Grid-guided Neural Radiance Fields for Large Urban Scenes

Linning Xu[⋆1], Yuanbo Xiangli[⋆1], Sida Peng[4], Xingang Pan[3],
Nanxuan Zhao[5], Christian Theobalt[3], Bo Dai[2✉], Dahua Lin[1,2]
[1]The Chinese University of Hong Kong, [2]Shanghai AI Laboratory,
[3]Max Planck Institute for Informatics, [4]Zhejiang University, [5]Adobe Research
{xl020,xy019,dhlin}@ie.cuhk.edu.hk, {xpan,theobalt}@mpi-inf.mpg.de,
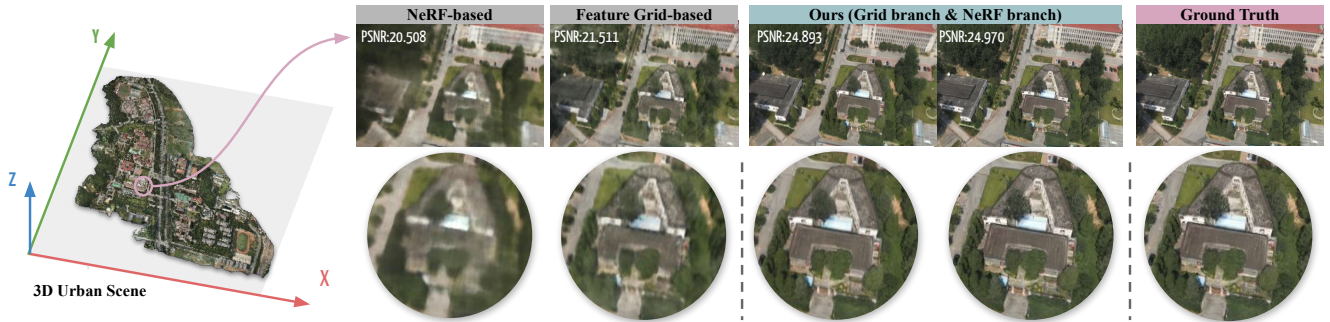pengsida@zju.edu.cn, nanxuanzhao@gmail.com, daibo@pjlab.org.cn

Figure 1. We perform large urban scene rendering with a novel grid-guided neural radiance fields. An example of our target large urban scene is shown on the left, which spans over $2.7km^2$ ground areas captured by over $5k$ drone images. We show that the rendering results from NeRF-based methods [34, 62] are blurry and overly smoothed with limited model capacity, while feature grid-based methods [9, 36] tend to display noisy artifacts when adapting to large-scale scenes with high-resolution feature grids. Our proposed two-branch model combines the merits from both approaches and achieves photorealistic novel view renderings with remarkable improvements over existing methods. Both two branches gain significant enhancements over their individual baselines. (Project page: https://city-super.github.io/gridnerf)

Abstract

Purely MLP-based neural radiance fields (NeRF-based methods) often suffer from underfitting with blurred renderings on large-scale scenes due to limited model capacity. Recent approaches propose to geographically divide the scene and adopt multiple sub-NeRFs to model each region individually, leading to linear scale-up in training costs and the number of sub-NeRFs as the scene expands. An alternative solution is to use a feature grid representation, which is computationally efficient and can naturally scale to a large scene with increased grid resolutions. However, the feature grid tends to be less constrained and often reaches suboptimal solutions, producing noisy artifacts in renderings, especially in regions with complex geometry and texture. In this work, we present a new framework that realizes high-fidelity rendering on large urban scenes while being computationally efficient. We propose to use a compact multiresolution ground feature plane representation to coarsely capture the scene, and complement it with positional encoding inputs through another NeRF branch for rendering in a joint learning fashion. We show that such an integration can utilize the advantages of two alternative solutions: a light-weighted NeRF is sufficient, under the guidance of the feature grid representation, to render photorealistic novel views with fine details; and the jointly optimized ground feature planes, can meanwhile gain further refinements, forming a more accurate and compact feature space and output much more natural rendering results.

1. Introduction

Large urban scene modeling has been drawing lots of research attention with the recent emergence of neural radiance fields (NeRF) due to its photorealistic rendering and model compactness [3, 34, 56, 59, 62, 64]. Such modeling can enable a variety of practical applications, including autonomous vehicle simulation [23, 39, 67], aerial surveying [6, 15], and embodied AI [35, 61]. NeRF-based methods have shown impressive results on object-level scenes with their *continuity prior* benefited from the MLP architecture and *high-frequency details* with the globally shared positional encodings. However, they often fail to model large and complex scenes. These methods suffer from underfit-

ting due to limited model capacity and only produce blurred renderings without fine details [56,62,64]. BlockNeRF [58] and MegaNeRF [62] propose to geographically divide urban scenes and assign each region a different sub-NeRF to learn in parallel. Subsequently, when the scale and complexity of the target scene increases, they inevitably suffer from a trade-off between the number of sub-NeRFs and the capacity required by each sub-NeRF to fully capture all the fine details in each region. Another stream of grid-based representations represents the target scene using a grid of features [27, 27, 30, 36, 55, 68, 69]. These methods are generally much faster during rendering and more efficient when the scene scales up. However, as each cell of the feature grid is individually optimized in a locally encoded manner, the resulting feature grids tend to be less continuous across the scene compared to NeRF-based methods. Although more flexibility is intuitively beneficial for capturing fine details, the lack of inherent continuity makes this representation vulnerable to suboptimal solutions with noisy artifacts, as demonstrated in Fig. 1.

To effectively reconstruct large urban scenes with implicit neural representations, in this work, we propose a two-branch model architecture that takes a unified scene representation that integrates both **grid-based** and **NeRF-based** approaches under a joint learning scheme. Our key insight is that these two types of representations can be used complementary to each other: while feature grids can easily fit local scene content with explicit and independently learned features, NeRF introduces an inherent *global continuity* on the learned scene content with its shareable MLP weights across all 3D coordinate inputs. NeRF can also encourage capturing high-frequency scene details by matching the *positional encodings* as Fourier features with the bandwidth of details. However, unlike feature grid representation, NeRF is less effective in compacting large scene contents into its globally shared latent coordinate space.

Concretely, we firstly model the target scene with a feature grid in a pre-train stage, which coarsely captures scene geometry and appearance. The coarse feature grid is then used to 1) guide NeRF's point sampling to let it concentrate around the scene surface; and 2) supply NeRF's positional encodings with extra features about the scene geometry and appearance at sampled positions. Under such guidance, NeRF can effectively and efficiently pick up finer details in a drastically compressed sampling space. Moreover, as coarse-level geometry and appearance information are explicitly provided to NeRF, a light-weight MLP is sufficient to learn a mapping from global coordinates to volume densities and color values. The coarse feature grids get further optimized with gradients from the NeRF branch in the second joint-learning stage, which regularizes them to produce more accurate and natural rendering results when applied in isolation. To further reduce memory footprint and learn a

reliable feature grid for large urban scenes, we adopt a compact factorization of the 3D feature grid to approximate it without losing representation capacity. Based on the observation that essential semantics such as the urban layouts are mainly distributed on the ground (*i.e.*, xy-plane), we propose to factorize the 3D feature grid into 2D ground feature planes spanning the scene and a vertically shared feature vector along the z-axis. The benefits are manifold: 1) The memory is reduced from $\mathcal{O}(N^3)$ to $\mathcal{O}(N^2)$. 2) The learned feature grid is enforced to be disentangled into highly compact ground feature plans, offering explicit and informative scene layouts. Extensive experiments show the effectiveness of our unified model and scene representation. When rendering novel views in practice, users are allowed to use either the grid branch at a faster rendering speed, or the NeRF branch, with more high-frequency details and spatial smoothness, yet at the cost of a relatively slower rendering.

2. Related Works and Background

Large-scale Scene Reconstruction and Rendering. This is a long-standing problem in computer vision and graphics, and many early works [1, 17, 24, 43, 48, 52, 72] have attempted to solve it. Most of these methods adopt a three-stage pipeline. They firstly detect salient points [19, 29] in 2D images and construct the point descriptors [4, 47]. The point descriptors are then matched across images to obtain 2D correspondences that are used for camera pose estimation and 3D point triangulation. Finally, the camera poses and 3D points are jointly optimized to minimize the difference between the projected and image points using bundle adjustment [18, 60]. These methods have shown impressive reconstruction performance on large-scale scenes. [28, 50] utilize the reconstruction results to accomplish the free-viewpoint navigation of scenes. However, there are often artifacts or holes in the reconstructed scenes, which limit the rendering quality. Recent methods [26, 31, 33] exploit deep learning techniques to improve the results of image synthesis. [33] rasterizes the recovered point clouds into deep buffers, which are then interpreted into 2D images using the 2D convolutional neural networks. More recently, [31] leverages the technique of generative latent optimization to reconstruct radiance fields from photo collections, enabling it to achieve photorealistic rendering results. [14, 51] adopted a similar ground plane representation as us for scene generation and static-dynamic disentanglement tasks respectively, yet the scale of their applicable scenes and the rendering qualities remain limited.

Volumetric Scene Representations. Coordinate-based multi-layer perceptrons have become a popular representation for 3D shape modeling [7, 10, 32, 40] and novel view synthesis [2, 3, 34, 38]. To represent high-resolution 3D shapes, some methods adopt MLP networks that take continuous 3D coordinates as input and predict the target val-

ues [13, 32, 40]. [11, 12, 42] introduces convolutional neural networks to learn powerful scene priors, enabling the coordinate-based representations to handle larger scenes. In the field of novel view synthesis, NeRF [34] represents 3D scenes as density and color fields and optimizes this representation from images through volume rendering techniques. [20,27,46,69] extend NeRF with efficient data structures to accelerate the rendering process. [8,37,49,66] combine NeRF with generative models and achieve 3D-aware image generation. [25, 41, 44] augment NeRF with motion fields, enabling them to handle dynamic scenes.

NeRF Scale-up. While the aforementioned NeRF approaches mainly consider scenes of a limited scale, scaling up NeRF to handle large-scale scenes such as cities would enable broader applications. [27, 62, 65] have tried to improve the rendering quality of NeRF on large-scale scenes. Feature grid methods [22,27] map input coordinates to a high dimensional space with a lookup from the predefined table of learnable feature vectors, which augments the approximation ability of neural networks. PointNeRF [65] regresses the radiance field from the point cloud and accomplishes the high-quality rendering of indoor scenes. BungeeNeRF [64] designs a multi-scale representation that efficiently models the scene content and improves the rendering quality. [56, 62] decompose the scene into several spatial regions that are separately represented by NeRF networks. When processing large scenes, another critical problem is how to reduce the training time. Several techniques such as image encoder [63,71], auto-decoder [16,45] and meta-learning [5, 57] are used to pretrain networks on a dataset to learn the scene prior for improving the optimization process. [36, 53, 54, 68] further explore the grid representations to accelerate the training speed. TensoRF [9] investigate the factorization of 3D scenes, enabling it to compactly represent scenes and achieve the fast training. Instant-NGP [36] adopts a multi-resolution hash table of feature vectors that enables extremely fast renderings. However, we notice both these methods suffer from noisy feature gridss when applied to large scenes.

3. Grid-guided Neural Radiance Fields

Recall that NeRF-based representations obtain the point density and color by passing the positional encoding (PE) of point coordinates into an 8-layer MLP [34]. Such a model is highly compact as the entire scene contents are encoded in the MLP weights taking PE embedding as inputs, yet they face difficulty in scaling up limited by model capacities. In contrast, grid-based representations encode a scene into a feature grid, which can be intuitively thought of as a 3D voxel grid with grid resolution matched with the actual 3D space. Each voxel stores a feature vector at the vertices and can then be interpolated to extract a feature value at the query point coordinate and converted to the point density

and color via a small network. As feature grids are often implemented as high-dimensional tensors, various factorization methods can also be applied to obtain more compact feature grid representations [9].

To effectively represent large urban scenes, we propose the grid-guided neural radiance fields, which combines the expertise of NeRF-based and grid-based methods. The grid features are enforced to reliably capture as much local information as possible with a *multi-resolution ground feature plane*. We then let the positional encoded coordinates information pick up the missing high-frequency details and produce high-quality renderings. With a rough construction of feature grid that captrue the scene at multiple resolution, the density field is also used to guide NeRF's sampling procedure. When training the NeRF branch, the grid features are jointly optimized and supervised with the reconstruction loss from both two branches.

Fig. 2 illustrates the overall pipeline of our system. In Sec. 3.1, we describe the pre-train of our multi-resolution ground feature plane representation; Sec. 3.2 introduces the grid-guided learning of neural radiance fields, corresponding to the NeRF branch in Fig. 2; Finally, we elaborate how the NeRF branch helps refine the pre-trained grid feature of the grid branch in Sec. 3.3.

3.1. Multi-resolution Feature Grid Pre-train

Fig. 1 illustrates a representative scenario of large urban scenes. Inspired by the fact that large urban scenes mainly ground on the xy-plane, we propose to represent the target large urban scene with a major *ground feature plane* by constructing a multi-resolution plane-vector feature space. Enforcing the ground plane compression gives more informative feature planes compared to the ones obtained from full 3D grids. This compact representation is especially suitable in urban scene scenarios and behaves robustly to sparse view training data. Various operations (*e.g.*, concatenation, outer product) can then be considered here to recover the 3D information from the 2D ground feature planes. The outer product operation was adopted in [9] from the perspective of tensor factorization with low-rank approximation, which achieves a more compressed memory footprint while maintaining high quality. The volume density $\sigma \in \mathbb{R}^+$ and view-dependent color $c \in \mathbb{R}^3$ grid-planes are separately learned to capture more environmental effects that only influences appearence. Formally, our grid-based radiance field is written as: $\sigma, c = F_\sigma(\mathcal{G}_\sigma(X)), F_c(\mathcal{G}_c(X), \mathrm{PE}(d))$ where $\mathcal{G}_\sigma(X) \in \mathbb{R}^{R_\sigma}$, $\mathcal{G}_c(X) \in \mathbb{R}^{R_c}$ are the extracted interpolated feature values from the two grid-planes at location $X \in \mathbb{R}^3$. F_σ, F_c are two fusing functions, implemented with two small MLP, that translate the concatenated density/appearance features to σ, c, and $d \in \mathbb{S}^2$ is the viewing direction. PE here represents the positional encoding $(\sin(\cdot), \cos(\cdot), \ldots, \sin(2^{L-1}(\cdot)), \cos(2^{L-1}(\cdot)))$
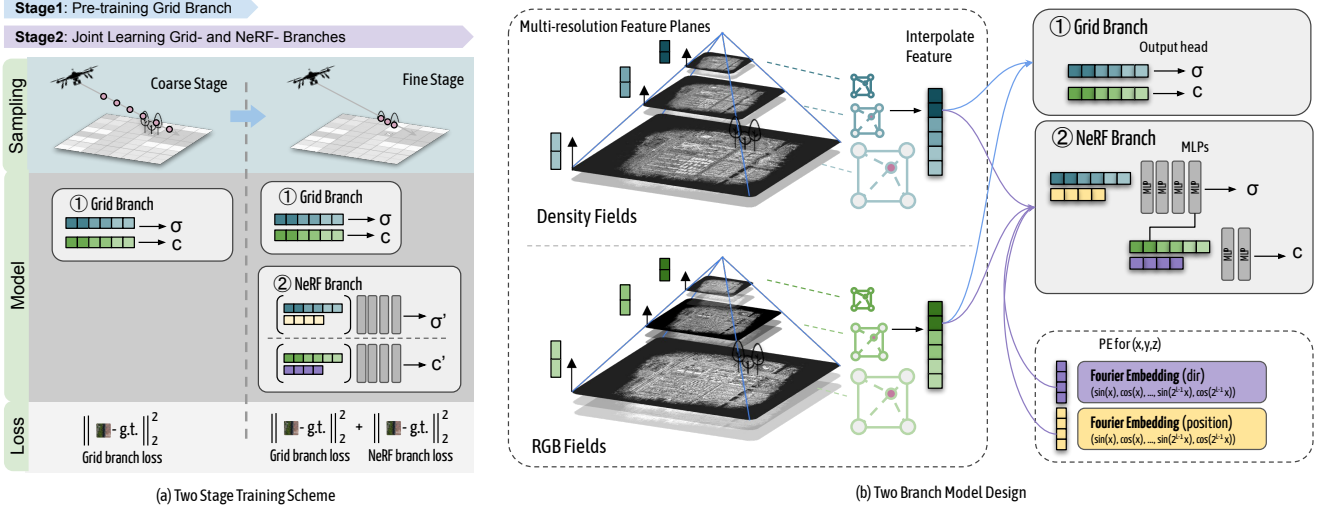
Figure 2. **Overview of our framework**. The core of our model is a novel two-branch structure, namely the grid branch and NeRF branch. 1) We start by capturing the scene with a pyramid of feature planes at the pre-train stage, and performing a coarse sampling of ray points and predicting their radiance values through a shallow MLP renderer (grid branch), supervised by the MSE loss on the volumetrically integrated pixel colors. This step yields a set of informative multi-resolution density/appearance feature planes. 2) Next, we proceed to the joint learning stage and perform a finer sampling. We use the learned feature grid to guide NeRF branch sampling to concentrate on the scene surface. The sampled points' grid feature is inferred by bilinear interpolation on the feature planes. The features are then concatenated with the positional encoding and fed to NeRF branch to predict volume density and color. Note that, the grid branch outputs maintain being supervised with the ground truth images along with the fine-rendering results from the NeRF branch during the joint training.

as in [34]. The grid branch is then trained with N query samples along the ray and predicts the pixel color following the volume rendering process as in [34], where the loss is the total squared error between the rendered and true pixel colors for this coarse sampling stage, as shown in Fig. 2 (a).

We approximate the full density and appearance grid features with the channel-wise outer product of the ground feature planes R_σ and R_c, as well as the globally encoded z-axis feature vectors, following the practice of [9]. For each channel $r \in R_\sigma$ and R_c, the corresponding tensor grid of features are: $\mathcal{T}_r = \mathbf{v}_r^z \circ \mathbf{M}_r^{xy}$, where $\mathbf{v}^z$ represents the vector along z-axis, $\mathbf{M}^{xy}$ denotes the matrix spanning xy-plane, and $\circ$ represents the outer product. With the constraints of learning a shared z-axis feature vector, the optimized ground feature plane is encouraged to encode sufficient local scene contents, that can be translated by a globally shared MLP renderer. For a specific grid resolution n, the density and the appearance tensors $\mathcal{G}_\sigma^n, \mathcal{G}_c^n$ are then obtained as the concatenation of R_σ, R_c feature components:

$$\mathcal{G}_\sigma^n = \oplus[(\mathbf{v}_{\sigma,r}^z \circ \mathbf{M}_{\sigma,r}^{xy})]_{R_\sigma}, \quad \mathcal{G}_c^n = \oplus[(\mathbf{v}_{c,r}^z \circ \mathbf{M}_{c,r}^{xy})]_{R_c},$$
(1)

where $\oplus$ denotes the concatenation operation over the R_σ and R_c dimension. To capture different degrees of scene local complexity, we learn a multi-resolution feature grid with $\mathcal{G}_\sigma = \{\mathcal{G}_\sigma^n\}$ and $\mathcal{G}_c = \{\mathcal{G}_c^n\}$. The yielding multi-resolution feature grid contains features at different granularity to describe the scene, which is particularly suitable for urban en-

vironments with objects appearing in different scales.

3.2. Grid-guided Neural Radiance Field

A NeRF trained from scratch is required to reason about the whole scene from purely positional inputs, which only provides a band of Fourier frequencies in PE. For large urban scenes that naturally bear a wide range of granularity for geometry and texture details, NeRF constantly biases towards learning low-frequency functions, as pointed out in [58, 64]. This problem gets amplified in large scenes where a large amount of information needs to be encoded. To remedy this, we propose to compress the sampling space of NeRF with the pre-trained feature grid density and enrich NeRF's pure coordinates inputs with the coarse grid features initialized in the pre-train stage.

Despite being of limited accuracy and granularity, the pre-trained grid feature can already offer an approximation of the scene which can be used to 1) guide NeRF's point sampling and 2) provide intermediate features as a complement to the coordinate inputs. As demonstrated in Fig. 2, instead of mapping coordinates spanning the entire sample space, NeRF can now concentrate on the approximated scene surface for more efficient and denser point sampling, and evoke high-frequency Fourier features in positional encoding to recover finer details. Meanwhile, points along the sampled ray are projected onto the multi-resolution feature planes to retrieve density and appearance features via bilinear interpolation. The inferred grid features are then

concatenated to the positional encoding as input to NeRF branch. The per-point density and color σ', c' are predicted via the NeRF branch network F' as:

$$(\sigma', c') = F'(\mathcal{G}_\sigma(X), \mathcal{G}_c(X), \text{PE}(X), \text{PE}(d)). \quad (2)$$

The multi-resolution feature plane plays a critical role as it provides information about the scene at multiple granularities, relieving the fitting burden of NeRF's PE so that it can concentrate on refining the fine details of the scene. Particularly, while a high grid resolution can guarantee that each voxel in space to captures its local contents, the quality grows at the cost of storage regardless of the possible heterogeneity of detail level across the scene. It is therefore more efficient to provide such high-frequency details with Fourier features that only cost several dimensions in PE and can be adapted to scene throughout the learning process.

Note that the two-branch supervision and the two-stage training are necessary as: (1) a randomly initialized feature grid can hardly provide informative scene contents and may entangle the role of two types of network inputs. (2) The pre-train stage is much faster than the one with NeRF branch included, making it more efficient to reliably construct a coarse geometry with grid branch only. (3) Unlike [53] which freezes the voxel grids when supplying PE inputs, we will later show that the feature grids can gain further refinement with its jointing learning with the NeRF branch. Moreover, as the grid branch is also supervised with reconstruction loss, it enforces the grid branch to continue enriching its captured scene information where the PE input can focus on the missing high-frequency details.

3.3. Refined Grid Feature Planes from NeRF

Recall that the feature grid relies on bilinear interpolation on the ground feature plane to obtain a feature vector of points within a voxel. The mechanism can yield detailed reconstruction results given sufficiently high grid resolution, such that the finest variation in the scene can be recovered. However, learning a grid with matching resolution can be highly memory-consuming for large urban scenes, as indicated in [36]. Moreover, the grid feature lacks the incentive to capture accurate variations within a voxel with merely reconstruction loss on ground truth RGB. We therefore jointly optimize the feature plane and vector with NeRF to enhance the supervision signal for grid features with pointwise guidance from the supplied NeRF inputs. Another benefit NeRF brings is the global regularization on the independently optimized grid features. Fig. 1 and Fig. 3 show that grid-based methods suffer from noisy artifacts because of the lack of constraints on space continuity and semantic similarity. NeRF, on the contrary, uses a shared MLP for the entire scene space. We will later show that the rendered novel views interpreted from the grid branch can get largely improved after its joint training with the NeRF branch.

4. Experiments

4.1. Experimental Setup

Dataset. Our main experiments are conducted on the real-world urban scenes. The three scenes cover different urban environments including rural rubble site [62] (*Rubble*), university campus (*Campus*), and residential complexes (*Residential*). The camera poses are obtained from the photogrammetry software ContextCapture. Additional experiments and results on general scene datasets, alterative camera poses, and techniques for further improvements can be found in the supplementary and our webpage.

Baselines and Implementations. We compare the performance of our approach with 1) NeRF [34] applied on the whole scene; 2) Mega-NeRF [62] with 4 partitions; 3) TensoRF [9], which reduces the memory footprint of feature grid via low-rank tensor factorization and considered suitable for large-scene scenarios. For NeRF and Mega-NeRF, we accordingly adopted larger model with 12 layers and 256 hidden units. The highest frequency of position encoding is set to 2^{15}, inserted to NeRF model via skip connection at the $4, 6, 8, 10$ layers. We use hierarchical sampling during training with 64 coarse and 128 fine samples per ray. All NeRF models are optimized using Adam optimizer [21] with a learning rate that decayed exponentially from $5e^{-4}$ and a batch size of 2048 rays, trained for $150k$ iterations. For TensoRF, in accordance with our observation on large urban scenes discussed previously in Sec. 3.1, we evaluated the simplified version that factorizes a feature grid into an xy-plane matrix and z-axis vector components. $16/48$ components are used for density and appearance feature grid respectively. Starting from an initial low-resolution grid with 128^3 voxels, the grid gets upsampled to 1024^3 linearly in logarithmic space during training. The grid resolution along each dimension is scaled by the x, y, z dimensions. A small MLP with 2 fully connected layers of 128 hidden layers and ReLU activation is used as the color output head. Adam optimizer is adopted with initial leaning rate of 0.02 for tensor factors and 0.01 for the MLP decoder. The batch size is 4096. The model is trained for $100k$ iterations.

Our method takes the matching grid resolution as the highest resolution feature plane and $8/16$ components for density/appearance grid respectively, with another two at the downsampled $\times 4$ and $\times 16$ resolution. The MLP head for the grid branch is same as TensoRF. The NeRF branch use a 4 MLP layer without skip layer. The highest frequency of position encoding is also set to 2^{15}. Adam optimizer is adopted with initial learning rate of 0.02 for tensor factors and 0.01 for the MLP layers with batch size 4096. We pretrain the grid branch for the first $10k$ iterations and joint optimization for another $100k$ iterations, and the time fraction between two stages is roughly 1:4. We use weighted loss 1:1 for two branches in joint training.

Figure 3. Qualitative comparison between baselines and ours. On large urban scenes, MLP-based methods (NeRF) and (Mega-NeRF) suffer from severe blurry artifacts. Grid-based method (TensoRF) shows better results, but tend to produce noisy appearance with inaccurate shapes. The pre-trained multi-resolution grid feature (Ours, Grid Pretrain) improves over the single high-resolution methods, yet the results is still suboptimal. Our final model achieves photo-realistic quality compared with ground-truth images on novel views. While grid branch (Ours, Grid branch) and NeRF branch (Ours, NeRF branch) receive similar metric scores, it is preferred to render from the NeRF branch which has sharper details and smoother spatial continuity, especially when rendering long videos in practice.

4.2. Results Analysis

We report the performance of baselines and our method in Fig. 3 and Tab. 1 both qualitatively and quantitatively. A significant improvement can be observed in visual quality and across all metrics. Our method reveals sharper geometry and more delicate details than purely MLP-based approaches (NeRF and Mega-NeRF). Especially, due to the limited capacity and spectral bias of NeRF, it always fails to model rapid changes in geometry and color, such as vegetation and stripes on the playground. Even though geographically partitioning scenes into small regions slightly helps as shown in the Mega-NeRF baseline, the rendered results still

appear to be overly smoothed. On the contrary, with the guidance from the learned feature grid, NeRF's sampling space is effectively and drastically compressed to near scene surface. The density and appearance features sampled from the ground feature planes explicitly indicate the scene contents, as depicted in Fig. 3. Despite being less accurate, it already offers informative local geometry and texture, and encourages NeRF's PE to pick up the missing scene details.

Refined Ground Feature Planes. Grid-based methods often require explicitly imposed regularization, such as the total variation loss or L1 loss [9], to avoid noises in regions with fewer observations, otherwise the independently op-

Table 1. Quantitative comparison on three large urban scene datasets. We report PSNR(↑), LPIPS(↑) [70], SSIM(↓) metric on the test views. The **best** and <u>second best</u> results are highlighted.

Scene	Rubble			Campus			Residential		
Metric	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM
NeRF	21.659	0.541	0.547	22.283	0.560	0.509	18.548	0.622	0.401
Mega-NeRF	23.505	0.516	0.565	22.365	0.496	0.544	19.350	0.561	0.452
TensoRF	23.800	0.478	0.670	20.915	0.471	0.571	18.332	0.575	0.428
Ours (pretrain)	22.617	0.451	0.622	24.542	0.385	0.698	21.032	0.428	0.620
Ours (grid branch)	**25.467**	<u>0.213</u>	**0.780**	**25.505**	<u>0.174</u>	**0.767**	**24.372**	<u>0.142</u>	**0.807**
Ours (nerf branch)	<u>24.130</u>	**0.207**	<u>0.767</u>	<u>24.903</u>	**0.162**	<u>0.757</u>	<u>23.765</u>	**0.137**	<u>0.802</u>

(a) Density feature plane (pre-trained -> refined)   (b) Appearance feature plane (pre-trained -> refined)
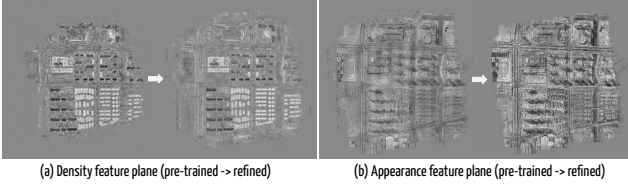
Figure 4. Visualization of one feature component in (a) density and (b) appearance feature plane (*Residential* scene). Compared to the pre-trained feature planes, the refined ones are less noisy; sharper edges and regular shapes of grouped objects can also be clearly identified. Since density and appearance features are independently learned, they encode different information that describes the scene. The appearance feature can capture environmental effects like shadows, as shown in (b).

timized grid features can easily result in fuzzy and wavy appearances, as illustrated in Fig. 3. By jointly optimizing with the NeRF branch, the xy-plane and z-axis encoding are constantly improved to encode more local details while becoming less noisy. A drastic improvement in fidelity can be observed in Fig. 6. Similar refinement can also be observed in feature space. Take one dimensional density plane on the *Residential* scene (Fig. 4) as an example, while a coarse floor layout of the targeting urban area can already be identified on the pre-trained xy-plane feature (Fig. 4(a)), it still missed details like sharp edges and varied shapes and colors widespread in the scene, which are hard to be represented with grids unless finer grid resolution is adopted. NeRF, on the other hand, searches for scene surfaces with points, which provides a more accurate and meaningful signal for grid feature optimization, and boosts it out of local minima. Another noticeable artifact on the pre-trained feature plane is the noise on continuous regions (*e.g.*, land, facade) due to grid variation, which is largely eased after jointly optimized with NeRF. This can be ascribed to NeRF's continuous representation of the scene, which imposes an implicit regularization on the feature grid by constructing a stronger correlation among coordinates. The resulting refined feature planes (Fig. 4(b)) exhibit smooth grid features with a cleaner silhouette where content-similar grids can be clustered together (*e.g.*, buildings, duplexes, and roads).

Compact Representations. While it is instinctive to design a heavy framework for modeling large scenes, our principle is to keep it compact and efficient without signif-
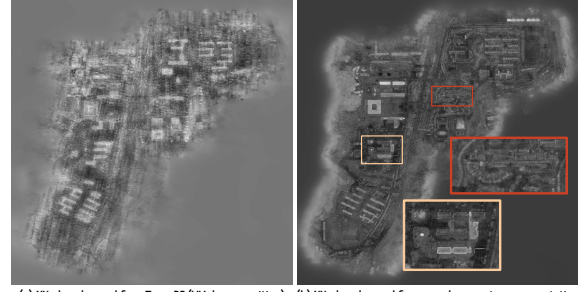

(a) XY-plane learned from TensoRF (VM decomposition)   (b) XY-plane learned from our plane-vector representation

Figure 5. Visualization of a slice of xy feature plane from (a) TensoRF's factorization; and (b) our ground plane representation. Our joint learning results in more accurate plane features with sharp region boundaries that is better aligned with the scene's physical ground plan, which is naturally more suitable for large urban scene modeling and downstream analysis. A cleaner feature grid also reveals that the learned latent space is more *compact*, which is critical for large-scale modeling even with limited model capacity.
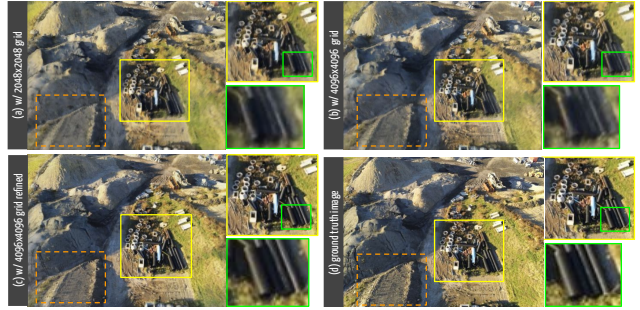

Figure 6. Qualitative comparison showing the rendering results using features learned (a) at a moderate grid resolution (2048^2), (b) at a high grid resolution (4096^2) and (c) from the re grid branch at resolution (4096^2). Despite higher grid resolution leads to better visual quality, adding NeRF supervision pushes the quality toward photorealistic one step further.

icantly reduced quality. Bearing that in mind, we modeled the full 3D feature grid with a succinct plane-vector representation. We demonstrate in Fig. 5 that, with similar performance on reconstructing large urban scenes (PSNR: (TensoRF) 21.075 vs. (Ours) 20.915), the 2D ground plane learned from TensoRF's VM decomposition [9] appears to be fuzzier and less informative than ours; and our representation uses less parameters ($3e8$) compared to TensoRF ($4e8$). Moreover, recall that grid resolution is critical for purely voxel-based representation to obtain high quality renderings, our method realizes photorealistic rendering of large scenes without further upsampling. Although supplying finer-grained feature planes to our framework is beneficial, the integration with NeRF largely alleviates the dependence on grid resolution to capture scene details. From NeRF's perspective, we show that a relative small MLP is sufficient to handle large scenes by taking the learned grid features with PE, and achieves superior result than the scaled-up NeRF and Mega-NeRF [62], as shown in Fig. 3.
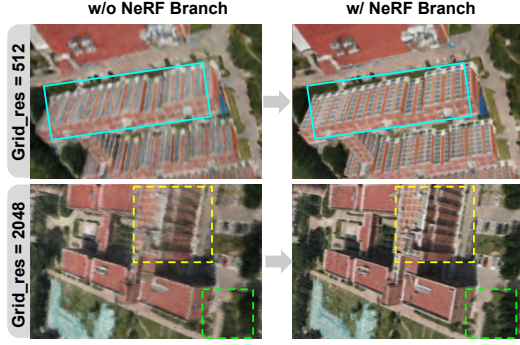
Figure 7. Using a single resolution grid feature results in inferior results. Low-resolution feature grid suffers from blurry artifacts and high-resolution grid gives noisy results. NeRF branch greatly helps remedy these issues with its point-wise supervision signal to pick up more details and global prior to regularize grid features.

4.3. Ablations

Ablations are conducted to verify the impact of 1) different model configurations: for the grid branch, we switch to single resolution feature grid at different resolutions; for NeRF branch, we inspect model capacity and the frequency bandwidth of PE in helping NeRF recover the scene details; Apart from the model architecture, we also look into 2) the efficacy of enriching NeRF's pure coordinated input with grid features; and 3) the efficacy of NeRF as a supervision signal to enhance feature grid.

Model configuration. For the grid branch, we show in Fig. 7 that adopting a single resolution feature grid leads to inferior performance. Concretely, results from a low resolution (512^2) grid branch already suffers from blurry artifacts during pre-training. Adding NeRF branch at the latter stage can help produce more details to the facade and rooftop but still lack sharp detail in general. On the other hand, results from a high resolution (2048^2) grid branch is fuzzy and noisy at the pre-training stage, which is eased by NeRF to a large extent, but still unstable at continuous regions, such as roads and walls. Existing works [9, 36, 53] usually adopt a small MLP as a renderer to translate grid features. In Fig. 8 We show that for our scenario, NeRF with a small model capacity (D=3, W=32) is insufficient to translate grid features of such complex scenes, giving inaccurate geometry and missing a large amount of scene details. Naively increasing frequency bandwidth in PE does little help under this circumstance. By enlarging the MLP (D=3, W=256), significant improvements can be observed, from which imposing higher frequency inputs via PE can help recover more scene details.

Efficacy of grid features to NeRF. We start by simply supplying NeRF with grid features without tuning grid features and supervision from the grid branch. NeRF can already benefit from the local features encoded in the grid features,
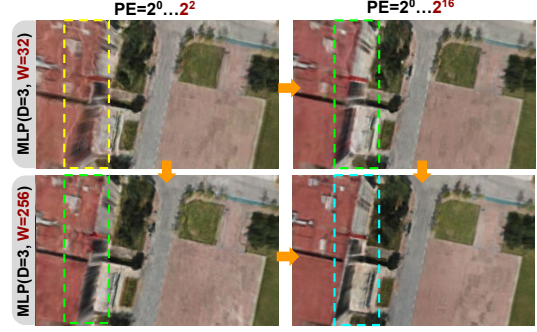

Figure 8. A small renderer is insufficient to translate the grid features, producing inaccurate geometry and scene contents. However, given enough modeling capacity, NeRF picks up more details with the help of increasingly higher PE frequency channels.

with $\sim 1db$ improvement in PSNR. Tuning the feature grid can further achieve $\sim 2.5db$ gain in PSNR.

Efficacy of NeRF supervision to feature grid. As depicted in Fig. 7, NeRF helps feature grids in recovering more details when grid resolution is inadequate, and smooth unregularized features with global prior to produce more consistent rendering results. On high-resolution grid, combining NeRF can raise PSNR by $\sim 2db$ on the *Campus* scene.

5. Discussion and Conclusion

In this work, we target on large urban scene rendering, and propose a novel framework that integrates the MLP-based NeRF with an explicitly constructed a feature grid to effectively encode both local and global scene informations. Our method overcomes various drawbacks of state-of-the-art methods when when applied to large-scale scenes. Our model achieves high visual fidelity rendering even for extremely large-scale urban scenes, which is crucial for real-world application scenarios.

While we mainly investigate the ground feature plane representation tailored for large urban scene scenario, our two-branch design can also be considered for other grid-based representation, serving as an additional regularization on the learned feature values by bringing more continuity. Still, our model inherits some limitations of NeRF-based methods, such as the slow training for our joint-learning stage. Another critical issue is dealing with a large amount of high-resolution images. The current batch sampling of shuffled rays is highly ineffective without distributed training. More discussions can be found in the supplementary.

References

[1] 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*, 2011. 2

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. *ICCV*, 2021. 2

[3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *arXiv preprint arXiv:2111.12077*, 2021. 1, 2

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *ECCV*, 2006. 2

[5] Alexander Bergman, Petr Kellnhofer, and Gordon Wetzstein. Fast training of neural lumigraph representations using meta learning. *Advances in Neural Information Processing Systems*, 34, 2021. 3

[6] Ilker Bozcan and Erdal Kayacan. Au-air: A multi-modal unmanned aerial vehicle dataset for low altitude traffic surveillance. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8504–8510, 2020. 1

[7] Rohan Chabra, Jan E Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *European Conference on Computer Vision*, pages 608–625. Springer, 2020. 2

[8] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. *arXiv preprint arXiv:2112.07945*, 2021. 3

[9] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. *arXiv preprint arXiv:2203.09517*, 2022. 1, 3, 4, 5, 6, 7, 8

[10] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2

[11] Zhang Chen, Yinda Zhang, Kyle Genova, Sean Fanello, Sofien Bouaziz, Christian Häne, Ruofei Du, Cem Keskin, Thomas Funkhouser, and Danhang Tang. Multiresolution deep implicit functions for 3d shape representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13087–13096, 2021. 3

[12] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6970–6981, 2020. 3

[13] Julian Chibane, Gerard Pons-Moll, et al. Neural unsigned distance fields for implicit function learning. *Advances in Neural Information Processing Systems*, 33:21638–21652, 2020. 3

[14] Terrance DeVries, Miguel Angel Bautista, Nitish Srivastava, Graham W Taylor, and Joshua M Susskind. Unconstrained scene generation with locally conditioned radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14304–14313, 2021. 2

[15] Dawei Du, Yuankai Qi, Hongyang Yu, Yi-Fan Yang, Kaiwen Duan, Guorong Li, W. Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. In *ECCV*, 2018. 1

[16] Emilien Dupont, Hyunjik Kim, SM Eslami, Danilo Rezende, and Dan Rosenbaum. From data to functa: Your data point is a function and you should treat it like one. *arXiv preprint arXiv:2201.12204*, 2022. 3

[17] Christian Früh and Avideh Zakhor. An automated method for large-scale, ground-based city model acquisition. *IJCV*, 2004. 2

[18] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 2

[19] Tal Hassner, Viki Mayzels, and Lihi Zelnik-Manor. On sifts and their scales. In *CVPR*, 2012. 2

[20] Peter Hedman, Pratul P Srinivasan, Ben Mildenhall, Jonathan T Barron, and Paul Debevec. Baking neural radiance fields for real-time view synthesis. *arXiv:2103.14645*, 2021. 3

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 5

[22] Rui Li, Darius RÜckert, Yuanhao Wang, Ramzi Idoughi, and Wolfgang Heidrich. Neural adaptive scene tracing. *arXiv preprint arXiv:2202.13664*, 2022. 3

[23] W. Li, C. W. Pan, R. Zhang, J. P. Ren, Y. X. Ma, J. Fang, F. L. Yan, Q. C. Geng, X. Y. Huang, H. J. Gong, W. W. Xu, G. P. Wang, Dinesh Manocha, and R. G. Yang. Aads: Augmented autonomous driving simulation using data-driven algorithms. *Science Robotics*, 4, 2019. 1

[24] Xiaowei Li, Changchang Wu, Christopher Zach, Svetlana Lazebnik, and Jan-Michael Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. *ECCV*, 2008. 2

[25] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021. 3

[26] Zhengqi Li, Wenqi Xian, Abe Davis, and Noah Snavely. Crowdsampling the plenoptic function. In *European Conference on Computer Vision*, pages 178–196. Springer, 2020. 2

[27] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *NeurIPS*, 2020. 2, 3

[28] Frank Losasso and Hugues Hoppe. Geometry clipmaps: terrain rendering using nested regular grids. *Siggraph*, 2004. 2

[29] David G Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 2

[30] Julien N. P. Martel, David B. Lindell, Connor Z. Lin, Eric Chan, Marco Monteiro, and Gordon Wetzstein. Acorn: Adaptive coordinate networks for neural scene representation. *ACM Trans. Graph.*, 40:58:1–58:13, 2021. 2

[31] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. *CVPR*, 2021. 2

[32] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019. 2, 3

[33] Moustafa Meshry, Dan B Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. Neural rerendering in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6878–6887, 2019. 2

[34] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020. 1, 2, 3, 4, 5

[35] Steven D. Morad, Roberto Mecca, Rudra P. K. Poudel, Stephan Liwicki, and Roberto Cipolla. Embodied visual navigation with automatic curriculum learning in real environments. *IEEE Robotics and Automation Letters*, 6:683–690, 2021. 1

[36] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv:2201.05989*, Jan. 2022. 1, 2, 3, 5, 8

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

[38] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 2

[39] Julian Ost, Fahim Mannan, Nils Thuerey, Julian Knodt, and Felix Heide. Neural scene graphs for dynamic scenes. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2855–2864, 2021. 1

[40] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019. 2, 3

[41] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 3

[42] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 523–540. Springer, 2020. 3

[43] Marc Pollefeys, David Nistér, J-M Frahm, Amir Akbarzadeh, Philippos Mordohai, Brian Clipp, Chris Engels, David Gallup, S-J Kim, Paul Merrell, et al. Detailed real-time urban 3d reconstruction from video. *IJCV*, 2008. 2

[44] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 3

[45] Eduard Ramon, Gil Triginer, Janna Escur, Albert Pumarola, Jaime Garcia, Xavier Giro-i Nieto, and Francesc Moreno-Noguer. H3d-net: Few-shot high-fidelity 3d head reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5620–5629, 2021. 3

[46] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. KiloNeRF: Speeding up neural radiance fields with thousands of tiny MLPs. *ICCV*, 2021. 3

[47] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, 2011. 2

[48] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. *CVPR*, 2016. 2

[49] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems*, 33:20154–20166, 2020. 3

[50] Qi Shan, Riley Adams, Brian Curless, Yasutaka Furukawa, and Steven M. Seitz. The visual turing test for scene reconstruction. *3DV*, 2013. 2

[51] Prafull Sharma, Ayush Tewari, Yilun Du, Sergey Zakharov, Rares Ambrus, Adrien Gaidon, William T Freeman, Fredo Durand, Joshua B Tenenbaum, and Vincent Sitzmann. Seeing 3d objects in a single image via self-supervised static-dynamic disentanglement. *arXiv preprint arXiv:2207.11232*, 2022. 2

[52] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: Exploring photo collections in 3d. *SIGGRAPH*, 2006. 2

[53] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. *arXiv preprint arXiv:2111.11215*, 2021. 3, 5, 8

[54] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Improved direct voxel grid optimization for radiance fields reconstruction. *arXiv preprint arXiv:2206.05085*, 2022. 3

[55] Towaki Takikawa, Joey Litalien, K. Yin, Karsten Kreis, Charles T. Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11353–11362, 2021. 2

[56] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. *arXiv preprint arXiv:2202.05263*, 2022. 1, 2, 3

[57] Matthew Tancik, Ben Mildenhall, Terrance Wang, Divi Schmidt, Pratul P Srinivasan, Jonathan T Barron, and Ren

Ng. Learned initializations for optimizing coordinate-based neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2846–2855, 2021. 3

[58] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*, 2020. 2, 4

[59] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Yifan Wang, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, et al. Advances in neural rendering. *arXiv preprint arXiv:2111.05849*, 2021. 1

[60] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. *International workshop on vision algorithms*, 1999. 2

[61] Joanne Truong, S. Chernova, and Dhruv Batra. Bi-directional domain adaptation for sim2real transfer of embodied navigation agents. *IEEE Robotics and Automation Letters*, 6:2634–2641, 2021. 1

[62] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. *arXiv preprint arXiv:2112.10703*, 2021. 1, 2, 3, 5, 7

[63] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 3

[64] Yuanbo Xiangli, Linning Xu, Xingang Pan, Nanxuan Zhao, Anyi Rao, Christian Theobalt, Bo Dai, and Dahua Lin. Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering. In *European Conference on Computer Vision*, 2022. 1, 2, 3, 4

[65] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. *arXiv preprint arXiv:2201.08845*, 2022. 3

[66] Yinghao Xu, Sida Peng, Ceyuan Yang, Yujun Shen, and Bolei Zhou. 3d-aware image synthesis via learning structural and textural representations. *arXiv preprint arXiv:2112.10759*, 2021. 3

[67] Zhenpei Yang, Yuning Chai, Dragomir Anguelov, Yin Zhou, Pei Sun, D. Erhan, Sean Rafferty, and Henrik Kretzschmar. Surfelgan: Synthesizing realistic sensor data for autonomous driving. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11115–11124, 2020. 1

[68] Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. *arXiv preprint arXiv:2112.05131*, 2021. 2, 3

[69] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5752–5761, 2021. 2, 3

[70] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7

[71] Xiaoshuai Zhang, Sai Bi, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Nerfusion: Fusing radiance fields for large-scale scene reconstruction. *arXiv preprint arXiv:2203.11283*, 2022. 3

[72] Siyu Zhu, Runze Zhang, Lei Zhou, Tianwei Shen, Tian Fang, Ping Tan, and Long Quan. Very large-scale global SFM by distributed motion averaging. *CVPR*, 2018. 2