Neural 3D Video Synthesis from Multi-view Video

Tianye Li[1,2,*] Mira Slavcheva[2,*] Michael Zollhoefer[2]
Simon Green[2] Christoph Lassner[2] Changil Kim[3] Tanner Schmidt[2]
Steven Lovegrove[2] Michael Goesele[2] Richard Newcombe[2] Zhaoyang Lv[2]
[1]University of Southern California [2]Reality Labs Research [3]Meta

Figure 1. We propose a novel method for representing and rendering high quality 3D video. Our method trains a novel and compact dynamic neural radiance field (DyNeRF) in an efficient way. Our method demonstrates near photorealistic dynamic novel view synthesis for complex scenes including challenging scene motions and strong view-dependent effects. We demonstrate three synthesized 3D video, and show the associated high quality geometry in the heatmap visualization in each top right corner. **Please see the supplemental materials for the high-quality video and additional information**.

Abstract

We propose a novel approach for 3D video synthesis that is able to represent multi-view video recordings of a dynamic real-world scene in a compact, yet expressive representation that enables high-quality view synthesis and motion interpolation. Our approach takes the high quality and compactness of static neural radiance fields in a new direction: to a model-free, dynamic setting. At the core of our approach is a novel time-conditioned neural radiance field that represents scene dynamics using a set of compact latent codes. We are able to significantly boost the training speed and perceptual quality of the generated imagery by a novel hierarchical training scheme in combination with ray importance sampling. Our learned representation is highly compact and able to represent a 10 second 30 FPS multi-view video recording by 18 cameras with a model size of only 28MB. We demonstrate that our method can render high-fidelity wide-angle novel views at over 1K resolution, even for complex and dynamic scenes. We perform an extensive qualitative and quantitative evaluation that shows that our approach outperforms the state of the art. Project website: `https://neural-3d-video.github.io/`.

∗ Equal contribution. TL's work was done during an internship at Reality Labs Research.

1. Introduction

Photorealistic representation and rendering of dynamic real-world scenes are highly challenging research topics, yet with many important applications that range from movie production to virtual and augmented reality. Dynamic real-world scenes are notoriously hard to model using classical mesh-based representations, since they often contain thin structures, semi-transparent objects, specular surfaces, and topology that constantly evolves over time due to the often complex scene motion of multiple objects and people.

In theory, the 6D plenoptic function $P(\mathbf{x}, \mathbf{d}, t)$ is a suitable representation for this rendering problem, as it completely explains our visual reality and enables rendering every possible view at every moment in time [1]. Here, $\mathbf{x} \in \mathbb{R}^3$ is the camera position in 3D space, $\mathbf{d} = (\theta, \phi)$ is the viewing direction, and t is time. Thus, fully measuring the plenoptic function requires placing an omnidirectional camera at every position in space at every possible time.

Neural radiance fields (NeRF) [38] offer a way to circumvent this problem: instead of directly encoding the plenoptic function, they encode the radiance field of the scene in an implicit, coordinate-based function, which can be sampled through ray casting to approximate the plenoptic function. However, the ray casting, which is required to train and to render a neural radiance field, involves hundreds of MLP evaluations for *each* ray. While this might be acceptable for a static snapshot of a scene, directly reconstructing a dynamic scene as a sequence of per-frame neural radiance fields would be prohibitive as both storage and training time increase linearly with time. For example, to represent a 10 second, 30 FPS multi-view video recording by 18 cameras, which we later demonstrate with our method, a per-frame NeRF would require about 15 000 GPU hours in training and about 1 GB in storage. More importantly, such obtained representations would only reproduce the world as a discrete set of snapshots, lacking any means to reproduce the world in-between. On the other hand, Neural Volumes [32] is able to handle dynamic objects and even renders at interactive frame rates. Its limitation is the dense uniform voxel grid that limits resolution and/or size of the reconstructed scene due to the inherent $O(n^3)$ memory complexity.

In this paper, we propose a novel approach for 3D video synthesis of complex, dynamic real-world scenes that enables high-quality view synthesis and motion interpolation while being compact. Videos typically consist of a time-invariant component under stable lighting and a continuously changing time-variant component. This dynamic component typically exhibits locally correlated geometric deformations and appearance changes between frames. By exploiting this fact, we propose to reconstruct a dynamic neural radiance field based on two novel contributions.

First, we extend neural radiance fields to the space-time domain. Instead of directly using time as input, we parameterize scene motion and appearance changes by a set of compact latent codes. Compared to the more obvious choice of an additional 'time coordinate', the learned latent codes show more expressive power, allowing for recording the vivid details of moving geometry and texture. They also allow for smooth interpolation in time, which enables visual effects such as slow motion or 'bullet time'. Second, we propose novel importance sampling strategies for dynamic radiance fields. Ray-based training of neural scene representations treats each pixel as an independent training sample and requires thousands of iterations to go through all pixels observed from all views. However, captured dynamic video often exhibits a small amount of pixel change between frames. This opens up an opportunity to significantly boost the training progress by selecting the pixels that are most important for training. Specifically, in the time dimension, we schedule training with coarse-to-fine hierarchical sampling in the frames. In the ray/pixel dimension, our design tends to sample those pixels that are more time-variant than others. These strategies allow us to shorten the training time of long sequences significantly, while retaining high quality reconstruction results. We demonstrate our approach using a multi-view rig based on 18 GoPro cameras. We show results on multiple challenging dynamic environments with highly complex view-dependent and time-dependent effects. Compared to the naïve per-frame NeRF baseline, we show that with our combined temporal and spatial importance sampling we achieve one order of magnitude acceleration in training speed, with a model that is 40 times smaller in size for 10 seconds of a 30 FPS 3D video. In summary we make the following contributions:

- We propose a novel dynamic neural radiance field based on temporal latent codes that achieves high quality 3D video synthesis of complex, dynamic real-world scenes.
- We present novel training strategies based on hierarchical training and importance sampling in the spatiotemporal domain, which boost training speed significantly and lead to higher quality results for longer sequences.
- We provide our datasets of time-synchronized and calibrated multi-view videos that covers challenging 4D scenes for research purposes*.

2. Related Work

Our work is related to several research domains, such as novel view synthesis for static scenes, 3D video synthesis for dynamic scenes, image-based rendering, and neural rendering approaches. For a detailed discussion of neural rendering applications and neural scene representations, we refer to the surveys [54] and [55].

Novel View Synthesis for Static Scenes. Novel view synthesis has been tackled by explicitly reconstructing textured 3D models of the scene and rendering from arbitrary viewpoints. Multi-view stereo [15, 49] and visual hull reconstructions [13, 27] have been successfully employed. Complex view-dependent effects can be captured by light transport acquisition methods [11, 59]. Learning-based methods have been proposed to relax the high number of required views and to accelerate the inference speed for geometry reconstruction [19, 24, 61] and appearance capture [5, 35], or combined reconstruction techniques [39, 62]. Novel view synthesis can also be achieved by reusing input image pixels. Early works using this approach interpolate the viewpoints [8]. The Light Field/Lumigraph method [10, 18, 28, 41] resamples input image rays to generate novel views. One drawback of these approaches is that it require dense sampling for high quality rendering of complex scenes. More recently, [14, 22, 37, 51, 66] learn to fuse and resample pixels from reference views using neural networks. Neural Radiance Fields (NeRFs) [38] train an MLP-based radiance and opacity field and achieve state-of-the-art quality for novel view synthesis. Other approaches [36, 58] employ an explicit point-based scene representation combined with a screen space neural network for hole filling. [26] push this further and encode the scene appearance in a differentiable sphere-based representation. [50] employs

*https://github.com/facebookresearch/Neural_3D_Video a dense voxel grid of features in combination with a screen space network for view synthesis. All these methods are excellent at interpolating views for static scenes, but it is unclear how to extend them to the dynamic setting.

3D Video Synthesis for Dynamic Scenes. Techniques in this category enable view synthesis for dynamic scenes and might also enable interpolation across time. For video synthesis, [23] pioneers in showing the possibility of explicitly capture geometry and textures. [67] proposes a temporal layered representation that can be compressed and replayed at an interactive rate. Reconstruction and animation is particularly well studied for humans [7,20,52], but is usually performed model-based and/or only works with high-end capture setups. [29] captures temporally consistent surfaces by tracking and completion. [9] proposes a system for capturing and compressing streamable 3D video with high-end hardware. More recently, learning-based methods such as [21] achieve volumetric video capture for human performances from sparse camera views. [3] focus on more general scenes. They decompose them into a static and dynamic component, re-project information based on estimated coarse depth, and employ a U-Net in screen space to convert the intermediate result to realistic imagery. [4] uses a neural network for space-time and illumination interpolation. [63] uses a model-based step for merging the estimated depth maps to a unified representation that can be rendered from novel views. Neural Scene Flow Fields [30] incorporates a static background model. Space-time Neural Irradiance Fields [60] employs video depth estimation to supervise a space-time radiance field. [17] recently proposes a time-conditioned radiance field, supervised by its own predicted flow vectors. These works have limited view angle due to their single-view setting and require additional supervision, such as depth or flow. [12, 42, 45, 56] explicitly model dynamic scenes by a warp field or velocity field to deform a canonical radiance field. STaR [64] models scenes of rigidly moving objects using several canonical radiance fields that are rigidly transformed. These methods cannot model challenging dynamic events such as topology changes. Several radiance field approaches have been proposed for modeling digital humans [16, 31, 40, 44, 46], but they can not directly be applied to general non-rigid scenes. Furthermore, there have been efforts in improving neural radiance fields for in-the-wild scenes [34], generalization across scenes. HyperNeRF [43] is a concurrent work on dynamic novel view synthesis, but they focus on monocular video in a short sequence. Neural Volumes [32] employs volume rendering in combination with a view-conditioned decoder network to parameterize dynamic sequences of single objects. Their results are limited in resolution and scene complexity due to the inherent $O(n^3)$ memory complexity. [6] enable 6DoF video for VR applications based on independent alpha-textured meshes that can be streamed at the rate of hundreds of Mb/s. This approach employs a capture setup with 46 cameras and requires a large training dataset to construct a strong scene-prior. In contrast, we seek a unified space-time representation that enables continuous viewpoint and time interpolation, while being able to represent an entire multi-view video sequence of 10 seconds in as little as 28MB.

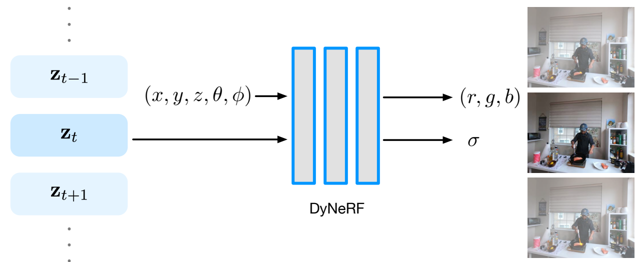

Figure 2. We learn the 6D plenoptic function by our novel dynamic neural radiance field (DyNeRF) that conditions on position, view direction and a compact, yet expressive time-variant latent code.

3. DyNeRF: Dynamic Neural Radiance Fields

We address the problem of reconstructing dynamic 3D scenes from time-synchronized multi-view videos with known intrinsic and extrinsic parameters. The representation we aim to reconstruct from such multi-camera recordings should allow us to render photorealistic images from a wide range of viewpoints at arbitrary points in time.

Building on NeRF [38], we propose *dynamic neural radiance fields (DyNeRF)* that are directly optimized from input videos captured with multiple video cameras. DyNeRF is a novel continuous space-time neural radiance field representation, controllable by a series of temporal latent embeddings that are jointly optimized during training. Our representation compresses a huge volume of input videos from multiple cameras to a compact 6D representation that can be queried continuously in both space and time. The learned embedding faithfully captures detailed temporal variations of the scene, such as complex photometric and topological changes, without explicit geometric tracking.

3.1. Representation

The problem of representing 3D video comprises learning the 6D plenoptic function that maps a 3D position $\mathbf{x} \in \mathbb{R}^3$, direction $\mathbf{d} \in \mathbb{R}^2$, and time $t \in \mathbb{R}$, to RGB radiance $\mathbf{c} \in \mathbb{R}^3$ and opacity $\sigma \in \mathbb{R}$. Based on NeRF [38], which approximates the 5D plenoptic function of a static scene with a learnable function, a potential solution would be to add a time dependency to the function:

$$F_\Theta : (\mathbf{x}, \mathbf{d}, t) \longrightarrow (\mathbf{c}, \sigma) \quad , \tag{1}$$

which is realized by a Multi-Layer Perceptron (MLP) with trainable weights Θ. The 1-dimensional time variable t can be mapped via positional encoding [53] to a higher dimensional space, in a manner similar to how NeRF handles the inputs $\mathbf{x}$ and $\mathbf{d}$. However, we empirically found that it is challenging for this design to capture complex dynamic 3D scenes with challenging topological changes and time-dependent volumetric effects, such as flames.

Dynamic Neural Radiance Fields. We model the dynamic scene by time-variant latent codes $\mathbf{z}_t \in \mathbb{R}^D$, as shown in Fig. 2. We learn a set of time-dependent latent codes, indexed by a discrete time variable t:

$$F_\Theta : (\mathbf{x}, \mathbf{d}, \mathbf{z}_t) \longrightarrow (\mathbf{c}, \sigma) \ . \tag{2}$$

The latent codes provide a compact representation of the state of a dynamic scene at a certain time, which can handle various complex scene dynamics, including deformation, topological and radiance changes. We apply positional encoding [53] to the input position coordinates to map them to a higher-dimensional vector. However, no positional encoding is applied to the time-dependent latent codes. Before training, the latent codes $\{\mathbf{z}_t\}$ are randomly initialized independently across all frames.

Rendering. We use volume rendering techniques to render the radiance field given a query view in space and time. Given a ray $\mathbf{r}(s) = \mathbf{o} + s\mathbf{d}$ with the origin $\mathbf{o}$ and direction $\mathbf{d}$ defined by the specified camera pose and intrinsics, the rendered color of the pixel corresponding to this ray $\mathbf{C}(\mathbf{r})$ is an integral over the radiance weighted by accumulated opacity [38]:

$$\mathbf{C}^{(t)}(\mathbf{r}) = \int_{s_n}^{s_f} T(s)\sigma(\mathbf{r}(s), \mathbf{z}_t)\mathbf{c}(\mathbf{r}(s), \mathbf{d}, \mathbf{z}_t))\, ds \ . \tag{3}$$

where s_n and s_f denote the bounds of the volume depth range and the accumulated opacity $T(s) = \exp(-\int_{s_n}^{s} \sigma(\mathbf{r}(p), \mathbf{z}_t))\, dp)$. We apply a hierarchical sampling strategy as [38] with stratified sampling on the coarse level followed by importance sampling on the fine level.

Loss Function. The network parameters Θ and the latent codes $\{\mathbf{z}_t\}$ are simultaneously trained by minimizing the ℓ_2-loss between the rendered colors $\hat{\mathbf{C}}(\mathbf{r})$ and the ground truth colors $\mathbf{C}(\mathbf{r})$, and summed over all rays $\mathbf{r}$ that correspond to the image pixels from all training camera views $\mathcal{R}$ and throughout all time frames $t \in \mathcal{T}$ of the recording:

$$\mathcal{L} = \sum_{t\in\mathcal{T}, \mathbf{r}\in\mathcal{R}} \sum_{j\in\{c,f\}} \left\| \hat{\mathbf{C}}_j^{(t)}(\mathbf{r}) - \mathbf{C}^{(t)}(\mathbf{r}) \right\|_2^2 \ . \tag{4}$$

We evaluate the loss at both the coarse and the fine level, denoted by $\hat{\mathbf{C}}_c^{(t)}$ and $\hat{\mathbf{C}}_f^{(t)}$ respectively, similar to NeRF. We train with a stochastic version of this loss function, by randomly sampling ray data and optimizing the loss of each ray batch. Please note that our dynamic radiance field is trained with this plain ℓ_2-loss without any special regularization.

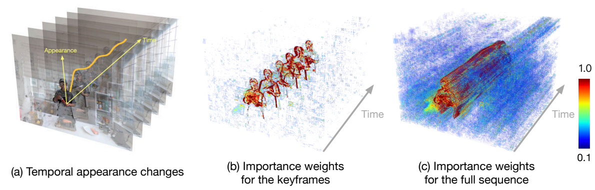

Figure 3. **Overview of our efficient training strategies.** We perform hierarchical training first using keyframes (b) and then on the full sequence (c). At both stages, we apply the ray importance sampling technique to focus on the rays with high time-variant information based on weight maps that measure the temporal appearance changes (a). We show a visualized example of the sampling probability based on global median map using a heatmap (red and opaque means high probability).

3.2. Efficient Training

An additional challenge of ray casting–based neural rendering on video data is the large amount of training time required. The number of training iterations per epoch scales linearly with the total number of pixels in the input multi-view videos. For a 10 second, 30 FPS, 1 MP multi-view video sequence from 18 cameras, there are about 7.4 billion ray samples in one epoch, which would take about half a week to process using 8 NVIDIA Volta class GPUs. Given that each ray needs to be re-visited several times to obtain high quality results, this sampling process is one of the biggest bottlenecks for ray-based neural reconstruction methods to train 3D videos at scale.

However, for a natural video a large proportion of the dynamic scene is either time-invariant or only contains a small time-variant radiance change at a particular timestamp across the entire observed video. Hence, uniformly sampling rays causes an imbalance between time-invariant observations and time-variant ones. This means it is highly inefficient *and* impacts reconstruction quality: time-invariant regions reach high reconstruction quality sooner and are uselessly oversampled, while time-variant regions require additional sampling, increasing the training time.

To explore temporal redundancy in the context of 3D video, we propose two strategies to accelerate the training process (see Fig. 3): (1) hierarchical training that optimizes data over a coarse-to-fine frame selection and (2) importance sampling that prefers rays around regions of higher temporal variance. In particular, these strategies form a different loss function by paying more attention to the "important" rays in time frame set $\mathcal{S}$ and pixel set $\mathcal{I}$ for training:

$$\mathcal{L}_{\text{efficient}} = \sum_{t\in\mathcal{S}, \mathbf{r}\in\mathcal{I}} \sum_{j\in\{c,f\}} \left\| \hat{\mathbf{C}}_j^{(t)}(\mathbf{r}) - \mathbf{C}^{(t)}(\mathbf{r}) \right\|_2^2 \ . \tag{5}$$

These two strategies combined can be regarded as an adaptive sampling approach, contributing to significantly faster training and improved rendering quality.

Hierarchical Training. Instead of training DyNeRF on all video frames, we first train it on keyframes, which we sample all images equidistantly at fixed time intervals K, i.e. $\mathcal{S} = \{t \mid t = nK, n \in \mathbb{Z}^+, t \in \mathcal{T}\}$. Once the model converges with keyframe supervision, we use it to initialize the final model, which has the same temporal resolution as the full video. Since the per-frame motion of the scene within each segment (divided by neighboring keyframes) is smooth, we initialize the fine-level latent embeddings by linearly interpolating between the coarse embeddings. Finally, we train using data from all the frames jointly, $\mathcal{S} = \mathcal{T}$, further optimizing the network weights and the latent embeddings. The coarse keyframe model has already captured an approximation of the time-invariant information across the video. Therefore, the fine full-frame training only needs to learn the time-variant information per-frame.

Ray Importance Sampling. We propose to sample rays $\mathcal{I}$ across time with different importance based on the temporal variation in the input videos. For each observed ray $\mathbf{r}$ at time t, we compute a weight $\omega^{(t)}(\mathbf{r})$. In each training iteration we pick a time frame t at random. We first normalize the weights of the rays across all input views for frame t, and then apply inverse transform sampling to select rays based on these weights.

To calculate the weight of each ray, we propose three implementations based on different insights.

- **Global-Median (DyNeRF-ISG)**: We compute the weight of each ray based on the residual difference of its color to its the global median value across time.
- **Temporal-Difference (DyNeRF-IST)**: We compute the weight of each ray based on the color difference in two consecutive frames.
- **Combined Method (DyNeRF-IS*)**: Combine both strategies above.

We empirically observed that training DyNeRF-ISG with a high learning rate leads to very quick recovery of dynamic detail, but results in some jitter across time. On the other hand, training DyNeRF-IST with a low learning rate produces a smooth temporal sequence which is still somewhat blurry. Thus, we combine the benefits of both methods in our final strategy, DyNeRF-IS* (referred as DyNeRF in later sections), which first obtains sharp details via DyNeRF-ISG and then smoothens the temporal motion via DyNeRF-IST. We explain the details of the three strategies in the *Supp. Mat.* All importance sampling methods assume a static camera rig.

4. Experiments

We demonstrate our approach on a large variety of captured daily events with challenging scene motions, varying illuminations and self-cast shadows, view-dependent appearances and highly volumetric effects. We performed detailed ablation studies and comparisons to various baselines on our multi-view data and immersive video data [6].

Supplementary materials. We strongly recommend the reader to watch our *supplemental video* to better judge the photorealism of our approach at high resolution, which cannot be represented well by the metrics. We demonstrate interactive playback of our 3D videos in commodity VR headset *Quest 2* in the *supplemental video*. We further provide comprehensive details of our capture setup, dataset descriptions, comparison settings, more ablations studies on parameter choices and failure case discussions.

4.1. Evaluation Settings

Plenoptic Video Datasets. We build a mobile multi-view capture system using 21 GoPro Black Hero 7 cameras. We capture videos at a resolution of 2028×2704 (2.7K) and frame rate of 30 FPS. The multi-view inputs are time-synchronized. We obtain the camera intrinsic and extrinsic parameters using COLMAP [48]. We employ 18 views for training, and 1 view for qualitative and quantitative evaluations for all datasets except one sequence observing multiple people moving, which uses 14 training views. For more details on the capture setup, please refer to the *Supp. Mat.*

Our captured data demonstrates a variety of challenges for video synthesis, including (1) objects of high specularity, translucency and transparency, (2) scene changes and motions with changing topology (poured liquid), (3) self-cast moving shadows, (4) volumetric effects (fire flame), (5) an entangled moving object with strong view-dependent effects (the torch gun and the pan), (6) various lighting conditions (daytime, night, spotlight from the side), and (7) multiple people moving around in open living room space with outdoor scenes seen through transparent windows with relatively dark indoor illumination. Our collected data can provide sufficient synchronized camera views for high quality 4D reconstruction of challenging dynamic objects and view-dependent effects in a natural daily indoor environment, which, to our knowledge, did not exist in public 4D datasets. We will release the datasets for research purposes.

Immersive Video Datasets. We also demonstrate the generality of our method using the multi-view videos from [6] directly trained on their fisheye video input.

Baselines. We compare to the following baselines:

- **Multi-View Stereo (MVS)**: frame-by-frame rendering of the reconstructed and textured 3D meshes using commercial software RealityCapture†.
- **Local Light Field Fusion (LLFF)** [37]: frame-by-frame rendering of the LLFF-produced multiplane images with the pretrained model‡.
- **NeuralVolumes (NV)** [32]: One prior-art volumetric video rendering method using a warped canonical model.

†https://www.capturingreality.com/
‡https://github.com/Fyusion/LLFF

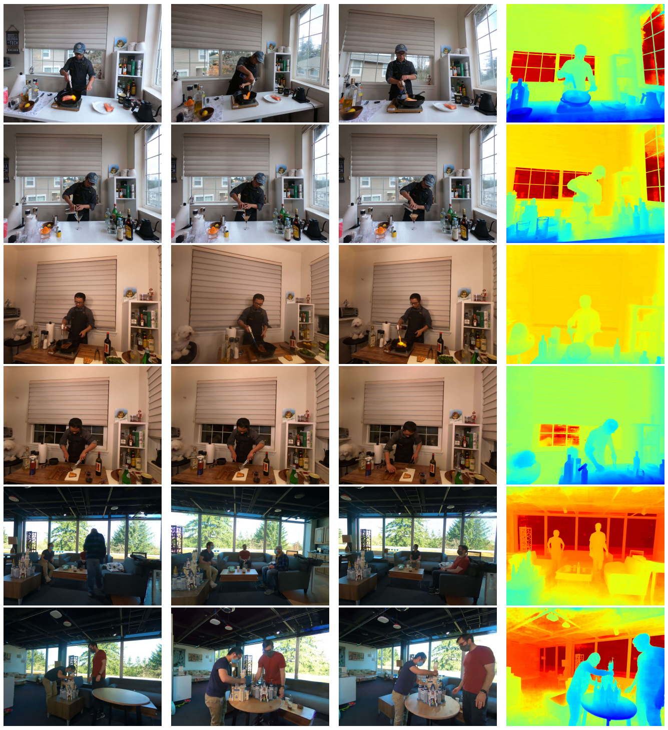

Figure 4. **High-quality novel view videos** synthesized by our approach for dynamic real-world scenes. We visualize normalized depth in color space on the last column in the each row. Our representation is compact, yet expressive and even handles complex specular reflections and translucency.

We follow the same setting as the original paper.

- **NeRF-T**: a temporal NeRF baseline as described in Eq. 1.
- **DyNeRF**†: An ablation setting of DyNeRF without our proposed hierarchical training and importance sampling.

Due to page limit, we provide more ablation analysis of our importance sampling strategies and latent code dimension in Supp. Mat.

Metrics. We evaluate the rendering quality on test view and the following quantitative metrics: (1) Peak signal-to-noise ratio (PSNR); (2) Mean square error (MSE); (3) Structural dissimilarity index measure (DSSIM) [47,57] (4) Perceptual quality measure LPIPS [65]; (5) Perceived error difference FLIP [2]; (6) Just-Objectionable-Difference (JOD) [33]. Higher PSNR and scores indicate better reconstruction quality and higher JOD represents less visual difference compared to the reference video. For all other metrics, lower numbers indicate better quality.

For any video of length shorter than 60 frames, we evaluate the model frame-by-frame on the complete video. Considering the significant amount required for high resolution rendering, we evaluate the model every 10 frames to calculate the frame-by-frame metrics reported for any video of length equal or longer than 300 frames in Tab. 1. For video

Table 1. **Quantitative comparison** of our proposed method to baselines of existing methods and radiance field baselines trained at 200K iterations on a 10-second sequence.

Method	PSNR ↑	MSE ↓	DSSIM ↓	LPIPS ↓	FLIP ↓
MVS	19.1213	0.01226	0.1116	0.2599	0.2542
NeuralVolumes	22.7975	0.00525	0.0618	0.2951	0.2049
LLFF	23.2388	0.00475	0.0762	0.2346	0.1867
NeRF-T	28.4487	0.00144	0.0228	0.1000	0.1415
DyNeRF†	28.4994	0.00143	0.0231	0.0985	0.1455
DyNeRF	**29.5808**	**0.00110**	**0.0197**	**0.0832**	**0.1347**

metric JOD which requires a stack of continuous video frames, we evaluate the model on the whole sequence reported in Tab. 2. We verified on 2 video sequences with a frame length of 300 that the PSNR differs by at most 0.02 comparing evaluating them every 10th frame vs. on all frames. We evaluate all the models at 1K resolution, and report the average of the result from every evaluated frame.

Implementation Details. We implement our approach in PyTorch. We use the same MLP architecture as in NeRF [38] except that we use 512 activations for the first 8 MLP layers instead of 256. We employ 1024-dimensional latent codes. In the hierarchical training we first only train on keyframes that are $K = 30$ frames apart. We employ the Adam optimizer [25] with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. In the keyframe training stage, we set a learning rate of $5\mathrm{e}{-4}$ and train for $300K$ iterations. We include the details on the important sampling scheme in the Supp. Mat. We set the latent code learning rate to be $10\times$ higher than for the other network parameters. The per-frame latent codes are initialized from $\mathcal{N}(0, \frac{0.01}{\sqrt{D}})$, where $D = 1024$. The total training takes about a week with 8 NVIDIA V100 GPUs and a total batch size of 24576 rays.

4.2. Results

We demonstrate our novel view rendering results on different sequences in Fig. 1 and Fig. 4. Our method can represent a 30 FPS multi-view video of up to 10 seconds in length with at high quality. Our reconstructed model can enable near photorealistic continuous novel-view rendering at 1K resolution. In the Supp. Video, we render special visual effects such as slow motion by interpolating sub-frame latent codes between two discrete time-dependent latent codes and the "bullet time" effect with view-dependent effect by querying any latent code at any continuous time within the video. Rendering with interpolated latent codes resulted in a smooth and plausible representation of dynamics between the two neighboring input frames. Please refer to our supplementary video for the 3D video visualizations.

Quantitative Comparison to the Baselines. Tab. 1 shows the quantitative comparison of our methods to the baselines using an average of single frame metrics and Tab. 2 shows

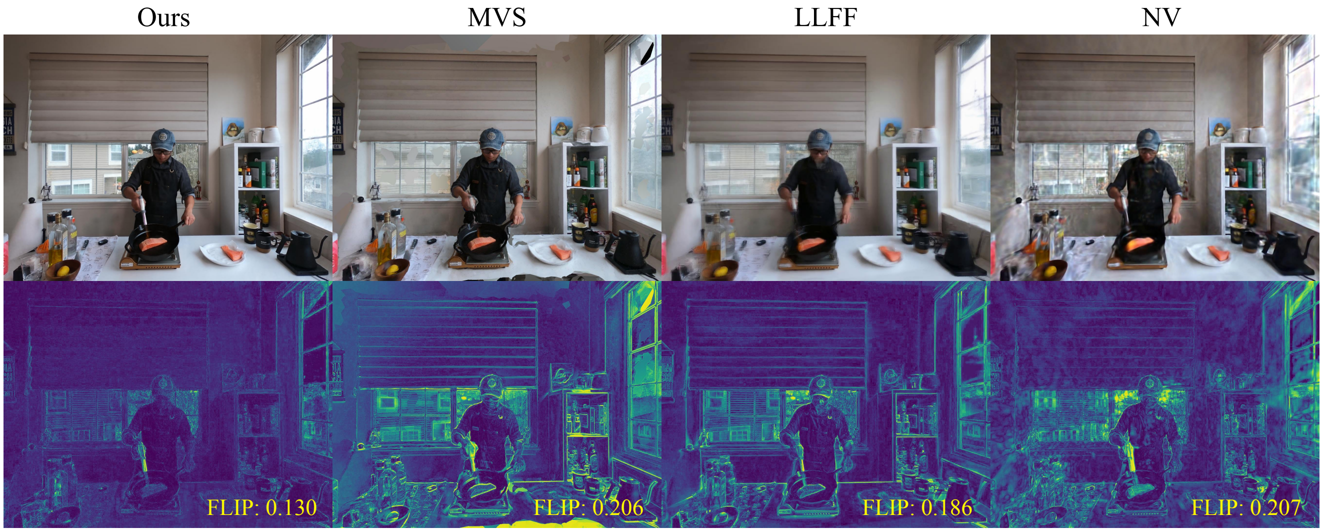

Figure 5. **Comparison of our final model to existing methods**, including Multi-view Stereo (MVS), local light field fusion (LLFF) [37] and NeuralVolume (NV) [32]. The first row shows novel view rendering on a test view. The second row visualizes the FLIP compared to the ground truth image. Compared to alternative methods, our method can achieve best visual quality.

Table 2. **Quantitative comparison** of our proposed method to baselines using perceptual video quality metric Just-Objectionable-Difference (JOD) [33]. Higher number (maximum 10) indicates less noticeable visual difference to the ground truth.

Method	NeuralVolumes	LLFF	NeRF-T	DyNeRF
JOD ↑	6.50	6.48	7.73	**8.07**

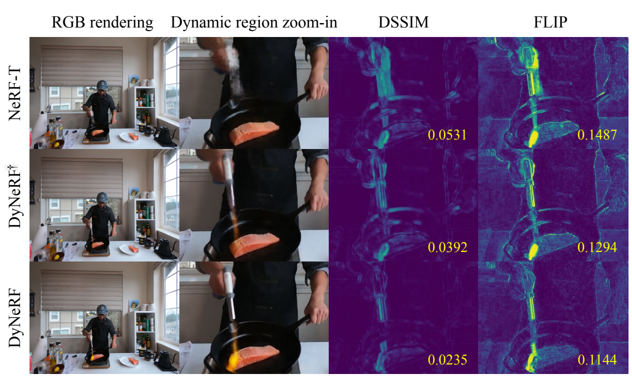

Figure 6. **Qualitative comparisons** of DyNeRF variants on one image of the sequence whose averages are reported in Tab. 1. From left to right we show the rendering by each method, then zoom onto the moving flame gun, then visualize DSSIM and FLIP for this region using the *viridis* colormap (dark blue is 0, yellow is 1, lower is better).

the comparison to baselines using a perceptual video metric. We train all the neural radiance field based baselines and our method the same number of iterations for fair comparison. Compared to the existing methods, MVS, NeuralVolumes and LLFF, our method is able capture and render significant more photo-realistic images, in all the quantitative measures. Compared to the time-variant NeRF baseline NeRF-T and our basic DyNeRF model without our proposed training strategy (DyNeRF†), our DyNeRF model variants trained with our proposed training strategy perform significantly better in all metrics.

Qualitative Comparison to the Baselines. We highlight visual comparisons of our methods to the baselines in Fig. 5 and Fig. 6. The visual results of the rendered images and FLIP error maps highlight the advantages of our approach in terms of photorealism that are not well quantified using the metrics. In Fig. 5 we compare to the existing methods. MVS with texturing suffers from incomplete reconstruction, especially for occlusion boundaries, such as image boundaries and the window regions. The baked-in textures also cannot capture specular and transparent effects properly, e.g., the window glasses. LLFF [37] produces blurred images with ghosting artifacts and less consistent novel view across time, especially for objects at occlusion boundaries and greater distances to the foreground, e.g., trees through the windows behind the actor. The results from Neural Volumes [32] contain cloudy artifacts and suffer from inconsistent colors and brightness (which can be better observed in the supplemental video). In contrast, our method achieves clear images, unobstructed by "cloud artifacts" and produces the best results compared to the existing methods. In particular, the details of the actor (e.g., hat, hands) and important details (e.g., flame torch, which consists of a highly reflective surface as well as the volumetric

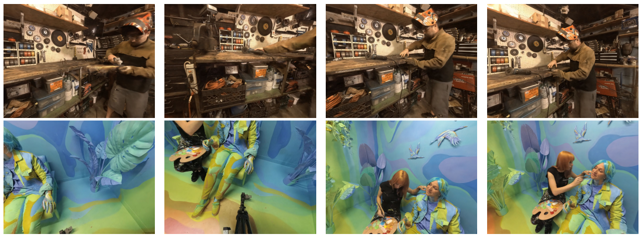

Figure 7. **Snapshots of novel view rendered videos** on immersive video datasets [6].

flame appearance) are faithfully captured by our method. Furthermore, MVS and LLFF and NeuralVolume cannot model scenes as compact and continuous spatio-temporal representation as our DyNeRF representation. In Fig. 6, we compare various settings of the dynamic neural radiance fields. NeRF-T can only capture a blurry motion representation, which loses all appearance details in the moving regions and cannot capture view-dependent effects. Though DyNeRF† has a similar quantitative performance as NeRF-T, it has significantly improved visual quality in the moving regions compared to NeRF-T, but still struggles to recover the sharp appearance details. DyNeRF with our proposed training strategy can recover sharp details in the moving regions, including the torch gun and the flames.

Comparisons on Training Time. Our proposed method is computationally more efficient compared to alternative solutions. Training a NeRF model frame-by-frame is the only baseline that can achieve the same photorealism as DyNeRF. However, we find that training a single frame NeRF model to achieve the same photorealism requires about 50 GPU hours, which in total requires 15K GPU hours for a 30 FPS video of 10 seconds length. Our method only requires 1.3K GPU hours for the same video, which reduces the required compute by one order of magnitude.

Results on Immersive Video Datasets [6]. We further demonstrates our DyNeRF model can create reasonably well 3D immersive video using non-forward-facing and spherically distorted multi-view videos with the same parameter setting and same training time. Fig. 7 shows a few novel views rendered from our trained models. We include the video results in the supplementary video. DyNeRF is able to generate an immersive coverage of the whole dynamic space with a compact model. Compared to the frame-by-frame multi-spherical images (MSI) representation used in [6], DyNeRF represents the video as one spatial temporal model which is more compact in size (28MB for a 5s 30 FPS video) and can better represent the view-dependent effects in the scene. Given the same amount of training time, we also observe there are some challenges, particularly the blurriness in the fast moving regions given the same compute budget and as above. We estimate one epoch of training time will take 4 weeks while we only trained all models

Figure 8. A few examples of failed outdoor reconstruction using DyNeRF.

using 1/4 of all pixels for a week. It requires longer training time to gain sharpness, which remains as a challenge to our current method in computation.

Limitations. There are a few challenging scenarios that our method is currently facing. (1) Highly dynamic scenes with large and fast motions are challenging to model and learn, which might lead to blur in the moving regions. As shown in Fig. 8, we observe it is particularly difficult to tackle fast motion in a complex environment, e.g. outdoors with forest structure behind. An adaptive sampling strategy during the hierarchical training that places more keyframes during the challenging parts of the sequence or more explicit motion modeling could help to further improve results. (2) While we already achieve a significant improvement in terms of training speed compared to the baseline approaches, training still takes a lot of time and compute resources. Finding ways to further decrease training time and to speed up rendering at test time are required. (3) Viewpoint extrapolation beyond the bounds of the training views is challenging and might lead to artifacts in the rendered imagery. We hope that, in the future, we can learn strong scene priors that will be able to fill in the missing information. (4) We discussed the importance sampling strategy and its effectiveness based on the assumption of videos observed from static cameras. We leave the study of this strategy on videos from moving cameras as future work. We believe these current limitations are good directions to explore in follow-up work and that our approach is a stepping stone in this direction.

5. Conclusion

We have proposed a novel neural 3D video synthesis approach that is able to represent real-world multi-view video recordings of dynamic scenes in a compact, yet expressive representation. As we have demonstrated, our approach is able to represent a 10 second long multi-view recording by 18 cameras in under 28MB. Our model-free representation enables both high-quality view synthesis as well as motion interpolation. At the core of our approach is an efficient algorithm to learn dynamic latent-conditioned neural radiance fields that significantly boosts training speed, leads to fast convergence, and enables high quality results. We see our approach as a first step forward in efficiently training dynamic neural radiance fields and hope that it will inspire follow-up work in the exciting and emerging field of neural scene representations.

References

[1] Edward H. Adelson and James R. Bergen. The plenoptic function and the elements of early vision. In *Computational Models of Visual Processing*, pages 3–20. MIT Press, 1991. 1

[2] Pontus Andersson, Jim Nilsson, Tomas Akenine-Möller, Magnus Oskarsson, Kalle Åström, and Mark D Fairchild. Flip: a difference evaluator for alternating images. *Proceedings of the ACM on Computer Graphics and Interactive Techniques (HPG 2020)*, 3(2), 2020. 6

[3] Aayush Bansal, Minh Vo, Yaser Sheikh, Deva Ramanan, and Srinivasa Narasimhan. 4d visualization of dynamic events from unconstrained multi-view videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5366–5375, 2020. 3

[4] Mojtaba Bemana, Karol Myszkowski, Hans-Peter Seidel, and Tobias Ritschel. X-fields: implicit neural view-, light- and time-image interpolation. *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020. 3

[5] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. *arXiv preprint arXiv:2007.09892*, 2020. 2

[6] Michael Broxton, John Flynn, Ryan Overbeck, Daniel Erickson, Peter Hedman, Matthew Duvall, Jason Dourgarian, Jay Busch, Matt Whalen, and Paul Debevec. Immersive light field video with a layered mesh representation. *ACM Transactions on Graphics (TOG)*, 39(4):86–1, 2020. 3, 5, 8

[7] Joel Carranza, Christian Theobalt, Marcus A Magnor, and Hans-Peter Seidel. Free-viewpoint video of human actors. *ACM Transactions on Graphics (TOG)*, 22(3):569–577, 2003. 3

[8] Shenchang Eric Chen and Lance Williams. View interpolation for image synthesis. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 279–288, 1993. 2

[9] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (ToG)*, 34(4):1–13, 2015. 3

[10] Abe Davis, Marc Levoy, and Fredo Durand. Unstructured light fields. In *Computer Graphics Forum*, volume 31, pages 305–314. Wiley Online Library, 2012. 2

[11] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 145–156, 2000. 2

[12] Yilun Du, Yinan Zhang, Hong-Xing Yu, Joshua B Tenenbaum, and Jiajun Wu. Neural radiance flow for 4d view synthesis and video processing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14324–14334, 2021. 3

[13] Carlos Hernández Esteban and Francis Schmitt. Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*, 96(3):367–392, 2004. 2

[14] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2367–2376, 2019. 2

[15] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–1376, 2009. 2

[16] Guy Gafni, Justus Thies, Michael Zollhofer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8649–8658, 2021. 3

[17] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE International Conference on Computer Vision*, 2021. 3

[18] Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. The lumigraph. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 43–54, 1996. 2

[19] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020. 2

[20] Kaiwen Guo, Peter Lincoln, Philip Davidson, Jay Busch, Xueming Yu, Matt Whalen, Geoff Harvey, Sergio Orts-Escolano, Rohit Pandey, Jason Dourgarian, et al. The relightables: Volumetric performance capture of humans with realistic relighting. *ACM Transactions on Graphics (TOG)*, 38(6):1–19, 2019. 3

[21] Zeng Huang, Tianye Li, Weikai Chen, Yajie Zhao, Jun Xing, Chloe LeGendre, Linjie Luo, Chongyang Ma, and Hao Li. Deep volumetric video from very sparse multi-view performance capture. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 336–354, 2018. 3

[22] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. Learning-based view synthesis for light field cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–10, 2016. 2

[23] Takeo Kanade, Peter Rander, and PJ Narayanan. Virtualized reality: Constructing virtual worlds from real scenes. *IEEE Multimedia*, 4(1):34–47, 1997. 3

[24] Abhishek Kar, Christian Häne, and Jitendra Malik. Learning a multi-view stereo machine. In *Advances in neural information processing systems*, pages 365–376, 2017. 2

[25] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 6

[26] Christoph Lassner and Michael Zollhöfer. Pulsar: Efficient sphere-based neural rendering. *arXiv:2004.07484*, 2020. 2

[27] Aldo Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on pattern analysis and machine intelligence*, 16(2):150–162, 1994. 2

[28] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, 1996. 2

[29] Hao Li, Linjie Luo, Daniel Vlasic, Pieter Peers, Jovan Popović, Mark Pauly, and Szymon Rusinkiewicz. Temporally coherent completion of dynamic shapes. *ACM Transactions on Graphics (TOG)*, 31(1):1–11, 2012. 3

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

[31] Lingjie Liu, Marc Habermann, Viktor Rudnev, Kripasindhu Sarkar, Jiatao Gu, and Christian Theobalt. Neural actor: Neural free-view synthesis of human actors with pose control. *ACM Trans. Graph.(ACM SIGGRAPH Asia)*, 2021. 3

[32] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *ACM Trans. Graph.*, 38(4):65:1–65:14, July 2019. 2, 3, 5, 7

[33] Rafał K. Mantiuk, Gyorgy Denes, Alexandre Chapiro, Anton Kaplanyan, Gizem Rufo, Romain Bachy, Trisha Lian, and Anjul Patney. Fovvideovdp: A visible difference predictor for wide field-of-view video. *tog*, 2021. 6, 7

[34] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *CVPR*, 2021. 3

[35] Abhimitra Meka, Christian Haene, Rohit Pandey, Michael Zollhöfer, Sean Fanello, Graham Fyffe, Adarsh Kowdle, Xueming Yu, Jay Busch, Jason Dourgarian, et al. Deep reflectance fields: high-quality facial reflectance field inference from color gradient illumination. *ACM Transactions on Graphics (TOG)*, 38(4):1–12, 2019. 2

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

[37] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 2, 5, 7

[38] 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 *ECCV*, 2020. 1, 2, 3, 4, 6

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

[40] Atsuhiro Noguchi, Xiao Sun, Stephen Lin, and Tatsuya Harada. Neural articulated radiance field. In *International Conference on Computer Vision*, 2021. 3

[41] Ryan S. Overbeck, Daniel Erickson, Daniel Evangelakos, and Paul Debevec. The making of welcome to light fields vr. In *ACM SIGGRAPH 2018 Talks*, SIGGRAPH '18, New York, NY, USA, 2018. Association for Computing Machinery. 2

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

[43] 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. 3

[44] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *CVPR*, 2021. 3

[45] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. *arXiv preprint arXiv:2011.13961*, 2020. 3

[46] Amit Raj, Michael Zollhöfer, Tomas Simon, Jason M. Saragih, Shunsuke Saito, James Hays, and Stephen Lombardi. Pixel-aligned volumetric avatars. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11728–11737, 2021. 3

[47] Umme Sara, Morium Akter, and Mohammad Shorif Uddin. Image quality assessment through fsim, ssim, mse and psnr - a comparative study. *Journal of Computer and Communications*, 7(3):8–18, 2019. 6

[48] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5

[49] 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. 2

[50] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer. Deepvoxels: Learning persistent 3d feature embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2437–2446, 2019. 2

[51] Pratul P Srinivasan, Richard Tucker, Jonathan T Barron, Ravi Ramamoorthi, Ren Ng, and Noah Snavely. Pushing the boundaries of view extrapolation with multiplane images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 175–184, 2019. 2

[52] Jonathan Starck and Adrian Hilton. Surface capture for performance-based animation. *IEEE computer graphics and applications*, 27(3):21–31, 2007. 3

[53] Matthew Tancik, Pratul P Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *arXiv preprint arXiv:2006.10739*, 2020. 4

[54] A. Tewari, O. Fried, J. Thies, V. Sitzmann, S. Lombardi, K. Sunkavalli, R. Martin-Brualla, T. Simon, J. Saragih, M. Nießner, R. Pandey, S. Fanello, G. Wetzstein, J.-Y. Zhu, C. Theobalt, M. Agrawala, E. Shechtman, D. B Goldman, and M. Zollhöfer. State of the Art on Neural Rendering. *Computer Graphics Forum (EG STAR 2020)*, 2020. 2

[55] Ayush Tewari, O Fried, J Thies, V Sitzmann, S Lombardi, Z Xu, T Simon, M Nießner, E Tretschk, L Liu, et al. Advances in neural rendering. In *ACM SIGGRAPH 2021 Courses*, pages 1–320. 2021. 2

[56] 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 *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2021. 3

[57] Paul Upchurch, Noah Snavely, and Kavita Bala. From a to z: supervised transfer of style and content using deep neural network generators. *arXiv preprint arXiv:1603.02003*, 2016. 6

[58] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7467–7477, 2020. 2

[59] Daniel N Wood, Daniel I Azuma, Ken Aldinger, Brian Curless, Tom Duchamp, David H Salesin, and Werner Stuetzle. Surface light fields for 3d photography. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 287–296, 2000. 2

[60] Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9421–9431, 2021. 3

[61] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018. 2

[62] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[63] Jae Shin Yoon, Kihwan Kim, Orazio Gallo, Hyun Soo Park, and Jan Kautz. 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*, pages 5336–5345, 2020. 3

[64] Wentao Yuan, Zhaoyang Lv, Tanner Schmidt, and Steven Lovegrove. Star: Self-supervised tracking and reconstruction of rigid objects in motion with neural rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13144–13152, 2021. 3

[65] 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

[66] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018. 2

[67] C Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High-quality video view interpolation using a layered representation. *ACM transactions on graphics (TOG)*, 23(3):600–608, 2004. 3