S}_1$), middle 10 ($\mathbf{S}_2$), 3/4th 10 ($\mathbf{S}_3$), and bottom 10 ($\mathbf{S}_4$) views, respectively. Fig. 4 shows visual examples of $\mathbf{S}_1$ and $\mathbf{S}_4$ for illustration.

4.2. Results

In both the scene-agnostic (Sec. 4.2.1) and per-scene fine-tuning experiments (Sec. 4.2.2), we evaluate competing methods in scenarios where the source views belong to different source view sets $\{\mathbf{S}_i\}_{i=1}^{N_s}$ defined in Sec. 4.1. To render a testing view, each competing approach receives as input the same source-view set. We also conduct ablation study in Sec. 4.2.3 to analyze the influence of submodules to the model performance. In Sec. 4.2.4, we provide the interpretation results of InNeRF.

4.2.1 Scene-agnostic Experiments

In scene-agnostic experiments, InNeRF is compared with PixelNeRF [35], MVSNeRF [6], IBRNet [31], GeFu [17], and GNT [27] on the real forward-facing dataset [20] and the realistic synthetic dataset [21].

Tab. 1 shows that the proposed InNeRF outperforms other methods on both datasets under the scene-agnostic setting. To facilitate the quantitative comparison in each metric, the best scores are marked in bold. It shows that InNeRF has a better generalization ability to novel scenes though it is trained on datasets with noticeably different scenes and view distribution. The detailed results in the supplementary material also reveal that InNeRF has a better performance for each scene.

The superior generalization ability of InNeRF is also reflected in qualitative results. As shown in Fig. 2, we compare the performance of methods on rendering the same randomly-selected testing view based on different source-view sets. The results of other approaches contain more obvious artifacts than InNeRF and even become worse in the $\mathbf{S}_3$ scenario where the difference between source views and the target view is larger than that in $\mathbf{S}_1$ and $\mathbf{S}_2$. As highlighted in colored frames, other methods cannot synthesize clean boundaries of guardrails and fronds and recover thin structures. For simplicity, we only plot color frames in PixelNeRF's results on $\mathbf{S}_1$. In the results of each method, the first column shows the rendered region for the red-framed area in the ground-truth target view, and the second column shows the rendered region for the orange-framed area.

From the above qualitative results, we observe that there exists a gradual degradation in the synthesized view when the difference between source views and the target rendering
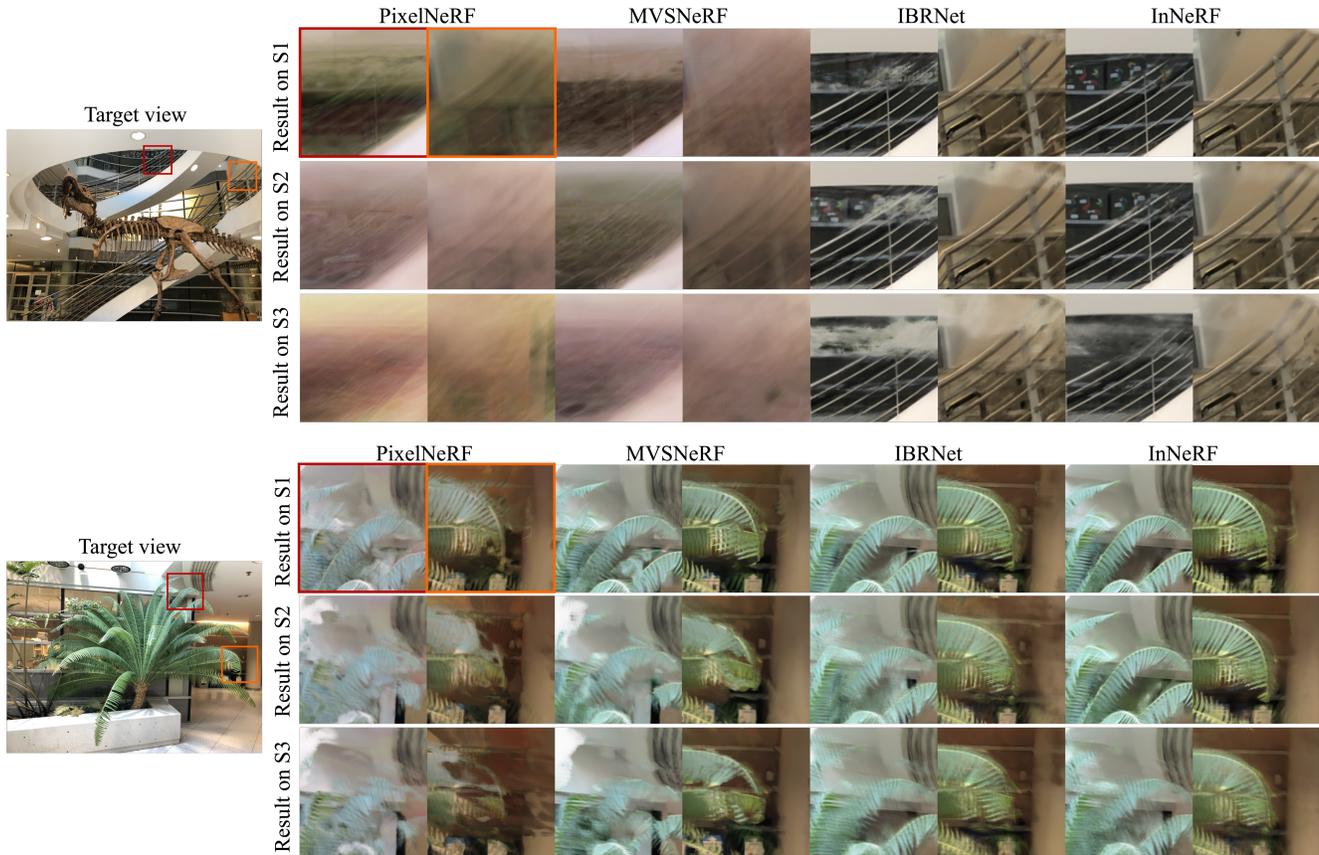
Figure 2. Qualitative results for the Trex and the Fern scenes [20] under the scene-agnostic setting.

Table 2. Quantitative comparisons of methods on the per-scene fine-tuning setting for the realistic synthetic dataset [21] and the real forward-facing dataset [20].

Dataset	$\mathbf{S}_i$	PSNR ↑				SSIM ↑				LPIPS ↓			
		PixelNeRF	MVSNeRF	IBRNet	InNeRF	PixelNeRF	MVSNeRF	IBRNet	InNeRF	PixelNeRF	MVSNeRF	IBRNet	InNeRF
realistic synthetic	S1	24.06	27.04	29.27	**30.79**	0.877	0.913	0.940	**0.952**	0.140	0.103	0.076	**0.064**
	S2	20.15	23.30	25.91	**27.76**	0.770	0.813	0.847	**0.881**	0.263	0.221	0.187	**0.142**
	S3	19.27	22.56	25.23	**27.35**	0.714	0.759	0.802	**0.849**	0.301	0.256	0.216	**0.165**
	S4	18.23	21.57	24.33	**26.65**	0.639	0.689	0.739	**0.803**	0.358	0.306	0.254	**0.195**
real forward-facing	S1	20.72	23.32	26.61	**26.65**	0.693	0.758	0.847	**0.853**	0.325	0.260	0.177	**0.173**
	S2	18.28	21.11	24.69	**24.99**	0.625	0.696	0.788	**0.811**	0.384	0.313	0.225	**0.212**
	S3	15.66	18.62	22.62	**23.25**	0.544	0.623	0.727	**0.767**	0.458	0.377	0.276	**0.256**

view increases from $\mathbf{S}_1$ to $\mathbf{S}_3$. Similarly, in quantitative results from $\mathbf{S}_1$ to $\mathbf{S}_3$, PSNR and SSIM values both decrease while LPIPS increases for all competing methods. It reveals that the more different the source views are with respect to the target rendering view, the more difficult novel view synthesis becomes. Tab. 1 also indicates that the advantage of InNeRF becomes more significant than other methods with the increase of the difference between source views and the target view. It demonstrates that InNeRF has a strong ability to explore complicated relationships between source views and the target view and learn a better scene representation in challenging scenarios. More results are provided in the supplementary material.

4.2.2 Per-scene Finetuning Experiments

In the per-scene finetuning experiment, pretrained models of competing methods are finetuned for each scene.

As shown in Tab. 2, InNeRF outperforms other methods after per-scene finetuning. Similar to scene-agnostic results, per-scene finetuning results further validate that InNeRF can provide more satisfactory novel view rendering than other methods in different source-view settings. Meanwhile, performance gaps between InNeRF and other methods become
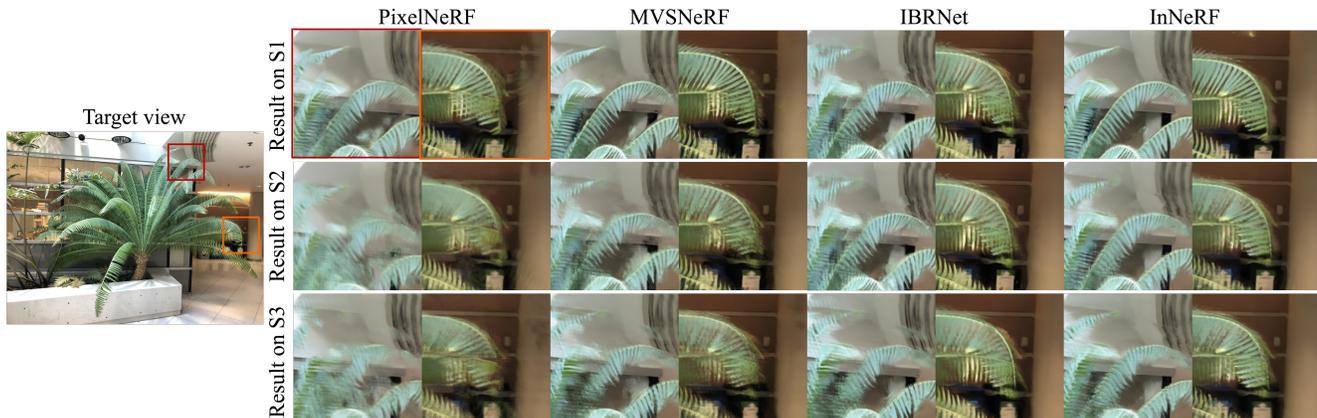
Figure 3. Qualitative results for the Fern scene [20] under the per-scene finetuning setting.
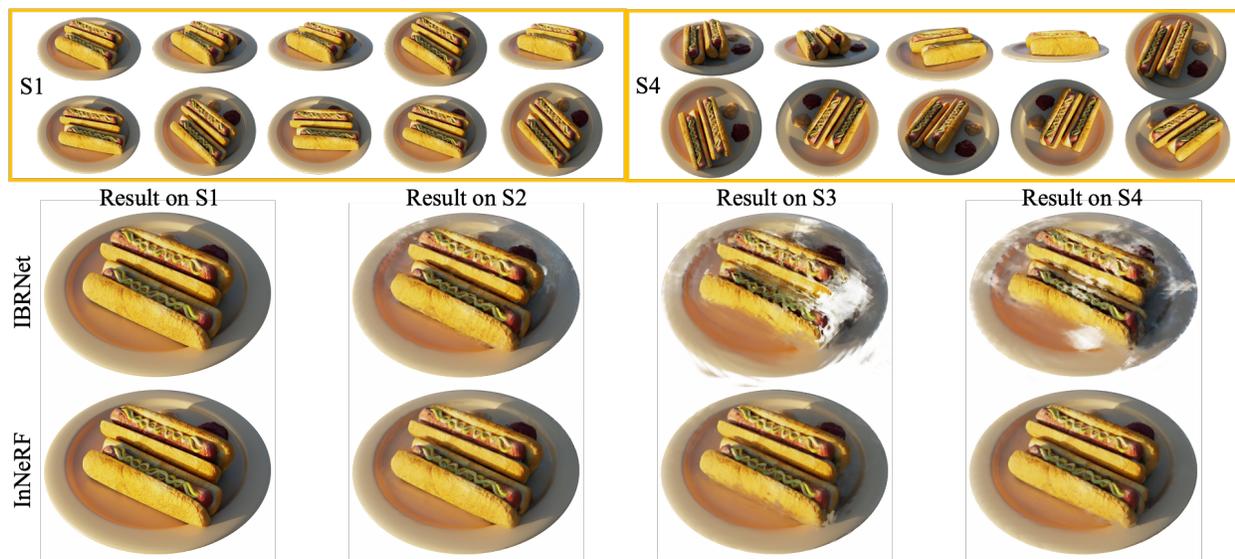

Figure 4. Qualitative results for the Hotdog scene under the per-scene finetuning setting. The source-view sets S1 and S4 are listed in the yellow frame.

larger in contrast with that in the scene-agnostic setting, which indicates that per-scene finetuning can further fulfill the potential of InNeRF. Similar to quantitative results, Fig. 3 shows InNeRF provides more realistic view synthesis results with fewer artifacts in comparison to other approaches.

In Fig. 4, InNeRF is compared with IBRNet in four source-view sets ($\mathbf{S}_1$, $\mathbf{S}_2$, $\mathbf{S}_3$ and $\mathbf{S}_4$). Here, we randomly select one view from the Hotdog scene as the target rendering view. To show the difference between source view sets, we display overall source views in $\mathbf{S}_1$ and $\mathbf{S}_4$ at the bottom of Fig. 4. It is obvious that the view angles of source views in $\mathbf{S}_1$ are closer to the rendering view compared with those in $\mathbf{S}_4$. In Fig. 4, the top two rows display the rendering views of competing methods based on four source-view sets. The artifacts in the rendering views of IBRNet are perceptible

in $\mathbf{S}_2$ and become worse in $\mathbf{S}_3$ and $\mathbf{S}_4$. In contrast, the artifacts in rendering views of InNeRF remain at a low degree in four source-view sets. It illustrates InNeRF can obtain better rendering results than IBRNet in different source-view sets, especially when there is a large difference between the source views and the rendering view.

4.2.3 Ablation Study

We conduct ablation study on the proposed components to highlight their crucial roles. Specifically, we examine the influence of our density $\text{Decoder}_\sigma^{ray}$ and color Decoder_c^{ray} in the ray-cast space on performance at the real forward-facing dataset. The results in Tab. 3 demonstrate a notable improvement with each submodule, indicating their effectiveness
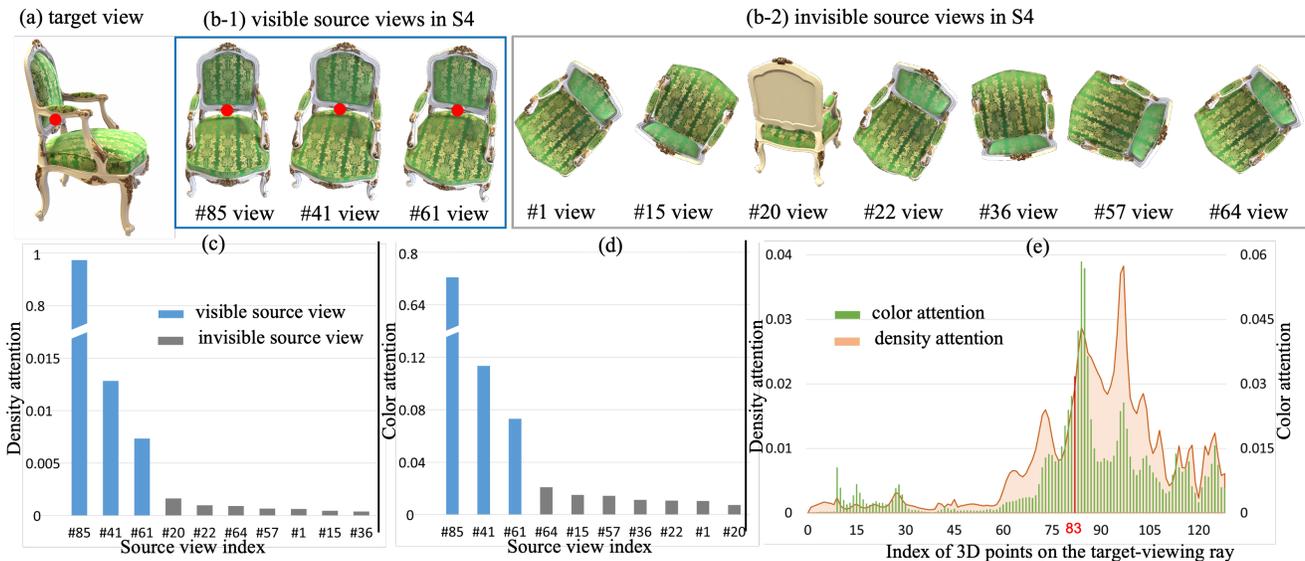
Figure 5. Interpretation results of finetuned InNeRF for a target view of Chair scene based on source-view set S4.
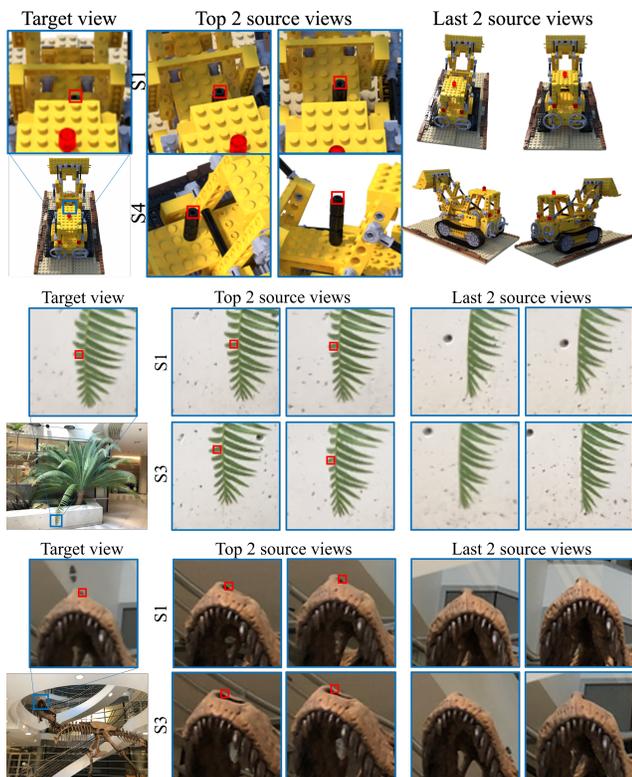

Figure 6. Interpretation results of fine-tuned InNeRF for (top) the black tile in Lego, (middle) the leaves on the left side in Fern, and (bottom) the wings of the nose spot in Trex scene.

in enhancing the model's ability to learn a consistent 3D representation.

4.2.4 Analysis of Interpretability in InNeRF

Based on the attention mechanism, InNeRF utilizes shape and appearance consistency in both the surrounding-view space and the ray-cast space, thus improving the model interpretability. Here, we evaluate the interpretability of InNeRF to examine whether it is consistent with human perception.

In the surrounding-view space, we visualize the attention of different source views to a target 3D point to interpret its rendering in $\text{Decoder}_\sigma^{views}$ and Decoder_c^{views}. Similarly, in the ray-cast space, the rendering process of $\text{Decoder}_\sigma^{ray}$ and Decoder_c^{ray} can be explored by visualizing the attention of surrounding 3D points on the target-viewing ray to the target 3D point. Specifically, for a 2D region (a 5×5 pixel region) in the rendering view, we first compute the average depth value of the corresponding view directions for the target pixels based on our learned neural radiance field. Then we retrieve the 3D point that is located closest to the average depth in the average viewing direction as the target-interpreted 3D point. For the target 3D point, we can explain its rendering process in both surrounding-view and ray-cast spaces by visualizing the corresponding attention layers in InNeRF.

To analyze the interpretability of InNeRF, we provide interpretation to a randomly selected testing view of the Chair scene based on source-view sets $\mathbf{S}_4$ in Fig. 5. The target rendering view is shown in Fig. 5 (a) and the target location for interpretation is marked as a red dot. For human visual perception, the source views are divided into two groups depending on whether they capture the target location (red dot) in the rendering view. Fig. 5 (b-1) shows source views that capture the target location, and Fig. 5 (b-2) shows source views that fail to capture.

8

Table 3. Ablation study of components in InNeRF.

$\text{Decoder}_\sigma^{ray}$	Decoder_c^{ray}	$\text{Decoder}_{\sigma/c}^{views}$	PSNR↑	SSIM↑	LPIPS↓
×	×	✓	19.42	0.742	0.524
✓	×	✓	22.30	0.791	0.382
×	✓	✓	23.95	0.806	0.271
✓	✓	✓	**24.97**	**0.816**	**0.205**

For the target location (red dot) in Fig. 5 (a), Fig. 5 (c) and (d) display attention of source views to the target location for rendering the query density and color in $\text{Decoder}_\sigma^{views}$ and Decoder_c^{views}, respectively. In Fig. 5 (c) and (d), attention of the visible source views in Fig. 5 (b) are colored blue for clarity. Source views (85, 41, and 61) with high attention values are consistent with those with the visible target location. It indicates that the attention layers in $\text{Decoder}_\sigma^{views}$ and Decoder_c^{views} can learn the important source views that meet human perception. Fig. 5 (e) depicts the density attention (green) and color attention (orange) among 3D points along the target-viewing ray for rendering the query 3D point in $\text{Decoder}_\sigma^{ray}$ and Decoder_c^{ray}. Here, the red index (83) denotes the retrieved 3D point for the target location in the rendering view. As shown in Fig. 5 (e), both density attention and color attention in $\text{Decoder}_\sigma^{ray}$ and Decoder_c^{ray} exist a crest near the query 3D point, which illustrates that InNeRF in the ray-cast space takes into account the consistency of neighbor points when rendering the query point. Since density reflects geometry and color represents appearance, their consistency patterns may differ. Therefore, points of high relevance for density and color may vary slightly. Overall, wave peaks near the query point reflect the consistency between neighboring points and the query point.

Fig. 6 provides the interpretation results for the nearest and furthest source view sets for Lego, Fern, and Trex scenes. The leftmost column shows the target rendering view and an enlarged region framed by a blue box. The small red box in the rendering view illustratively shows the target location. The second and third columns show the top two source views that are of high-density attention for the target location, and the last two columns show the last two source views with low-density attention. For the target location, the high-attention source views, as shown in the second and third columns, can capture the interested location. As shown in the middle two rows of Fig. 6, for source view set $\mathbf{S}_1$ and $\mathbf{S}_3$, the top two source views of the framed leaf region both include the corresponding leaf region while the last two source views do not. This observation also applies to (top) the black tile in Lego and (bottom) the wings of the nose in Trex. Given that the top-attention source views capture the target location (red box), it is reasonable that they receive more attention for the query rendering.

5. Conclusion

We propose a unified Transformer-based NeRF framework to learn a general neural radiance field for novel view synthe-sis. The proposed framework can explore complex relationships between source views and the target rendering view. Meanwhile, the framework improves intrinsic interpretability by utilizing the shape and appearance consistency of 3D scenes. Experiments demonstrate that InNeRF achieves state-of-the-art performance on real and synthetic datasets in both scene-agnostic and per-scene finetuning settings. In the future, we intend to extend InNeRF to conditional generative radiance fields, employing learned prior knowledge to generate a more expressive and interpretable 3D scene representation.

References

[1] Haotian Bai, Yiqi Lin, Yize Chen, and Lin Wang. 2023. Dynamic PlenOctree for Adaptive Sampling Refinement in Explicit NeRF. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8785–8795.

[2] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. 2023. Zip-NeRF: Anti-Aliased Grid-Based Neural Radiance Fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 19697–19705.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European Conference on Computer Vision*. Springer, Springer International Publishing, Cham, 213–229.

[5] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. 2024. pixelSplat: 3D Gaussian Splats from Image Pairs for Scalable Generalizable 3D Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 19457–19467.

[6] 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.

[7] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. 2020. Generative pretraining from pixels. In *International Conference on Machine Learning*. PMLR, 1691–1703.

[8] Zhang Chen, Zhong Li, Liangchen Song, Lele Chen, Jingyi Yu, Junsong Yuan, and Yi Xu. 2023. NeuRBF: A Neural Fields Representation with Adaptive Radial Basis Functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 4182–4194.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR)*.

[10] Yiming Gao, Yan-Pei Cao, and Ying Shan. 2023. SurfelNeRF: Neural Surfel Radiance Fields for Online Photorealistic Reconstruction of Indoor Scenes. *arXiv preprint arXiv:2304.08971* (2023).

[11] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. 2019. Mesh r-cnn. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9785–9795.

[12] Wenbo Hu, Yuling Wang, Lin Ma, Bangbang Yang, Lin Gao, Xiao Liu, and Yuewen Ma. 2023. Tri-MipRF: Tri-Mip Representation for Efficient Anti-Aliasing Neural Radiance Fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 19774–19783.

[13] L. Jiang, Z. Yang, S. Shi, V. Golyanik, D. Dai, and B. Schiele. 2023. Self-Supervised Pre-Training with Masked Shape Prediction for 3D Scene Understanding. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1168–1178.

[14] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. 2022. Stratified transformer for 3d point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8500–8509.

[15] Gengyan Li, Abhimitra Meka, Franziska Mueller, Marcel C. Buehler, Otmar Hilliges, and Thabo Beeler. 2022. EyeNeRF: A Hybrid Representation for Photorealistic Synthesis, Animation and Relighting of Human Eyes. *ACM Trans. Graph.* 41, 4 (2022).

[16] Zhengqi Li, Qianqian Wang, Forrester Cole, Richard Tucker, and Noah Snavely. 2023. Dynibar: Neural dynamic image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4273–4284.

[17] Tianqi Liu, Xinyi Ye, Min Shi, Zihao Huang, Zhiyu Pan, Zhan Peng, and Zhiguo Cao. 2024. Geometry-aware Reconstruction and Fusion-refined Rendering for Generalizable Neural Radiance Fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7654–7663.

[18] Zhijian Liu, Xinyu Yang, Haotian Tang, Shang Yang, and Song Han. 2023. FlatFormer: Flattened Window Attention for Efficient Point Cloud Transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[19] Xiaoxu Meng, Weikai Chen, and Bo Yang. 2023. NeAT: Learning Neural Implicit Surfaces with Arbitrary Topologies from Multi-view Images. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (June 2023).

[20] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. 2019. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. *ACM Transactions on Graphics (TOG)* (2019).

[21] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*. Springer, 405–421.

[22] Charlie Nash, Yaroslav Ganin, SM Ali Eslami, and Peter Battaglia. 2020. Polygen: An autoregressive generative model of 3d meshes. In *International Conference on Machine Learning*. PMLR, 7220–7229.

[23] Thu Nguyen-Phuoc, Feng Liu, and Lei Xiao. 2022. SNeRF: Stylized Neural Implicit Representations for 3D Scenes. *ACM Trans. Graph.* 41, 4 (2022).

[24] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 652–660.

[25] Ari Seff, Brian Cera, Dian Chen, Mason Ng, Aurick Zhou, Nigamaa Nayakanti, Khaled S. Refaat, Rami Al-Rfou, and Benjamin Sapp. 2023. MotionLM: Multi-Agent Motion Forecasting as Language Modeling. arXiv:2309.16534 [cs.CV]

[26] Qing Shuai, Chen Geng, Qi Fang, Sida Peng, Wenhao Shen, Xiaowei Zhou, and Hujun Bao. 2022. Novel View Synthesis of Human Interactions from Sparse Multi-View Videos. In *ACM SIGGRAPH 2022 Conference Proceedings* (Vancouver, BC, Canada) *(SIGGRAPH '22)*. Association for Computing Machinery, New York, NY, USA.

[27] Mukund Varma T, Peihao Wang, Xuxi Chen, Tianlong Chen, Subhashini Venugopalan, and Zhangyang Wang. 2023. Is Attention All That NeRF Needs?. In *International Conference on Learning Representations (ICLR)*.

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

[29] Dan Wang, Xinrui Cui, Xun Chen, Rabab Ward, and Z. Jane Wang. 2021. Interpreting Bottom-Up Decision-Making of CNNs via Hierarchical Inference. *IEEE Transactions on Image Processing* 30 (2021), 6701–6714.

[30] Dan Wang, Xinrui Cui, Xun Chen, Zhengxia Zou, Tianyang Shi, Septimiu Salcudean, Z. Jane Wang, and Rabab Ward. 2021. Multi-View 3D Reconstruction With Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 5722–5731.

[31] 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.

[32] 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.

[33] Jilan Xu, Junlin Hou, Yuejie Zhang, Rui Feng, Yi Wang, Yu Qiao, and Weidi Xie. 2023. Learning open-vocabulary semantic segmentation models from natural language supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2935–2944.

[34] Bo Yang, Stefano Rosa, Andrew Markham, Niki Trigoni, and Hongkai Wen. 2018. Dense 3D object reconstruction from a single depth view. *IEEE transactions on pattern analysis and machine intelligence* 41, 12 (2018), 2820–2834.

[35] 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.

[36] Heng Yu, Joel Julin, Zoltan A Milacski, Koichiro Niinuma, and László A Jeni. 2023. DyLiN: Making Light Field Networks Dynamic. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12397–12406.

[37] H. Yu, Z. Qin, J. Hou, M. Saleh, D. Li, B. Busam, and S. Ilic. 2023. Rotation-Invariant Transformer for Point Cloud Matching. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5384–5393.

[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.