Learning Spherical Radiance Field for Efficient 360° Unbounded Novel View Synthesis

Minglin Chen, *Graduate Student Member, IEEE*, Longguang Wang, Yinjie Lei, *Senior Member, IEEE*, Zilong Dong, and Yulan Guo, *Senior Member, IEEE*

***Abstract*— Novel view synthesis aims at rendering any posed images from sparse observations of the scene. Recently, neural radiance fields (NeRF) have demonstrated their effectiveness in synthesizing novel views of a bounded scene. However, most existing methods cannot be directly extended to 360° unbounded scenes where the camera orientations and scene depths are unconstrained with large variations. In this paper, we present a spherical radiance field (SRF) for efficient novel view synthesis in 360° unbounded scenes. Specifically, we represent a 3D scene as multiple concentric spheres with different radii. In particular, each sphere encodes its corresponding layered scene into implicit representations and is parameterized with an equirectangular projection image. A shallow multi-layer perceptron (MLP) is then used to infer the density and color from these sphere representations for volume rendering. Moreover, an occupancy grid is introduced to cache the density field and guide the ray sampling, which accelerates the training and rendering procedures by reducing the number of samples along the ray. Experiments show that our method can well fit 360° unbounded scenes and produces state-of-the-art results on three benchmark datasets with less than 30 minutes of training time on a 3090 GPU, surpassing Mip-NeRF 360 with a 400× speedup. In addition, our method achieves competitive performance in terms of both accuracy and efficiency on a bounded dataset. Project page: https://minglin-chen.github.io/SphericalRF**

***Index Terms*— Novel view synthesis, neural radiance fields, equirectangular projection image.**

I. Introduction

NOVEL view synthesis is a long-standing problem in computer vision and graphics, which aims to render images at arbitrary viewpoints from a set of captured images at some sparse viewpoints. Traditional methods synthesize novel views by formulating the scene as a textured point cloud [1] or mesh based on unstructured multi-view stereo (MVS) [2], [3], [4], [5], or learning a 4D light field function [6], [7], [8], [9], [10] without geometric reconstruction. Recently, neural radiance fields (NeRF) [11] is introduced as a powerful paradigm for novel view synthesis. NeRF combines implicit scene representation [12], [13] with traditional volumetric rendering [14] to achieve photo-realistic novel view synthesis. Later works further advances NeRF by improving synthesized image quality [15], [16], [17], [18], [19], training and rendering speed [20], [21], [22], [23], [24], [25], [26], [27], generalization ability [28], [29], [30], [31], and extending to other research fields, such as large-scale scene reconstruction [32], [33], [34], [35], [36], embodied artificial intelligence [37], [38], [39], and text-conditioned 3D object generation [40], [41].

Despite the great success of NeRF in novel view synthesis, existing NeRF-based approaches [11], [15], [16], [21], [22], [23], [25], [42] mainly focus on bounded scenes. These methods are applied to synthesized small objects (e.g., *lego*) or captured real-world images after background removal using segmentation methods [43], [44], [45], [46], [47], [48], [49], [50]. However, in real-world scenes, acquired images contain not only nearby objects but also their surrounding environments, e.g., distant trees and buildings. As a result, the aforementioned NeRF-based methods suffer from blurry results in distance regions. In essence, the challenges of 360° unbounded scenes are twofold:

(i) Sparse Rays. For NeRF-based methods, the perception of depth relies on the intersection between rays, which is largely affected by the density of rays. In a 360° unbounded scene, the outward rays for backgrounds are much sparser than those for central objects such that the under-sampling of backgrounds leads to the difficulty of reconstruction.

(ii) Large Depth Variation. NeRF-based methods commonly employ a uniform sampling strategy in Euclidean space to render bounded scenes. However, the depth for a 360° unbounded scene varies significantly, usually ranging from ten of centimeters to tens of meters. Consequently, the uniform sampling strategy cannot be directly extended to unbounded scenes as too many samples are required to render the backgrounds.

In this paper, we propose a spherical radiance field (SRF) for efficient novel view synthesis in 360° unbounded scenes. Our SRF encodes the scene into layered spherical implicit

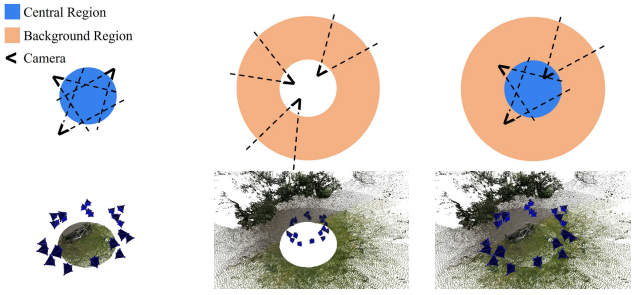
Manuscript received 30 July 2023; revised 25 December 2023 and 5 February 2024; accepted 5 March 2024. Date of publication 10 June 2024; date of current version 14 June 2024. This work was supported in part by the National Natural Science Foundation of China under Grant U20A20185, Grant 62372491, Grant 62301601, Grant U23B2013, and Grant 62276176; in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2022B1515020103 and Grant 2023B1515120087; and in part by Shenzhen Science and Technology Program under Grant RCYX20200714114641140. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Sebastian Knorr. *(Corresponding author: Yulan Guo.)*

Minglin Chen, Longguang Wang, and Yulan Guo are with the School of Electronics and Communication Engineering, Sun Yat-sen University, Shenzhen Campus, Shenzhen 518107, China (e-mail: chenmlin8@mail2.sysu.edu.cn; wanglongguang15@nudt.edu.cn; guoyulan@sysu.edu.cn).

Yinjie Lei is with the College of Electronics and Information Engineering, Sichuan University, Chengdu 610065, China (e-mail: yinjie@scu.edu.cn).

Zilong Dong is with the Alibaba Group, Hangzhou 310056, China (e-mail: list.dzl@alibaba-inc.com).

Digital Object Identifier 10.1109/TIP.2024.3409052



(a) inward (b) outward (c) inward & outward

Fig. 1. Comparison of typical settings of scenes in novel view synthesis. (a) All cameras face inward to capture central objects. Existing accelerated NeRF-based methods [22], [23], [24], [25], [51] are suitable for this setting. (b) All cameras face outward to capture the background. MSI-based methods [52] are used for these scenes. (c) Cameras capture not only the central objects but also the background. While previous methods cannot well handle this complicated setting, the proposed method can efficiently synthesize novel views with unconstrained perspectives.

representations and employs MLPs to render novel views. Specifically, the 3D scene is represented as multiple concentric spheres with different radii, where each sphere is parameterized using an equirectangular projection (ERP) image. On the ERP image, each pixel encodes the scene within its neighboring 3D space into a trainable feature, which is fed to a shallow MLP to infer the density and color for volume rendering. By incorporating layered spherical structure with implicit representations, our SRF can well model a 360° unbounded scene with superior performance in terms of both accuracy and efficiency.

Our main contributions can be summarized as follows:

- We propose a spherical radiance field for 360° unbounded novel view synthesis. Our SRF formulates a 3D scene as multiple concentric spheres and employs volume rendering to synthesize novel views.
- Our method achieves state-of-the-art performance on three real-world 360° unbounded datasets in 23 minutes of training time while producing competitive results on bounded datasets.

II. Related Work

In this section, we first review NeRF-based methods developed for bounded and unbounded scenes. Then, we briefly discuss the multi-sphere images that are related to our method.

A. Bounded View Synthesis

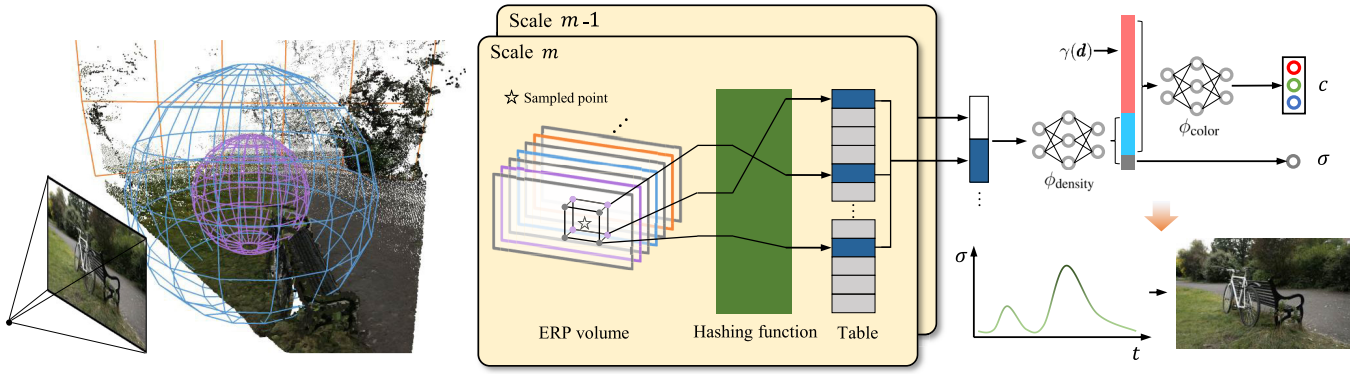
Bounded view synthesis assumes that the relevant scene is contained within a limited range, e.g., object-level and enclosed room-level view synthesis. As shown in Fig. 1(a), the methods of bounded view synthesis can only model the central region. In real-world scenarios, additional processing is required to remove the background regions [21], [42] before bounded view synthesis. Recently, Mildenhall et al. [11] proposed neural radiance field, which synthesizes photorealistic views by combining implicit representation [12], [13] with volume rendering. Instead of using one MLP to represent the whole scene, Liu et al. [42] embedded features in sparse voxel grids which were considered as local properties of implicit fields. Barron et al. [15] introduced conical frustums rendering to NeRF, which helps to reduce aliasing artifacts and achieve higher image quality. Verbin et al. [16] introduced the bidirectional reflectance distribution function (BRDF) in NeRF to model shiny surfaces. Subsequent works further advance NeRF to improve its capability to reconstruct fine details, making NeRF generalizable [28], [29], [30], [31], [53], dynamic [54], [55], [56], and controllable [57], [58].

Despite the promising results produced by NeRF-based methods [12], [15], [16], they are high in computational complexity due to the overhead of MLP query on each sampled point. To remedy this, Lindell et al. [59] proposed AutoInt [59] to learn closed-form solutions of integrals, which replaces hundreds of forwarding passes with two queries. Reiser et al. [20] utilized thousands of shallow MLPs instead of one deep MLP to represent a scene. In addition, hybrid radiance field representation has been studied to accelerate NeRF-based approaches, which combines an explicit data structure (e.g., dense or sparse voxel grid [20], [22], [23], [24], [42], and Octree [21]) with shallow MLPs.

B. Unbounded View Synthesis

Different from bounded view synthesis, unbounded view synthesis aims to synthesize novel views under an unconstrained setting. Early works [6], [60], [61] focus on forward-facing unbounded scenes and employ multi-plane images to represent the scene as a set of RGBA images in different depths. However, these methods can only synthesize novel views within a limited angle range, and cannot adapt to 360° novel view synthesis. As shown in Fig. 1(c), the methods of 360° unbounded view synthesis can model both central and background regions. Recently, several methods extend NeRF to 360° unbounded scenes. Specifically, Zhang et al. [62] first proposed inverted sphere parameterization that employs another single MLP to model the background. Neff et al. [63] explored to warp the metric space using a radial distortion function. However, the available camera region is limited to a small cell. Barron et al. [64] adapted Mip-NeRF [15] to 360° unbounded scene with a newly-developed space contraction method. Nevertheless, these methods synthesize highly realistic images at the cost of a very high computational cost, usually requiring a training time of several days to one week for a single scene.

To accelerate training in unbounded scenes, a naive idea is to transfer the core techniques of Mip-NeRF 360 [64] to NGP [22], as done in [65]. However, such a straightforward method is not trivial, as the integrated positional encoding (IPE) of Mip-NeRF 360 is incompatible with the grid-based method (e.g., NGP). Recently, several concurrent methods [66], [67], [68] attempt to achieve fast training in unbounded scenes. Barron et al. [66] proposed a spiral multisampling strategy and an anti-aliased interlevel loss to solve the spatial and z-axial aliased problem caused by directly combining Mip-NeRF 360 and NGP. Kerbl et al. [67] represent the scene as a set of optimizable 3D Gaussian points, and accelerate training based on the point splatting rendering



(a) Multiple concentric spheres (b) Multi-scale hashing encoding (c) Volumetric rendering

Fig. 2. Illustration of our method for 360° unbounded novel view synthesis. (a) a 3D scene is represented using multiple concentric spheres with different radii, where each sphere is parameterized by an equirectangular projection (ERP) image. ERP images are stacked as an ERP volume. (b) Multi-resolution ERP volumes with hashing encoding are used to maintain the trainable features as input to the following shallow MLPs. (c) The density σ and color c predicted from the MLPs are used in the volumetric rendering algorithm to synthesize an image.

method. Orthogonal to these concurrent works, we explore the effect of spherical mapping in grid-based methods for unbounded scenes.

C. Multi-Sphere Images

Inspired by the success of multi-plane images [60] in forward-facing unbounded view synthesis, multi-sphere images (MSI) is introduced for unbounded novel view synthesis under the setting that all cameras face outward to the background. As shown in Fig. 1(b), the methods of outward view synthesis require modeling of the background region. Specifically, MatryODShka [52] is developed to synthesize novel views from an omnidirectional stereo image pair. The scene is decomposed into multi-sphere images (MSI) to explicitly encode RGB and alpha values. However, this method can only model the outward background (Fig. 1(b)) and cannot handle central objects.

Motivated by the powerful capacity of implicit representation in modeling central objects and the ability of sphere images in handling 360° unbounded backgrounds, we develop a spherical radiance field by incorporating advantages of both techniques. Different from NeRF-based methods that can only model central objects, the spherical structure enables our SRF to efficiently model background regions. In contrast to MatryODShka [52], our SRF encodes layered scenes into implicit representations such that central objects can be well synthesized.

III. Methodology

Given a set of posed images acquired from 360° unbounded scenes, our task is to synthesize a novel view with a specified pose. In this section, we first introduce the formulation of our spherical radiance field. Then, we describe the spherical warped occupancy grid used to accelerate the training and rendering. Finally, we introduce the implementation details of our approach. Figure 2 illustrates the proposed SRF for efficient 360° unbounded novel view synthesis.

A. Spherical Radiance Field

1) Overview: Our spherical radiance field models a 3D scene using multiple concentric spheres with different radii, as shown in Fig. 2(a). Specifically, the spherical radiance field encodes scene information at different radii into their corresponding spheres. Formally, N concentric spheres $\mathcal{S} = \{S_1, \ldots, S_N\}$ are constructed, and the radius r_n of sphere S_n is set as:

$$r_n = \begin{cases} 2 \cdot \frac{n}{N}, & n \in [1, \frac{N}{2}) \\ ((1-k) \cdot r_{\text{near}}^{-1} + k \cdot r_{\text{far}}^{-1})^{-1}, & n \in [\frac{N}{2}, N] \end{cases} \tag{1}$$

where $k = 2 \cdot \frac{n}{N} - 1$, r_{near} and r_{far} denote the nearest and farthest distances, respectively. Note that, a half of these spheres are uniformly placed inside the unit sphere to model central objects. Meanwhile, the other half spheres are scattered outside the unit sphere, with radii being uniformly sampled in the disparity space to better model unbounded background. In addition, we translate and scale the scene to ensure it is located at the origin with central objects being bounded by a unit sphere.

2) ERP Volume: For each sphere S_n, an equirectangular projection image (ERP) of size $H \times W \times 2$ is used to encode its corresponding layered scene at different positions into a representation of length 2.

For a 3D position (x, y, z) on the surface of sphere S_n with a radius of $r_n = \sqrt{x^2 + y^2 + z^2}$, we map it to spherical coordinate (θ, ϕ) as:

$$\begin{cases} \theta = \arctan(\frac{y}{x}) \\ \phi = \arcsin(\frac{z}{||r_n||}). \end{cases} \tag{2}$$

Correspondingly, the converse transformation can be written as:

$$\begin{cases} x = r_n \sin(\phi)\cos(\theta) \\ y = r_n \sin(\phi)\sin(\theta) \\ z = r_n \cos(\phi). \end{cases} \tag{3}$$

To map a 3D sphere to a 2D image, several spherical projections have been exploited to preserve the properties, such as equidistant, equal-area, and equal-angle. As an equidistant projection, the equirectangular projection (ERP) directly maps the spherical coordinate to the pixel coordinate with a simple

normalization: $(i, j) = \text{normalize}(\theta, \phi)$, where $i \in [0, H)$, $j \in [0, W)$ are the pixel coordinate. As an equal-area projection, the sinusoidal projection represents the poles as points using the transformation as: $(i, j) = \text{normalize}(\theta\cos(\phi), \phi)$. As an equal-angle projection, Mercator projection preserves the angle locally as: $(i, j) = \text{normalize}(\theta, \ln(\tan(\frac{\pi}{4} + \frac{\phi}{2}))$. We use the ERP in our method, as it performs better than others (as demonstrated in Table V).

By stacking all ERP images, the 3D scene is encoded in an ERP volume (as shown in Fig. 2(b)) of size $N \times H \times W \times 2$.

3) Hashing Encoding: Instead of using a dense grid to store the ERP volume, we leverage a hashing encoding technique [22] to reduce storage consumption. For each ERP, a hashing table of size $T \times 2$ is stored and a spatial hashing function [22] is used to calculate hashing index using the indices (n, h, w) in the ERP volume as follows:

$$I(n, h, w) = \left(n \bigoplus h\pi_h \bigoplus w\pi_w\right) \bmod T, \tag{4}$$

where $\bigoplus$ is the bit-wise XOR operation, π_h and π_w are large prime numbers.

4) Multi-Scale Strategy: To efficiently model the scene at different scales, we construct multi-scale ERP volumes to capture details of different granularities. Specifically, the 3D scene is represented using M ERP volumes with resolution ranging from $N_{\min} \times H_{\min} \times W_{\min}$ to $N_{\max} \times H_{\max} \times W_{\max}$. The resolution of the m-th ERP volume (N_m, H_m, and W_m) grows in a geometric progression as in [22] and [69]:

$$N_m = [N_{\min} \cdot b^m], \tag{5}$$

$$b = \exp\left(\frac{\ln o_{\max} - \ln o_{\min}}{M - 1}\right). \tag{6}$$

5) Rendering: Volumetric rendering [14] is used to render the ray casting from each pixel. Specifically, a ray of the pixel $\boldsymbol{r} = \boldsymbol{o} + t \cdot \boldsymbol{d}$ is first generated from the camera pose, where $\boldsymbol{o}$ and $\boldsymbol{d}$ are the camera position and view direction, respectively. Then, several points are sampled along the ray, with their features being generated using trilinear interpolation in the multi-scale ERP volumes. Next, the concatenation of the sampled features $\boldsymbol{f}$ is fed to two MLPs (i.e., ϕ_{density}, and ϕ_{color}) to obtain the densities $\{\sigma_i\}$ and colors $\{c_i\}$ for these sampled points $\{t_i\}$, as shown in Fig. 2(c).

$$[\sigma, \boldsymbol{f}_\sigma] = \phi_{\text{density}}(\boldsymbol{f}), \tag{7}$$

$$c = \phi_{\text{color}}(\boldsymbol{f}_\sigma, \gamma(\boldsymbol{d})), \tag{8}$$

where $\boldsymbol{f}_\sigma$ is the bottleneck feature used for color prediction, $\boldsymbol{d}$ and $\gamma(\cdot)$ are the view direction and spherical harmonics encoding [22], respectively. Note that, the ray color $\hat{C}(\boldsymbol{r})$ is obtained by:

$$\hat{C}(\boldsymbol{r}) = \sum_i T_i \sigma_i c_i \Delta t_i, \tag{9}$$

where $T_i = \exp(-\sum_{j<i} \sigma_j \Delta t_j)$ and $\Delta t_i = t_{i+1} - t_i$. The density measures the objectness of the corresponding position in the 3D scene and the view-dependent color captures the appearance at the corresponding position by considering diverse influences, including ambient, diffuse, and specular light.

6) Discussion: We compare three space warping methods (i.e., linear warping [22], radial distortion [63], and space contraction [64]) for 360° unbounded novel view synthesis, and highlight the motivation of our spherical mapping of ERP volume.

Since NeRF-based methods require a bounded domain, e.g., $[-1, 1]^3$ or $[0, 1]^3$, a straightforward strategy to handle 360° unbounded scene is to linearly scale and translate the large scene. Specifically, a 3D point $\boldsymbol{p} \in \mathcal{R}^3$ in Euclidean space is transformed into the bounded domain as:

$$\hat{\boldsymbol{p}} = s \cdot \boldsymbol{p} + \boldsymbol{t}, \tag{10}$$

where $s \in \mathcal{R}$ and $\boldsymbol{t} \in \mathcal{R}^3$ are the predefined scale and the translation of the scene. In practice, linear warping is usually performed on camera positions such that the large scene is fit in a bounded domain as in [22]. However, the ray intersection far from the origin is highly sparse (Fig. 3(g)) and thus lacks sufficient samples for NeRF training, leading to artifacts in distant background of synthesized novel views.

To migrate the above issue, the radial distortion [63] and space contraction [64] methods are proposed to non-linearly warp the space along the radial axis. Specifically, radial distortion employs the following transformation:

$$\hat{\boldsymbol{p}} = \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}| \cdot d_{\max}}}, \tag{11}$$

where $d_{\max}$ is the maximum depth. To eliminate the need of a predefined maximum depth, space contraction is introduced to transform a 3D point $\boldsymbol{p}$ when $|\boldsymbol{p}| > 1$:

$$\hat{\boldsymbol{p}} = (2 - \frac{1}{|\boldsymbol{p}|}) \cdot \frac{\boldsymbol{p}}{|\boldsymbol{p}|}. \tag{12}$$

In this way, the distant region becomes closer in the warped space.

However, as pointed out in the Mip-NeRF-360 benchmark [64], the ray intersections are still sparse in the warped space (Fig. 3(h) and (i)), which motivates us to explore more suitable space warping method for unbounded scenes. As shown in Fig. 3(j), our spherical mapping of ERP volume produces denser ray intersections in the warp space.

Compared with other space warping methods, our method proposes to use spherical mapping for 360° unbounded novel view synthesis, which has the following unique advantages: (1) With spherical coordinates, most trainable features are gathered near the central object (Fig. 4), providing more capacities to reconstruct the object with finer details. The effectiveness of spherical coordinates is highlighted in Table IV. (2) The trainable features are widely scattered in the background regions of sparse rays, as the 3D distance at the same angular resolution increases with the radius. This facilitates background regions to be efficiently rendered, as shown in Table I.

B. Spherical Warped Occupancy Grid

We develop an occupancy grid to accelerate the ray marching of spherical radiance fields. The caching strategy is well explored in NeRF to accelerate both training [22], [24] and rendering [21], as the sampling skips the empty space and reduces the number of sampled points. In the original

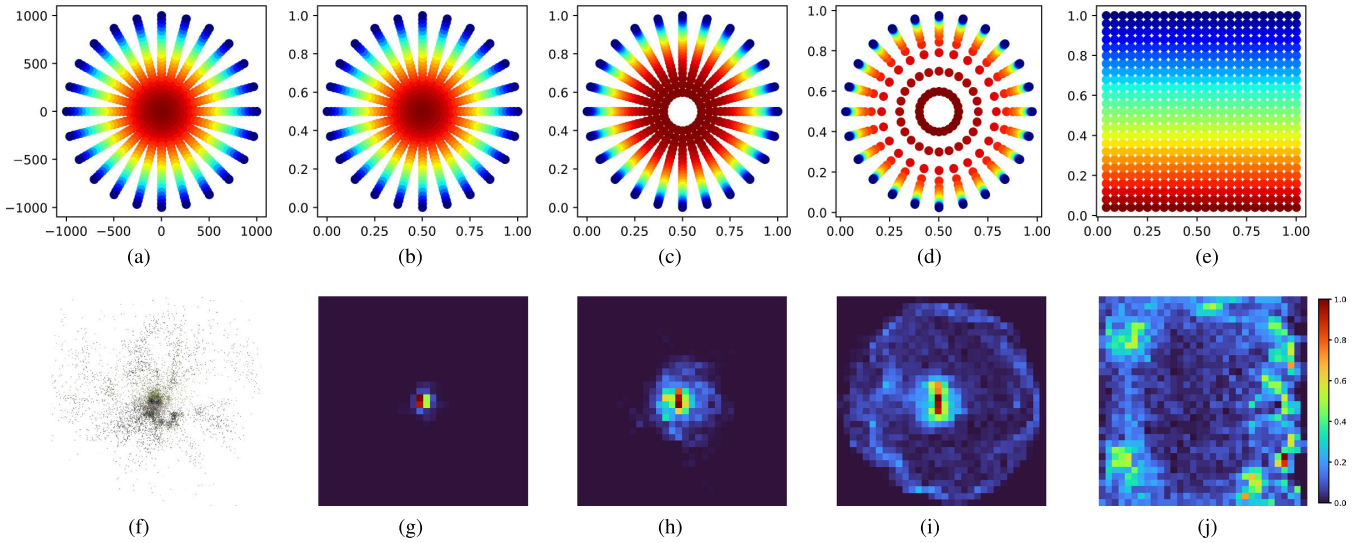


Fig. 3. Comparison of different space warping methods for unbounded novel view synthesis. *First row*: 2D visualization of original Euclidean space (a), and warped space for linear warping (b), radial distortion (c), space contraction (d), and spherical mapping (e). *Second row*: top-down view of the *bicyle* scene in the Mip-NeRF-360 dataset [64] (f), and distribution of ray intersection for linear warping (g), radial distortion (h), space contraction (i), and spherical mapping (j). Both radial distortion and space contraction warps the space along the radial axis, making distant areas closer in warped space. With these warping methods, the distribution of ray intersection becomes denser in the warped space (see (h) and (i)). Our method warps the metric space along the angle axis using spherical mapping, making the distance points as close as those of the same angle after warping. The ray intersection distributes densely in spherical warped space (see (j)). (Note that, we visualize the ray intersection distribution by counting the number of points in warped space, where the point cloud of the scene is obtained from the rendered depth image by [64]).

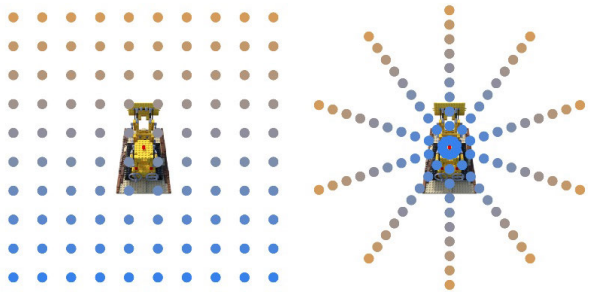


Fig. 4. Euclidean vs Spherical Coordinates. The colored dot denotes the trainable feature. (a) With Euclidean coordinates, the trainable features uniformly distribute in the space. (b) With spherical coordinates, the trainable features are dense near the central object for fine-grained reconstruction and scattered away from the central object for efficient rendering of backgrounds.

NGP [22], they use multi-layer occupancy grids with exponentially growing sizes to cache the density in the Euclidean space. However, the strategy is not suitable for unbounded scenes, as the memory consumption of the occupancy grid is growing exponentially.

In contrast, we employ a single-layer occupancy grid $\mathcal{G}$ with a size of $g \times g \times g$ in the warped space of SRF. During training, we alternatively update the occupancy grid and train the SRF. The occupancy grid is updated with the exponential moving average (EMA) from the uniform samples in the warped spaced. Specifically, suppose $\boldsymbol{p}_w = (\theta', \phi', r)$ is the sampled point in the warped space and σ is its density from Eq. (7).

$$\mathcal{G}(g\theta', g\phi', gr) = \omega\mathcal{G}(g\theta', g\phi', gr) + (1-\omega)\sigma \tag{13}$$

where θ' and ϕ' are the [0, 1]-normalized versions of θ and ϕ, respectively, ω is the decay weight (0.98 in our experiments). During the SRF training, we discard the sampled points whose value in the occupancy grid is below the threshold T (0.01 in our experiments).

C. Implementation

We train the proposed SRF for each scene. In the training phase, the multi-resolution ERP volumes and MLPs are optimized simultaneously. During preprocessing, the scene is translated and scaled to ensure that the mean of camera positions is zeros and all cameras are located at the unit sphere. To render a ray, we sample points along the ray using the ray-marching algorithm [22]. The marching interval of ray at length t is set as $\Delta t = k \cdot t$, where $k = 1/256$. Our marching interval has a small interval at close sample points, and a large interval at distant sample points.

During training, the photometric loss is used for optimization:

$$\mathcal{L} = \sum_{\boldsymbol{r}} ||\hat{C}(\boldsymbol{r}) - C(\boldsymbol{r})||^2, \tag{14}$$

where $\hat{C}(\boldsymbol{r})$ and $C(\boldsymbol{r})$ are the rendered color and ground-truth color of ray $\boldsymbol{r}$.

In our experiments, we implemented our method using TCNN [70] with an NVIDIA RTX3090. A 16-level ERP volume is constructed (i.e., $M = 16$) with the coarsest and finest resolutions being set as $N_{\min} = H_{\min} = W_{\min} = 16$, $N_{\max} = H_{\max} = W_{\max} = 2048$. The hashing table size T for each ERP volume is 2^{19}. The density MLP (f_{density}) has one hidden layer, while the color MLP (f_{density}) has two hidden layers. We randomly sampled 4096 rays from all images for each training iteration. We used the ADAM [71] optimizer with an initial learning rate of 0.01.

IV. Experiments

In this section, we first present the datasets and metrics. Then, we compare our method with previous state-of-the-art approaches on unbounded and bounded scenes. Finally, we conduct ablation experiments to validate the effectiveness of our method designs.

A. Datasets and Metrics

We evaluated our method on 3 public real-world 360° unbounded benchmarks, including Mip-NeRF-360 [64], Light-Field [72], and Tanks-and-Temples [73]. For a fair comparison with existing NeRF-based methods, we also evaluate our method on a bounded benchmark, i.e., the NeRF-Synthetic dataset [11].

1) Mip-NeRF-360: We use 9 scenes from the Mip-NeRF-360 dataset [64], which contains 5 outdoor scenes (i.e., bicycle, flowers, garden, stump, and treehill) and 4 indoor scenes (i.e., room, counter, kitchen, and bonsai). For each scene, there are 125-311 images captured surrounding a central object. These images observe not only the central object but also the complex background, e.g., irregular plants, and distant buildings. We use the poses provided by [64] and split the train/test sets with a ratio of 7:1 as in [64].

2) Light-Field: We use 4 scenes from the Light-Field dataset [72], i.e., africa, basket, ship, and torch. Each scene contains a sequence of images captured by the hand-held camera. We use the temporally subsampled images provided by [62], resulting in 64-109 images for each scene. We use the train/val sets and poses provided by [62].

3) Tanks-and-Temples: We use a subset of the Tanks-and-Temples dataset [73], i.e., m60, playground, train, and truck. These images are captured with a hand-held camera in large-scale outdoor scenes. Although the camera faces toward the central object, its trajectory is not close to a sphere. We evaluate our method on this dataset to validate its robustness to complicated camera distributions. We use the train/val sets and poses provided by [62].

4) NeRF-Synthetic: The NeRF-Synthetic dataset [11] consists of 8 human-designed objects, i.e., chair, drums, ficus, hotdog, lego, materials, mic, and ship. For each object, there are 100 and 200 images at 800×800 resolution with different poses for training and testing, respectively. These images are rendered using the Blender Cycles engine according to the poses perfectly sampled on an upper hemisphere.

To measure the quality of synthesized images, we use PSNR, SSIM, and LPIPS-VGG as evaluation metrics following [11]. In addition, we focus on the evaluation of training time on an NVIDIA 3090 GPU.

B. Results on 360° Unbounded Scenes

We evaluate our method on 360° unbounded scenes and compare it to 3 NeRF-based methods, including NeRF [11], NeRF++ [62], and Mip-NeRF 360 [64]. Note that, NeRF++ and Mip-NeRF 360 are developed for 360° unbounded scenes. In particular, Mip-NeRF achieves state-of-the-art 360° unbounded novel view synthesis in terms of image quality. In addition, we also include 3 accelerated NeRF methods (i.e., Plenoxels [23], DVGO v2 [24], [74], and NGP [22]) for comparison to evaluate the efficiency of our method.

For Mip-NeRF 360, Plenoxels, DVGO v2 and NGP, we re-trained these models on 360° unbounded scenes (i.e., the Mip-NeRF-360 dataset, the Tanks-and-Temples dataset, and the Light-Field dataset) for comparison with our SRF. We present the training details for these methods as follows:

Mip-NeRF 360: For each scene in the Tanks-and-Temples dataset and the Light-Field dataset, we trained Mip-NeRF 360 with a batch size of 2^{14} using 4 NVIDIA 3090 GPUs. It took approximately 42 hours for a training step of 250k. For other settings, we used the configuration file provided in the Mip-NeRF 360 codebase.[1] In addition, we provide the results of the Mip-NeRF-360 dataset at a training step of 5k on an NVIDIA 3090 GPU ($\sim$ 40 minutes).

Plenoxels: We trained Plenoxels on each scene in the Mip-NeRF-360 dataset and the Light-Field dataset. The resolution of voxel grids was initially set to 128^3, and gradually upscaled to 640^3, which is limited by the maximum memory of an NVIDIA RTX 3090. We trained Plenoxels with a total step of 102400 and a batch size of 5000.

DVGO v2: We trained DVGO v2 on three unbounded datasets using the configuration provided in their codebase.[2]

NGP: We trained NGP on three unbounded datasets. For each scene, we recentered and scaled all camera positions such that they lie within the $[0, 1]^3$ cube. The marching step size was exponential with a factor of $1/256$. We tuned the scale of axis-aligned bounding box (AABB) on each scene for better results. The number of training steps was set to 300k. There was no significant improvement with more training steps.

1) Quantitative Results: Tables I, II, and III show the comparison result on the Mip-NeRF-360, Light-Field, and Tanks-and-Temples datasets, respectively. Our SRF achieves the best performance in terms of PSNR and SSIM among all efficient approaches with training time less than an hour. Specifically, compared to Plenoxels, DVGO v2, and NGP, our SRF produces much better performance with a comparable training time.

As compared to NeRF and NeRF++, our method achieves better performance while outperforming them by two orders of magnitude in training time. Compared to Mip-NeRF 360, our SRF produces comparable or even better results with a 438× speedup during training. In particular, our method outperforms Mip-NeRF 360 on several outdoor scenes (i.e., the *bicyle*, *flowers*, and *treehill* scenes) of the Mip-NeRF-360 dataset in terms of both PSNR and SSIM (Table I). Under comparable training time, our SRF surpasses Mip-NeRF 360 with PSNR scores being improved from 21.17 to 27.06. For the Light-Field dataset, our method consistently outperforms Mip-NeRF 360 in all scenes of the Light-Field dataset (Table II), while achieving better performance on the Tanks-and-Temples dataset in terms of averaged PSNR and SSIM (Table III). Note that, the camera distribution of the Tanks and Temples dataset is more complicated than the other two datasets. Therefore, the higher accuracy of our SRF clearly

[1] https://github.com/google-research/multinerf
[2] https://github.com/sunset1995/DirectVoxGO

TABLE I

QUANTITATIVE RESULTS ON THE MIP-NERF-360 DATASET. 'MRF 360" REPRESENTS THE MIP-NERF 360. "TIME" REPRESENTS THE EQUIVALENT TRAINING TIME ON AN NVIDIA 3090 GPU

	bicycle		flowers		garden		stump		treehill		room		counter		kitchen		bonsai		mean		
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	Time
Implicit representation																					
NeRF [11]	21.76	0.455	19.40	0.376	23.11	0.546	21.73	0.453	21.28	0.459	28.56	0.843	25.67	0.775	26.31	0.749	26.81	0.792	23.85	0.605	4 days
NeRF++ [62]	22.64	0.526	20.31	0.453	24.32	0.635	24.34	0.594	22.20	0.530	28.87	0.852	26.38	0.802	27.80	0.816	29.15	0.876	25.11	0.676	9 days
MRF 360 [64]	**24.37**	**0.685**	**21.73**	**0.583**	**26.98**	**0.813**	**26.40**	**0.744**	**22.87**	**0.632**	**31.63**	**0.913**	**29.55**	**0.894**	**32.23**	**0.920**	**33.46**	**0.941**	**27.69**	**0.792**	7 days
MRF 360 [64]	19.51	0.339	17.70	0.261	20.66	0.351	20.79	0.374	20.66	0.392	23.89	0.733	21.66	0.651	22.30	0.527	23.35	0.675	21.17	0.478	40 min
Hybrid representation																					
Plenoxels [23]	19.32	0.429	20.30	0.451	23.13	0.628	15.38	0.359	22.25	0.531	28.24	0.853	21.57	0.751	24.16	0.701	21.58	0.793	21.77	0.611	24 min
DVGO v2 [74]	22.09	0.481	19.22	0.361	24.38	0.630	23.57	0.580	20.96	0.483	28.35	0.852	25.79	0.783	25.99	0.710	27.83	0.829	24.24	0.634	16 min
NGP [22]	22.49	0.528	19.76	0.439	24.94	0.695	24.81	0.617	22.55	0.564	29.19	0.878	26.07	0.829	27.18	0.881	27.70	0.881	24.97	0.701	30 min
SRF (Ours)	**24.52**	**0.689**	**22.01**	**0.591**	**26.83**	**0.812**	**26.32**	**0.739**	**23.12**	**0.633**	**30.99**	**0.897**	**27.78**	**0.851**	**30.16**	**0.894**	**31.82**	**0.907**	**27.06**	**0.779**	23 min

TABLE II

QUANTITATIVE RESULTS ON THE LIGHT-FIELD DATASET.
"TIME" REPRESENTS THE EQUIVALENT TRAINING TIME ON AN NVIDIA 3090 GPU

	africa		basket		ship		torch		mean		
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	Time↓
Implicit representation											
NeRF [11]	26.16	0.894	20.83	0.805	23.24	0.801	22.81	0.811	23.26	0.828	4 days
NeRF++ [62]	27.41	0.923	21.84	0.884	25.35	0.867	24.68	0.867	24.82	0.885	9 days
Mip-NeRF 360 [64]	**30.85**	**0.941**	**22.56**	**0.911**	**27.24**	**0.903**	**27.03**	**0.894**	**26.92**	**0.894**	7 days
Hybrid representation											
Plenoxels [23]	23.03	0.848	17.60	0.753	26.17	0.832	24.84	0.834	22.91	0.817	18 min
DVGO v2 [74]	15.82	0.706	17.77	0.720	25.11	0.798	22.83	0.778	20.38	0.751	11 min
NGP [22]	28.87	0.911	19.86	0.849	27.80	0.878	23.78	0.854	25.08	0.873	30 min
SRF (Ours)	**31.08**	**0.948**	**22.87**	**0.915**	**28.27**	**0.908**	**28.18**	**0.899**	**27.60**	**0.918**	23 min

TABLE III

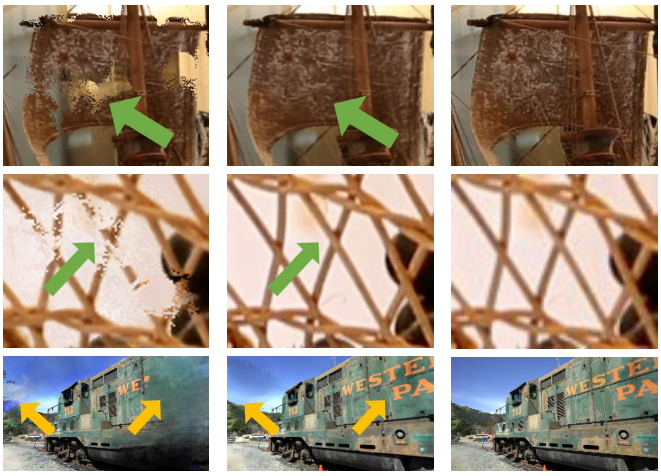
QUANTITATIVE RESULTS ON THE TANKS-AND-TEMPLES DATASET.
"TIME" REPRESENTS THE EQUIVALENT TRAINING TIME ON AN NVIDIA 3090 GPU

	m60		playground		train		truck		mean		
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	Time↓
Implicit representation											
NeRF [11]	16.86	0.702	21.55	0.765	16.64	0.635	20.85	0.747	18.98	0.712	4 days
NeRF++ [62]	**17.88**	**0.738**	22.37	**0.799**	**17.17**	**0.672**	22.77	**0.823**	**20.05**	**0.758**	9 days
Mip-NeRF 360 [64]	16.36	0.664	**24.05**	0.772	14.53	0.529	**23.66**	0.818	19.65	0.696	7 days
Hybrid representation											
Plenoxels [23]	17.93	0.684	23.03	0.711	17.97	0.628	22.67	0.756	20.40	0.695	27 min
DVGO v2 [74]	17.54	0.650	22.70	0.669	17.82	0.564	22.01	0.704	20.02	0.647	15 min
NGP [22]	18.91	0.798	22.60	0.765	17.88	0.692	22.17	0.796	20.39	0.763	30 min
SRF (Ours)	**18.95**	**0.799**	**23.12**	**0.794**	**17.92**	**0.697**	**23.21**	**0.806**	**20.80**	**0.774**	23 min

demonstrates its superior robustness against different camera trajectories.

As compared to existing accelerated methods with hybrid representation (i.e., Plenoxels, DVGO v2, and NGP), our method consistently outperforms in all scenes of three datasets. Specifically, our method outperforms NGP from 24.97 to 27.06 in the Mip-NeRF-360 dataset (Table I), from 25.08 to 27.60 in the Light-Field dataset (Table II), from 20.39 to 20.80 in the Tanks-and-Temples dataset (Table III), in terms of averaged PSNR. This demonstrates the proposed spherical mapping is more suitable for 360° unbounded scenes.

2) Qualitative Results: Figure 5 shows the visual results on the Light-Field dataset and the Tanks-and-Temples dataset. We can observe that Mip-NeRF 360 fails to render the structure near the camera, leading to "missing" (as shown in the *ship* and *basket* scenes) or blurring (e.g., missing words as shown in the *train* scene) artifacts. In contrast, our method can model the scene from a more complicated camera trajectory, and produce a better rendering result for the thin object near the camera. This is because, more capacity is contributed to the central area under our spherical mapping.



(a) Mip-NeRF 360 [64] (b) SRF (Ours) (c) Ground-Truth

Fig. 5. Qualitative comparison on the Light-Field dataset and the Tanks-and-Temples dataset. (1st row: the *ship* scene, 2nd row: the *basket* scene, 3rd row: the *train* scene).

Figure 6 compares the visual results produced by different methods on the Mip-NeRF-360 dataset. It can be observed that Mip-NeRF 360 fails to render the thin structure, as the missing spoke in the *bicyle* scene (first row in Fig. 6), while NGP suffers "floaters" artifacts and produces blurring artifacts and noise in the distant background. In contrast, our method produces results with finer details both on the central object and distant background even with less than 30 minutes of training. This further demonstrates the superiority of our spherical radiance field.

C. Results on Bounded Scenes

Although our method aims at synthesizing novel views for 360° unbounded scenes, it can be seamlessly adapted to the bounded dataset. We conduct experiments on the NeRF-Synthetic dataset, and compare our method with four implicit NeRF-based methods (i.e., NeRF, NSVF, Mip-NeRF, and RefNeRF), two generalizable NeRF-based methods (i.e., IBRNet, and Point-NeRF), and five accelerated NeRF-based methods (i.e., TensoRF, DVGO, Plenoxels, ReLUField, and NGP). Quantitative results are presented in Table IV and visual results are provided in Fig. 7.

As shown in Table IV, our SRF achieves better performance in terms of PSNR as compared to implicit representation based NeRF methods with training time being shorter than 10 minutes. Specifically, our method produces competitive results to the state-of-the-art method (i.e., RefNeRF) with much higher efficiency. RefNeRF requires more than one day to train on a modern GPU while our method achieves over 70× speedup. As compared to generalizable NeRF-based methods, our SRF outperforms them without using any additional datasets. With 7 minutes of training, our method outperforms the state-of-the-art accelerated NeRF-based method (i.e., NGP) in terms of PSNR (33.21 vs 33.18). This clearly demonstrates the effectiveness of our spherical radiance field.

Figure 7 compares the visual results produced our method and other approaches. It can be observed that NGP and RefNeRF suffer blurring artifacts and produce images that lack fine details. In contrast, our method produces results with clearer details (e.g., the ball in materials scene, and the structure of ship) and much higher perceptual quality even with only 20-minute training. This further demonstrates the superiority of our method.

TABLE IV

COMPARISON ON THE NERF-SYNTHETIC DATASET. (PT. DENOTES WHETHER THE MODEL NEEDS TO BE PRETRAINED ON A LARGE DATASET)

	PT.	PSNR↑	SSIM↑	LPIPS↓	Time↓
Implicit representation					
NeRF [11]	×	31.01	0.947	0.081	41 h
NSVF [42]	×	31.75	0.954	0.048	53 h
Mip-NeRF [15]	×	33.09	0.961	0.043	36 h
RefNeRF [16]	×	**33.99**	**0.966**	**0.038**	38 h
Generalizable representation					
IBRNet [29]	✓	28.14	0.942	0.072	15 m
Point-NeRF [75]	✓	**33.31**	**0.978**	**0.049**	20 m
Hybrid representation					
TensoRF [30]	×	33.14	0.963	-	17 m
DVGO [24]	×	31.95	0.957	-	14 m
Plenoxels [23]	×	31.71	0.958	0.049	11 m
ReLUField [51]	×	30.04	-	0.050	10 m
NGP [22]	×	33.18	-	-	5 m
SRF (Ours)	×	33.21	0.974	0.039	7 m
SRF (Ours)	×	**33.40**	**0.977**	**0.038**	20 m

TABLE V

RESULTS ACHIEVED ON THE MIP-NERF-360 DATASET

	PSNR↑	SSIM↑	LPIPS↓
Linear warping	26.25	0.728	0.316
Radial distortion	26.82	0.741	0.296
Space contraction	26.95	0.748	0.285
sinusoidal projection	25.47	0.721	0.283
Mercator Projection	24.98	0.612	0.512
ERP (Ours)	**27.06**	**0.779**	**0.213**

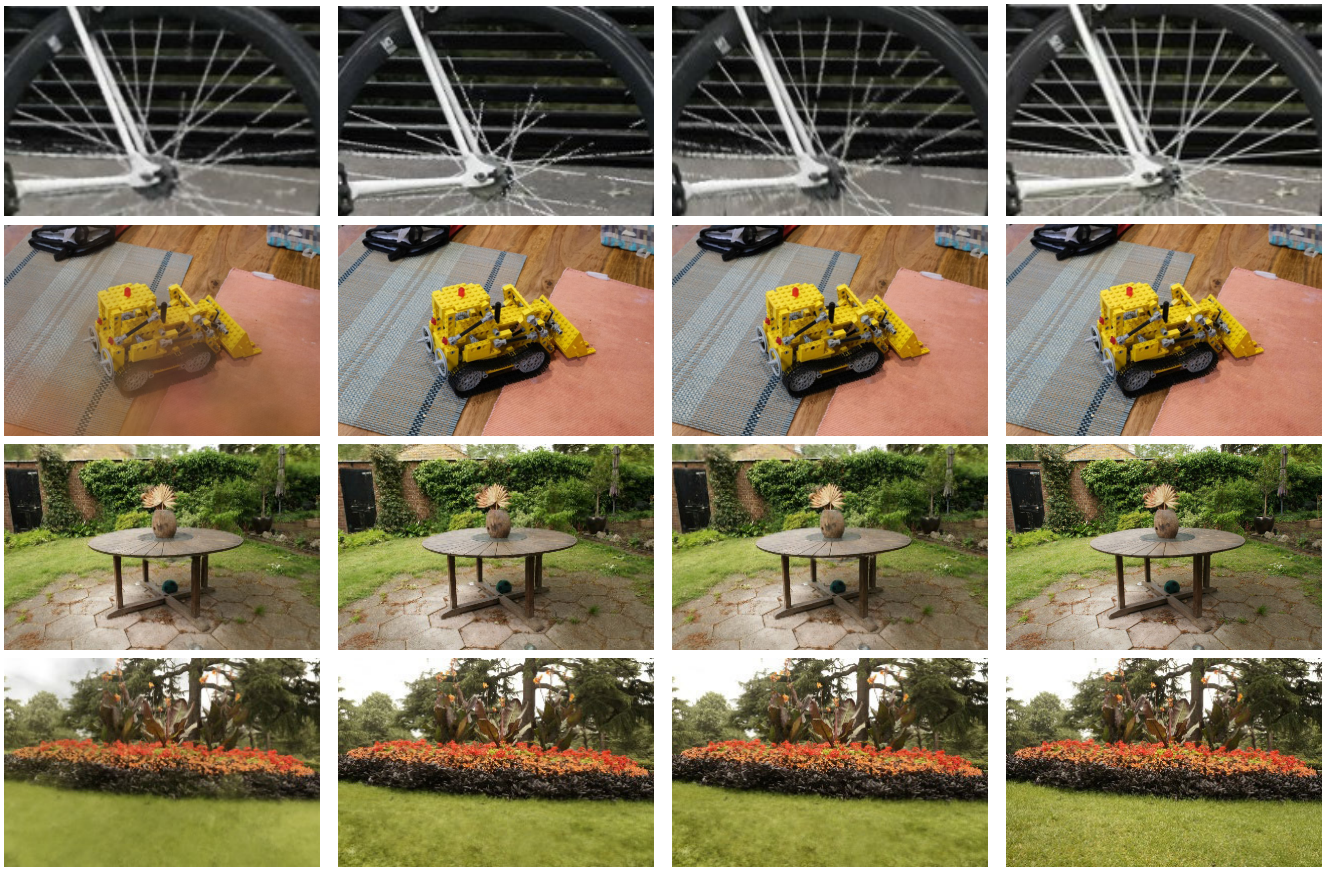
D. Ablation Study

In this section, we conduct ablative experiments to evaluate the design choices in our method. Specifically, we evaluate the SRF on the following aspects:

1) Space Warping Methods: We conduct experiments to compare our ERP mapping with another three space warping methods and two spherical projections: linear warping (NGP [22]), radial distortion (DONeRF [63]), space contraction (Mip-NeRF 360 [64]), sinusoidal projection, and Mercator projection.

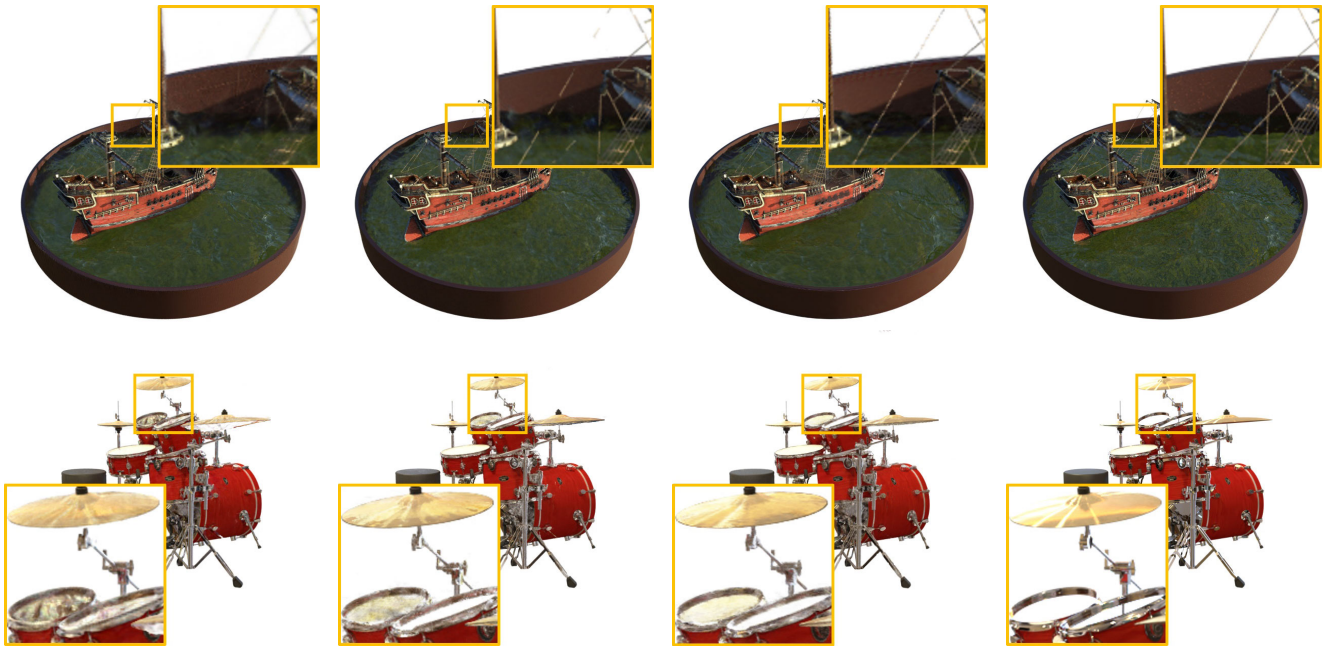
In Table V, the superior performance of our ERP mapping demonstrates it is more suitable for 360° unbounded scenes. That is because, our method not only contracts the distant space as radial distortion in DONeRF, but also increases the adjacent ERP voxel distance along with the radius in spherical mapping.

2) ERP Volume Visualization: We investigate our spherical radiance field by visualizing the learned ERPs in Fig. 8. It can be observed that our spherical radiance field can disentangle the whole scene into layered ones and encode them into spheres with different radii.



(a) NGP [22] (b) Mip-NeRF 360 [64] (c) SRF (Ours) (d) Ground-Truth

Fig. 6. Qualitative comparison on the Mip-NeRF-360 benchmark.



(a) Plenoxels [23] (b) NGP [22] (c) SRF (Ours) (d) Ground-Truth

Fig. 7. Qualitative comparison on the Synthetic-NeRF test set.

3) Multi-Scale Strategy & Hashing Encoding: We conducted experiments to study the effectiveness of our multi-scale strategy and hashing encoding in constructing ERP volumes. Specifically, we first constructed a baseline model by removing hashing encoding and employing a single-scale ERP volume. Then, we introduced model 2 by increasing the

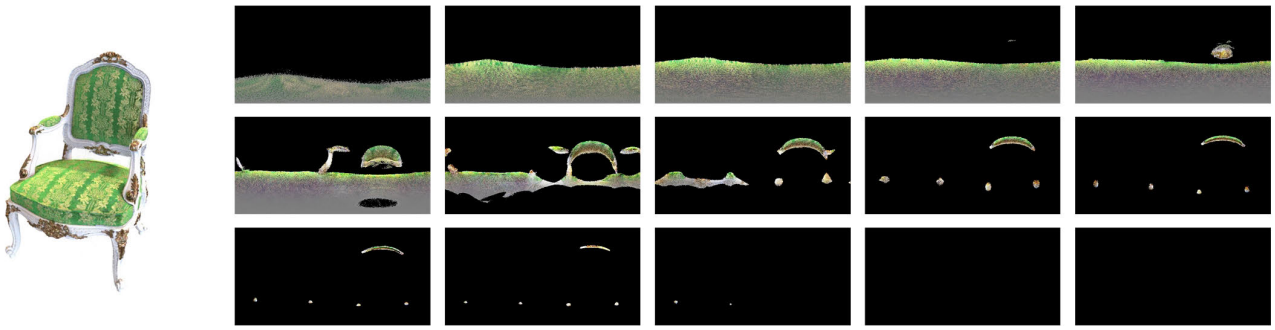
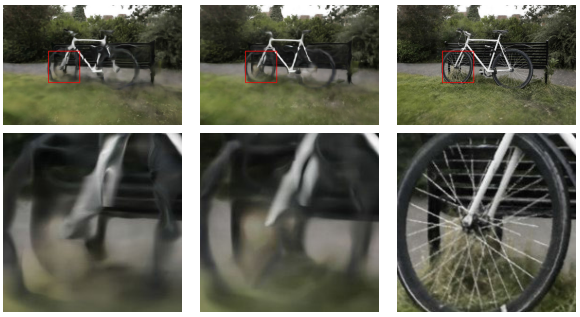


Fig. 8. Visualization of learned ERPs for spheres with different radii.

TABLE VI
RESULTS ACHIEVED ON THE *Bicylce* SCENE OF THE MIP-NERF-360 DATASET

#Scales	Hashing	PSNR↑	SSIM↑	#Params. (M)↓
1	×	9.31	0.001	0.5
2	×	17.96	0.412	268
4	×	19.43	0.430	277
4	✓	19.45	0.427	17
8	✓	20.83	0.460	29
16	✓	**24.52**	**0.689**	95
32	✓	22.24	0.543	105



(a) Model 1 (b) Model 2 (c) Model 3 (Ours)

Fig. 9. Visual results produced by models with different ERP volume representations.

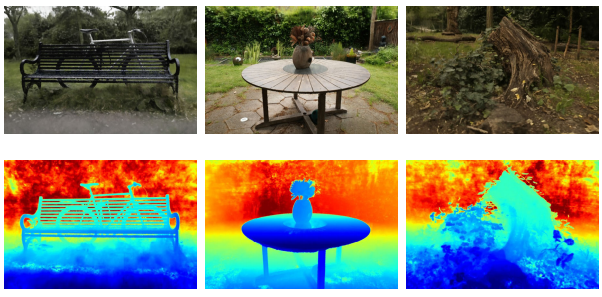


Fig. 10. Depth estimation results achieved on the Mip-NeRF-360 test set.

number of scales for ERP volumes to 4. Next, model 3 is developed by adopting hashing encoding. These models are compared to our SRF with full settings on the Mip-NeRF-360 dataset in Table VI.

As we can see, with only a single-scale ERP volume, the baseline model produces limited performance. By increasing the number of scales to 4, model 1 outperforms the

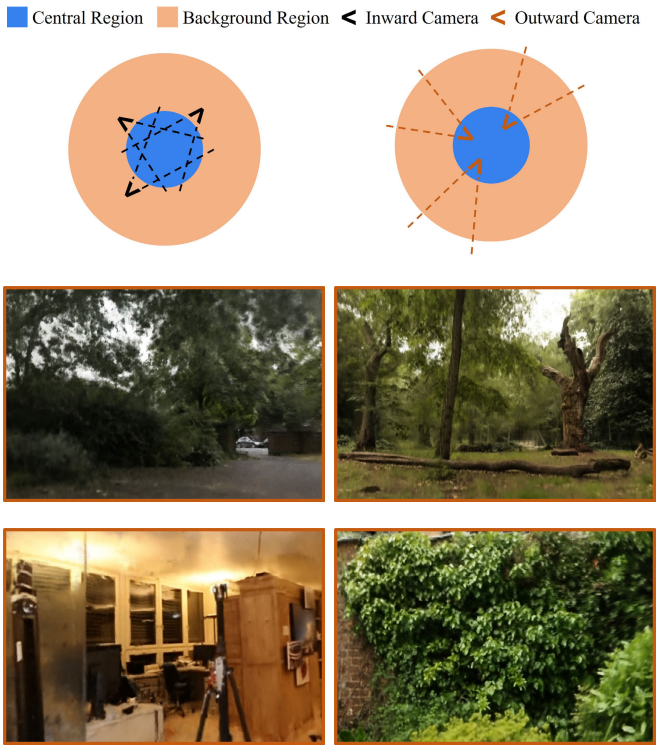


Fig. 11. Visualization of outward views produced by our method.

baseline with significant improvements. Meanwhile, the number of parameters is also increased from 0.5M to 277M. This demonstrates the effectiveness of multi-scale strategy in reconstructing finer details. By adopting hashing encoding, model 2 reduces over 93% parameters while maintaining comparable performance. With the help of hashing encoding, we increase the number of scales to 16 with an affordable number of parameters (i.e., 95M), which improves model 2 with notable margins.

Figure 9 further compares the visual results produced by models 1-3. As we can see, model 1 and model 2 produce inferior results with blurry artifacts. With more scales to model finer-grained details, our method produces results with higher perceptual quality, such as bicycle steel wire (as shown in Fig. 9(c)).

4) Depth Estimation: Our SRF inherits the capability of NeRF to capture 3D geometric information of the scene.

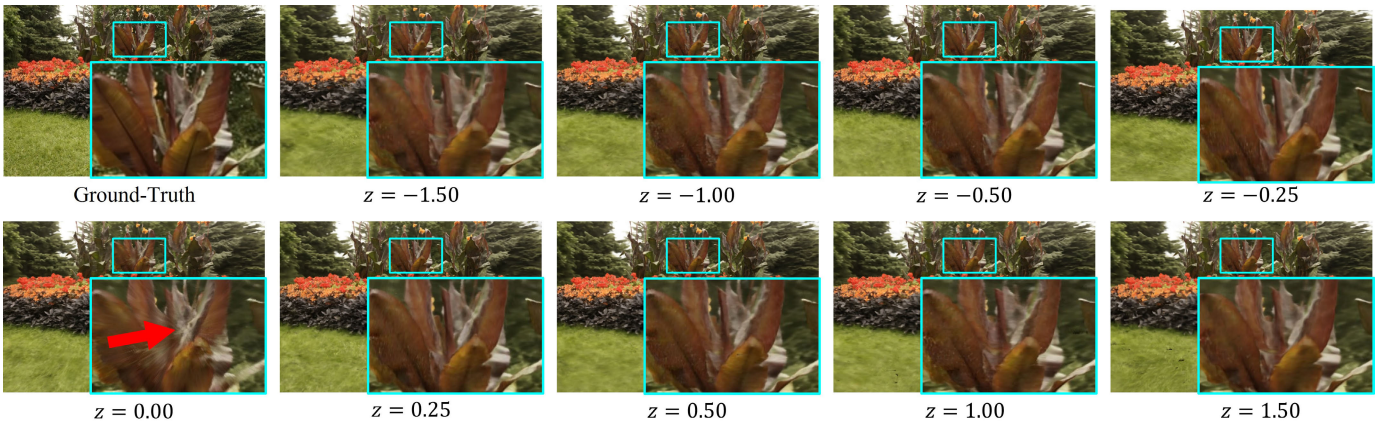


Fig. 12. Qualitative comparison on various ERP origins.

TABLE VII
RESULTS ACHIEVED ON VARIOUS ERP ORIGINS

z	-1.50	-1.00	-0.50	-0.25	0.00	0.25	0.50	1.00	1.50
PSNR	21.11	21.60	21.90	22.01	21.95	21.93	21.81	21.39	20.99
SSIM	0.530	0.560	0.581	0.591	0.586	0.583	0.574	0.552	0.525

To demonstrate this, we visualize the depth maps estimated from our spherical radiance field in Fig. 10. Note that, we train the proposed method without any depth supervision. It can be observed that our spherical radiance field produces promising depth estimation. This clearly demonstrates that our spherical radiance field can achieve a good 3D perception of the scene both foreground objects and backgrounds.

5) Results on Outward View Synthesis: Figure 11 shows the outward views produced by our method. Note that, only inward views capturing central objects are provided during training. As we can see, our method is capable of synthesizing high-quality outward background images from only inward views. This demonstrates the effectiveness of our method in modeling backgrounds in an unbounded scene.

6) Results on Various ERP Origins: We analyze the sensitivity to the choice of ERP origin on the *flowers* scene. The image poses are preprocessed such that the averaged position of cameras is $(0, 0, 0)$, all cameras are in a unit cube, and the $+z$ axis denotes the upward of the scene. We set the ERP origin as $(0, 0, z)$. Table VII shows the quantitative results with different z values. The best performance in terms of PSNR and SSIM is achieved when $z = -0.25$. The performance drops significantly when the ERP origin is far from the scene origin, e.g., a PSNR of 21.11 and 20.99 for $z = -1.50$ and $z = 1.50$, respectively. This is because, the ERP trainable features are sparse in the scene origin when it is far from ERP origin, resulting in blurred rendering. Figure 12 visualizes the rendering results achieved on various ERP origins. Some distortion artifacts occur near the ERP origin within small radii (see red arrow in case of $z = 0.00$). To remedy these artifacts, we can translate the ERP origins to an empty region ($z \geq 0.25$) or to be inside an object ($z \leq -0.25$).

V. CONCLUSION

In this paper, we propose a spherical radiance field (SRF) for efficient 360° unbounded novel view synthesis. Our SRF uses multiple concentric spheres with different radii to encode layered scenes into implicit representations. The spherical structure of our SRF facilitates our method to naturally fit 360° unbounded backgrounds, and the implicit representation enables our model to simultaneously model the central objects. Extensive experiments demonstrate the effectiveness of our SRF on both unbounded and bounded benchmark datasets in terms of both accuracy and efficiency.

REFERENCES

[1] Y. Wang, Q. Zhao, Y. Gan, and Z. Xia, "Joint-confidence-guided multi-task learning for 3D reconstruction and understanding from monocular camera," *IEEE Trans. Image Process.*, vol. 32, pp. 1120–1133, 2023.

[2] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 501–518.

[3] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1362–1376, Aug. 2010.

[4] J. Li, Z. Lu, Y. Wang, J. Xiao, and Y. Wang, "NR-MVSNet: Learning multi-view stereo based on normal consistency and depth refinement," *IEEE Trans. Image Process.*, vol. 32, pp. 2649–2662, 2023.

[5] H. Cui, D. Tu, F. Tang, P. Xu, H. Liu, and S. Shen, "VidSfM: Robust and accurate structure-from-motion for monocular videos," *IEEE Trans. Image Process.*, vol. 31, pp. 2449–2462, 2022.

[6] B. Mildenhall et al., "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–14, Jul. 2019.

[7] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–10, Nov. 2016.

[8] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field reconstruction using deep convolutional network on EPI," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6319–6327.

[9] Y. Wang et al., "Disentangling light fields for super-resolution and disparity estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 425–443, Jan. 2023.

[10] J. Jin and J. Hou, "Occlusion-aware unsupervised learning of depth from 4D light fields," *IEEE Trans. Image Process.*, vol. 31, pp. 2216–2228, 2022.

[11] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, Aug. 2020, pp. 405–421.

[12] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3D reconstruction in function space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4460–4470.

[13] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 165–174.

[14] J. T. Kajiya and B. P. Von Herzen, "Ray tracing volume densities," *ACM SIGGRAPH Comput. Graph.*, vol. 18, no. 3, pp. 165–174, Jul. 1984.

[15] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 5855–5864.

[16] D. Verbin, P. Hedman, B. Mildenhall, T. Zickler, J. T. Barron, and P. P. Srinivasan, "Ref-NeRF: Structured view-dependent appearance for neural radiance fields," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5491–5500.

[17] Y. Jiang et al., "AligNeRF: High-fidelity neural radiance fields via alignment-aware training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 46–55.

[18] S. Sabour, S. Vora, D. Duckworth, I. Krasin, D. J. Fleet, and A. Tagliasacchi, "RobustNeRF: Ignoring distractors with robust losses," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 20626–20636.

[19] T. Fujitomi, K. Sakurada, R. Hamaguchi, H. Shishido, M. Onishi, and Y. Kameda, "LB-NERF: Light bending neural radiance fields for transparent medium," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 2142–2146.

[20] C. Reiser, S. Peng, Y. Liao, and A. Geiger, "KiloNeRF: Speeding up neural radiance fields with thousands of tiny MLPs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14335–14345.

[21] A. Yu, R. Li, M. Tancik, H. Li, R. Ng, and A. Kanazawa, "PlenOctrees for real-time rendering of neural radiance fields," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 5752–5761.

[22] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, vol. 41, no. 4, pp. 102:1–102:15, Jul. 2022, doi: 10.1145/3528223.3530127.

[23] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5501–5510.

[24] C. Sun, M. Sun, and H.-T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 5459–5469.

[25] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "TensoRF: Tensorial radiance fields," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 333–350.

[26] W. Zhang, R. Xing, Y. Zeng, Y.-S. Liu, K. Shi, and Z. Han, "Fast learning radiance fields by shooting much fewer rays," *IEEE Trans. Image Process.*, vol. 32, pp. 2703–2718, 2023.

[27] C. Reiser et al., "MERF: Memory-efficient radiance fields for real-time view synthesis in unbounded scenes," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 1–12, Aug. 2023.

[28] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "PixelNeRF: Neural radiance fields from one or few images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4578–4587.

[29] Q. Wang et al., "IBRNet: Learning multi-view image-based rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4690–4699.

[30] A. Chen et al., "MVSNeRF: Fast generalizable radiance field reconstruction from multi-view stereo," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 14124–14133.

[31] 15 A. Trevithick and B. Yang, "GRF: Learning a general radiance field for 3D representation and rendering," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15182–15192.

[32] K. Rematas et al., "Urban radiance fields," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 12932–12942.

[33] H. Turki, D. Ramanan, and M. Satyanarayanan, "Mega-NeRF: Scalable construction of large-scale NeRFs for virtual fly-throughs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 12922–12931.

[34] M. Tancik et al., "Block-NeRF: Scalable large scene neural view synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 8248–8258.

[35] Z. Xie, J. Zhang, W. Li, F. Zhang, and L. Zhang, "S-NeRF: Neural radiance fields for street views," in *Proc. Int. Conf. Learn. Represent.*, 2023.

[36] L. Xu et al., "Grid-guided neural radiance fields for large urban scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 8296–8306.

[37] Q. Dai, Y. Zhu, Y. Geng, C. Ruan, J. Zhang, and H. Wang, "GraspNeRF: Multiview-based 6-DoF grasp detection for transparent and specular objects using generalizable NeRF," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 1757–1763.

[38] A. Byravan et al., "NeRF2Real: Sim2real transfer of vision-guided bipedal motion skills using neural radiance fields," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 9362–9369.

[39] M. Adamkiewicz et al., "Vision-only robot navigation in a neural radiance world," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4606–4613, Apr. 2022.

[40] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "DreamFusion: Text-to-3D using 2D diffusion," in *Proc. Int. Conf. Learn. Represent.*, 2023.

[41] G. Metzer, E. Richardson, O. Patashnik, R. Giryes, and D. Cohen-Or, "Latent-NeRF for shape-guided generation of 3D shapes and textures," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 12663–12673.

[42] L. Liu, J. Gu, K. Z. Lin, T.-S. Chua, and C. Theobalt, "Neural sparse voxel fields," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 15651–15663.

[43] R. Yi, Y. Huang, Q. Guan, M. Pu, and R. Zhang, "Learning from pixel-level label noise: A new perspective for semi-supervised semantic segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 623–635, 2022.

[44] T. Zhou, L. Li, X. Li, C.-M. Feng, J. Li, and L. Shao, "Group-wise learning for weakly supervised semantic segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 799–811, 2022.

[45] X. Zhang, W. Zhao, W. Zhang, J. Peng, and J. Fan, "Guided filter network for semantic image segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 2695–2709, 2022.

[46] X. He, J. Liu, W. Wang, and H. Lu, "An efficient sampling-based attention network for semantic segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 2850–2863, 2022.

[47] T. Chen, X. Hu, J. Xiao, G. Zhang, and S. Wang, "Accurate instance segmentation via collaborative learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 1880–1884.

[48] C. B. Kuhn, M. Hofbauer, G. Petrovic, and E. Steinbach, "Reverse error modeling for improved semantic segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 106–110.

[49] Q. Wang, S. Zhang, and X. He, "Robust temporally-coherent strategy for few-shot video instance segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 251–255.

[50] Z. Liang, X. Dai, Y. Wu, X. Jin, and J. Shen, "Multi-granularity context network for efficient video semantic segmentation," *IEEE Trans. Image Process.*, vol. 32, pp. 3163–3175, 2023.

[51] A. Karnewar, T. Ritschel, O. Wang, and N. Mitra, "ReLU fields: The little non-linearity that could," in *Proc. ACM SIGGRAPH Conf.*, Aug. 2022, pp. 1–9.

[52] B. Attal, S. Ling, A. Gokaslan, C. Richardt, and J. Tompkin, "MatryODShka: Real-time 6DoF video view synthesis using multi-sphere images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2020, pp. 441–459.

[53] J. Chibane, A. Bansal, V. Lazova, and G. Pons-Moll, "Stereo radiance fields (SRF): Learning view synthesis for sparse views of novel scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7911–7920.

[54] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-NeRF: Neural radiance fields for dynamic scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10318–10327.

[55] K. Park et al., "HyperNeRF: A higher-dimensional representation for topologically varying neural radiance fields," *ACM Trans. Graph.*, vol. 40, no. 6, pp. 1–12, Dec. 2021.

[56] K. Park et al., "Nerfies: Deformable neural radiance fields," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 5845–5854.

[57] K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger, "GRAF: Generative radiance fields for 3D-aware image synthesis," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 20154–20166.

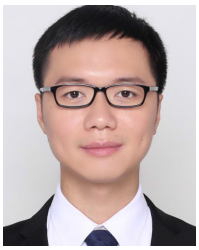
[58] M. Niemeyer and A. Geiger, "GIRAFFE: Representing scenes as compositional generative neural feature fields," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11453–11464.

[59] D. B. Lindell, J. N. P. Martel, and G. Wetzstein, "AutoInt: Automatic integration for fast neural volume rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14556–14565.
[60] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely, "Stereo magnification: Learning view synthesis using multiplane images," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 1–12, Jul. 2018.
[61] J. Flynn et al., "DeepView: View synthesis with learned gradient descent," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2362–2371.
[62] K. Zhang, G. Riegler, N. Snavely, and V. Koltun, "NeRF++: Analyzing and improving neural radiance fields," 2020, *arXiv:2010.07492*.
[63] T. Neff et al., "DONeRF: Towards real-time rendering of compact neural radiance fields using depth Oracle networks," *Comput. Graph. Forum*, vol. 40, no. 4, pp. 45–59, 2021.
[64] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-NeRF 360: Unbounded anti-aliased neural radiance fields," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 5470–5479.
[65] M. Tancik et al., "Nerfstudio: A modular framework for neural radiance field development," in *Proc. ACM SIGGRAPH*, Jul. 2023, pp. 1–12.
[66] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Zip-NeRF: Anti-aliased grid-based neural radiance fields," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 19697–19705.
[67] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian splatting for real-time radiance field rendering," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 1–14, Aug. 2023.
[68] P. Wang et al., "F2-NeRF: Fast neural radiance field training with free camera trajectories," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 4150–4159.
[69] T. Takikawa et al., "Neural geometric level of detail: Real-time rendering with implicit 3D shapes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11358–11367.
[70] T. Müller. (2021). *Tiny-CUDA-NN*. [Online]. Available: https://github.com/NVlabs/tiny-cuda-nn
[71] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[72] K. Yücer, A. Sorkine-Hornung, O. Wang, and O. Sorkine-Hornung, "Efficient 3D object segmentation from densely sampled light fields with applications to 3D reconstruction," *ACM Trans. Graph.*, vol. 35, no. 3, pp. 1–15, Jun. 2016.
[73] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–13, 2017.
[74] C. Sun, M. Sun, and H.-T. Chen, "Improved direct voxel grid optimization for radiance fields reconstruction," 2022, *arXiv:2206.05085*.
[75] Q. Xu et al., "Point-NeRF: Point-based neural radiance fields," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 5438–5448.

Minglin Chen (Graduate Student Member, IEEE) received the B.E. degree in communication engineering from South China Normal University (SCNU), Guangzhou, China, in 2017, and the M.E. degree in software engineering from the University of Chinese Academy of Science (UCAS), Beijing, China, in 2020. He is currently pursuing the Ph.D. degree with Sun Yat-sen University (SYSU). His research interests include neural rendering and 3D vision.

Longguang Wang received the B.E. degree in electrical engineering from Shandong University (SDU), Jinan, China, in 2015, and the Ph.D. degree in information and communication engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2022. His current research interests include low-level vision and 3D vision.

Yinjie Lei (Senior Member, IEEE) received the M.S. degree in image processing from Sichuan University (SCU), China, in 2009, and the Ph.D. degree in computer vision from The University of Western Australia (UWA), Australia, in 2013. Since 2017, he has been the Vice Dean of the College of Electronics and Information Engineering, SCU, where he is currently a Professor. His research interests include 3D biometrics, object recognition, and semantic segmentation.



Zilong Dong received the B.S. and Ph.D. degrees in computer science from Zhejiang University, in 2004 and 2010, respectively. He is currently a Staff Researcher with the Alibaba Group. His research interests include large-scale structure-from-motion, SLAM, 3D reconstruction and segmentation, augmented reality, and neural rendering.

Yulan Guo (Senior Member, IEEE) received the B.E. and Ph.D. degrees from the National University of Defense Technology (NUDT) in 2008 and 2015, respectively. He has authored over 200 papers at highly referred journals and conferences. His research interests include 3D vision, low-level vision, and machine learning. He is a Senior Member of ACM. He served as the Area Chair for ECCV 2024, CVPR 2023/2021, ICCV 2021, NeurIPS 2024, and ACM Multimedia 2021. He served as an Associate Editor for IEEE TRANSACTIONS ON IMAGE PROCESSING, *IET Computer Vision*, *IET Image Processing*, and *Computers and Graphics*.