RING-NeRF : Rethinking Inductive Biases for Versatile and Efficient Neural Fields

Doriand Petit[1,2], Steve Bourgeois[1], Dumitru Pavel[1], Vincent Gay-Bellile[1], Florian Chabot[1], and Loïc Barthe[2]

[1] Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
[2] IRIT, Université Toulouse III, CNRS, France

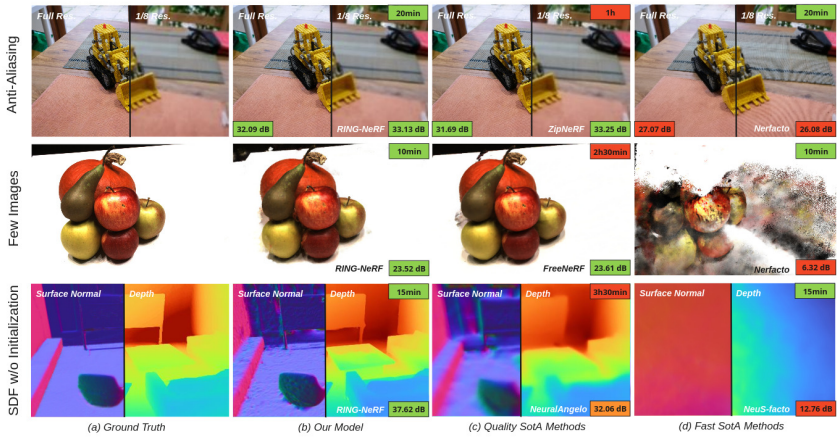

Fig. 1: RING-NeRF is a simple and versatile architecture which tackles many NeRF common issues such as robustness to distance of observation, few view supervision and lack of scene-specific initialization for SDF-based reconstruction. It provides on-par performances in terms of quality with SotA dedicated solutions [4, 11, 25] and and in terms of efficiency with fast methods [20, 26].

Abstract. Recent advances in Neural Fields mostly rely on developing task-specific supervision which often complicates the models. Rather than developing hard-to-combine and specific modules, another approach generally overlooked is to directly inject generic priors on the scene representation (also called inductive biases) into the NeRF architecture. Based on this idea, we propose the RING-NeRF architecture which includes two inductive biases : a continuous multi-scale representation of the scene and an invariance of the decoder's latent space over spatial and scale domains. We also design a single reconstruction process that takes advantage of those inductive biases and experimentally demonstrates on-par performances in terms of quality with dedicated architecture on multiple tasks (anti-aliasing, few view reconstruction, SDF reconstruction without scene-specific initialization) while being more efficient. Moreover, RING-NeRF has the distinctive ability to dynamically increase the resolution of the model, opening the way to adaptive reconstruction. Project page can be found at : `https://cea-list.github.io/RING-NeRF`

Keywords: NeRF · 3D Reconstruction · Implicit Neural Representation

1 Introduction

Neural Radiance Fields (NeRF) have emerged as a novel method for representing 3D scenes using neural networks. In its original design [13], a simple multi-layer perceptron (MLP) is trained to reproduce a continuous 5D function that outputs the density and radiance emitted in each direction (θ, ϕ) at each point (x, y, z) in space. This approach inspired many works due to the impressive quality of its novel view synthesis as well as its simplicity. However, the so-called "vanilla" NeRF architectures converge very slowly as they use large deep neural networks. Instant-NGP [14] introduced a new architecture based on a hierarchical hash-grid pyramid of learnable feature codes describing the 3D scene and combined with a much shallower MLP, called decoder, that transforms the concatenation of the codes, interpolated from the grids, into density and radiance. Resulting in local updates, this method reduced the training duration from hours to minutes.

The great majority of current solutions are now based on these two standard architectures, and many of them are focused on overtaking their associated limitations in terms of:
Nature of scene - by transitioning from object-centric scenes to open unbounded scenes, using mostly space contraction [3];
Robustness - by managing free motion trajectories with variations of the observation distance while avoiding aliasing artefacts, mostly through the integration of Level of Detail (LOD) in the model [2, 4]; or by reducing drastically the number of supervised views through different kinds of regularization [8, 15, 25];
Extensibility - by shifting from a holistic and fixed reconstruction process to an incremental (extensibility in the number of views) and adaptive (extensibility in resolution) process, mainly through the use of a frozen decoder [12, 29].

However, most solutions solely focus on one of these limitations and introduce specific and complex mechanisms that both increase the computational cost and lessen the possibilities of combination. The ability to solve jointly these main issues is however essential in real-world applications which often require a robust and extensible reconstruction in an unknown and unbounded environment.

In this article, rather than introducing yet another heavy and task-specific solution, we propose RING-NeRF, a versatile and simple NeRF architecture by rethinking usual grid-based models to introduce two inductive biases. We first represent a 3D scene as a continuous multi-scale representation and also make the decoder's latent space invariant in position and scale. Together, these two priors enable the production of intrinsic continuous LOD of the scene without explicit supervision. We demonstrate experimentally that, when combined with adapted cone casting and coarse-to-fine optimization, the resulting architecture is able to compete on several tasks with on-par quality performances with dedicated state-of-the-art solutions while improving speed, robustness and extensibility. The overall process is also simple, easy to implement and generic enough to be coupled with specific solutions. Our contributions can be summarized as follows:

1. an architecture that, by construction, represents the scene with a continuous and unbounded level of detail without the need for LOD-specific supervision and which permits resolution-adaptive reconstruction;
2. a distance-aware forward mapping compatible with scene contraction, that takes benefits of the continuous multi-scale representation with an adapted cone casting process to avoid aliasing artefacts when varying the observation distance.
3. a continuous coarse-to-fine reconstruction process that improves the convergence and stability (especially in challenging setups such as supervision with few viewpoints or no scene-specific initialization for SDF reconstruction).

2 Related Work

From its original iteration [13], a majority of current research focuses on overtaking limitations of NeRF-based reconstruction in terms of adaptability to various natures of scene, robustness (to varying observation distances or limited amount of viewpoints) and reconstruction extensibility.

Adaptability to various natures of the scene. The ability of Neural Fields to reconstruct various natures of scene depends on three factors. The first one is related to its architecture itself. Tri-plane architectures [5, 7, 30] are mostly designed for object-centric reconstruction (as they provide a higher density of information in the center of the scene) while vanilla and 3D grid-based NeRF are able to cover a wider variety of scenes, though they initially were still constrained to a limited volume. A second factor is related to the representation of the 3D space of the scene, especially to represent distant elements in open scenes. Some approaches use two different NeRF models to reconstruct separately the foreground and the background [28], whereas others apply space contraction to the 3D scene coordinates [3, 4, 28] to map the infinite scene volume into a bounded one. The last factor is related to the initialization of the Neural Fields. While the random initialization of density-based NeRF can adjust to almost any nature of scenes, the convergence of SDF-based (Signed Distance Function) Neural Fields is extremely sensitive to their parameters' initialization, as stated in [1]. Current solutions rely on a scene-specific initialization (using an SDF field representing a sphere for outdoor scenes or an inverted sphere for indoor scenes), making them unable to adapt automatically to any scene.

Robustness to observation distance variations. The initial NeRF model, as well as most of the subsequent works, relies implicitly on the hypothesis of a constant distance of observation to the scene. Indeed, the NeRF model provides a per-3D-point scene density representation and a rendering process which does not take into account the increasing volume covered by a pixel with respect to the distance to the cameras. As underlined in Mip-NeRF [2], this discrepancy induces artefacts such as over-contrasted images or aliasing phenomenon when the distance of observation differs from the ones used at reconstruction time. To avoid these artefacts, the rendering process needs to assess the density and color for a volume instead of a point. In the current state-of-the-art, two main

approaches are used (or their combination, as in Zip-NeRF [4]). The first solution consists of representing the scene with different levels of detail (LOD), in order to vary the precision of the reconstruction based on the observation distance. This is usually done by using a LOD-aware latent space [2,4,6,19,21,24], meaning that the LOD information is already encoded in the inputs of the MLP. This can be achieved by using a per-LOD decoder as in PyNeRF [21] or by incorporating LOD information in the latent feature as done in Zip-NeRF [4] and VR-NeRF [24]. One main flaw of these solutions is that they require the supervision of every used LOD which makes it impossible to vary the observation distances between the train views and novel synthesised views. A second approach consists of defining the latent representation of a volume as the mean of the latent features of the points included in the volume. It requires integrating the features over the volume, which can be achieved through convolutions for tri-plan representation [7,30], or through super-sampling of the latent space for 3D grids as also done in Zip-NeRF [4]. However, these approaches lengthen the training and rendering processes as they increase the number of computations.

Robustness to limited amount of viewpoints. By construction, NeRF is subject to the shape-radiance ambiguity [28]. If not enough supervision viewpoints are available, the optimization might overfit them while not providing consistent 3D reconstruction nor generalization to non-supervised viewpoints. A first family of solutions to overcome this limitation consists of regularizing the reconstruction process through additional losses, whether via geometric [15,23,27] or semantic [8] regularization. The second family of solutions relies on a progressive reconstruction of the details of the scene, as introduced in FreeNeRF [25] and Nerfies [17]. Restraining the ability of the model to reconstruct a complex scene at early stages enforces the consistency of the reconstruction over the different supervision viewpoints. This presents the advantage of bringing stability while keeping a fairly simple training process. However, these latter solutions mostly rely on Vanilla models for stability purposes and require long training duration.

Reconstruction extensibility. Two kinds of extensibility should be distinguished: extensibility with respect to the number of views or the resolution. The first one is related to reconstructing new scene areas while keeping the previously reconstructed ones unchanged. The usual solution consists of using grid-based approaches with a pre-trained position-invariant decoder that is frozen during the reconstruction [29]. On the other side, the extensibility of the resolution consists in dynamically increasing the level of detail of a previously reconstructed scene. This problem is intractable for classic architectures such as vanilla NeRF and I-NGP since the number of layers (resp. grids) of those approaches cannot increase during the reconstruction. Some rare solutions, such as Neural Sparse Voxel Fields [12], combine a pre-trained decoder with a data structure allowing to dynamically allocate voxels to increase the reconstruction resolution. This is however a crucial stake, as being able to adapt the precision of the representation based on the complexity of the scene permits to optimize the computational efficiency (both in speed and memory requirements) with minimal loss in quality.

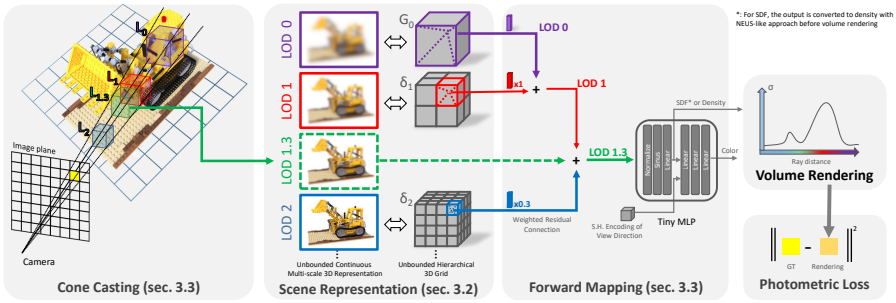

Fig. 2: Overview of RING-NeRF: to render a pixel, the casted cone is sampled with cubes. Depending on the cube volume, the corresponding LOD of the scene is selected and the latent feature is computed using a weighted sum of the grid hierarchy. The density (or SDF) and color of the cube are first decoded from the latent feature with a tiny MLP and then integrated with other samples through volume rendering.

3 RING-NeRF

Rather than focusing on solving one specific problem of NeRF using complex mechanisms, we propose a simple architecture called RING-NeRF constructed with novel inductive biases to tackle NeRF's common issues.

3.1 Overview

RING-NeRF relies on the classic NeRF [13] inverse rendering pipeline which is used to reconstruct a 3D scene from a set of localized frames. For a given image pixel (with its camera's pose), a 3D ray is cast and the 3D scene representation is sampled at N various locations along the ray. The resulting density σ_i (or SDF converted to density [22]) and color c_i of the samples are then combined with usual volume rendering techniques: $\hat{C}(r) = \sum_{i=0}^{N-1} T_i(1-exp(-\sigma_i\delta_i))c_i$ with $T_i = exp(\sum_{j=0}^{i-1} \sigma_j\delta_j)$ and δ_i is the distance between samples. The parameters of the scene representation are then optimized by minimizing the MSE loss $||\hat{C}(r)-C(r)||^2$ between the rendered ray color and the ground truth pixel value, the rendering being differentiable[3].

The originality of RING-NeRF relies on its neural architecture, illustrated in figure 2, which is designed to represent the scene with a continuous and unbounded level of detail without the need for LOD-specific supervision (see section 3.2). We then use this LOD inductive bias to adjust the LOD of the samples with respect to the distance to the camera in contracted space for more accurate renderings (see section 3.3). Finally, combined with a continuous coarse-to-fine optimization, RING-NeRF results in a more robust reconstruction process with an intrinsic LOD extensibility property (see section 3.4).

[3] With SDF, an additional Eikonal loss is also being optimized for SDF regularization.

3.2 NeRF Representation with Inductive Biases

Classic grid-based representations, such as the one introduced by I-NGP [14], rely on two key elements: a unique MLP decoder which transforms a latent space into an output space (color, density or SDF), and a 3D (hierarchical) grid of latent features that implicitly defines a unique mapping function from the scene space onto the decoder latent space. We can distinguish two different approaches used to extend the mono-scale representation to a multi-scale representation: conditioning with scale information the decoding process [4,21] and/or defining separate per-LOD mapping functions [21]. In both cases, the training becomes more complex, leading to potential convergence issues as it relies solely on additional specific supervision. Instead, we propose to condition the mapping function itself with scale information and introduce inductive biases to guide the convergence.

Scene representation as multi-scale mapping function. As illustrated in Figure 2, the mapping function at level N is controlled by a grid G_N that is implicitly defined through a recursive refinement process over scale:

$$\begin{cases} G_0 \leftarrow \delta_0 \\ G_N \leftarrow S(G_{N-1}) + \delta_N \end{cases} \tag{1}$$

where δ_i is the deviation defined by the i-th latent feature grid and S is the subdivision scheme consisting in increasing the resolution of G_{N-1} up to the N-th grid resolution (with tri-linear interpolation). In practice, the latent vector associated with a point in the scene for a level of detail N is simply obtained by interpolating linearly the point in the cell for each level inferior or equal to N, and summing the results. Since linear interpolation is differentiable, our mapping function is also optimizable. The summing of the interpolation is comparable to a residual connection and gives the name of the architecture RING-NeRF for Residual Implicit Neural Grids.

Such representation provides several advantages. First, since the mapping function is constructed in a top-down manner, refining recursively one level of detail from the previous one, its number of LOD is unbounded and adding a finer level of detail keeps the coarser ones valid. Secondly, because each δ_i only represents a deviation of the coarser mapping value G_{i-1}, a continuous LOD representation can be easily obtained by incorporating a weighting factor $\alpha \in [0, 1]$ in the recursive process:

$$G_L \leftarrow S(G_{N-1}) + \alpha\delta_N \tag{2}$$

with $L \in [N-1, N]$ and $\alpha = L - (N-1)$.

Spatial and scale invariance of decoder. Since our architecture does not rely on a decoder conditioned on position (unlike [19]) nor scale (unlike [2–4, 21]), the decoder latent space is invariant to translations and scale changes in the scene coordinate. This property makes RING-NeRF more suited for incremental reconstruction in both spatial and scale space, since it ensures that local updates in the spatial and scale domain of the scene can be achieved through the

Fig. 3: We demonstrate the LOD inductive bias by training our model with a hierarchy of $N = 7$ levels where only the last level of the mapping function G_N is supervised. We then compute renders at different levels $L \leq N$. Other examples (including of entire scenes) can be found in the supplementary materials.

hierarchical grid. The decoder architecture is illustrated in figure 2 and further details can be found in the supplementary materials.

LOD inductive bias. During the reconstruction process, the gradients of the ray samples are backpropagated through the residual connections and aggregated for each grid level. Due to the pyramidal resolution of the hierarchical grid, a grid code at coarse levels influences a large scene volume and is thus supervised by more ray samples. Therefore, the gradient of a grid code increases as the level's resolution decreases, and as long as the associated samples' gradients are uniform. However, once the backpropagation through the residual connections reach a level whose associated samples have divergent enough gradients (meaning the error is finer than this level's resolution), the result of the aggregation will be mitigated. Hence, the level with the maximum correction is always the coarser level where the samples' gradients are still uniform. This property naturally induces corrections at the proper grid level and LOD. In figure 3, we illustrate this inductive bias by displaying different continuous LOD of a scene while the reconstruction is only supervised for its finest LOD. Not only this is a useful property when an unsupervised multi-scale representation is needed (as demonstrated in section 4.2), this also guarantees a more robust convergence (as illustrated in section 4.3 and section 4.4).

3.3 Distance-Aware Forward Mapping

Cone Casting in contracted space. In order to accommodate the scene rendering process to the variation of observation distances, we introduce a distance-aware forward mapping mechanism. Similarly to the cone casting of [7], it relies on assigning to each sample a latent feature whose LOD is inversely proportional to the sample-camera distance. However, unlike [7, 21], our model relies on the use of space contraction to allow the reconstruction of unbounded scenes.

To define the LOD of a sample at distance d of the camera, we first compute the associated volume of the cone cast pixel cube in the world space coordinates (see fig. 2). Assuming the pixel is a square of size c at distance 1 of the camera (c depends on the image resolution and camera's FOV), the volume is thus $V = (dc)^3$. To take into account the space contraction, we then proceed to contract the volume. Denoting J the Jacobian of the contraction function at the sample's location p, $V_{contract} = Vdet(J(p))$. In practice, we compute the

analytical derivation of the contraction function depending on p. More details can be found in the supplementary materials.

Assuming the N-th feature grid's resolution in our hierarchy can be written as $f^N b$ with b the resolution of the grid δ_0 and f the growth factor, finding the appropriate LOD $L \in \mathbb{R}^+$ means finding a virtual grid[4] of resolution $f^L b$ whose cell's volume is equal to the previously computed volume (in contracted space). Because we are working in the contracted space of size 1, the volume of a cell of the virtual grid of LOD L is $(\frac{1}{f^L b})^3$. The LOD L associated to one sample in the contracted space is thus given by:

$$(dc)^3 \det(J(p)) = \left(\frac{1}{f^L b}\right)^3 \iff L = -\frac{\log\left(dcb\sqrt[3]{\det(J(p))}\right)}{\log(f)} \tag{3}$$

Note that this process is close to ZipNeRF [4] and VR-NeRF [24]. However, the first one rather derives a contracted scale factor from its Gaussian samples while the latter directly computes the LOD in the contracted space, which can be considered an approximation of our computation.

Forward mapping. As illustrated in Figure 2, we use the determined LOD to compute a distance-dependent weighted sum of the features, which is fed to our decoder and transformed into density (or SDF) and radiance.

3.4 Continuous Coarse-To-Fine and Resolution Extensibility

Recent works proposed to use coarse-to-fine optimization to improve the stability of NeRF models, especially when facing more challenging setups, including with few images [25] and surface-based models [11]. It consists of optimizing progressively the different LOD of the representation, from coarse levels to the most precise ones. The goal of this progressive optimization is to avoid the shape-radiance ambiguity [28] by introducing a strong regularization through LOD restriction, then relaxing progressively this regularization once the coarse geometry of the scene is reconstructed to recover the details of the scene.

The coarse-to-fine reconstruction process of RING-NeRF consists of estimating progressively the LOD of the mapping function from the coarsest to the finest ones. In practice, it implies, during the cone casting, to clamp the samples' LOD up to a maximal LOD l, the grids of level l and above being set to zero and not optimized. Moreover, since our architecture provides continuous LOD, the coarse-to-fine optimization can be achieved continuously in the LOD space by using a linear scheduler $l = (l_0 + \frac{n}{n_{ctf}}) \in \mathbb{R}^+$ with n the current epoch, n_{ctf} a hyperparameter describing the speed of the process and l_0 defining the number of used grids at initialization; up to a specified maximum resolution.

Furthermore, the RING-NeRF architecture is more adapted than I-NGP-based architectures [11] for coarse-to-fine training. Indeed, for solutions based on the concatenation of features, keeping grid values to zero implies that some

[4] A virtual grid of LOD $L \in \mathbb{R}^+$ corresponds to a grid of resolution $f^L b$ that is not explicitly stored in memory but whose elements can be computed from other grids.

dimensions of the decoder's latent space are not supervised. When a grid starts to be optimized, those unsupervised dimensions are suddenly used. The weights of the decoder thus need to be refined, with a global effect on the whole scene. On the opposite, our solution keeps supervising all the dimensions of the decoder latent space, and when a new grid gets optimized, it only implies more degrees of freedom to define the mapping function between the scene space and the decoder latent space. This also means that our scene reconstruction can be refined by adding dynamically new grid levels without modifying the decoder's weights or previously trained grids, as we demonstrate experimentally in section 4.5. This resolution extensibility property opens the path to adaptive resolution models, where the precision used to describe an area depends on the details needed, to optimize efficiency both in memory consumption and training duration.

4 Experiments

In these experiments, we intend to highlight the versatility of RING-NeRF by evaluating it on several tasks. After introducing implementation details in section 4.1, we evaluate our model on novel view synthesis with changes in observation distances (sec. 4.2). Then, we explore how robust is RING-NeRF first with few view reconstruction (sec. 4.3) and then without scene-specific initialization for SDF reconstruction (sec. 4.4). Finally, we demonstrate the capacity of our architecture to perform LOD extensibility, as a first step towards adaptive reconstruction (sec. 4.5).

4.1 Implementation

Our model is based on the PyTorch framework *Nerfstudio* [20]. We build upon its core method named Nerfacto, which combines ideas from several papers for fast and qualitative renders of unbounded complex scenes. This makes it an accessible baseline with a state-of-the-art quality/time ratio. Because NeRF pipelines contain a high number of small but decisive choices of implementation (eg. some frameworks choose to train their models image by image while Nerfstudio jointly and randomly samples across all images), we decided to use as much as possible Nerfstudio-based baselines for fairer comparisons. All of these models are trained on one Nvidia-A100 GPU. The reported times correspond to the approximated training duration of the models. Configuration details, further experiments and ablatives on our contributions can be found in the supplementary materials.

4.2 Novel View Synthesis and Anti-Aliasing

This experiment aims to evaluate the reconstruction quality through the ability to synthesize viewpoints that are not supervised during the reconstruction. These new viewpoints differ from the angle of observation, but also from the distance of observation. The latter is particularly important since a reconstruction or a

Table 1: Novel View Synthesis performances for the **Mono-Scale setup** (trained on the full resolution images only) on the 360 Dataset. The indicated resolutions refer to the resolution of the renders.

	Full Res.			1/2 Res.			1/4 Res.			1/8 Res.			
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Time ↓
Nerfacto [20]	27.09	0.779	0.181	26.81	0.782	0.162	25.22	0.711	0.234	23.32	0.636	0.297	0.45h
PyNeRF [21]	27.87	0.802	0.160	21.93	0.713	0.211	21.15	0.662	0.264	20.39	0.619	0.312	0.96h
Zip-NeRF [4]	28.06	0.808	0.154	16.58	0.596	0.319	11.88	0.424	0.465	9.66	0.323	0.523	1.10h
RING-NeRF	28.09	0.799	0.157	27.18	0.804	0.138	25.82	0.786	0.142	24.38	0.737	0.167	0.45h

rendering process that does not take correctly into account the distance of observation leads to artefacts, from over-contrasted rendering to aliasing phenomenon. The challenge is to avoid these artefacts while keeping the reconstruction and rendering process as fast as possible.

Dataset. The evaluation relies on the dataset introduced by Mip-NeRF-360 [3]. This dataset is composed of 9 scenes, each containing both a central area and complex background in both inside and outside setups. Since the trajectory keeps a constant distance to the central part, each viewpoint is represented with a pyramid of 4 different image resolutions to simulate a variation of the distance of observation combined with a change of the camera FOV, following the Mip-NeRF [2] original anti-aliasing evaluation pipeline.

Algorithms. We compare our solution against several grid-based NeRF baselines, both with (PyNeRF [21] and Zip-NeRF [4]) and without anti-aliasing processing (Nerfacto), using their Nerfstudio implementations.

Protocol. For each scene, we train the models with two different setups: the *mono-scale* setup that uses only the image with the highest resolution for each viewpoint, and the *multi-scale* setup which uses the whole pyramid of images for each viewpoint. Note that, for these two setups, we evaluate the performances on the whole resolution pyramid, with usual metrics (PSNR, SSIM, LPIPS).

Results. First of all, regarding novel view synthesis quality using the mono-scale setup for both training and testing, as referred to in the "Full Res." column of table 1, we notice that our architecture provides on par performances with Zip-NeRF, slightly better results than PyNeRF, and a more important gap with Nerfacto. While simple, RING-NeRF succeeds in performing state-of-the-art performances on a real single-scale dataset. Regarding quality for the multi-scale setup, as presented in table 2, we observe that all the algorithms that consider the distance of observation perform very similarly in terms of quality. On the

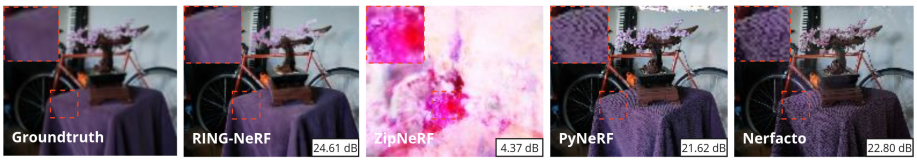

Fig. 4: Examples of image renderings at 1/8th resolution from models solely trained with the full resolution images. RING-NeRF is the only method capable of producing coherent aliasing-free renderings thanks to its LOD inductive bias.

Table 2: Novel View Synthesis performances for the **Multi-Scale setup** (trained jointly on every resolution) on the 360 Dataset. The indicated resolutions refer to the resolution of the renders.

	Full Res.			1/2 Res.			1/4 Res.			1/8 Res.			
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Time ↓
Nerfacto [20]	25.98	0.668	0.367	27.31	0.792	0.176	27.16	0.805	0.139	25.30	0.743	0.199	0.45h
PyNeRF [21]	27.65	0.781	0.194	29.47	0.857	0.092	30.51	0.887	0.063	30.86	0.893	0.055	0.96h
Zip-NeRF [4]	27.54	0.781	0.191	29.34	0.858	0.099	30.36	0.889	0.064	30.90	0.900	0.047	1.10h
RING-NeRF	27.66	0.782	0.194	29.55	0.856	0.094	30.51	0.888	0.062	30.86	0.901	0.048	0.45h

other side, Nerfacto performs quite poorly. Its performances are especially low at the coarsest resolution, with overflowing artefacts, as illustrated in figure 1.

Finally, we evaluate the capacity of the models to generalize over new resolutions by training them on full-resolution images and then evaluate them on smaller resolutions. Qualitative results are shown in figure 4 while quantitative results can be found in the 1/2-th, 1/4-th and 1/8-th res. columns of table 1. Because most anti-aliasing methods need LOD-specific supervision to correctly function (including Zip-NeRF and PyNeRF), they cannot render coherent anti-aliased images in this setup. However, PyNeRF behaves better than ZipNeRF with coherent although very aliased renderings, as it decides to limit the LOD used for rendering inside the range of LOD seen during training. RING-NeRF, with his LOD inductive bias, is the only architecture capable of producing anti-aliased renderings from novel observation distances and thus outperforms every other method. While this experiment can seem somewhat esoteric as training on a multi-resolution images pyramid is rather easy, this increases GPU memory and total training time. Moreover, depending on the resolution and the trajectory, it is not trivial to choose the accurate number of scales in the image pyramid to supervise correctly every grid in the hierarchy and especially the coarsest grids.

Regarding the reconstruction processing time reported in the "Time" column of both table 1 and table 2, since RING-NeRF does not rely on the multiplication of either sample or decoder, it processes as quickly as the fastest Nerfacto both on mono-scale and multi-scale setups. On the other hand, the other anti-aliasing methods, PyNeRF with its per-LOD MLP and Zip-NeRF with its super-sampling, are approximately 2.5 times slower.

In conclusion, our solution provides the best quality-speed trade-off since it is both on par with the best quality method and the fastest method, in mono-scale as well as in multi-scale setups. Furthermore, RING-NeRF is the only solution capable of creating coherent and anti-aliased renderings when facing novel observation distances unseen during training.

4.3 Few Viewpoints Supervision

This experiment aims to evaluate the influence of the RING-NERF architecture and pipeline on the reconstruction robustness to limited supervision viewpoints.
Dataset. We evaluate our contribution on the object-centric real dataset DTU [9], often used in few-viewpoints evaluations.
Algorithms. We compare our architecture against several baselines: Mip-NeRF

[2], and FreeNerf [25] (a state-of-the-art method for this task), for vanilla architectures and Nerfacto for grid-based architectures. To better demonstrate the intrinsic stability brought by RING-NeRF, we also developed the Nerfacto+ architecture, which corresponds to a Nerfacto architecture coupled with a coarse-to-fine training based on a progressive activation of the grids (the rest of the decoder's input being filled with zeros). For Nerfacto+ and RING-NeRF, we also add FreeNeRF's loss that penalizes the density of the first $M = 10$ samples of each ray to reduce as much as possible artefacts in front of the cameras. As an ablative experiment, we also evaluate RING-NeRF using discrete coarse-to-fine (fixed LOD increment of 1 rather than the proposed continuous increase).

Protocol. We follow FreeNeRF's evaluation pipeline, including the number of supervision viewpoints (3 to 9), the choice of these views among the dataset and the evaluations using masks of the object. Since we are using the same protocol, the results of FreeNeRF and Mip-NeRF were taken out of FreeNeRF's article.

Results. Evaluation results are reported in table 3. We first notice an important difference between vanilla and grid-based baselines. While Mip-NeRF faces troubles in reconstructing the scene with 3 views, the method seems to find coherency when adding more images. However, Nerfacto struggles much more to form a consistent 3D scene even when using 9 images (see figure 1). Even though the grid-based baseline is way faster to train, its design implies more instability when facing a small number of images. This does not mean however that few viewpoints are incompatible with grid-based methods. Using progressive training coupled with a very simple and generalizable regularization, both Nerfacto+ and our architecture succeed in creating coherent geometry. Nonetheless, RING-NeRF considerably outperforms Nerfacto+, with a PSNR difference varying from 3 to 4, demonstrating the stability increase brought by our architecture, and also outperforms Mip-NeRF for the configuration with 3 and 6 supervision images. The discrete coarse-to-fine version of RING-NeRF performs in-between Nerfacto+ and the complete RING-NeRF. This showcases both the intrinsic interest of the proposed architecture against the Nerfacto+ and the relevance of the continuous coarse-to-fine mechanism. FreeNerf remains the best performer of all in terms of quality, but with a reconstruction time that is extremely slower than RING-NeRF, the former achieving its reconstruction in 2.56 hours while the latter only requires less than 10 minutes. RING-NeRF thus offers a better quality-speed trade-off for the few-view reconstruction issue (see figure 1).

Table 3: Performances of reconstruction from few viewpoints on the DTU dataset. The reported metrics are computed based on the mask of the object.

	PSNR ↑			SSIM ↑			LPIPS ↓			
#Images	3	6	9	3	6	9	3	6	9	Time ↓
Mip-NeRF [2]	8.68	16.54	23.58	0.571	0.741	0.879	0.353	0.198	0.092	2.56h
FreeNeRF [25]	19.92	23.25	25.38	0.787	0.844	0.888	0.135	0.095	0.067	2.56h
Nerfacto [20]	9.35	9.75	9.78	0.567	0.604	0.647	0.385	0.331	0.326	0.15h
Nerfacto+	13.61	16.61	19.33	0.639	0.699	0.759	0.276	0.218	0.151	0.15h
RING-NeRF	16.18	20.47	23.19	0.713	0.808	0.847	0.200	0.127	0.085	0.15h
w/ discrete CtF	15.79	20.16	22.93	0.706	0.785	0.847	0.201	0.127	0.085	0.15h

Table 4: SDF reconstruction performances when foregoing the scene-specific SDF Initialization on the Replica Dataset. The Chamfer distance is in centimeters.

	PSNR ↑	SSIM ↑	LPIPS ↓	Chamfer-L1 ↓	Training Time ↓
NeuS-facto [26]	24.62	0.778	0.347	17.61	1.4h
NeuralAngelo [11]	30.79	0.916	0.0761	13.69	3.40 h
RING-NeRF	37.18	0.969	0.0194	5.71	1.28 h

4.4 SDF Reconstruction without Initialization

SDF reconstruction is known to be a more unstable process than density-based NeRF [1], requiring a scene-specific initialization to converge. This initialization becomes an issue in complex environments and incremental setups, where several types of scenes can co-exist and are not necessarily known beforehand. Therefore, in this experiment, we evaluate the ability of RING-NeRF and other SotA architectures to achieve SDF reconstruction without scene-specific initialization.
Dataset. The evaluation is achieved on a subset of 7 scenes of the *Replica* [18] synthetic indoor dataset. A Tanks & Temples [10] example is provided in supplementary materials with corresponding analysis.
Algorithms. We compare our architecture to two SDF methods, all of them implemented in the same *SDFStudio* [26] branch of the *Nerfstudio* framework for fairer comparisons: NeuS-facto, an adaptation of NeuS for grid-based methods with Nerfacto modules, and an implementation of NeuralAngelo. For these experiments, our model is built upon the NeuS-facto baseline, using in particular the same NeuS-based SDF-to-density transformation [22].
Protocol. To evaluate the impact of the architecture and pipeline over the convergence and stability, we suppress the inverted sphere SDF initialization scheme and use a random initialization for the model.
Results. Evaluation results are shown in table 4. First of all, NeuS-facto faces low rendering and reconstruction metrics, due to catastrophic failure in most of the tested scenes (see figure 1) since the model tends to re-draw the 2D images in front of the camera. Regarding NeuralAngelo, its relatively high PSNR demonstrates its ability to synthesize satisfying RGB. However, as illustrated in figure 1 and highlighted by the reconstruction metrics, the underlying geometry of the scene is poorly reconstructed, without any fine details. Finally, our method RING-NeRF is by far the best performer, with much higher PSNR and a better geometry including fine details (see figure 1), although a bit noisy. The simple architecture of RING-NeRF also permits faster epochs, thus faster training.

4.5 LOD Extensibility

This experiment aims to demonstrate RING-NeRF's unique ability to increase dynamically the level of detail of the scene representation.
Dataset. The scan 114 of the DTU dataset is used for this experiment.
Algorithms. Because I-NGP architectures [4, 11, 14] cannot perform LOD extensibility (due to the fixed decoder's input size), only RING-NeRF is evaluated.
Protocol. We train our model using two different configurations : one low resolution with a 3 levels grid hierarchy and one high resolution with 5 levels (the

Fig. 5: Learning curves and final renderings of RING-NeRF models with different grid configurations trained either jointly or incrementally.

grid resolutions are 16, 32, 64, 128 and 256). For these two configurations named "3 levels" and "5 levels", every grid and the decoder are trained simultaneously. We proceed to showcase the extensibility of RING-NeRF by adding grids to the "3 levels" configuration that is previously trained. We first train one grid of resolutions 128 ("3+1 levels" in Figure 5) with the three initial grids and the decoder frozen and then train another grid of resolutions 256 ("3+1+1 levels" in Figures 5) with the four grids and the decoder frozen.

Results. Figure 5 shows that the configuration "3+1+1 levels" results in the same rendering quality than the "5 levels" one. This demonstrates the ability of our model to dynamically change the resolution of the grid hierarchy. This is an important step towards the development of an adaptive architecture which locally chooses the resolution of the representation based on the scene's content. This allows to drastically reduces the number of parameters, helping in improving the memory footprint, the training duration and the model's robustness.

5 Conclusion and Perspectives

In this work, we introduced RING-NeRF, a simple and versatile NeRF pipeline that provides two inductive biases by design: a continuous multi-scale representation of the scene, and an invariance of the decoder latent space over spatial and scale domains. Coupled with a distance-aware forward mapping and a continuous coarse-to-fine reconstruction process, our pipeline demonstrated experimentally its versatility with on-par performances with dedicated state-of-the-art solutions for anti-aliasing or reconstruction from few viewpoints. It even outperforms them in terms of robustness to scene-specific initialization for SDF reconstruction. Furthermore, it is highly efficient and is not limited to object-centric scenes.

Future work will study the impact of RING-NeRF on other challenging use cases, such as facing inaccurate camera poses [16] and SLAM [29]. We will also use the extensibility property of our architecture to develop memory-efficient sparse Neural Fields, which is considered to be a limit of most grid-based models.

Acknowledgements

This publication was made possible by the use of the CEA List FactoryIA supercomputer, financially supported by the Ile-de-France Regional Council.

References

1. Atzmon, M., Lipman, Y.: Sal: Sign agnostic learning of shapes from raw data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2565–2574 (2020)
2. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5855–5864 (2021)
3. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5470–5479 (2022)
4. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Zip-nerf: Anti-aliased grid-based neural radiance fields. ICCV (2023)
5. Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H.: Tensorf: Tensorial radiance fields. In: European Conference on Computer Vision (ECCV) (2022)
6. Dou, Y., Zheng, Z., Jin, Q., Ni, B.: Multiplicative fourier level of detail. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1808–1817 (2023)
7. Hu, W., Wang, Y., Ma, L., Yang, B., Gao, L., Liu, X., Ma, Y.: Tri-miprf: Tri-mip representation for efficient anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19774–19783 (2023)
8. Jain, A., Tancik, M., Abbeel, P.: Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5885–5894 (2021)
9. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 406–413 (2014)
10. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Transactions on Graphics **36**(4) (2017)
11. Li, Z., Müller, T., Evans, A., Taylor, R.H., Unberath, M., Liu, M.Y., Lin, C.H.: Neuralangelo: High-fidelity neural surface reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8456–8465 (2023)
12. Liu, L., Gu, J., Zaw Lin, K., Chua, T.S., Theobalt, C.: Neural sparse voxel fields. Advances in Neural Information Processing Systems **33**, 15651–15663 (2020)
13. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
14. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Transactions on Graphics (ToG) **41**(4), 1–15 (2022)

15. Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S., Geiger, A., Radwan, N.: Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5480–5490 (2022)
16. Park, K., Henzler, P., Mildenhall, B., Barron, J.T., Martin-Brualla, R.: Camp: Camera preconditioning for neural radiance fields. ACM Transactions on Graphics (TOG) **42**(6), 1–11 (2023)
17. Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: Nerfies: Deformable neural radiance fields. ICCV (2021)
18. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., et al.: The replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
19. Takikawa, T., Litalien, J., Yin, K., Kreis, K., Loop, C., Nowrouzezahrai, D., Jacobson, A., McGuire, M., Fidler, S.: Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11358–11367 (2021)
20. Tancik, M., Weber, E., Ng, E., Li, R., Yi, B., Wang, T., Kristoffersen, A., Austin, J., Salahi, K., Ahuja, A., et al.: Nerfstudio: A modular framework for neural radiance field development. In: ACM SIGGRAPH 2023 Conference Proceedings. pp. 1–12 (2023)
21. Turki, H., Zollhöfer, M., Richardt, C., Ramanan, D.: Pynerf: Pyramidal neural radiance fields. In: Thirty-Seventh Conference on Neural Information Processing Systems (2023)
22. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. NeurIPS (2021)
23. Wynn, J., Turmukhambetov, D.: DiffusioNeRF: Regularizing Neural Radiance Fields with Denoising Diffusion Models. In: CVPR (2023)
24. Xu, L., Agrawal, V., Laney, W., Garcia, T., Bansal, A., Kim, C., Rota Bulò, S., Porzi, L., Kontschieder, P., Božič, A., Lin, D., Zollhöfer, M., Richardt, C.: VR-NeRF: High-fidelity virtualized walkable spaces. In: SIGGRAPH Asia Conference Proceedings (2023). https://doi.org/10.1145/3610548.3618139
25. Yang, J., Pavone, M., Wang, Y.: Freenerf: Improving few-shot neural rendering with free frequency regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8254–8263 (2023)
26. Yu, Z., Chen, A., Antic, B., Peng, S., Bhattacharyya, A., Niemeyer, M., Tang, S., Sattler, T., Geiger, A.: Sdfstudio: A unified framework for surface reconstruction (2022), https://github.com/autonomousvision/sdfstudio
27. Yu, Z., Peng, S., Niemeyer, M., Sattler, T., Geiger, A.: Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. Advances in neural information processing systems **35**, 25018–25032 (2022)
28. Zhang, K., Riegler, G., Snavely, N., Koltun, V.: Nerf++: Analyzing and improving neural radiance fields. arXiv preprint arXiv:2010.07492 (2020)
29. Zhu, Z., Peng, S., Larsson, V., Xu, W., Bao, H., Cui, Z., Oswald, M.R., Pollefeys, M.: Nice-slam: Neural implicit scalable encoding for slam. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12786–12796 (2022)
30. Zhuang, Y., Zhang, Q., Feng, Y., Zhu, H., Yao, Y., Li, X., Cao, Y.P., Shan, Y., Cao, X.: Anti-aliased neural implicit surfaces with encoding level of detail. Siggraph Asia 2023 (2023)