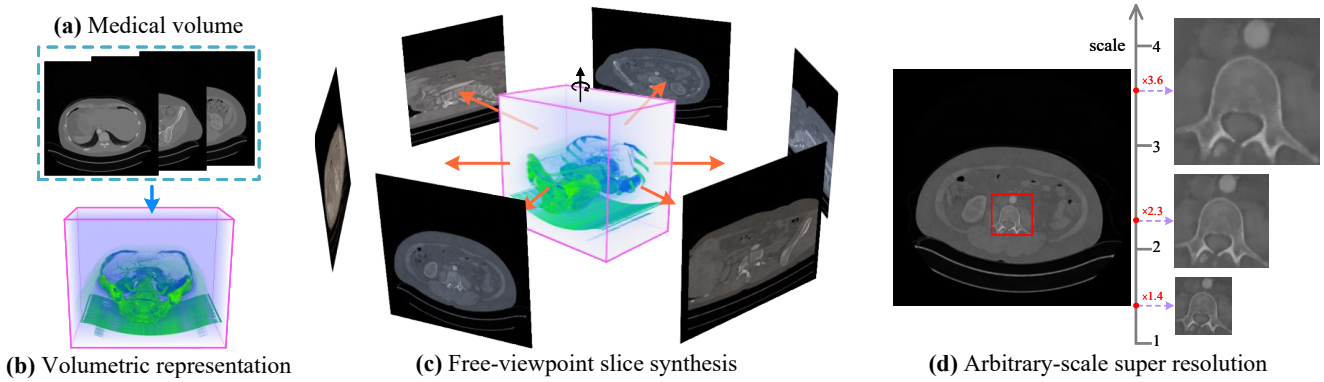
CuNeRF: Cube-Based Neural Radiance Field for Zero-Shot Medical Image Arbitrary-Scale Super Resolution

Zixuan Chen[1] Lingxiao Yang[1,2,3] Jian-Huang Lai[1,2,3] Xiaohua Xie[1,2,3*]
[1]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2]Guangdong Province Key Laboratory of Information Security Technology, Guangzhou, China
[3]Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
chenzx3@mail2.sysu.edu.cn, lingxiao.yang717@gmail.com, {stsljh, xiexiaoh6}@mail.sysu.edu.cn



(b) Volumetric representation **(c)** Free-viewpoint slice synthesis **(d)** Arbitrary-scale super resolution

Figure 1. *CuNeRF* is the first zero-shot Medical Image Arbitrary-Scale Super Resolution framework. After training on an LR medical volume **(a)** itself, *CuNeRF* can build the corresponding continuous volumetric representation **(b)**, which is able to achieve **(c)** Free-viewpoint slice synthesis: yielding novel-view medical slices from the arbitrary viewpoints, and **(d)** Arbitrary-scale super-resolution: upsampling medical images at arbitrary scales in a continuous domain. Project page is available at NarcissusEx.github.io/CuNeRF.

Abstract

*Medical image arbitrary-scale super-resolution (MIASSR) has recently gained widespread attention, aiming to supersample medical volumes at arbitrary scales via a single model. However, existing MIASSR methods face two major limitations: **(i)** reliance on high-resolution (HR) volumes and **(ii)** limited generalization ability, which restricts their applications in various scenarios. To overcome these limitations, we propose Cube-based Neural Radiance Field (CuNeRF), a zero-shot MIASSR framework that is able to yield medical images at arbitrary scales and free viewpoints in a continuous domain. Unlike existing MISR methods that only fit the mapping between low-resolution (LR) and HR volumes, CuNeRF focuses on building a continuous volumetric representation from each LR volume without the knowledge of the corresponding HR one. This is achieved by the proposed differentiable modules: cube-based sampling, isotropic volume rendering, and cube-based hierarchical rendering. Through extensive experiments on magnetic resource imaging (MRI) and computed tomography (CT) modalities, we demonstrate that CuNeRF can synthesize high-quality SR medical images, which outperforms state-of-the-art MISR methods, achieving better visual verisimilitude and fewer objectionable artifacts. Compared to existing MISR methods, our CuNeRF is more applicable in practice.*

1. Introduction

Medical imaging techniques such as computed tomography (CT) and magnetic resonance imaging (MRI) are critical tools in assisting clinical diagnosis. However, the acquisition of high-quality medical slices is a resource-intensive process, which requires subjects to be exposed to considerable ionizing radiations for a long time, increasing the lifetime risk of cancer [21]. To reduce the burden on subjects, a feasible approach is to reconstruct high-resolution (HR) medical volumes from low-resolution (LR) ones.

To tackle medical image super-resolution (MISR) challenges, early studies employed optimization methods [12,

*Corresponding author.

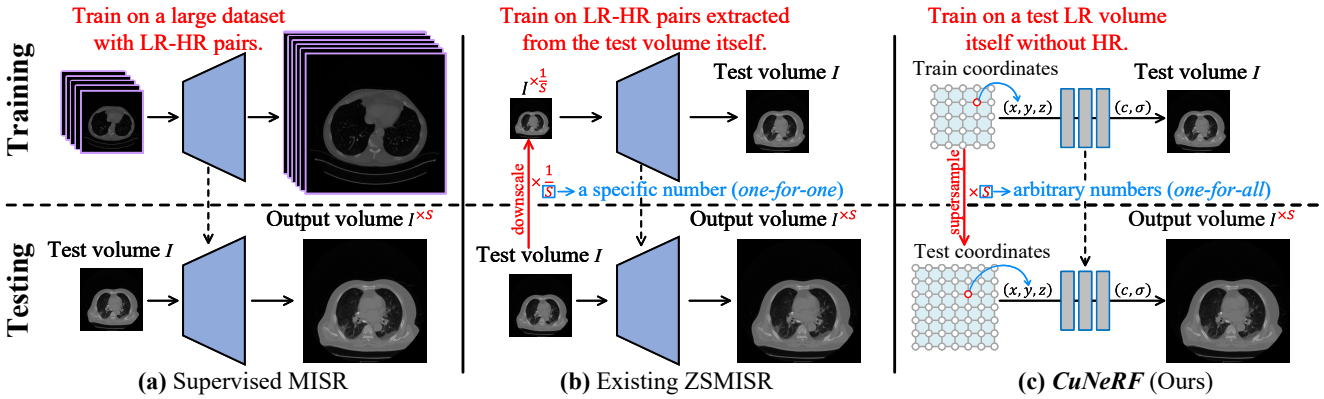


Figure 2. Difference between existing supervised MISR **(a)**, zero-shot MISR (ZSMISR) **(b)** and *CuNeRF* **(c)**. Visually, supervised MISR methods need to collect considerable LR-HR pairs for training, while ZSMISR and our *CuNeRF* only train the model on each test volume itself. However, given a test volume, ZSMISR methods can only upsample medical images at a specific scale (one-for-one), while our *CuNeRF* can handle arbitrary upsampling scales (one-for-all).

40] and interpolation methods [18]. Subsequently, a series of methods [8, 43, 47, 5, 36] have adopted convolutional neural networks to learn the LR-HR mappings. Recently, medical image arbitrary-scale super-resolution (MIASSR) methods [28, 41] have received widespread attention in the MISR community, aiming to employ a single model to upsample medical volumes at arbitrary scales. Although these methods achieve acceptable HR results, they still have two major issues: *(i)* Existing MIASSR methods rely on the supervision from HR volumes, yet high-quality HR volumes are not always available; *(ii)* These methods may be susceptible to the distribution gap between training and test data, producing non-existent details. These drawbacks limit the application scenarios of existing MIASSR methods.

To address the above-mentioned limitations, we present a zero-shot MIASSR framework – Cube-based NeRF (CuNeRF), which aims to yield arbitrary upsampling images after training on a test LR volume itself (see Figure 2). Specifically, we draw inspiration from the neural radiance field (NeRF) [23] to estimate the continuous volumetric representation from discrete samples (LR volumes) instead of fitting the mapping between LR and HR volumes. Since directly applying NeRF on medical volumes may result in grid-like artifacts (see Figure 3, detailed explanation is provided in Section 4.1), our *CuNeRF* tackles such aliasing issues via the proposed differentiable modules: cube-based sampling, isotropic volume rendering, and cube-based hierarchical rendering. As shown in Figure 1, *CuNeRF* can build a continuous mapping between the coordinate and the corresponding intensity value in the training data, which is capable of generating medical slices at arbitrary scales and free viewpoints in a continuous domain. Comprehensive experiments on the MSD Brain Tumour (MRI) [33] and KiTS19 (CT) [13] datasets show that *CuNeRF* yields impressive performance in 3D and volumetric MISR at various upsampling scales, outperforming state-of-the-art methods.

The main contributions are summarized:

- To the best of our knowledge, *CuNeRF* is the first zero-shot MIASSR framework that can continuously upsample medical volumes at arbitrary scales.
- We address the hole-forming issues via the proposed techniques: cube-based sampling, isotropic volume rendering, and cube-based hierarchical rendering.
- Extensive experiments on CT and MRI modalities for 3D MISR and volumetric MISR show *CuNeRF* favorably surpasses state-of-the-art MIASSR methods.

2. Related Works

In this section, we first review implicit neural representation and then introduce some impressive progress in medical image super-resolution. Recent surveys [39, 19] provide a comprehensive review of super-resolution methods.

2.1. Implicit Neural Representation

Learning implicit neural representations (INRs) from discrete samples to form a continuous function has been a long-standing research problem in computer vision for numerous tasks. A recent trend in this field is to map discrete representations to coordinate-based continuous neural representations through implicit functions formed by neural networks, such as multi-layer perceptron (MLP). Chen *et al.* [4] proposed a method to learn the INR of 2D images using the local implicit image function. Subsequent work [6] extended this [4] to apply in the video domain. Currently, most 3D view-synthesis methods are based on the neural radiance fields (NeRF) [23] framework. NeRF can model a volumetric radiance field to render novel views with impressive visual quality using standard volumetric rendering [15]

and alpha compositing techniques [29]. However, NeRF has the drawback of requiring massive training views and lengthy optimization iterations to learn the correct 3D geometry. Several follow-up works have attempted to optimize NeRF's training procedures, such as reducing the required training views [44, 10, 35], accelerating convergence and rendering speed [25]. Other works aim to adapt NeRF to various domains, such as generative modeling [31, 26], anti-aliasing [2], unbounded representation [3], and RGB-D scene synthesis [1]. Recently, some researchers employ INR-based methods to reconstruct medical images from discrete-sampled data [34, 46, 7, 9], more details can be seen in the recent survey [24].

2.2. Medical Image Super Resolution

Medical image super-resolution (MISR) is an important task in medical image processing, which aims to reconstruct high-resolution (HR) medical slices from corresponding low-resolution (LR) ones. Initially, some conventional methods like [12, 40] and widely-used interpolation methods like bicubic and tricubic interpolations [18] were employed in the early research. Inspired by [11], recent studies have shifted their focus towards using deep learning-based super-resolution networks in the medical domain. Lim *et al.* [20] employ deep learning-based super-resolution networks to upsample medical images. Some studies upsample each 2D LR medical slice to acquire the corresponding HR one, such as [8, 43, 47]. On the other hand, Chen *et al.* [5] and Wang *et al.* [36] use 3D DenseNet-based networks to generate HR volumetric patches from LR ones. Yu *et al.* [45] build a transformer-based MISR network to address volumetric MISR challenges. Recent studies have been focusing on medical image arbitrary-scale super-resolution (MIASSR) [41, 28], which aims to upsample medical slices at arbitrary scales by a single model. Inspired by Meta-SR [14], Peng *et al.* [28] deals with volumetric MISR on the z-axis at integer scales. Wu *et al.* [41] propose ArSSR, an INR-based method that can upsample MRI volumes at arbitrary scales in a continuous domain. Wang *et al.* [37] propose a weakly-supervised framework that uses unpaired LR-HR medical volumes for optimization. However, these methods deeply rely on the HR medical volumes, which limits the application scenarios.

3. Preliminary: NeRF

Neural radiance field (NeRF) [23] aims to build the continuous mapping from $(\mathbf{x}, \mathbf{d})$ to $(\mathbf{c}, \sigma)$, where $\mathbf{x} = (x, y, z)$ and $\mathbf{d} = (\theta, \phi)$ denote spatial location and viewing direction, while $\mathbf{c}$ and σ represent the content color and volume density, respectively. NeRF's techniques can be summarized as follow:

Ray sampling. NeRF first constructs the ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ that emits from the center of projection $\mathbf{o}$ and passes through the materials along the viewing direction $\mathbf{d}$. Subsequently, NeRF samples N points along the ray from near plane t_n to far plane t_f predefined. For each sampling point $\mathbf{r}(t_k)$, NeRF employs a positional encoding function $\gamma(\cdot)$ to map the location $\mathbf{x}_k$ and view direction $\mathbf{d}$ into higher dimensional space as:

$$\gamma(\rho) = \rho \bigcup_{i=0}^{L-1} (\sin(2^i\rho), \cos(2^i\rho)),\ where\ L \in \mathbb{N}. \quad (1)$$

where ρ denotes an arbitrary vector and L is a hyperparameter set to 10 as default.

Volume rendering. The pixel color $\mathbf{C}(\mathbf{r})$ can be modeled as the integral of the corresponding ray $\mathbf{r}$ based on Beer-Lambert Laws as:

$$\mathbf{C}(\mathbf{r}) = \int_{t_n}^{t_f} \frac{\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt}{\exp(\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds)}, \quad (2)$$

where $\mathbf{c}(\cdot)$ and $\sigma(\cdot)$ denote the color and volume density functions. In practice, NeRF employs a multi-layer perceptron (MLP) F_Θ to estimate these two functions. For each sampling point $\mathbf{r}(t_k)$, MLP F_Θ predicts the corresponding color $\mathbf{c}_k$ and volume density σ_k by:

$$(\mathbf{c}_k, \sigma_k) = F_\Theta(\gamma(\mathbf{x}_k), \gamma(\mathbf{d})). \quad (3)$$

Given the estimated results of the N sampling points from t_n to t_f, we can approximate the volume rendering integral using numerical quadrature as introduced by [22]:

$$\hat{\mathbf{C}}(\mathbf{r}) = \sum_{i=1}^{N} \frac{1 - \exp(-\sigma_i(t_{i+1} - t_i))}{\exp(\sum_{j=1}^{i} \sigma_j(t_{j+1} - t_j))}\mathbf{c}_i, \quad (4)$$

where $\hat{\mathbf{C}}(\mathbf{r})$ is the predicted color of the pixel.

Hierarchical volume rendering. NeRF also refine the result by allocating samples proportionally to their expected volume distribution based on the coarse estimations. NeRF simultaneously optimizes two MLPs, *i.e.*, the coarse one F_Θ^c and the fine one F_Θ^f. Specifically, NeRF first samples N_c points and obtain the coarse output $\hat{\mathbf{C}}_c(\mathbf{r})$ by Eq 4, which can be rewrited as $\hat{\mathbf{C}}_c(\mathbf{r}) = \sum_{i=1}^{N_c} w_i\mathbf{c}_i$. A piecewise-constant PDF related to the sampling points along the ray can be produced by $\hat{w} = w_i / \sum_{j=1}^{N_c} w_j$. NeRF then samples N_f points from this distribution by inverse transform sampling (ITS) and computes the fine outputs $\hat{\mathbf{C}}_f(r)$ using all $N_c + N_f$ sorted sampling points. Let $\mathcal{R}$ represent the batch, and these two MLPs can be optimized by the following rendering loss:

$$\mathcal{L} = \sum_{\mathbf{r}\in\mathcal{R}} \left[\|g.t. - \hat{\mathbf{C}}_c(\mathbf{r})\|_2^2 + \|g.t. - \hat{\mathbf{C}}_f(\mathbf{r})\|_2^2 \right], \quad (5)$$

where $g.t.$ denotes the ground truth of the rendering pixels.

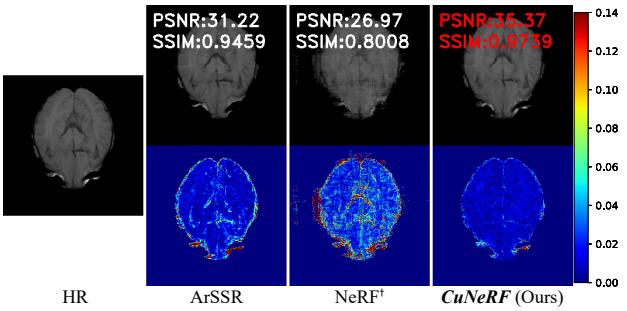


Figure 3. Visual examples of 3D MISR at $\times 2.5$ factor between ArSSR [41], NeRF† [23] and our *CuNeRF* on MSD [33] dataset. Heatmaps at the bottom visualize the difference between the results and the HR image. Visually, NeRF† yields grid-like artifacts, and ArSSR produces non-existent details. By contrast, our *CuNeRF* achieves better visual verisimilitude and fewer artifacts.

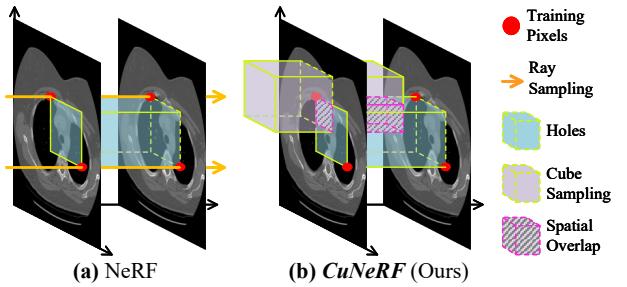


Figure 4. Visualization of the sampling strategies between NeRF [23] **(a)** and *CuNeRF* **(b)** applied on medical volumes. Visually, NeRF *only* samples the rays corresponding to each training pixel, which cannot cover the whole representation fields, leaving some "holes" (*i.e.*, unmodeled space within between adjacent training pixels. To address this issue, *CuNeRF* samples cubes centered by each training pixel, and therefore the "holes" are well-covered by the spatial overlaps.

4. Method

In this section, we first analyze the limitations of NeRF for rendering medical volumes and elaborate on our motivations. Subsequently, based on our findings, we propose cube-based NeRF (CuNeRF), a novel yet efficient method to deal with "zero-shot" medical image arbitrary-scale super-resolution (MIASSR), which extends NeRF's application scenarios in the medical domain. Specifically, we first normalize the medical volumes into the range of $[-1, 1]$ by volumetric normalization, and then train the model via proposed differentiable modules: cube-based sampling, isotropic volume rendering, and cube-based hierarchical rendering. During training, *CuNeRF* is building a coordinate-intensity continuous function whose input is a 3D location $\mathbf{x}=(x, y, z)$ and the output is the corresponding pixel value c. After optimization, *CuNeRF* can predict pixels at any spatial position within the range. As a result, *CuNeRF* is capable to render medical slices at free viewpoints and arbitrary scales by feeding the corresponding plane equations. Figure 5 depicts the overall framework of our *CuNeRF*, and the subsequent techniques are described in the following subsections.

4.1. Analysis & Motivation

As shown in Figure 3, NeRF's sampling strategy may not be suitable for directly applying to medical volumes, which may produce grid-like artifacts in the results. To explain this limitation, we provide an example of NeRF's modeling strategies applied to medical volumes in Figure 4 **(a)**. Visually, NeRF is trained to model the volumetric space along the ray emitted by each training pixel. Since medical volumes only contain three orthogonal slices, which differs from multi-view photos collected by conventional cameras, and thus NeRF's modeling techniques cannot cover the entire representation fields, leaving some "holes" (*i.e.,* unmodeled space) within the regions between adjacent training pixels. Consequently, NeRF may produce sub-optimal results while rendering the contents within the holes. As shown in Figure 3, NeRF†[1] produces grid-like artifacts in upsampling medical volumes, which demonstrates NeRF may struggle to render high-quality HR medical volumes from the corresponding LR ones.

To address the hole-forming issues caused by NeRF's ray sampling, we introduce cube-based sampling, which samples cubes (3D volumetric space) instead of rays (1D space) to fill the hole regions between adjacent training pixels by the spatial overlaps, as demonstrated in Figure 4 **(b)**. To adapt cube-based sampling, we further propose isotropic volume rendering and cube-based hierarchical rendering modules. These modules will be introduced in the following subsections.

4.2. Volumetric Normalization

To build the continuous volumetric representations for the given medical volumes, we first normalize the whole volumetric space $H \times W \times L$ into an ℓ_∞ open ball as:

$$\mathcal{B}(\hat{\mathbf{x}}_\mathbf{o}, 1) = \{\hat{\mathbf{x}} : \|\hat{\mathbf{x}} - \hat{\mathbf{x}}_\mathbf{o}\|_\infty < 1\}, \tag{6}$$

where $\hat{\mathbf{x}}_\mathbf{o}$ is set to $(0, 0, 0)$ as the center $\mathbf{x}_\mathbf{o} = (\frac{H}{2}, \frac{W}{2}, \frac{L}{2})$ of the medical volume. To adapt the positional encoding $\gamma(\cdot)$ introduced in Eq 1, each positional coordinate $\mathbf{x}_t = (x_t, y_t, z_t)$ within the medical volume is transformed into the field coordinate $\hat{\mathbf{x}}_t = (\hat{x}_t, \hat{y}_t, \hat{z}_t)$. The normalization function $\mathcal{N}(\cdot)$ is formulated as:

$$\hat{\mathbf{x}}_t = \left(\frac{2(x_t - \frac{H}{2})}{H + 2P}, \frac{2(y_t - \frac{W}{2})}{W + 2P}, \frac{2(z_t - \frac{L}{2})}{L + 2P} \right), \tag{7}$$

where P is a hyperparameter as the padding size.

[1] NeRF† is trained on three-orthogonal views.

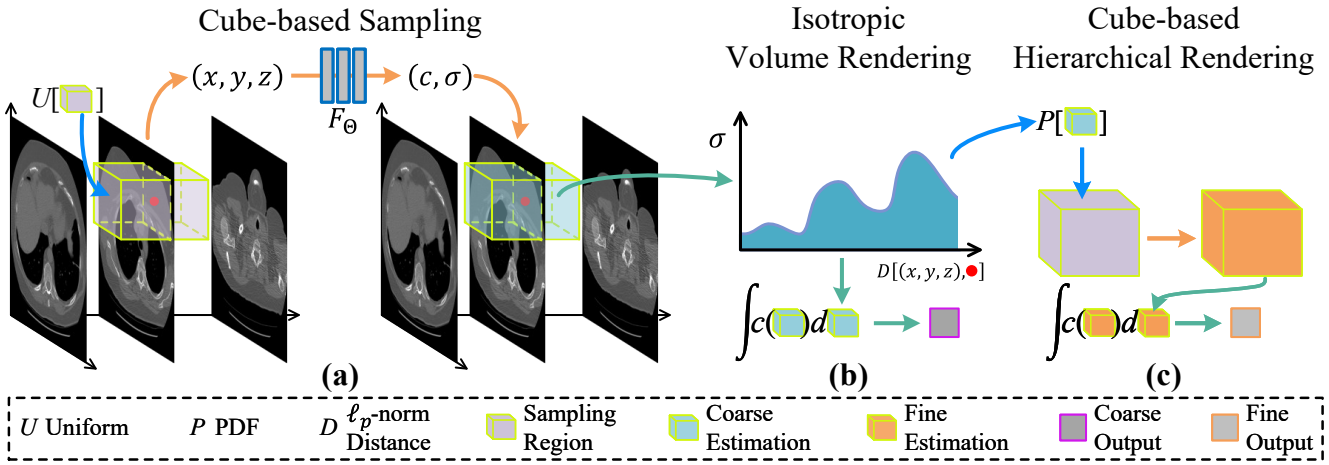


Figure 5. The overall framework of our *CuNeRF*. To synthesize a pixel (red circle) with the spatial position $\mathbf{x}_t$, **(a)** *CuNeRF* first uniformly samples N points as a point set $\{\hat{\mathbf{x}}_i\}_{i=1}^N$ within the cube space (purple cube) centered by $\mathbf{x}_t$. Then, *CuNeRF* obtains the coarse estimation (blue cube) by feeding the sampling points into an MLP F_Θ to produce the set of corresponding pixel intensity $\{c_i\}_{i=1}^N$ and volume density $\{\sigma_i\}_{i=1}^N$. **(b)** Subsequently, assuming σ of each sampling point is only related to the distance with the cube center $\mathbf{x}_t$, *CuNeRF* computes the coarse output of the target pixel via volume integral. **(c)** Finally, *CuNeRF* resamples the points under the probability density function (PDF) of coarse estimation to acquire the fine estimation (orange cube) of the cube. The fine output is generated by the same procedures as **(b)**. Since these two rendering functions are differentiable, *CuNeRF* can be optimized by minimizing the rendering loss in Eq 13. The fine output is the final rendering result of the target spatial position $\mathbf{x}_t$.

4.3. Cube-based Sampling

Implicit neural representation methods aim to build the continuous representation of medical volumes. However, NeRF suffers from hole-forming issues, which may leave some unmodeled spaces in their representation fields, and thus synthesizes grid-like artifacts in upsampled results. To circumvent the holes forming in the representation fields, we propose a novel sampling strategy: cube-based sampling, which samples cubes (3D volumetric space) instead of rays (1D space). Specifically, for the spatial position $\hat{\mathbf{x}}_t$, *CuNeRF* samples a set of points within the cube space $\mathcal{B}(\hat{\mathbf{x}}_t, \frac{l}{2})$. Each point $\hat{\mathbf{x}}_i$ is chosen under the uniform distribution $\mathcal{U}$ by:

$$\hat{\mathbf{x}}_i \sim \mathcal{U}\left[\mathcal{B}(\hat{\mathbf{x}}_t, \frac{l}{2})\right], \tag{8}$$

where l denotes the edge length of the cube. We employ the group of these N sampling points to approximate the cube space. Due to the spatial overlaps between adjacent cubes, the representation fields can be well-covered by employing the proposed cube-based sampling in optimization. As a result, the representation fields can be densely modeled with the same sampling time as NeRF [23].

4.4. Isotropic Volume Rendering

As introduced in 3, the pixel color related to the ray $\mathbf{r}$ is computed by an integral in Eq 2. Intuitively, the pixel color $\mathbf{C}(\hat{\mathbf{x}}_t, l)$ related to the cube space $\mathcal{B}(\hat{\mathbf{x}}_t, \frac{l}{2})$ can be computed by the following triple integral as:

$$\mathbf{C}(\hat{\mathbf{x}}_t, l) = \iiint_{\mathcal{B}(\hat{\mathbf{x}}_t, \frac{l}{2})} \frac{\sigma(\hat{x}, \hat{y}, \hat{z})\mathbf{c}(\hat{x}, \hat{y}, \hat{z})d\hat{x}d\hat{y}d\hat{z}}{\exp(\int_{\hat{x}_n}^{\hat{x}}\int_{\hat{y}_n}^{\hat{y}}\int_{\hat{z}_n}^{\hat{z}}\sigma(x, y, z)dxdydz)}, \tag{9}$$

where $(\hat{x}_n, \hat{y}_n, \hat{z}_n) = (\hat{x}_t - \frac{l}{2}, \hat{y}_t - \frac{l}{2}, \hat{z}_t - \frac{l}{2})$ denotes the initial location of the triple integral while $\mathbf{c}(\cdot)$ and $\sigma(\cdot)$ represent the color and volume density functions. However, since NeRF samples N points to approximate the volume rendering integral of the ray using numerical quadrature in Eq 4, it is required to sample N^3 points to model the cube with the same density, leading to massive computational costs.

Inspired by CRF [17] that assigns the nearby pixels with similar potentials, we assume the volume density σ of each point $\hat{\mathbf{x}}$ within the cube $\mathcal{B}(\hat{\mathbf{x}}_t, \frac{l}{2})$ is only related to the ℓ_p distance $r = \|\hat{\mathbf{x}} - \hat{\mathbf{x}}_t\|_p$ between the centroid and itself. Hence, the volumetric distribution of the cube is isotropic towards the value of r. The above triple integral can be converted into the spherical coordinate system by:

$$\mathbf{C}(\hat{\mathbf{x}}_t, l) = 4\pi \int_0^{\hat{r}} \frac{r^2\sigma(\hat{\mathbf{x}}_t, r)c(\hat{\mathbf{x}}_t, r)dr}{\exp(4\pi\int_0^r s^2\sigma(\hat{\mathbf{x}}_t, s)ds)}, \tag{10}$$

where $\hat{r} = \|(\frac{l}{2}, \frac{l}{2}, \frac{l}{2})\|_p$ denotes the max distance of r within the cube. The derivation detail of Eq 10 is shown in the supplementary materials. Given N sampling points by the proposed cube-based sampling, *CuNeRF* first sorts these points by the distance r. Subsequently, the integral of

the cube is approximated via numerical quadrature rules:

$$\hat{\mathbf{C}}(\hat{\mathbf{x}}_t, l) = 4\pi \sum_{i=1}^{N} \frac{r_i^2 (1 - \exp(-\sigma_i (r_{i+1} - r_i)))}{\exp(4\pi \sum_{j=1}^{i} r_i^2 \sigma_j (r_{j+1} - r_j))} \mathbf{c}_i, \quad (11)$$

where $\hat{\mathbf{C}}(\hat{\mathbf{x}}_t, l)$ denotes the predicted color of $\hat{\mathbf{x}}_t$.

4.5. Cube-based Hierarchical Rendering

To refine the results, *CuNeRF* allocates sampling points proportionally to their expected volume distribution within the cube. Similar to NeRF, *CuNeRF* also simultaneously optimizes the coarse and fine MLPs. As obtaining the coarse output $\hat{\mathbf{C}}_c(\hat{\mathbf{x}}_t, l)$, *CuNeRF* first samples N_f numbers of r using ITS. Subsequently, for each r, we use the hierarchical sampling function $\zeta_p(\cdot)$ to select important points:

$$\hat{\mathbf{x}}_f = \zeta_p(r, \varphi, \theta), \quad (12)$$

where φ and θ are the randomly sampled spherical coordinates, and $\zeta_p(\cdot)$ converts the ℓ_p spherical coordinates (r, φ, θ) to the Cartesian coordinates $\hat{\mathbf{x}}$. If $p \neq \infty$, we allow $\hat{\mathbf{x}}_f$ can beyond the cube space $\mathcal{B}(\hat{\mathbf{x}}_t, \frac{l}{2})$. After obtaining fine outputs $\hat{\mathbf{C}}_f(\hat{\mathbf{x}}_t, l)$ at Eq 11 using the sorted union of $N_c + N_f$ sampling points, *CuNeRF* can be optimized in each batch $\mathcal{R}$ by the proposed adaptive rendering loss:

$$\mathcal{L}_A = \sum_{\hat{\mathbf{x}}_t \in \mathcal{R}} \left[\lambda \| g.t. - \hat{\mathbf{C}}_c(\hat{\mathbf{x}}_t, l) \|_2^2 + \| g.t. - \hat{\mathbf{C}}_f(\hat{\mathbf{x}}_t, l) \|_2^2 \right], \quad (13)$$

where $\lambda = \| g.t. - \hat{\mathbf{C}}_f(\hat{\mathbf{x}}_t, l) \|^{\frac{1}{2}}$ is an adaptive regularization term to alleviate the overfitting brought by the "coarse" part.

4.6. Medical Slice Synthesis

After optimization, *CuNeRF* can predict the pixels at any spatial coordinates within the representation fields. Therefore, *CuNeRF* can represent medical slices with free viewpoints and arbitrary scales by feeding the corresponding plane coordinates. Detailed techniques are described in the following, and we show some examples in Section 5.2.

Free-Viewpoint Rendering. To render a medical slice with the given position $\hat{\mathbf{x}}$ and viewpoint $\mathbf{d}$, we first construct a base plane $\mathcal{P}_o$ at $\hat{\mathbf{x}}_o$. Subsequently, we employ the translation matrix $\mathcal{M}_T$ to move slices from $\hat{\mathbf{x}}_o$ to $\hat{\mathbf{x}}$. Finally, since the viewpoint $\mathbf{d}$ can be represented as rotating ϕ degrees around a certain axis $n_\perp$, we can obtain the rotation matrix $\mathcal{M}_R$ via Rodrigues' rotation formula [30]. Thus, the target plane $\mathcal{P}_t$ can be calculated as:

$$\mathcal{P}_t = \mathcal{M}_T \mathcal{M}_R \mathcal{P}_o. \quad (14)$$

The target medical slices can be obtained by feeding the points sampled within $\mathcal{P}_t$ into our *CuNeRF*.

Arbitrary-Scale Rendering. To render a medical slice with the given sampling scale δ, we first follow the above process to obtain the target plane $\mathcal{P}_t$. Then, we calculate the scale matrix $\mathcal{M}_S$ based on δ, and sample the points under the translation $\mathcal{M}_S \mathcal{P}_t$. By feeding the sampling points into our *CuNeRF*, we can obtain the desired medical slices.

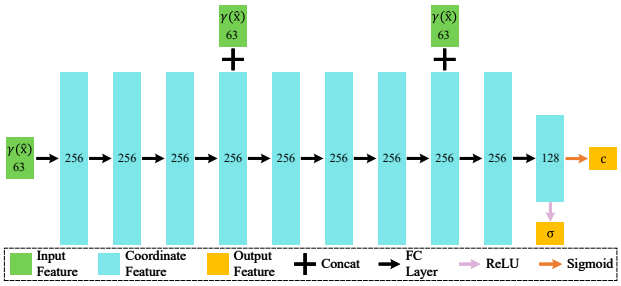


Figure 6. The architecture of MLP. For a given coordinate $\hat{\mathbf{x}}$, it is first encoded by $\gamma(\cdot)$ in Eq 1 as input features. Then, we pass it through 9 fully-connected layers, each having 256 channels with a ReLU. We concatenate the input features to the 4th and 8th hidden layers as the skip connections. Finally, we downscale the feature channels to predict a volume density σ and pixel intensity $\mathbf{c}$.

5. Experiments

In this section, we conduct extensive experiments and in-depth analysis to demonstrate the superiority of our *CuNeRF* in representing high-quality medical images at arbitrary scales. For fair comparisons, the hyperparameters and model settings are consistent in all experiments.

5.1. Experimental Details

Datasets. We comprehensively compare our *CuNeRF* and the existing advances in 2 different modalities: CT and MRI. More specifically, we select T1-weighted MRI volumes from the Medical Segmentation Decathlon (MSD) [33] while we also take CT volumes from the 2019 Kidney Tumor Segmentation Challenge (KiTS19) [13] datasets, respectively. All MRI volumes have the same dimension of $240 \times 240 \times 155$. The image size of each CT slice is 512×512, while the number of CT slices is different.

In experiments, we resize all the CT volumes into 512^3. All the MRI and CT volumes are normalized into $[0, 1]$. The degradation strategy is the nearest-neighbor interpolation. *For the compared supervised methods* [41, 28, 45], we select 50 LR-HR MRI pairs to finetune the pre-trained model of [41], and also select 150 LR-HR CT pairs to train[28, 45]. The evaluation set consists of 80 medical volumes, including 40 CT and 40 MRI volumes. *Note*, following the ZSSR settings reported in [32, 39], we train NeRF† and our *CuNeRF* on each LR test volume itself (see Figure 2), while the HR volumes are only used for evaluations.

Multi-Layer Perceptron Architecture. Figure 6 depicts MLP's architecture, where the input is a 3D location $\hat{\mathbf{x}} = (\hat{x}, \hat{y}, \hat{z})$ encoded by $\gamma(\cdot)$ and the output is a 2D union of

Table 1. 3D MISR comparisons on MSD [33] dataset. **Bold** and underline texts indicate the best and second best performance.

	×2		×2.5		×3		×4		×5		×6		×8	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Conventional methods														
Bicubic	33.75	0.9469	30.84	0.9271	30.74	0.9161	28.67	0.8721	28.14	0.8687	26.83	0.8521	26.54	0.8376
Supervised MIASSR methods														
ArSSR [41]	36.98	0.9690	35.24	0.9398	34.69	0.9199	33.21	0.8910	30.50	0.8624	29.96	0.8543	28.43	0.8353
Zero-Shot MIASSR methods														
NeRF† [23]	29.33	0.8472	27.03	0.8392	25.98	0.8220	25.12	0.8088	24.50	0.7767	23.45	0.7549	22.63	0.7275
CuNeRF	**39.62**	**0.9786**	**37.56**	**0.9441**	**36.24**	**0.9267**	**35.01**	**0.9031**	**34.73**	**0.8952**	**33.69**	**0.8800**	**31.19**	**0.8675**

pixel intensity $\mathbf{c}$ and volume density σ. The parameter size of the proposed MLP is about 0.58M

Implementation Details. *CuNeRF* is implemented on top of [42], a Pytorch [27] re-implementation of NeRF. Our experiments run on a single NVIDIA RTX 3090 GPU with 24G memory. We employ the ℓ_2 distance for isotropic volume rendering and hierarchical cubic rendering while the edge length l of the cube is set to 1. The hierarchical sampling function $\zeta_2(\cdot)$ converts (r, φ, θ) to $\hat{\mathbf{x}} = (\hat{x}, \hat{y}, \hat{z})$ by:

$$\hat{\mathbf{x}} = (r\sin\varphi\cos\theta, r\sin\varphi\sin\theta, r\cos\varphi), \tag{15}$$

where $\varphi \sim \mathcal{U}[0,\pi]$ and $\theta \sim \mathcal{U}[0,2\pi]$, respectively. Let $p = 2$ as default, we consider a spherical parameterized as $(\hat{x}, \hat{y}, \hat{z}) = \zeta_2(r, \varphi, \theta)$, where $\varphi \in [0,\pi], \theta \in [0, 2\pi], r > 0$. This change of variables from the Cartesian system gives us a differential term:

$$d\hat{x}d\hat{y}d\hat{z} = |det(D\zeta_2)|drd\varphi d\theta \tag{16}$$

$$= r^2\sin\varphi drd\varphi d\theta. \tag{17}$$

Therefore, the volume rendering function in Eq 10 can be simplified from Eq 9 as follow:

$$\mathbf{C}(\hat{\mathbf{x}}_t, l) = \int_0^{2\pi}\int_0^{\pi}\int_0^{\frac{\sqrt{3}}{2}l} \frac{\sigma(\hat{\mathbf{x}}_t, r)\mathbf{c}(\hat{\mathbf{x}}_t, r)r^2\sin\varphi d\theta d\varphi dr}{\exp(\underset{\mathcal{B}(\hat{\mathbf{x}}_t, r)}{\iiint}\sigma(\hat{\mathbf{x}}_t, s)s^2\sin\varphi' d\theta' d\varphi' ds)} \tag{18}$$

$$= 4\pi\int_0^{\frac{\sqrt{3}}{2}l} \frac{\sigma(\hat{\mathbf{x}}_t, r)c(\hat{\mathbf{x}}_t, r)r^2 dr}{\exp(4\pi\int_0^r \sigma(\hat{\mathbf{x}}_t, s)s^2 ds)}. \tag{19}$$

For training, we employ Adam [16] as the optimizer with a weight decay of 10^{-6} and a batch size of 2048. The maximum iteration is set to 250000, and the learning rate is annealed logarithmically from 2×10^{-3} to 2×10^{-5}. Similar to NeRF, *CuNeRF* first samples 64 points for the coarse MLP F_Θ^c and feeds 192 points (the sorted union of 64 coarse and 128 fine points) into the fine MLP F_Θ^f. The training time for each $512\times512\times512$ volume is about $0.8\sim3$ *hours*.

For testing, the number of sampling points is set to 16 (8 for coarse MLP and 8 for fine MLP), which can reduce considerable computational costs. The results are obtained by feeding all the coordinates of the given plane equations into our model (seeing details in Section 4.6). The inference time for rendering a $256\times256\times256$ volume is about 30 *secs*. *Note* that we do not use any pre- and post-processing techniques to improve our results in the experiments.

Table 2. Quantitative comparisons of start-of-the-art methods on KiTS19 [13] dataset for volumetric MISR. **Bold** and underline texts indicate the best and second best performance.

	×2		×4		×8	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Conventional methods						
Bicubic	37.75	0.9498	33.76	0.9149	29.03	0.8572
Supervised MISR methods						
TVSRN [45]	39.32	**0.9790**	36.62	0.9532	32.10	0.9163
Supervised MIASSR methods						
SAINT [28]	**39.47**	0.9782	36.61	**0.9574**	31.78	0.9188
Zero-Shot MIASSR methods						
NeRF† [23]	36.50	0.9383	34.14	0.9181	30.56	0.8748
CuNeRF	38.33	0.9663	**36.64**	0.9480	**32.44**	**0.9216**

Evaluation Metrics. We use two quantitative metrics: Peak Signal-to-Noise Ratio (PSNR) and Structured Similarity Index (SSIM) [38] to measure the image quality of different methods. Note we report the average SSIM on axial, coronal, and sagittal planes for volumetric MISR.

5.2. Experimental Results

We compare the proposed *CuNeRF* with 5 state-of-the-art methods, including 2 supervised MIASSR methods: ArSSR [41] and SAINT [28], 1 supervised MISR method: TVSRN [45], 1 conventional method: bicubic interpolation, and NeRF† [23]. Given a upsampling scale δ, we evaluate these methods under the following two settings: *(i)* **3D MISR.** Upsampling the donwsampled volume from $\frac{H}{\delta}\times\frac{W}{\delta}\times\frac{L}{\delta}$ to $H\times W\times L$; *(ii)* **volumetric MISR.** Upsampling the donwsampled volume from $H\times W\times\frac{L}{\delta}$ to $H\times W\times L$. *Note* that NeRF† and *CuNeRF* are trained with the same settings and similar parameter size ($\pm$0.02M).

Quantitative Comparison. We report 3D MISR and volumetric MISR based on the increasing upsampled scales in Table 1 and Table 2, respectively. As demonstrated, for the 3D MISR challenge on MRI volumes, *CuNeRF* surpasses all the competitors with a consistent preferable performance at various upsampling scales. For volumetric MISR challenge on CT volumes, *CuNeRF* achieves compa-

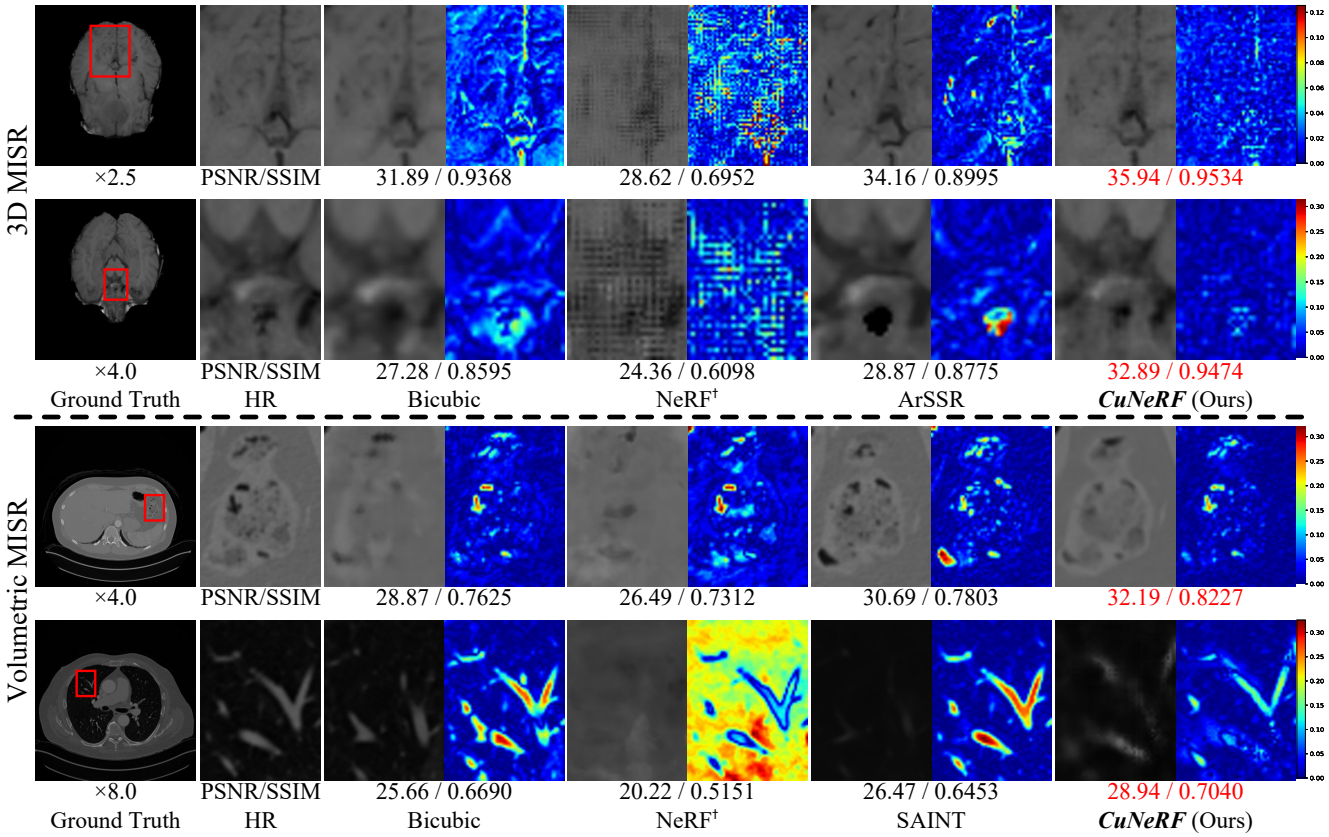


Figure 7. Visual comparisons between our *CuNeRF* and 4 state-of-the-art methods: Bicubic, NeRF† [23], ArSSR [41] and SAINT [28] for 3D MISR and volumetric MISR. The heatmaps on the right of the results visualize the difference related to HR patches.

rable performance to SAINT [28] and TVSRN [45]. Compared to fully-supervised MIASSR methods: ArSSR [41] and SAINT [28], our *CuNeRF* is more robust at presenting large-scale medical slices and capable to deal with different modalities (CT and MRI), suggesting *CuNeRF* owns broader application scenarios. It is also worth noting that NeRF† achieves comparable performance for volumetric MISR but fails in 3D MISR. Since volumetric MISR only aims to acquire the pixels along the z-axis, the experimental results of NeRF† confirm our motivations.

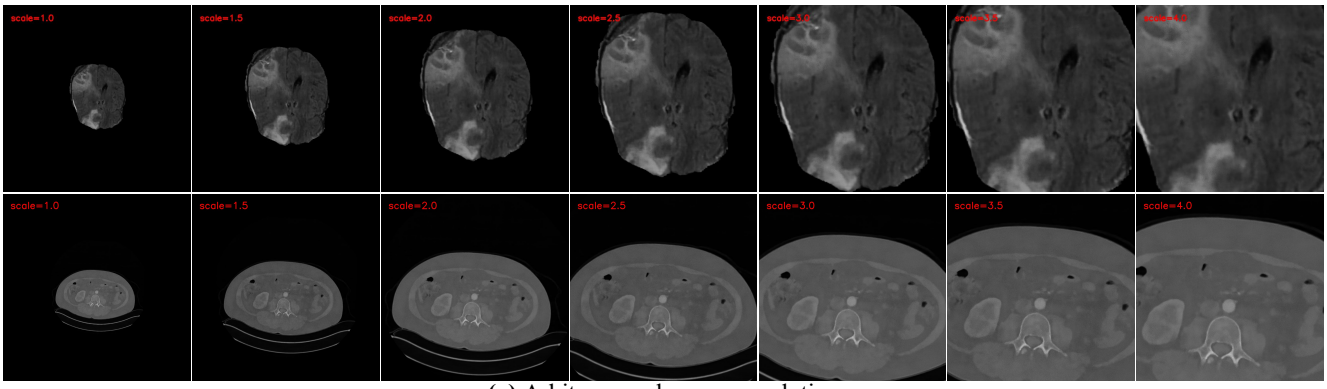
Visual Comparison. We visualize the rendering results of *CuNeRF* and other competitors on MRI (rows 1 and 2) and CT (rows 3 and 4) modalities in Figure 7. It can be observed that *CuNeRF* well represents the medical slices at various scales. Compared to the exhibited methods, *CuNeRF* is most similar to the ground truths, achieving better visual verisimilitude and reducing aliasing artifacts, especially in representing large-scale medical slices. Since NeRF† exhibits grid-like artifacts in rendering high-quality medical slices at larger-valued scales, the visualization results prove the effectiveness of *CuNeRF*, which extends NeRF's capability to continuously represent medical images.

Free-Viewpoint & Arbitrary-Scale Rendering. As shown in Figure 8, *CuNeRF* can synthesize medical images at continuous-valued scales **(a)**. Moreover, *CuNeRF* is capable to yield medical slices with a viewpoint rotating 360 degrees around an arbitrary coordinate axis $n_{\perp}$. Compared to existing methods, *CuNeRF* is capable to provide richer visual information for clinical diagnosis.

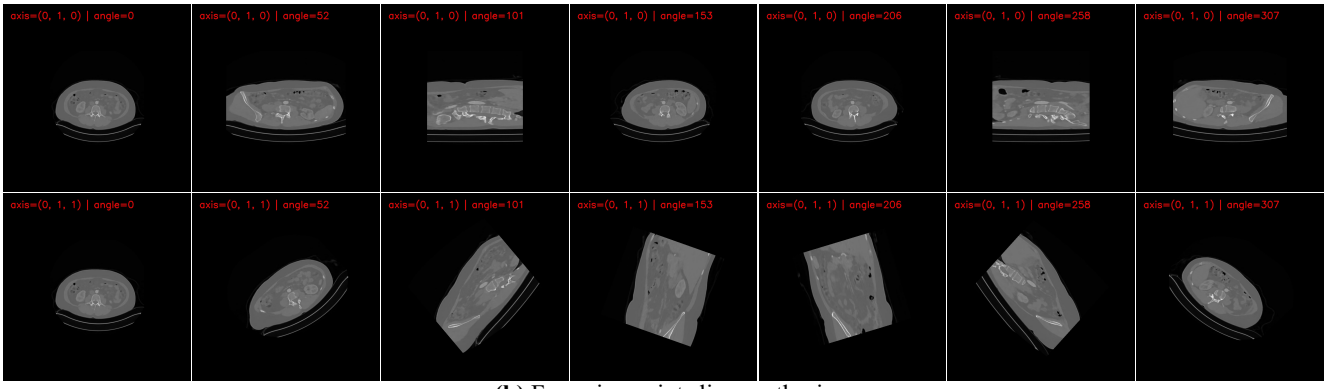
5.3. Ablation Study

In this subsection, we conduct comprehensive experiments to prove the correctness of *CuNeRF*'s design. We first carry out ablation studies to investigate the effectiveness of the proposed modules. Subsequently, we evaluate the *CuNeRF*'s performance under different settings.

***CuNeRF*'s ablation variants.** We evaluate against several ablations of the proposed *CuNeRF* with each module: *CuS*, *IVR* and $\mathcal{L}_A$ represent cube-based sampling, isotropic volume rendering, and adaptive rendering loss, respectively. The baseline model here is NeRF†. As reported in Table 3, the baseline model struggles to deal with 3D MISR issues (row 1), while adopting *CuS* instead of ray sampling can significantly improve the performance (row 2). Compared to NeRF's volume rendering function, employing *IVR* (row 3) can further improve the slice synthesis quality, suggest-



(a) Arbitrary-scale super resolution



(b) Free-viewpoint slice synthesis

Figure 8. Visualization results at arbitrary scales **(a)** and free viewpoints **(b)** within a 1024×1024 range.

Table 3. Comparisons of ablation variants on MSD [33] dataset for 3D MISR. **Bold** text indicates the best performance.

CuS	*IVR*	$\mathcal{L}_A$	×2		×4		×8	
			PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
			29.33	0.8472	25.12	0.8088	22.63	0.7275
✓			35.82	0.9244	33.29	0.8703	27.77	0.8398
✓	✓		38.15	0.9524	34.27	0.8887	29.34	0.8455
✓	✓	✓	**39.62**	**0.9786**	**35.01**	**0.9031**	**31.19**	**0.8675**

ing *IVR* can better estimate the volumetric distribution, reducing aliasing artifacts raised by undersampling. Since the coarse term of NeRF's rendering loss may affect the optimization, $\mathcal{L}_A$ (row 4) is able to alleviate this distraction.

***CuNeRF* under different settings.** We evaluate the performance of *CuNeRF* under different settings: "$p{=}\infty$" employs ℓ_∞ distance of r, "$l{=}0.5$" and "$l{=}2$" represent to set the edge length to 0.5 and 2 pixel distance, respectively. The default is introduced in Section 5.1, where $p{=}2$ and $l{=}1$. As reported in Table 4, the default setting of *CuNeRF* achieves consistent outperformance at various scales. In contrast, employing the ℓ_∞ is significantly inferior to default, which means ℓ_2 distance is more suitable to model the continuous representation for medical volumes. Meanwhile, different cube edge l acquire comparable performance to default, suggesting our *CuNeRF* is a parameter-insensitive method with good robustness under different experimental settings.

Table 4. Quantitative comparisons of *CuNeRF* under different settings on MSD [33] dataset for 3D MISR. **Bold** and underline texts indicate the best and second best performance.

	×2		×4		×8	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
CuNeRF default.	<u>39.62</u>	**0.9786**	**35.01**	**0.9031**	<u>31.19</u>	<u>0.8675</u>
CuNeRF $p=\infty$	35.78	0.9348	32.85	0.8853	29.25	0.8433
CuNeRF $l=0.5$	38.17	0.9621	<u>35.00</u>	0.9018	**31.53**	**0.8704**
CuNeRF $l=2$	**39.65**	<u>0.9723</u>	34.57	<u>0.9011</u>	30.85	0.8608

6. Conclusion

In this paper, we present Cube-based NeRF (CuNeRF), a zero-shot framework for medical image arbitrary-scale super-resolution (MIASSR). Instead of learning the mapping between LR-HR pairs, *CuNeRF* learns the continuous volumetric representation from LR volumes, thus a well-trained model can yield medical images at arbitrary viewpoints and scales in a continuous domain. Extensive experiments demonstrate that *CuNeRF* outperforms state-of-the-art methods, yielding better visual effects and reducing artifacts at various upsampling factors.

Acknowledgement

This project is supported by the Natural Science Foundation of China (No. 62072482).

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6290–6301, 2022. 3

[2] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 3

[3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5470–5479, 2022. 3

[4] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8628–8638, 2021. 2

[5] Yuhua Chen, Feng Shi, Anthony G Christodoulou, Yibin Xie, Zhengwei Zhou, and Debiao Li. Efficient and accurate mri super-resolution using a generative adversarial network and 3d multi-level densely connected network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 91–99. Springer, 2018. 2, 3

[6] Zeyuan Chen, Yinbo Chen, Jingwen Liu, Xingqian Xu, Vidit Goel, Zhangyang Wang, Humphrey Shi, and Xiaolong Wang. Videoinr: Learning video implicit neural representation for continuous space-time super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2047–2057, 2022. 2

[7] Zixuan Chen, Lingxiao Yang, Jianhuang Lai, and Xiaohua Xie. Aprf: Anti-aliasing projection representation field for inverse problem in imaging. *arXiv preprint arXiv:2307.05270*, 2023. 3

[8] Venkateswararao Cherukuri, Tiantong Guo, Steven J Schiff, and Vishal Monga. Deep mr brain image super-resolution using spatio-structural priors. *IEEE Transactions on Image Processing (IEEE TIP)*, 29:1368–1383, 2019. 2, 3

[9] Abril Corona-Figueroa, Jonathan Frawley, Sam Bond-Taylor, Sarath Bethapudi, Hubert P. H. Shum, and Chris G. Willcocks. Mednerf: Medical neural radiance fields for reconstructing 3d-aware ct-projections from a single x-ray, 2022. 3

[10] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12882–12891, 2022. 3

[11] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–199. Springer, 2014. 3

[12] Ali Gholipour, Judy A Estroff, and Simon K Warfield. Robust super-resolution volume reconstruction from slice acquisitions: application to fetal brain mri. *IEEE Transactions on Medical Imaging (IEEE TMI)*, 29(10):1739–1758, 2010. 1, 3

[13] Nicholas Heller, Niranjan Sathianathen, Arveen Kalapara, Edward Walczak, Keenan Moore, Heather Kaluzniak, Joel Rosenberg, Paul Blake, Zachary Rengel, Makinna Oestreich, et al. The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. *arXiv preprint arXiv:1904.00445*, 2019. 2, 6, 7

[14] Xuecai Hu, Haoyuan Mu, Xiangyu Zhang, Zilei Wang, Tieniu Tan, and Jian Sun. Meta-sr: A magnification-arbitrary network for super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1575–1584, 2019. 3

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

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[17] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in Neural Information Processing Systems (NeurIPS)*, 24, 2011. 5

[18] Francois Lekien and J Marsden. Tricubic interpolation in three dimensions. *International Journal for Numerical Methods in Engineering*, 63(3):455–471, 2005. 2, 3

[19] Y Li, Bruno Sixou, and F Peyrin. A review of the deep learning methods for medical images super resolution problems. *Innovation and Research in BioMedical engineering*, 42(2):120–133, 2021. 2

[20] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 136–144, 2017. 3

[21] Diego R Martin and Richard C Semelka. Health effects of ionising radiation from diagnostic ct. *The Lancet*, 367(9524):1712–1714, 2006. 1

[22] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics (IEEE TVCG)*, 1(2):99–108, 1995. 3

[23] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 405–421. Springer, 2020. 2, 3, 4, 5, 7, 8

[24] Amirali Molaei, Amirhossein Aminimehr, Armin Tavakoli, Amirhossein Kazerouni, Bobby Azad, Reza Azad, and Dorit Merhof. Implicit neural representation in medical imaging: A comparative survey. *arXiv preprint arXiv:2307.16142*, 2023. 3

[25] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ACM ToG)*, 41(4):1–15, 2022. 3

[26] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11453–11464, 2021. 3

[27] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *Proceedings of Neural Information Processing Systems (NeurIPS)*, 2017. 7

[28] Cheng Peng, Wei-An Lin, Haofu Liao, Rama Chellappa, and S. Kevin Zhou. Saint: Spatially aware interpolation network for medical slice synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 3, 6, 7, 8

[29] Thomas Porter and Tom Duff. Compositing digital images. In *Proceedings of the Annual Conference on Computer Graphics and Interactive Techniques*, pages 253–259, 1984. 3

[30] Olinde Rodrigues. Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace, et de la variation des coordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire. *J. Math. Pures Appl*, 5(380-400):5, 1840. 6

[31] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 20154–20166. Curran Associates, Inc., 2020. 3

[32] Assaf Shocher, Nadav Cohen, and Michal Irani. "zero-shot" super-resolution using deep internal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3118–3126, 2018. 6

[33] Amber L Simpson, Michela Antonelli, Spyridon Bakas, Michel Bilello, Keyvan Farahani, Bram Van Ginneken, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, et al. A large annotated medical image dataset for the development and evaluation of segmentation algorithms. *arXiv preprint arXiv:1902.09063*, 2019. 2, 4, 6, 7, 9

[34] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:7537–7547, 2020. 3

[35] Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. *arXiv preprint arXiv:2303.16196*, 2023. 3

[36] Jiancong Wang, Yuhua Chen, Yifan Wu, Jianbo Shi, and James Gee. Enhanced generative adversarial network for 3d brain mri super-resolution. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3627–3636, 2020. 2, 3

[37] Jiale Wang, Runze Wang, Rong Tao, and Guoyan Zheng. Uassr: Unsupervised arbitrary scale super-resolution reconstruction of single anisotropic 3d images via disentangled representation learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 453–462. Springer, 2022. 3

[38] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (IEEE TIP)*, 13(4):600–612, 2004. 7

[39] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE TPAMI)*, 43(10):3365–3387, 2020. 2, 6

[40] Stefan Wesarg et al. Combining short-axis and long-axis cardiac mr images by applying a super-resolution reconstruction algorithm. In *Medical Imaging 2010: Image Processing*, volume 7623, page 76230I. International Society for Optics and Photonics, 2010. 1, 3

[41] Qing Wu, Yuwei Li, Yawen Sun, Yan Zhou, Hongjiang Wei, Jingyi Yu, and Yuyao Zhang. An arbitrary scale super-resolution approach for 3d mr images via implicit neural representation. *IEEE Journal of Biomedical and Health Informatics*, 27(2):1004–1015, 2023. 2, 3, 4, 6, 7, 8

[42] Lin Yen-Chen. Nerf-pytorch. https://github.com/yenchenlin/nerf-pytorch/, 2020. 7

[43] Chenyu You, Guang Li, Yi Zhang, Xiaoliu Zhang, Hongming Shan, Mengzhou Li, Shenghong Ju, Zhen Zhao, Zhuiyang Zhang, Wenxiang Cong, et al. Ct super-resolution gan constrained by the identical, residual, and cycle learning ensemble (gan-circle). *IEEE Transactions on Medical Imaging (IEEE TMI)*, 39(1):188–203, 2019. 2, 3

[44] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4578–4587, 2021. 3

[45] Pengxin Yu, Haoyue Zhang, Han Kang, Wen Tang, Corey W Arnold, and Rongguo Zhang. Rplhr-ct dataset and transformer baseline for volumetric super-resolution from ct scans. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 344–353. Springer, 2022. 3, 6, 7

[46] Guangming Zang, Ramzi Idoughi, Rui Li, Peter Wonka, and Wolfgang Heidrich. Intratomo: self-supervised learning-based tomography via sinogram synthesis and prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1960–1970, 2021. 3

[47] Xiaole Zhao, Yulun Zhang, Tao Zhang, and Xueming Zou. Channel splitting network for single mr image super-resolution. *IEEE Transactions on Image Processing (IEEE TIP)*, 28(11):5649–5662, 2019. 2, 3