BLiRF: Band Limited Radiance Fields for Dynamic Scene Modeling

Sameera Ramasinghe, Violetta Shevchenko, Gil Avraham, Anton van den Hengel

Amazon, Australia

Abstract

Inferring the 3D structure of a non-rigid dynamic scene from a single moving camera is an under-constrained problem. Inspired by the remarkable progress of neural radiance fields (NeRFs) in photo-realistic novel view synthesis of static scenes, it has also been extended to dynamic settings. Such methods heavily rely on implicit neural priors to regularize the problem. In this work, we take a step back and investigate how current implementations may entail deleterious effects including limited expressiveness, entanglement of light and density fields, and sub-optimal motion localization. Further, we devise a factorisation-based framework that represents the scene as a composition of bandlimited, high-dimensional signals. We demonstrate compelling results across complex dynamic scenes that involve changes in lighting, texture and long-range dynamics.

Introduction

The problem of scene modeling (Kolmogorov and Zabih 2002; Dyer 2001) is one of the fundamental challenges in computer vision, and underpins many of the field's most prominent applications including novel view synthesis (Avidan and Shashua 1997; Daribo and Pesquet-Popescu 2010), augmented and virtual reality (Azuma 1997; Burdea and Coiffet 2003), and SLAM (Grisetti et al. 2010; Mur-Artal, Montiel, and Tardos 2015). In this vein, Neural Rendering Fields (NeRFs) (Mildenhall et al. 2021) have recently exhibited remarkable progress in synthesizing photorealistic novel views from sparse 2D images.

One of the factors underpinning the success of NeRFs is the architectural bias of neural networks. The (Lipschitz) smoothness of neural networks acts as an implicit *neural prior* for self-regularizing the optimization process, which is otherwise ill posed (Zhang et al. 2020). Recently, multiple works have extended NeRFs to dynamic settings, leveraging the same neural smoothness prior that made NeRFs successful. For modeling the evolution of scene geometry over time, these works have primarily resorted to using ray deformation paradigms, which parameterize rays cast from the camera as functions of time (Pumarola et al. 2021; Tretschk et al. 2021; Park et al. 2021a; Li et al. 2021; Gao et al. 2021; Xian et al. 2021; Park et al. 2021b). Although these approaches have yielded impressive results, we show that their over-reliance on implicit neural priors gives rise to fundamental problems; *a*) Dependency on a canonical frame which harms modeling long range motion, *b*) Entanglement of the light and density fields *c*) Limited expressiveness due to network bottlenecks, and *d*) substandard localization of motion due to the difference in the spectral properties of space and time, *i.e.*, space typically consists of sharp/high-frequency details, whereas temporal dynamics are generally smooth and continuous.

To overcome the above drawbacks, we propose a theoretical framework that enables efficient integration of implicit neural priors and well-defined explicit priors. On this basis we also propose a set of explicit priors, partially inspired by non-rigid-structure-from-motion (NRS*f*M). In particular, we model the light and density fields of a 3D scene as bandlimited, high-dimensional signals. This standpoint enables complete factorization of spatio-temporal dynamics, allowing us to inject explicit priors on the time and space dynamics independently. To demonstrate the practical utility of our framework, we offer an example implementation that enforces 1) a low-rank constraint on the shape space, along with 2) a neural prior over the frequency domain and 3) a union-of-subspaces prior on the deformation of a shape over time. We show that the strong regularization effects of these priors enable our model to reconstruct long-range dynamics and localize motion accurately. Further, our model does not rely on complex optimization procedures (Pumarola et al. 2021; Li et al. 2021; Park et al. 2021a; Yoon et al. 2020; Park et al. 2021b) or multiple explicit loss regularizations (Tretschk et al. 2021; Gao et al. 2021; Park et al. 2021a; Li et al. 2021; Wang et al. 2021) that are common in existing dynamic NeRF works, indicating the stability of our formulation. Finally, our implementation efficiently disentangles light and density fields, allowing the model to capture challenging scenes with dynamic lighting and textures. Our contributions are summarized as follows.

- We show that existing extensions of NeRF to dynamic scenes suffer from critical drawbacks, primarily due to their over-reliance on implicit neural priors.
- We propose a novel framework for modeling dynamic 3D scenes that overcomes the above drawbacks by formulating radiance fields as bandlimited signals.
- We empirically validate the efficacy of our framework by

Figure 1: We recover the 3D structure of a dynamic scene given sparse RGB views from a monocular, moving camera. The figure shows a comparison between novel views of a challenging scene with long-range dynamics. As illustrated, our model is able to capture fine details better and accurately localize the motion compared to NR-NeRF (Tretschk et al. 2021). We attribute the superior performance of our model to efficient factorization of time and space dynamics that enable incorporating well-defined spatio-temporal priors.

demonstrating better modeling of long-range dynamics, motion localization, and light/texture changes, achieving state-of-the-art results over competitive datasets. Our model only takes around 3 hours per scene to train (more than 10 times faster than NR-NeRF, D-NeRF, and HyperNeRF), and does not require complex loss regularizers or optimization procedures.

Related Work

Recovering the 3D structure of a scene using a monocular camera is a challenging task that has been approached from various angles (Davison et al. 2007; Newcombe et al. 2011; Yoon et al. 2020; Dou et al. 2016; Avidan and Shashua 2000; Wexler and Shashua 2000; Niemeyer et al. 2019; Li et al. 2021). Dynamic scenes add complexity to the 3D modelling process due to the extra mode of variation that must be accommodated, and many methods require either multiple cameras(Zhang, Curless, and Seitz 2003; Tung, Nobuhara, and Matsuyama 2009; Zhang et al. 2004; Dou, Fuchs, and Frahm 2013; Dou et al. 2016) or active depth sensors(Newcombe et al. 2011; Newcombe, Fox, and Seitz 2015; Slavcheva et al. 2017; Yu et al. 2017) to resolve the ambiguity. Following NeRF, many dynamic neural radiance field models have been developed, primarily using the concept of ray deformation (Pumarola et al. 2021; Tretschk et al. 2021; Park et al. 2021a; Li et al. 2021; Gao et al. 2021; Park et al. 2021b). D-NeRF(Pumarola et al. 2021) first proposed a general framework which learns a per-point displacement against a canonical location to model ray deformation. Both(Tretschk et al. 2021; Gao et al. 2021) further introduced a constraint to model the foreground and background separately. (Gao et al. 2021) disambiguated self-occlusions that hinders the performance of these approaches. Nerfies (Park et al. 2021a) achieves remarkable results by incorporating elastic regularization, but only targets self-portraits. Several other NeRF extensions have also been proposed that require depth estimates (Xian et al. 2021), optical flow (Wang et al. 2021), foreground masks (Johnson et al. 2022; Gao et al. 2021), meshes (Xu and Harada 2022), or assume that dynamic objects are distractors to be removed (Martin-Brualla et al. 2021).

Revisiting Ray Deformation Networks

Extending NeRF to dynamic scenes requires representing the scene as a continuous function with 6D inputs $(x, y, z, \theta, \phi, t)$, where t is the time and (θ, ϕ) is the viewing direction. However, it has been shown empirically (Pumarola et al. 2021) that employing a single MLP to learn a mapping from 6D inputs to density and color fields yields sub-optimal results. Hence, existing works decompose the aforementioned task into two modules (Pumarola et al. 2021; Tretschk et al. 2021; Park et al. 2021a; Li et al. 2021; Gao et al. 2021; Park et al. 2021b; Fang et al. 2022): 1) the first MLP learns a warping field of 3D points $(\Delta x, \Delta y, \Delta z)$ sampled along the rays with respect to a canonical setting; 2) the second module then acts similarly to the original NeRF formulation, regressing the density and light fields given the warped samples along the rays $(x + \Delta x, y + \Delta y, z + \Delta z)$. Since the warping is applied to points sampled along the ray, this formulation is interpreted as deforming the rays as a function of time. Further, note that this assumes that objects do not enter or leave the scene, and that lighting/texture is consistent. However, we notice that existing implementations of this framework do not adhere to these constraints (see supplementary). Specifically, we show that such networks can indeed model light

and density changes separately (to an extent), which is infeasible with a model that only learns ray deformations (see Fig. 3 and supplementary). However, to avoid confusion, we will keep referring to this class of models as ray deformation networks. Next, we discuss several critical limitations thereof.

Limitations of Ray Deformation Networks

In this section, we present a brief exposition of the limitations entailed in the ray deformation approach. For an extended analysis, refer to supplementary.

Dependency on a canonical frame: Ray deformation networks require choosing a canonical frame, and most models choose the frame at $t = 0$ to this end. The particular choice of frame can significantly harm model performance in cases where 1) objects or the camera exhibit long-range translations, and 2) new objects appear in subsequent frames of the video. The canonical frame thus needs to provide a form of average scene representation where all future information is present. This becomes increasingly infeasible as the scene becomes more complex. On the other hand, the ray deformation model also needs to preserve continuity; the model output at $(t = \delta t)$ needs to embody a smooth transition of the canonical scene at $t = 0$, which can be impractical if the scene comprises abundant future information. In contrast, our framework does not use ray deformation and thus does not depend on a canonical scene. **Entanglement of light and density fields:** Although ray deformation networks are able to deform the light and density fields, they are still highly entangled. More precisely, it can be shown that in order to achieve complete disentanglement of the light and density fields, the network needs to preserve a specific block-diagonal Jacobian structure in one of the hidden layers, which is an extremely restrictive requirement (see supplementary). In comparison, our framework achieves complete disentanglement by design, modeling the light and density fields independently. **Limited expressiveness:** Ray deformation networks comprise a bottleneck of dimension three. Therefore, each of the density and light fields modeled by this network becomes a three dimensional manifold. They cannot thus encode complex dynamics that need to be parameterized by four variables (x, y, z, t) (see Supplementary). **Substandard separation of background and motion:** Ray deformation networks model the warp field using a single MLP. However, this is a substandard design choice since the space and time variations have different spectral properties. Natural scenes often exhibit high-frequency spatial characteristics such as fine-grained surface details for example, but temporal changes are generally smoother. Therefore, using an MLP with a particular bandwidth for learning both spatial and time variations together leads to sub-optimal reconstructions. One way to overcome this problem is to use separate MLPs for time and space dynamics, and control their bandwidth via positional encodings. This strategy requires involved optimization procedures that demand careful coarse-to-fine hyperparameter annealing that depends on the dataset, and incur longer training times (64 hours on 4 TPU v4s vs ours (3 hours on a single V100)) (Park et al. 2021b). In contrast, our framework enables much more elegant factorization of space and time dynamics by modeling the scene as bandlimited signals, allowing better separation of static and dynamic regions.

Our Framework

Consider a set of 2D projections $\{I(t_n)\}_{n=1}^{N}$ of a 3D scene captured from a moving camera. For brevity, we drop the dependency on the camera poses from the notation. Without the loss of generality, we assume that the scene is bounded within a cube with side length D. We begin by observing that there exists a latent density and color field corresponding to each $I(t_n)$ that can be discretized into a cubic grid of D^3 nodes. Then, rewriting the latent states of either field in the matrix form, gives

$$\mathbf{S} = \begin{bmatrix} s(t_1, \mathbf{x}_1) & s(t_1, \mathbf{x}_2) & \dots & s(t_1, \mathbf{x}_{D^3}) \\ \vdots & \vdots & \ddots & \vdots \\ s(t_N, \mathbf{x}_1) & s(t_N, \mathbf{x}_2) & \dots & s(t_N, \mathbf{x}_{D^3}) \end{bmatrix}_{N \times D^3} \tag{1}$$

where $\mathbf{x}_i \in \mathbb{R}^3$ and $s(t_n, \mathbf{x}_i)$ can be either the density or the color values emitted from $\mathbf{x}_i$ at time t_n. Note that $s(\cdot, \cdot)$ can be either a scalar valued function for density and a vector valued function for color. To avoid cluttered notation, we consider the scalar valued case for the following derivations. However, our analysis holds true for the vector valued case as well; Let $\mathrm{rank}(\mathbf{S}) = K$. Then, there exist K basis vectors, each with dimension D^3, that can perfectly reconstruct (memorize) $\mathbf{S}$. More precisely, in this case, each row of $\mathbf{S}$ can be reconstructed as

$$\mathbf{S}(t_n, \cdot) = \sum_{j=1}^{K} a_{j,n}\, \hat{\boldsymbol{\alpha}}_j, \tag{2}$$

where $\mathbf{S}(t_n, \cdot)$ is the n^{th} row of $\mathbf{S}$, $\{\hat{\boldsymbol{\alpha}}_j\}_{j=1}^{K}$ are basis vectors of dimension D^3, and $\{a_{j,n}\}$ are scalar coefficients. Intuitively, each row of $\mathbf{S}$ corresponds to a snapshot of the field (in space) at a particular time instant. On the contrary, each column of $\mathbf{S}$ is a representation evolution of a particular point $\mathbf{x}$ over time. We note an interesting duality here; since the dimension of the row space and the column space of $\mathbf{S}$ are equal, it should be possible to reconstruct the evolution of the density/color value of each position $\mathbf{x}$ over time using K basis vectors. Thus, we model the time evolution of each point as

$$\mathbf{S}(\mathbf{x}_i, \cdot) = \sum_{j=1}^{K} b_{j,i}\, \hat{\boldsymbol{\beta}}_j, \tag{3}$$

where $\mathbf{S}(\mathbf{x}_i, \cdot)$ is the i^{th} column of $\mathbf{S}$, $\{\hat{\boldsymbol{\beta}}_j\}_{j=1}^{K} \in \mathbb{R}^N$ are basis vectors, and $\{b_{j,i}\}$ are scalars. This change of perception is crucial for generalizing to unseen time instances and obtaining a space-time factorization, as we shall discuss next. Using Eq. 3, the value of color/density of a point $\mathbf{x}$ at a particular continuous time instance t can be obtained as

$$\mathbf{S}(\mathbf{x}, t) = \sum_{j=1}^{K} \tilde{b}_j(\mathbf{x})\, \psi_j(t), \tag{4}$$

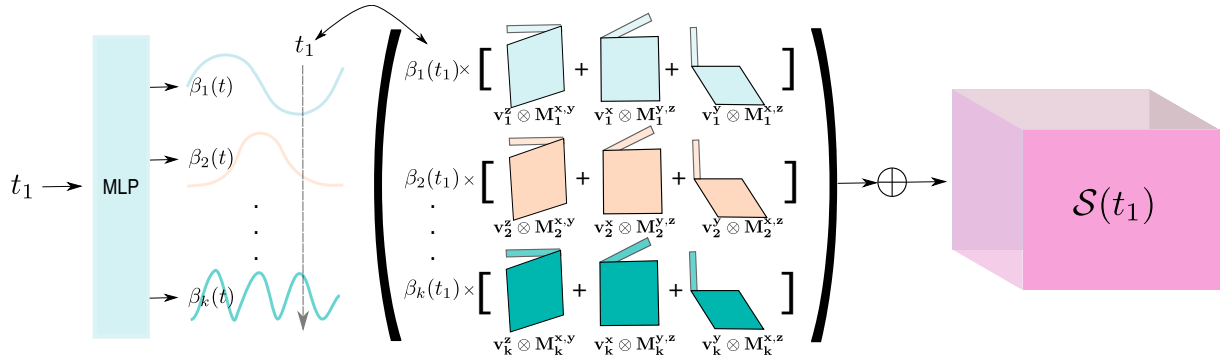

Figure 2: The proposed implementation of our framework. We treat the light and density fields as bandlimited, high-dimensional signals (only a single field is shown in the figure). The time evolution of each 3D point (x, y, z) of the field is modeled as a finite linear combination of time-basis functions $\{\beta_j(t)\}$. The coefficients of the $\{\beta_j(t)\}$ are decomposed into outer products between learnable matrices ($\mathbf{M}$) and vectors ($\mathbf{v}$). Our formulation allows efficient factorization of time and space dynamics, leading to high-quality reconstructions of complex dynamics, with faster convergence.

where $\psi_j(t) = \sum_{n=1}^{N} \hat{\boldsymbol{\beta}}_{j,n}\delta(t-t_n)$, $\delta(\cdot)$ is the Dirac delta function, and $\hat{\boldsymbol{\beta}}_{j,n}$ is the n^{th} element of $\hat{\boldsymbol{\beta}}_j$. $\tilde{b}_j(\mathbf{x})$ is a scalar valued function. A key problem associated with Eq. 4 is that $\psi(t)$ is an infinite bandwidth function because δ is of infinite bandwidth. As a consequence, $\mathbf{S}(\mathbf{x}, t)$ also becomes an infinite bandwidth function. Equivalently, an infinite number of time-sample points are required to reconstruct the continuous signal $\mathbf{S}(\mathbf{x}, t)$[1]. Recall, however, that, in practice, only a sparse, finite set of 2D observations $\{I(t_n)\}_{n=1}^{N}$ are at our disposal. Therefore, the infinite bandwidth representation of Eq 4 is not ideal for obtaining a $\mathbf{S}(\mathbf{x}, t)$ that can be queried at arbitrary continuous time instances. Therefore, we replace $\{\psi_j\}$ with a set of bandlimited scalar-valued functions $\{\beta_j(\cdot)\}$.

$$\mathbf{S}(\mathbf{x}, t) = \sum_{j=1}^{K} \tilde{b}_j(\mathbf{x})\ \beta_j(t). \tag{5}$$

From a signal processing perspective, this can also be considered as reconstructing a signal from discrete samples using a linear combination of bandlimited basis functions. Observe that from this perspective, the $\psi_j(t)$'s can be considered discrete samples of the continuous functions $\beta_j(t)$. Consequently, $\mathbf{S}(\mathbf{x}, t)$ also becomes a bandlimited function (a linear combination of bandlimited functions is bandlimited). Note that now we have also obtained a factorization of time and spatial dynamics that will allow us to impose priors on time and space independently. In the next section, we present an implementation of the proposed framework. In this implementation, we inject a low-rank prior on space, along with smoothness and compact manifold priors on time. It is worth to note that our framework is generic enough to support alternative implementations and more complex priors, which we leave to future explorations.

[1]Recall that in order to reconstruct a continuous signal as a linear combination of shifted Dirac delta functions, an infinite number of sampling points are needed.

Implementation

Leveraging the factorization we achieved in Eq. 5, we can formulate the entire 3D field volume as a time-dependent higher dimensional signal, that can be decomposed into a linear combination of 3D tensors $\mathcal{A}_j^{xyz} \in \mathbb{R}^{D\times D\times D}$:

$$\mathcal{S}(t) = \sum_{j=1}^{K} \beta_j(t)\mathcal{A}_j^{xyz}, \tag{6}$$

where $\mathcal{S}(t) \in \mathbb{R}^{D\times D\times D}$ is the state of the field at time t. Note that we adopt the tensor notation here where the superscripts denote the dimensions, *i.e.*, $x = 1, \dots D, y = 1, \dots, D$, and $z = 1, \dots, D$. To regularize the spatial variations, we employ a low-rank constraint on $\mathcal{A}_j$ as,

$$\mathcal{S}(t) = \sum_{j=1}^{K} \beta_j(t)(\mathbf{v}_j^z \otimes \mathbf{M}_j^{xy} + \mathbf{v}_j^x \otimes \mathbf{M}_j^{yz} + \mathbf{v}_j^y \otimes \mathbf{M}_j^{xz}), \tag{7}$$

where $\mathbf{v}_j \in \mathbb{R}^D$ and $\mathbf{M}_j \in \mathbb{R}^{D\times D}$ are one- and two-dimensional tensors, respectively, and $\otimes$ is the outer product. The above choice of factorization is inspired by the *VM-decomposition* proposed in (Chen et al. 2022). This factorization accomplishes two goals: 1) enforcing a low rank constraint on the spatial variations of the field, and 2) significantly reducing the size of the model and the number of trainable parameters. We note that such low-rank priors have been widely employed in the NRS*f*M literature for the same purpose (Torresani et al. 2001; Torresani, Hertzmann, and Bregler 2003; Rabaud and Belongie 2008).

Neural Trajectory Basis

In theory, it is possible to use any class of bandlimited functions that form a complete basis in $L^2(\mathbb{R}, dt)$ as $\{\beta_j(t)\}$. Popular choices include the DCT, Fourier, and Bernstein basis functions, among many others. Nonetheless, we use neural networks to parameterize our basis functions, leveraging the implicit architectural smoothness constraint built

into them. We label these basis functions as the *neural trajectory basis*. The neural trajectory basis presents an important implicit prior to our model, that the field values should evolve smoothly. We also empirically note that neural basis functions are naturally more expressive and adaptive as they are learned end-to-end, as opposed to other choices (see Table 3). Expressiveness is crucial, as it is desirable to model the dynamics of each point with a minimal number of basis functions. Thus, we compute $\{\beta_j(t)\}$ via an MLP $\mathcal{F}(t) : \mathbb{R} \rightarrow \mathbb{R}^K$ as,

$$\mathcal{F}(t) = [\beta_1(t), \beta_2(t), \ldots, \beta_K(t)]. \tag{8}$$

We also show that the smoothness prior embedded into the neural trajectory basis closely aligns with (Valmadre and Lucey 2012), where they showed that, in NRS*f*M, a trajectory's response to high-pass filters should be minimal. We validate that neural trajectory basis exhibits this property in supplementary. Overall architecture is depicted in Fig. 2.

Manifold Regularization

Multiple works in NRS*f*M have explored restricting the subspace of dynamics in order to obtain better reconstructions. The high-level objective is to temporally cluster the motion in order to restrict similar dynamics to a low-dimensional subspace (Kumar, Dai, and Li 2017; Agudo and Moreno-Noguer 2017; Zhu et al. 2014; Zappella et al. 2013). We observed that such a constraint can improve our reconstructions also. More formally, we empirically asserted that better results are obtained by locally restricting the dimension of the submanifold that $\mathcal{S}(t)$ is immersed in. Instead of clustering the motion across the entire sequence, we assume that dynamics are locally compact: movements that occur over a small time period can be described using a smaller subspace. To enforce this constraint, we adopt the following procedure.

Observe that $\mathcal{S}(t)$ is a 1-dimensional manifold embedded in a D^3-dimensional space (its local coordinate chart is a compact subspace in $\mathbb{R}$). Further, at any given time t, $\mathcal{S}(t)$ is a linear combination of K points $\{\mathbf{v}_j^z \otimes \mathbf{M}_j^{xy} + \mathbf{v}_j^x \otimes \mathbf{M}_j^{yz} + \mathbf{v}_j^y \otimes \mathbf{M}_j^{xz}\}_{j=1}^K \in \mathbb{R}^{D\times D\times D}$. Therefore, $\mathcal{S}(t)$ is a submanifold of $\mathbb{R}^K$.

Now, let $\mathbf{P}_j^{xyz} = (\mathbf{v}_j^z \otimes \mathbf{M}_j^{xy} + \mathbf{v}_j^x \otimes \mathbf{M}_j^{yz} + \mathbf{v}_j^y \otimes \mathbf{M}_j^{xz})$. Suppose the dimension of the local submanifold we need is W, such that $K = dW$ for some integer d. Then, we define the 4D tensor $\mathbf{Q}_{j:j+W}^{xyzu} \in \mathbb{R}^{D\times D\times D\times W}$ such that $\mathbf{Q}_{j:j+W}^{xyzu} = \{\mathbf{P}_u^{xyz}\}_{u=j}^{j+W}$. Next, we obtain

$$\tilde{\mathbf{Q}}^{xyzu}(t) = \sum_{n=0}^{d-1} \mathbf{Q}_{(nW+1):W(n+1)}^{xyzu} \odot \mathrm{sinc}\big((d-1)(t - \tfrac{n}{(d-1)})\big), \tag{9}$$

where $\odot$ represents element-wise multiplication, and

$$\mathrm{sinc}(r) = \begin{cases} 1, & \text{if } r = 0 \\ \frac{\sin(r)}{r}, & \text{otherwise} \end{cases}.$$

The choice of the $\mathrm{sinc}(\cdot)$ function here is not arbitrary, and is crucial for the smooth transition between submanifolds as the time progresses. More precisely, the sinc interpolation ensures that no frequencies higher than $(d-1)/2$ can be presented in $\tilde{\mathbf{Q}}^{xyzu}(t)$ along the temporal dimension. Finally, we can obtain the regularized field as

$$\tilde{\mathcal{S}}(t) = \sum_{u=1}^{W} \beta_u(t) \tilde{\mathbf{Q}}^{xyzu}(t). \tag{10}$$

From a strict theoretical perspective, one can argue that Eq. 10 violates the time and space factorization we obtained in Eq. 7. However, in practice, the sinc interpolation ensures that $\tilde{\mathbf{Q}}^{xyzu}(t)$ is locally almost constant as long as we choose d to be suitably small, as $\tilde{\mathbf{Q}}^{xyzu}(t)$ cannot then have higher frequencies than $(d-1)/2$. Further, Eq. 10 ensures that $\tilde{\mathcal{S}}(t)$ can only locally traverse within an $\mathbb{R}^W$ subspace where $W < K$, which is a more regularized setting than Eq. 7, where $\mathcal{S}(t)$ is allowed to traverse within an $\mathbb{R}^K$ subspace.

Training

Let $\sigma(\mathbf{x}, t), c(\mathbf{x}, t)$ be density and light values, queried at position $\mathbf{x}$ at time t (obtained via Eq.10). To compute the above values at an arbitrary continuous position $\mathbf{x}$, we tri-linearly interpolate the grids. We perform volumetric rendering as in (Mildenhall et al. 2021) to predict pixel colors $\tilde{p}$ for each training image (see Supplementary for more details). Then, the following loss is minimized for training:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \|p(t) - \tilde{p}(t)\| + \lambda_1 TV(\mathcal{Z}(t)) + \lambda_2 TV(\mathcal{C}(t)), \tag{11}$$

where p is the ground truth pixel color, and $TV(\mathcal{Z}(t))$ and $TV(\mathcal{C}(t))$ are the total variation losses on the density and light fields. λ_1, λ_2 are hyperparameters.

Two important remarks are in order: *a*) our model only requires the TV loss as a loss regularizer, as opposed to multiple explicit regularizations that are used in many existing dynamic NeRF architectures such as explicit foreground-background modeling(Tretschk et al. 2021; Gao et al. 2021), energy-preservation (Park et al. 2021a), or temporal consistency losses (Li et al. 2021; Wang et al. 2021). *b*) To address the insufficiency of neural priors in regularizing the architecture, many dynamic NeRF methods tend to adopt cumbersome training procedures to converge to a good minimum, *e.g.*, sequential training of temporally-ordered frames (Pumarola et al. 2021; Li et al. 2021), coarse-to-fine annealing of hyperparameters (Park et al. 2021a,b), or morphology processing (Yoon et al. 2020). In contrast, we simply randomly sample points in time and space and feed them to the model for training. We argue that this is a strong indicator of the well-built inductive bias/implicit regularization of our architecture and the stability of our formulation.

Experiments

Datasets: We collect four synthetic scenes and four real-world scenes as our dataset. The synthetic scenes include texture changes, lighting changes, scale changes, and long-range movements. The real-world scenes include lighting changes, long-range movements, and spatially concentrated

Method	Cat			Climbing			Flashlight			Flower		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
TensoRF	24.05	0.81	0.35	22.53	0.76	0.34	26.70	0.90	0.36	26.36	0.86	0.29
T-TensoRF	29.45	0.88	0.24	27.08	0.81	0.30	28.93	0.91	0.31	27.10	0.85	0.33
D-NeRF	27.49	0.86	0.30	28.90	0.85	0.27	31.79	**0.95**	0.23	28.56	**0.90**	0.25
NR-NeRF	26.63	0.82	0.35	25.59	0.79	0.35	30.59	0.93	0.29	25.57	0.84	0.37
HyperNeRF	17.13	0.64	0.52	17.66	0.71	0.41	23.51	0.89	0.41	22.74	0.83	0.36
TiNeuVox	22.41	0.81	0.36	24.99	0.81	0.35	31.13	0.91	0.27	28.11	0.88	0.27
BLiRF (ours)	**29.69**	**0.89**	**0.21**	**29.14**	**0.86**	**0.25**	**31.80**	**0.95**	**0.19**	**29.98**	**0.90**	**0.22**
Method	Color Change			Falling and Scale			Light Move			Ball Move		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
TensoRF	19.08	0.89	0.22	17.30	0.86	0.35	19.61	0.79	0.47	24.31	0.94	0.28
T-TensoRF	35.16	**0.97**	0.09	24.30	0.90	0.32	36.49	0.97	**0.10**	28.27	0.96	0.33
D-NeRF	17.42	0.89	0.28	24.60	0.92	0.23	19.15	0.91	0.25	22.58	0.95	0.20
NR-NeRF	16.37	0.89	0.27	15.97	0.86	0.26	18.55	0.91	0.26	23.21	0.95	0.21
HyperNeRF	16.19	0.84	0.35	14.46	0.83	0.34	16.10	0.83	0.40	20.27	0.93	0.27
TiNeuVox	17.01	0.84	0.31	16.19	0.86	0.22	15.01	0.81	0.38	22.41	0.95	0.23
BLiRF (ours)	**36.68**	**0.97**	**0.08**	**35.74**	**0.97**	**0.11**	**38.04**	**0.98**	**0.10**	**39.32**	**0.99**	**0.09**

Table 1: Quantitative comparison of novel view synthesis on our real and synthetic datasets.

	Expressions		Teapot		Chicken		Fist		Banana		Lemon	
	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓
NV	26.7	0.215	26.2	0.216	22.6	0.243	29.3	0.213	24.8	0.209	28.8	0.190
NSFF	26.6	0.283	25.8	**0.210**	27.7	0.173	24.9	0.329	26.1	0.243	28.0	0.283
Nerfies	27.5	0.224	25.7	0.225	28.7	0.141	29.9	0.171	27.9	0.209	30.8	0.223
HyperNeRF	27.9	0.218	**26.4**	0.212	28.7	0.156	**30.7**	**0.150**	**28.4**	0.191	31.8	**0.210**
BLiRF (ours)	**28.2**	**0.213**	26.1	0.215	**29.8**	**0.141**	28.4	0.161	28.2	**0.191**	**32.1**	0.223

Table 2: Comparison on the HyperNeRF dataset. Numbers for the competing methods are extracted from Park et al. (2021b).

Basis	Color Change			Falling and Scale			Light Move			Ball Move		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
DCT	33.99	0.93	0.14	32.61	0.89	0.19	33.77	0.92	0.16	33.59	0.93	0.16
Fourier	31.33	0.89	0.19	29.74	0.89	0.21	31.99	0.91	0.23	33.45	0.94	0.19
Bernstein	27.81	0.86	0.21	28.90	0.87	0.25	31.57	0.91	0.25	33.53	0.94	0.18
Neural	**36.68**	**0.97**	**0.08**	**35.74**	**0.97**	**0.11**	**38.04**	**0.98**	**0.10**	**39.32**	**0.99**	**0.09**

Table 3: Ablation of different time-basis functions. Learned neural basis functions perform best.

dynamic objects (see supplementary). To demonstrate the ability of our method to capture topologically varying deformations, we also evaluate against the HyperNeRF dataset (Park et al. 2021b). We also achieve state of the art results on the NVIDIA dynamic scene dataset and Dycheck dataset, showcasing our model's ability to model deeper real world scenes (see supplementary).

Baselines: We choose D-NeRF (Pumarola et al. 2021), NR-NeRF (Tretschk et al. 2021), TiNeuVox (Fang et al. 2022) and HyperNeRF (Park et al. 2021b) as our main baselines. All are recently proposed Dynamic-NeRF models that adopt the ray deformation paradigm. NR-NeRF comprises an explicit neural network for isolating the motion of a scene, and HyperNeRF consists of separate MLPs for modeling time and space deformations, providing ideal baselines for evaluating the efficacy of our space-time priors. For baselines, we performed a grid search for the optimal hyperparameters for each scene. In contrast, our model uses a **single hyperparameter setting** across all the scenes, demonstrating its robustness (see supplementary for hyperparameter and training details). Further, it is essential to validate whether the superior performance of our model stems from the light/density disentanglement or the space-time factorization. Thus, we design another baseline T-TensoRF, which disentangles the light and density fields, but do not factorize time and space dynamics (see supplementary).

Synthetic Scenes

The synthetic scenes consist of four scenes: *texture change, falling and scale, light move, and ball move*. See supplemen-

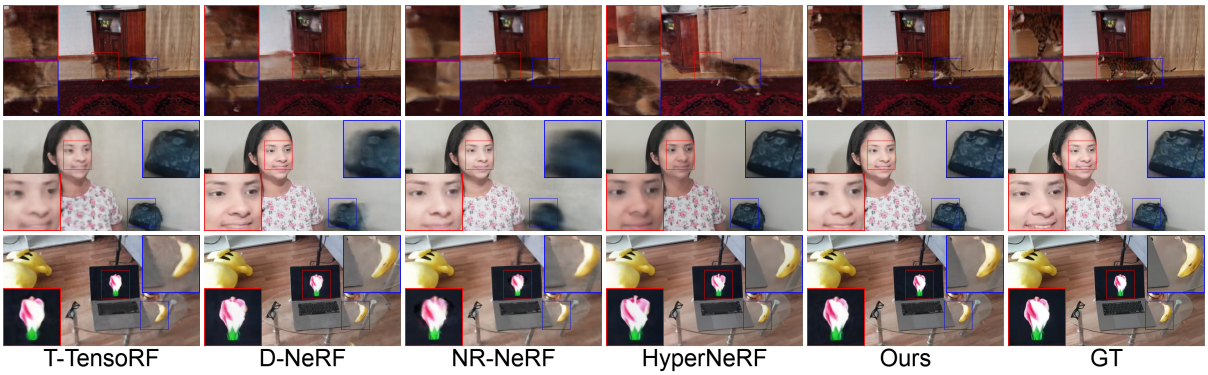

Figure 3: Qualitative comparison on the real-world dataset (zoom-in for a better view). The shown examples represent novel views. Note that in the flashlight scene (second row), D-NeRF, NR-NeRF and T-NeRF fail to capture high-fidelity details in the background. In the cat-walking scene (top row) where the object moves across a considerable range in space, all the baselines fail to recover the moving object accurately. In the flower scene (third row), where the motion is constrained within a small region, the baselines perform fairly well. Our method exhibits superior performance in all the cases.

tary for a qualitative comparison. As shown, D-NeRF, NR-NeRF, and HyperNeRF fail to accurately model the color and light changes. This validates our claim that for full disentanglement of light and density fields, the above methods require a block diagonal Jacobian structure, which is an extremely restrictive condition. Similarly, they tend to deform the objects when scale changes and long-range movements are present. T-TensoRF, due its ability to disentangle light and density fields, adequately recovers light/texture changes. However, all the baselines fail to accurately learn the 3D positions of the objects showcasing their inability to precisely disentangle camera and scene dynamics. In comparison, our method achieves significantly superior results in all above aspects. See Table 1 for quantitative results.

Real-World Scenes

The real-world scenes contain four scenes; *cat walking, flashlight, flower, and climbing*. Cat walking and climbing scenes contain long-range movements. See Fig. 3 and Fig. 1 for qualitative comparisons on these scenes. When long-range movements are present, the baselines fail to recover the high-fidelity details of the moving objects. In the flashlight scene, baselines fail to accurately capture granular details in the background or light change. In the flower scene, where the dynamics are concentrated spatially, the baselines perform well. Our method generates better results in all of the aforementioned aspects. Note that D-NeRF, NR-NeRF can model lighting changes as shown in the flashlight scene (see also supplementary), validating our insight that ray deformation models indeed encode density and light field dynamics. Table 1 depicts quantitative results. (See supplementary for comparisons over NVIDIA dynamic scene dataset and Dycheck dataset).

Topologically Varying Scenes

Park et al. (2021b) showed that most existing dynamic NeRF methods cannot model topologically varying scenes effectively. To remedy this, they proposed a method that models discontinuities of the evolving field as continuous deformations, using a collection of MLPs. In contrast, our method can implicitly model such scenes since we model the evolution of each 3D point in the fields as bandlimited signals. To showcase this, we conduct experiments on the topologically varying interpolation dataset provided by (Park et al. 2021b). In contrast to our dataset, these scenes exhibit a camera revolving around quasi-repeated actions which adhere to the camera motion and object centric biases baked into ray-deformation methods, and do not include long-range motions or light/texture changes. The results are depicted in Table 2. As is shown, we achieve near-identical[2] or better results compared to baselines.

Ablation Study

We compare other possible time-basis functions that are complete in $L^2(\mathbb{R}, dt)$ against the neural trajectory basis. Table 3 presents a quantitative comparison with the DCT, Fourier, and Bernstein bases. Although these basis functions are also capable of providing acceptable results, neural basis performs best. We provide ablations for other design choices as well; manifold regularization, # basis functions, neural prior, and low rank factorization in Supplementary. T-TensoRF demonstrates the effect of factorization of space-time dynamics.

Conclusion

We offer a novel framework for modeling dynamic 3D scenes allowing factorization of the space and time dynamics. This presents a platform to impose well-designed space-time priors on NeRF, enabling high-fidelity novel view synthesis of dynamics scenes. Finally, we present an implementation that demonstrates compelling results across complex dynamics scenes containing long-range movements, scale changes, and light/texture changes.

[2]HyperNeRF(Park et al. 2021b) uses distortion coefficients to correct the rays, we omit this detail from our implementation to maintain a fair comparison with the other baselines.

References

Agudo, A.; and Moreno-Noguer, F. 2017. Dust: Dual union of spatio-temporal subspaces for monocular multiple object 3d reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6262–6270.

Avidan, S.; and Shashua, A. 1997. Novel view synthesis in tensor space. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1034–1040. IEEE.

Avidan, S.; and Shashua, A. 2000. Trajectory triangulation: 3D reconstruction of moving points from a monocular image sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(4): 348–357.

Azuma, R. T. 1997. A survey of augmented reality. *Presence: teleoperators & virtual environments*, 6(4): 355–385.

Burdea, G. C.; and Coiffet, P. 2003. *Virtual reality technology*. John Wiley & Sons.

Chen, A.; Xu, Z.; Geiger, A.; Yu, J.; and Su, H. 2022. TensoRF: Tensorial Radiance Fields. *arXiv preprint arXiv:2203.09517*.

Daribo, I.; and Pesquet-Popescu, B. 2010. Depth-aided image inpainting for novel view synthesis. In *2010 IEEE International workshop on multimedia signal processing*, 167–170. IEEE.

Davison, A. J.; Reid, I. D.; Molton, N. D.; and Stasse, O. 2007. MonoSLAM: Real-Time Single Camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6): 1052–1067.

Dou, M.; Fuchs, H.; and Frahm, J.-M. 2013. Scanning and tracking dynamic objects with commodity depth cameras. In *2013 IEEE international symposium on mixed and augmented Reality (ISMAR)*, 99–106. Ieee.

Dou, M.; Khamis, S.; Degtyarev, Y.; Davidson, P.; Fanello, S. R.; Kowdle, A.; Escolano, S. O.; Rhemann, C.; Kim, D.; Taylor, J.; et al. 2016. Fusion4d: Real-time performance capture of challenging scenes. *ACM Transactions on Graphics (ToG)*, 35(4): 1–13.

Dyer, C. R. 2001. Volumetric scene reconstruction from multiple views. In *Foundations of image understanding*, 469–489. Springer.

Fang, J.; Yi, T.; Wang, X.; Xie, L.; Zhang, X.; Liu, W.; Nießner, M.; and Tian, Q. 2022. Fast Dynamic Radiance Fields with Time-Aware Neural Voxels. In *SIGGRAPH Asia 2022 Conference Papers*.

Gao, C.; Saraf, A.; Kopf, J.; and Huang, J.-B. 2021. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5712–5721.

Grisetti, G.; Kümmerle, R.; Stachniss, C.; and Burgard, W. 2010. A tutorial on graph-based SLAM. *IEEE Intelligent Transportation Systems Magazine*, 2(4): 31–43.

Johnson, E.; Habermann, M.; Shimada, S.; Golyanik, V.; and Theobalt, C. 2022. Unbiased 4D: Monocular 4D Reconstruction with a Neural Deformation Model. *arXiv preprint arXiv:2206.08368*.

Kolmogorov, V.; and Zabih, R. 2002. Multi-camera scene reconstruction via graph cuts. In *European conference on computer vision*, 82–96. Springer.

Kumar, S.; Dai, Y.; and Li, H. 2017. Spatio-temporal union of subspaces for multi-body non-rigid structure-from-motion. *Pattern Recognition*, 71: 428–443.

Li, Z.; Niklaus, S.; Snavely, N.; and Wang, O. 2021. 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*, 6498–6508.

Martin-Brualla, R.; Radwan, N.; Sajjadi, M. S.; Barron, J. T.; Dosovitskiy, A.; and Duckworth, D. 2021. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7210–7219.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106.

Mur-Artal, R.; Montiel, J. M. M.; and Tardos, J. D. 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE transactions on robotics*, 31(5): 1147–1163.

Newcombe, R. A.; Fox, D.; and Seitz, S. M. 2015. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 343–352.

Newcombe, R. A.; Izadi, S.; Hilliges, O.; Molyneaux, D.; Kim, D.; Davison, A. J.; Kohi, P.; Shotton, J.; Hodges, S.; and Fitzgibbon, A. 2011. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, 127–136. Ieee.

Niemeyer, M.; Mescheder, L.; Oechsle, M.; and Geiger, A. 2019. Occupancy flow: 4d reconstruction by learning particle dynamics. In *Proceedings of the IEEE/CVF international conference on computer vision*, 5379–5389.

Park, K.; Sinha, U.; Barron, J. T.; Bouaziz, S.; Goldman, D. B.; Seitz, S. M.; and Martin-Brualla, R. 2021a. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5865–5874.

Park, K.; Sinha, U.; Hedman, P.; Barron, J. T.; Bouaziz, S.; Goldman, D. B.; Martin-Brualla, R.; and Seitz, S. M. 2021b. HyperNeRF: A Higher-Dimensional Representation for Topologically Varying Neural Radiance Fields. *ACM Trans. Graph.*, 40(6).

Pumarola, A.; Corona, E.; Pons-Moll, G.; and Moreno-Noguer, F. 2021. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10318–10327.

Rabaud, V.; and Belongie, S. 2008. Re-thinking non-rigid structure from motion. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 1–8. IEEE.

Slavcheva, M.; Baust, M.; Cremers, D.; and Ilic, S. 2017. Killingfusion: Non-rigid 3d reconstruction without correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1386–1395.

Torresani, L.; Hertzmann, A.; and Bregler, C. 2003. Learning non-rigid 3D shape from 2D motion. *Advances in neural information processing systems*, 16.

Torresani, L.; Yang, D. B.; Alexander, E. J.; and Bregler, C. 2001. Tracking and modeling non-rigid objects with rank constraints. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, I–I. IEEE.

Tretschk, E.; Tewari, A.; Golyanik, V.; Zollhöfer, M.; Lassner, C.; and Theobalt, C. 2021. 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*, 12959–12970.

Tung, T.; Nobuhara, S.; and Matsuyama, T. 2009. Complete multi-view reconstruction of dynamic scenes from probabilistic fusion of narrow and wide baseline stereo. In *2009 IEEE 12th International Conference on Computer Vision*, 1709–1716. IEEE.

Valmadre, J.; and Lucey, S. 2012. General trajectory prior for non-rigid reconstruction. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 1394–1401. IEEE.

Wang, C.; Eckart, B.; Lucey, S.; and Gallo, O. 2021. Neural trajectory fields for dynamic novel view synthesis. *arXiv preprint arXiv:2105.05994*.

Wexler, Y.; and Shashua, A. 2000. On the synthesis of dynamic scenes from reference views. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 1, 576–581. IEEE.

Xian, W.; Huang, J.-B.; Kopf, J.; and Kim, C. 2021. Spacetime neural irradiance fields for free-viewpoint video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9421–9431.

Xu, T.; and Harada, T. 2022. Deforming Radiance Fields with Cages. In *European Conference on Computer Vision*, 159–175. Springer.

Yoon, J. S.; Kim, K.; Gallo, O.; Park, H. S.; and Kautz, J. 2020. Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5336–5345.

Yu, T.; Guo, K.; Xu, F.; Dong, Y.; Su, Z.; Zhao, J.; Li, J.; Dai, Q.; and Liu, Y. 2017. Bodyfusion: Real-time capture of human motion and surface geometry using a single depth camera. In *Proceedings of the IEEE International Conference on Computer Vision*, 910–919.

Zappella, L.; Del Bue, A.; Lladó, X.; and Salvi, J. 2013. Joint estimation of segmentation and structure from motion. *Computer Vision and Image Understanding*, 117(2): 113–129.

Zhang, K.; Riegler, G.; Snavely, N.; and Koltun, V. 2020. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*.

Zhang, L.; Curless, B.; and Seitz, S. M. 2003. Spacetime stereo: Shape recovery for dynamic scenes. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, II–367. IEEE.

Zhang, L.; Snavely, N.; Curless, B.; and Seitz, S. M. 2004. Spacetime faces: high resolution capture for modeling and animation. In *ACM SIGGRAPH 2004 Papers*, 548–558.

Zhu, Y.; Huang, D.; De La Torre, F.; and Lucey, S. 2014. Complex non-rigid motion 3d reconstruction by union of subspaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1542–1549.