Robustifying the Multi-Scale Representation of Neural Radiance Fields

Nishant Jain[1]
njain@cs.iitr.ac.in
Suryansh Kumar[2†]
sukumar@vision.ee.ethz.ch
Luc Van Gool[2,3]
vangool@vision.ee.ethz.ch

[1] Indian Institute of Technology Roorkee, India
[2] ETH Zürich Switzerland
[3] KU Leuven Belgium

Abstract

Neural Radiance Fields (NeRF) recently emerged as a new paradigm for object representation from multi-view (MV) images. Yet, it cannot handle multi-scale (MS) images and camera pose estimation errors, which generally is the case with multi-view images captured from a day-to-day commodity camera. Although recently proposed Mip-NeRF could handle multi-scale imaging problems with NeRF, it cannot handle camera pose estimation error. On the other hand, the newly proposed BARF can solve the camera pose problem with NeRF but fails if the images are multi-scale in nature. This paper presents a robust multi-scale neural radiance fields representation approach to simultaneously overcome both real-world imaging issues. Our method handles multi-scale imaging effects and camera-pose estimation problems with NeRF-inspired approaches by leveraging the fundamentals of scene rigidity. To reduce unpleasant aliasing artifacts due to multi-scale images in the ray space, we leverage Mip-NeRF multi-scale representation. For joint estimation of robust camera pose, we propose graph-neural network-based multiple motion averaging in the neural volume rendering framework. We demonstrate, with examples, that for an accurate neural representation of an object from day-to-day acquired multi-view images, it is crucial to have precise camera-pose estimates. Without considering robustness measures in the camera pose estimation, modeling for multi-scale aliasing artifacts via conical frustum can be counterproductive. We present extensive experiments on the benchmark datasets to demonstrate that our approach provides better results than the recent NeRF-inspired approaches for such realistic settings.

1 Introduction

NeRF has emerged as a popular method of choice for object representation from its multi-view (MV) images [23]. This new 3D representation has shown promising results on several computer vision, graphics and robotics problems [16, 18, 21, 22, 28, 34, 35]. Yet, it has some inherent challenges in handling day-to-day captured multi-view images. For instance, NeRF shows observable artifacts on multiple scale images [6], and its performance degrades even with subtle inaccuracies in camera pose estimates [20]. Now, most of us might have

†Corresponding Author.

experienced that with everyday commodity cameras, it is challenging, if not impossible, to acquire an object's MV images at the same scale and recover correct camera poses using them simultaneously. On the other hand, popular off-the-self pose solvers such as COLMAP [27] have limitations in providing accurate camera poses [6, 15], which inherently limits the broader application of NeRF. Such limitations with NeRF were easily noticeable, leading to a few recent follow-ups addressing those limitations with NeRF, yet independently.

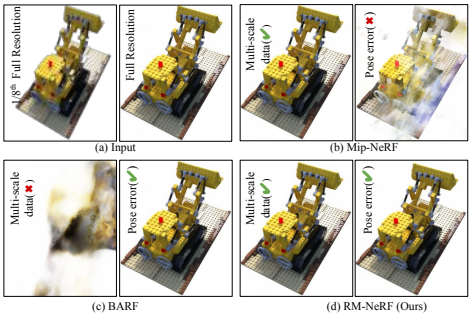
Concretely, the recently proposed BARF [20], and NeRF– [31] method can overcome the requirement of correct camera pose for NeRF, assuming that images are captured at equidistant from an object. On a separate line of research—inspired by anti-aliasing techniques in computer graphics rendering—the newly proposed Mip-NeRF [5] solves the multi-scale problem with NeRF by leveraging the mipmapping approach to rendering. Yet, it assumes ground-truth camera poses are known or estimated well via COLMAP. As is known, ground-truth pose estimation is a challenging task, and a popular framework such as COLMAP has its challenges in recovering robust camera pose from real-world images [6, 15]. In both of these groups of independent research, as mentioned above, there exists a gap, *i.e.*, BARF and similar methods can handle the camera pose problem but cannot handle multi-scale image issue, whereas, Mip-NeRF can handle the multi-scale problem but assumes the correct camera pose. In this paper, we propose a simple and effective approach that can fill this gap by utilizing the fundamentals of a rigid scene. Our method jointly addresses the multi-scale problem, and the robust camera poses estimation requirements with neural volume rendering (see Fig.1(a)). Furthermore, our approach comprehensively solves NeRF problems and eliminates third-party dependencies for pose estimation, hence a self-contained approach.

Meanwhile, we would like to show that Mip-NeRF multi-scale modeling in the ray space could break down if the camera pose error is not considered. As shown in Fig.1(b), the correct intersection of the conical frustum that can localize the object is possible only if both the camera poses are correctly known. One trivial way to handle this is to jointly optimize for object representation and camera pose as done in BARF [20] and NeRF– [31] but using Mip-NeRF [5] formulation. However, the bundle-adjustment (BA) based joint optimization is complex, sub-optimal, requires good initialization, and can handle only certain types of noise and outlier distribution [6, 25]. And therefore, conditioning the multi-scale rendering representation based on BA-type complex optimization can complicate the overall approach, hence not an encouraging take on the problem.

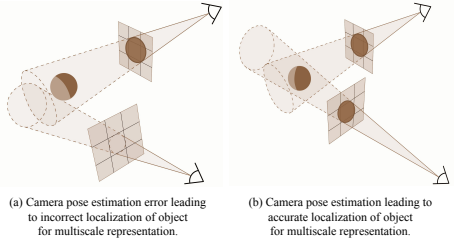
To solve the challenges mentioned above, we resort to fundamentals of scene rigidity [10]. If the scene is rigid, we can estimate the camera motion robustly without having explicit information about the object's 3D position. Accordingly, our proposed method initially disentangles the camera pose estimation from neural volume rendering to recover a good pose for joint optimization of the proposed loss function. Our approach uses graph-neural network-based multiple motion averaging with multi-scale feature modeling for the robust camera pose to solve the problem. We evaluated our method's performance on the widely used benchmark dataset [5, 23], which clearly shows superior results to the competing baseline methods. In this paper, we make the following contributions.

Contributions

- We propose a method that jointly solves camera pose and multi-scale object representation for day-to-day captured multi-view images using NeRF based representation.
- Our method uses scene rigidity fundamentals to jointly optimize camera pose and rendering loss function. To this end, our approach leverage multi-scale representation [5] and introduces graph neural network-based multiple motion averaging to learn the noisy cam-



(a) **Left**: Result Illustration



(b) **Right**: Intuition on camera pose.

Figure 1: **Left**: (a) Multi-scaled, multi-view images of Lego example with camera pose error is fed to the NeRF inspired methods [5, 20, 23]. (b) [5] can handle multi-scale imaging effect but fails if camera pose error also persists. (c) [20] can handle the camera pose error for same scale images but fails for multi-scale input images. (d) Our approach works for both cases. **Right**: A visual illustration: (a) Error in the camera pose estimation can lead to incorrect cones casting in the volume space leading to misguided localization of the object for proper modeling. (b) Correct camera pose certify the proper modeling of the object volume for each sampled canonical frustum.

era motion estimates from the images. As a result, our approach helps in better estimation of the network's model parameters for the multi-scale scene or object representation.

- The proposed method achieves better camera pose estimates and novel view rendering results than the existing NeRF-based baseline approaches when tested on the standard benchmark sequence [23] and other popular real-world sequences [17].

2 Background and Preliminaries

Recently, implicit neural representation for object or scene inspired by NeRF [23] has gained significant attention in the computer vision and graphics community with many extensions [7]. Consequently, discussing all the NeRF-related methods is beyond the scope of the paper, and readers may refer to [7, 29] for a quick reference. Nevertheless, to keep the discussion concise, we discuss the papers directly relevant to our proposed approach.

2.1 Closely Related Work

Lately, NeRF has become a popular method of choice for representing a rigid scene as a continuous volumetric field parametrized by a multi-layer perceptron (MLP) [23]. Assuming a calibrated setting with well-posed input images, NeRF for each pixel sample points along rays that are traced from the camera's center of projection. Further, these sampled points are transformed using positional encoding to represent each point in a high-dimensional feature vector before being fed to an MLP for density and color estimation for novel view synthesis.

***(i)* Multiscale NeRF.** Barron *et al.* [5] introduced Mip-NeRF to overcome the limitation with NeRF in rendering multi-resolution images, *i.e.*, multi-view images that are taken at a different distance from the object. Instead of sampling points along the rays traced from the camera center of projection, Mip-NeRF queries samples along a conical frustum interval region approximated using 3D Gaussian to render the corresponding pixel. As alluded to above, acquisition of images at a perfect scale is unrealistic using day-to-day cameras, and therefore, Mip-NeRF broadens the scope of neural volume rendering approaches to commonly

acquired multi-view and multi-scale image acquisition setup. Yet, Mip-NeRF assumption on the availability of ground-truth camera pose parameters is rather unrealistic and could substantially constrain its broader usage.

(ii) **Uncalibrated NeRF.** Not long ago, few methods have appeared to solve both for camera pose and object 3D representation extending the neural radiance fields formulation. For instance, BARF [20] leverages photometric BA to jointly register the camera poses and recover object representation. On the other hand, NeRF–[31] solves for both intrinsic and extrinsic camera calibration while training NeRF model. Nonetheless, these extensions of NeRF works well only for the same scale images; hence its usage is limited to a synthetic multi-view dome or hemispherical setup. Other related work includes iNeRF [33] that solves camera poses given a well-trained NeRF model.

2.2 Camera Pose Estimation

Widely used approaches to camera pose estimation from multi-view images are based on filtering of image key-points and incrementally solve pose [2] or use global BA [30] that generally has five-point [24] or eight-point algorithm [14] at the back-end. Yet, we know that such methods can provide sub-optimal solutions and may not robustly handle outliers inherent to the unstructured set of images. To address such an intrinsic challenge with pose estimation, Govindu [9] initiated and later authored/co-authored a series of robust multiple rotation averaging (MRA) approaches [6, 10]. The benefit of using MRA is that it uses multiple estimates of noisy relative motion to solve absolute camera pose based on view-graph representation and rotation group structure [10] *i.e.*, $SO(3)$. Contrary to the conventional robust rotation averaging approaches [1, 4, 6, 13], in this work, we adhere to recent graph neural network-based approaches [8, 19, 25, 32] for robust camera pose estimation.

3 Proposed Approach

This paper introduces an approach that unifies two independent research fields in geometric computer vision and volume rendering for scene representation. Our method exploits the multi-scale model of Mip-NeRF and composes it with graph-neural network-based robust motion averaging in a joint optimization cost function. We begin our discussion with multiscale representation for NeRF followed by robust multiple motion averaging.

(i) **Multiscale Representation for NeRF.** By leveraging pre-filtering [3] techniques in rendering *i.e.*, tracing a cone instead of ray, Mip-NeRF [5] learns the scene representation by training a single neural network, which can be queried at arbitrary scales. Further, contrary to NeRF, which uses point-based sampling along each pixel ray to form their positional encoding (PE) feature vector, Mip-NeRF uses the volume of each conical frustum along the cone to model the integrated positional encoding (IPE) features. The positional encoding $\gamma(\mathbf{x})$ (as defined in NeRF [23]) of all the point within the conical frustum is formulated as:

$$\gamma^*(\mathbf{o},\mathbf{d},\dot{r},t_0,t_1) = \frac{\int \gamma(\mathbf{x})\mathbf{F}(\mathbf{x},\mathbf{o},\mathbf{d},\dot{r},t_0,t_1)d\mathbf{x}}{\int \mathbf{F}(\mathbf{x},\mathbf{o},\mathbf{d},\dot{r},t_0,t_1)d\mathbf{x}} \tag{1}$$

where $\mathbf{F}$ is an indicator function regarding whether a point lies inside the frustum in the given range $[t_0, t_1]$. Nevertheless, Eq.(1) is computationally intractable with no closed form solu-

tion and therefore, it is approximated using multivariate Gaussian which provides "integrated positional encoding" (IPE) feature[5][1].

***(ii)* Scene Rigidity and Multiple Motion Averaging.** Assume a pin-hole camera model with known intrinsic calibration matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$ with $\mathbf{R} \in SO(3), \mathbf{t} \in \mathbb{R}^{3\times1}$ as the rotation and translation w.r.t reference frame. We can relate i^{th} image pixel $x = [u_i, v_i, 1]^T$ to its corresponding 3D point $\mathbf{x} = [x_i, y_i, z_i]^T$ as follows:

$$[u_i, v_i, 1]^T = \mathbf{K}[\mathbf{R} \mid \mathbf{t}]\ [x_i, y_i, z_i, 1]^T \tag{2}$$

Eq.(2) indicate a non-linear interaction between 3D scene point and camera motion. Yet, the classical epipolar geometry model suggests that if the scene is rigid $x'^T \mathbf{E}\, x = 0$ must hold [12], where x' is the image correspondence of x in the next image frame. It is well-studied that $\mathbf{E}$ can be decomposed into $\mathbf{R}$ and $\mathbf{t}$ such that $\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$, where $\mathbf{E} \in \mathbb{R}^{3\times3}$ is the essential matrix and $[\mathbf{t}]_\times \in \mathbb{R}^{3\times3}$ is the skew-symmetric matrix representation of the translation vector [12]. And therefore, we can estimate rigid motion without making use of any actual 3D observation. Nonetheless, rigid motion solution based on epipolar algebraic relation is not robust to outliers and may provide unreliable results with more multi-view images [6]. So to estimate robust camera motion independent of 3D scene point in a computationally efficient way led to the success of robust motion averaging approaches in geometric computer vision [1, 6, 10]. Further, given rotations, solving translations generally becomes a linear problem [6]. Consequently, solution to motion averaging reduces to rotation averaging problem.

3.1 Formulation, Loss Function and Optimization

Let $\mathcal{I}$ be the set of multi-view images taken at different distances from the object (see top left: Fig.2). We aim to simultaneously update the MLP parameterized multi-scale representation network (θ) and set of camera poses $\mathcal{P}$, given initial set of noisy estimated pose $\tilde{\mathcal{P}}$. In probabilistic term, we can formulate it as

$$\theta, \mathcal{P} \sim \Phi(\theta, \mathcal{P} | \mathcal{I}, \tilde{\mathcal{P}}) \tag{3}$$

The above formulation can further be simplified based on the assumption that scene is rigid, we can optimize for the camera pose without *explicit* notion of 3d object points in the scene space. So, we simplified the Eq.(3) as follows:

$$\Phi(\theta, \mathcal{P} | \mathcal{I}, \tilde{\mathcal{P}}) = \overbrace{\Phi(\theta | \mathcal{I}, \mathcal{P})}^{\text{Multiscale MLP}} \cdot \overbrace{\Phi(\mathcal{P} | \mathcal{I}, \tilde{\mathcal{P}})}^{\text{Motion averaging}} \tag{4}$$

Graph Neural Networks for MRA. Assume a directed view-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ (see Fig.2 bottom left). A vertex $\mathcal{V}_j \in \mathcal{V}$ in this view graph corresponds to j^{th} camera absolute rotation R_j and $\mathcal{E}_{ij} \in \mathcal{E}$ corresponds to the relative orientation $\tilde{R}_{ij}$ between view i and j (in Fig.2 represented in quaternions). For our problem, relative orientations which can be noisy are used for initializing the graph. We aim to recover accurate absolute pose R_j and jointly model the object representation. Conventionally, in the presence of noise, the camera motion is obtained by solving the following optimization problem to satisfy compatibility criteria.

$$\underset{\{R_j\}}{\operatorname{argmin}} \sum_{\mathcal{E}_{ij} \in \mathcal{E}} \rho\left(d(\tilde{R}_{ij}, R_j R_i^{-1})\right) \tag{5}$$

[1] For more details and derivations, kindly refer [5]

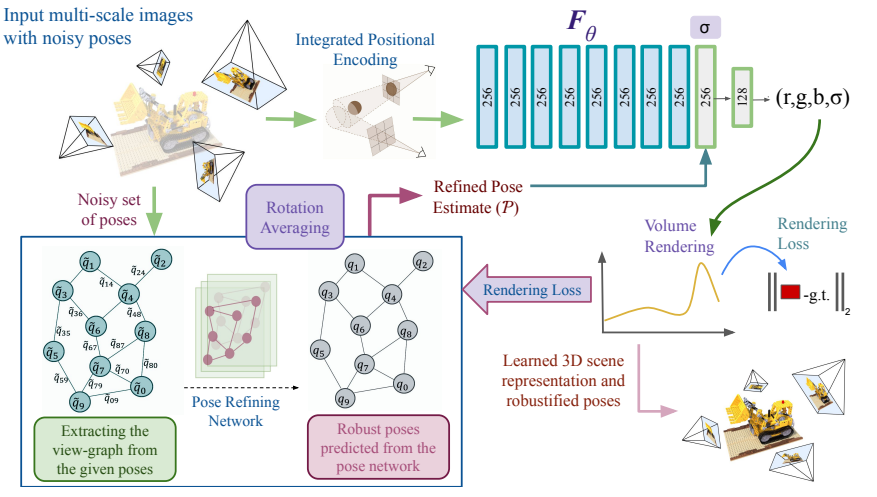


Figure 2: Our method jointly solve poses and learn the multi-scale object representation. The input consists of multi-scale image set and noisy set of pose. The pipeline consists of a pose-refining network to recover robust pose and estimate the IPE (Integrated Positional Encoding) by casting well-posed conical frustums through the pixels. Later, those are fed to the MLP network for learning the object 3D representation. $\mathcal{P}$ denotes set of pose.

where, $d(.)$ denotes a suitable metric on $SO(3)$ and $\rho(.)$ is the robust loss function defined over that metric. Minimizing this cost function $\rho(.)$ in Eq.(5) using conventional method may not be apt for several types of noise distribution observed in the real-world multi-view images. Therefore, we adhere to learn the noise distribution from the input data at train time and infer the noisy pattern to robustly predict absolute rotation. We pre-train graph neural network in a supervised setting to learn the mapping f that takes noisy relative rotation $\tilde{R}_{ij}$ and predict absolute rotations *i.e.*, $\{R_j^f\} := f(\tilde{R_{ij}};\Theta)$, where Θ is the network parameters. We train to minimize the discrepancy between ground-truth relative rotation $R_{ij} = R_j R_i^{-1}$ and estimated relative rotations $R_{ij}^f = R_j^f R_i^{-1}$ and add an extra regularizer to further learn one-to-one absolute rotation mapping.

$$\underset{\Theta}{\operatorname{argmin}} \sum_{\mathcal{G}\in\mathcal{D}} \sum_{\mathcal{E}_{ij}\in\mathcal{E}} d(R_{ij}^f, R_{ij}) + \beta \sum_{\mathcal{V}_j\in\mathcal{V}} d(R_j^f, R_j) \tag{6}$$

We fix the reference rotation to be $\mathbf{I}_{3\times3}$ identity matrix. The mapping f can now be optimized accurately [25][2]. Thus, our overall loss solves for accurate poses and object representation via a joint cost function. Concretely, we combine Eq.(6) ($\mathcal{L}_{mra}$) with the squared error between the true $C(\mathbf{r})$ and predicted $\hat{C}(\mathbf{r})$ pixel colors ($\mathcal{L}_{rgb}$).

$$\mathcal{L} = \overbrace{\sum_{\mathbf{r}\in\mathcal{R}} \|C(\mathbf{r}) - \hat{C}(\mathbf{r})\|_2^2}^{\mathcal{L}_{rgb}} + \overbrace{\sum_{\mathcal{E}_{ij}\in\mathcal{E}} d_Q(q_{ij}^f, q_{ij}) + \beta \sum_{\mathcal{V}_j\in\mathcal{V}} d_Q(q_j^f, q_j)}^{\mathcal{L}_{mra}} \tag{7}$$

where, $d_Q = \min\{||\mathbf{p}-\mathbf{q}||_2, ||\mathbf{p}+\mathbf{q}||_2\}$ measures distance between two quaternion say $\mathbf{p},\mathbf{q}$. Here, β is a scalar constant. q_{ij}'s symbolizes corresponding quaternion representation of the rotation matrix defined in Eq.(6). $\mathcal{V}$ denotes the vertex set of the view graph corresponding to the scene being optimized and $\mathcal{E}$ denotes the corresponding edge set.

[2]For more implementation details and view-graph initialization refer supplementary.

3.1.1 Joint Optimization of Pose and Multi-Scale Image Rendering

Denoting the parameters of the MLP rendering network as θ and the parameters of pose network as Θ, the complete optimization objective is to search for parameters θ and Θ jointly such that loss $\mathcal{L}$ defined in Eq.(7) is minimized. Using gradient based optimization for this search process requires calculating $\nabla_\theta \mathcal{L}$ and $\nabla_\Theta \mathcal{L}$. As $\mathcal{L}_{mra}$ is independent of rendering network, we have $\nabla_\theta \mathcal{L} = \nabla_\theta \mathcal{L}_{rgb}$. This appears to be similar as previous optimization landscape for the rendering network, but here the poses would be changing continuously resulting in different numeric value of the gradient, making the optimization difficult to converge. Now, for the pose network $\nabla_\Theta \mathcal{L}$ will have 2 terms: $\nabla_\Theta \mathcal{L}_{rgb}$ and $\nabla_\Theta \mathcal{L}_{mra}$. The second term is easy to handle given the pose network is able to solve the rotations as shown in [25]. The first term is something that would entangle the search process for θ and Θ. For better understanding, let's assume the loss due to predicted color as $\Phi(\theta, \gamma(\mathcal{P}))$, where $\mathcal{P}$ (with slight abuse of notation) denotes the poses having rotations predicted by the pose network, γ denotes the positional encoding[26], then the gradient of $\Phi(\theta, \gamma(\mathcal{P}))$ w.r.t the pose network parameters Θ can be computed using backpropagation as:

$$\nabla_\Theta \mathcal{L}_{rgb} = \frac{\partial \Phi(\theta, \gamma(\mathcal{P}))}{\partial \Theta} = \frac{\partial \Phi(\theta, \gamma(\mathcal{P}))}{\partial \gamma(\mathcal{P})} \frac{\partial \gamma(\mathcal{P})}{\partial \mathcal{P}} \frac{\partial \mathcal{P}}{\partial \Theta} \tag{8}$$

Differentiating this γ function might result in updates favourable to higher frequencies (k) as pointed out previously in [20], therefore we modify this γ function further to:

$$\gamma^*(x,k) = e^{g(k)} \gamma(x) \tag{9}$$

where $g(k) = \min(\frac{t-k}{b}, 0)$, t is annealed from 0 to maximum number of modes and b is a scalar constant. The term $\nabla_\Theta \mathcal{L}_{rgb}$ shown in Eq.(8) results in correlated updates on MLP network and pose network parameters and can result in a highly non-convex optimization. To make optimization stable, we use the following weighted loss function:

$$\mathcal{L} = \lambda \mathcal{L}_{mra} + (1-\lambda)\mathcal{L}_{rgb} \tag{10}$$

where λ is a scalar constant. Fig.(2) provides the overall flow-diagram of our approach.

Optimization Strategy. We begin with a disjoint optimization scheme for poses and structure, fixing $\lambda = 1$ fixed for some initial number of epochs. For this case, Eq.(4) depicts the modified formulation of the problem statement. After the initial optimization of both the networks via biased weighting strategy, λ is annealed by using an exponential decay, *i.e.*, $\lambda = \lambda_0 e^{-kt}$ where $\lambda_0 = 1$. This annealing goes till $\lambda = 0.5$ and then we fix it at 0.5 for the remaining optimization process.

4 Experimental Setup, Results and Ablations

Our approach requires optimization of two networks *(i)* Graph Neural Network (GNN) for robust rotation averaging optimization based on message passing strategy [8, 25], *(ii)* Multi-layer Perceptron (MLP) network optimization for neural multi-scale scene representation. Our GNN architecture for pose optimization is inspired from Purkait *et al.* [25] FineNet, whereas the MLP based rendering network is similar to Mip-NeRF [5]. Implementation details for reproducibility and hyperparameter values are provided in the supplementary. Also, pre-training scheme of our GNN, extensive evaluation on pose graphs and robustness to noisy correspondences are provided in the supplementary.

	Lego		Ship		Drums		Mic		Chair		Ficus		Materials		Hotdog	
	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓
Mip-NeRF	21.52	0.06	24.54	0.07	13.34	0.075	24.71	0.05	29.1	0.049	22.47	0.055	19.7	0.089	27.09	0.053
BARF	10.88	0.55	8.81	0.74	11.56	0.76	12.35	0.57	14.35	0.47	11.88	0.65	12.28	0.61	14.28	0.46
Base A	11.67	0.49	14.28	0.28	13.25	0.67	12.28	0.41	15.12	0.20	12.31	0.25	13.31	0.42	16.17	0.39
Base B	12.46	0.37	13.43	0.31	11.32	0.58	14.26	0.29	13.71	0.42	11.56	0.52	12.22	0.47	15.87	0.42
NeRF–	16.89	0.094	19.89	0.118	15.67	0.074	18.35	0.08	20.22	0.098	14.44	0.13	15.77	0.22	18.69	0.20
Base C	18.28	0.089	16.32	0.22	17.25	0.070	19.42	0.073	18.67	0.114	16.32	0.12	16.58	0.207	17.55	0.223
Ours	**27.01**	**0.044**	**26.59**	**0.067**	**26.07**	**0.043**	**32.8**	**0.008**	**35.23**	**0.031**	**29.28**	**0.032**	**24.8**	**0.061**	**32.5**	**0.028**

Table 1: Performance comparison with other competing approaches. We used widely used PSNR and LPIPS performance metric to document the results. Clearly, our method supersede the results of BARF[20], Mip-NeRF[5] and NeRF–[31] on the multi-scale blender synthetic dataset proposed by [5]. The details of other baselines are given in Sec. §4. For this experiment, we synthetically perturbed the poses. The results suggest that our approach can favorably deal with multi-scale and camera pose problem.

Baselines. We compared our method's performance with the existing methods that either resolves multi-scale or pose problems with NeRF [23] such as BARF[20], NeRF–[31], and Mip-NeRF[5]. To further show the effectiveness of our method, we define new baselines. Our newly defined baseline A (Base A) combines BARF and Mip-NeRF loss. This is done by making the poses input to the Mip-NeRF trainable and updating the positional encoding scheme similar to BARF. Next, we define baseline B (Base B) where we first run BARF on the multi-scale scene and then train the Mip-NeRF model with the output poses. Finally, we define baseline C (Base C) where we combine Mip-NeRF and NeRF– by just updating NeRF– with the Mip-NeRF positional encoding scheme. Refer supplementary for further reasoning behind using these baselines.

4.1 Test sets and Results

To compare our method with the baselines, we used the Blender dataset provided by the authors of NeRF [23][3] and its multi-scale version provided by Barron *et al.* [5]. It consists of single object scenes comprising synthetic objects and corresponding ground truth poses, with each scene consisting of $M = 100$ images with 800×800 resolution. To simulate the effect of pose estimation errors in real-world datasets, we add noise to the ground truth poses. More details regarding datasets are provided in the following subsections. Furthermore, we also test our approach on real-world dataset[17], where COLMAP poses are treated as G.T., and show improvements over these roughly accurate poses in the supplementary.

Multi-Scaled Images of Object. We first study the multi-scaled version of the Blender dataset proposed in the Mip-NeRF. It consists of 400 image scenes generated by scaling every image in the original Blender dataset to 4 different resolutions. These different resolution images are synthesized by concatenation of actual resolution images with downsampled images by a factor of 2, 4, and 8. This scale can also be interpreted as the distance of the object from the camera. Therefore, it resembles the real-world datasets much more closely than the original Blender dataset, which contains all the images at the same image resolution and nearly similar distances. The ground truth extrinsic poses are the same as the original dataset, but the camera intrinsics are changed according to the image resolution. We perturb the poses for every scene by first sampling the noise from a normal distribution $\delta\mathbf{p} \sim \mathcal{N}(\mathbf{0}, 1e^{-1}\mathbf{I})$, and adding the noise to rotation in its axis-angle form. It is then converted to the rotation matrix representation and multiplied with the ground truth poses, disturbing their orientations. We did this purposely to make the dataset resemble real-world settings closely, thus making it challenging to learn the multi-scale scene representation.

[3] CC-BY-3.0 license.

	MipNeRF			Ours		
	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑
Truck	23.1	0.308	0.812	**24.7**	**0.296**	**0.828**
Tank	24.8	0.313	0.823	**26.6**	**0.302**	**0.851**

Table 2: Performance comparison of our approach and Mip-NeRF[5] on the Truck and Tank sequence [17]. The result shows that our method performs better as compared to Mip-NeRF on both the freely moving sequences demonstrating our method's advantage. It can be inferred from the above statistics that just relying on COLMAP poses for solving image based rendering on unconstrained sequence can demonstrably give inferior results. On the contrary, our approach can handle bad camera poses and provide favorable novel view rendering.

Table (1) provide the results with multi-scale images and pose error as input. The results are compared using the popular PSNR and LPIPS metric averaged across all the four resolution images. The results show that our method can jointly solve the multi-scale and pose problems with NeRF and gives results better than other baseline approaches. Fig.(3) provide the qualitative result comparison for the same.

Tanks and Temples. Tanks and Temples is a well-known challenging dataset containing real-world scenes [9]. It consists of images showing large scale scenes to simulating the realistic conditions and largely used for evaluating 3D reconstruction methods. Further, this dataset can be very useful for testing unconstrained view-synthesis methods. Accordingly, we used couple of sequence to test our method's performance. Specifically, we used "truck" and "tank" sequence, which consists of image set containing a 360° view of the subject captured freely at a varying distance from the object. Since there are no ground-truth poses provided by the dataset, we used COLMAP[27] to estimate the initial poses and feed them to our network. Table (2) shows the comparison between our method and Mip-NeRF[5] for these two sequence. Clearly, our method efficiently optimizes over the poses esimated by the COLMAP and is able to generate better image renderings when compared Mip-NeRF+COLMAP setting. For more details regarding initial camera estimation, evaluation details on this dataset and experimental observations, pleae refer supplementary.

Randomly Captured Black-Box Sequence. In Fig.(3), we also introduced a new sequence, containing randomly captured images of a black box, imitating a general purpose multi-image acquisition *i.e.*, multi-scale and with non-smooth pose trajectory. Again, we use COLMAP to estimate the initial poses. From the results, it can be concluded that our approach clearly outperforms all the existing methods in this realistic scenario due to its robust pipeline jointly estimating scene and structure. Refer supplementary for more details regarding the images and the COLMAP estimated cameras for this sequence.

4.2 Ablations

(a) Same Scale Images with Pose Error. Similar to the multi-scale case setup as described in §Sec.4.1, we perturb the pose estimates of the Blender dataset, which contains object multi-view images captured from the same distance. Table (3) shows the PSNR, LPIPS, and SSIM results for this setting corresponding to four scenes in the dataset namely *Lego*, *ship*, *drums* and *mic*. The performance confirms that our method more often than not supersedes the baselines results with similar scale images. This is expected as BARF [20] was designed to handle pose for images taken at same distant from the object.

(b) Unbiased Optimization of Eq. (10) ($\lambda = 0.5$). To better understand the behaviour of our joint optimization and utility of our annealing strategy, we conducted this experiment. By setting $\lambda = 0.5$ in Eq.(10), we assign equal weight to color rendering cost ($\mathcal{L}_{rgb}$) and robust MRA loss ($\mathcal{L}_{mra}$) loss during optimization. Table (4) shows a comparison of this

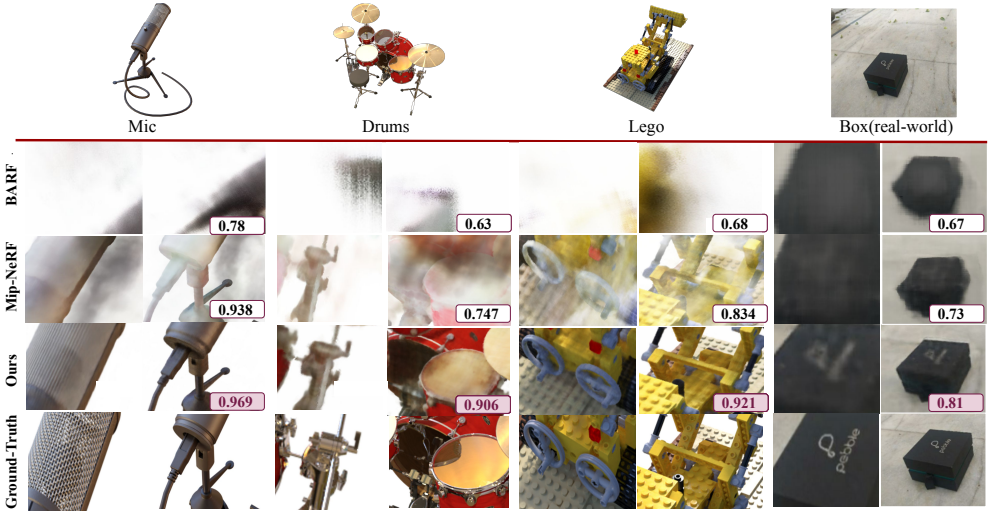


Figure 3: Qualitative Comparison of our method, Mip-NeRF[5] and BARF[20] on the multi-scale of blender dataset with synthetically added pose errors. We have visualized the synthesized images from all the approaches on 3 scenes namely Mic, Drums, Lego and our real sequence. Our method clearly provide better synthesized images.

	Lego			Ship			Drums			Mic		
	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM
Mip-NeRF	17.90	0.089	0.82	22.90	0.107	0.71	14.07	0.11	0.799	21.90	0.064	0.93
BARF	27.61	0.05	0.92	26.18	0.121	0.74	23.68	0.095	0.88	27.03	0.06	0.96
RM-NeRF(ours)	27.10	0.048	0.92	25.45	0.0690	0.735	24.98	0.072	0.907	30.03	0.027	0.963

Table 3: PSNR, LPIPS and SSIM comparison of our method with BARF[20] and Mip-NeRF[5] on Blender dataset[2] with synthetically introduced pose errors.

	Lego			Ship			Drums			Mic		
	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM
RM-NeRF(ours†)	22.20	0.067	0.87	23.34	0.071	0.71	15.07	0.079	0.789	22.60	0.049	0.92
RM-NeRF(ours)	27.01	0.044	0.92	26.59	0.067	0.74	26.07	0.043	0.92	32.8	0.008	0.97

Table 4: Comparison of our proposed optimization method (ours) with its variant (ours†) where we fix $\lambda = 0.5$ on Mulit-scale Blender dataset with introduced pose errors.

method with our optimization method on the 4 scenes of the multi-scale blender dataset with synthetic noise. The statistics indicate that utilizing the rigid scene prior during optimization helps. Furthermore, in supplementary, we perform another ablation where we compare all the methods on the Multi-scale Blender dataset using the available ground truth poses.

5 Conclusion and Future Direction

We introduced an approach that enhances the use of neural radiance fields representation to general daily acquired multi-view images, where multi-scale images and camera pose errors are inevitable. By unifying the concepts from multi-view geometry in computer vision, multi-scale NeRF, and graph neural networks, we propose a method that can robustly solve multi-scale image rendering issues in continuous volume rendering. Of course, the proposed method is not a perfect solution to the problem; however, it suggests an important area for research that could enable continuous neural volume rendering to daily acquired multi-view images. A straightforward future direction is to extend the proposed approach for jointly estimating the camera intrinsics and the extrinsic for multi-scale neural scene representation.

References

[1] Khurrum Aftab, Richard Hartley, and Jochen Trumpf. Generalized weiszfeld algorithms for lq optimization. *IEEE transactions on pattern analysis and machine intelligence*, 37(4):728–745, 2014.

[2] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

[3] John Amanatides. Ray tracing with cones. *ACM SIGGRAPH Computer Graphics*, 18 (3):129–135, 1984.

[4] Federica Arrigoni, Beatrice Rossi, Pasqualina Fragneto, and Andrea Fusiello. Robust synchronization in so (3) and se (3) via low-rank and sparse matrix decomposition. *Computer Vision and Image Understanding*, 174:95–113, 2018.

[5] 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*, pages 5855–5864, 2021.

[6] Avishek Chatterjee and Venu Madhav Govindu. Robust relative rotation averaging. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):958–972, 2017.

[7] Frank Dellaert and Lin Yen-Chen. Neural volume rendering: Nerf and beyond. *arXiv preprint arXiv:2101.05204*, 2020.

[8] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[9] Venu Madhav Govindu. Combining two-view constraints for motion estimation. In *CVPR*, volume 2. IEEE, 2001.

[10] Venu Madhav Govindu. Robustness in motion averaging. In *Asian Conference on Computer Vision*, pages 457–466. Springer, 2006.

[11] Venu Madhav Govindu. Motion averaging in 3d reconstruction problems. In *Riemannian computing in computer vision*, pages 145–164. Springer, 2016.

[12] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[13] Richard Hartley, Khurrum Aftab, and Jochen Trumpf. L1 rotation averaging using the weiszfeld algorithm. In *CVPR 2011*, pages 3041–3048. IEEE, 2011.

[14] Richard I Hartley. In defense of the eight-point algorithm. *IEEE Transactions on pattern analysis and machine intelligence*, 19(6):580–593, 1997.

[15] Rajbir Kataria, Joseph DeGol, and Derek Hoiem. Improving structure from motion with reliable resectioning. In *2020 International Conference on 3D Vision (3DV)*, pages 41–50. IEEE, 2020.

[16] Berk Kaya, Suryansh Kumar, Francesco Sarno, Vittorio Ferrari, and Luc Van Gool. Neural radiance fields approach to deep multi-view photometric stereo. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1965–1977, 2022.

[17] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36 (4), 2017.

[18] Soomin Lee, Le Chen, Jiahao Wang, Alexander Liniger, Suryansh Kumar, and Fisher Yu. Uncertainty guided policy for active robotic 3d reconstruction using neural radiance fields. *arXiv preprint arXiv:2209.08409*, 2022.

[19] Xinyi Li and Haibin Ling. Pogo-net: Pose graph optimization with graph neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5895–5905, 2021.

[20] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5741–5751, 2021.

[21] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020.

[22] Julien NP Martel, David B Lindell, Connor Z Lin, Eric R Chan, Marco Monteiro, and Gordon Wetzstein. Acorn: adaptive coordinate networks for neural scene representation. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021.

[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. *Communications of the ACM*, 65(1):99–106, 2021.

[24] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE transactions on pattern analysis and machine intelligence*, 26(6):756–770, 2004.

[25] Pulak Purkait, Tat-Jun Chin, and Ian Reid. Neurora: Neural robust rotation averaging. In *European Conference on Computer Vision*, pages 137–154. Springer, 2020.

[26] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019.

[27] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

[28] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. imap: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6229–6238, 2021.

[29] Ayush Tewari et al. Advances in neural rendering. In *Computer Graphics Forum*, volume 41, pages 703–735. Wiley Online Library, 2022.

[30] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, ICCV '99, pages 298–372, London, UK, UK, 2000. Springer-Verlag. ISBN 3-540-67973-1.

[31] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021.

[32] Luwei Yang, Heng Li, Jamal Ahmed Rahim, Zhaopeng Cui, and Ping Tan. End-to-end rotation averaging with multi-source propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11774–11783, June 2021.

[33] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1323–1330. IEEE, 2021.

[34] 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*, pages 4578–4587, 2021.

[35] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021.