DP-NeRF: Deblurred Neural Radiance Field with Physical Scene Priors

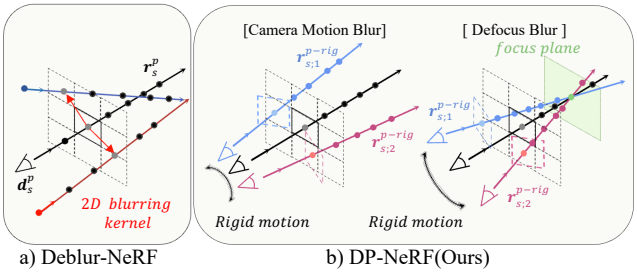
Dogyoon Lee[1] Minhyeok Lee[1] Chajin Shin[1] Sangyoun Lee[1,2]
[1]Yonsei University
[2]Korea Institute of Science and Technology (KIST)
{nemotio, hydragon516, chajin, syleee}@yonsei.ac.kr

Abstract

Neural Radiance Field (NeRF) has exhibited outstanding three-dimensional (3D) reconstruction quality via the novel view synthesis from multi-view images and paired calibrated camera parameters. However, previous NeRF-based systems have been demonstrated under strictly controlled settings, with little attention paid to less ideal scenarios, including with the presence of noise such as exposure, illumination changes, and blur. In particular, though blur frequently occurs in real situations, NeRF that can handle blurred images has received little attention. The few studies that have investigated NeRF for blurred images have not considered geometric and appearance consistency in 3D space, which is one of the most important factors in 3D reconstruction. This leads to inconsistency and the degradation of the perceptual quality of the constructed scene. Hence, this paper proposes a DP-NeRF, a novel clean NeRF framework for blurred images, which is constrained with two physical priors. These priors are derived from the actual blurring process during image acquisition by the camera. DP-NeRF proposes rigid blurring kernel to impose 3D consistency utilizing the physical priors and adaptive weight proposal to refine the color composition error in consideration of the relationship between depth and blur. We present extensive experimental results for synthetic and real scenes with two types of blur: camera motion blur and defocus blur. The results demonstrate that DP-NeRF successfully improves the perceptual quality of the constructed NeRF ensuring 3D geometric and appearance consistency. We further demonstrate the effectiveness of our model with comprehensive ablation analysis. [1] [2]

1. Introduction

The synthesis of the photo-realistic novel view image of complex three-dimensional (3D) scenes has advanced rapidly due to the emergence of the Neural Radiance Field(NeRF) [26]. NeRF has introduced implicit scene representation to the field, which maps an arbitrary continuous 3D coordinate to the volume density and radiance color using volume-rendering technique and implicit neural representation. NeRF densely reconstructs continuous 3D space to produce photorealistic rendered images with novel view.



a) Deblur-NeRF b) DP-NeRF(Ours)

Figure 1. (a) Deblur-NeRF models blurring kernel based on 2D offset on the image pixels. This modeling break the consistency in trained neural radiance field due to the lack of a 3D consistency priors. However, (b) DP-NeRF can render clean neural radiance field guaranteeing the 3D consistency with rigid motion of the camera based on the physical priors of the blur occurrence.

Though NeRF has achieved remarkable success in a variety of fields, most of the NeRF variants have been designed and tested for a carefully controlled environment that requires well-captured images from multiple views with calibrated camera parameters. However, various forms of noises are usually included in the data captured for the NeRF in real scenarios, complicating geometric and appearance consistency in 3D representation.

Several NeRF variants have attempted to reconstruct 3D scene in the presence of noise, including exposure noise [8, 11, 25], motion [17–19, 29, 30, 33, 49, 55], illumination changes [6, 23, 57], and aliasing [1, 2]. However, although it frequently occurs in real-world settings, blur has not been sufficiently addressed to date, despite the fact that it generates critical artifacts in 3D scene reconstruction. Deblur-NeRF [22] introduced blurring kernel estimation for a NeRF by imitating in-camera blurred image acquisition based on a blind deblurring method. Their method demonstrated excellent performance and produced clearly rendered images from multi-view images. However, the blurring kernel in [22] is implemented by optimizing ray deformation and composition weights depending on the 2D

[1]Code: https://github.com/dogyoonlee/DP-NeRF
[2]Project: https://dogyoonlee.github.io/dpnerf/

pixel location independently, leading to insufficient 3D information. In reality, the blurring process occurs simultaneously for all pixels in an image due to the physical process of in-camera image acquisition, but [22] overlooks the prior for blurring, leading to a lack of consistency in an image. Moreover, the designed kernel can be inherently optimized to subobtimal in regions with complex depth or similar appearance due to the independent optimization of the deformation of each ray. As a result, the estimated kernel has difficulty aggregating 3D information in a way that guarantees geometric and appearance consistency.

In this paper, we propose a deblurred NeRF based on two physical scene priors (hereafter, **DP-NeRF**) with a novel rigid blurring kernel (RBK) and adaptive weight proposal (AWP). The RBK consists of rigid ray transformation (RRT) and coarse composition weights (CCW), which utilize explicit physical scene priors derived from the blurring process to construct a consistent 3D scene representation from blurred images. In addition, the AWP proposes fine-grained color composition weights considering the relationship between depth and blur to create more realistic and clean 3D representation. Furthermore, we propose coarse-to-fine optimization for stable training and to gradually increase the effect of the AWP during training by introducing exponential weight decay between the two losses from the RBK and AWP. Figure 1 summarizes the DP-NeRF's system using the rigid motion of the camera.

The RBK generates a 3D deformation field and coarse weights for color composition based on the view information for each scene regardless of the pixel for each ray. This architecture is inspired by the physical scene prior that the blurring process consistently occurs for all pixels for a specific view. Specifically, the deformation field is constructed as the 3D rigid motion of the camera for each view and does not depend on the 2D spatial position of each ray. In contrast to Deblur-NeRF [22], our model successfully models 3D space with consistent geometry and appearance due to the use of these conditional physical priors and not fully depending on 2D pixel-wise independent ray optimization.

Previous studies have claimed that color composition process in a blurring kernel are affected by the depth values of the pixels when compositing blurred colors from both camera motion and defocus blur [43, 44]. Hence, RBK can lose detail in regions that have a complex depth or similar textures even though it achieves remarkably realistic 3D scene. For this reason, the AWP refines the composition weights using feature modulation (FM) [59] and novel motion feature aggregation module(MAM) based on the depth features of samples for transformed rays, the viewing direction, and the view information. Following the [22], we jointly optimize the RBK, AWP, and sharp NeRF with only the reconstruction loss from the blurred input as supervision. During inference stage, we can clearly render a reconstructed 3D scene using only the trained sharp NeRF model.

The rest of the paper is structured as follows. In Section 3, we describe the RBK and AWP in detail. In Section 4.1 and supplementary material, we provide experimental results for novel view synthesis using synthetic and real scene datasets with two types of blur that are provided from [22]. The results show that DP-NeRF achieves significant quantitative and qualitative improvement, preserving 3D consistency with a cleanly rendered novel view. In addition, we extensively analyze the effectiveness of the proposed model in Section 4.2. We also demonstrate how the RBK approximately models the blurring process in the supplementary material. To summarize, this paper offers the following major contributions.

- *Rigid blurring kernel.* We propose a novel RBK to construct a clean NeRF from blurred images utilizing physical scene priors derived from the blurring process during image acquisition.
- *Adaptive weight proposal.* We propose an AWP to refine the composition weights in the RBK considering the relationship between depth and blur to generate more realistic results.
- *Coarse-to-fine optimization.* To fully utilize proposed methods in training, we propose coarse-to-fine optimization by applying exponential weight decay between the reconstruction loss from the RBK and AWP.
- *Significant improvement in perceptual quality.* DP-NeRF produces enhanced 3D scene representation with greater perceptual quality and clean photo-realistic rendered images.

2. Related work

NeRF under various conditions. NeRF has become widespread in computer vision and graphics tasks related to neural rendering, utilizing coordinate-based implicit neural representation (INR). Due to the success of the NeRF in neural rendering, several studies have applied NeRF to other areas such as dynamic scenes [17–19, 29, 30, 33, 49, 55], generative models [28, 38], relighting [3, 23, 32, 42], human avatars [31, 45, 58], and 3D reconstruction [47, 50]. However, few studies have been conducted under non-ideal conditions [1, 2, 8, 11, 22, 25]. Mip-NeRF [1] addresseed the aliasing issue of ray samples by introducing 3D conical frustum ray casting with integrated positional encoding. Mip-NeRF 360 [2] then extended Mip-NeRF [1] to unbounded 360-degree scenes using shrunken space parametrization and online distillation to improve its quality and efficiency. To address the inconsistent appearance and transient objects in the uncarefully collected images, NeRF-W [23] introduced appearance and transient latent codes to the NeRF. HDR-NeRF [8] and HDR-Plenoxel [11] modeled the high dynamic range (HDR) radiance, imitating the physical process of in-camera image acquisition. [8] modeled

camera response function with the exposure value for the NeRF and [11] modeled white balance function for Plenoxels [52]. Deblur-NeRF [22] explored a new area of research constructing a clean NeRF from blurred images, which regularly occur during image acquisition in real-scenario.

Image Deblurring. Blur can be categorized into four types: camera motion, defocus, moving object and mixed blur. Image deblurring aims to recover a sharp image from images degraded by these types of blur. The recovery process can be expressed as to solve the equation: $B = I * K$, where B, I, and K denote the blurred image, sharp image, and blurring kernel, respectively. Deblurring can be divided into two categories, non-blind and blind, whose difference is whether the blurring kernel is known or not. Recent studies have focused primarily on blind deblurring because the blurring kernel is typically unknown in real-scenarios.

Several traditional image deblurring techniques [9, 10, 14, 39] have been proposed based on maximum a posterior (MAP) estimation [20, 34] based on a prior condition derived from natural images as a form of regularization. [39] uses global and local image priors as two piece-wise continuous functions and local smoothness constraints, while [9] proposes generalized transparency to efficiently estimate the blur filter by selecting useful pixels based on new transparency map. [10, 14] propose a sparse prior derived from local color statistics and a regularization function as the ratio of the l1-norm to the l2-norm for the high frequencies of an image, respectively.

Recently, deep image deblurring has been investigated following the success of deep learning networks in computer vision field. In this approach, a general blurring kernel is usually employed and latent images constructed based on blind deblurring and a data-driven prior through network training with paired datasets. Several studies have been proposed the use of convolutional neural network(CNN) [27, 46, 48, 53, 54] and generative models [15, 16]. We focus on priors from traditional methods because physical priors are helpful for constructing a blurring kernel in a NeRF system.

NeRF from Blurred Images. Deblur-NeRF [22] models the blurring kernel with the NeRF imitating the blind deblurring to produce clean and sharp NeRF. In contrast to recent deblurring methods that operate in the image space, the target blur types are camera motion and defocus blur in a static scene. Motion blur is excluded as a separate problem that needs to be overcome because temporal inconsistency is another challenge in 3D reconstruction. In [22], blurring kernel is optimized based on the kernel points on the 2D image pixels and view-embedded information. The kernel is designed around transformed rays that penetrate the kernel points on the image plane and camera origin, which are independently optimized during the training. However, the kernel relies heavily on the training of the deep neural network without cues for geometric and appearance consistency in 3D scene representation, which leads to a lack of consistency in 3D scene. Our method focuses on this limitation and proposes a novel blurring kernel with two physical priors derived from physical process of blur acquisition and ray casting as a form of regularization for kernel estimation.

3. Deblurred Neural Radiance Field

In this section, we describe our process for constructing a clean NeRF given a set of blurred inputs. Initially, we model the RBK to use the blur consistency in an image as a physical scene priors(Section 3.2). To consider the relationship between depth and blur, we then model the AWP module(Section 3.3). Finally, we explain our loss function and coarse-to-fine optimization strategy for the training of DP-NeRF(Section 3.4). Overall process for DP-NeRF is summarized in Figure 2.

3.1. Preliminary

Neural Radiance Field (NeRF). NeRF [26] constructs a continuous, volumetric representation of a 3D scene based on INR. It uses a multi layer perceptron(MLP) to approximate the function

$$F : (\gamma_{\mathbf{x}}(\mathbf{x}), \gamma_{\mathbf{d}}(\mathbf{d})) \rightarrow (\mathbf{c}, \sigma), \tag{1}$$

which maps 3D position $\mathbf{x} = (x, y, z)$ and viewing direction $\mathbf{d} = (\phi, \theta)$ to a color $c = (r, g, b)$ and volume density σ.

Specifically, the 3D position and viewing direction are independently projected to a higher dimension by applying the sinusoidal positional encoding function $\gamma : \mathbb{R}^3 \rightarrow \mathbb{R}^{3+6m}$, which is defined as

$$\gamma(\mathbf{x}) = \left(\mathbf{x}, \dots, sin(2^f \pi \mathbf{x}), cos(2^f \pi \mathbf{x}), \dots\right), \tag{2}$$

where $f = \{0, \dots, m-1\}$ and m is a hyper-parameter that decides the frequency band. For clarity, we abbreviate the positional encoding and represent the NeRF as

$$F(\mathbf{x}, \mathbf{d}) = (\mathbf{c}, \sigma). \tag{3}$$

To train the NeRF with input images, the NeRF renders each color $\hat{C}^p$ of pixel p using a rendering technique [24] that is an approximated version of classical volume rendering [12]. For given ray origin $\mathbf{o}^p$ and viewing direction $\mathbf{d}^p$ along a pixel p, the i_{th} sample on the ray $\mathbf{r}^p$ is defined as $\mathbf{r}_i^p = \mathbf{o}^p + t_i \mathbf{d}^p$, where t_i is drawn from N evenly spaced bins with stratified sampling [26] in near-to-far bounded partition $[t_n, t_f]$ as shown in Eq. 4:

$$t_i \sim \mathcal{U}\left[t_n + \frac{i-1}{N}\left(t_f - t_n\right), t_n + \frac{i}{N}\left(t_f - t_n\right)\right]. \tag{4}$$

Pixel color $\hat{C}^p$ is computed from the predicted color $\mathbf{c}_i^p$ and density σ_i^p of each sample $\mathbf{r}_i^p$ as shown in Eq. 5:

$$\hat{C}(\mathbf{r}) = \sum_{i=1}^{N} w_i \mathbf{c}_i = \sum_{i=1}^{N} T_i \left(1 - exp(-\sigma_i \delta_i)\right) \mathbf{c}_i, \tag{5}$$

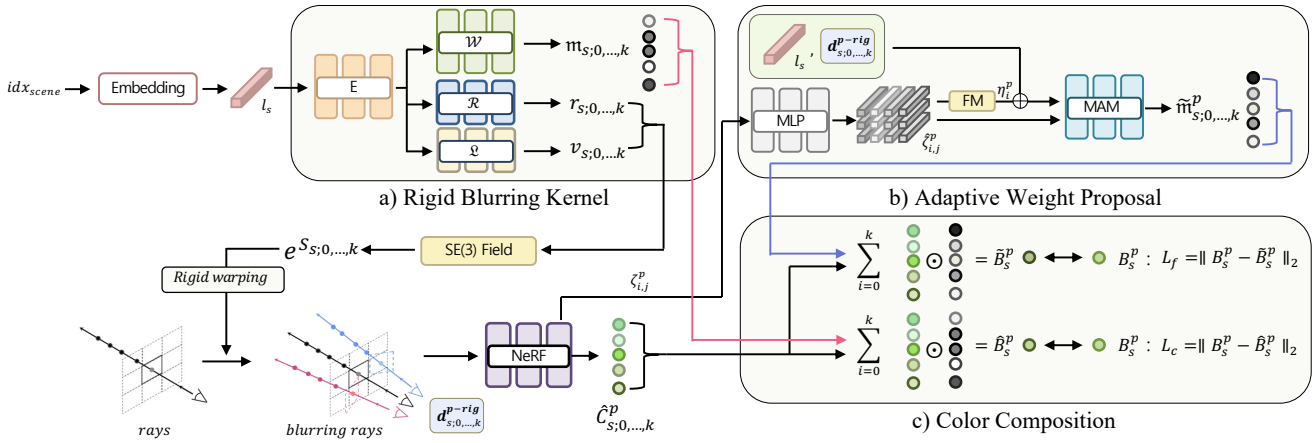


Figure 2. Overall pipeline for DP-NeRF. DP-NeRF consists of three stages. (a) The rigid blurring kernel (RBK) constructs the blurring system using the $SE(3)$ Field based on the physical priors. (b) The adaptive weight proposal (AWP) refines the composition weights using the depth feature ($\zeta^p_{i,j}$) of the samples on the ray of the target pixel (p), the scene (s) information, and the rigidly transformed ray directions ($\mathbf{r}^p_{s;0,\dots,k}$). (c) Finally the coarse and fine blurred colors, $\hat{B}^p$ and $\tilde{B}^p$, are composited using the weighted sum of the ray transformed colors. $\mathcal{L}_c$ and $\mathcal{L}_f$ denote the coarse and fine RGB reconstruction loss, respectively.

where $T_i = exp(-\sum_{j=1}^{i-1}\sigma_j\delta_j)$ is the transmittance and $\delta_i = t_{i+1} - t_i$ is the distance between adjacent samples. Note that, we abbreviate the notation for pixel p for clarity.

Blind Deblurring in the NeRF. Our goal is to solve the blind deblurring process with respect to sharp pixel color $\hat{C}^p$ and blurring kernel h^p in a similar manner to [22] as

$$\hat{B}^p = \hat{C}^p * h^p, \tag{6}$$

where $\hat{C}^p$ is computed from sample color $\mathbf{c}^p_i$ and density σ^p_i, which are predicted by the NeRF. [22] models kernel h^p by introducing approximated n sparse kernel points in $K \times K$ sized window $\mathcal{N}(p)$ at the 2D pixel location, which are optimized by the MLP. The rendered pixel colors from the kernel points are then composited by the corresponding weights w^p_q, which are also predicted by the MLP, as Eq. 7:

$$\hat{B}^p = \sum_{q\in\mathcal{N}(p)} w^p_q \hat{C}^p_q \tag{7}$$

The kernel points and weights for each pixel ray are optimized independently depending on the 2D spatial pixel coordinates and view-information. Thus, the blurring kernel is dynamic, but they overlook the importance of geometric and appearance consistency in overall 3D space.

3.2. Rigid Blurring Kernel (RBK)

Physical Scene Priors. As we mentioned in Section 3.1, deblurring in the NeRF should consider 3D consistency. To address this, we impose two priors as constraints inspired by the physical process of image blurring.

Prior 1: A blurred image is generated during in-camera image acquisition. The first prior shares ray rigid transformation (RRT) from camera motion through all of the pixels of a blurred image because same camera is used.

Prior 2: The blurring process for all of the pixels in a blurred image occurs simultaneously. The second prior shares the coarse composition weights (CCW) across all of the pixels of an image because the color composition of all of the pixels in a blurred image is affected simultaneously.

Ray Rigid Transformation. From the first prior, we mimic the blurring process of an image using RRT based on each image's view information. RRT is formulated as ray transformation derived from the deformation of rigid camera motion, defined as the dense $SE(3)$ field for scene s, approximated by the MLPs, which consists of shared encoder MLP E and independent MLPs ($\mathcal{R}$ and $\mathfrak{L}$), as shown in Eq. 8:

$$\mathcal{S}_s = \big(\mathcal{R}(E(l_s)); \mathfrak{L}(E(l_s))\big), \quad where \ \ s \in N_{img}, \tag{8}$$

where l_s denotes the latent code for each scene through the embedding layer [4], and N_{img} denotes a set of image indices. The scene-wise $SE(3)$ field can encode the rigid motion of the camera for the scene, thus consistently transforming the rays of the scene to imitate the blurring process. Inspired by Nerfies [29], rigid motion is encoded as the screw axis [21] $\mathcal{S}_s = (r_s; v_s) \in \mathbb{R}^6$, where $r_s \in \mathfrak{so}(3)$ encodes rotation. $\hat{r}_s = r_s/||r_s||$ is the axis of rotation and $\theta = ||r_s||$ is the angle of rotation. Rotation matrix $e^{r_s} \in SO(3)$ is taken from Rodrigues' formula [35]:

$$e^{r_s} \equiv e^{[r_s]} = \mathbf{I} + \frac{\sin\theta}{\theta}[r_s]_\times + \frac{1-\cos\theta}{\theta^2}[r_s^2]_\times, \tag{9}$$

where $[\mathbf{x}]_\times$ denotes the cross-product matrix of vector $\mathbf{x}$. The translation matrix, which is encoded by screw motion $\mathcal{S}_s$, is taken as $\mathbf{p}_s = \mathbf{G}_s v_s$, where

$$\mathbf{G}_s = \mathbf{I} + \frac{1-\cos\theta}{\theta^2}[r_s]_\times + \frac{\theta-\sin\theta}{\theta^3}[r_s]^2_\times. \tag{10}$$

For given ray $\mathbf{r}^p_s$ on arbitrary pixel p in scene s, we can define the RRT as

$$\mathbf{r}^{p-rig}_{s;q} = e^{\mathcal{S}^p_{s;q}}\mathbf{r}^p_s = e^{\mathcal{S}_{s;q}}\mathbf{r}^p_s = e^{r_{s;q}}\mathbf{r}^p_s + \mathbf{p}_{s;q}, \tag{11}$$

where $q \in \{1,\dots,k\}$, k is a hyper-parameter that controls the number of camera motions contributing to the blur in

scene s, and $\mathbf{r}_{s;q}^{p-rig}$ denotes the rigidly transformed (RT) ray from $\mathbf{r}_s^p$. Note that, $\mathcal{S}_{s;q}^p = \mathcal{S}_{s;q}$ due to our first prior. To ensure that, for $\mathfrak{se}(3)$, $e^{\mathcal{S}_{s;q}}$ is the identity when $\mathcal{S}_{s;q} = 0$, we initialize the weights of the last layer of MLP $\mathcal{R}$ from $\mathcal{U}(-10^{-5}, 10^{-5})$ following [29]. The transformed sharp colors $\hat{C}_{s;q}^{p-rig}$ are then rendered from $\mathbf{r}_{s;q}^{p-rig}$ using a volume-rendering technique to composite the blurry color $\hat{B}_s^p$ based on the CCW described in the following section.

Coarse Composition Weights. From the second prior, we model the CCW field for scene s from the MLP $\mathcal{W}$, which shares the encoding MLP E with other MLPs ($\mathcal{R}$ and $\mathfrak{L}$).

$$\mathfrak{m}_{s;0,\dots,k} = \sigma\left(\mathcal{W}(E(l_s))\right), \quad where \quad \sum_{i=0}^{k} \mathfrak{m}_{s;i} = 1, \quad (12)$$

where k denotes the number of camera motions shared with the RRT. $\mathfrak{m}_s$ is the CCW for scene s. σ represents the sigmoid function. Note that, the number of $\mathfrak{m}_s$ is $k+1$, where $\mathfrak{m}_{s;0}$ is the weight for original ray $\mathbf{r}_s^p$. Finally, blurry color $\hat{B}_s^p$ for pixel p in scene s, is composited by the weighted sum of the rendered colors of original ray $\hat{C}_{s;0}^p$ and RT rays $\hat{C}_{s;1,\dots,k}^{p-rig}$ using the corresponding per-scene CCW $\mathfrak{m}_{s;0,\dots,k}$ as shown in Eq. 13:

$$\hat{B}_s^p = \mathfrak{m}_{s;0}\hat{C}_{s;0}^p + \sum_{i=1}^{k} \mathfrak{m}_{s;i}\hat{C}_{s;i}^{p-rig} \quad (13)$$

The RBK pipeline is summarized in Figure 2 (a) and (c).

3.3. Adaptive Weight Proposal (AWP)

Relationship between Depth and Blur. Though the proposed RBK models the blurring kernel successfully with realistic rendering results (Figure 4), there is still room for improvement in the relationship between depth and blur. Several past studies have described the relationship between depth and each type of blur [43, 44]. Specifically, camera motion blur is affected by depth when the camera motion is out-of-plane [44], while defocus blur is usually affected by depth [43]. Therefore, we employ the AWP module to alleviate the color composition errors by flexibly refining the CCW $\mathfrak{m}_{s;0,\dots,k}$ along the depth features of the samples on the original and RT rays for pixel p. Further description for motivation of AWP is attached in supplementary material.

AWP Network. We define the AWP as a function that infers the per-pixel adaptive composition weights $\tilde{\mathfrak{m}}_{s;0,\dots,k}^p$ utilizing the depth features of each sample on the rays, corresponding to the latent code of each scene, and the direction of each ray. The function $\mathcal{AWP}$ is defined as Eq. 14:

$$\tilde{\mathfrak{m}}_{s;0,\dots,k}^p = \mathcal{AWP}(\zeta_{i=0,\dots,k;j=1,\dots,N}^p, \mathbf{d}_{i=0,\dots,k}^{p-rig}, l_s), \quad (14)$$

where i denotes the original and RT rays, and j denotes the sample on each ray. In addition, ζ denotes the corresponding depth feature and $\mathbf{d}_{0,\dots,k}^{p-rig}$ denotes the directions of original and RT rays. We then composite the adaptive blurred color $\tilde{B}_s^p$ through $\tilde{\mathfrak{m}}_{s;0,\dots,k}^p$ as shown in Eq. 15:

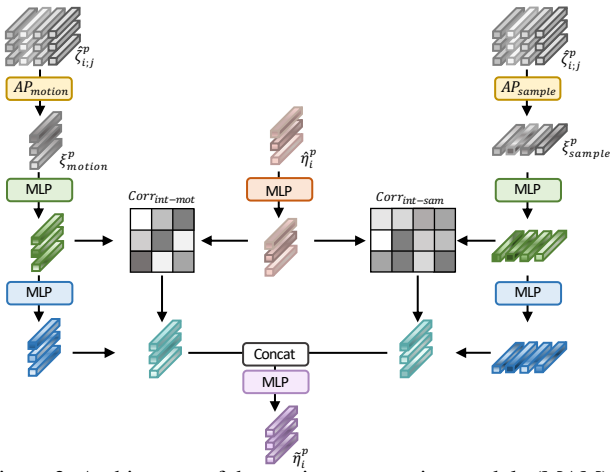


Figure 3. Architecture of the motion aggregation module (MAM).

$$\tilde{B}_s^p = \tilde{\mathfrak{m}}_{s;0}^p\hat{C}_{s;0}^p + \sum_{i=1}^{k} \tilde{\mathfrak{m}}_{s;i}^p\hat{C}_{s;i}^{p-rig} \quad (15)$$

We design $\mathcal{AWP}$ to be an approximation using a deep learning network (Figure 2 (b) and (c)).

Network Architecture. Inspired by CurveNet [51], the AWP generates the fine weights $\tilde{\mathfrak{m}}_{s;i}^p$ utilizing the inter-sample and inter-motion correlation of the rays transformed by the RBK, which are denoted as $Corr_{int-sam}$ and $Corr_{int-mot}$ in Figure 3, respectively. The depth features for each sample on the rays ($\zeta_{i;j}^p$) are extracted from second-to-last layer of the NeRF, which contains implicit occupancy information. First, we construct motion-wise modulated features, $\eta_{i=0,\dots,k}^p$, through feature modulation (FM) [59] from N samples on each ray as shown in Figure 2 (b) and Eq. 16:

$$\eta_i^p = \sum_{l=1}^{N}\left(\exp(-\sum_{m=1}^{l-1}\delta_m\hat{\zeta}_{i;m}^p \circ (1-\exp(-\delta_l\hat{\zeta}_{i;l}^p) \circ \hat{\zeta}_l^p)\right), \quad (16)$$

where $\hat{\zeta}_{i;j}^p$ denotes the embedded features of $\zeta_{i;j}^p$ from the simple MLP and δ is the same as in Eq. 5. Second, we add the viewing direction and scene-information by using a simple MLP with η_i^p, $\mathbf{d}_{i=0,\dots,k}^{p-rig}$, and l_s, as shown in Eq. 17:

$$\hat{\eta}_i^p = MLP\left(\eta_i^p, \gamma_{\mathbf{d}}(\mathbf{d}_i^{p-rig}), l_s\right), \quad where \ i \in \{0,\dots,k\}, \quad (17)$$

where $\hat{\eta}_i^p$ denotes the modulated features with the viewing information. We then forward ($\zeta_{i;j}^p$ and $\hat{\eta}_i^p$) to the MAM to aggregate the implicit depth-derived information based on attentive feature extraction as in Eq. 18:

$$\tilde{\eta}_i^p = MAM(\hat{\zeta}_{i;j}^p, \hat{\eta}_i^p), \quad (18)$$

where $\tilde{\eta}_i^p$ denotes the aggregated features.

For the MAM, we first generate motion-wise and sample-wise representative features, ξ_{motion}^p and ξ_{sample}^p, by forwarding the embedding MLP and employing attentive pooling [7] along each axis from the embedded features

$\hat{\zeta}^p_{i;j}$. Note that, we omit the embedding MLP in Figure 3 for clarity. Second, we compute the inter-motion and inter-sample correlation using CurveNet-like architecture via matrix multiplication. Third, the correlated features update each embedded features using matrix multiplication. Finally, aggregated features $\tilde{\eta}^p_i$ are extracted by concatenating the correlated embedded features and forwarding the simple MLP. The overall process for the MAM is described as Figure 3 and Eq. 19:

$$MAM(\hat{\zeta}^p_{i;j}, \hat{\eta}^p_i) = MLP\big(cat(\textbf{corr}(\hat{\eta}^p_i, \xi^p_{motion}, \xi^p_{sample}))\big), \tag{19}$$

where cat denotes the concatenating operation and **corr** represents computing operations of inter-motion and inter-sample correlation. To this end, the adaptive composition weights $\tilde{\mathfrak{m}}^p_{s;0,\dots,k}$ are predicted using global average pooling (GAP) along the motion axis and the linear layer as shown in Eq. 20. We omit the GAP and the final Linear layer for clarity in Figure 2 (b).

$$\tilde{\mathfrak{m}}^p_{s;i} = \sigma\big(Linear(GAP(\tilde{\eta}^p_{i=0,\dots,k}))\big), \ \ where \ \sum_{i=0}^{k} \tilde{\mathfrak{m}}^p_{s;i} = 1 \tag{20}$$

Details of the operation and notations for the feature dimensions are presented in the supplementary material.

3.4. Training & Optimization

Training Loss. DP-NeRF takes only the RGB reconstruction loss for the blurred color of a pixel with a corresponding ray because our goal is to approximate the blurred color of the pixel using the RBK and AWP. In contrast to DeblurNeRF [22], we employ two blurred colors $\hat{B}^p_s$ and $\tilde{B}^p_s$ to optimize the DP-NeRF. Our RGB reconstruction loss $\mathcal{L}_{recon}$ consists of two reconstruction losses from these predicted colors and the ground truth color as shown in Eq. 21:

$$\mathcal{L}_{recon} = ||B^p_s - \hat{B}^p_s|| + ||B^p_s - \tilde{B}^p_s||, \tag{21}$$

where B^p_s and $\mathcal{L}$ denote the ground truth blurred RGB for pixel p and the loss function, respectively.

Coarse-to-Fine Optimization. However, it is difficult to simultaneously optimize the loss function from scratch due to the complex geometry and texture of 3D scenes. Hence, we propose a coarse-to-fine optimization strategy for the two losses by introducing coarse-to-fine weight λ, which exponentially decays from λ_s to λ_e during training as shown in Eq. 22:

$$\begin{aligned} \alpha &= -log(\lambda_s/\lambda_e)/(e_f - e_c), \\ \lambda &= \lambda_s(\exp(\alpha(e_c - e_i))), \end{aligned} \tag{22}$$

where e_c, e_i, and e_f denote the current, initial, and final iteration of the training process, respectively. Therefore, our final loss function $\mathcal{L}_{final}$ is defined as shown in Eq. 23:

$$\mathcal{L}_{final} = \lambda||B^p_s - \hat{B}^p_s|| + (1-\lambda)||B^p_s - \tilde{B}^p_s||, \tag{23}$$

where the first and second terms without λ and $(1-\lambda)$ are denoted as $\mathcal{L}_c$ and $\mathcal{L}_f$ in the caption of Figure 2, referring to coarse and fine RGB reconstruction loss, respectively.

4. Experiment

Implementation Details. DP-NeRF is implemented using official code published for a previous work [22]. Note that, our blur operation should be applied to scene irradiance instead of image intensity, as pointed out by [5], following [22]. Hence, a tone mapping function is applied to the predicted radiance color from the NeRF in the same manner as the gamma function in [22]. For fair comparison with [22], we set the default configuration to be same as in that study. The number of camera motions k is set to 4 as the default because the number of kernel points in [22] is set to 5. We use a batch size of 1024 rays, with 64 coarse samples, and 64 fine samples on the rays. We use the Adam [13] optimizer with default parameters. For the scheduling learning rate, we exponentially weight decay from 5×10^{-4} to 8×10^{-5}. In addition, we set λ_s and λ_e at 0.9 and 0.1 for the coarse-to-fine optimization, respectively. We also use $200k$ iterations to train each scene. Further details are provided in the supplementary material.

Datasets. We train DP-NeRF using the synthetic and real scene datasets provided by [22]. Both dataset consist of two blur types: camera motion and defocus blur. There are five scenes in the synthetic dataset and ten in the real dataset for each blur type. The camera poses for all of the images are calibrated using COLMAP [36, 37]. As mentioned in [22], the real scenes were manually captured with a Canon EOS RP under manual exposure mode.

4.1. Novel View Synthesis

Evaluations. In this section, we summarize the results of our model for the synthetic and real scenes. The quantitative quantitative and qualitative results for the synthetic dataset are presented in Table 1 and Figure 4. Three commonly used evaluation metrics are adopted in the present study to compare the synthesized and ground truth images: the peak signal-to-noise ratio (PSNR), the structural similarity index measure (SSIM), and learned perceptual image patch similarity (LPIPS) [56], which assess relative sharpness, structural similarity, and perceptual quality, respectively. Due to the length, we only present the average results here for the real scene dataset. A more version of the experimental results is available in the supplementary material.

Comparisons. Tables 1 and 2 show that our model produces excellent results for all metrics compared to the other models, including [22]. In particular, LPIPS is significantly improved by DP-NeRF, indicating a 3D scene with a higher rendered image quality in terms of perceptual quality. Note that, the results for single image deblurring methods, MPR+NeRF, PVD+NeRF, and KPAC+NeRF, in Tables 1 and 2 are taken from [22]. They are trained with a Naive NeRF using images deblurred using MPR [53], PVD [41], and KPAC [40] methods, respectively.

Camera Motion	Factory			Cozyroom			Pool			Tanabata			Trolley			Average		
	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)
Naive NeRF [26]	19.32	.4563	.5304	25.66	.7941	.2288	30.45	.8354	.1932	22.22	.6807	.3653	21.25	.6370	.3633	23.78	.6807	.3362
MPR [53] + NeRF	21.70	.6153	.3094	27.88	.8502	.1153	30.64	.8385	.1641	22.71	.7199	.2509	22.64	.7141	.2344	25.11	.7476	.2148
PVD [41] + NeRF	20.33	.5386	.3667	27.74	.8296	.1451	27.56	.7626	.2148	23.44	.7293	.2542	23.81	.7351	.2567	24.58	.7190	.2475
Deblur-NeRF [22]	25.60	.7750	.2687	32.08	.9261	.0477	31.61	.8682	.1246	27.11	.8640	.1228	27.45	.8632	.1363	28.77	.8593	.1400
DP-NeRF	25.91	.7787	.2494	32.65	.9317	.0355	31.96	.8768	.0908	27.61	.8748	.1033	28.03	.8752	.1129	29.23	.8674	.1184
Defocus	**Factory**			**Cozyroom**			**Pool**			**Tanabata**			**Trolley**			**Average**		
	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)	PSNR(↑)	SSIM(↑)	LPIPS(↓)
Naive NeRF [26]	25.36	.7847	.2351	30.03	.8926	.0885	27.77	.7266	.3340	23.80	.7811	.2142	22.67	.7103	.2799	25.93	.7791	.2303
KPAC [40] + NeRF	26.40	.8194	.1624	28.15	.8592	.0815	26.69	.6589	.2631	24.81	.8147	.1639	23.42	.7495	.2155	25.89	.7803	.1773
Deblur-NeRF [22]	28.03	.8628	.1127	31.85	.9175	.0481	30.52	.8246	.1901	26.26	.8517	.0995	25.18	.8067	.1436	28.37	.8527	.1188
DP-NeRF	29.26	.8793	.1035	32.11	.9215	.0386	31.44	.8529	.1563	27.05	.8635	.0779	26.79	.8395	.1170	29.33	.8713	.0987

Table 1. Quantitative results for the synthetic scene. Each color shading indicates the best and second-best result, respectively.

a) Naïve NeRF b) Deblur-NeRF c) DP-NeRF(RBK) d) DP-NeRF(RBK+AWP) e) Reference

Figure 4. Rendered novel view synthesis results of DP-NeRF for synthetic scenes. Top and bottom row denote results of camera motion and defocus blur scene, respectively. Figure (a)-(e) denote Naive NeRF, Deblur-NeRF, DP-NeRF (RBK), DP-NeRF (RBK+AWP), and ground truth images, respectively. Each colored box in corner of images are enlarged parts of the colored box region in reference images.

Camera Motion	PSNR(↑)	SSIM(↑)	LPIPS(↓)	Defocus	PSNR(↑)	SSIM(↑)	LPIPS(↓)
Naive NeRF [26]	22.69	.6347	.3687	Naive NeRF [26]	22.40	.6661	.2310
MPR [53] + NeRF	23.38	.6655	.3140	KPAC [40] + NeRF	23.04	.6917	.1847
PVD [41] + NeRF	23.10	.6389	.3425	-	-	-	-
Deblur-NeRF [22]	25.63	.7675	.1820	Deblur-NeRF [22]	23.46	.7199	.1207
DP-NeRF	25.91	.7751	.1602	DP-NeRF	23.67	.7299	.1082

Table 2. Average results for the real scene dataset. Each color shading indicates the best and second-best result, respectively.

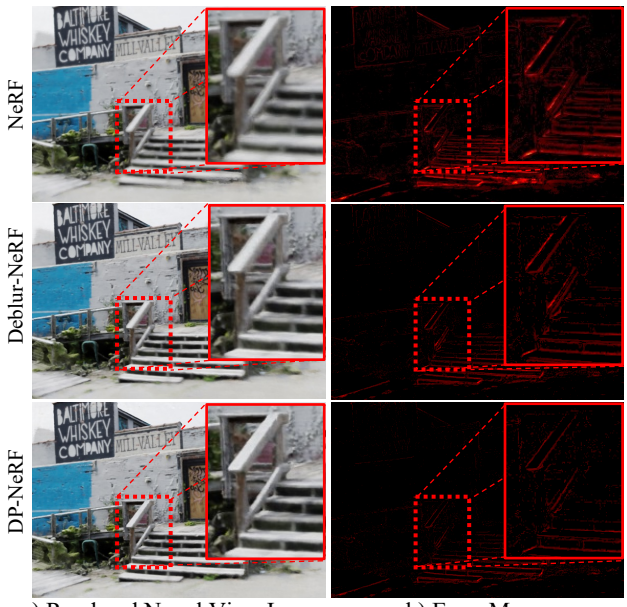
The qualitative results presented in Figure 4 demonstrate the effectiveness of DP-NeRF. The RBK more successively models a clean 3D scene representation with geometric and appearance consistency compared to previous approaches. It more cleanly reproduces the structure of the object in the scene compared to Naive NeRF and Deblur-NeRF.

4.2. Ablation Study

Effectiveness of the RBK and AWP. The DP-NeRF successfully constructs clean NeRF with geometric and appearance consistency using RBK and AWP as shown in Figure 4. The RBK by itself still has difficulty in inferring the correct texture and geometry in some regions in which the texture is confused with the background, the structure is thin, or the depth is complex, as indicated by the red box in Figure 4 (c). However, the rendered results in Figure 4 (d) demonstrate that the AWP effectively refines the geometric and appearance consistency in this region. In particular, the upper and lower rows exhibit detailed enhancement of the geometric structure and texture of the edge of the box and stairs, respectively.

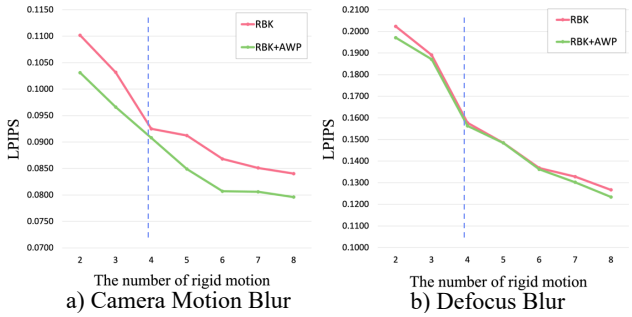
To understand the effectiveness of DP-NeRF more clearly, Figure 5 present an error map visualization of the stairs scene, with the brighter colors indicating greater error. DP-NeRF produces a lower error than the baselines, reconstructing the fine details of the objects in the scene. We also provide additional error map images without the red boxes in the supplementary material, including those for another type of blur in synthetic scenes.

The Number of Rigid Motions. Figure 6 presents ablation analysis for the number of rigid motions, which defines the number of transformed rays, for the two types of the blur. We use LPIPS as the evaluation metric because it represents perceptual quality well, as demonstrate in Nerfies [29]. The results show that the performance of the RBK and RBK+AWP improves as the number of rigid motions increases, while the LPIPS for RBK+AWP is better than that for the RBK alone in all experiments. Full quantitative PSNR, SSIM, and LPIPS results and qualitative results for the RBK and RBK+AWP are additionally presented in the supplementary material.



a) Rendered Novel View Image b) Error Map

Figure 5. Visual comparison with error maps of NeRF, Deblur-NeRF, and DP-NeRF (ours) in defocus **Factory** scene. Regions with red box indicate emphasized regions of error map.



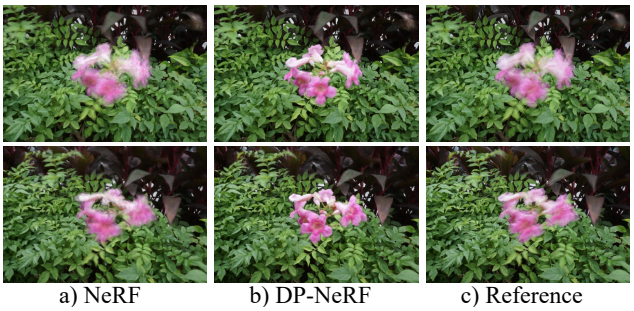
a) Camera Motion Blur b) Defocus Blur

Figure 6. Ablation analysis on the number of rigid camera motions for the two types of blur for synthetic **Pool** scene. (a) and (b) show the results on camera motion and defocus blur, respectively. magenta and yellowgreen colors indicate results of RBK and RBK+AWP, respectively. blue color line indicates the results when the number of rigid motions is 4, which is same as of the kernel points 5 in Deblur-NeRF [22] for fair comparison in Table 1.

RBK Analysis. The RBK is shown to successfully model the blur derived from both camera movement and a change in the focus plane as the rigid motion of the camera. We demonstrate the validity of the RBK design by presenting additional kernel analysis in the supplementary material.

5. Limitation & Future Work

Temporal Motion Blur. Our model fails when object motion blur is present in a scene (Figure 7). Though DP-NeRF produces images of higher quality than does a standard NeRF, motion blur still occurs in reconstructed scenes because we impose the physical priors based on the assumption of a static scene. Object motion blur is an issue associated with temporal information and there is no multi-view data for a specific time in the given dataset. This dataset



a) NeRF b) DP-NeRF c) Reference

Figure 7. Visual comparison for the NeRF, the DP-NeRF, and the reference image at specific time on object motion blur. There are still in artifacts in region where motion blur occurred.

structure leads to inherent geometric and appearance inconsistency in the 3D scene if we construct the NeRF without temporal modeling. Therefore, we are confident that this limitation can be addressed with an additional temporal component in DP-NeRF in the future, such as bending the rays with $SE(3)$ field warping [29], scene flow [18], or 3D displacement [33] on ray samples. Actually, although Figure 7 (b) seems to be quite clean, the results show the explicit inconsistency when we render the spiral video to assess the 3D consistency. Please refer the rendered videos in our project page to check the temporal inconsistency.

Various Types of Image Noise. As mentioned in Section 2, several previous studies have addressed other types of image noise for NeRF, such as temporal and exposure variation. The modeling for these noise could be integrated with the DP-NeRF system because they are independent of each other, including our target noise. This can be addressed in the future research if an appropriate dataset is constructed.

6. Conclusion

This paper proposes DP-NeRF, a novel NeRF framework from blurry inputs, that imposes the two physical priors to effectively construct a clean NeRF. We propose the RBK to maintain geometric and appearance consistency in continuous 3D space. In addition, We employ the AWP module to alleviate the color composition errors by considering the relationship between depth and blur. We also introduce coarse-to-fine optimization of the two losses from the proposed modules to effectively utilize both during the training process. Extensive experiments using synthetic and real scene datasets verified that DP-NeRF produces an improved, clean NeRF with high perceptual quality and 3D consistency in terms of geometry and appearance. We believe that DP-NeRF represents an advance in NeRF and can be used in conjunction with other methods to construct clean NeRF, covering the images with other types of noise.

Acknowledgements. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2021-0-02068, Artificial Intelligence Innovation Hub) and the Yonsei University Research Fund of 2021 (2021-22-0001).

References

[1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 1, 2

[2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022. 1, 2

[3] Sai Bi, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Neural reflectance fields for appearance acquisition. *arXiv preprint arXiv:2008.03824*, 2020. 2

[4] Piotr Bojanowski, Armand Joulin, David Lopez-Paz, and Arthur Szlam. Optimizing the latent space of generative networks. *arXiv preprint arXiv:1707.05776*, 2017. 4

[5] Xiaogang Chen, Feng Li, Jie Yang, and Jingyi Yu. A theoretical analysis of camera response functions in image deblurring. In *European Conference on Computer Vision*, pages 333–346. Springer, 2012. 6

[6] Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12943–12952, 2022. 1

[7] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11108–11117, 2020. 5

[8] Xin Huang, Qi Zhang, Ying Feng, Hongdong Li, Xuan Wang, and Qing Wang. Hdr-nerf: High dynamic range neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18398–18408, 2022. 1, 2

[9] Jiaya Jia. Single image motion deblurring using transparency. In *2007 IEEE Conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007. 3

[10] Neel Joshi, C Lawrence Zitnick, Richard Szeliski, and David J Kriegman. Image deblurring and denoising using color priors. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1550–1557. IEEE, 2009. 3

[11] Kim Jun-Seong, Kim Yu-Ji, Moon Ye-Bin, and Tae-Hyun Oh. Hdr-plenoxels: Self-calibrating high dynamic range radiance fields. *arXiv preprint arXiv:2208.06787*, 2022. 1, 2, 3

[12] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*, 18(3):165–174, 1984. 3

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[14] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR 2011*, pages 233–240. IEEE, 2011. 3

[15] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8183–8192, 2018. 3

[16] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8878–8887, 2019. 3

[17] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5521–5531, 2022. 1, 2

[18] 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. 1, 2, 8

[19] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751*, 2019. 1, 2

[20] Leon B Lucy. An iterative technique for the rectification of observed distributions. *The astronomical journal*, 79:745, 1974. 3

[21] Kevin M Lynch and Frank C Park. *Modern robotics*. Cambridge University Press, 2017. 4

[22] Li Ma, Xiaoyu Li, Jing Liao, Qi Zhang, Xuan Wang, Jue Wang, and Pedro V Sander. Deblur-nerf: Neural radiance fields from blurry images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12861–12870, 2022. 1, 2, 3, 4, 6, 7, 8

[23] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7210–7219, 2021. 1, 2

[24] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. 3

[25] Ben Mildenhall, Peter Hedman, Ricardo Martin-Brualla, Pratul P Srinivasan, and Jonathan T Barron. Nerf in the dark: High dynamic range view synthesis from noisy raw images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16190–16199, 2022. 1, 2

[26] 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 *Computer Vision–ECCV 2020: 16th European*

Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I, pages 405–421, 2020. 1, 3, 7

[27] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3883–3891, 2017. 3

[28] 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. 2

[29] 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. 1, 2, 4, 5, 7, 8

[30] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021. 1, 2

[31] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Animatable neural radiance fields for modeling dynamic human bodies. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14314–14323, 2021. 2

[32] Julien Philip, Sébastien Morgenthaler, Michaël Gharbi, and George Drettakis. Free-viewpoint indoor neural relighting from multi-view stereo. *ACM Transactions on Graphics (TOG)*, 40(5):1–18, 2021. 2

[33] 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. 1, 2, 8

[34] William Hadley Richardson. Bayesian-based iterative method of image restoration. *JoSA*, 62(1):55–59, 1972. 3

[35] Olinde Rodrigues. *De l'attraction des sphéroïdes, Correspondence sur l'É-cole Impériale Polytechnique*. PhD thesis, PhD thesis, Thesis for the Faculty of Science of the University of Paris, 1816. 4

[36] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 6

[37] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision*, pages 501–518. Springer, 2016. 6

[38] 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. 2

[39] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *Acm transactions on graphics (tog)*, 27(3):1–10, 2008. 3

[40] Hyeongseok Son, Junyong Lee, Sunghyun Cho, and Seungyong Lee. Single image defocus deblurring using kernel-sharing parallel atrous convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2642–2650, 2021. 6, 7

[41] Hyeongseok Son, Junyong Lee, Jonghyeop Lee, Sunghyun Cho, and Seungyong Lee. Recurrent video deblurring with blur-invariant motion estimation and pixel volumes. *ACM Transactions on Graphics (TOG)*, 40(5):1–18, 2021. 6, 7

[42] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7495–7504, 2021. 2

[43] Pratul P Srinivasan, Rahul Garg, Neal Wadhwa, Ren Ng, and Jonathan T Barron. Aperture supervision for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6393–6401, 2018. 2, 5

[44] Pratul P Srinivasan, Ren Ng, and Ravi Ramamoorthi. Light field blind motion deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3958–3966, 2017. 2, 5

[45] Shih-Yang Su, Frank Yu, Michael Zollhöfer, and Helge Rhodin. A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. *Advances in Neural Information Processing Systems*, 34:12278–12291, 2021. 2

[46] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 769–777, 2015. 3

[47] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021. 2

[48] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8174–8182, 2018. 3

[49] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12959–12970, 2021. 1, 2

[50] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 2

[51] Tiange Xiang, Chaoyi Zhang, Yang Song, Jianhui Yu, and Weidong Cai. Walk in the cloud: Learning curves for point clouds shape analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 915–924, 2021. 5

[52] 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. 3

[53] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021. 3, 6, 7

[54] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. Deep stacked hierarchical multi-patch network for image deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5978–5986, 2019. 3

[55] Jiakai Zhang, Xinhang Liu, Xinyi Ye, Fuqiang Zhao, Yanshun Zhang, Minye Wu, Yingliang Zhang, Lan Xu, and Jingyi Yu. Editable free-viewpoint video using a layered neural representation. *ACM Transactions on Graphics (TOG)*, 40(4):1–18, 2021. 1, 2

[56] 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. 6

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

[58] Fuqiang Zhao, Wei Yang, Jiakai Zhang, Pei Lin, Yingliang Zhang, Jingyi Yu, and Lan Xu. Humannerf: Efficiently generated human radiance field from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7743–7753, 2022. 2

[59] Zelin Zhao and Jiaya Jia. End-to-end view synthesis via nerf attention. *arXiv preprint arXiv:2207.14741*, 2022. 2, 5