Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

SPECIAL ISSUE PAPER

# De-NeRF: Ultra-High-Definition NeRF with Deformable Net Alignment

**Jianing Hou**[1,2] | **Runjie Zhang**[3] | **Zhongqi Wu**[4] | **Weiliang Meng**[2,1] | **Xiaopeng Zhang**[2,1] | **Jianwei Guo**[2,1]

[1]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China

[2]State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China

[3]UC San Diego, University of California San Diego, La Jolla, USA

[4]Institutes of Science and Development, Chinese Academy of Sciences, Beijing, China

**Correspondence**

Corresponding author: Jianwei Guo.

Email: jianwei.guo@nlpr.ia.ac.cn

Funding Information

This research was supported by the National Natural Science Foundation of China (Nos. 62172416, U22B2034, U21A20515, 62262043, 62376271), Guangdong Basic and Applied Basic Research Foundation (2023B1515120026), Beijing Natural Science Foundation (No. L231013), and CAS Youth Innovation Promotion Association (2022131).

## Abstract

Neural Radiance Field (NeRF) can render complex 3D scenes with viewpoint-dependent effects. However, less work has been devoted to exploring its limitations in high-resolution environments, especially when upscaled to ultra-high resolution (*e.g.,* 4k). Specifically, existing NeRF-based methods face severe limitations in reconstructing high-resolution real scenes, *e.g.,* a large number of parameters, misalignment of the input data, and over-smoothing of details. In this paper, we present a novel and effective framework, called *De-NeRF*, based on NeRF and deformable convolutional network, to achieve high-fidelity view synthesis in ultra-high resolution scenes: 1) marrying the deformable convolution unit which can solve the problem of misaligned input of the high-resolution data. 2) presenting a density sparse voxel-based approach which can greatly reduce the training time while rendering results with higher accuracy. Compared to existing high-resolution NeRF methods, our approach improves the rendering quality of high-frequency details and achieves better visual effects in 4K high-resolution scenes.

**KEYWORDS**

Neural radiance fields, Deformable convolution net, Voxel-based embedding

## 1 | INTRODUCTION

Synthesizing novel views of complex scenes from sparse observed images is a long-standing problem in computer graphics and vision. Recently, NeRF[1] and its variants[2,3,4,5] have provided a new approach for such 3D scene reconstruction and rendering task through deep neural networks. They possess strong performance in learning geometric 3D representations from images, and the resulting high-quality representations of the scene compare well with traditional viewpoint interpolation methods[3]. However, current 3D reconstruction techniques usually use low-resolution datasets (*e.g.,* 1K HD format), compared to which high-resolution data tend to contain richer and more accurate detail rendering. Since ultra-high resolution is becoming increasingly popular as a standard for recording and displaying images and videos, 3D reconstruction of ultra-high-resolution scenes is essential for many applications, such as providing a more immersive virtual experience in AR and VR.

The original NeRF[1] uses a simple 8-layer MLP to associate each 3D position given a viewing direction with its corresponding radial color and volume density, whereas achieving view-dependent effects requires querying a large network hundreds of times through ray cast of each pixel. However, it is difficult for NeRF's simple network to synthesize novel high-resolution views directly. Regarding this phenomenon, we have conducted experiments on the rendering effect of NeRF with different resolutions, as shown in Fig. 1. We found that the clarity of details and the accuracy of the rendered images do not improve as the resolution of the training data increases, or even decreases, which poses an obstacle to the reconstruction of high-resolution real scenes. In

  

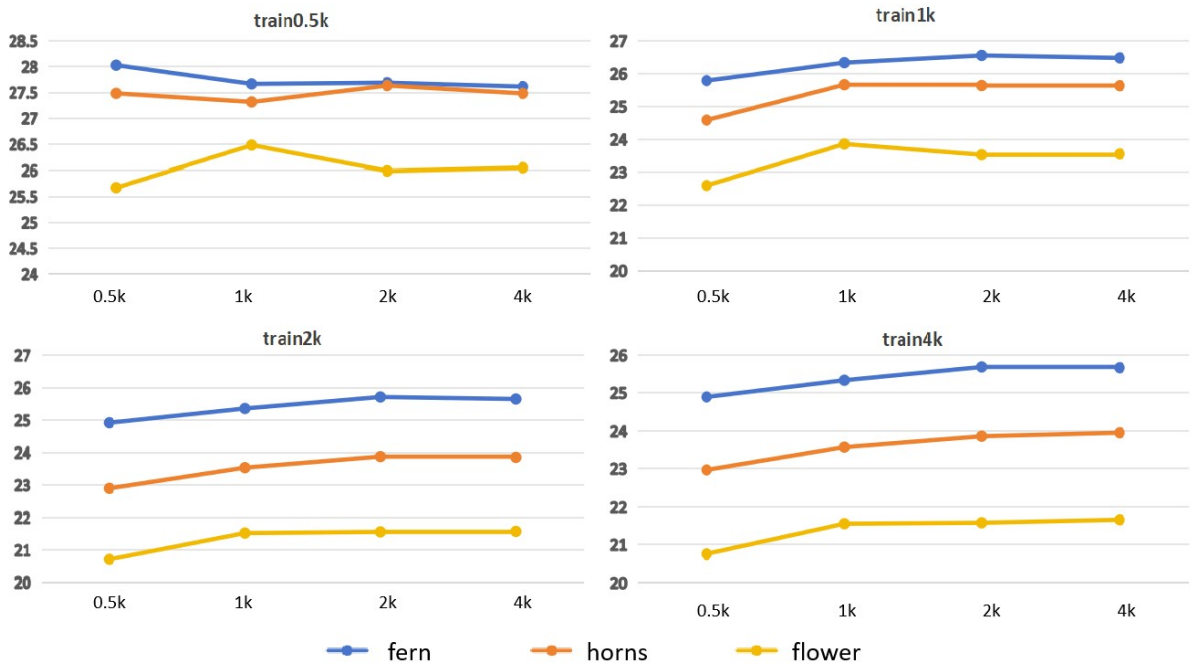


**FIGURE 1** Illustrating the motivations of our work. We applied the original NeRF for rendering pictures on three scenes with different resolutions, and we found that the PSNR metrics of the rendered images are essentially unchanged, or even declining when the resolution of the picture is increasing (2k–4k data in the figure). We hypothesize that NeRF still has a bottleneck in processing high-resolution scenes.

this work, we focus on the new high-resolution view synthesis task and investigate the challenges of achieving high-fidelity reconstruction results at high resolution.

NeRF requires ray casting of every pixel of every viewpoint image, leading to a long training time on the ultra-high definition (UHD) dataset. To address this problem, previous methods (NSVF[4], DVGO[6], and instant-ngp[7]) use a bounded implicit representation of voxels consisting of a set of voxels in a sparse voxel octree. Using sparse voxels could accelerate rendering significantly at inference time by skipping empty voxels with no scene content. However, we found that by raising the number of voxel grid layers and the resolution of the voxel grid, we can make the rendered images more accurate.

Moreover, NeRF needs accurate camera poses for still scenes if it is to render high-quality images with high-frequency details. However, in practice, camera poses of real scene images recovered by the COLMAP program based on the SFM algorithm[8] inevitably contain pixel-point inaccuracies after they have been captured. These inaccuracies are not noticeable when training low-resolution images but lead to blurring when NeRF is trained with higher-resolution inputs. In addition, the real scene being photographed may also contain the motion of non-rigid objects, such as moving clouds and plants. Such inclusion of motion breaks the assumption of a static scene and reduces the accuracy of the estimated camera poses. Due to inaccurate camera poses and scene motions, the rendered output of NeRF tends to be slightly misaligned with the ground truth (GT) image.

To address these issues, we propose *De-NeRF*, a new neural network that corrects alignment errors and can be trained more efficiently than 4k-NeRF[9] and AligNeRF[10]. Our approach combines explicit voxel grids and NeRF as the overall pipeline. To avoid a drastic increase in the compute resources of NeRF under high-resolution data, we adopt a multi-layer sparse voxel grid structure, which can greatly speed up the rendering during inference by skipping empty voxels with no scene content to ensure that the size of the overall work overhead and will not take as long as the training time of the super-resolution module behind the 4k-NeRF[9]. We add the deformable feature unit module to the back end of the NeRF training. Instead of aligning at pixel-level as in the flow-aligned AligNeRF[10], we operate in the feature space, which will accelerate the error training of the deformable alignment unit and can save training resources even more in the case of ultra-high-resolution data.

In summary, the main contributions of this work include the following:

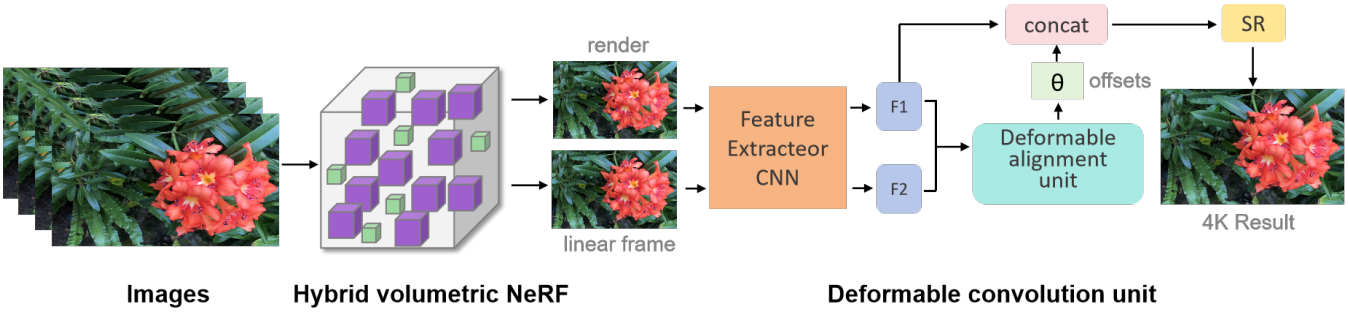


**FIGURE 2** Network architecture of our proposed *De-NeRF*. The high-resolution image first passes through the *hybrid volumetric representations* module, due to its voxel grid embedding feature representation it can generate the rendering frames needed by the next alignment module quickly, then the *deformable alignment unit* module calculates the offset of the features and aligns them, and finally boosts the resolution to complete the final rendering process.

- A deformable alignment unit that employs a deformable network to correct the offset of the feature-level image, saving more compute resources as opposed to employing pixel-level offset correction (*e.g.,* AligNeRF[10]).
- A direct training method, avoiding the need to use flow-align to align the rendered image with the GT image multiple times, can simplify the training process.
- A NeRF strategy using voxel-based feature encoding representation, which can significantly speed up the training and improve the accuracy of the final rendered image after increasing the voxel grid density.

## 2 | RELATED WORK

### 2.1 | Neural Radiance Field

NeRF utilizes a deep neural network to directly learn continuous mapping from spatial coordinates and view orientations to view-dependent color and volume densities and obtains pixel colors through volume rendering techniques. Implicit neural representations associated with it have demonstrated its effectiveness in representing shapes and scenes, which typically utilize multi-layer perceptrons (MLPs) to encode signed distance fields [11,12], occupancy [13,14,15], or volume density [16,1]. These methods utilize microscopic rendering[17,18] to reconstruct the geometry and appearance of objects and scenes [4,16,19]. Several optimized extensions of NeRF have also emerged, e.g., few view inputs [3,20], reconstruction of non-rigid scenes [5,21,22] and object categories [20,23,24]. [24,25,26] accelerates the rendering speed from initially multi-second to millisecond. [27,5,21,28] introduced a volumetric radiation field and successfully reduced the training cost by an order of magnitude. Several approaches have focused on improving the rendering quality of NeRF[1]. [2] introduced mipmap for anti-aliasing, while [15,19] improved NeRF's ability to model surfaces with high reflectivity.

### 2.2 | Super-Resolution

This work is very similar to the super-resolution task requirements in 2D images. Currently mainstream deep learning-based approach using CNNs is to learn the relationship between HR and LR images in CNNs by minimizing the mean square error between SR images and GT images[27,29]. It has also become popular to introduce generative adversarial networks (GANs) [30] in super-resolution tasks, which can produce high-resolution details that match human intuition through adversarial learning [25,26,31]. However, the pixels generated by the network may have a large disparity compared to the real ground truth, resulting in a decrease in accuracy [9]. Furthermore, these 2D methods have some problems when applied directly to 3D reconstruction: they all obtain feature information from large-scale datasets or existing HR and LR pairs and do not take into account view consistency, which is sub-optimal for the current new view synthesis task.[32] uses the flow-based pixel-level alignment of neighboring frames in the video, but due to the pixel-level in high-resolution video consumes substantial compute resources, the subsequent [33,34] and other

work to solve this problem by using the flow-free feature-level operation before entering the neural network to downsample the image into the feature space, which achieves better results. The deformable convolution module[35] in the deformable convolution network work used in these two works to align pixels in neighboring frames is also the inspiration for our work.

## 2.3 | Super-Resolution for 3D Reconstruction

Our aim is to reconstruct the scene in 3D from a set of images, and instead of performing the super-resolution (SR) task in the 2D image space, we perform the [36,37,38,9,10] in the 3D scene space. [38] is able to synthesize higher-resolution images than low-resolution images by super-sampling strategy. [36,37] are classical methods for optimizing geometry and texture. [37] integrates low-resolution depth, color, and RGB-D sensors into SR key frames then fuses these key frames into a texture map, and [36] uses convolution with a gaussian kernel to describe their SR process. [9] uses SR networks at the end of NeRF to output a more detailed image. All of the above work can't depart from the idea of 2D SR tasks. [10] suggests that the problem that causes blurring of the new perspective for high-resolution NeRF rendering comes from the fact that the real captured UHD dataset has misalignment due to its bit-pose being generated by COLMAP. We continue along this line in this work and make a more precise proof of this problem (e.g., Fig. 1).

# 3 | PROBLEM STATEMENT AND OVERVIEW

The input to our algorithm consists of high-resolution images and the corresponding five-dimensional coordinate poses (3D location $\mathbf{x} = (x, y, z)$ and 2D viewing direction $(\theta, \phi)$). Such image coordinates in real scene datasets are usually generated using COLMAP, which may be unaligned with small biases. Our goal is to render the new viewpoint image as the original NeRF and to minimize the MSE with the GT image:

$$\mathcal{L}_{\text{photo}} = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \left\| \hat{C}(\boldsymbol{r}) - C(\boldsymbol{r}) \right\|_2^2, \tag{1}$$

where $\mathbf{R}$ is the set of rays in each batch , $C(\boldsymbol{r})$ is observed pixel color of GT and $\hat{C}(r)$ is the rendered color.

The major steps of our algorithm are shown in Fig. 2. We start by applying the explicit hybrid volumetric grid[8,7] to encode the images of the UHD dataset, and we increase the number of voxel grid layers as well as raise the resolution of the grid at each layer, thus improving the accuracy of the final generated images. Next, the core of our approach is that we correct the offsets between different video frames based on the *deformable network*, which are widely used in the field of computer vision and video super-resolution. Therefore, the rendered images of the new views generated based on the NeRF principle are corrected according to our proposed *deformable alignment unit* which learns the offsets. In addition, the unaligned feature of the input image can be aligned and ensure the generation of more accurate photos.

# 4 | METHODOLOGY

We introduce the method of voxel-based NeRF and discuss the limitations of modeling and rendering scenes with extremely high resolution. In the following, we will present principles of implementation of each part in our framework and introduce the training strategy with loss functions.

## 4.1 | NeRF

The main idea of NeRF takes a 3D point position $\mathbf{X} = (x, y, z)$ and a viewing direction $\mathbf{d}(\theta, \phi)$ as input and learns a continuous mapping function to estimate the color $\mathbf{c} = (r, g, b)$ and volume density $\sigma$.NeRF accomplishes view synthesis by training a continuous mapping function that predicts the color $\mathbf{c} \in \mathbb{R}^3$ and volume density $\sigma \in \mathbb{R}$ of a 3D point position $\mathbf{x} \in \mathbb{R}^3$ and a viewing direction $\mathbf{d} \in \mathbb{R}^3$. This mapping function, denoted as $\Phi : (\mathbf{x}, \mathbf{d}) \mapsto (\mathbf{c}, \sigma)$, allows for estimating the appearance and properties of each point in the scene. When rendering an image with a given camera pose, the expected color $\widehat{\mathbf{C}}(\mathbf{r})$ of a camera ray $\mathbf{r} = \mathbf{o} + t\mathbf{d}$, where $\mathbf{o}$ represents the camera center, is determined by sampling multiple points along the ray and integrating

their colors. This integration process approximates the volumetric rendering integral[39], resulting in a synthesized image that captures the desired viewpoint.

$$\widehat{\mathbf{C}}(\mathbf{r}) = \sum_{i=1}^{N} T_i \cdot \alpha_i \cdot \mathbf{c}_i, \tag{2}$$

$$\alpha_i = 1 - \exp(-\sigma_i \delta_i), \quad T_i = \prod_{j=1}^{i-1}(1 - \alpha_j), \tag{3}$$

where $\alpha_i$ denotes the ray termination probability at the point $i$, $\delta_i = t_{i+1} - t_i$ represents the distance between two adjacent points, and $T_i$ indicates the accumulated transmittance when reaching $i$.

## 4.2 | Hybrid Volumetric Representations

4.2.0.1 | *Voxel-grid representation.*
A voxel-grid representation explicitly models the modalities of interest (*e.g.,* , density, color, or feature) in its grid cells. Such an explicit scene representation is efficient for querying for any 3D positions via interpolation:

$$\text{interp}(x, V) : \left(\mathbb{R}^3, \mathbb{R}^{C \times N_x \times N_y \times N_z}\right) \rightarrow \mathbb{R}^C, \tag{4}$$

where $x$ is the queried 3D point, $V$ is the voxel grid, $C$ is the dimension of the modality, and $N_x \cdot N_y \cdot N_z$ is the total number of voxels. Trilinear interpolation is applied if not specified otherwise.

4.2.0.2 | *Density voxel grid for volume rendering.*
We instantiate the front network based on the formulation defined in the DVGO[6] and instant-ngp[7], where voxel-grid-based representations are learned to encode geometric structure explicitly,

$$(\mathbf{x}, \mathbf{V}) : \left(\mathbb{R}^3, \mathbb{R}^{N_c \times N_x \times N_y \times N_z}\right) \rightarrow \mathbb{R}^{N_c}, \tag{5}$$

where$N_c$ represents the channel dimension for density ($N_c = 1$) and color modality. For each sampled point, the density is estimated using trilinear interpolation with a softplus activation function, given by $\sigma = \text{softplus}\left(\text{interp}\left(\mathbf{x}, \mathbf{V}_d\right)\right)$. The colors are estimated using a shallow Multi-Layer Perceptron (MLP),

$$\begin{aligned} \mathbf{c} &= f_{\text{MLP}}\left(\text{interp}\left(\mathbf{x}, \mathbf{V}_c\right), \mathbf{x}, \mathbf{d}\right) \\ &= f_{\text{RGB}}\left(g_\theta(\text{interp}(\mathbf{x}, \mathbf{V}_c), \mathbf{x}, \mathbf{d})\right), \end{aligned} \tag{6}$$

where $g_\theta(\cdot)$ extracts volumetric features for color information, and $f_{\text{RGB}}$ denotes the mapping (with one or multiple layers) from the features to RGB images. The output $\mathbf{g} = g(\theta; \mathbf{x}, \mathbf{d})$ represents the volumetric feature for a point $\mathbf{x}$ with a given viewing direction $\mathbf{d}$. By accumulating the features of the sampled points along the ray $\mathbf{r}$, we obtain the descriptor for each ray (or pixel) following Equation 2,

$$\mathbf{f}(\mathbf{r}) = \sum_{i=1}^{N} T_i \cdot \alpha_i \cdot \mathbf{g}_i. \tag{7}$$

To leverage the geometric properties encoded in the encoder, we also generate a depth map by estimating the depth along the camera axis for each ray $\mathbf{r}$. This is achieved using the following equation:

$$M(\mathbf{r}) = \sum_{i=1}^{N} T_i \cdot \alpha_i \cdot t_i, \tag{8}$$

where $t_i$ denotes the distance of the sampling point $i$ to the camera center as in Eqn.2. The estimated depth map provides a strong guidance for understating the 3D structure of a scene, e.g., nearby pixels on the image plane may be far away in the original 3D space. Assume the spatial dimension is $H' \times W'$, the formed feature maps $\mathbf{F}_{\text{en}} \in \mathbb{R}^{C' \times H' \times W'}$ and the depth map $\mathbf{M} \in \mathbb{R}^{H' \times W'}$ are fed into the decoder for pursuing high-fidelity reconstruction of fine details.

## 4.3 | Deformable Convolution Network

The 2D convolution consists of two steps: 1) sampling using a regular grid $\mathcal{R}$ over the input feature map $\mathbf{x}$; 2) summation of sampled values weighted by $\mathbf{w}$. The grid $\mathcal{R}$ defines the receptive field size and dilation. For example,

$$\mathcal{R} = \{(-1,-1),(-1,0),\ldots,(0,1),(1,1)\}$$

defines a $3 \times 3$ kernel with dilation 1. For each location $\mathbf{p}_0$ on the output feature map $\mathbf{y}$, we have

$$\mathbf{y}(\mathbf{p}_0) = \sum_{\mathbf{p}_n \in \mathcal{R}} \mathbf{w}(\mathbf{p}_n) \cdot \mathbf{x}(\mathbf{p}_0 + \mathbf{p}_n), \tag{9}$$

where $\mathbf{p}_n$ enumerates the locations in $\mathcal{R}$. In deformable convolution, the regular grid $\mathcal{R}$ is augmented with offsets $\{\Delta\mathbf{p}_n | n = 1, ..., N\}$, where $N = |\mathcal{R}|$. Eq. (9) becomes

$$\mathbf{y}(\mathbf{p}_0) = \sum_{\mathbf{p}_n \in \mathcal{R}} \mathbf{w}(\mathbf{p}_n) \cdot \mathbf{x}(\mathbf{p}_0 + \mathbf{p}_n + \Delta\mathbf{p}_n). \tag{10}$$

Now, the sampling is on the irregular and offset locations $\mathbf{p}_n + \Delta\mathbf{p}_n$. As the offset $\Delta\mathbf{p}_n$ is typically fractional, Eq. (10) is implemented via bilinear interpolation as

$$\mathbf{x}(\mathbf{p}) = \sum_{\mathbf{q}} G(\mathbf{q}, \mathbf{p}) \cdot \mathbf{x}(\mathbf{q}), \tag{11}$$

where $\mathbf{p}$ denotes an arbitrary (fractional) location ($\mathbf{p} = \mathbf{p}_0 + \mathbf{p}_n + \Delta\mathbf{p}_n$ for Eq. (10)), $\mathbf{q}$ enumerates all integral spatial locations in the feature map $\mathbf{x}$, and $G(\cdot,\cdot)$ is the bilinear interpolation kernel. Note that $G$ is two-dimensional. It is separated into two one-dimensional kernels as

$$G(\mathbf{q}, \mathbf{p}) = g(q_x, p_x) \cdot g(q_y, p_y), \tag{12}$$

where $g(a, b) = max(0, 1 - |a - b|)$. Eq. (11) is fast to compute as $G(\mathbf{q}, \mathbf{p})$ is non-zero only for a few $\mathbf{q}$s.

## 4.4 | Loss Functions

Our De-NeRF model integrates the above-described modules in a unified network architecture, as shown in Fig. 2, which is trained in a supervised fashion by an efficient loss function.

We found that only using distortion-oriented loss (e.g., MSE, $\ell_1$ and Huber loss) as objective tends to produce blurry or over-smoothed visual effects on fine details. In order to solve the problem, we add the adversarial loss and the perceptual loss to regularize fine detail synthesis. We use $\ell_1$ loss instead of MSE for directly supervising the reconstruction of high-frequency details,

$$\mathcal{L}_1 = \frac{1}{N_p^2} \left| \mathbf{C}(\hat{\mathbf{p}}) - \mathbf{C}(\mathbf{p}) \right|. \tag{13}$$

We add an auxiliary MSE loss to facilitate the training of the encoder with down-scaled training views, the ray features produced by the encoder are fed into an extra fully connected layer to regress RGB values in the lower-resolution images. The overall training objective is defined as,

$$\mathcal{L} = \lambda_h \mathcal{L}_1 + \lambda_l \mathcal{L}_{\text{MSE}}^l. \tag{14}$$

where $\lambda_h$ and $\lambda_l$ denote the hyper-parameters for weighting the losses.

## 4.5 | Implementation Details

Our De-NeRF framework is implemented in PyTorch on a server equipped with an Intel Xeon Gold 6226R CPU and NVIDIA RTX A6000 (48 GB memory) graphics cards. The operating system is Ubuntu 20.04.3. So we train the network via Adam with the batch size $B = 1024$. In our network, both learning rate of Hybrid Volumetric Representations and Deformable Alignment

Unit is 0.1, and the learning rate of the RGB net (MLP) is 0.001. The total number of iterations is 10000. We multiply the learning rate by 0.1 per 1000 iterations.

**TABLE 1** Quantitative performance comparison of different registration methods on LLFF dataset. The best results are marked in red color, and the second best results are in blue color.

| Metrics | Methods | fern | flower | fortress | horns | leaves | orchids | room | trex |
|---|---|---|---|---|---|---|---|---|---|
| PSNR↑ | NeRF | 33.57 | 34.77 | 37.92 | 33.95 | 36.78 | 34.03 | 39.21 | 32.15 |
| | 4K-NeRF | 34.73 | 36.95 | 38.65 | 35.10 | 38.71 | 35.63 | 40.88 | 34.23 |
| | ours | 34.84 | 37.11 | 38.68 | 35.24 | 38.85 | 35.79 | 41.14 | 34.42 |
| SSIM ↑ | NeRF | 0.647 | 0.783 | 0.759 | 0.704 | 0.581 | 0.566 | 0.783 | 0.681 |
| | 4K-NeRF | 0.721 | 0.843 | 0.772 | 0.791 | 0.647 | 0.751 | 0.881 | 0.701 |
| | ours | 0.767 | 0.848 | 0.767 | 0.772 | 0.691 | 0.767 | 0.893 | 0.736 |
| Lpips ↓ | NeRF | 0.529 | 0.575 | 0.495 | 0.492 | 0.520 | 0.427 | 0.432 | 0.549 |
| | 4K-NeRF | 0.307 | 0.289 | 0.258 | 0.296 | 0.231 | 0.242 | 0.394 | 0.192 |
| | ours | 0.241 | 0.294 | 0.261 | 0.282 | 0.227 | 0.225 | 0.327 | 0.198 |

## 5 | EXPERIMENTAL RESULTS

In this section, we present the settings of our experiment and details for reproducing the results. The main principle of our experimental setup is to fairly compare the original NeRF[5] with the existing open-source usable 4K-NeRF[9] work to exemplify the advantages of the proposed new approach. Our experimental settings will uniformly follow original papers to produce the baselines, unless we specify otherwise.

### 5.1 | Experimental Setting

**Datasets.** LLFF dataset[40] is utilized in our experiments and ablation studies due to its provision of real-world scenes with 4K ultra-high resolution. This dataset consists of eight scenes captured from the forward-facing perspectives, with different amount of training views ranging from 20 to 60. The original resolution of the datasets is 4032 x 3024 pixels. However, existing NeRF-based methods typically use 4x down-scale the images (resulting in 1008 x 756 pixels) for both training and inference. In our experiments, we deviate from this practice and use the original 4K images as ground truth (GT) for training and evaluation in the primary experiments. For the ablation study, we employ corresponding lower-resolution images to assess the framework's impact on visual quality improvement at different resolutions, specifically 2K and 1K. We adopt the camera poses estimated by COLMAP[41], following the same procedure as other methods in the field.

### 5.2 | Evaluation Metrics

Peak signal-to-noise ratio (PSNR) is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed as a logarithmic quantity using the decibel scale. PSNR is commonly used to quantify reconstruction quality for images and videos subject to loss compression. PSNR is the most popular metric of image quality assessment[42]. For simplicity, we take grey-level (8-bit) images as examples. Given a test image $I_a$ and a reference image $I_b$, both of size $W \times H$, the PSNR can be defined as

$$(I_a, I_b) = 10 \log_{10} \left( \frac{255^2}{(I_a, I_b)} \right), \tag{15}$$

where $(I_a, I_b) = \frac{1}{WH} \sum_{i,j,k} (I_{b,ijk} - I_{a,ijk})^2$. It is easy to see that PSNR directly depends on MSE and overlooks the collective information of a group of pixels.

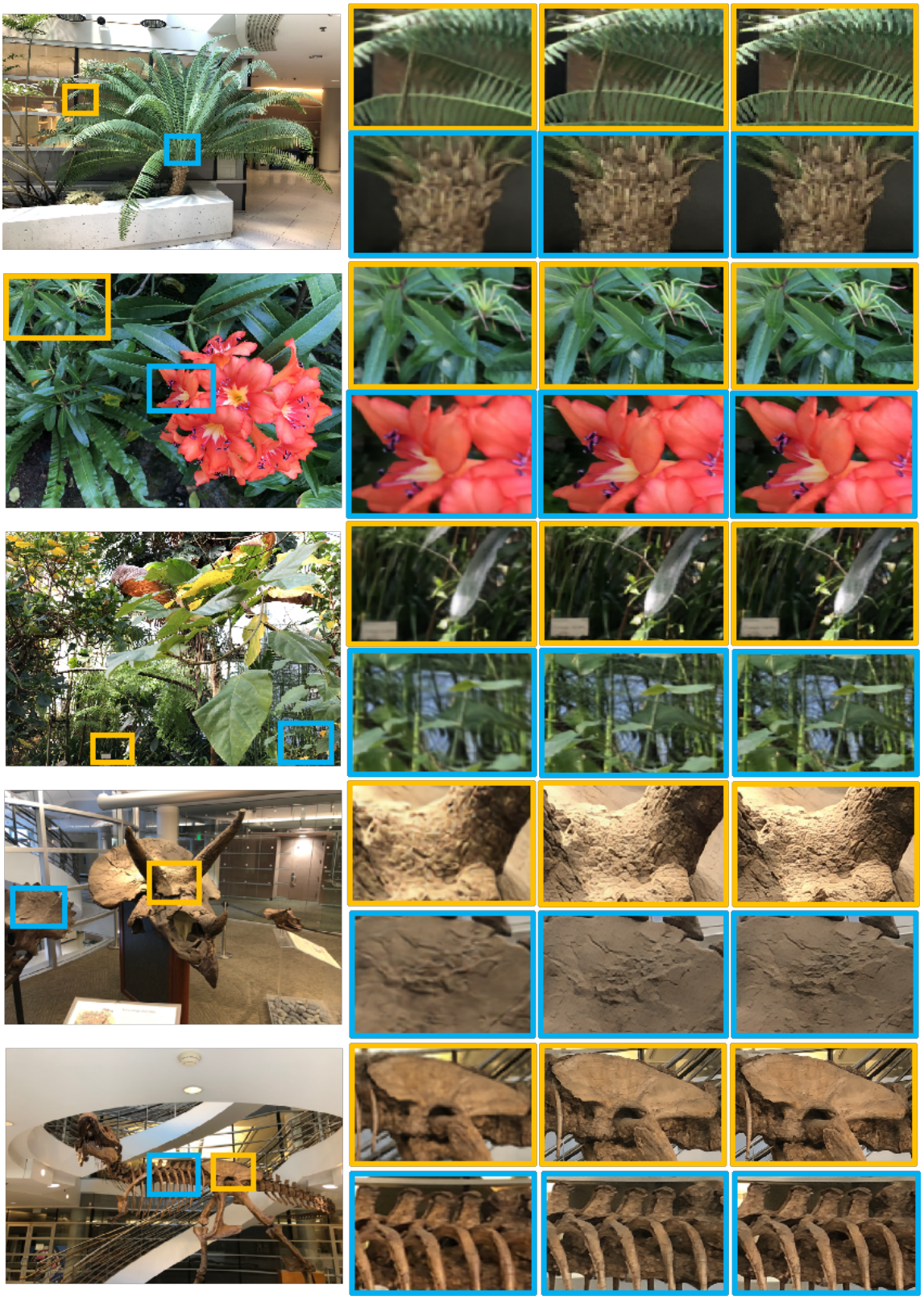


**FIGURE 3** Qualitative comparison of NeRF (a), 4K-NeRF (b), and our method (c).

**TABLE 2** Ablation studies on LLFF dataset. The best result of each measurement is marked in **bold** font.

| Module | fern | flower | fortress | horns | leaves | orchids | room | trex |
|---|---|---|---|---|---|---|---|---|
| base | 34.47 | 36.84 | 38.35 | 34.94 | 37.61 | 35.43 | 40.76 | 34.23 |
| base+vol | 34.69 | 37.02 | 38.52 | 35.16 | 37.75 | 35.51 | 40.99 | 34.39 |
| base+vol+def | 34.84 | 37.11 | 38.68 | 35.24 | 38.85 | 35.79 | 41.14 | 34.42 |

We further evaluate the method with more metrics, including LPIPS (Learned Perceptual Image Patch Similarity) and SSIM (Structure Similarity Index Measure) metrics for assessing perceptual effect, as well as another distortion-oriented metric SSIM. LPIPS is calculated with AlexNet[43].

## 5.3 | Comparisons

Firstly, the four tables displayed in Fig. 1 show the line graphs of the original NeRF performance when rendering images of different resolutions after training with the LLFF dataset at different resolutions of data, respectively. It is obvious that NeRF doesn't improve performance while training 2K-4K resolution images, even after using higher-resolution images. This is why we explore the ultra-high-definition NeRF problem.

We compare our method with various existing competitive methods for ultra resolution: NeRF[1], 4K-NeRF[9] (alignNeRF[10] cannot be tested uniformly due to the unopened source code). About the novel view synthesis tasks for ultra-high resolution competitors, we select two representative methods, NeRF(use UHD data) and 4K-NeRF. For deep learning-based methods, 4K-NeRF performs image super-resolution after NeRF has rendered the novel view and fed it into the next super-resolution network which is more complex than NeRF-SR[38], making the details more accurate. Some common characteristics of works are the insertion of a super-resolution module for resolution enhancement after NeRF reconstruction, while our work uses a similar framework. We will primarily compare performances to 4k-NeRF which has the same structure and input data with us.

For a fair comparison with the baselines, we experimented with two settings for them: 1) with standard configuration expect training on 4K resolution, and 2) using standard configuration with network parameters and voxel grid sizes if used and training on 4K resolution. The statistical results of each method on the whole dataset are reported in Table 1. Our method also performs competitively in terms of evaluation metrics compared to all baselines. The original NeRF method is poor at reconstructing fine details in 4K scenes, resulting in lost or blurred details. Notably, our method can achieve even higher PSNR and most of SSIM, Lpips values compared to 4K-NeRF. Although our method only uses the simplest combination of L2 loss and L1 loss and has fewer computing resources consumed by the neural network, it is also very close to 4K-NeRF in terms of the generated texture and visual quality, which despite having a more human-intuitive generation effect due to GAN network. It is also better at generating details for some cases with fine textures (shown in Fig. 3). This is attributed to the deformable alignment unit module connected to our framework. These results demonstrate that our method performs well on both qualitative and quantitative results.

Due to the voxel grid embedding approach and a feature-level alignment network with shallow layers, we achieve impressive performance in terms of inference efficiency and memory cost, allowing a 4K image to be rendered within 300 ms. Compared to the nearly half-day training time required for normal NeRF and the super-resolution networks of 4K-NeRF, our approach achieves a significant improvement saving more than half of the time.

## 5.4 | Ablation Study

Finally, we also conduct experiments on the dataset LLFF datasets to evaluate the performance of different components of our designed network. To demonstrate the effectiveness of our proposed method in high-resolution novel view synthesis tasks, we progressively add the proposed modules to a baseline network and prove the functionality of the Hybrid Volumetric Representations module and Deformable Alignment Unit module, respectively. To build the baseline network, we remove the later alignment module and only use networks with an explicit voxel grid number of 1 to facilitate the comparison operation afterward.

**Hybrid volumetric representation.** By comparing NeRF without voxel grid embedding, we validate the effectiveness of NeRF after embedding based on Hybrid volumetric representations. As shown in Table 2, our work achieves better PSNR values than the baseline on all 8 cases of LLFF data. This module contributes to that our training inference time can be reduced exponentially. Furthermore, increasing the density of the voxel grid can lead to even higher PSNR values, as a denser voxel grid captures finer image features, as shown in Figure 4.

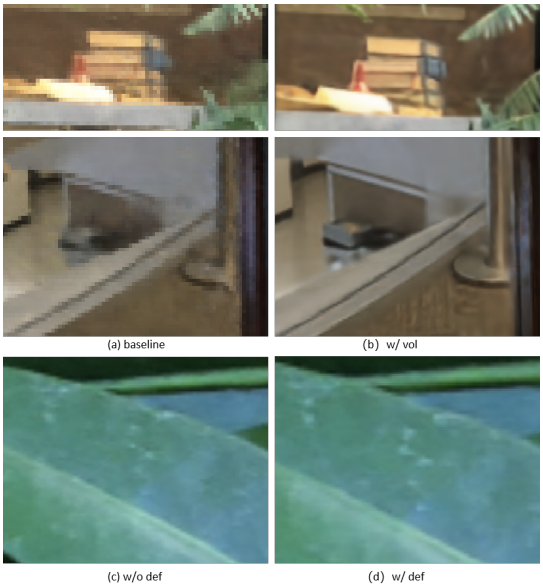


**FIGURE 4** Qualitative comparison of the baseline method (a), baseline with hybrid volumetric representation model (b), without deformable alignment unit (c), and our full model with all components (d).

**Deformable alignment unit.** After adding the deformable alignment unit module, we obtain the feature map from the reference frame image and the current rendering frame image through feature extraction CNN. Then we fed them into the deformable CNN for unaligned feature offset calculation and concate the calculated offset to the current rendering frame for alignment. The results show that the deformable alignment unit can further capture missing details from the reference frame to the rendered frame, both in the PSNR values and the rendered results, as shown in Figure 4. The PSNR value of most cases is improved by about 0.1-0.2, as shown in Table 2.

## 5.5 | Limitations

We successfully apply our proposed method to high-resolution 3D reconstruction and synthesizing novel views tasks. However, there still exist several limitations in our De-NeRF method.

Firstly, we found that our model does not perform very well for samples with overlapping floral details (e.g., comparison with 4K-NeRF shown in rows 1, 2, and 3 in Figure 3). A reasonable guess is that the image detail capturer CNN of our deformable alignment network is simple due to the desire to save training time. If the deformable alignment network continues to deepen or is fine-tuned using pre-training weights from more advanced super-resolution or other large models for 2D image vision tasks may work better.

Secondly, the deformable alignment unit using deformable networks did not incorporate the training loss used to compute the pixel or feature offsets of the image as in other video-resolution work [33,34]. Joining this approach may reduce the probability of overfitting and potentially improve the final rendering effect. However, it may result in a more complex model and increase the training difficulty.

Finally, our deformable alignment network training is slow. Our current voxel grid embedding was implemented based on the PyTorch approach and with native cuda-accelerated code like the original instant-ngp[7]. Theoretically, it still has plenty of room for improvement in terms of generation speed and accuracy.

## 6 | CONCLUSION AND FUTURE WORK

In this paper, we have explored the challenges of 3D reconstruction on high-resolution datasets with NeRF and presented an innovative framework for fast reconstruction with improved details. We first quantitatively and qualitatively analyzed the performance bottleneck problem of NeRF while increasing the resolution of datasets. It motivates us to propose the alignment

operation inspired by a deformable convolution network that shifts the spatial location of the unaligned image feature rendered by the NeRF to the true position. In the case of high-resolution data causing problems with large network parameters and slow training speeds, we use the voxel grid feature embedding strategy, which increases the training speed while generating a more accurate rendered image when using a denser voxel grid. Our experiments on the challenging real-world high-resolution datasets validate the ability of our framework to achieve high-fidelity rendering results.

In the future, we plan to extend this network as a generalized patch behind various NeRF networks which need to improve the accuracy. We would also like to handle dynamic scene 3D reconstruction tasks. Dynamic scenes have richer linear frames just like video super-resolution tasks, and the addition of our deformable alignment unit after the dynamic NeRF reconstruction work is believed to achieve better results for the dynamic scene reconstruction.

## References


1. Mildenhall B, Srinivasan PP, Tancik M, others . Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*. 2021;65(1):99–106.
2. Barron JT, Mildenhall B, Tancik M, Hedman P, Martin-Brualla R, Srinivasan PP. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. *IEEE International Conference on Computer Vision (ICCV)*. 2021:5855–5864.
3. Chen A, Xu Z, Zhao F, others . Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. *IEEE International Conference on Computer Vision (ICCV)*. 2021:14124–14133.
4. Liu L, Gu J, Zaw Lin K, others . Neural sparse voxel fields. *Advances in Neural Information Processing Systems*. 2020;33:15651–15663.
5. Martin-Brualla R, Radwan N, Sajjadi MS, others . Nerf in the wild: Neural radiance fields for unconstrained photo collections. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2021:7210–7219.
6. Sun C, Sun M, Chen HT. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2022:5459–5469.
7. Müller T, Evans A, Schied C, Keller A. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*. 2022;41(4):1–15.
8. Lindenberger P, Sarlin PE, Larsson V, Pollefeys M. Pixel-perfect structure-from-motion with featuremetric refinement. *IEEE International Conference on Computer Vision (ICCV)*. 2021:5987–5997.
9. Wang Z, Li L, Shen Z, others . 4k-nerf: High fidelity neural radiance fields at ultra high resolutions. *arXiv preprint arXiv:2212.04701*. 2022.
10. Jiang Y, Hedman P, Mildenhall B, others . AligNeRF: High-Fidelity Neural Radiance Fields via Alignment-Aware Training. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2023:46–55.
11. Park JJ, Florence P, Straub J, others . Deepsdf: Learning continuous signed distance functions for shape representation. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2019:165–174.
12. Duan Y, Zhu H, Wang H, others . Curriculum deepsdf. *European Conference on Computer Vision (ECCV)*. 2020:51–67.
13. Chen Z, Zhang H. Learning implicit fields for generative shape modeling. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2019:5939–5948.
14. Mescheder L, Oechsle M, Niemeyer M, others . Occupancy networks: Learning 3d reconstruction in function space. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2019:4460–4470.
15. Peng S, Niemeyer M, Mescheder L, others . Convolutional occupancy networks. *European Conference on Computer Vision (ECCV)*. 2020:523–540.
16. Niemeyer M, Mescheder L, Oechsle M, Geiger A. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2020:3504–3515.
17. Kato H, Ushiku Y, Harada T. Neural 3d mesh renderer. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2018:3907–3916.
18. Liu S, Li T, Chen W, Li H. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. *IEEE International Conference on Computer Vision (ICCV)*. 2019:7708–7717.
19. Saito S, Huang Z, Natsume R, others . Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. *IEEE International Conference on Computer Vision (ICCV)*. 2019:2304–2314.
20. Yu A, Ye V, Tancik M, Kanazawa A. pixelnerf: Neural radiance fields from one or few images. *IEEE Computer Vision and Pattern Recognition (CVPR)*. 2021:4578–4587.

21. Park K, Sinha U, Barron JT, others . Nerfies: Deformable neural radiance fields. *IEEE International Conference on Computer Vision (ICCV).* 2021:5865–5874.

22. Pumarola A, Corona E, Pons-Moll G, Moreno-Noguer F. D-nerf: Neural radiance fields for dynamic scenes. *IEEE Computer Vision and Pattern Recognition (CVPR).* 2021:10318–10327.

23. Trevithick A, Yang B. Grf: Learning a general radiance field for 3d representation and rendering. *IEEE International Conference on Computer Vision (ICCV).* 2021:15182–15192.

24. Jang W, Agapito L. Codenerf: Disentangled neural radiance fields for object categories. *IEEE International Conference on Computer Vision (ICCV).* 2021:12949–12958.

25. Ledig C, Theis L, Huszár F, others . Photo-realistic single image super-resolution using a generative adversarial network. *IEEE Computer Vision and Pattern Recognition (CVPR).* 2017:4681–4690.

26. Menon S, Damian A, Hu S, others . Pulse: Self-supervised photo upsampling via latent space exploration of generative models. *IEEE Computer Vision and Pattern Recognition (CVPR).* 2020:2437–2445.

27. Dong C, Loy CC, He K, Tang X. Learning a deep convolutional network for image super-resolution. *European Conference on Computer Vision (ECCV).* 2014:184–199.

28. Schonberger JL, Frahm JM. Structure-from-motion revisited. *IEEE Computer Vision and Pattern Recognition (CVPR).* 2016:4104–4113.

29. Dong C, Loy CC, He K, Tang X. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell..* 2015;38(2):295–307.

30. Goodfellow I, Pouget-Abadie J, Mirza M, others . Generative adversarial nets. *Advances in neural information processing systems.* 2014;27.

31. Sajjadi MS, Scholkopf B, Hirsch M. Enhancenet: Single image super-resolution through automated texture synthesis. *IEEE International Conference on Computer Vision (ICCV).* 2017:4491–4500.

32. Zeng Y, Fu J, Chao H. Learning joint spatial-temporal transformations for video inpainting. *European Conference on Computer Vision (ECCV).* 2020:528–543.

33. Tian Y, Zhang Y, Fu Y, Xu C. Tdan: Temporally-deformable alignment network for video super-resolution. *IEEE Computer Vision and Pattern Recognition (CVPR).* 2020:3360–3369.

34. Wang X, Chan KC, Yu K, others . Edvr: Video restoration with enhanced deformable convolutional networks. *IEEE Computer Vision and Pattern Recognition (CVPR).* 2019:0–0.

35. Dai J, Qi H, Xiong Y, others . Deformable convolutional networks. *IEEE International Conference on Computer Vision (ICCV).* 2017:764–773.

36. Goldlücke B, Aubry M, Kolev K, Cremers D. A super-resolution framework for high-accuracy multiview reconstruction. *Int. Journal of Computer Vision.* 2014;106:172–191.

37. Maier R, Stückler J, Cremers D. Super-resolution keyframe fusion for 3D modeling with high-quality textures. *2015 International Conference on 3D Vision.* 2015:536–544.

38. Wang C, Wu X, Guo YC, others . NeRF-SR: High Quality Neural Radiance Fields using Supersampling. *Proceedings of the 30th ACM International Conference on Multimedia.* 2022:6445–6454.

39. Max N. Optical models for direct volume rendering. *IEEE Trans. on Vis. and Comput. Graph..* 1995;1(2):99–108.

40. Mildenhall B, Srinivasan PP, Ortiz-Cayon R, others . Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Trans. Graph..* 2019;38(4):1–14.

41. Fisher A, Cannizzaro R, Cochrane M, others . ColMap: A memory-efficient occupancy grid mapping framework. *Robotics and Autonomous Systems.* 2021;142:103755.

42. Blau Y, Michaeli T. The perception-distortion tradeoff. *IEEE Computer Vision and Pattern Recognition (CVPR).* 2018:6228–6237.

43. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems.* 2012;25.

## AUTHOR BIOGRAPHY

**Jianing Hou** received a B.S. degree in Mechanical Design, Manufacture and Automation from Dalian Jiaotong University, China. He is currently an M.S. student majoring in Computer Application Technology in the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include computer vision and computer graphics.

**Runjie Zhang** is currently an undergraduate student at the University of California San Diego, pursuing a double major in Data Science and Math-CS. She's interested in leveraging her background in Machine Learning, with a specific focus on Computer Vision, Causual Inference, and Optimization..

**Zhongqi Wu** received her master's degree in the School of Artificial Intelligence of University of Chinese Academy of Sciences in 2019 and received her Ph.D degree at the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Now she is working in Institutes of Science and Development, Chinese Academy of Sciences. Her research interests include image processing and computer vision.

**Weiliang Meng** received his PhD degree in Computer Application from the State Key Laboratory of Computer Science at the Institute of Software, Chinese Academy of Sciences, in 2010. He is currently an Associate Professor in the State Key Laboratory of Multimodal Artificial Intelligence Systems at the Institute of Automation, Chinese Academy of Sciences. His main research fields include artificial intelligence, computer vision, 3D scene analysis, 3D geometry processing, and computer graphics.

**Xiaopeng Zhang** received the B.S. degree and M.S. degree in Mathematics from Northwest University, Xi'an, China, in 1984 and 1987, respectively, and the Ph.D. degree in Computer Science from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 1999. He is currently a Professor with the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His main research interests are computer graphics and computer vision.

**Jianwei Guo** is an Associate Professor at the State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS), Institute of Automation, Chinese Academy of Sciences (CASIA). He received his Ph.D. degree in computer science from CASIA in 2016, and bachelor's degree from Shandong University in 2011. His research interests include computer graphics, geometric processing and 3D vision.