GNeRF: GAN-based Neural Radiance Field without Posed Camera

Quan Meng[1] Anpei Chen[1] Haimin Luo[1] Minye Wu[1]
Hao Su[2] Lan Xu[1] Xuming He[1] Jingyi Yu[1]
[1] Shanghai Engineering Research Center of Intelligent Vision and Imaging
School of Information Science and Technology,
ShanghaiTech University [2] University of California, San Diego
{mengquan,chenap,luohm,wumy,xulan1,hexm,yujingyi}@shanghaitech.edu.cn {haosu}@eng.ucsd.edu

Abstract

We introduce GNeRF, a framework to marry Generative Adversarial Networks (GAN) with Neural Radiance Field (NeRF) reconstruction for the complex scenarios with unknown and even randomly initialized camera poses. Recent NeRF-based advances have gained popularity for remarkable realistic novel view synthesis. However, most of them heavily rely on accurate camera poses estimation, while few recent methods can only optimize the unknown camera poses in roughly forward-facing scenes with relatively short camera trajectories and require rough camera poses initialization. Differently, our GNeRF only utilizes randomly initialized poses for complex outside-in scenarios. We propose a novel two-phases end-to-end framework. The first phase takes the use of GANs into the new realm for optimizing coarse camera poses and radiance fields jointly, while the second phase refines them with additional photometric loss. We overcome local minima using a hybrid and iterative optimization scheme. Extensive experiments on a variety of synthetic and natural scenes demonstrate the effectiveness of GNeRF. More impressively, our approach outperforms the baselines favorably in those scenes with repeated patterns or even low textures that are regarded as extremely challenging before.

1. Introduction

Recovering 3D representations from multi-view 2D images is one of the core tasks in computer vision. Recently, significant progress has been made with the emergence of neural radiance fields methods (e.g., NeRF [31]), which represents a scene as a continuous 5D function and uses volume rendering to synthesize new views. Although NeRF and its follow-ups [6, 26, 29, 53, 61] achieve an unprecedented level of fidelity on a range of challenging scenes, most of these methods rely heavily on knowing the accurate camera poses, which is yet a long-standing but challenging task. The conventional camera pose estimation process suffers in challenging scenes with repeated patterns, varying lighting, or few keypoints, and building on these methods adds additional uncertainty to the NeRF training process.

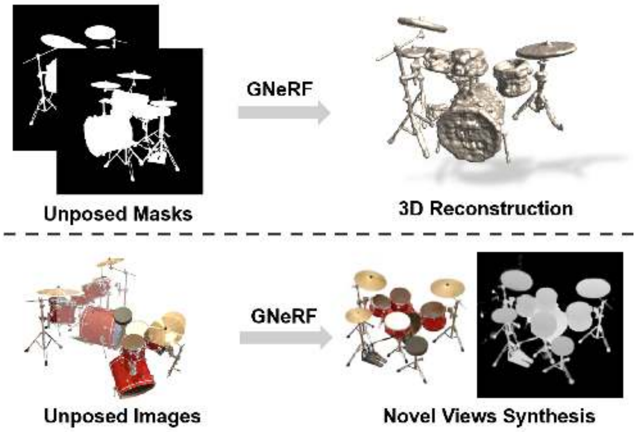

Figure 1. Our approach estimates both camera poses and neural radiance fields using only randomly initialized poses in complex scenarios, even in the extreme case when the input views are only texture-less gray masks.

To explore the possibilities of alleviating the dependence on accurate camera pose information, recently, iNeRF [60] and NeRF−− [55] attempt to optimize camera pose along with other parameters when training NeRF. While certain progress has been made, both of them can only optimize camera poses when relatively short camera trajectories with reasonable camera pose initialization are available. It is worth noting that, NeRF−− is limited to roughly forward-facing scenes, the focus of iNeRF is camera pose estimation but not radiance field estimation, and it assumes a trained NeRF which in turn requires known camera poses as supervision. When greater viewpoint uncertainty presents, camera poses estimation is extremely challenging and prone to falling into local minima.

To this end, we propose **GNeRF**, a novel algorithm that can estimate both camera poses and neural radiance fields when the cameras are initialized at random poses in complex scenarios. Our algorithm has two phases: the first phase gets coarse camera poses and radiance fields with adversarial training; the second phase refines them jointly with a photometric loss. Taking the use of Generative Adversarial Networks (GANs) into the realm of camera poses estimation, we extend the NeRF model to jointly optimize 3D representation and camera poses in complex scenes with large displacements. Instead of directly propagating the photometric loss back to the camera pose parameters, which is sensitive to challenging conditions (e.g., less texture and varying lighting) and apt to fall into local minima, we propose a hybrid and iterative optimization scheme. Our learning pipeline is fully differentiable and end-to-end trainable, allowing our algorithm to perform well in the challenging scenes where COLMAP-based [44] methods suffer from challenges such as repeated patterns, low textures, noise, even in the extreme cases when the input views are a collection of gray masks, as shown in Fig. 1. In addition, our method can predict new poses of images belonging to the same scene through the trained inversion network without tedious per-scene pose estimation (e.g., COLMAP-like methods) or time-consuming gradient-based optimization (e.g., iNeRF and NeRF−−). We experiment with our GNeRF on a variety of synthetic and natural scenes. We demonstrate results on par with COLMAP-based NeRF methods in regular scenes; more impressively, our method outperforms the baselines in cases with less texture that are regarded as extremely challenging before.

2. Related Works

Neural 3D Representations Classic approaches largely rely on discrete representations such as meshes [13], voxel grids [7, 49, 58], point clouds [10]. Recent neural continuous implicit fields are gaining increasing popularity, due to their capability of representing a high level of details [30, 39, 40]. But these methods need costly 3D annotations. To bridge the gap between 2D information and 3D representations, differential rendering tackles such integration for end-to-end optimization by obtaining useful gradients of the rendering process [18, 27, 31, 43, 48]. Liu *et al.* [27] proposes the first usage of neural implicit surface representations in differentiable rendering. Mildenhall *et al.* [31] proposes differentiable volume rendering and achieves more view-consistent reconstructions of the scene. However, they all assume accurate camera poses as a prerequisite.

Recently, several methods attempt to reduce dependence on precomputed camera poses. Adding noise to the ground-truth camera poses, IDR [59] produces accurate 3D surface reconstruction by simultaneously learning 3D representation and camera poses. Adding random offset to ground-truth camera poses, iNeRF [60] performs pose estimation by inverting a trained neural radiance field. Initializing camera poses to the identity matrix, NeRF−− [55] demonstrates satisfactory novel view synthesis results in forward-facing scenes by optimizing camera parameters and radiance field jointly. In contrast to these methods, our method does not depend on camera pose initialization and is not sensitive to challenging scenes with less texture and repeated patterns.

Pose Estimation Traditional techniques typically rely on Structured-from-Motion (SfM) [1, 11, 56, 44] which extracts local descriptor (e.g., SIFT [28]), performs matching to find 2D-3D correspondence, estimates candidate poses, and then chooses the best pose hypothesis by RANSAC [12]. Other retrieval-based methods [8, 16, 41, 47] find images similar to the query image and establish the 2D-3D correspondence efficiently by matching the query image against the database images. Recently, deep learning-based methods attempt to regress the camera pose directly from 2D images without the need of tracking. PoseNet [22] is the firstly end-to-end approach that adopts a modified truncated GoogleNet as pose regressor. Different architectures [35, 52, 57] or pose losses [3, 21] are utilized which lead to a significant improvement. Auxiliary tasks such learning relative pose estimation [51, 42] or semantic segmentation [42] lead to a further improvement. For a better generalization of the network, hybrid pose learning methods shift the learning towards local or related problems: [2, 25] propose to regress the relative pose of a query image to the known poses based on image retrieval.

These learning-based methods require large labeled training data, SSV [34] proposes to estimate viewpoints from unlabeled images via self-supervision. Although great progress has been made, it still needs abundant training images. Our method belongs to learning-based methods but is trained per scene in a self-supervised manner.

3D-Aware Image Synthesis Generative adversarial nets, or more generally the paradigm of adversarial learning, have led to significant progress in various image synthesis tasks [20, 32, 46]. But these methods operate on 2D space of pixels, ignoring the 3d structure of our natural scene. 3D-aware image synthesis correlates 3D model with 2D images, enabling explicit modification of 3D model [4, 5, 15, 36, 37, 38, 45]. Earlier 3D-aware image synthesis methods like RenderNet [36] introduce rendering convolutional networks with a projection unit that can render 2D images from 3D shapes. PLATONICGAN [15] uses a voxel-based representation and a family of differentiable rendering layers to discover the 3D structure of an object from an unstructured collection of 2D images. HoloGAN [37] introduces deep voxels representation and learns it also without any 3D shapes supervision. For these methods, the combination of differentiable rendering layers and implicit 3D representation can lead to entangled latent variables and destroy multi-view consistency. The most recent and relevant to ours are GRAF [45], GIRAFFE [38] and pi-GAN [4], with the expressiveness of NeRF, these methods allow disentangled shape, appearance modification of the generated objects.

However, these methods require abundant data and focus on simplistic objects (e.g., faces, cars) instead of photo-realistic and complex scenes. Conversely, our method can handle complex real scenes with limited data by learning a coarse generative network with limited data and refining it with photometric constraints.

3. Preliminary

We first introduce the basic camera and scene representation, as well as notations for our method in this section.

Camera Pose Formally, we represent the camera pose/extrinsic parameters based on its position/location in 3D space and its rotation from a canonical view. For the camera position, we simply adopt a 3D embedding vector in Euclidean space, denoted as $\mathbf{t} \in \mathbb{R}^3$. For the camera rotation, the widely-used representations such as quaternions and Euler angles are discontinuous and difficult for neural networks to learn. Following the seminal work [64], we use a continuous 6D embedding vector $\mathbf{r} \in \mathbb{R}^6$ to represent 3D rotations, which is more suitable for learning. Concretely, given a rotation matrix $\mathbf{R} = \begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3\end{bmatrix} \in \mathbb{R}^{3\times3}$, we compute the rotation vector $\mathbf{r}$ by dropping the last column of the rotation matrix.

From the 6D pose embedding vector, we can also recover the original rotation matrix using a Gram-Schmidt-like process, in which the last column is computed by a generalization of the cross product to three dimension [64].

NeRF Scene Representation We adopt the NeRF [31] framework to represent the underlying 3D scene and image formation, which encodes a scene as continuous volumetric radiance field of color and density. Specifically, given a 3D location $\mathbf{x} \in \mathbb{R}^3$ and 2D viewing direction $\mathbf{d} \in [-\pi, \pi]^2$ as inputs, the NeRF model defines a 5D vector-valued function $F_\Theta : (\mathbf{x}, \mathbf{d}) \to (\mathbf{c}, \sigma)$ based on an MLP network, where its outputs are an emitted color $\mathbf{c} \in \mathbb{R}^3$ and volume density σ, and Θ are network parameters. To render an image from a NeRF model, the NeRF model follows the classical volume rendering principles [19].

For each scene, the NeRF framework learns a separate neural representation network with a dataset of RGB images of the scene, the corresponding camera poses and intrinsic parameters, and scene bounds. Concretely, given a dataset of calibrated RGB images $\mathcal{I} = \{I_1, I_2, \cdots, I_n\}$ of a single scene, the corresponding camera poses $\Phi = \{\phi_1, \phi_2, \cdots, \phi_n\}$ and a differentiable volume renderer G, the NeRF model optimizes the continuous volumetric scene function F_Θ by a photometric loss as below,

$$\mathcal{L}_N(\Theta, \Phi) = \frac{1}{n}\sum_{i=1}^{n} \|I_i - \hat{I}_i\|_2^2, \quad \hat{I}_i = G(\phi_i; F_\Theta) \quad (1)$$

4. Methods

Our goal is to learn a NeRF model F_Θ from n uncalibrated images $\mathcal{I}$ of a single scene without knowing their camera poses. To this end, we treat the camera poses Φ of those images as values of a latent variable, and propose an iterative learning strategy that jointly estimates the camera poses and learns the NeRF model. As the overview of our approach in Fig. 2 illustrates, the key ingredient of our method is a novel NeRF estimation strategy based on an integration of an adversarial loss and an inversion network (Phase A). This enables us to generate a coarse estimate of the implicit scene representation F_Θ and the camera poses Φ from a learned inversion network. Given the initial estimate, we utilize photometric loss to refine the NeRF scene model and those camera poses (Phase B). Interestingly, our pose-free NeRF estimation process can also further improve the refined scene representation and camera poses. Additionally, we develop a regularized NeRF optimization step that refines the NeRF scene model and those camera poses. Consequently, our learning algorithm also iterates over the NeRF estimation and optimization step to further overcome local minima between the two phases (AB...AB).

In the following, we first present our pose-free NeRF estimation procedure in Sec 4.1, and then introduce the regularized and iterative NeRF optimization step in Sec 4.2. The training strategy is detailed in Sec 4.3 and model architecture is detailed in Sec 4.4.

4.1. Pose-free NeRF Estimation

As the initial stage of our method, in phase A, we do not have a reasonable camera pose estimation for each image or a pre-trained radiance field. Our goal for this stage is to predict a rough pose for each image and also learn a rough radiance field of the scene. As shown in the left part of Fig. 2, we use adversarial learning to achieve the goals. Our architecture contains two parts: a generator G and a discriminator D. Taking a random camera pose ϕ as input, the generator G will synthesize the image observed at the view by querying the neural radiance field and performing NeRF-like volume rendering. The set of synthesized images from many sampled camera poses will be decomposed into patches and compared against the set of real patches by the discriminator D. The fake and real patches are sampled via the dynamic patch sampling strategy which will be described in Sec 4.3. G and D are trained adversarially, as is done by the classical GAN work [14]. This adversarial

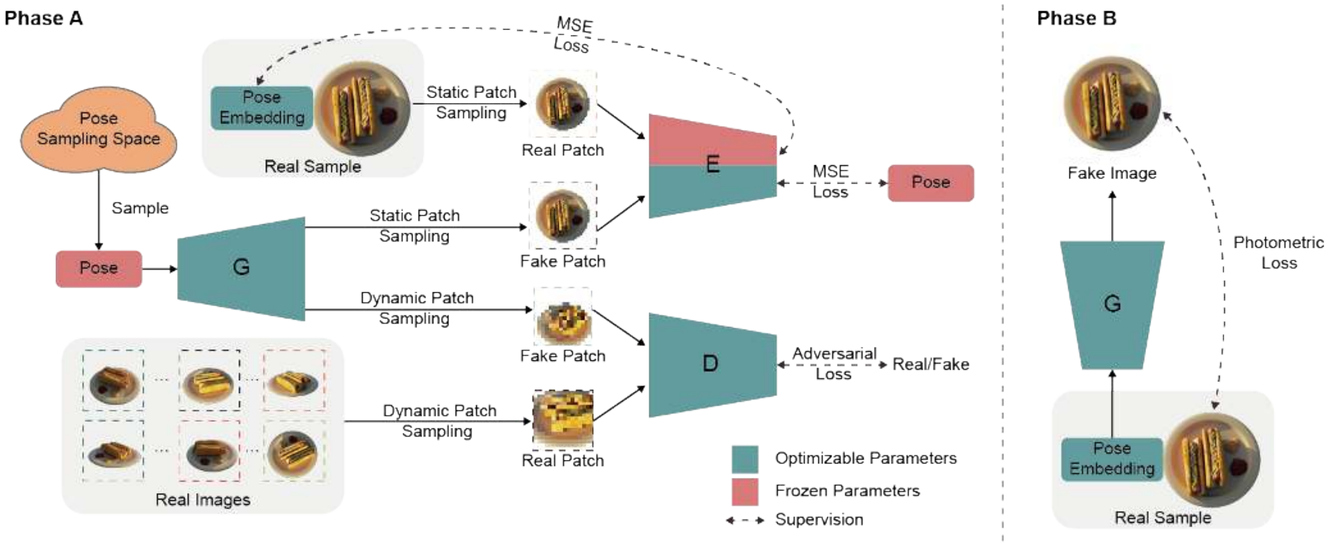

Figure 2. **The pipeline of GNeRF.** Our pipeline learns the radiance fields and camera poses jointly in two phases. In phase A, we randomly sample poses from a predefined poses sampling space and generate corresponding images with the NeRF (G) model. The discriminator (D) learns to classify real and fake image patches. The inversion network (E) takes in the fake image patches and learns to output their poses. Then, with the inversion network's parameters frozen, we optimize the pose embeddings of real images in the dataset. In phase B, we utilize the photometric loss to refine radiance fields and pose embeddings jointly. We follow a hybrid and iterative optimization strategy of the pattern 'A $\rightarrow$ AB. . . AB $\rightarrow$ B' in the training process.

training allows us to roughly learn the radiance field and estimate camera poses at random initialization.

Formally, we minimize a distribution distance between the real image patches $P_d(I)$ from the training set $\mathcal{I}$ and the generated image patches $P_g(I|\Theta)$, which are defined as below:

$$\Theta^* = \arg\min_{\Theta} Dist\left(P_g(I|\Theta)||P_d(I)\right) \qquad (2)$$

$$P_g(I|\Theta) = \int_{\phi} G(\phi; F_\Theta)P(\phi)d\phi \qquad (3)$$

To minimize the distribution distance, we adopt the following GAN learning framework based on an adversarial loss $\mathcal{L}_A$ defined as follows:

$$\begin{aligned}\min_{\Theta}\max_{\eta} \mathcal{L}_A(\Theta,\eta) =& \mathbb{E}_{I\sim P_d}[\log(D(I;\eta))] \\ &+\mathbb{E}_{\hat{I}\sim P_g}[\log(1-D(\hat{I};\eta))] \qquad (4)\end{aligned}$$

where η are the network parameters of the discriminator D and $\mathbb{E}$ denotes expectation.

Along with the two standard components, we train an inversion network E that maps image patches to the corresponding camera poses. We train the inversion network with the pairs of randomly sampled camera poses and generated image patches. The image patches are deterministically sampled from original images via a static sampling strategy which will be described in Sec 4.3. The inputs of the inversion network are these image patches, and the outputs are the corresponding camera poses. Formally, we denote the parameters of the inversion network E as θ_E, and its loss function can be written as,

$$\mathcal{L}_E(\theta_E) = \mathbb{E}_{\phi\sim P(\phi)}\left[\|E(G(\phi;F_\Theta);\theta_E)-\phi\|_2^2\right] \qquad (5)$$

We note that the inversion network is trained in a self-supervised manner, which exploits the synthetic image patches and their corresponding camera poses as the training data. With the increasingly better-trained generator, the inversion network would be able to predict camera poses for real image patches. After the overall training is converged, we apply the inverse network to generate camera pose estimates $\{\phi_i^{'} = E(I_i), I_i \in \mathcal{I}\}$ for the training set $\mathcal{I}$.

4.2. Regularized Learning Strategy

After the pose-free NeRF estimation step, we obtain an initial NeRF model and camera pose estimates for the training images. Due to the sparse sampling of the input image patches and the constrained capability of the inversion network, neither the NeRF representation nor the estimated camera poses $\Phi' = \{\phi_i^{'}\}$ are accurate enough. However, they provide a good initialization for the overall training procedure. This allows us to introduce a refinement step for the NeRF model and camera poses, phase B, as illustrated in the right part of Fig. 2. Specifically, this phase optimizes the pose embedding and the NeRF model by minimizing the photometric reconstruction error $\mathcal{L}_N(\Theta, \Phi)$ as defined in Eqn. 1.

We note that existing work like iNeRF and NeRF$--$ can search a limited scope in the pose space during NeRF optimization. However, the pose optimization problem in the standard NeRF model is highly non-convex, and hence their results strongly depend on camera pose initialization and are still insufficient for our challenging test scenarios. To mitigate this issue, we propose a regularized learning strategy (AB ... AB) by interleaving the pose-free NeRF estimation step (phase A) and the NeRF refinement step (phase B) to further improve the quality of the NeRF model and pose estimation. Such a design is based on our empirical findings that the pose-free NeRF estimation can also improve NeRF model and camera poses from the refinement step.

This strategy regularizes the gradient descent-based model optimization by the pose prediction from the learned inversion network. Intuitively, with the adversarial training of the NeRF model, the domain gap between synthesized fake images and true images is narrowing, so those pose predictions provide a reasonable and effective constraint for the joint radiance fields and pose optimization. Formally, we define a hybrid loss function $\mathcal{L}_R$ that combines the photometric reconstruction errors and an L2 loss penalizing the deviation from the predictions of the inversion network, which can be written as below,

$$\mathcal{L}_R(\Theta, \Phi) = \mathcal{L}_N(\Theta, \Phi) + \frac{\lambda}{n}\sum_{i=1}^{n} \|E(I_i; \theta_E) - \phi_i\|_2^2 \quad (6)$$

where λ is the weighting coefficient and $\mathcal{L}_N(\Theta, \Phi)$ is the photometric loss defined in Eqn. 1.

4.3. Training

Initially, we set all camera extrinsics to be an identity matrix. In phase A, we sample camera poses ϕ randomly from the prior pose distribution. In the Synthetic-NeRF dataset, the cameras are uniformly distributed at the upper hemisphere and towards the origin. In practice, we compute the rotation matrix directly from the camera position and the lookat point. In the DTU dataset, the cameras are uniformly distributed at the upper hemisphere with an azimuth range of $[0, 150]$, and the lookat point is distributed at a gaussian distribution $\mathcal{N}(0, 0.01^2)$. We analyze how the mismatch of prior pose distribution influences the performance in the supplemental material.

To train the generative radiance field, we follow a similar patch sampling strategy as GRAF [45] for computation and memory efficiency. Specifically, for the GAN training process, we adopt a dynamic patch sampling strategy, as is illustrated in the lower left part of Fig. 2. Each patch is sampled within the image domain with a fixed size of 16×16 but dynamic scale and random offset. For the pose optimization process, we adopt a static patch sampling strategy, as is illustrated in the upper left part of Fig. 2. Each patch is uniformly sampled across the whole image domain with a fixed size of 64×64. This sampling strategy uniquely represents the whole image with a sparse patch with which we estimate the corresponding camera pose. We also scale the camera intrinsics at the beginning to maximize the receptive field and progressively increase it to the original value to concentrate on fine details. In practice, these strategies bring great benefits to the stability of the GAN training process.

4.4. Implementation Details

We adopt the network architecture of the original NeRF [31] and its hierarchical sampling strategy to our generator. The numbers of sampled points of both coarse sampling and importance sampling are set to 64. Differently, because the GAN training only narrows the distribution of real patches and fake patches ("coarse" and "fine"), we utilize the same MLPs in hierarchical sampling strategy to ensure the pose spaces of "coarse" and "fine" networks are aligned. For a fair comparison, we increase the dimension of the MLPs from the original 256 to 360 to keep the overall parameters nearly unchanged. The discriminator network follows GRAF [45], in which instance normalization [50] over features and spectral normalization [33] over weights are applied. We borrow the Vision Transformer Network [9] to build our inversion network, whose last layer is modified to output a camera pose.

We use RMSprop [24] algorithm to optimize the generator and the discriminator with learning rates of 0.0005 and 0.0001, respectively. As for the inversion network and camera poses, we use Adam [23] algorithm with learning rates of 0.0001 and 0.005.

5. Experiments

Here we compare our method with other approaches which require camera poses or a coarse camera initialization on view synthesis task and evaluate our method in various scenarios. We run our experiments on a PC with Intel i7-8700K CPU, 32GB RAM, and a single Nvidia RTX TITAN GPU, where our approach takes 30 hours to train the network on a single scene.

5.1. Performance Evaluations

Novel View Synthesis Comparison We firstly compare novel view synthesis quality on the Synthetic-NeRF [31] and DTU [17] datasets with three other approaches: Original NeRF [31] with precalibrated camera poses from COLMAP [44], denoted by **C+n**; Original NeRF with precalibrated camera poses from COLMAP but jointly refined via gradient descent, denoted by **C+r**; Original NeRF with ground-truth camera poses, denoted by **G+n**. We report the standard image quality metrics Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [54] and Learned Perceptual Image Patch Similarity (LPIPS) [62] to evaluate image perceptual quality.

Data	Scene	↑ PSNR				↑ SSIM				↓ LPIPS			
		C+n	**C+r**	Ours	**G+n**	**C+n**	**C+r**	Ours	**G+n**	**C+n**	**C+r**	Ours	**G+n**
Synthetic-NeRF	Chair	33.75	32.70	31.30	32.84	0.97	0.95	0.94	0.97	0.03	0.05	0.08	0.04
	Drums	22.39	23.42	24.30	26.71	0.91	0.88	0.90	0.93	0.10	0.13	0.13	0.07
	Hotdog	25.14	33.59	32.00	29.72	0.96	0.97	0.96	0.95	0.05	0.03	0.07	0.04
	Lego	29.13	28.73	28.52	31.06	0.93	0.92	0.91	0.95	0.06	0.08	0.09	0.04
	Mic	26.62	31.58	31.07	34.65	0.96	0.97	0.96	0.97	0.04	0.03	0.06	0.02
	Ship	27.49	28.04	26.51	28.97	0.88	0.86	0.85	0.82	0.16	0.18	0.21	0.15
DTU	Scan4	22.05	24.23	22.88	25.52	0.69	0.72	0.82	0.78	0.32	0.20	0.37	0.18
	Scan48	6.718	10.40	23.25	26.20	0.52	0.62	0.87	0.90	0.65	0.60	0.21	0.21
	Scan63	27.80	26.61	25.11	32.19	0.90	0.90	0.90	0.93	0.21	0.19	0.29	0.24
	Scan104	10.52	13.92	21.40	23.35	0.48	0.55	0.76	0.82	0.60	0.59	0.44	0.36

Table 1. **Quantitative comparison among COLMAP-based NeRF [31] (C+n), COLMAP-based NeRF with additional refinement (C+r), NeRF with ground-truth poses(G+n), and ours on the Synthetic-NeRF [31] dataset and DTU [17] dataset.** We report PSNR, SSIM and LPIPS metrics to evaluate novel view synthesis quality. Our method without posed camera generates novel views on par with COLMAP-based NeRF and is more robust to challenging scene where COLMAP-based NeRF fails.

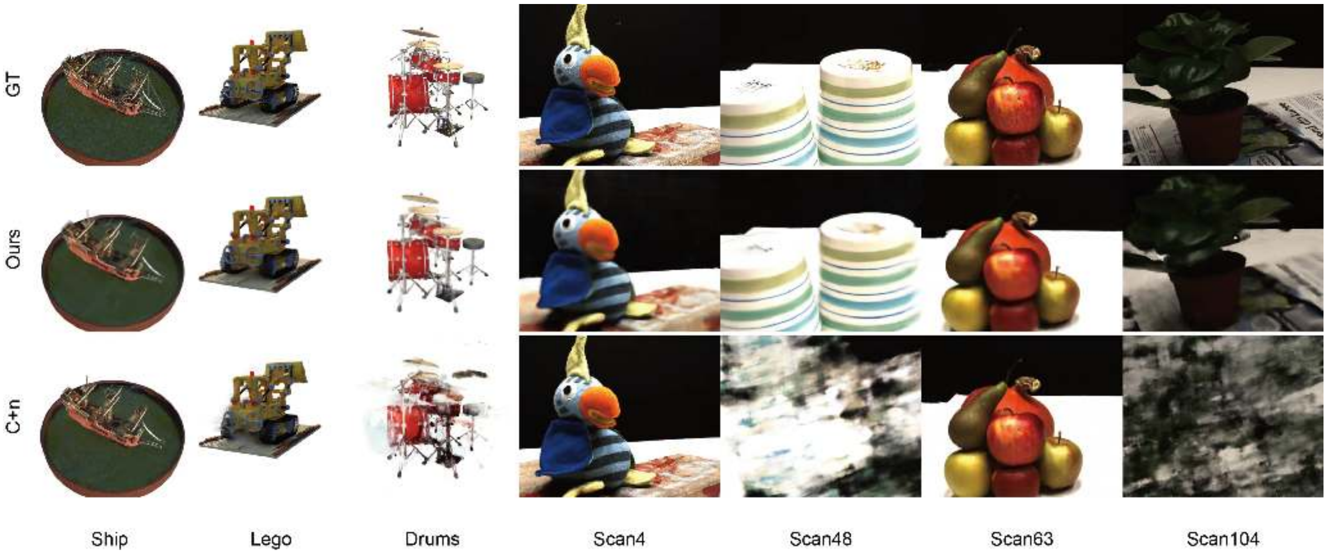

Figure 3. **Qualitative comparison between COLMAP-based NeRF (C+n) and ours on novel view synthesis quality on Synthetic-NeRF [31] dataset and DTU [17] dataset.** 'GT' means ground-truth images.

For evaluation, we need to estimate the camera poses of the test view images. Since our method can predict the poses of new images, the camera poses of the test view are directly estimated by our well-trained model. Conversely, for the COLMAP-based methods, we need to estimate the camera poses of images in the training set and test set together to keep them lie in the same space. We note that the COLMAP produces more accurate poses estimation with more input images, so for fair evaluation, we only choose a limited number of test images. The selection is based on maximizing their mutual angular distance between views so that test samples can cover different perspectives of the object as much as possible. For the Synthetic-NeRF dataset, we follow the same split as the original but randomly sample eight images from the test set for testing. The COLMAP is incapable to register the images with the resolution of 400×400 as shown in the supplement material, so 108 images of 800×800 are used for camera registration with which COLMAP performs much better. The training image resolution for all the methods is 400×400. For the DTU dataset, we use four representative scenes, on each of which we take every 8-th image as test images and take the rest 43 images for training. The input image resolution is 500×400. The scene selection is based on consideration of diversity: synthetic scenes (Synthetic-NeRF); real scenes with rich texture (scan4 and scan63); real scenes with less texture (scan48 and scan104).

As in Tab. 1, We also show the quantitative performance

Methods	Scan48	Scan97	Scan104
IDR(masked) [59]	21.17	17.42	12.26
Ours(masked)	20.40	19.40	19.81
Ours	25.71	24.52	25.70

Table 2. **Quantitiative rendering quality comparison between IDR and ours on DTU [17] dataset.** The evaluation metric is PSNR.

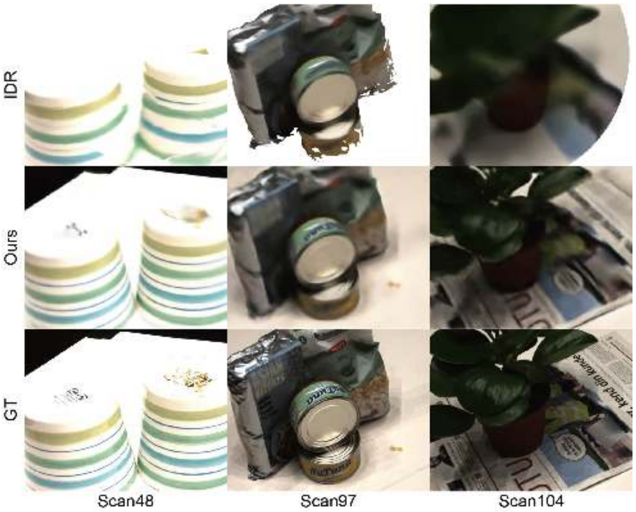

Figure 4. **Qualitative rendering quality comparison between IDR [59] and ours on DTU dataset.**

of all the three methods on the Synthetic-NeRF and DTU datasets. We notice that our method outperforms the **C+n** in scenes (drums, hotdog, mic, ship, scan48, and scan104) without enough reliable keypoints. **C+r** has a better performance than **C+n**'s. However, limited by the poor pose initialization, **C+r** can not produce the same performance as ours in some challenging scenes (scan48 and scan104). For other scenes, our method generates satisfactory results on par with the COLMAP-based NeRF methods. As in Fig. 3, We also show the visualization comparison. Our method outperforms the **C+n** in those challenging scenes while achieving similar results on regular scenes with enough keypoints. These challenging scenes do not have enough keypoints for pose estimation, so make NeRF which needs precise poses as input fail to synthesis good results. Conversely, our method optimizes the pose and radiance fields jointly by learning the global appearance distribution, so does not rely on texture or keypoints.

Additionally, to further demonstrate our architecture's ability to learn the high-quality 3D representation without camera poses, we also compare with the state-of-the-art 3D surface reconstruction method, IDR [59], by comparing the rendering quality. Note that the IDR method requires image masks and noisy camera initializations, while our method does not need them. We follow the same setting of optimizing the model and camera extrinsics jointly on 49 training images of each scene and report the mean PSNR as evaluation metrics. We report the PSNR computed on the whole image and within the mask, which is the same evaluation protocol as IDR. The qualitative and quantitative results are in Tab. 2 and Fig. 4. It can be seen that our volume-rendering-based method produces more natural images, while IDR produces results with more artifacts and fewer fine details.

Scene	COLMAP [44]		Ours	
	↓ Rot(deg)	↓ Trans	↓ Rot(deg)	↓ Trans
Chair	0.119	0.006	0.363	0.018
Drums	9.985	0.522	0.204	0.010
Hotdog	0.542	0.024	2.349	0.122
Lego	7.492	0.332	0.430	0.023
Mic	0.746	0.047	1.865	0.031
Ship	0.191	0.010	3.721	0.176

Table 3. **Quantitative camera poses accuracy comparison between COLMAP and ours on Synthetic-NeRF [31] dataset.** We report the mean camera rotation difference (Rot) and translation difference (Trans) over the training set.

Camera Poses Comparison We evaluate the accuracy of camera poses estimation on the Synthetic-NeRF dataset which contains several relatively challenging scenes with repeated patterns or less texture. The camera model of COLMAP is SIMPLE PINHOLE with shared intrinsics, $f = 1111.111$, $cx = 400$, $cy = 400$. For COLMAP, the input image size is 800×800 and the number is 108, while for our method, the input image size is 400×400 and the number is 100. We note that COLMAP produces more accurate estimates with more input images. In Tab. 3, we report the mean translation and rotation difference on the training set computed with the ATE toolbox [63]. Our method outperforms the COLMAP [44] on the drums and lego scenes which have less texture and repeated patterns. However, on the other scenes, which still contain enough reliable keypoints, our method is not accurate as the COLMAP.

5.2. Ablation Study

In Tab. 4 and Fig. 5, we show an ablation study over different components of our model. Our full architecture of the combination of adversarial training, inversion network, and photometric loss achieves the best performance. Without either the adversarial loss or the inversion network, the model is incapable to learn correct geometry, as illustrated in the depth map; without the photometric loss, the model is only capable to get coarse radiance fields.

In Tab. 5 and Fig. 6, we analyze different optimization schemes. We represent Phase A and Phase B as A and B respectively. Our adopted iterative optimization scheme on

Adver	Inver	Photo	↑ PSNR	↓ Rot(deg)	↓ Trans
	✓	✓	19.31	108.22	2.53
✓		✓	13.82	132.85	3.05
✓	✓		20.60	5.91	0.24
✓	✓	✓	31.30	0.36	0.02

Table 4. **Ablation study.** We report PSNR, camera rotation difference (Rot), and translation difference (Trans) of the full model (the last row) and three configurations by removing the adversarial loss (Adver), the inversion network (Inver), and the photometric loss (Photo), respectively. Removing adversarial loss and inversion network prevents the model from learning reasonable camera poses. Removing photometric loss prevents the model from getting accurate camera poses.

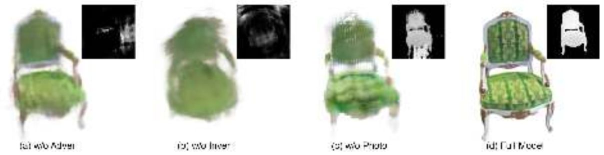
(a) w/o Adver (b) w/o Inver (c) w/o Photo (d) Full Model

Figure 5. **Ablation study.** We visualize novel view RGB images and depth maps of the four different configurations.

A, B	A, AB...AB, B	↑ PSNR	↓ Rot(deg)	↓ Trans
✓		29.23	0.592	0.034
	✓	31.30	0.363	0.018

Table 5. **Optimization schemes analysis.** We compare two optimization schemes: ‘A, B’ and ‘A, AB...AB, B’. The additional iterative optimization step enables our model to achieve much better results.

.

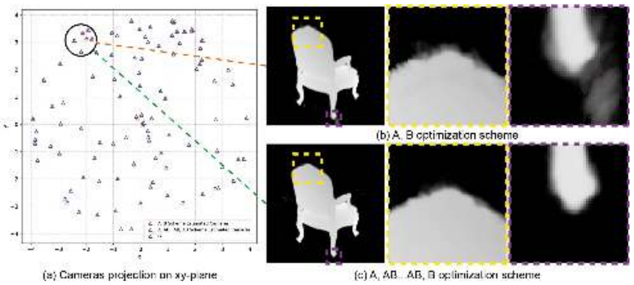
(a) Cameras projection on xy-plane (b) A, B optimization scheme (c) A, AB...AB, B optimization scheme

Figure 6. **Optimization schemes analysis.** On the left, we visualize the projection of camera poses on xy-plane of the obtained image from the two optimization schemes. On the right, we show depth maps of the view in the circled camera region and two detailed parts (yellow and purple insets) of them.

the pattern ‘A, AB...AB, B’ achieves much higher image quality and camera pose accuracy than that of ‘A, B’. In Fig. 6, the iterative optimization scheme gets much finer geometry along the edge, and the estimated camera poses align much closer to the ground-truth camera poses. These results demonstrate that the iterative learning strategy can further help overcome local minima.

6. Discussion and Conclusion

Discussion First, our method does not depend on camera pose initialization, but it does require a reasonable camera pose sampling distribution. For different datasets, we rely on a camera sampling distribution not far from the true distribution to alleviate the difficulties for radiance field and pose estimation. This could potentially be mitigated by learning the underlying pose sampling space automatically. A promising future direction would be combining global appearance distribution optimization (our approach) and local feature matching (pose distribution estimator) for the appearance and geometric reconstruction in an end-to-end manner. This combination potentially preserves our capability to challenging cases and relax to more general scenes without accurate distribution prior. Second, jointly optimizing camera poses and scene representation is a challenging task and opt to fall in local minima. Although in real datasets, we achieve good novel view synthesis quality on par with NeRF if the accurate camera poses are present, our optimized camera poses are still not so accurate as of the COLMAP when there are sufficient amount of reliable keypoints. This may be due to that our inversion network, which maps images to camera poses, could only take in image patches with limited size for computation efficiency. This might be fixed by importance sampling.

Conclusion We have presented GNeRF, a GAN-based framework to reconstruct neural radiance fields and estimate camera poses when the camera poses are completely unknown and scene conditions can be complicated. Our framework is fully differentiable and end-to-end trainable. Specifically, our first phase enables GAN-based joint optimization for the 3D representation and the camera poses, and our hybrid and iterative scheme by interleaving the first and second phases would further refine the results robustly. Extensive experiments demonstrate the effectiveness of our approach. Impressively, our approach has demonstrated promising results on those scenes with repeated patterns or even less texture, which have been regarded as extremely challenging before. We believe our approach is a critical step towards the more general neural scene modeling goal using less human-crafted priors.

Acknowledgements

We would like to thank the anonymous reviewers for their detailed and constructive comments which were helpful in refining the paper. This work was supported by NSFC programs (61976138, 61977047), the National Key Research and Development Program (2018YFB2100500), STCSM (2015F0203-000-06) and SHMEC (2019-01-07-00-01-E00003)

References

[1] Alex M Andrew. Multiple view geometry in computer vision. *Kybernetes*, 2001. 2

[2] Vassileios Balntas, Shuda Li, and Victor Prisacariu. Relocnet: Continuous metric learning relocalisation using neural nets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2

[3] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[4] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3

[5] Anpei Chen, Ruiyang Liu, Ling Xie, and Jingyi Yu. A free viewpoint portrait generator with dynamic styling. *ACM Transactions on Graphics*, 2021. 2

[6] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. *arXiv preprint arXiv:2103.15595*, 2021. 1

[7] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. 2

[8] Ondrej Chum, James Philbin, Josef Sivic, Michael Isard, and Andrew Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2007. 2

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. 5

[10] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[11] Olivier Faugeras, Quang-Tuan Luong, and Theo Papadopoulo. *The geometry of multiple images: the laws that govern the formation of multiple images of a scene and some of their applications*. MIT press, 2001. 2

[12] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981. 2

[13] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh r-cnn. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2014. 3

[15] Philipp Henzler, Niloy J Mitra, and Tobias Ritschel. Escaping plato's cave: 3d shape from adversarial rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[16] Arnold Irschara, Christopher Zach, Jan-Michael Frahm, and Horst Bischof. From structure-from-motion point clouds to fast location recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 2

[17] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 5, 6, 7

[18] Zhang Jiakai, Liu Xinhang, Ye Xinyi, Zhao Fuqiang, Zhang Yanshun, Wu Minye, Zhang Yingliang, Xu Lan, and Yu Jingyi. Editable free-viewpoint video using a layered neural representation. In *ACM Transactions on Graphics*, 2021. 2

[19] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM Transactions on Graphics*, 1984. 3

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[21] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[22] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2015. 2

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 5

[24] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2013. 5

[25] Zakaria Laskar, Iaroslav Melekhov, Surya Kalia, and Juho Kannala. Camera relocalization by computing pairwise relative poses using convolutional neural network. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017. 2

[26] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 1

[27] Shichen Liu, Shunsuke Saito, Weikai Chen, and Hao Li. Learning to infer implicit surfaces without 3d supervision. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2019. 2

[28] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004. 2

[29] 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. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

[30] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[31] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2020. 1, 2, 3, 5, 6, 7

[32] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2

[33] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2018. 5

[34] Siva Karthik Mustikovela, Varun Jampani, Shalini De Mello, Sifei Liu, Umar Iqbal, Carsten Rother, and Jan Kautz. Self-supervised viewpoint learning from image collections. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[35] Tayyab Naseer and Wolfram Burgard. Deep regression for monocular camera-based 6-dof global localization in outdoor environments. In *International Conference on Intelligent Robots and Systems (IROS)*, 2017. 2

[36] Thu Nguyen-Phuoc, Chuan Li, Stephen Balaban, and Yong-Liang Yang. Rendernet: A deep convolutional network for differentiable rendering from 3d shapes. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2018. 2

[37] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[38] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3

[39] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[40] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[41] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. 2

[42] Noha Radwan, Abhinav Valada, and Wolfram Burgard. Vlocnet++: Deep multitask learning for semantic visual localization and odometry. *IEEE Robotics and Automation Letters*, 3(4):4407–4414, 2018. 2

[43] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[44] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 5, 7

[45] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 2, 3, 5

[46] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. Singan: Learning a generative model from a single natural image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[47] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2003. 2

[48] Guoxing Sun, Xin Chen, Yizhang Chen, Anqi Pang, Pei Lin, Yuheng Jiang, Lan Xu, Jingya Wang, and Jingyi Yu. Neural free-viewpoint performance rendering under complexhuman-object interactions. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2021. 2

[49] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017. 2

[50] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 5

[51] Abhinav Valada, Noha Radwan, and Wolfram Burgard. Deep auxiliary learning for visual localization and odometry. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2018. 2

[52] Florian Walch, Caner Hazirbas, Laura Leal-Taixe, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. Image-based localization using lstms for structured feature correlation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017. 2

[53] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

[54] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5

[55] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf −−: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 1, 2

[56] Changchang Wu. Towards linear-time incremental structure from motion. In *International Conference on 3D Vision (3DV)*, 2013. 2

[57] Jian Wu, Liwei Ma, and Xiaolin Hu. Delving deeper into convolutional neural networks for camera relocalization. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2017. 2

[58] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[59] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 2, 7

[60] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *International Conference on Intelligent Robots and Systems (IROS)*, 2021. 1, 2

[61] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 1

[62] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5

[63] Zichao Zhang and Davide Scaramuzza. A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry. In *International Conference on Intelligent Robots and Systems (IROS)*, 2018. 7

[64] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3