PriNeRF

Prior constrained Neural Radiance Field for robust novel view synthesis of urban scenes with fewer views

Chen, Kaiqiang; Dong, Bo; Wang, Zhirui; Cheng, Peirui; Yan, Menglong; Sun, Xian; Weinmann, Michael; Weinmann, Martin

DOI
10.1016/j.isprsjprs.2024.07.015

Publication date
2024

Document Version
Final published version

Published in
ISPRS Journal of Photogrammetry and Remote Sensing

ISPRS Journal of Photogrammetry and Remote Sensing 215 (2024) 383–399

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

PriNeRF: Prior constrained Neural Radiance Field for robust novel view synthesis of urban scenes with fewer views

Kaiqiang Chen [a,b], Bo Dong [a,b,c,d], Zhirui Wang [a,b,*], Peirui Cheng [a,b], Menglong Yan [b,e,f], Xian Sun [a,b,c,d], Michael Weinmann [g], Martin Weinmann [h]

[a] Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, 100190, China
[b] University of Chinese Academy of Sciences, Beijing, 100190, China
[c] School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing, 100190, China
[d] Key Laboratory of Network Information System Technology (NIST), Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, 100190, China
[e] Jigang Defence Technology Company, Ltd., Jinan, 250132, China
[f] Cyber Intelligent Technology (Shandong) Co., Ltd, Jinan, 250132, China
[g] Intelligent Systems Department, Delft University of Technology, Delft, The Netherlands
[h] Institute of Photogrammetry and Remote Sensing, Karlsruhe Institute of Technology, Karlsruhe, Germany

ARTICLE INFO

Keywords:
Novel view synthesis
Neural radiance fields
Urban scenes
Multi-view urban priors
Camera parameter optimization

ABSTRACT

Novel view synthesis (NVS) of urban scenes enables the exploration of cities virtually and interactively, which can further be used for urban planning, navigation, digital tourism, etc. However, many current NVS methods require a large amount of images from known views as input and are sensitive to intrinsic and extrinsic camera parameters. In this paper, we propose a new unified framework for NVS of urban scenes with fewer required views via the integration of scene priors and the joint optimization of camera parameters under an geometric constraint along with NeRF weights. The integration of scene priors makes full use of the priors from the neighbor reference views to reduce the number of required known views. The joint optimization can correct the errors in camera parameters, which are usually derived from algorithms like Structure-from-Motion (SfM), and then further improves the quality of the generated novel views. Experiments show that our method achieves about 25.375 dB and 25.512 dB in average in terms of peak signal-to-noise (PSNR) on synthetic and real data, respectively. It outperforms popular state-of-the-art methods (i.e., BungeeNeRF and MegaNeRF) by about 2–4 dB in PSNR. Notably, our method achieves better or competitive results than the baseline method with only one third of the known view images required for the baseline. The code and dataset are available at https://github.com/Dongber/PriNeRF.

1. Introduction

Novel View Synthesis (NVS) refers to the process of generating new and unseen views of a scene or object that were not captured during the initial data acquisition (Debevec et al., 1998; Kang, 1998; Cooke et al., 2006; Bach et al., 2022; Fülöp-Balogh et al., 2022). When applied to urban scenes, NVS allows users to explore the city from different viewpoints, enhancing user experience in industries such as property marketing, navigation and wayfinding, gamified entertainment, cultural relics protection and more (Meshry et al., 2019; Li et al., 2023; Rematas et al., 2022; Martin-Brualla et al., 2018; Condorelli et al., 2021; Qi et al., 2009).

As the pioneer, Neural Radiance Fields (NeRF) (Mildenhall et al., 2021) have been proposed to synthesize photo-realistic novel views not contained in the set of the input images, bringing a novel solution to NVS, which has been extensively studied by researchers in various indoor (Haitz et al., 2023) and outdoor (Kniaz et al., 2023) scenes, and remote sensing (Derksen and Izzo, 2021; Marí et al., 2022; Semeraro et al., 2023). It represents a 3D scene with a Multilayer Perceptron (MLP), which is optimized based on the supervision solely from 2D images. The MLP takes as input the coordinates of in-scene 3D points and a certain viewing direction vector, and outputs the color and density of the points. Then, Volume Rendering (Kajiya and Von Herzen, 1984) is applied to synthesize the pixels required for the 2D image of the new view. The solutions of the NeRF models (Mildenhall et al., 2021; Barron et al., 2021) lie in a very high-dimensional space, while the inputs and outputs are only low-dimensional vectors. Furthermore,

* Corresponding author at: Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, 100190, China.
E-mail address: zhirui1990@126.com (Z. Wang).

https://doi.org/10.1016/j.isprsjprs.2024.07.015
Received 23 August 2023; Received in revised form 16 July 2024; Accepted 17 July 2024
Available online 29 July 2024

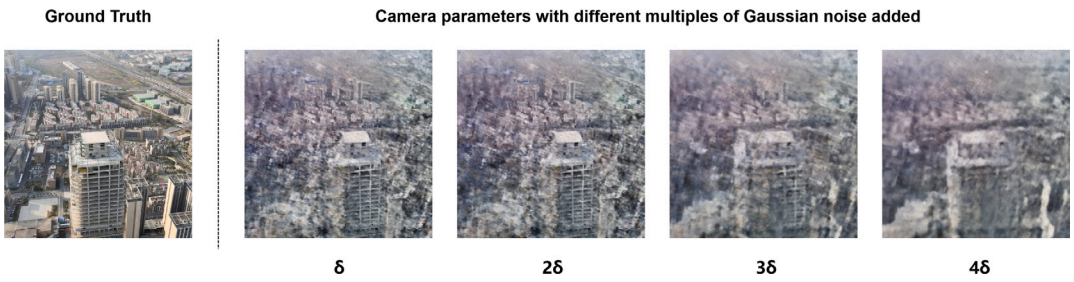


Fig. 1. The impact of noise in camera parameters. We add Gaussian noises to camera parameters in increments of δ, where the mean equals to 0, and the variance equals to 10, 0.1 and 0.05 for intrinsic parameters in $\boldsymbol{K}$, rotation matrix $\boldsymbol{R}$ and translation vector $\boldsymbol{t}$, respectively. It reveals that noise leads to inferior results of synthesized views and demonstrates the importance of accurate camera parameters.

the solution spaces are highly free since the views are regarded as independent of each other, which means that two image pixels corresponding to a specific scene position are not constrained by photometric consistency. Therefore, the optimization of such NeRF models in their basic form (Mildenhall et al., 2021; Turki et al., 2022) requires a large number of views as the supervision to achieve decent results. For instance, the vanilla NeRF (Mildenhall et al., 2021) uses 100 views for the reconstruction of small toys like Lego, and MegaNeRF (Turki et al., 2022) uses 2k–5k views to reconstruct a complex scene of 0.1 km^2.

To reduce the number of views required for NVS, PixelNeRF (Yu et al., 2021) has been proposed to extract view features with a pre-trained ResNet (He et al., 2016) as prior constraints to NeRF. It achieves NVS with only one or few images on the popular ShapeNet (Chang et al., 2015) and DTU (Jensen et al., 2014) benchmarks. However, PixelNeRF only focuses on models such as chairs or cars in controlled laboratory environments and has not been generalized to urban scenes. Alternatively, some NeRF methods proposed for urban-scale scenes (Turki et al., 2022; Tancik et al., 2022; Xiangli et al., 2022; Rematas et al., 2022) still depend heavily on a large amount of views. If the PixelNeRF model is naively applied to a large scene, due to the downsampling process of ResNet, the feature map extracted by the network is inconsistent in size with the input view, failing to learn a high-dimensional feature for each pixel, resulting in pixel locations within a small area sharing the same feature. Correspondingly, the 3D points of a spatial cluster share the same features, which is obviously inconsistent with the complex situation of large scenes. Therefore, we argue that the coarse features derived from interpolation cannot accurately represent the 2D features of 3D points, resulting in inferior NVS performance. In our paper, we adopt a new structure to extend the idea of a prior constraints for multiple views to urban scenes. And experiments demonstrate the importance of setting an accurate and more refined per-pixel representation for NVS.

In addition, accurate camera parameters (including intrinsics and extrinsics) are essential for the NeRF. Most NeRF methods use Structure from Motion (SfM) (Schönberger and Frahm, 2016) to estimate camera parameters for 2D images and fix them during the training process of NeRF models (Mildenhall et al., 2021; Turki et al., 2022; Tancik et al., 2022; Xiangli et al., 2022; Yu et al., 2021). SfM also relies on plenty of views, but it is impossible to obtain dense views from any angle as small-sized items when collecting views of a large-scale scene. Therefore, the limited view of a large-scale scene will cause errors in SfM. The error accumulates during the forward propagation process, causing the network to learn an inaccurate 3D-2D mapping relationship. The impact of different levels of error is shown in Fig. 1, revealing the importance of accurate camera parameters.

Some studies (Wang et al., 2021b; Yen-Chen et al., 2021; Lin et al., 2021; Chen et al., 2022c) have suggested treating camera parameters as learnable variables and incorporating them into NeRF MLP training. However, these studies only utilize the photometric loss of NeRF to calculate the gradient for camera parameters, without using spatial geometric constraints. Jeong et al. (2021) improves upon this by introducing additional geometric consistency constraints. Nevertheless, they are only confined to small objects and have not been applied to large urban scenes. In this work, we address the challenges in NVS involving sparser views, camera pose refinement and large scenes by solving them within a unified, end-to-end trainable network. Subsequent experiments revealed the effectiveness of our algorithm qualitatively and quantitatively.

In general, we propose a new unified framework to address the challenges of training NeRF for large-scale scenes with a fewer required view number and a joint camera parameter refinement. Specifically, we build the implementation upon BungeeNeRF (Xiangli et al., 2022), which achieves descent NVS performance on large urban scenes. However, BungeeNeRF requires dense views. Furthermore, the experiments are on Google Earth Studio with ideal camera parameters, which is impossible in real applications. We additionally integrate scene priors and jointly optimize camera parameters under spatial geometric constraints along with NeRF MLP weights. Firstly, a feature network is employed to extract features from the adjacent reference views of the target view. the sampled points are projected along rays onto these feature maps, obtaining 2D features of points from different perspectives as scene priors. The prior features, along with the coordinates of the sampled points and viewing directions, are fed into the NeRF pipeline. Then, we introduce a learnable camera module, in which the intrinsic and extrinsic camera parameters are initialized based on SfM, which are jointly optimized with the NeRF MLP weights under the photometric supervision and geometric constraint.

The main contributions of this paper can be summarized as follows:

- We propose a unified framework involving the prior and geometric constraints that jointly optimize the NeRF model and camera parameters. Our method achieves competitive or better results than the baseline with only one third of the original number of views on both real and synthetic scenes.
- We introduce prior constraints of multiple views to the NVS for urban scenes, and extract finer and accurate image features of the 3D points as the scene priors. Further experiments reveal that our method can generate superior novel views with fewer known views.
- We propose a new camera parameter optimization method building upon the view-prior NeRF framework with introducing spatial geometric constraints. It effectively addresses the issue of imperfect estimation of camera parameters for large-scale scenes with sparse views. Both qualitative and quantitative experiments demonstrate the benefits of the algorithm.

The rest of the paper is organized as follows. We first introduce related work in Section 2. Subsequently, we explain the overall and details of our method in Section 3. For the validation of our method, we describe the used datasets, evaluation metrics, qualitative and quantitative results and ablation studies in Section 4. Finally, the conclusions are drawn in Section 5.

2. Related work

In this section, we firstly review the popular Neural Radiance Fields (NeRF) methods for Novel View Synthesis (NVS) in Section 2.1. Then we focus on the research on leveraging scene priors and the optimization of camera parameters in Section 2.2 and Section 2.3, respectively.

2.1. Neural radiance fields

Compared with traditional 3D reconstruction (Yang et al., 2011; Huang et al., 2020), NeRF does not generate an explicit 3D model, but synthesizes new views from any angle. The success of the NeRF methods originates from the many advantages that are inherent to the underlying principle. Firstly, it achieves high-quality visual effects, i.e., NeRF can generate realistic images and capture details in the scene. Secondly, it leverages self-supervised learning, which means that NeRF learns 3D scene representations directly from only image data, without knowledge regarding depth, illumination or occlusion information.

However, the vanilla NeRF (Mildenhall et al., 2021) can produce excessively blurry close-up views and aliasing artifacts in distant view when the images used for training or testing contain multiple resolutions with varying distances. This is a result of the limited information provided by the finite sampling points. However, increasing the number of sampling points is costly for NeRF as it involves performing hundreds of additional MLP queries to render each ray. Furthermore, NeRF (Mildenhall et al., 2021) was initially developed to synthesize novel views of controlled, bounded small-scale scenes.

Some subsequent works (Barron et al., 2021; Martin-Brualla et al., 2021) attempted to overcome limitations of NeRF. Specifically, Mip-NeRF (Barron et al., 2021) introduced the use of cone sampling instead of the ray sampling in NeRF and furthermore integrated position encoding (IPE) to address the blurriness issue. NeRF-W (Martin-Brualla et al., 2021) extends NeRF to outdoor scenes, and addresses the challenges of complex outdoor lighting and transient occlusion by learning the appearance variation of each view in a low-dimensional latent space. These two works have fundamentally improved NeRF, but the model capacity prevents them from extending to large-scale urban scenes.

Recently, grid-based methods have gained popularity as a solution to the challenge that NeRF requires numerous MLP forward calculations, resulting in very low efficiency. Instant-ngp (Müller et al., 2022) introduces a hash-searching strategy for 3D mesh features and connects the NeRF pipeline to achieve rendering output. Plenoxels (Fridovich-Keil et al., 2022) and DVGO (Sun et al., 2022) directly replace the MLP with a dense voxel grid, performing volume rendering on the interpolated 3D features. However, as the scene size increases, using voxel grid representation will lead to an exponential increase in memory.

There are two ways of applying NeRF for large urban scenes. One (Tancik et al., 2022; Turki et al., 2022) is to partition the scene into multiple smaller blocks, with each block being modeled by a separate NeRF network. Another way (Xiangli et al., 2022) involves using a scalable model that gradually increases in size during training to achieve multi-scale supervision from drone to satellite perspectives, enabling the synthesis of new views for large scenes in a unified model.

However, these methods require a large amount of view data for training. In addition, the camera parameters used by these methods are computed in advance, which usually contain noise and hence cause the networks to fail to learn the 3D information of the scene correctly. To improve the synthesis quality and increase the robustness, these methods increase the number of views to ensure sufficient information for fine-grained synthesis of new views and reduce the impact of the noise in the camera parameters.

In this paper, we propose a novel pipeline for NVS in urban scenes that exploits adjacent views to obtain scene priors to reduce the number of views required. At the same time, we jointly optimize the camera parameters and the NeRF network, which effectively reduces the impact of inaccurate camera parameters. The related work on scene priors utilization and the camera parameter optimization are reviewed in Sections 2.2 and 2.3, respectively.

2.2. Scene priors utilization

When it comes to 3D reconstruction and NVS, scene priors are crucial for scene understanding and representation, especially for ill-posed scenes. For example, photometric stereo (Ju et al., 2023, 2021) uses images of objects taken from different directions of light to infer the geometric information of the object's surface. Some works used depth-guided image interpolation to synthesize new views (Zhou et al., 2016; Flynn et al., 2016; Riegler and Koltun, 2020). Other works (Xu et al., 2019; Saito et al., 2019) demonstrated that detailed synthesis can be achieved using spatially aligned local image features.

More recently, adding priors to NeRF has been explored (Yu et al., 2021; Wang et al., 2021a), but these methods study the generalization problem by making the network a universal view interpolation function, allowing for fine-tuning in new scenes without requiring extensive retraining from different viewpoints. PixelNeRF (Yu et al., 2021) was the first work to combine CNN with NeRF, utilizing ResNet (He et al., 2016) to extract scene features as prior inputs to the NeRF network. IBRNet (Wang et al., 2021a) and GARF (Shi et al., 2022) followed a similar method and introduce the Transformer component. These methods employ down-sampling feature extraction networks, resulting in lower-resolution feature maps. Fine-grained details are lost when projected onto feature maps to obtain prior features, especially at the edges or intricate parts of the image.

To mitigate the need for densely captured images, additional features are incorporated into the NeRF structure. ViTNeRF (Lin et al., 2023) uses the Transformer encoder to obtain 1D latent features to represent global information, and then combines 2D CNN features to improve the local representation. MVSNeRF (Chen et al., 2021) constructs a 3D cost volume based on the 2D features extracted from each image and is applied to the 3D CNN. The MLP is responsible for predicting the volume density and RGB from the 3D interpolated features. GeoNeRF (Johari et al., 2022) extends MVSNeRF by using cascaded cost volumes trained in a semi-supervised manner to obtain high-resolution priors for tuning the renderer.

Further methods towards dealing scenes with sparsely sampled RGB input views rely on the incorporation of priors on the structure of the scene into the optimization process, including paired view features (Chibane et al., 2021), visibility prior (Somraj and Soundararajan, 2023a), depth smoothness (Niemeyer et al., 2022), surface smoothness (Oechsle et al., 2021; Zhang et al., 2021), semantic similarity (Jain et al., 2021), Manhattan world assumptions (Guo et al., 2022), monocular geometric priors in terms of supervising inferred depth and surface normal information together with the RGB image reconstruction loss (Yu et al., 2022), or visibility priors (Somraj and Soundararajan, 2023b).

Our proposed method differs from previous works (Yu et al., 2021; Wang et al., 2021a) in the underlying motivation and technical method. Firstly, our goal is to reduce the number of views needed in large-scale scenes, rather than focusing on generalization. Secondly, we use a tiny U-shaped network to learn the upsampling process instead of using fixed interpolation to generate multi-channel feature maps that align with the input views' pixels, providing more precise features for projection. We demonstrate this in the scope of our experiments.

2.3. Camera parameter optimization

The estimation and optimization of camera parameters play a critical role for NVS that have received extensive research attention. A commonly used method is Structure-from-Motion (SfM) (Schönberger and Frahm, 2016), which can simultaneously recover the 3D structure of the scene and camera poses from sparse 2D feature correspondences matched across different images. Relevant works (Schönberger and Frahm, 2016; Chen et al., 2022b; Wilson and Snavely, 2014) involve extracting feature points from each image, performing matching, and inferring camera poses using either the five-point algorithm (Nistér,

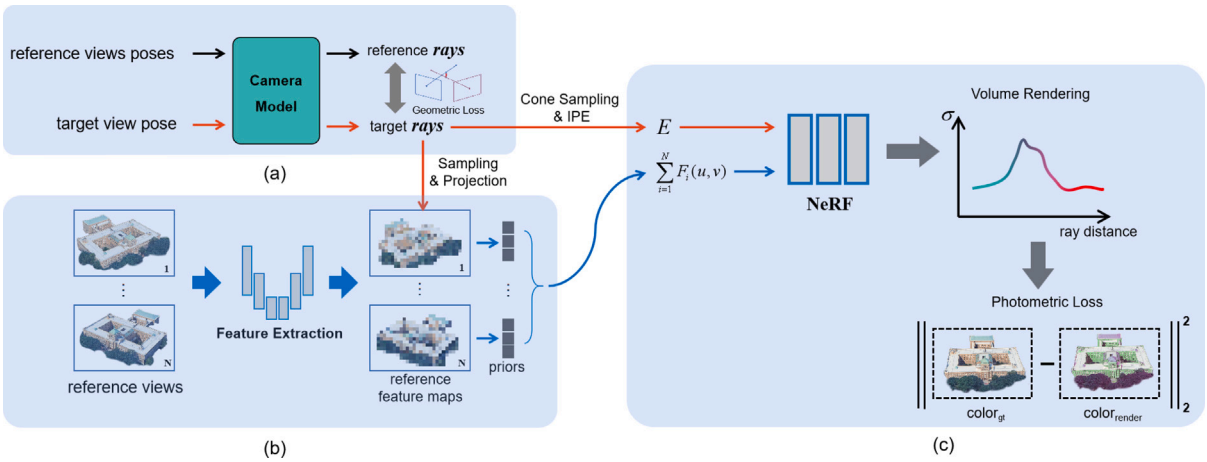


Fig. 2. Method Overview. Our method involves three parts. (a) The camera model is used to generate rays from the target and reference views and to calculate the distance between corresponding rays from different views. (b) The feature extraction module extracts the deep features of the reference view image and aggregates the features under different reference views as priors. (c) The NeRF takes as input the scene priors along with position and orientation encodings to generate color and opacity, which are rendered into a novel view. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2004) or the eight-point algorithm (Hartley, 1997). Subsequently, Bundle Adjustment (BA) is employed to refine the camera poses. There are also deep learning-based methods (Kendall et al., 2015; Sundermeyer et al., 2018) that attempt to directly recover camera poses from 2D images.

Meanwhile, camera pose optimization is also extensively studied in field of Simultaneous Localization and Mapping (SLAM). iMap (Sucar et al., 2021) and NICE-SLAM (Zhu et al., 2022) have demonstrated the utilization of SLAM for camera pose estimation, followed by scene representation using NeRF. However, their methods relies on RGB-D input data. Several works (Rosinol et al., 2022; Chung et al., 2022; Zhu et al., 2023; Maggio et al., 2022) have showed robust performance with only RGB input. Nonetheless, these SLAM-based methods excel primarily within indoor settings and fall short when applied to expansive outdoor environments.

As NeRF gains popularity, researchers are increasingly exploring possibilities of reconstructing scenes without explicit knowledge about camera poses. NeRF– (Wang et al., 2021b) is the first to attempt reconstruction for unknown camera parameters by jointly optimizing the NeRF model and camera parameters of input images based on the minimization of photometric reconstruction errors. iNeRF (Yen-Chen et al., 2021) applied a pre-trained NeRF model to render images under known intrinsic camera parameters and continuously optimized the extrinsic parameters by minimizing photometric errors with respect to the real images. These methods do not consider exploiting 3D constraints, but crudely incorporate camera parameters into the NeRF architecture. Subsequently, some works (Lin et al., 2021; Chng et al., 2022; Chen et al., 2022a) established a theoretical connection between classic bundle adjustment and NeRF by optimizing the camera parameters through minimizing reprojection errors. Chen et al. (2022c) introduced epipolar constraints from SfM using an additional CNN network to learn the camera parameters. However, the above-mentioned and some other methods (Truong et al., 2022; Heo et al., 2023) optimize each camera parameter independently without considering the interdependencies among them (such as the orthogonality of the rotation matrix in the extrinsic parameters). As a result, the obtained parameters are aimed at achieving optimal network performance, disregarding their inherent mathematical or physical meaning.

Our proposed method differs from previous work (Wang et al., 2021b; Yen-Chen et al., 2021; Lin et al., 2021; Chen et al., 2022c) in that we focus on maintaining the orthogonality constraint of the camera extrinsic rotation matrix. In addition, we introduce the geometric loss to our framework to help better optimize the camera parameters and improve the synthesis effect for new views.

3. Methodology

We start this section by providing an overview of our proposed method (Section 3.1) and then provide details on the extraction of scene priors (Section 3.2), the optimization of camera parameters (Section 3.3) and the training process (Section 3.4). The pipeline of our method is presented in Fig. 2. The extraction of scene priors enables the NeRF to synthesize new views with fewer known images, and the joint optimization of camera parameters further improves the performance of NVS.

3.1. Overview

The framework of our method consists of three parts, a camera model, a feature extraction module and a NeRF, as presented in Fig. 2.

The camera model plays an essential role in the optimization of camera parameters via the minimization of the distance between a pair of rays through corresponding points in the reference and target view images. The respective camera calibration procedures firstly derive the corresponding points using descriptors like SIFT (Lowe, 1999, 2004) in the reference and target view images. A pair of corresponding points represents a same spatial point in a scene, and the corresponding rays should intersect in an ideal and perfect condition, which rarely happens in a real situation due to the noise in the camera parameters and measurement. In this regard, we minimize the distance between the corresponding rays to consistently update the camera parameters.

The feature extraction module extracts the deep features for the reference view images and then derives scene priors for the target view. We extract deep feature maps of the reference views with a tiny U-shaped network, which is jointly optimized with NeRF. Then the sampling points in a ray of the target view are projected onto the feature maps of the reference views to obtain the corresponding features of the 3D points. The features of the 3D points are aggregated and then fed into the NeRF model as scene priors.

We use the scalable BungeeNeRF (Xiangli et al., 2022) with scene priors as additional input. The training of our NeRF model is similar to other classical NeRF methods (Mildenhall et al., 2021; Barron et al., 2021), including sampling along rays and 3D position encoding of the sampling points. We use the cone sampling and integrated position encoding (IPE) strategy proposed by Barron et al. (2021).

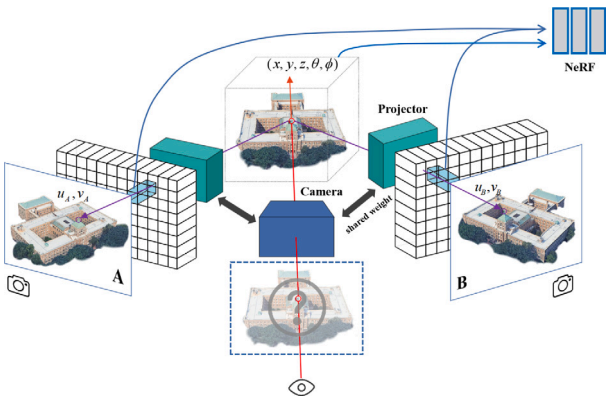


Fig. 3. Scene priors. We project the sampling points on the ray emitted from the target view into the reference views, and then obtain the features in the reference views by interpolation, which are fed into the NeRF as scene priors along with the coordinate and view direction codes of sampling points.

3.2. Scene priors

In this section, we introduce a projection method for obtaining view-dependent scene priors. Our method employs a brief and lightweight U-shaped network to extract features from the reference view. Then, the spatial sampling points of the target view are projected onto the feature map of the nearest reference views, and then we obtain pixel-aligned scene prior information via interpolation. These prior features describe the local geometric and photometric characteristics of the scene. By using a feature map of the same size as the original view, our method learns a feature vector representation for each pixel coordinate of the reference view, which can provide finer and more accurate information during interpolation.

3.2.1. Reference view selection

To obtain valid reference views, we select the views closest to the target view. Specifically, we calculate the angles between the target view and the reference views. Let $\boldsymbol{R}_i \in \mathbb{R}^{3\times3}$ denotes the camera rotation matrix of the ith known view. The angle (in degrees) between the views i and j is then computed as

$$\theta_{ij} = \frac{\arccos(\frac{tr(\boldsymbol{R}_i \boldsymbol{R}_j^{-1})-1}{2}) \cdot 180}{\pi} \tag{1}$$

where tr represents the trace of the matrix.

We then obtain a candidate set of reference views by choosing those that fall within a certain angular threshold. From this set, we select the top N views with the smallest angular difference, which are most similar to the target view, to form the input set for the feature extraction network. The choice of the N depends on the specific scene and the view sparsity.

The feature extraction network is a lightweight U-shaped structure that generates feature maps of the same size as the input views. We represent each view in a working set using its feature map $\boldsymbol{F}_i \in [0,1]^{H_i\times W_i\times 8}$ and camera projection matrix $\boldsymbol{P}_i \in \mathbb{R}^{3\times4}$ for the ith reference view, where $\boldsymbol{P}$ is composed of a rotation matrix $\boldsymbol{R} \in \mathbb{R}^{3\times3}$ and a translation vector $\boldsymbol{t} \in \mathbb{R}^{3\times1}$. We input the set of tuples $(\boldsymbol{F}_i, \boldsymbol{P}_i)_{i=1}^N$ into the projector, a component responsible for projecting points in 3D space onto the feature map $\boldsymbol{F}_i$.

3.2.2. Feature mapping

Given a collection of feature maps of the reference views $(\boldsymbol{F}_i, \boldsymbol{P}_i)_{i=1}^N$, we project a sampling point $\boldsymbol{X}$ on a ray onto the feature maps of each reference view using the projector, which shares intrinsic and extrinsic parameters with the camera model. This derives the pixel coordinates (u, v) of $\boldsymbol{X}$ on the image plane at each view, which can be expressed as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{K} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{X} \\ 1 \end{bmatrix} \tag{2}$$

where $\boldsymbol{K}$ refers to the intrinsic parameters of the camera, $\boldsymbol{R}$ refers to the rotation matrix in the camera's extrinsic parameters, and $\boldsymbol{t}$ refers to the translation vector.

Then bilinear interpolation is performed according to pixel coordinate to obtain an aggregated representation of the 3D points in different reference views as PixelNeRF (Yu et al., 2021). It is then fed into the NeRF network along with the encoded ray origin and direction vectors as shown in Fig. 3.

3.3. Camera model and pose optimization

In this section, we introduce a learnable rotation orthogonality invariant camera model based on the pinhole camera. In 3D rendering, the camera model maps a 3D point $\boldsymbol{X}$ in space to a 2D coordinate $\boldsymbol{p}$ in the image plane, where $\boldsymbol{p} = F(\boldsymbol{X})$. However, the main focus of NeRF is the inverse of this process, using the camera model to generate rays from the imaging plane into the 3D scene and infer the respectively observed pixel color values. Each ray is represented by a pair of 3D vectors $\boldsymbol{r}_o$ and $\boldsymbol{r}_d$, representing the origin and direction of the ray, respectively.

3.3.1. Camera model

Our camera model is derived from the pinhole camera model, which consists of intrinsic parameters $\boldsymbol{K}$ and extrinsic parameters $\boldsymbol{E} = [\boldsymbol{R}|\boldsymbol{t}]$. The camera model can establish the mapping relationship between the 3D coordinates in space and the 2D coordinates in the image plane.

Our method relies on an initial coarse camera pose, which can be obtained through SfM methods. We use COLMAP (Schönberger and Frahm, 2016) for the initialization of camera parameters. The intrinsic parameters in $\boldsymbol{K}$ include four unknown variables related to the construction of the camera. f_x, f_y represent the focal length of the camera along the x-axis and y-axis, and define the magnification or zoom factor in the horizontal and vertical directions. c_x, c_y are the coordinates of the principal point, i.e. the point where the optical axis intersects the image plane. They represent the offset from the origin of the image plane to the principal point along the x, y axes. Optimizing $\boldsymbol{K}$ involves optimizing these four variables. In our method, instead of optimizing them directly, we optimize a residual value based on the initial value $f_x = f_{x_0} + \Delta f_x$. The other three variables f_y, c_x, c_y are manipulated in the same way. We also constrain the norm of $\Delta\boldsymbol{c} = (\Delta c_x, \Delta c_y)$ and $\Delta\boldsymbol{f} = (\Delta f_x, \Delta f_y)$ to be bounded to be two percent and one percent of the focal length f, respectively. Similarly, we optimize the translation vector $\boldsymbol{t}$ in the extrinsic parameters using the same way: $\boldsymbol{t} = \boldsymbol{t}_0 + \Delta\boldsymbol{t}$, where $\boldsymbol{t}_0$ is the initial value, and $\Delta\boldsymbol{t} = (\Delta t_1, \Delta t_2, \Delta t_3)$ is the residual term of the optimization.

$$\boldsymbol{K} = \begin{bmatrix} f_{x_0} + \Delta f_x & 0 & c_{x_0} + \Delta c_x \\ 0 & f_{y_0} + \Delta f_y & c_{y_0} + \Delta c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$\boldsymbol{t} = \boldsymbol{t}_0 + \Delta\boldsymbol{t} = \begin{bmatrix} t_1 + \Delta t_1 \\ t_2 + \Delta t_2 \\ t_3 + \Delta t_3 \end{bmatrix} \tag{4}$$

For the rotation matrix $\boldsymbol{R}$ in the extrinsic parameters, which belongs to a special orthogonal group $\mathrm{SO}(3) = \{\boldsymbol{R} \in \mathbb{R}^{3\times3} | \boldsymbol{R}\cdot\boldsymbol{R}^T = \boldsymbol{I}, \det(\boldsymbol{R}) = 1\}$ Optimizing each single element in the matrix independently destroys the inherent properties of the matrix. Therefore, we use Rodrigues formula in rigid body motion to optimize the rotation matrix. The formula can convert the rotation represented by the rotation matrix into the axis angle representation. Specifically, using a normalized unit vector $\boldsymbol{\phi} = \begin{bmatrix} \phi_x & \phi_y & \phi_z \end{bmatrix}^T$ represents the axis of rotation, and using

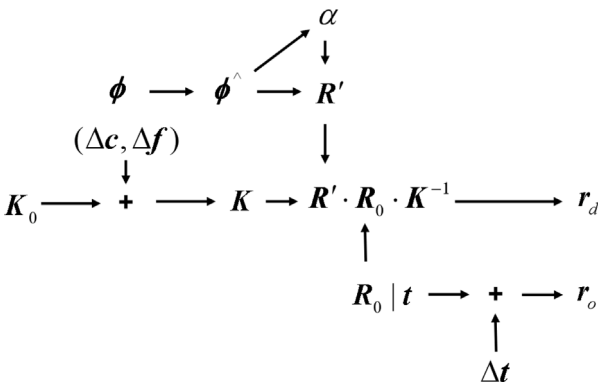


Fig. 4. Computation graph of the ray parameters. We optimize the intrinsic parameters and translation vectors by optimizing the residual term. The orthogonal rotation matrix is generated by two unit vectors as the optimization term, which are optimized jointly with the NeRF weights. The final outputs are the origin $\boldsymbol{r}_d$ and direction $\boldsymbol{r}_o$ of the rays.

an angle α represents the magnitude of the rotation. The mathematical expression of Rodrigues formula is

$$\boldsymbol{\Delta R} = \boldsymbol{I} + \frac{\sin(\alpha)}{\alpha}\boldsymbol{S}_{\boldsymbol{\phi}} + \frac{1-\cos(\alpha)}{\alpha^2}(\boldsymbol{S}_{\boldsymbol{\phi}})^2, \tag{5}$$

where $\boldsymbol{S}_{\boldsymbol{\phi}}$ denotes the skew-symmetric matrix of $\boldsymbol{\phi}$, and α can be calculated the second norm of $\boldsymbol{S}_{\boldsymbol{\phi}}$.

$$\boldsymbol{S}_{\phi} = \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix} \tag{6}$$

The optimization of the rotation matrix is obtained by multiplying the $\boldsymbol{\Delta R}$ with the initial rotation matrix R_0, ensuring optimization in the SO(3) space. Consequently, by training learnable parameters $\boldsymbol{\phi}_i$ and $\boldsymbol{\Delta t}_i$, the extrinsics for each input image I_i are optimized.

The next step involves mapping a pixel $\boldsymbol{p}$ in the image plane to the ray parameters $(\boldsymbol{r}_o, \boldsymbol{r}_d)$ in world coordinates using the optimized intrinsic and extrinsic parameters, which can be formulated according to

$$\begin{aligned} \boldsymbol{r}_d &= \boldsymbol{R}\boldsymbol{K}^{-1}\boldsymbol{p} \\ \boldsymbol{r}_o &= \boldsymbol{t} \end{aligned} \tag{7}$$

Since the ray parameters $(\boldsymbol{r}_d, \boldsymbol{r}_o)$ are functions of the optimized variables $(\boldsymbol{\Delta f}, \boldsymbol{\Delta c}, \boldsymbol{\phi}, \boldsymbol{\Delta t})$ of the intrinsic and extrinsic camera parameters, they are optimized using gradient descent. It is worth mentioning that we do not directly optimize $\boldsymbol{K}_0$, $\boldsymbol{R}_0$, or t_0. The computation graph for generating ray origin and ray direction using the camera model is summarized in Fig. 4.

3.3.2. Pose optimization

To jointly optimize the camera parameters, we introduce the widely-used reprojection error as the geometric constraint via applying the visual consistency of observations at the same spatial location in multiple views, as conducted by Jeong et al. (2021). Assume that the pixel coordinates of a spatial point in the target view $\boldsymbol{I}^A$ and the reference view $\boldsymbol{I}^B$ are $\boldsymbol{p}^A$ and $\boldsymbol{p}^B$ respectively. Let the rays emitted from pixels $\boldsymbol{p}^A$, $\boldsymbol{p}^B$ be $\boldsymbol{r}^A$ and $\boldsymbol{r}^B$. Ideally these two rays should intersect, but in practice there will be deviations. So geometric loss can be constructed based on this. Let a point on $\boldsymbol{r}^A$ be $\boldsymbol{X}^A = \boldsymbol{r}_o^A + t^A\boldsymbol{r}_d^A$ and a point on $\boldsymbol{r}^B$ be $\boldsymbol{X}^B = \boldsymbol{r}_o^B + t^B\boldsymbol{r}_d^B$, where t^A and t^B represent the distances from the 3D point to the principal point.

The distance d of $\boldsymbol{X}^A$ and $\boldsymbol{X}^B$ in space is not directly used as the optimization term since the corresponding points far away from the camera have large deviation values, while the corresponding points closer to the camera have small deviation values. So in order to normalize the scale of distances, we project point $\boldsymbol{X}^A$ onto image plane $\boldsymbol{I}^B$ and $\boldsymbol{X}^B$ onto $\boldsymbol{I}^A$ to compute the reprojection error.

$$d_\pi = \frac{\left\|\pi_A(\boldsymbol{X}^B) - p^A\right\| + \left\|\pi_B(\boldsymbol{X}^A) - p^B\right\|}{2} \tag{8}$$

Finally, the loss function is constructed by minimizing the sum of the reprojection errors for each target-reference views pair.

$$\mathcal{L}_G = \sum_{i=0}^{N} d_\pi^i \tag{9}$$

We present the computation process of the geometric loss in Fig. 5.

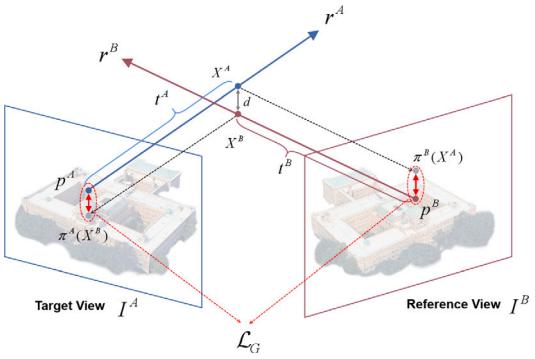


Fig. 5. Illustration of the geometric loss. It measures the reprojection error of 3D points in the target view and the reference view.

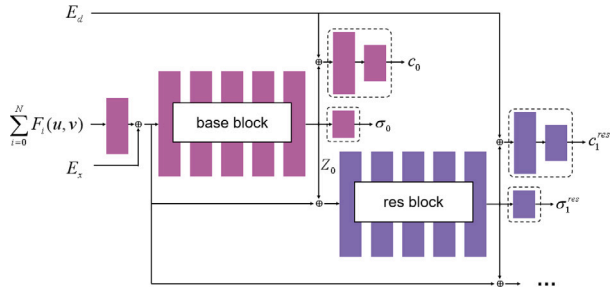


Fig. 6. The scalable NeRF structure. It consists of a base block and several res blocks. The position encoding results are injected into each residual block via skip connections. The outputs of each block are the opacity and color information at different scales. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.4. Neural radiance fields training

In this section, we offer details of the model structure (including the model structure, the sampling and encoding strategy of rays) in Section 3.4.1 and optimization objectives in Section 3.4.2.

3.4.1. Model structure and sampling, encoding strategy

We use the BungeeNeRF (Xiangli et al., 2022) as the baseline. The training data and model grow in a synchronized multi-stage manner, building a scalable structure by continuously adding residual blocks as shown in Fig. 6. The output of each block is used to predict the color and opacity residuals of the scene content viewed at progressively closer scales, corresponding to different levels of detail. Positional encodings are injected into each block via skip connections, which helps to maintain the integrity of information and the efficient transfer of gradients.

For the sampling and encoding strategy, we adopt cone sampling and integrated positional encoding (IDE) proposed by Barron et al. (2021). Cone Sampling extends NeRF's point-based sampling input to

Algorithm 1 The joint optimization of photometric and geometric consistency.

Input: multi-view images $\{\boldsymbol{I}_i\}_{i=1}^{N}$, corresponding camera poses $\{\boldsymbol{P}_i\}_{i=1}^{N}$, system initialization parameters $\mathcal{S}$
Parameter: number of sampling points N, max frequency for positional encoding M, angle threshold of the reference view θ

for iter=1,2,… **do**
  $\{(\boldsymbol{I}_j, \boldsymbol{P}_j)_{j=1}^{K}\}^{N}$ = GetRefViews $(\{\boldsymbol{I}_i\}_{i=1}^{N}, \{\boldsymbol{P}_i\}_{i=1}^{N}, \theta)$
  $\boldsymbol{r}_d, \boldsymbol{r}_o, t$ = CameraModel$(\{\boldsymbol{P}_i\}_{i=1}^{N})$
  $\boldsymbol{F}_p$ = GetScenePrior$(\boldsymbol{r}_d, \boldsymbol{r}_o, t, \{(\boldsymbol{I}_j, \boldsymbol{P}_j)_{j=1}^{K}\}^{N})$
  $\boldsymbol{E}_x, \boldsymbol{E}_d$ = Encoder$(\boldsymbol{r}_d, \boldsymbol{r}_o, t)$
  $\boldsymbol{C}, \sigma$ = ScalableNeRF$(\boldsymbol{E}_x, \boldsymbol{E}_d, \boldsymbol{F}_p)$
  $\{\boldsymbol{I'}_i\}_{i=1}^{N}$ = VolumetricRendering$(\boldsymbol{C}, \sigma)$
  $\mathcal{L}_P$ = PhotometricLoss$(\{\boldsymbol{I}_i\}_{i=1}^{N}, \{\boldsymbol{I'}_i\}_{i=1}^{N})$
  C = SIFT$(\boldsymbol{I}, \boldsymbol{I'})$ ▷ Get correspondence
  $\mathcal{L}_G$ = GeometricLoss$(\{\boldsymbol{I}_i\}_{i=1}^{N}, \{(\boldsymbol{I}_j, \boldsymbol{P}_j)_{j=1}^{K}\}^{N})$
  $\mathcal{L} = \mathcal{L}_P + \mathcal{L}_G$
  $\mathcal{S} \leftarrow \mathcal{S} + \nabla_S \mathcal{L}$
end for

Output: $\mathcal{S}$

volume frustums, reducing the blurring and aliasing artifacts that NeRF produces when rendering at different resolutions (Barron et al., 2021). Cone sampling uses a multivariate Gaussian to approximate a frustum. For each interval $T_k = [t_k, t_{k+1})$ along the ray, the frustum is represented by its mean and covariance $(\mu, \boldsymbol{\Sigma}) = \boldsymbol{r}(T_k)$, and further converted to Fourier features using IPE:

$$\gamma(\mu, \boldsymbol{\Sigma}) = \left\{ \begin{bmatrix} \sin(2^m \mu) \exp\left(-2^{2m-1} \operatorname{diag}(\boldsymbol{\Sigma})\right) \\ \cos(2^m \mu) \exp\left(-2^{2m-1} \operatorname{diag}(\boldsymbol{\Sigma})\right) \end{bmatrix} \right\}_{m=0}^{M-1} \tag{10}$$

The encoding results are fed into NeRF together with the prior features obtained in Section 3.2.2 to predict view-dependent colors and opacities.

3.4.2. Optimization of the model

The supervision of our method comes from the geometric consistency presented in Section 3.3.2 and photometric consistency.

Photometric consistency considers the color difference between the rendered view and the real view. The rendered view can be obtained using the color and opacity of the NeRF output, rendered by classical volume rendering (Kajiya and Von Herzen, 1984). In our scalable NeRF architecture, the output of the base-block is

$$\left(\boldsymbol{c}_0, \boldsymbol{\sigma}_0, \boldsymbol{Z}_0\right) = f_0\left(\boldsymbol{E}_{\boldsymbol{x}}, \boldsymbol{E}_{\boldsymbol{d}}\right) \tag{11}$$

the output of each residual block is

$$\left(\boldsymbol{c}_L^{\mathrm{res}}, \sigma_L^{\mathrm{res}}, \boldsymbol{Z}_{\mathrm{L}}\right) = f_L^{\mathrm{res}}\left(\boldsymbol{Z}_{L-1}, \boldsymbol{E}_{\boldsymbol{x}}, \boldsymbol{E}_{\boldsymbol{d}}\right) \tag{12}$$

where $\boldsymbol{E}_{\boldsymbol{x}}$ and $\boldsymbol{E}_{\boldsymbol{d}}$ are the encoding results of the spatial position and viewing direction; $\boldsymbol{c}_L$ and $\boldsymbol{\sigma}_L$ are the predicted color and opacity; $\boldsymbol{Z}_L$ is the feature that each block passes to the next block.

The aggregated output of the Lth layer of the model is

$$\boldsymbol{c}_{\boldsymbol{L}} = \boldsymbol{c}_0 + \sum_{l=2}^{L} \boldsymbol{c}_l^{\mathrm{res}}, \quad \sigma_L = \sigma_0 + \sum_{l=2}^{L} \sigma_l^{\mathrm{res}} \tag{13}$$

We approximate volume rendering integrals using numerical quadrature:

$$\boldsymbol{C}(\boldsymbol{r}) = \sum_k T_k \left(1 - \exp\left(-\sigma_k \left(t_{k+1} - t_k\right)\right)\right) \boldsymbol{c}_k$$
$$T_k = \exp\left(-\sum_{k'<k} \sigma_{k'} \left(t_{k'+1} - t_{k'}\right)\right) \tag{14}$$

where $\boldsymbol{C}(\boldsymbol{r})$ is the predicted color of the pixel, and t represents the distance from sampling points on the ray to the starting point of the ray. The final photometric consistency loss consists of the total squared error between the RGB values of rendering results and ground truth at each hierarchical scale from the nearest scale to the farthest:

$$\mathcal{L}_P = \sum_{l=1}^{L} \left(\left\|C_l(\boldsymbol{r}) - \hat{C}(\boldsymbol{r})\right\|\right) \tag{15}$$

where $\hat{C}(\boldsymbol{r})$, $C_l(\boldsymbol{r})$ is the ground truth and RGB results rendered at different levels. The design of this multi-level supervision uses a level of details to unify different levels of details into a single model, with deeper output heads providing more complex details in the rendered view.

For the MLP, we utilize photometric loss to calculate the gradient. For the camera model, we use geometric loss and photometric loss to calculate the gradient with a weight of 6:4 (both loss terms are scaled to the same order of magnitude). In the initial phase, the camera model and NeRF MLP weights are iteratively optimized. Specifically, the camera model weights are frozen while the MLP weights undergo the optimization, and vice versa. the camera parameters are updated once for every 10 iterations of the MLP parameters, which last for 5000 iterations. Subsequently, the update frequency is increased, with the camera parameters being updated once every 5 iterations of the MLP parameters for XX epochs. Finally, both the MLP model weights and the camera model weights are unfrozen and iterated simultaneously for the last 5000 iterations The final learning algorithm is shown in Algorithm 1.

4. Experiments and analysis

Our experiments are validated on two datasets, using three evaluation metrics to compare the performance of our method with that of other methods. We introduce the used datasets in Section 4.1, the evaluation metrics in Section 4.2, the experimental results in Section 4.3, and the ablation analysis in Section 4.4.

4.1. Data

We validate our method using synthetic data (Xiangli et al., 2022) and our own collected data. In addition, to demonstrate that our method can reduce the number of required views for NVS, we select various percentages of images for training by uniformly sampling from the initial dataset.

4.1.1. Synthetic scenes data from google earth studio

We use same synthetic scene data from Google Earth Studio as (Xiangli et al., 2022). By capturing multi-scale city images with specified camera positions, the data quality is sufficient to simulate real-world challenges. We tested our model on the two scenes and compared it with other methods. The information about the captured scenes is listed in Table 1. We note that only one third of the images per scene is used for training.

4.1.2. Real scenes data collected by drones

In order to further demonstrate the performance of our method, we conduct extensive experiments on real scenes. Due to the lack of publicly available data for NVS of urban scenes, we create a multi-view dataset of real scenes collected by drones or from the Internet, on which we train the model and evaluate the performance. The dataset includes four buildings with multiple views, each taken at different heights. The preliminary camera poses are obtained using COLMAP (Schönberger and Frahm, 2016). The details of the real scenes are given in Table 2.

Table 1
Details of the synthetic scene data (Xiangli et al., 2022).

City scene	Number of images					Height (m)
	total	stage1	stage2	stage3	stage4	
56Leonard, New York	463	188	313	393	463	290–3,389
Transamerica, San Francisco	455	130	260	390	455	326–2,962

Table 2
Details of the real scene data we collected via drones or from the Internet.

City Scene	Building height (m)	Flight altitude (m)
Aerospace Information Museum, Jinan	21	25–30
Yellow River Tower, Binzhou	55.6	10–80
Meixihu Arts Center, Changsha	46.8	60–80
Greenland Xindu Mall, Hefei	188	100–200

Table 3
Quantitative results of averaged metrics across multiple scales for 56Leonard and Transamerica scenes in peak signal-to-noise ratio (PSNR) in dB, structural similarity (SSIM), and perceptual similarity (LPIPS). ↑ indicates the higher the better and ↓ is vice versa.

	56Leonard (Avg.)			Transamerica (Avg.)		
	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
NeRF (Mildenhall et al., 2021)	21.107	0.335	0.611	21.420	0.344	0.625
Mip-NeRF (Barron et al., 2021)	21.642	0.299	0.695	21.820	0.331	0.687
BungeeNeRF (Xiangli et al., 2022)	23.058	0.245	0.736	23.232	0.232	0.721
MegaNeRF (Turki et al., 2022)	22.425	0.372	0.680	22.546	0.283	0.707
PriNeRF (Ours)	**25.591**	**0.147**	**0.871**	**25.158**	**0.168**	**0.835**

Table 4
Quantitative comparison with other methods on 56Leonard and Transamerica scenes at different scales, in peak signal-to-noise ratio (PSNR) in dB as the evaluation metric.

	56Leonard (PSNR ↑)				Transamerica (PSNR ↑)			
	Scale I.	Scale II.	Scale III.	Scale IV.	Scale I.	Scale II.	Scale III.	Scale IV.
NeRF (Mildenhall et al., 2021)	21.209	20.972	21.373	20.872	21.123	21.565	21.733	21.261
Mip-NeRF (Barron et al., 2021)	21.879	21.927	21.720	21.040	22.142	22.016	21.939	21.182
BungeeNeRF (Xiangli et al., 2022)	22.972	23.364	23.719	22.177	23.539	23.026	23.769	22.595
MegaNeRF (Turki et al., 2022)	22.414	22.338	22.540	22.272	22.389	22.332	22.597	22.579
PriNeRF (Ours)	**26.247**	**25.277**	**25.583**	**25.257**	**25.481**	**24.932**	**25.207**	**25.010**

Table 5
Quantitative comparison with other methods on four real-world scenes in peak signal-to-noise ratio (PSNR) in dB, structural similarity (SSIM), and perceptual similarity (LPIPS). ↑ indicates the higher the better and ↓ is vice versa.

	Aerospace Information Museum (avg)			Yellow River Tower (avg)			Meixihu Arts Center (avg)			Greenland Xindu Mall (avg)		
	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
NeRF (Mildenhall et al., 2021)	21.390	0.259	0.666	21.577	0.223	0.603	22.580	0.193	0.701	20.976	0.279	0.523
Mip-NeRF (Barron et al., 2021)	22.257	0.202	0.696	21.624	0.241	0.650	22.518	0.199	0.710	22.976	0.183	0.714
BungeeNeRF (Xiangli et al., 2022)	22.955	0.185	0.716	23.525	**0.147**	0.774	23.465	0.167	0.742	24.119	0.161	0.814
MegaNeRF (Turki et al., 2022)	22.557	0.212	0.706	22.456	0.209	0.724	23.065	0.187	0.727	23.119	0.181	0.754
PriNeRF (Ours)	**25.618**	**0.142**	**0.850**	**26.132**	0.158	**0.835**	**24.522**	**0.153**	**0.774**	**25.775**	**0.127**	**0.879**

4.2. Evaluation metrics

We use three metrics to evaluate the performance of methods for NVS task. including peak signal-to-noise ratio (PSNR), structural similarity (SSIM) (Sitzmann et al., 2019), and perceptual similarity (LPIPS) (Zhang et al., 2018) using a pre-trained VGG (Simonyan and Zisserman, 2014) encoder. LPIPS is more consistent with human perceptual situations than traditional metrics. A lower value of LPIPS means that the two images are more similar. Conversely, the greater the difference, the larger the deviation of the compared images. Considering the great discrepancy across different scales on the synthetic dataset, averaging the results under all scales does not fairly reflect the rendering quality. Therefore, we additionally report the average performance metrics at each scale for the synthetic dataset. For the real dataset, since the images are all collected by drones and the scale variation is not so large as that of the synthetic dataset, we only report the average metrics.

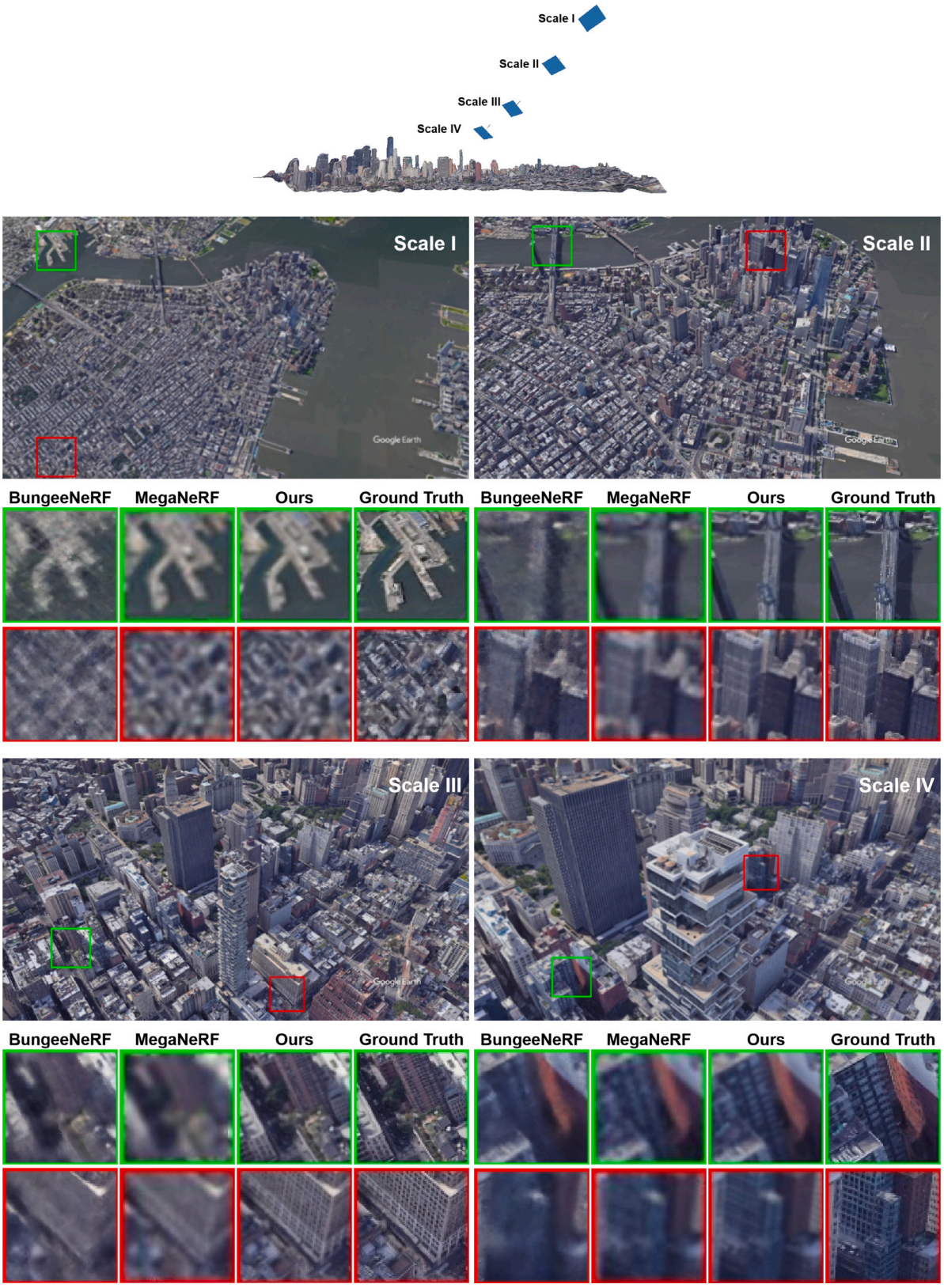


Fig. 7. Qualitative results on 56Leonard. Our method consistently outperforms the baseline method in robustly capturing the finer details at all scales.

Table 6
The results of camera parameters optimization on the 56Leonard and Transamerica scenes are compared with other methods, using rotation radian and Euclidean distance as evaluation metrics (R represents Radian, ED represents Euclidean Distance).

	56 Leonard (Avg.)				Transamerica (Avg.)			
	Rotation (R)	Translation (ED)	Focal Length (ED)	Principal Points (ED)	Rotation (R)	Translation (ED)	Focal Length (ED)	Principal Points (ED)
NeRF– (Wang et al., 2021b)	0.259	0.120	16.198	11.045	0.214	0.107	15.425	9.992
BARF (Lin et al., 2021)	0.233	0.107	14.459	10.109	0.215	0.113	13.884	9.637
PriNeRF (Ours)	**0.195**	**0.102**	**10.752**	**9.537**	**0.191**	**0.086**	**10.343**	**8.308**

4.3. Experimental results

4.3.1. Implementation details

We gradually add residual blocks to the basic block, up to four, as the scale changes. The base-block has 5 layers with 256 hidden units in each layer except the input and output layers, and residual inputs in layers 2 and 6. The residual block is similar to base-block, but has no prior input layer. We present the network structure in Fig. 6. For fair comparison, we also implemented the baseline model with the similar configuration. For IPE, we set the highest frequency setting to $M = 10$ with 65 sample sampling points per ray. Each scene is trained until the model converges. All models are optimized using Adam (Kingma and Ba, 2014), and the learning rate decays exponentially from $5e^{-4}$.

4.3.2. Synthetic scenes

Quantitative results on synthetic scenes are presented in Table 3 and demonstrate that our method outperforms the baseline method BungeeNeRF (Xiangli et al., 2022), the vanilla NeRF (Mildenhall et al., 2021), and Mip-NeRF (Barron et al., 2021) in terms of PSNR, LPIPS and SSIM. Specifically, our method achieves 25.591 dB and 25.158 dB in PSNR on 56Leonard and Transamerica, and 25.375 dB in average. It improves by 2.533 dB and 1.926 dB over BungeeNeRF (Xiangli et al., 2022) on the two scenes. We further present the PSNR metrics for each scale in Table 4. The numerical results show that our method significantly outperforms other methods at all scales and achieves a PSNR gain of about 2–4 dB over BungeeNeRF.

The qualitative results presented in Figs. 7 and 8 reveal that our method is able to synthesize finer details for the novel views than the baseline BungeeNeRF (Xiangli et al., 2022). Especially for distant and large urban scenes when the camera is far away from the surface (scale = 1 and scale = 2 in Fig. 7 and Fig. 8), the BungeeNeRF generates noisy images while our method provides clearer and finer results. As the camera moves forward, the BungeeNeRF generates clearer and less noisy images, but our method is still superior both in general quality and details.

4.3.3. Real scenes

To further demonstrate the generality and the performance of our method on real scenes, we conduct experiments on the data captured by unmanned aerial vehicles and on data collected from the Internet. Fig. 9 shows pictures of these scenes.

Quantitative results presented in Table 5 demonstrate the consistent conclusion as on the synthetic scenes, i.e., our method outperforms other methods (Mildenhall et al., 2021; Barron et al., 2021; Xiangli et al., 2022) by a large margin in terms of PSNR and SSIM. Specifically, the PSNR of our method achieves 25.512 dB in average. It improves by 2.663 dB, 2.607 dB, 1.057 dB and 1.656 dB over BungeeNeRF (Xiangli et al., 2022) on the four real scenes, respectively.

We present the qualitative results on the real scenes in Fig. 10, which reveals that our method is able to synthesize photo-realistic novel views and applicable to real scenes. Particularly, our method dramatically outperforms the baseline BungeeNeRF (Xiangli et al., 2022) in terms of details. As can be seen from Fig. 10, BungeeNeRF generates blurry textures and smooth boundaries. Instead, our method can synthesize novel views with finer textures and sharp boundaries, which are very close to the ground truth. We attribute the superiority of our method to the joint optimization of the camera parameters and the NeRF weights as well as the usage of scene priors.

4.3.4. Camera parameter optimization

To effectively validate our proposed camera parameter optimization method, we manually add noise to the camera intrinsics and extrinsics of the synthetic scene and input them into NeRF, without utilizing multi-view priors. We adopt different evaluation metrics for different parts of camera parameters. For the intrinsics, we evaluate it by comparing the Euclidean Distance (ED) before and after optimizing the focal length and principal point coordinates. For the extrinsics, we evaluate it by computing the average difference in rotation (in Radians) and the average ED in translation. We conduct comparisons with BARF (Lin et al., 2021) and NeRF– (Wang et al., 2021b), which concurrently optimize NeRF and camera parameters. The results are detailed in Table 6. From the table, our method performs better in the extrinsics optimization with smaller rotation and translation errors, and the intrinsics is closer to the ground truth than other methods. Fig. 13 visualizes the extrinsics (pose) optimized by different methods, where red denotes the ground truth, and blue denotes the noisy input or optimized poses. Compared with other methods, the camera poses optimized by ours are closer to the ground truth.

4.4. Ablation analysis

In this section, we present ablation studies in terms of scene priors (Section 4.4.1), the joint optimization of camera parameters with NeRF weights (Section 4.4.2), and model scales (Section 4.4.3).

4.4.1. Analysis of scene priors

Scene priors significantly improve the performance of the model in NVS. As presented in Table 6, it shows that the PSNR improves by 1.926 dB and 1.19 dB with the benefit of scene priors on synthetic and real scenes, respectively. We conduct further experiments to verify the robustness of scene priors under various conditions of using the increasing number of views for training, as presented in Fig. 11. It demonstrates that the scene priors consistently improve the PSNR metric with different view numbers both on real and synthetic scenes.

The further analysis from Fig. 11 reveals that scene priors reduce the number of views required for NVS. Our method achieves competitive or even better performance compared to the baseline method BungeeNeRF (Xiangli et al., 2022) with only one third of the view numbers required for BungeeNeRF. Specifically, BungeeNeRF achieves an average PSNR of 23.145 dB using all images (195, 266, 119, and 238 views for the four scenes, respectively) on real scenes. In contrast, our method achieves a similar average PSNR of 23.135 dB with just one-third of the images. On synthetic scenes, the comparison is 23.916 dB (BungeeNeRF with 463 and 455 view images for the two scenes, respectively) versus 24.157 dB (our method with one third of the views) as presented in Fig. 11. This underscores the efficacy of scene priors in extracting more information from fewer views with greater efficiency.

Deriving pixel-wisely aligned and accurate features as scene priors is indeed important in improving the performance of the model. We replace our tiny U-shaped network for extracting features of view images with a ResNet and present the results in Fig. 12. The U-shaped network can upsample the downsampled features in a learning way (lines in blue) instead of a fixed mode as interpolation for ResNet (lines in red). It shows that using the learning mode for upsampling consistently outperforms the fixed mode by an average of 0.5 dB in PSNR.

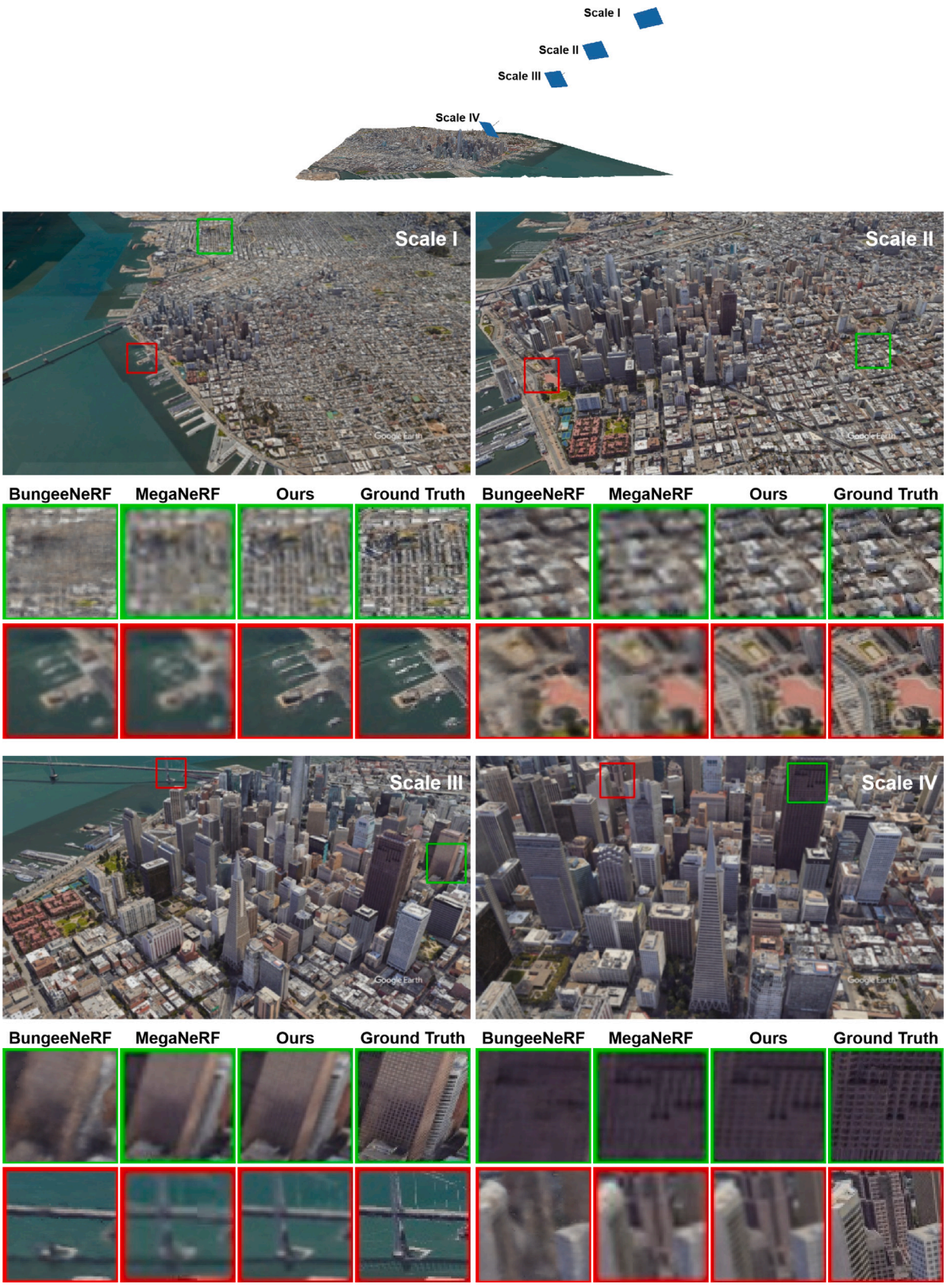
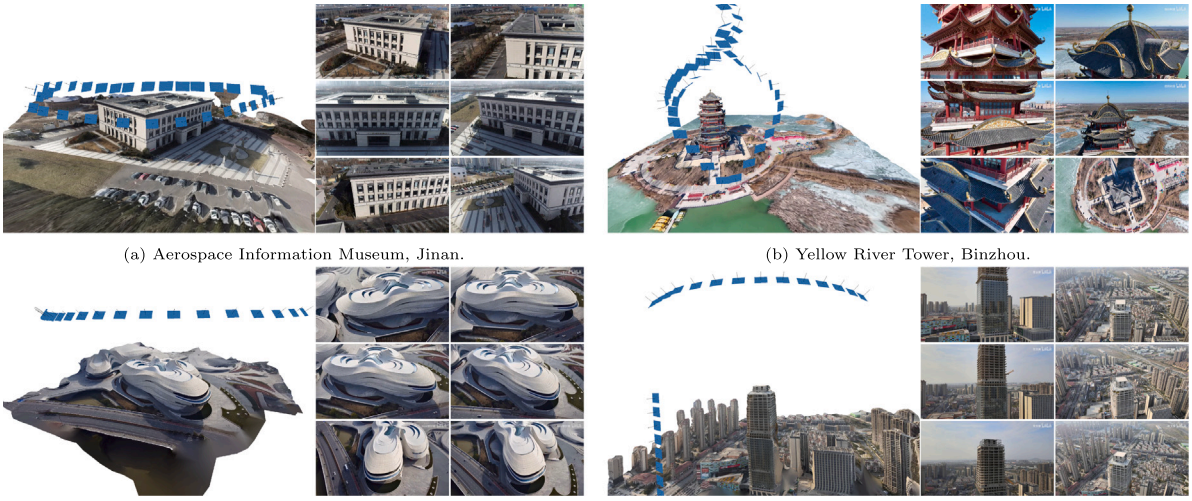


Fig. 8. Qualitative results on Transamerica. Our method consistently outperforms the baseline method in robustly capturing the finer details at all scales.



(a) Aerospace Information Museum, Jinan. (b) Yellow River Tower, Binzhou.

(c) Meixihu Arts Center, Changsha. (d) Greenland Xindu Mall, Hefei.

Fig. 9. The visualization of real scenes captured by drone or collected from the Internet.

4.4.2. Analysis of pose optimization

The joint optimization of camera parameters improves the results of NVS. The numerical results presented in Table 6 show that the PSNR improves by 0.304 dB and 0.812 dB with camera parameter optimization on synthetic and real scenes, respectively. It can be found that the camera parameter optimization is more important for real scenes than synthetic scenes. That is because for synthetic scenes, camera parameters are directly derived from Google Earth studio, and there is little room for further improvement. However, in real scenes, there are errors in estimating the initial camera parameters via the SfM methods (Schönberger and Frahm, 2016). Therefore, jointly optimizing camera parameters can effectively minimize errors and improve the synthesized views. The qualitative results in Fig. 14 as well demonstrate it, and image artifacts and blurring problems caused by inaccurate camera poses are significantly reduced.

4.4.3. Analysis of model scale

Increasing the model scale, i.e., adding more residual blocks to the NeRF in Fig. 6, improves the quality of view synthesis. As shown in Fig. 15, the PSNR also increases with more residual blocks. However, adding more residual blocks cannot improve the PSNR infinitely. The performance converges to 25.35 dB and 25.44 dB on real and synthetic scenes respectively as the number of residual blocks reaches 4.

4.5. Limitations

Although our method achieves impressive NVS results, it shares a same challenge as other NeRF-based methods, which is the high computational cost for rendering images. Since rendering a single image pixel requires hundreds of neural network calls, rendering an image takes many seconds even on advanced GPUs. Specifically, our method takes about 7 s to render a 1080p resolution image on a single NVIDIA RTX 3090 GPU. The training time takes an average of 28 h per scene on a single NVIDIA RTX 3090 GPU. There is currently some work dedicated to NeRF acceleration (both for training (Sun et al., 2022; Müller et al., 2022) and inference (Garbin et al., 2021; Hu et al., 2022; Wadhwani and Kojima, 2022)), but this is beyond the focus of our work in this paper, and we will follow up with efforts on this limitation as well.

5. Conclusion

In this work, we propose a new unified framework for Novel View Synthesis (NVS) of urban scenes, which integrates together the scene priors and the joint optimization of camera parameters under the proposed orthogonality constraint. Experiments on the two synthetic scenes and four real scenes show that our method outperforms other popular methods (Mildenhall et al., 2021; Barron et al., 2021; Xiangli et al., 2022) by about 2–5 dB in PSNR. Our method shows superior results on both synthetic and real scenes, with PSNR reaching 25.375 dB and 25.512 dB in average, respectively. The integration of the scene priors effectively reduces the number of views required for NVS and achieves better or competitive performance with only one third of the views for the baseline. The ablation study reveals that extracting finer and more accurate scene features can further improve the results with only a tiny U-shaped network. The addition of the joint optimization of camera parameters can further improve the NVS performance in terms of numerical PSNR metric and visualization.

CRediT authorship contribution statement

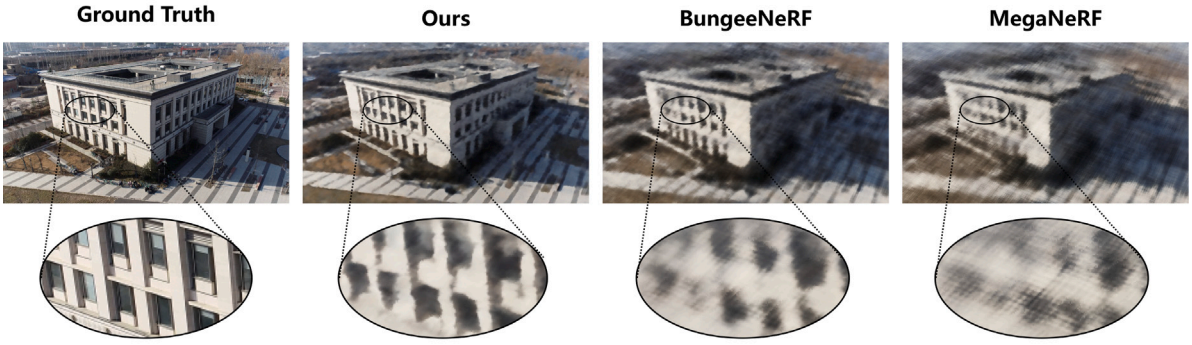
Kaiqiang Chen: Writing – review & editing, Writing – original draft, Validation, Resources, Methodology, Conceptualization. **Bo Dong:** Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Conceptualization. **Zhirui Wang:** Writing – review & editing, Validation, Project administration. **Peirui Cheng:** Validation, Software. **Menglong Yan:** Supervision, Data curation. **Xian Sun:** Validation, Funding acquisition. **Michael Weinmann:** Writing – review & editing, Validation. **Martin Weinmann:** Writing – review & editing, Validation.

Declaration of competing interest

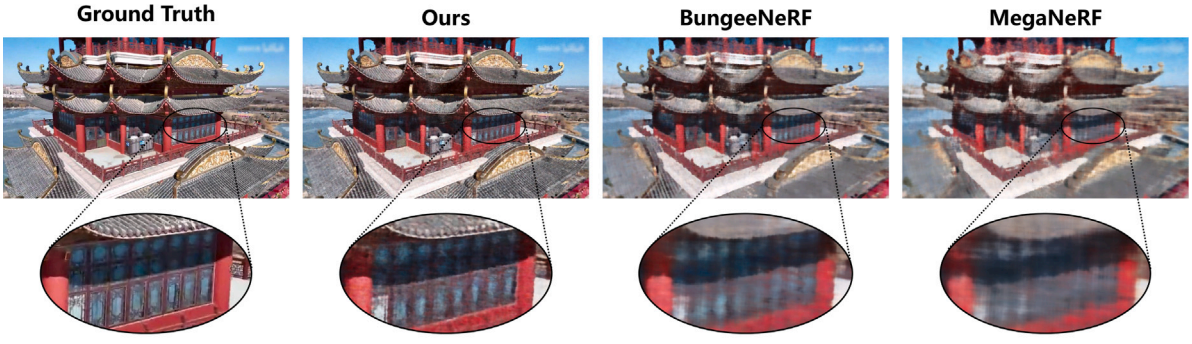
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

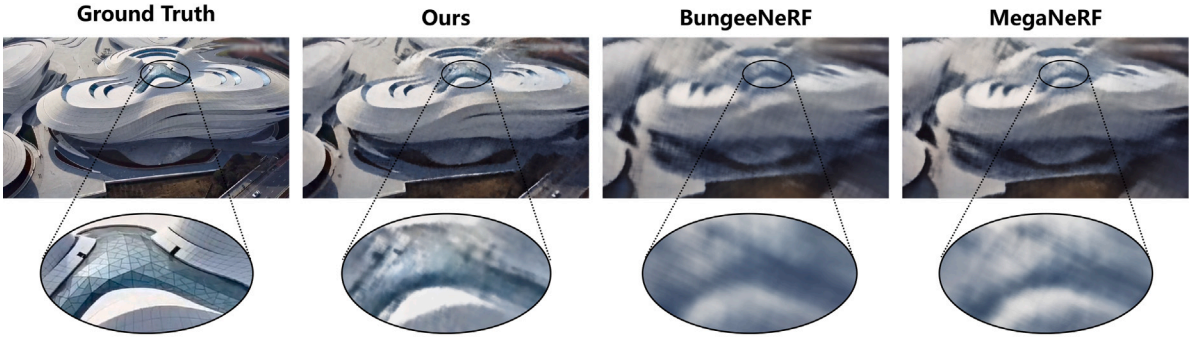
This work was supported by the National Natural Science Foundation of China under Grant 62331027 and Grant 62076241, and supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDA0360300.



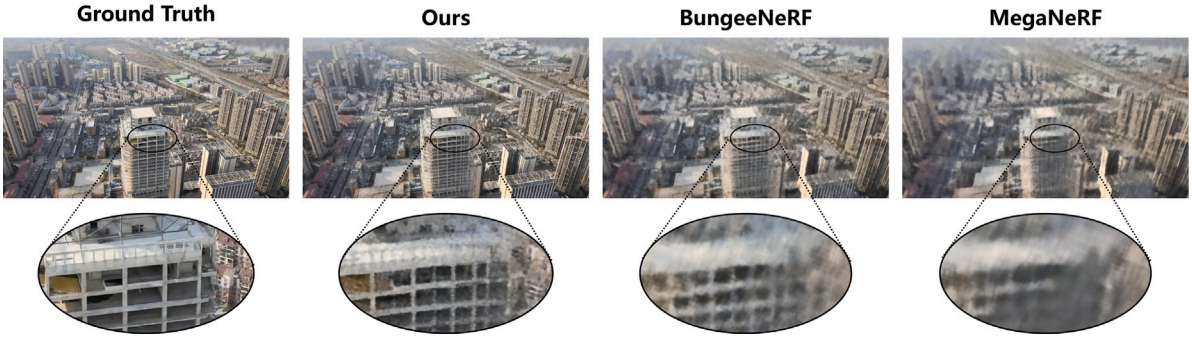
(a) Aerospace Information Museum, Jinan.



(b) Yellow River Tower, Binzhou.



(c) Meixihu Arts Center, Changsha.



(d) Greenland Xindu Mall, Hefei.

Fig. 10. A qualitative comparison of four real scenes. Our method consistently outperforms the baseline method in capturing fine details and color fidelity of the scene. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

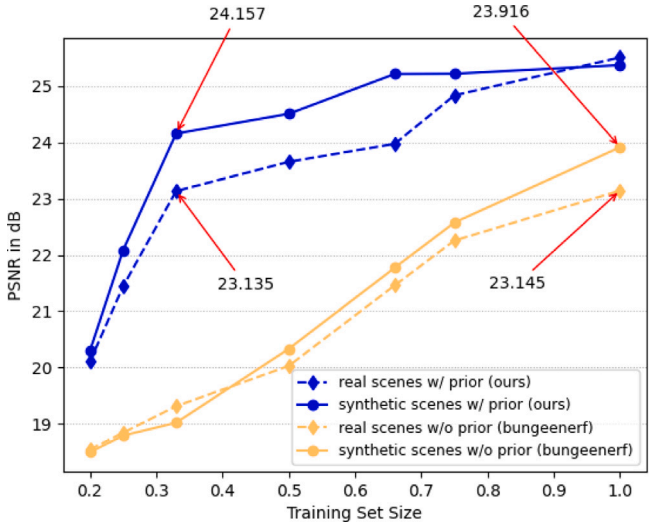


Fig. 11. The PSNR metrics derived from different models under various conditions. The vertical axis represents the PSNR and the horizontal axis represents the training set scale from 20% of the complete training images in the datasets to 100%. It is noted that samples are drawn uniformly across the entire image sequence with the same starting and ending viewing directions at varying sparsity degrees of views in between. The solid and dashed lines correspond to results on synthetic and real scenes, respectively. Specifically, lines in blue represent the performance of our method on synthetic scenes and real scenes respectively; lines in yellow represent the performance of the baseline BungeeNeRF, that means without any prior. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

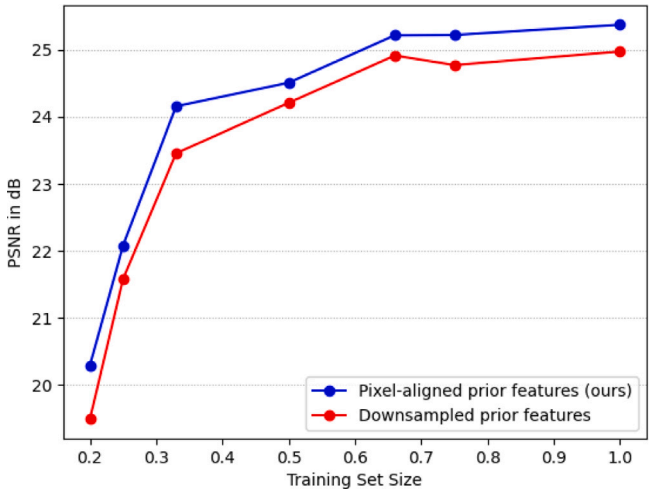


Fig. 12. PSNR of training with 20% to 100% view images on synthetic scenes. The blue line represents pixel-aligned features extracted using U-Net (ours), and the red line represents downsampled features extracted using ResNet. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

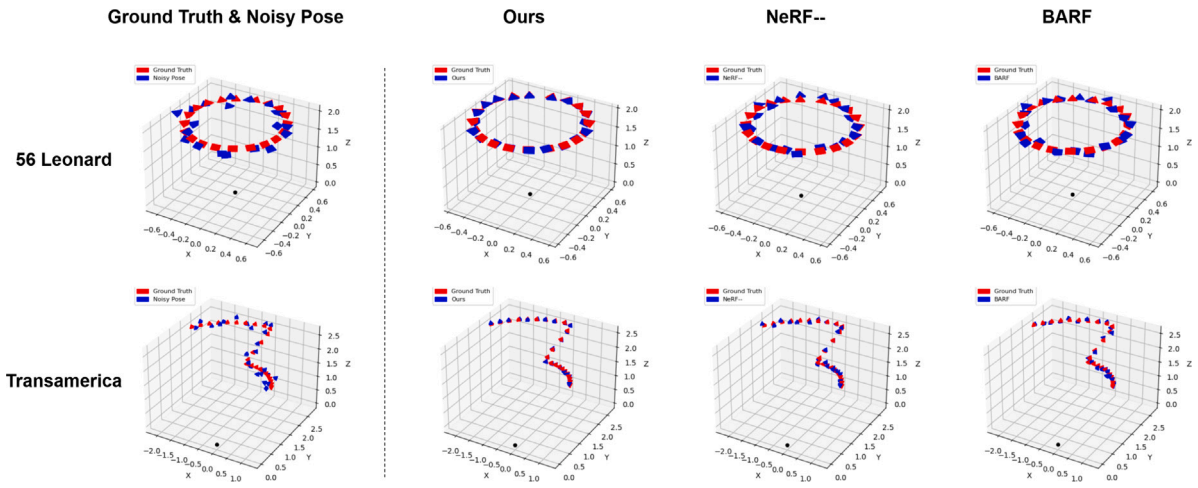


Fig. 13. Visualizations of optimizing camera extrinsics (poses) using different methods, where the red represents the ground truth, and the blue represents the pose after adding noise or optimization. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

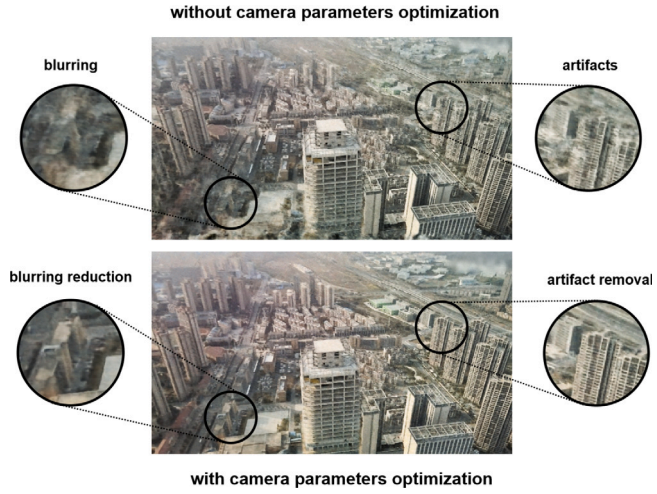


Fig. 14. The impact of the joint optimization of camera parameters initialized with the SfM (Schönberger and Frahm, 2016). It demonstrates that the joint optimization of camera parameters can reduce image blurring and artifacts, and improve the quality of the synthesized novel views.

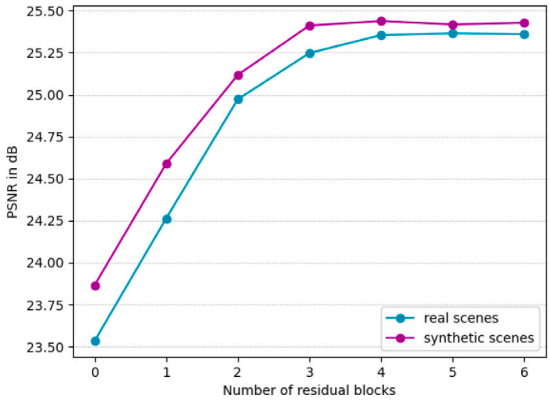


Fig. 15. The impact of adding more residual blocks. 0 represents only the base block used.

References

Bach, T.B., Dinh, T.T., Lee, J.-H., 2022. FeatLoc: Absolute pose regressor for indoor 2D sparse features with simplistic view synthesizing. ISPRS J. Photogramm. Remote Sens. 189, 50–62.

Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P., 2021. 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.

Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F., 2015. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago.

Chen, Y., Chen, X., Wang, X., Zhang, Q., Guo, Y., Shan, Y., Wang, F., 2022a. Local-to-global registration for bundle-adjusting neural radiance fields. CoRR arXiv:2211.11505.

Chen, S., Liang, L., Ouyang, J., 2022b. Accurate structure from motion using consistent cluster merging. Multimedia Tools Appl. 81 (17), 24913–24935.

Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H., 2021. MVSNeRF: Fast generalizable radiance field reconstruction from multi-view stereo. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision. IEEE, pp. 14104–14113.

Chen, S., Zhang, Y., Xu, Y., Zou, B., 2022c. Structure-aware NeRF without posed camera via epipolar constraint. ArXiv preprint arXiv:2210.00183.

Chibane, J., Bansal, A., Lazova, V., Pons-Moll, G., 2021. Stereo radiance fields (SRF): learning view synthesis for sparse views of novel scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, Virtual, June 19–25, 2021. Computer Vision Foundation / IEEE, pp. 7911–7920.

Chng, S., Ramasinghe, S., Sherrah, J., Lucey, S., 2022. Gaussian activated neural radiance fields for high fidelity reconstruction and pose estimation. In: Avidan, S., Brostow, G.J., Cissé, M., Farinella, G.M., Hassner, T. (Eds.), Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIII. In: Lecture Notes in Computer Science, vol. 13693, Springer, pp. 264–280.

Chung, C., Tseng, Y., Hsu, Y., Shi, X.Q., Hua, Y., Yeh, J., Chen, W., Chen, Y., Hsu, W.H., 2022. Orbeez-SLAM: A real-time monocular visual SLAM with ORB features and nerf-realized mapping. CoRR arXiv:2209.13274.

Condorelli, F., Rinaudo, F., Salvadore, F., Tagliaventi, S., 2021. A comparison between 3D reconstruction using nerf neural networks and mvs algorithms on cultural heritage images. Int. Arch. Photogramm. Remote Sens. Spat. Inform. Sci. 43, 565–570.

Cooke, E., Kauff, P., Sikora, T., 2006. Multi-view synthesis: A novel view creation approach for free viewpoint video. Signal Process., Image Commun. 21 (6), 476–492.

Debevec, P., Yu, Y., Boshokov, G., 1998. Efficient view-dependent IBR with projective texture-mapping. In: Proceedings of the EG Rendering Workshop. vol. 4, (no. 11).

Derksen, D., Izzo, D., 2021. Shadow neural radiance fields for multi-view satellite photogrammetry. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1152–1161.

Flynn, J., Neulander, I., Philbin, J., Snavely, N., 2016. Deepstereo: Learning to predict new views from the world's imagery. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5515–5524.
Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., Kanazawa, A., 2022. Plenoxels: Radiance fields without neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5501–5510.
Fülöp-Balogh, B.-E., Tursman, E., Tompkin, J., Digne, J., Bonneel, N., 2022. Dynamic scene novel view synthesis via deferred spatio-temporal consistency. Comput. Graph..
Garbin, S.J., Kowalski, M., Johnson, M., Shotton, J., Valentin, J., 2021. Fastnerf: High-fidelity neural rendering at 200fps. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14346–14355.
Guo, H., Peng, S., Lin, H., Wang, Q., Zhang, G., Bao, H., Zhou, X., 2022. Neural 3D scene reconstruction with the manhattan-world assumption. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, pp. 5501–5510.
Haitz, D., Jutzi, B., Ulrich, M., Jäger, M., Hübner, P., 2023. Combining hololens with instant-nerfs: advanced real-time 3D mobile mapping. Int. Arch. Photogramm. Remote Sens. Spat. Inform. Sci. 48, 167–174.
Hartley, R.I., 1997. In defense of the eight-point algorithm. IEEE Trans. Pattern Anal. Mach. Intell. 19 (6), 580–593.
He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778.
Heo, H., Kim, T., Lee, J., Lee, J., Kim, S., Kim, H.J., Kim, J., 2023. Robust camera pose refinement for multi-resolution hash encoding. CoRR arXiv:2302.01571.
Hu, T., Liu, S., Chen, Y., Shen, T., Jia, J., 2022. Efficientnerf efficient neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12902–12911.
Huang, H., Michelini, M., Schmitz, M., Roth, L., Mayer, H., 2020. LOD3 building reconstruction from multi-source images. Int. Arch. Photogramm. Remote Sens. Spat. Inform. Sci. 43, 427–434.
Jain, A., Tancik, M., Abbeel, P., 2021. Putting NeRF on a diet: Semantically consistent few-shot view synthesis. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision. IEEE, pp. 5865–5874.
Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H., 2014. Large scale multi-view stereopsis evaluation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 406–413.
Jeong, Y., Ahn, S., Choy, C., Anandkumar, A., Cho, M., Park, J., 2021. Self-calibrating neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5846–5854.
Johari, M.M., Lepoittevin, Y., Fleuret, F., 2022. Geonerf: Generalizing nerf with geometry priors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18365–18375.
Ju, Y., Dong, J., Chen, S., 2021. Recovering surface normal and arbitrary images: A dual regression network for photometric stereo. IEEE Trans. Image Process. 30, 3676–3690.
Ju, Y., Shi, B., Chen, Y., Zhou, H., Dong, J., Lam, K.-M., 2023. GR-PSN: learning to estimate surface normal and reconstruct photometric stereo images. IEEE Trans. Vis. Comput. Graphics.
Kajiya, J.T., Von Herzen, B.P., 1984. Ray tracing volume densities. ACM SIGGRAPH Comput. Graph. 18 (3), 165–174.
Kang, S.B., 1998. Geometrically valid pixel reprojection methods for novel view synthesis. ISPRS J. Photogramm. Remote Sens. 53 (6), 342–353.
Kendall, A., Grimes, M., Cipolla, R., 2015. Convolutional networks for real-time 6-DOF camera relocalization. ArXiv preprint arXiv:1505.07427.
Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. ArXiv preprint arXiv:1412.6980.
Kniaz, V., Knyaz, V., Bordodymov, A., Moshkantsev, P., Novikov, D., Barylnik, S., 2023. Double nerf: Representing dynamic scenes as neural radiance fields. Int. Arch. Photogramm. Remote Sens. Spat. Inform. Sci. 48, 115–120.
Li, Z., Li, L., Zhu, J., 2023. Read: Large-scale neural scene rendering for autonomous driving. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, (no. 2), pp. 1522–1529.
Lin, K.-E., Lin, Y.-C., Lai, W.-S., Lin, T.-Y., Shih, Y.-C., Ramamoorthi, R., 2023. Vision transformer for nerf-based view synthesis from a single input image. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 806–815.
Lin, C.-H., Ma, W.-C., Torralba, A., Lucey, S., 2021. Barf: Bundle-adjusting neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5741–5751.
Lowe, D.G., 1999. Object recognition from local scale-invariant features. In: Proceedings of the IEEE International Conference on Computer Vision. 2, pp. 1150–1157.
Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. 60, 91–110.
Maggio, D., Abate, M., Shi, J., Mario, C., Carlone, L., 2022. Loc-NeRF: Monte Carlo localization using neural radiance fields. CoRR arXiv:2209.09050.
Marí, R., Facciolo, G., Ehret, T., 2022. Sat-NeRF: Learning multi-view satellite photogrammetry with transient objects and shadow modeling using RPC cameras. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. CVPRW, IEEE, pp. 1310–1320.
Martin-Brualla, R., Pandey, R., Yang, S., Pidlypenskyi, P., Taylor, J., Valentin, J., Khamis, S., Davidson, P., Tkach, A., Lincoln, P., Kowdle, A., Rhemann, C., Goldman, D.B., Keskin, C., Seitz, S., Izadi, S., Fanello, S., 2018. LookinGood: Enhancing performance capture with real-time neural re-rendering. ACM Trans. Graph. 37 (6).
Martin-Brualla, R., Radwan, N., Sajjadi, M.S., Barron, J.T., Dosovitskiy, A., Duckworth, D., 2021. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7210–7219.
Meshry, M., Goldman, D.B., Khamis, S., Hoppe, H., Pandey, R., Snavely, N., Martin-Brualla, R., 2019. Neural rerendering in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6878–6887.
Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R., 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. Commun. ACM 65 (1), 99–106.
Müller, T., Evans, A., Schied, C., Foco, M., Bódis-Szomorú, A., Deutsch, I., Shelley, M., Keller, A., 2022. Instant neural radiance fields. In: ACM SIGGRAPH 2022 Real-Time Live!. pp. 1–2.
Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S., Geiger, A., Radwan, N., 2022. 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.
Nistér, D., 2004. An efficient solution to the five-point relative pose problem. IEEE Trans. Pattern Anal. Mach. Intell. 26 (6), 756–770.
Oechsle, M., Peng, S., Geiger, A., 2021. UNISURF: unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision. IEEE, pp. 5569–5579.
Qi, Y., Yang, S., Cai, S., Hou, F., Shen, X., Zhao, Q., 2009. A method of 3D modeling and codec. Sci. China Ser. F 52 (5), 758–769.
Rematas, K., Liu, A., Srinivasan, P.P., Barron, J.T., Tagliasacchi, A., Funkhouser, T., Ferrari, V., 2022. Urban radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12932–12942.
Riegler, G., Koltun, V., 2020. Free view synthesis. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16. Springer, pp. 623–640.
Rosinol, A., Leonard, J.J., Carlone, L., 2022. NeRF-SLAM: Real-time dense monocular SLAM with neural radiance fields. CoRR arXiv:2210.13641.
Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H., 2019. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2304–2314.
Schönberger, J.L., Frahm, J.-M., 2016. Structure-from-motion revisited. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4104–4113.
Semeraro, F., Zhang, Y., Wu, W., Carroll, P., 2023. NeRF applied to satellite imagery for surface reconstruction. ArXiv preprint arXiv:2304.04133.
Shi, Y., Rong, D., Ni, B., Chen, C., Zhang, W., 2022. GARF: geometry-aware generalized neural radiance field. CoRR arXiv:2212.02280.
Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. ArXiv preprint arXiv:1409.1556.
Sitzmann, V., Zollhöfer, M., Wetzstein, G., 2019. Scene representation networks: Continuous 3d-structure-aware neural scene representations. Adv. Neural Inf. Process. Syst. 32.
Somraj, N., Soundararajan, R., 2023a. ViP-NeRF: Visibility prior for sparse input neural radiance fields. ArXiv preprint arXiv:2305.00041.
Somraj, N., Soundararajan, R., 2023b. ViP-NeRF: Visibility prior for sparse input neural radiance fields. In: Brunvand, E., Sheffer, A., Wimmer, M. (Eds.), ACM SIGGRAPH 2023 Conference Proceedings, SIGGRAPH 2023, Los Angeles, CA, USA, August 6–10, 2023. ACM, pp. 71:1–71:11.
Sucar, E., Liu, S., Ortiz, J., Davison, A.J., 2021. iMAP: Implicit mapping and positioning in real-time. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10–17, 2021. IEEE, pp. 6209–6218.
Sun, C., Sun, M., Chen, H.-T., 2022. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5459–5469.
Sundermeyer, M., Marton, Z.-C., Durner, M., Brucker, M., Triebel, R., 2018. Implicit 3d orientation learning for 6d object detection from rgb images. In: Proceedings of the European Conference on Computer Vision. pp. 699–715.
Tancik, M., Casser, V., Yan, X., Pradhan, S., Mildenhall, B., Srinivasan, P.P., Barron, J.T., Kretzschmar, H., 2022. Block-nerf: Scalable large scene neural view synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8248–8258.
Truong, P., Rakotosaona, M., Manhardt, F., Tombari, F., 2022. SPARF: neural radiance fields from sparse and noisy poses. CoRR arXiv:2211.11738.
Turki, H., Ramanan, D., Satyanarayanan, M., 2022. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12922–12931.
Wadhwani, K., Kojima, T., 2022. SqueezeNeRF: Further factorized FastNeRF for memory-efficient inference. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2717–2725.

Wang, Q., Wang, Z., Genova, K., Srinivasan, P.P., Zhou, H., Barron, J.T., Martin-Brualla, R., Snavely, N., Funkhouser, T., 2021a. Ibrnet: Learning multi-view image-based rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4690–4699.

Wang, Z., Wu, S., Xie, W., Chen, M., Prisacariu, V.A., 2021b. NeRF–: Neural radiance fields without known camera parameters. ArXiv preprint arXiv:2102.07064.

Wilson, K., Snavely, N., 2014. Robust global translations with 1dsfm. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part III 13. Springer, pp. 61–75.

Xiangli, Y., Xu, L., Pan, X., Zhao, N., Rao, A., Theobalt, C., Dai, B., Lin, D., 2022. Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII. Springer, pp. 106–122.

Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U., 2019. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. Adv. Neural Inf. Process. Syst. 32.

Yang, M.Y., Cao, Y., McDonald, J., 2011. Fusion of camera images and laser scans for wide baseline 3D scene alignment in urban environments. ISPRS J. Photogramm. Remote Sens. 66 (6), S52–S61.

Yen-Chen, L., Florence, P., Barron, J.T., Rodriguez, A., Isola, P., Lin, T.-Y., 2021. Inerf: Inverting neural radiance fields for pose estimation. In: Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, IEEE, pp. 1323–1330.

Yu, Z., Peng, S., Niemeyer, M., Sattler, T., Geiger, A., 2022. MonoSDF: Exploring monocular geometric cues for neural implicit surface reconstruction. Adv. Neural Inf. Process. Syst..

Yu, A., Ye, V., Tancik, M., Kanazawa, A., 2021. Pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4578–4587.

Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O., 2018. The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 586–595.

Zhang, X., Srinivasan, P.P., Deng, B., Debevec, P.E., Freeman, W.T., Barron, J.T., 2021. NeRFactor: neural factorization of shape and reflectance under an unknown illumination. ACM Trans. Graph. 40 (6), 237:1–237:18.

Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.A., 2016. View synthesis by appearance flow. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, Part IV 14. Springer, pp. 286–301.

Zhu, Z., Peng, S., Larsson, V., Cui, Z., Oswald, M.R., Geiger, A., Pollefeys, M., 2023. NICER-SLAM: neural implicit scene encoding for RGB SLAM. CoRR arXiv:2302.03594.

Zhu, Z., Peng, S., Larsson, V., Xu, W., Bao, H., Cui, Z., Oswald, M.R., Pollefeys, M., 2022. NICE-SLAM: neural implicit scalable encoding for SLAM. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18–24, 2022. IEEE, pp. 12776–12786.