

DaRePlane: Direction-aware Representations for Dynamic Scene Reconstruction

Ange Lou^{1,2}, Benjamin Planche², Zhongpai Gao², Yamin Li¹, Tianyu Luan^{2,3}, Hao Ding^{2,4}, Meng Zheng,

Terrence Chen², Ziyan Wu², Jack Noble¹

¹Vanderbilt University, Nashville TN, USA

²United Imaging Intelligence, Boston MA, USA

³Johns Hopkins University, Baltimore MD, USA

⁴University of Buffalo, Buffalo NY, USA

first.last@vanderbilt.edu, first.last@uii-ai.com, tianyulu@buffalo.edu, hding15@jhu.edu

arXiv:2410.14169v1 [cs.CV] 18 Oct 2024

Abstract—Numerous recent approaches to modeling and re-rendering dynamic scenes leverage plane-based explicit representations, addressing slow training times associated with models like neural radiance fields (NeRF) and Gaussian splatting (GS). However, merely decomposing 4D dynamic scenes into multiple 2D plane-based representations is insufficient for high-fidelity re-rendering of scenes with complex motions. In response, we present DaRePlane, a novel direction-aware representation approach that captures scene dynamics from six different directions. This learned representation undergoes an inverse dual-tree complex wavelet transformation (DTCWT) to recover plane-based information. Within NeRF pipelines, DaRePlane computes features for each space-time point by fusing vectors from these recovered planes, then passed to a tiny MLP for color regression. When applied to Gaussian splatting, DaRePlane computes the features of Gaussian points, followed by a tiny multi-head MLP for spatial-time deformation prediction. Notably, to address redundancy introduced by the six real and six imaginary direction-aware wavelet coefficients, we introduce a trainable masking approach, mitigating storage issues without significant performance decline. To demonstrate the generality and efficiency of DaRePlane, we test it on both regular and surgical dynamic scenes, for both NeRF and GS systems. Extensive experiments show that DaRePlane yields state-of-the-art performance in novel view synthesis for various complex dynamic scenes.

Index Terms—DaRePlane, NeRF, Gaussian Splatting, Dynamic Scene, 3D Reconstruction.

I. INTRODUCTION

THE reconstruction and re-rendering of 3D scenes from a set of 2D images pose a fundamental challenge in computer vision, holding substantial implications for a range of AR/VR applications [2]–[4]. Despite recent progress in reconstructing static scenes, significant challenges remain. Real-world scenes are inherently dynamic, characterized by intricate motion, further adding to the task complexity.

Recent popular reconstruction approaches can be summarized into two main categories: neural radiance fields (NeRF) [5] and Gaussian splatting (GS) [6]. NeRF-related methods are known for achieving high-fidelity reconstruction performance,

This work was partly carried out during the internship of Ange Lou, Tianyu Luan, and Hao Ding at United Imaging Intelligence, Boston MA, USA.

Corresponding authors: Benjamin Planche and Jack Noble
(benjamin.planche@uii-ai.com, jack.noble@vanderbilt.edu).

capturing fine details and complex scene geometries. NeRF works by formulating a scene as a continuous volumetric field, where each point in space (static) or space-time (dynamic) has a corresponding color and density. This information is then rendered using a differentiable volumetric rendering process. However, these methods suffer from extensive optimization times and low inference speeds. In contrast, GS represents a scene using an explicit cloud of point-like Gaussians and employs a real-time differentiable renderer. This significantly reduces both optimization and novel-view synthesis times, making GS a more practical choice for real-time applications.

Recent dynamic scene reconstruction methods build on NeRF’s implicit representation. Some utilize a large MLP to process spatial and temporal point positions, generating color outputs [7]–[9]. Others aim to disentangle scene motion and appearance [10]–[14]. However, both approaches face computational challenges, requiring extensive MLP evaluations for novel view rendering. The slow training process, often spanning days or weeks, and the reliance on additional supervision like depth maps [8], [14], [15] limit their widespread adoption for dynamic scene modeling. Several recent studies [16]–[18] have proposed decomposition-based methods to address the training time challenge. Similar techniques are also introduced in GS to model the temporal deformations of Gaussians for dynamic scenes [19]–[21]. However, relying solely on decomposition limits both NeRF’s and GS’s ability to capture high-fidelity texture details.

Recent studies have explored the possibility of incorporating frequency information into NeRF [22]–[26] and GS [27]. These frequency-based representations demonstrate promising performance in static-scene rendering, particularly in recovering detailed information. However, there is limited exploration w.r.t. the ability of these methods to scale from static to dynamic scenes. Additionally, HexPlane [16] has noted a significant degradation in reconstruction performance when using wavelet coefficients as a basis. This limitation is inherent to wavelets themselves, and we delve into a detailed discussion in the following paragraph.

Traditional 2D discrete wavelet transform (DWT) employs low/high-pass real wavelets to decompose a 2D image or grid into approximation and detail wavelet coefficients across

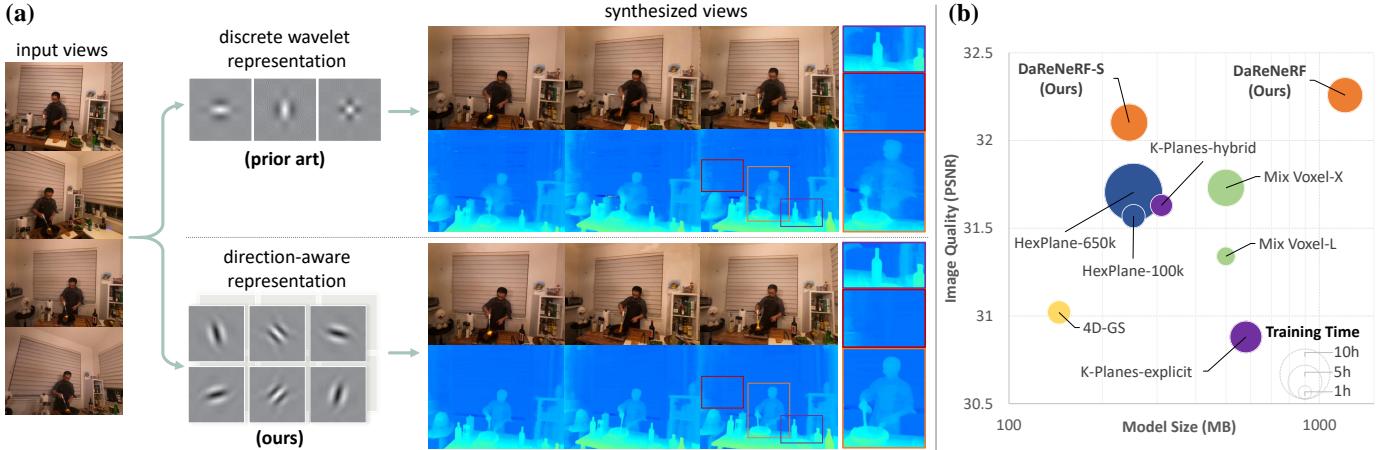


Fig. 1. Performance of dynamic NeRF and Gaussian splatting (GS) with DaRePlane on 4D scenes. Our direction-aware representation excels by capturing features of dynamic scenes from six different directions—a capability beyond the reach of traditional discrete-wavelet representations, *c.f.* sub-figure (a). Built upon this advanced representation, our NeRF method first introduced in [1] outperforms prior work in challenging 4D scenarios while being competitive in terms of training time and model size, offering the best trade-off overall, *c.f.* sub-figure (b). Similar results for our GS solution are shared in Figure 7.

different scales. These coefficients offer an efficient representation of both global and local image details. However, there are two significant drawbacks hindering the successful application of 2D DWT-based representations to dynamic scenes. The first is the **shift variance** problem [28], where even a small shift in the input signal significantly disrupts the wavelets’ oscillation pattern. In dynamic 3D scenes, shifts are more pronounced than in static scenarios due to factors such as multi-object motion, camera motion, reflections, and variations in illumination. Simple DWT wavelet representations struggle to handle such variability, yielding poor results in dynamic regions. Another critical issue is the **poor direction selectivity** [29] in DWT representations. A 2D DWT produces a checkered pattern that blends representations from $\pm 45^\circ$, lacking directional selectivity, which is less effective for capturing lines and edges in images. Consequently, DWT-based representations fail to adequately model dynamic scenes, leading to results with noticeable ghosting artifacts around moving objects as shown in Figure 1.

In a preliminary publication [1], we addressed these key limitations of the discrete wavelet transform (DWT) by introducing an efficient and robust frequency-based representation designed to overcome the challenges of shift variance and lack of direction selectivity in modeling dynamic scenes. Inspired by the dual-tree complex wavelet transform (DTCWT) [30], we proposed a direction-aware representation, aiming to learn features from six distinct orientations without introducing the checkerboard pattern observed in DWT. Leveraging the properties of complex wavelet transforms, our approach ensures shift invariance within the representation. This direction-aware representation proves successful in modeling complex dynamic scenes, achieving state-of-the-art performance.

In the present article, we propose to generalize our DaRePlane representation and apply it to both NeRF and GS systems, testing it on regular and surgical dynamic scenes, each presenting their own challenges. Additionally, to highlight the generalizability of our proposed method (aimed at 4D scenarios), we extend its application to modelling static 3D

scenes. In this context, our proposed DaRePlane demonstrates high-fidelity reconstruction performance and efficient storage capabilities. This versatility underscores the efficacy of our approach not only in dynamic scenes but also in static environments, affirming its potential as a general representation utility across various scenarios.

In summary, our contributions are as follows:

- We are the first to leverage DTCWT in NeRF optimization, introducing a direction-aware representation to address the shift-variance and direction-ambiguity shortcomings in DWT-based representations. DaReNeRF thereby outperforms prior decomposition-based methods in modeling complex dynamic scenes.
- We implement a trainable mask method for dynamic scene reconstruction, effectively resolving the storage limitations associated with the direction-aware representation. This adaptation ensures memory efficiency comparable with current state-of-the-art methods.
- We extend our direction-aware representation to static scene reconstruction, and experiments demonstrate that our proposed method outperforms other state-of-the-art approaches, achieving a superior trade-off between performance and model size.
- [Extension – contribution] We further prove that our proposed representation can transfer to Gaussian splatting solutions (DaReGS), to similarly improve their modeling capability.
- [Extension – contribution] To demonstrate the generalizability of our proposed method, we test DaReNeRF and DaReGS in various surgical scenarios, including microscopy, endoscopy, and laparoscopy. Experiments show that our method is effective for reconstruction tasks across different areas.

II. RELATED WORK

A. Learnable Scene Representations

Neural Radiance Field. Neural Radiance Fields (NeRF) represent three-dimensional scenes by approximating a

radiance field using a neural network. This radiance field describes the color and density values for each sample point along a ray from a specific view direction. Novel views can be synthesized through the process of volume rendering [5]. NeRF [5] and its variants [31]–[38] show impressive results on novel view synthesis and many other application including 3D reconstruction [39]–[42], semantic segmentation [43], [44], object detection [45]–[48], generative model [49]–[51], and 3D content creation [52]–[54]. However, implicit neural representations suffer from slow rendering due to the numerous costly MLP evaluations required for each pixel. Various spatial-decomposition methods [49], [55]–[57] have been proposed to address the challenge of training speed in static scenes.

Gaussian Splatting. As another answer to NeRF’s costly optimization time and inference, 3D-GS has recently revolutionized the field of neural rendering. It employs a set of anisotropic 3D Gaussians, each parameterized by its position, covariance, color, and opacity, in order to explicitly represent a scene. To generate views, these 3D Gaussians are projected onto the camera’s imaging plane and rendered using point-based volume rendering [6]. Due to its compactness and rasterization speed, 3D-GS is applied to various scenarios, including 3D generation [58]–[60], autonomous driving [61], [62], scene understanding [63], and medical imaging [64]–[66].

Extension to Dynamic Scenes. Both NeRF and Gaussian Splatting (GS) can be extended to dynamic versions for modeling time-varying scenes. In the NeRF framework, one straightforward approach is to extend a static NeRF by introducing an additional time dimension [13] or by incorporating a latent code [14], [15], [67], [68]. While these methods demonstrate strong capabilities in modeling complex real-world dynamic scenes, they face a severely under-constrained problem that necessitates additional supervision—such as depth, optical flow, or dense observations—to achieve satisfactory results. The substantial model size and weeks-long training times associated with these approaches further hinder their real-world applicability. An alternative solution involves employing separate MLPs to represent the deformation field and a canonical field [13], [38], [69]–[71]. Here, the canonical field captures the static scene, while the deformation field learns coordinate mappings to the canonical space over time. Although this method offers improvements over the previous approach, it still demands significant training time.

In the GS setting, a more common method for depicting dynamic scenes involves using explicit plane-based representations to model the spatiotemporal deformation of 3D Gaussians [19], [72]–[75].

B. Scene Decomposition

Plane-Based Representations. Plane-based representations applied to dynamic scenes have first been proposed for NeRF methods [16]–[18]. These approaches aim to alleviate the lengthy training times associated with dynamic scenes while maintaining the ability to model their complexity. They decompose a 4D scene into plane-based representations and

employ a compact MLP to aggregate features for volumetric rendering of resulting images. A similar plane-based representation has then been integrated into the 3D-GS system [76], to aggregate the spatial-temporal deformation features of 3D Gaussians. Subsequently, multiple tiny MLPs are employed to predict the time-variant deformation of both position and covariance. While plane-based representation significantly reduces training time and memory storage for dynamic NeRF, and enhances training time and inference speed for dynamic Gaussian Splatting, it still faces challenges in preserving detailed texture information during rendering.

Wavelet Optimization. To further enhance rendering quality, wavelet-based representations [22], [23], [77] have gained significant attention for their ability to improve NeRF’s capability in capturing fine texture details, due to their proficiency in recovering high-fidelity signals. However, there has been limited exploration of the potential of wavelet-based representations for dynamic scene modeling. Applying wavelet-based representations directly to plane-based methods can lead to a significant performance decay, as illustrated in Figure 1. Similar degradation is also reported by HexPlane [16], highlighting the inherent limitations of wavelets, namely, shift variance and direction ambiguity. To overcome these limitations and build a more effective general wavelet-based representation for both NeRF and GS, we propose a direction-aware representation, which preserves the ability to detect detailed textures without requiring additional supervision, achieving state-of-the-art performance in real-world and surgical dynamic scene reconstruction.

C. Application to Surgical Videos

One of the most promising and impactful application areas for NeRF and GS is the reconstruction of dynamic surgical scenes from videos captured by surgical robots. Accurate reconstruction of surgical scenes from video is critical for precise image-guided surgery. Current NeRF-based methods [2], [78]–[81] achieve superior reconstructions compared to traditional SLAM-based approaches. The use of plane-based representations in dynamic NeRF [80], [81] significantly reduces training time to just a few minutes, thereby improving the feasibility of clinical applications. Moreover, plane-based 4D Gaussian Splatting (4D-GS) [82] further minimizes training time and enables real-time inference. However, reconstructing surgical scenes poses the challenge of requiring high-fidelity anatomical reconstructions, which are crucial for accurate registration with pre-operative imaging or other intraoperative imaging modalities and for providing precise 3D feedback to the surgeon.

III. METHOD

We seek to develop a model for a dynamic scene using a collection of images captured from different viewpoints, each timestamped. The objective is to fit a model capable of rendering new images at varying poses and time stamps. Similar to D-NeRF [13], this model assigns color and opacity to points in both space and time. The rendering process involves differentiable volumetric rendering along rays. Training the

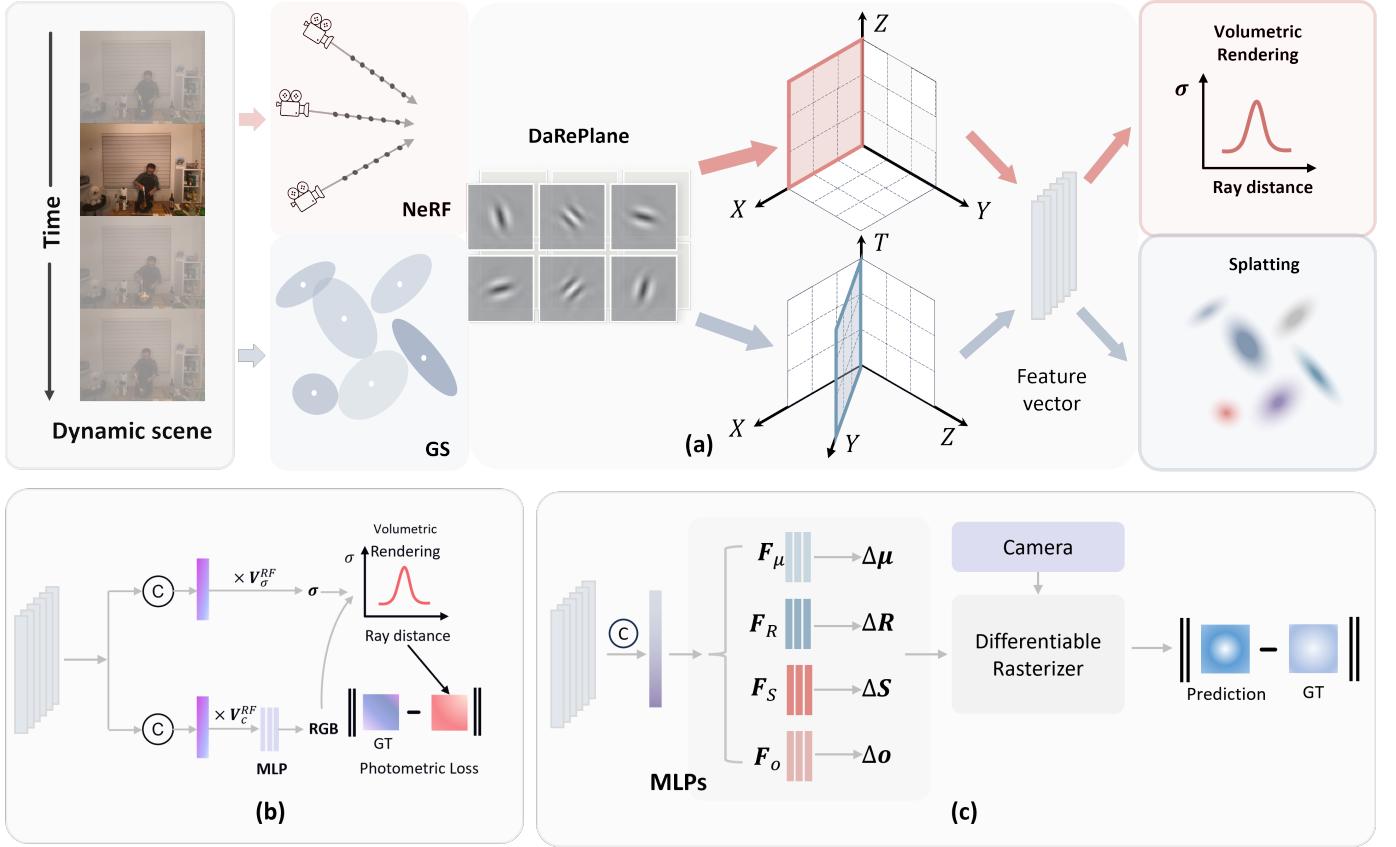


Fig. 2. Method Overview. (a) In the given sequence of images, NeRF and GS initialize the spatial-temporal points and a set of 3D Gaussians, respectively. Voxel features of these points (for NeRF) or Gaussians (for GS) are then computed by querying voxel planes in DaRePlane. These features are subsequently fed into the volumetric rendering process (for NeRF) or the splatting process (for GS) to synthesize the final images. **Bottom:** (b) **NeRF:** Feature vectors queried from DaRePlane are concatenated into a single vector, and then multiplies them by learned tensor V^{RF} for final results. RGB colors are regressed by a compact MLP, and images are synthesized via volumetric rendering. (c) **GS:** The concatenated feature vector is decoded using a multi-head deformation decoder to obtain the deformation of Gaussians at a specific timestamp t . These deformed Gaussians are then splatted to render the final images.

entire model relies on a photometric loss function, comparing rendered images with ground-truth images to optimize model parameters.

Our primary innovation lies in introducing a novel direction-aware representation for dynamic scenes. This distinctive representation is coupled with the inverse dual-tree complex wavelet transform (IDTCWT) and a compact implicit multi-layer perceptron (MLP) to enable the generation of high-fidelity novel views. Figure 2 shows an overview of the model. Note that for simplicity, we refer to the wavelet representation as wavelet coefficients in this section.

A. Plane-Based Representation

A natural dynamic scene can be represented as a 4D spatio-temporal volume denoted as D . This 4D volume comprises individual static 3D volume for each time step, namely $\{V_1, V_2, \dots, V_T\}$. Directly modeling a 4D volume would entail a memory complexity of $\mathcal{O}(N^3TF)$, where N, T, F are spatial resolution, temporal resolution and feature size (e.g., with $F = 3$ representing RGB colors). To improve the overall performance, we propose a direction-aware representation applied to baseline plane-based 4D volume decomposition [16]. In such baseline, a representation of the 4D volume can be

represented as follows:

$$D = \sum_{r=1}^{R_1} M_r^{XY} \circ M_r^{ZT} \circ v_r^1 + \sum_{r=1}^{R_2} M_r^{XZ} \circ M_r^{YT} \circ v_r^2 + \sum_{r=1}^{R_3} M_r^{YZ} \circ M_r^{XT} \circ v_r^3 \quad (1)$$

where each $M_r^{AB} \in \mathbb{R}^{AB}$ represents a learned 2D plane-based representation with $\{(A, B) \in \{X, Y, Z, T\}^2 \mid A \neq B\}$, and $v_r^i \in \mathbb{R}^F$ are learned vectors along F axes. The parameters R_1, R_2 and R_3 correspond to the number of low rank components. By defining $R = R_1 + R_2 + R_3 \ll N$, the model's memory complexity can be notably reduced from $\mathcal{O}(N^3TF)$ to $\mathcal{O}(RN^2TF)$. This reduction in memory requirements proves advantageous for efficiently modeling dynamic scenes while preserving computational resources.

Plane-based 4D NeRF models predict the density and appearance features of points in space-time by multiplying the feature vectors extracted from paired planes (e.g., XY and ZT), concatenating the results into a single vector, and then multiplying them by V^{RF} . The point opacities are directly queried from the density features, whereas the color values are regressed by a compact MLP conditioned on the appearance features and view directions. Finally, images are synthesized

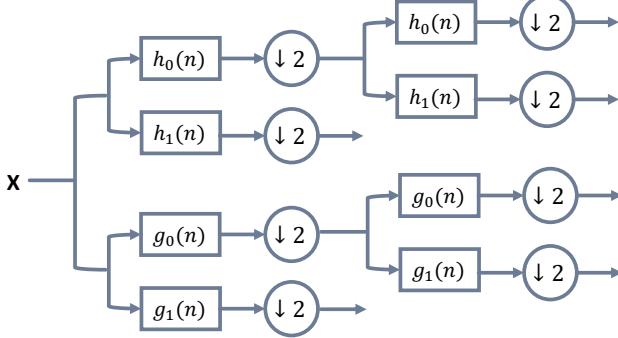


Fig. 3. **Analysis Filter Bank**, for the dual tree complex wavelet transform.

via volumetric rendering as shown in the **NeRF** setting of Figure 2.

In **4D-GS** settings, the multiple feature vectors from the paired planes are also concatenated into a single feature vector, which is then processed through a multi-head MLP to predict the deformation of the Gaussians. Finally, the images are synthesized from the deformed Gaussians using a differentiable rasterizer, as shown in the **GS** setting of Figure 2.

To improve the overall performance, we apply our proposed direction-aware representation to both NeRF and GS baselines.

B. Direction-Aware Representation

Built upon plane-based 4D volume decomposition and drawing inspiration from the dual-tree complex wavelet transform, we introduce a direction-aware representation. This innovative approach enables the modeling of representations from six different directions. In contrast to the prevalent use of 2D discrete wavelet transforms (DWT), the dual tree complex wavelet transform (DTCWT) [30] employs two complex wavelets as illustrated in Figure 3. Given $h = [h_0, h_1]$ and $g = [g_0, g_1]$ low/high pass filter pairs for upper (real) and lower (imaginary) filter banks, the low-pass and high-pass complex wavelet transforms in DTCWT are denoted as $\phi(x) = \phi_h(x) + j\phi_g(x)$ and $\psi(x) = \psi_h(x) + j\psi_g(x)$. Consequently, applying low- and high-pass complex wavelet transforms to rows and columns of a 2D grid yields wavelet coefficients $\phi(x)\psi(y)$, $\psi(x)\phi(y)$ and $\psi(x)\psi(y)$. Due to filter design, the upper (real) and lower (imaginary) filter satisfy the Hilbert transform, denoted as $\psi_g(x) \approx \mathcal{H}(\psi_h(x))$. Finally, three additional wavelet coefficients, $\phi(x)\overline{\psi(y)}$, $\psi(x)\overline{\phi(y)}$ and $\psi(x)\overline{\psi(y)}$, can be obtained, where ϕ and ψ represent the complex conjugate of ϕ and ψ . From these 2D wavelet coefficients, we derive six direction-aware real and imaginary wavelet coefficients, each with the same set of six directions. Compared to 2D DWT, the six wavelet coefficients align along specific directions, eliminating the checkerboard effect, with more results in the supplementary material.

Exploiting the properties of DTCWT, we aim for the plane-based representation $M_r^{AB} \in \mathbb{R}^{m,n}$ of the 4D volume to possess direction-aware capabilities as illustrated in the top section of Figure 3. Here, m and n denote the resolution of the 2D plane-based representation. To imbue each 2D plane-based representation with direction-aware capabilities, we introduce twelve learned wavelet coefficients—six for

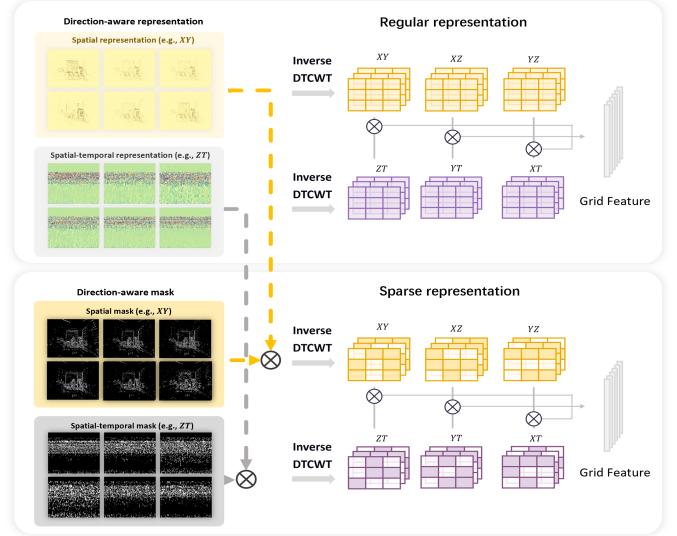


Fig. 4. **DaRePlane and DaRePlane-S Overview.** **Top:** The regular DaRePlane architecture comprises an approximation and 12 direction-aware coefficient maps for both spatial (e.g., XY) and spatial-temporal (e.g., ZT) plane-based representation. To compute the features of points in space-time, it multiplies feature vectors extracted from paired planes (e.g., XY and ZT). **Bottom:** The trainable mask is combined with the top architecture to create DaRePlane-S. Each direction-aware representation and the approximation representation are assigned their own sparse masks. The sparse representation undergoes an inverse dual tree complex wavelet transform to obtain plane-based spatial and spatial-temporal representations.

the real part and six for the imaginary part—denoted as $\mathbf{R}\{\mathcal{W}_i^{AB}\}_{i=1}^6 \in \mathbb{R}^{m/2^l, n/2^l}$ and $\mathbf{I}\{\mathcal{W}_i^{AB}\}_{i=1}^6 \in \mathbb{R}^{m/2^l, n/2^l}$, respectively. Additionally, a learned approximation coefficient is defined as $\mathcal{W}_a^{AB} \in \mathbb{R}^{m/2^{l-1}, n/2^{l-1}}$, with l the DTCWT transformation level. Consequently, a specific plane-based representation can be expressed as:

$$M_r^{AB} = IDTCWT([W_{a,r}^{AB}, \mathbf{R}\{\mathcal{W}_{i,r}^{AB}\}_{i=1}^6, \mathbf{I}\{\mathcal{W}_{i,r}^{AB}\}_{i=1}^6]) \quad (2)$$

Importantly, our representation is not only applicable for modeling dynamic 3D scenes but is also well-suited for static 3D scenes, following a TensorRF-like [56] decomposition:

$$D = \sum_{r=1}^{R_1} M_r^{XY} \circ v_r^Z \circ v_r^1 + \sum_{r=1}^{R_2} M_r^{XZ} \circ v_r^Y \circ v_r^2 \\ + \sum_{r=1}^{R_3} M_r^{YZ} \circ v_r^X \circ v_r^3 \quad (3)$$

In this formulation, a plane-based representation $M_r^{AB} \in \mathbb{R}^{AB}$ and a vector-based representation $v_r^C \in \mathbb{R}^C$ are employed to model a 3D volume. For static scenes, our direction-aware representations also could be applied to represent the plane-based representations.

C. Sparse Representation and Model Compression

In contrast to the classical 2D discrete wavelet transform (DWT), our direction-aware representation excels in modeling dynamic 3D scenes. However, it is worth noting that a single-level dual tree complex wavelet transform (DTCWT) necessitates six real direction-aware wavelet coefficients and six imaginary direction-aware wavelet coefficients to impart directional information to the plane-based representation. In

contrast, a single-level 2D DWT only has three real wavelet coefficients, albeit with inherent direction ambiguity. To enhance the storage efficiency of our solution, we employ learned masks [22] for each directional wavelet coefficient, selectively masking out less important features.

As illustrated in the bottom section of Figure 4, to address the 2^d redundancies, where $d = 2$ for the 2D DTCWT transform, we employ learned masks $\mathbf{R}\{\mathcal{M}_i^{AB}\}_{i=1}^6 \in \mathbb{R}^{m/2^l, n/2^l}$, $\mathbf{I}\{\mathcal{M}_i^{AB}\}_{i=1}^6 \in \mathbb{R}^{m/2^l, n/2^l}$ and $\mathcal{M}_a^{AB} \in \mathbb{R}^{m/2^{l-1}, n/2^{l-1}}$ for the six real wavelet coefficients, six imaginary wavelet coefficients and the approximation coefficients, respectively. The masked wavelet coefficients can be denoted as:

$$\widehat{\mathcal{W}}^{AB} = \text{sg}\left(\left(\mathbf{H}(\mathcal{M}^{AB}) - \text{sigmoid}(\mathcal{M}^{AB})\right) \odot \mathcal{W}^{AB}\right), \quad (4)$$

with $\{\mathbf{R}\{\mathcal{M}_i^{AB}\}_{i=1}^6, \mathbf{I}\{\mathcal{M}_i^{AB}\}_{i=1}^6, \mathcal{M}_a^{AB}\} \in \mathcal{M}^{AB}$ and $\{\mathbf{R}\{\mathcal{W}_i^{AB}\}_{i=1}^6, \mathbf{I}\{\mathcal{W}_i^{AB}\}_{i=1}^6, \mathcal{W}_a^{AB}\} \in \mathcal{W}^{AB}$. The functions sg , \mathbf{H} and sigmoid represent the stop-gradient operator, Heaviside step and element-wise sigmoid function, respectively. The masked plane-based representation is obtained from the masked wavelet coefficients through the equation:

$$\widehat{\mathcal{W}}_r^{AB} = IDTCWT([\widehat{\mathcal{W}}_{a,r}^{AB}, \mathbf{R}\{\widehat{\mathcal{W}}_{i,r}^{AB}\}_{i=1}^6, \mathbf{I}\{\widehat{\mathcal{W}}_{i,r}^{AB}\}_{i=1}^6]) \quad (5)$$

To encourage sparsity in the generated masks, we introduce an additional loss term \mathcal{L}_m , defined as the sum of all masks. We employ λ_m as the weight of \mathcal{L}_m to control the sparsity of the representation.

Following the removal of unnecessary representations through masking, we adopt a compression strategy akin to the one employed in masked wavelet NeRF [22], originally designed for static scenes, to compress the sparse representation and masks that identify non-zero elements. The process involves converting the binary mask values to 8-bit unsigned integers and subsequently applying run-length encoding (RLE). Finally, the Huffman encoding algorithm is employed on the RLE-encoded streams to efficiently map values with a high probability to shorter bits.

D. Optimization

We leverage our proposed direction-aware representation to effectively represent 3D dynamic scenes. The model is then optimized through a photometric loss function, which measures the difference between rendered images and target images. Additionally, we also add regularization items to reduce the artifacts and utilize the mask loss to control the sparsity of the DaRePlane. The overall loss is expressed as:

$$\mathcal{L} = \lambda_{photo}\mathcal{L}_{photo} + \lambda_{reg}\mathcal{L}_{reg} + \lambda_m\mathcal{L}_m, \quad (6)$$

with \mathcal{L}_{photo} , λ_{photo} , \mathcal{L}_{reg} , λ_{reg} and \mathcal{L}_m , λ_m the photometric loss, regularization loss and mask loss with respective weights. For both the NeRF and GS setting, we utilize the total variational (TV) loss as regularization item.

1) *NeRF*: In the NeRF framework, for a given point (x, y, z, t) , its opacity and appearance features are represented by DaRePlane. The final color is obtained through a small multi-layer perceptron (MLP), which takes the appearance feature and view direction as inputs. Using the point's opacities

and colors, images are generated through volumetric rendering. **Photometric Loss.** For the photometric loss \mathcal{L}_{photo} , we utilize the mean square error (MSE) as the loss function.

Training Strategy. We employ the same coarse-to-fine training strategy as in [16], [56], [83], where the resolution of grids progressively increases during training. This strategy not only accelerates the training process but also imparts an implicit regularization on nearby grids.

Emptiness Voxel. We maintain a small 3D voxel representation that indicates the emptiness of specific regions in the scene, allowing us to skip points located in empty regions. Given the typically large number of empty regions, this strategy significantly aids in acceleration. To generate this voxel, we evaluate the opacities of points across different time steps and aggregate them into a single voxel by retaining the maximum opacities. While preserving multiple voxels for distinct time intervals could potentially enhance efficiency, for the sake of simplicity, we opt to keep only one voxel.

2) *GS*: In the GS framework, we obtain the spatial-temporal representation, denoted as \mathbf{f} , from DaRePlane, we use four tiny MLPs, denoted as $\mathbf{F} = \{\mathbf{F}_\mu, \mathbf{F}_R, \mathbf{F}_S, \mathbf{F}_o\}$, to predict the time-variant deformation of position, rotation, scaling, and opacity of Gaussians, respectively. With the deformation of position $\Delta\mu = \mathbf{F}_\mu(\mathbf{f})$, rotation $\Delta\mathbf{R} = \mathbf{F}_R(\mathbf{f})$, scaling $\Delta\mathbf{S} = \mathbf{F}_S(\mathbf{f})$, opacity $\Delta\mathbf{o} = \mathbf{F}_o(\mathbf{f})$, the time-variant deformed Gaussians \mathbf{G}_t at time t can be expressed as:

$$\mathbf{G}_t = \mathbf{G}_0 + \Delta\mathbf{G} = (\mu + \Delta\mu, \mathbf{R} + \Delta\mathbf{R}, \mathbf{S} + \Delta\mathbf{S}, \mathbf{o} + \Delta\mathbf{o}) \quad (7)$$

Photometric Loss. The photometric loss for the Gaussian Splatting setting consist of two main components: 1) color losses, and 2) depth loss as shown below:

$$\mathcal{L}_{color} = \sum_{x \in \mathcal{I}} \left\| \mathbf{M}(x)(\hat{\mathbf{C}}(x) - \mathbf{C}(x)) \right\|_1 \quad (8)$$

$$\mathcal{L}_{depth} = 1 - Cov(\mathbf{M} \odot \hat{\mathbf{D}}, \mathbf{M} \odot \mathbf{D}) / \sqrt{Var(\mathbf{M} \odot \hat{\mathbf{D}})Var(\mathbf{M} \odot \mathbf{D})} \quad (9)$$

where \mathbf{M} , $\{\hat{\mathbf{C}}, \hat{\mathbf{D}}\}$, $\{\mathbf{C}, \mathbf{D}\}$, and \mathcal{I} are binary tool masks, predicted colors and depths, real colors and depths, and 2D coordinate space, respectively. Cov and Var operations in 9 represent the covariance and variances of the prediction and ground truth, respectively. This is equivalent to using the Pearson Correlation Coefficients (PCC) as loss.

IV. EXPERIMENTS

We first demonstrate the capabilities of our DaRePlane system on real-world dynamic and static scenes within the NeRF framework. We conduct a comprehensive comparison with existing methods and investigate the advantages of DaRePlane through extensive ablation studies. These studies showcase DaRePlane’s robustness and effectiveness in handling both dynamic and static scenes.

Next, we demonstrate that DaRePlane is suitable for entirely different scenarios compared to regular scenes and can transfer flexibly to different rendering systems, such as Gaussian Splatting. We evaluate DaReNeRF and DaReGS across different types of surgical scenes to highlight its versatility and effectiveness in these specialized applications.

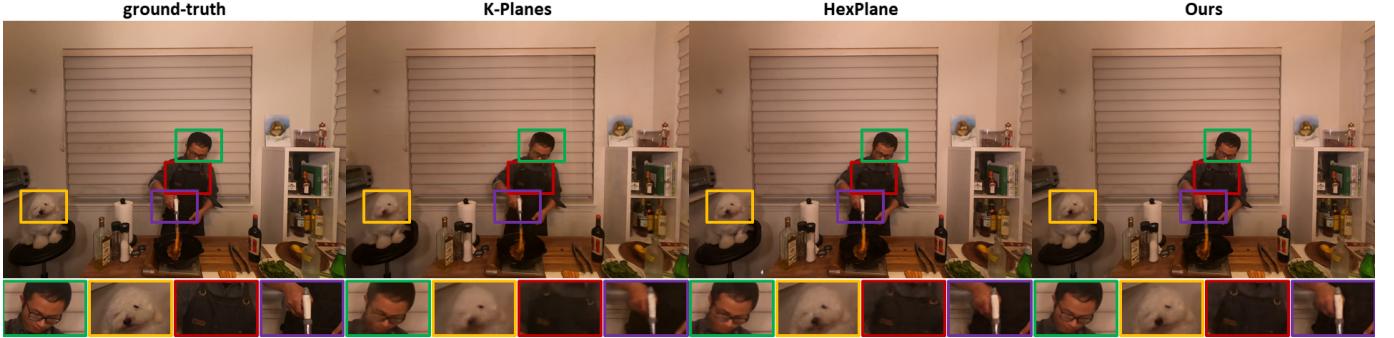


Fig. 5. **Visual Comparison on Dynamic Scenes (Plenoptic Data).** K-Planes and HexPlane are concurrent decomposition-based methods. As shown in the four zoomed-in patches, our method better reconstruct fine details and captures motion. Please refer to the supplementary material to see the figure animated.

A. Novel View Synthesis for Regular Dynamic Scenes

For dynamic scenes, we employ two distinct datasets with varying settings. Each dataset presents its own challenges, effectively addressed by our direction-aware representation.

Plenoptic Video Dataset [7] is a real-world dataset captured by a multi-view camera system using 21 GoPro cameras at a resolution of 2028×2704 and a frame rate of 30 frames per second. Each scene consists of 19 synchronized, 10-second videos, with 18 videos designated for training and one for evaluation. This dataset serves as an ideal testbed to assess the representation ability, featuring complex and challenging dynamic content, including highly specular, translucent, and transparent objects, topology changes, moving self-casting shadows, fire flames, strong view-dependent effects for moving objects, and more.

For a fair and direct comparison, we adhere to the same training and evaluation protocols as DyNeRF [7]. Our model is trained on a single A100 GPU, utilizing a batch size of 4,096. We adopt identical importance sampling strategies and hierarchical training techniques as DyNeRF, employing a spatial grid size of 512 and a temporal grid size of 300. The scene is placed under the normalized device coordinates (NDC) setting, consistent with the approach outlined in [5].

Quantitative compression results with state-of-the-art methods are presented in Table I. We utilize measurement metrics PSNR, structure dissimilarity index measure (DSSIM) [84], and perception quality measure LPIPS [85] to conduct a comprehensive evaluation. As demonstrated in Table I, leveraging the proposed direction-aware representation, both regular and sparse DaReNeRF achieve promising results compared to the most recent state-of-the-art, with analogous training time. This more ideal trade-off between performance and computational requirements, compared to prior art, is also illustrated in Figure 1.b, computed over Plenoptic data. Figure 5 presents some novel-view results on the Plenoptic dataset. Four small patches, each containing detailed texture information, are selected for comparison. DaReNeRF, equipped with the proposed direction-aware representation, excels in reconstructing moving objects (*e.g.*, dog and firing gun) and capturing better texture details (*e.g.*, hair and metal rings on the apron).

D-NeRF Dataset [13] is a monocular video dataset with 360° observations of synthetic objects. Dynamic 3D reconstruction from monocular video poses challenges as only

one observation is available for each timestamp. State-of-the-art methods for monocular video typically incorporate a deformation field. However, the underlying assumption is that the scenes undergo no topological changes, making them less effective for real-world cases (*e.g.*, Plenoptic dataset). Table II reports the rendering quality of different methods with and without deformation fields on the D-NeRF data, DaReNeRF outperforms all non-deformation methods, as well as some deformation methods, *e.g.* D-NeRF and TiNeuVox-S [90]. The superiority of our solution on topologically-changing scenes is further highlighted in appendix.

B. Novel View Synthesis of Regular Static Scenes

For static scenes, we test our proposed direction-aware representation on NeRF synthetic [5], Neural Sparse Voxel Fields (NSVF) [91] and LLFF [87] datasets. We use TensoRF-192 as baseline and apply our proposed representation. We report the performance on these three datasets in Tables III, IV, and V respectively.

Across these three static datasets, our direction-aware representation outperforms most compression-based NeRF models with model sizes ranging from 8 to 14MB. While our method's model size is larger than DWT-based solutions, it achieves comparable sparsity. For instance, with $\lambda_m = 2.5 \times 10^{-11}$, its *sparsity* reaches 94%, closely aligned with the 97% reported in the masked wavelet NeRF [22] paper. Notably, with similar sparsity, our direction-aware method exhibits PSNR improvements of 0.47, 1.57, and 0.60 over DWT-based methods on the three static datasets.

Figure 6 highlights the qualitative differences between DWT-based solutions and our proposed direction-aware method. In static scenes, our solution excels in reconstructing texture details compared to DWT representation, which is less sensitive to lines and edges patterns due to shift variance and direction ambiguity.

C. Novel View Synthesis of Dynamic Surgical Scenes

For dynamic surgical scene reconstruction, we employ four more distinct datasets with various types of surgical setting. Each dataset has different camera settings and its own challenges.

EndoNeRF Dataset [2]. The data was obtained from DaVinci robotic prostatectomy videos. Six clips, totaling 807 frames,

TABLE I

QUANTITATIVE COMPARISON ON PLENOPTIC VIDEO DATA. WE PRESENT RESULTS ON SYNTHESIS QUALITY AND TRAINING TIME (MEASURED IN GPU HOURS). FOLLOWING PRIOR ART, WE PROVIDE BOTH SCENE-SPECIFIC PERFORMANCE (FLAME-SALMON SCENE) AND MEAN PERFORMANCE ACROSS ALL CASES FROM THEIR ORIGINAL PAPERS.

	Model	Steps	PSNR↑	D-SSIM↓	LPIPS↓	Training Time↓	Model Size (MB) ↓
flame-salmon scene	Neural Volumes [86]	-	22.800	0.062	0.295	-	-
	LLFF [87]	-	23.239	0.076	0.235	-	-
	NeRF-T [7]	-	28.449	0.023	0.100	-	-
	DyNeRF [7]	650k	29.581	0.020	0.099	1,344h	28
	HexPlane [16]	650k	29.470	0.018	0.078	12h	252
	HexPlane [16]	100k	29.263	0.020	0.097	2h	252
	DaReNeRF-S	100k	<u>30.224</u>	<u>0.015</u>	0.089	5h	<u>244</u>
	DaReNeRF	100k	30.441	0.012	0.084	4.5h	1,210
all scenes (average)	NeRFPlayer [88]	-	30.690	0.034	0.111	6h	-
	HyperReel [57]	-	31.100	0.036	0.096	9h	-
	HexPlane [16]	650k	31.705	<u>0.014</u>	<u>0.075</u>	12h	252
	HexPlane [16]	100k	31.569	0.016	0.089	<u>2h</u>	252
	K-Planes-explicit [17]	120k	30.880	-	-	3.7h	580
	K-Planes-hybrid	90k	31.630	-	-	1.8h	310
	Mix Voxels-L [89]	25k	31.340	0.019	0.096	1.3h	500
	Mix Voxels-X [89]	50k	31.730	0.015	0.064	5h	500
	4D-GS [76]	-	31.020	-	0.150	<u>2h</u>	145
	DaReNeRF-S	100k	<u>32.102</u>	<u>0.014</u>	0.087	5h	<u>244</u>
	DaReNeRF	100k	32.258	0.012	0.084	4.5h	1,210

TABLE II

QUANTITATIVE STUDY ON D-NERF DATA. WITHOUT THE TOPOLOGICAL CONSTRAINTS OF USING DEFORMATION FIELDS, DARENERF OUTPERFORMS EVEN SOME DEFORMATION-BASED METHODS.

Model	Deform.	PSNR↑	SSIM↑	LPIPS↓
D-NeRF [13]	✓	30.50	0.95	0.07
TiNeuVox-S [90]	✓	30.75	0.96	0.07
TiNeuVox-B [90]	✓	<u>32.67</u>	<u>0.97</u>	<u>0.04</u>
4D-GS [76]	✓	33.30	0.98	0.03
T-NeRF [13]	-	29.51	<u>0.95</u>	0.08
HexPlane [16]	-	31.04	0.97	<u>0.04</u>
K-Planes [17]	-	31.05	0.97	-
DaReNeRF-S	-	<u>31.82</u>	0.97	0.03
DaReNeRF	-	31.95	0.97	0.03

TABLE III

QUANTITATIVE COMPARISON ON NERF SYNTH., WITH MODELS DESIGNED FOR DIFFERENT BIT-PRECISIONS (* DENOTES A MODEL QUANTIZED POST-TRAINING; NUMBERS IN BRACKETS DENOTE GRID RESOLUTIONS).

Precision	Method	Size (MB)	PSNR ↑
32-bit	KiloNeRF [92]	≤ 100	31.00
32-bit	CCNeRF (CP) [93]	4.4	30.55
8-bit*	NeRF [5]	1.25	31.52
8-bit	cNeRF [94]	0.70	30.49
8-bit	PREF [95]	9.88	31.56
8-bit*	VM-192 [56]	17.93	32.91
8-bit*	VM-192 (300) + DWT [22]	<u>0.83</u>	31.95
8-bit*	VM-192 (300) + Ours	8.91	<u>32.42</u>

were extracted from these videos, with each clip lasting 4 to 8 seconds at 15 *fps*. The footage is captured from stereo cameras at a single viewpoint and encompasses challenging scenes with non-rigid deformation and tool occlusion. Due to the privacy of the surgical data, only two clips from this dataset are publicly available: cutting tissues and pulling.

TABLE IV
QUANTITATIVE COMPARISON ON NSVF (STATIC SCENES).

Bit Precision	Model	Size (MB)	PSNR ↑
32-bit	KiloNeRF [92]	≤ 100	33.37
8-bit*	VM-192 [69]	17.77	<u>36.11</u>
8-bit*	VM-48 [56]	4.53	34.95
8-bit*	CP-384 [56]	0.72	33.92
8-bit*	VM-192 (300) + DWT [22]	<u>0.87</u>	34.67
8-bit*	VM-192 (300) + Ours	8.98	36.24

TABLE V
QUANTITATIVE COMPARISON ON LLFF (STATIC SCENES).

Bit Precision	Model	Size(MB)	PSNR ↑
8-bit	cNeRF [94]	<u>0.96</u>	26.15
8-bit*	PREF [69]	9.34	24.50
8-bit*	VM-96 [56]	44.72	26.66
8-bit*	VM-48 [56]	22.40	26.46
8-bit*	CP-384 [56]	0.64	25.51
8-bit*	VM-96 (640) + DWT [22]	1.34	25.88
8-bit*	VM-96 (640) + Ours	13.67	<u>26.48</u>

Hamlyn Dataset [96], [97]. The Hamlyn dataset includes both phantom heart and in-vivo sequences captured during da Vinci surgical robot procedures. The rectified images, stereo depth, and camera calibration information are sourced from [98]. To generate instrument masks, we use the Vision Foundation Model, Segment Anything Model [99], which enables the segmentation of surgical instruments. The Hamlyn dataset presents a rigorous evaluation scenario as it contains sequences depicting intracorporeal scenes with various challenges, such as weak textures, deformations, reflections, surgical tool occlusion, and illumination variations. Similar to previous work [81], we select seven specific sequences from the Hamlyn dataset (sequences: rectified01, rectified06, rectified08, and rectified09), each comprising 301 frames with a resolution of 480 × 640. These sequences

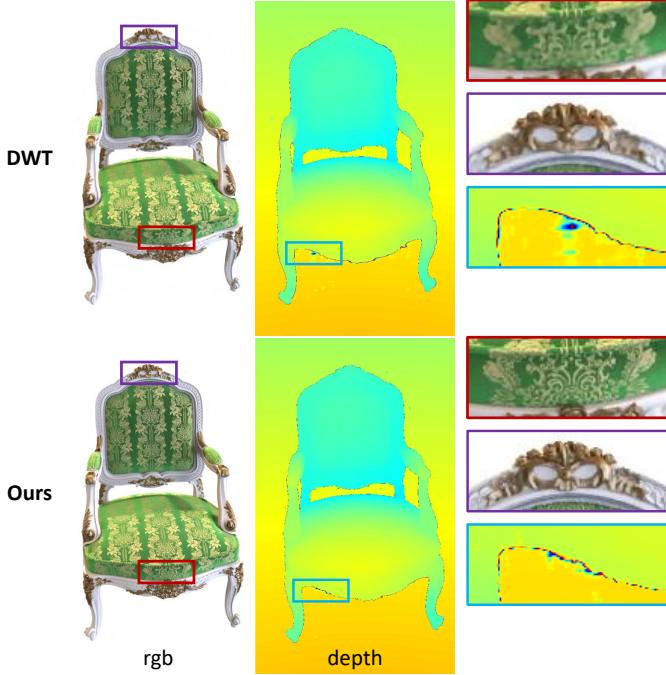


Fig. 6. **Visual Comparison of Static Scenes on NSVF Data.** Two representative patches are selected for closer inspection. Our method, free from the DWT limitations of shift variance and direction ambiguity, achieves superior texture reconstruction performance.

TABLE VI
QUANTITATIVE COMPARISON ON ENDONeRF DATASET (SURGICAL SCENE).

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	Training time \downarrow
NeRF	EndoNeRF [2]	0.933	0.089	12.0 hours
	EndoSurf [78]	0.954	0.074	8.5 hours
	LerPlane [80]	0.926	0.080	3.5 mins
	DaReNeRF	36.685	<u>0.947</u>	<u>4.0 mins</u>
GS	EndoGaussian [82]	<u>37.553</u>	<u>0.959</u>	2.0 mins
	DaReGS	38.348	0.966	<u>3.5 mins</u>

span approximately 10 seconds and feature scenarios involving surgical tool occlusion and extensive tissue exposure. The frames in each sequence are divided into two sets: 151 frames for training and the remainder for evaluation.

SCARED Dataset [101]. The SCARED dataset consists of fresh porcine cadaver abdominal anatomy captured using a da Vinci Xi endoscope and a projector to obtain high-quality depth maps of the scene. We selected five scenes (Sequences: dataset1/keyframe1, dataset2/keyframe1, dataset3/keyframe1, dataset6/keyframe1, and dataset7/keyframe1) from the dataset and split the frame data of each scene into 7:1 training and testing sets, following previous work [82].

Cochlear Implant Surgery [102], [103]. Unlike the aforementioned data, which were all obtained from the da Vinci Xi endoscope camera, we also selected two of our in-house cochlear implant surgery videos to test our proposed DaRePlane. These surgery videos were captured during real cochlear implant procedures at Vanderbilt University Medical Center (VUMC) and The Medical University of South Carolina (MUSC) using a surgical microscope with I.R.B. approval.

TABLE VII
QUANTITATIVE COMPARISON ON HAMLYN (SURGICAL SCENE).

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	Training time \downarrow
E-DSSR [100]	18.150	0.640	0.393	13 mins
EndoNeRF [2]	34.879	0.951	0.071	12.0 hours
TiNeu Vox-S [90]	35.277	0.953	0.085	12 mins
TiNeu Vox-B [90]	33.764	0.942	0.146	90 mins
LerPlane [80]	<u>35.504</u>	0.935	<u>0.083</u>	10 mins
ForPlane [81]	35.301	0.945	0.093	3.5 mins
DaReNeRF	35.641	<u>0.952</u>	0.085	<u>4.0 mins</u>

TABLE VIII
QUANTITATIVE COMPARISON ON SCARED (SURGICAL SCENE).

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	Training time \downarrow
EndoNeRF [2]	24.345	0.768	0.313	3.5 hours
EndoSurf [78]	25.020	0.802	0.356	5.8 hours
EndoGaussian [82]	<u>27.042</u>	<u>0.827</u>	<u>0.267</u>	2.2 mins
DaReGS	27.500	0.836	0.238	<u>3.5 mins</u>

Therefore, in Tables VI, VII, and IX, we present results across various surgical scenarios, including laparoscopy, endoscopy, and microscopy. Our proposed DaRePlane method demonstrates superior performance under both NeRF and Gaussian splatting settings, outperforming previous methods in all surgical scenarios. Specifically, in Table VI, our DaReNeRF surpasses all previous NeRF-based methods with only 4 minutes of training time. Additionally, our DaReGS not only outperforms the latest surgical Gaussian Splatting methods but also maintains a similar training time.

In surgical scenarios, explicit representation dynamic scene reconstruction methods such as LerPlane [80] and EndoGaussian [82] use K-Planes [17] to predict explicit representations and Gaussian deformations, respectively. Similar to regular scenes (based on HexPlane [16] and TensoRF [56]), we applied a trainable mask to each plane from the K-Planes method and tested our sparse DaRePlane on the SCARED dataset. From the results in Figure 7, EndoNeRF and EndoSurf, which are implicit representation-based methods, require significant training time for optimization. However, using explicit representation can achieve up to 100 \times acceleration. Additionally, our proposed frequency-based representation combined with trainable masks can save approximately 74% of memory storage compared to the previous state-of-the-art method, EndoGaussian.

In Figure 8, we provide visualization results from three distinct types of surgical datasets, demonstrating the efficacy of our proposed frequency-based representation. This method excels in recovering fine details in dynamic surgical scenes, such as tiny blood vessels and tissue textures. Moreover, our approach adeptly handles complex surgical scenarios, including significant tissue deformation and reflections in endoscopy, as well as transparent tools in microscopy surgery. These capabilities highlight the robustness and versatility of our method in accurately depicting various challenging surgical environments.

TABLE IX
QUANTITATIVE COMPARISON ON COCHLEAR IMPLANTS DATASET
(SURGICAL SCENE).

Method	PSNR ↑	SSIM ↑	LPIPS ↓	Training time ↓
EndoGaussian [82]	34.024	0.948	0.065	2.2 mins
DaReGS	34.386	0.952	0.052	3.5 mins

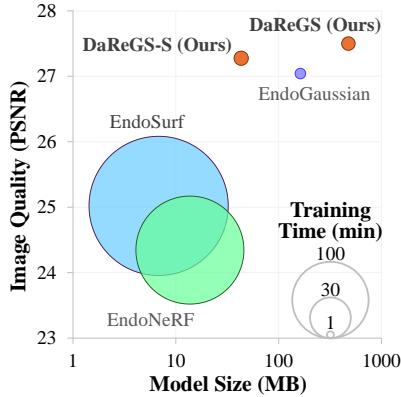


Fig. 7. Performance of Gaussian Splatting with DaRePlane on 4D scenes.

D. Ablations

Wavelet Function. We analyze the impact of different wavelet functions on reconstruction quality, aiming to facilitate a comparison between our direction-aware representation and DWT wavelet. The evaluation is conducted on NSVF data [91], where several complex wavelet functions with the approximate half-sample delay property—Antonini, LeGall, and two Near Symmetric filter banks (Near Symmetric A and Near Symmetric B)—are selected for comparison. Table X reveals that the choice of different wavelets has minimal effect on reconstruction quality. Even the worst-performing wavelet function outperforms the discrete wavelet transform, underscoring the advantages of our direction-aware representation.

Sparsity Analysis. We evaluate the sparsity of our direction-aware representation by varying the sparsity level using different λ_m values on the NSVF dataset. As depicted in Table XI, our direction-aware representation consistently achieves over 99% sparsity. This remarkable sparsity, coupled with a model size of approximately 1MB, demonstrates the efficiency of our method in modeling static scenes while outperforming state-of-the-art sparse representation methods.

Wavelet Levels. We investigated the impact of scene reconstruction performance across different wavelet levels, and the results are presented in supplementary material. We observed that increasing the wavelet level did not lead to significant performance improvements. Conversely, we noted a substantial increase in both training time and model size with the increment of wavelet level. As a result, throughout all experiments, we consistently set the wavelet level to 1.

V. CONCLUSION

We introduced a novel direction-aware representation capable of effectively capturing information from six different

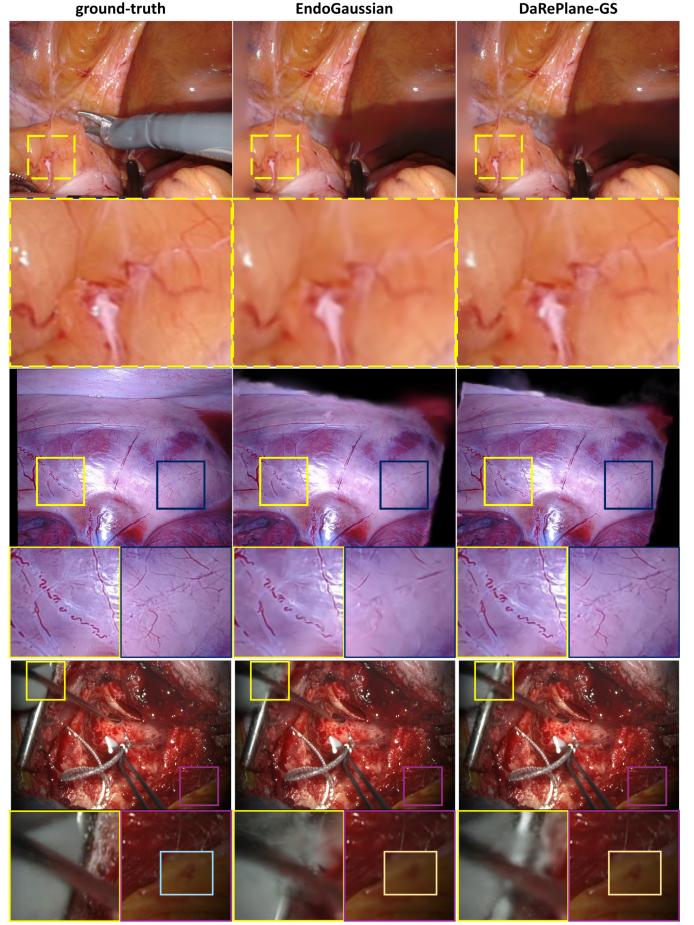


Fig. 8. Visual Comparison on Surgical Scenes. From top to bottom, the rows show results from the EndoNeRF, SCARED, and Cochlear Implant datasets. As demonstrated in the zoomed-in patches, our DaRePlane method recovers extremely fine details in the dynamic surgical scenes.

TABLE X
IMPACT OF WAVELET TRANSFORM TYPE/FUNCTION, ON
RECONSTRUCTION PERFORMANCE, EVALUATED ON NSVF DATA..

Wavelet Type	Wavelet Function	PSNR ↑
DWT	Haar	34.61
	Coiflets 1	34.56
	biorthogonal 4.4	34.67
	Daubechies 4	34.44
DTCWT	Antonini	36.10
	LeGall	36.14
	Near Symmetric A	36.24
	Near Symmetric B	36.17

directions. The shift-invariant and direction-selective nature of our proposed representation enables the high-fidelity reconstruction of challenging dynamic scenes without requiring prior knowledge about the scene dynamics. Although this approach introduces some storage redundancy, we mitigate this by incorporating trainable masks for both static and dynamic scenes, resulting in a model size comparable to recent methods. Our proposed method is applicable to both NeRF and Gaussian Splatting settings for various types of dynamic scene reconstruction and demonstrates superior performance in recovering extremely fine details.

TABLE XI
SPARSITY ANALYSIS OF DIRECTION-AWARE REPRESENTATION,
EVALUATED ON NVSF DATA.

λ_m	Sparsity \uparrow	Model Size (MB) \downarrow	PSNR \uparrow
1.0×10^{-10}	99.2%	1.16 MB	35.36
5.0×10^{-11}	97.3%	3.18 MB	35.81
2.5×10^{-11}	94.2%	8.98 MB	36.24
0	-	135 MB	36.34

ACKNOWLEDGMENTS

This work was supported in part by NIH grant R01DC014037 and R01DC008408 from the National Institute of Deafness and Other Communications Disorders. The content is solely the responsibility of the authors and does not necessarily reflect the views of this institute. The authors would like to acknowledge Dr. Robert Labadie at MUSC for providing de-identified surgery videos used in this study.

REFERENCES

- [1] A. Lou, B. Planche, Z. Gao, Y. Li, T. Luan, H. Ding, T. Chen, J. Noble, and Z. Wu, “Darenerf: Direction-aware representation for dynamic scenes,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5031–5042, 2024.
- [2] Y. Wang, Y. Long, S. H. Fan, and Q. Dou, “Neural rendering for stereo 3d reconstruction of deformable tissues in robotic surgery,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 431–441, Springer, 2022.
- [3] A. Moreau, N. Piasco, D. Tsishkou, B. Stanciulescu, and A. de La Fortelle, “Lens: Localization enhanced by nerf synthesis,” in *Conference on Robot Learning*, pp. 1347–1356, PMLR, 2022.
- [4] M. Wysocki, M. F. Azampour, C. Eilers, B. Busam, M. Salehi, and N. Navab, “Ultra-nerf: Neural radiance fields for ultrasound imaging,” *arXiv preprint arXiv:2301.10520*, 2023.
- [5] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “Nerf: Representing scenes as neural radiance fields for view synthesis,” *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.
- [6] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, “3d gaussian splatting for real-time radiance field rendering,” *ACM Transactions on Graphics*, vol. 42, no. 4, pp. 1–14, 2023.
- [7] T. Li, M. Slavcheva, M. Zollhoefer, S. Green, C. Lassner, C. Kim, T. Schmidt, S. Lovegrove, M. Goesele, R. Newcombe, et al., “Neural 3d video synthesis from multi-view video,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5521–5531, 2022.
- [8] Z. Li, S. Niklaus, N. Snavely, and O. 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*, pp. 6498–6508, 2021.
- [9] Q. Wang, Z. Wang, K. Genova, P. P. Srinivasan, H. Zhou, J. T. Barron, R. Martin-Brualla, N. Snavely, and T. Funkhouser, “Ibrnet: Learning multi-view image-based rendering,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4690–4699, 2021.
- [10] C. Gao, A. Saraf, J. Kopf, and J.-B. Huang, “Dynamic view synthesis from dynamic monocular video,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5712–5721, 2021.
- [11] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, “Nerfies: Deformable neural radiance fields,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5865–5874, 2021.
- [12] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz, “Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields,” *arXiv preprint arXiv:2106.13228*, 2021.
- [13] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, “D-nerf: Neural radiance fields for dynamic scenes,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10318–10327, 2021.
- [14] Y.-L. Liu, C. Gao, A. Meuleman, H.-Y. Tseng, A. Saraf, C. Kim, Y.-Y. Chuang, J. Kopf, and J.-B. Huang, “Robust dynamic radiance fields,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13–23, 2023.
- [15] Z. Li, Q. Wang, F. Cole, R. Tucker, and N. Snavely, “Dynibar: Neural dynamic image-based rendering,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4273–4284, 2023.
- [16] A. Cao and J. Johnson, “Hexplane: A fast representation for dynamic scenes,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 130–141, 2023.
- [17] S. Fridovich-Keil, G. Meanti, F. R. Warburg, B. Recht, and A. Kanazawa, “K-planes: Explicit radiance fields in space, time, and appearance,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12479–12488, 2023.
- [18] R. Shao, Z. Zheng, H. Tu, B. Liu, H. Zhang, and Y. Liu, “Tensor4d: Efficient neural 4d decomposition for high-fidelity dynamic reconstruction and rendering,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16632–16642, 2023.
- [19] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, “4d gaussian splatting for real-time dynamic scene rendering,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20310–20320, 2024.
- [20] Z. Xu, S. Peng, H. Lin, G. He, J. Sun, Y. Shen, H. Bao, and X. Zhou, “4k4d: Real-time 4d view synthesis at 4k resolution,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20029–20040, June 2024.
- [21] J. Ren, L. Pan, J. Tang, C. Zhang, A. Cao, G. Zeng, and Z. Liu, “Dreamgaussian4d: Generative 4d gaussian splatting,” *arXiv preprint arXiv:2312.17142*, 2023.
- [22] D. Rho, B. Lee, S. Nam, J. C. Lee, J. H. Ko, and E. Park, “Masked wavelet representation for compact neural radiance fields,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20680–20690, 2023.
- [23] M. Xu, F. Zhan, J. Zhang, Y. Yu, X. Zhang, C. Theobalt, L. Shao, and S. Lu, “Wavenerf: Wavelet-based generalizable neural radiance fields,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18195–18204, 2023.
- [24] L. Wang, J. Zhang, X. Liu, F. Zhao, Y. Zhang, Y. Zhang, M. Wu, J. Yu, and L. Xu, “Fourier plenocubes for dynamic radiance field rendering in real-time,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13524–13534, 2022.
- [25] Z. Wu, Y. Jin, and K. M. Yi, “Neural fourier filter bank,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14153–14163, 2023.
- [26] J. Yang, M. Pavone, and Y. Wang, “Freenerf: Improving few-shot neural rendering with free frequency regularization,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8254–8263, 2023.
- [27] J. Zhang, F. Zhan, M. Xu, S. Lu, and E. Xing, “Fregs: 3d gaussian splatting with progressive frequency regularization,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21424–21433, June 2024.
- [28] A. P. Bradley, “Shift-invariance in the discrete wavelet transform,” *Proceedings of VIIth Digital Image Computing: Techniques and Applications*. Sydney, 2003.
- [29] N. Kingsbury, “Image processing with complex wavelets,” *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 357, no. 1760, pp. 2543–2560, 1999.
- [30] I. W. Selesnick, R. G. Baraniuk, and N. C. Kingsbury, “The dual-tree complex wavelet transform,” *IEEE signal processing magazine*, vol. 22, no. 6, pp. 123–151, 2005.
- [31] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, “Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5855–5864, 2021.
- [32] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, “Mip-nerf 360: Unbounded anti-aliased neural radiance fields,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5470–5479, 2022.
- [33] B. Mildenhall, P. Hedman, R. Martin-Brualla, P. P. Srinivasan, and J. T. Barron, “Nerf in the dark: High dynamic range view synthesis from noisy raw images,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16190–16199, 2022.
- [34] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. Sajjadi, A. Geiger, and N. Radwan, “Regnerf: Regularizing neural radiance fields for

- view synthesis from sparse inputs,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5480–5490, 2022.
- [35] W. F. Low and G. H. Lee, “Robust e-nerf: Nerf from sparse & noisy events under non-uniform motion,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18335–18346, 2023.
- [36] P. Wang, Y. Liu, Z. Chen, L. Liu, Z. Liu, T. Komura, C. Theobalt, and W. Wang, “F2-nerf: Fast neural radiance field training with free camera trajectories,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4150–4159, 2023.
- [37] W. Bian, Z. Wang, K. Li, J.-W. Bian, and V. A. Prisacariu, “Nope-nerf: Optimising neural radiance field with no pose prior,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4160–4169, 2023.
- [38] Z. Yan, C. Li, and G. H. Lee, “Nerf-ds: Neural radiance fields for dynamic specular objects,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8285–8295, 2023.
- [39] R. Martin-Brualla, N. Radwan, M. S. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, “Nerf in the wild: Neural radiance fields for unconstrained photo collections,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7210–7219, 2021.
- [40] X. Zhang, S. Bi, K. Sunkavalli, H. Su, and Z. Xu, “Nerfusion: Fusing radiance fields for large-scale scene reconstruction,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5449–5458, 2022.
- [41] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, “Nice-slam: Neural implicit scalable encoding for slam,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12786–12796, 2022.
- [42] S. Kobayashi, E. Matsumoto, and V. Sitzmann, “Decomposing nerf for editing via feature field distillation,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 23311–23330, 2022.
- [43] Y. Liu, B. Hu, J. Huang, Y.-W. Tai, and C.-K. Tang, “Instance neural radiance field,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 787–796, 2023.
- [44] A. Mirzaei, T. Aumentado-Armstrong, K. G. Derpanis, J. Kelly, M. A. Brubaker, I. Gilitschenski, and A. Levinstein, “Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20669–20679, 2023.
- [45] B. Hu, J. Huang, Y. Liu, Y.-W. Tai, and C.-K. Tang, “Nerf-rpn: A general framework for object detection in nerfs,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23528–23538, 2023.
- [46] C. Xu, B. Wu, J. Hou, S. Tsai, R. Li, J. Wang, W. Zhan, Z. He, P. Vajda, K. Keutzer, et al., “Nerf-det: Learning geometry-aware volumetric representation for multi-view 3d object detection,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23320–23330, 2023.
- [47] Y. Xie, H. Jiang, G. Gkioxari, and J. Straub, “Pixel-aligned recurrent queries for multi-view 3d object detection,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18370–18380, 2023.
- [48] J. Xu, L. Peng, H. Cheng, H. Li, W. Qian, K. Li, W. Wang, and D. Cai, “Mononerd: Nerf-like representations for monocular 3d object detection,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6814–6824, 2023.
- [49] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. De Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis, et al., “Efficient geometry-aware 3d generative adversarial networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16123–16133, 2022.
- [50] E. R. Chan, M. Monteiro, P. Kellnhofer, J. Wu, and G. Wetzstein, “piggan: Periodic implicit generative adversarial networks for 3d-aware image synthesis,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5799–5809, 2021.
- [51] J. Xie, H. Ouyang, J. Piao, C. Lei, and Q. Chen, “High-fidelity 3d gan inversion by pseudo-multi-view optimization,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 321–331, 2023.
- [52] G. Metzler, E. Richardson, O. Patashnik, R. Giryes, and D. Cohen-Or, “Latent-nerf for shape-guided generation of 3d shapes and textures,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12663–12673, 2023.
- [53] Y. Wang, W. Wu, and D. Xu, “Learning unified decompositional and compositional nerf for editable novel view synthesis,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18247–18256, 2023.
- [54] C. Deng, C. Jiang, C. R. Qi, X. Yan, Y. Zhou, L. Guibas, D. Anguelov, et al., “Nerdi: Single-view nerf synthesis with language-guided diffusion as general image priors,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20637–20647, 2023.
- [55] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, “Plenoxels: Radiance fields without neural networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5501–5510, 2022.
- [56] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, “Tensorf: Tensorial radiance fields,” in *European Conference on Computer Vision*, pp. 333–350, Springer, 2022.
- [57] B. Attal, J.-B. Huang, C. Richardt, M. Zollhoefer, J. Kopf, M. O’Toole, and C. Kim, “Hyperreel: High-fidelity 6-dof video with ray-conditioned sampling,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16610–16620, 2023.
- [58] Z. Chen, F. Wang, Y. Wang, and H. Liu, “Text-to-3d using gaussian splatting,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21401–21412, June 2024.
- [59] J. Tang, J. Ren, H. Zhou, Z. Liu, and G. Zeng, “Dreamgaussian: Generative gaussian splatting for efficient 3d content creation,” *arXiv preprint arXiv:2309.16653*, 2023.
- [60] X. Liu, X. Zhan, J. Tang, Y. Shan, G. Zeng, D. Lin, X. Liu, and Z. Liu, “Humangaussian: Text-driven 3d human generation with gaussian splatting,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6646–6657, June 2024.
- [61] X. Zhou, Z. Lin, X. Shan, Y. Wang, D. Sun, and M.-H. Yang, “Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21634–21643, 2024.
- [62] H. Zhou, J. Shao, L. Xu, D. Bai, W. Qiu, B. Liu, Y. Wang, A. Geiger, and Y. Liao, “Hugs: Holistic urban 3d scene understanding via gaussian splatting,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21336–21345, 2024.
- [63] M. Qin, W. Li, J. Zhou, H. Wang, and H. Pfister, “Langsplat: 3d language gaussian splatting,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20051–20060, 2024.
- [64] Y. Cai, Y. Liang, J. Wang, A. Wang, Y. Zhang, X. Yang, Z. Zhou, and A. Yuille, “Radiative gaussian splatting for efficient x-ray novel view synthesis,” *arXiv preprint arXiv:2403.04116*, 2024.
- [65] E. Nikolakakis, U. Gupta, J. Vengosh, J. Bui, and R. Marinescu, “Gaspct: Gaussian splatting for novel ct projection view synthesis,” *arXiv preprint arXiv:2404.03126*, 2024.
- [66] Z. Gao, B. Planche, M. Zheng, X. Chen, T. Chen, and Z. Wu, “Ddgs-ct: Direction-disentangled gaussian splatting for realistic volume rendering,” *arXiv preprint arXiv:2406.02518*, 2024.
- [67] X. Guo, J. Sun, Y. Dai, G. Chen, X. Ye, X. Tan, E. Ding, Y. Zhang, and J. Wang, “Forward flow for novel view synthesis of dynamic scenes,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16022–16033, 2023.
- [68] C. Wang, L. E. MacDonald, L. A. Jeni, and S. Lucey, “Flow supervision for deformable nerf,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21128–21137, 2023.
- [69] L. Song, X. Gong, B. Planche, M. Zheng, D. Doermann, J. Yuan, T. Chen, and Z. Wu, “Pref: Predictability regularized neural motion fields,” in *European Conference on Computer Vision*, pp. 664–681, Springer, 2022.
- [70] J. Zhang, Y. Lan, S. Yang, F. Hong, Q. Wang, C. K. Yeo, Z. Liu, and C. C. Loy, “Deformtoon3d: Deformable neural radiance fields for 3d toonification,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9144–9154, 2023.
- [71] E. Johnson, M. Habermann, S. Shimada, V. Golyanik, and C. Theobalt, “Unbiased 4d: Monocular 4d reconstruction with a neural deformation model,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6597–6606, 2023.
- [72] B. P. Duisterhof, Z. Mandi, Y. Yao, J.-W. Liu, M. Z. Shou, S. Song, and J. Ichnowski, “Md-splatting: Learning metric deformation from 4d gaussians in highly deformable scenes,” *arXiv preprint arXiv:2312.00583*, 2023.
- [73] H. Liu, Y. Liu, C. Li, W. Li, and Y. Yuan, “Lgs: A light-weight 4d gaussian splatting for efficient surgical scene reconstruction,” *arXiv preprint arXiv:2406.16073*, 2024.
- [74] D. Li, S.-S. Huang, Z. Lu, X. Duan, and H. Huang, “St-4dgs: Spatial-temporally consistent 4d gaussian splatting for efficient dynamic scene

- rendering,” in *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.
- [75] Z. Lu, X. Guo, L. Hui, T. Chen, M. Yang, X. Tang, F. Zhu, and Y. Dai, “3d geometry-aware deformable gaussian splatting for dynamic view synthesis,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8900–8910, 2024.
- [76] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, “4d gaussian splatting for real-time dynamic scene rendering,” *arXiv preprint arXiv:2310.08528*, 2023.
- [77] V. Saragadam, D. LeJeune, J. Tan, G. Balakrishnan, A. Veeraraghavan, and R. G. Baraniuk, “Wire: Wavelet implicit neural representations,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18507–18516, 2023.
- [78] R. Zha, X. Cheng, H. Li, M. Harandi, and Z. Ge, “Endosurf: Neural surface reconstruction of deformable tissues with stereo endoscope videos,” in *International conference on medical image computing and computer-assisted intervention*, pp. 13–23, Springer, 2023.
- [79] A. Lou, Y. Li, X. Yao, Y. Zhang, and J. Noble, “Samsnerf: segment anything model (sam) guided dynamic surgical scene reconstruction by neural radiance field (nerf),” in *Medical Imaging 2024: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 12928, pp. 19–23, SPIE, 2024.
- [80] C. Yang, K. Wang, Y. Wang, X. Yang, and W. Shen, “Neural lerplane representations for fast 4d reconstruction of deformable tissues,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 46–56, Springer, 2023.
- [81] C. Yang, K. Wang, Y. Wang, Q. Dou, X. Yang, and W. Shen, “Efficient deformable tissue reconstruction via orthogonal neural plane,” *IEEE Transactions on Medical Imaging*, 2024.
- [82] Y. Liu, C. Li, C. Yang, and Y. Yuan, “Endogaussian: Gaussian splatting for deformable surgical scene reconstruction,” *arXiv preprint arXiv:2401.12561*, 2024.
- [83] A. Yu, R. Li, M. Tancik, H. Li, R. Ng, and A. Kanazawa, “Plenoctrees for real-time rendering of neural radiance fields,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5752–5761, 2021.
- [84] U. Sara, M. Akter, and M. S. Uddin, “Image quality assessment through fsim, ssim, mse and psnr—a comparative study,” *Journal of Computer and Communications*, vol. 7, no. 3, pp. 8–18, 2019.
- [85] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.
- [86] S. Lombardi, T. Simon, J. Saragih, G. Schwartz, A. Lehrmann, and Y. Sheikh, “Neural volumes: Learning dynamic renderable volumes from images,” *arXiv preprint arXiv:1906.07751*, 2019.
- [87] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar, “Local light field fusion: Practical view synthesis with prescriptive sampling guidelines,” *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–14, 2019.
- [88] L. Song, A. Chen, Z. Li, Z. Chen, L. Chen, J. Yuan, Y. Xu, and A. Geiger, “Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 5, pp. 2732–2742, 2023.
- [89] F. Wang, S. Tan, X. Li, Z. Tian, Y. Song, and H. Liu, “Mixed neural voxels for fast multi-view video synthesis,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19706–19716, 2023.
- [90] J. Fang, T. Yi, X. Wang, L. Xie, X. Zhang, W. Liu, M. Nießner, and Q. Tian, “Fast dynamic radiance fields with time-aware neural voxels,” in *SIGGRAPH Asia 2022 Conference Papers*, pp. 1–9, 2022.
- [91] L. Liu, J. Gu, K. Zaw Lin, T.-S. Chua, and C. Theobalt, “Neural sparse voxel fields,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 15651–15663, 2020.
- [92] C. Reiser, S. Peng, Y. Liao, and A. Geiger, “Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14335–14345, 2021.
- [93] J. Tang, X. Chen, J. Wang, and G. Zeng, “Compressible-composable nerf via rank-residual decomposition,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 14798–14809, 2022.
- [94] T. Bird, J. Ballé, S. Singh, and P. A. Chou, “3d scene compression through entropy penalized neural representation functions,” in *2021 Picture Coding Symposium (PCS)*, pp. 1–5, IEEE, 2021.
- [95] B. Huang, X. Yan, A. Chen, S. Gao, and J. Yu, “Pref: Phasorial embedding fields for compact neural representations,” *arXiv preprint arXiv:2205.13524*, 2022.
- [96] P. Mountney, D. Stoyanov, and G.-Z. Yang, “Three-dimensional tissue deformation recovery and tracking,” *IEEE Signal Processing Magazine*, vol. 27, no. 4, pp. 14–24, 2010.
- [97] D. Stoyanov, G. P. Mylonas, F. Deligianni, A. Darzi, and G. Z. Yang, “Soft-tissue motion tracking and structure estimation for robotic assisted mis procedures,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 139–146, Springer, 2005.
- [98] D. Recasens, J. Lamarca, J. M. Fácil, J. Montiel, and J. Civera, “Endo-depth-and-motion: Reconstruction and tracking in endoscopic videos using depth networks and photometric constraints,” *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7225–7232, 2021.
- [99] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., “Segment anything,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.
- [100] Y. Long, Z. Li, C. H. Yee, C. F. Ng, R. H. Taylor, M. Unberath, and Q. Dou, “E-dssr: efficient dynamic surgical scene reconstruction with transformer-based stereoscopic depth perception,” in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part IV 24*, pp. 415–425, Springer, 2021.
- [101] M. Allan, J. Mcleod, C. Wang, J. C. Rosenthal, Z. Hu, N. Gard, P. Eisert, K. X. Fu, T. Zeffiro, W. Xia, et al., “Stereo correspondence and reconstruction of endoscopic data challenge,” *arXiv preprint arXiv:2101.01133*, 2021.
- [102] A. Lou, K. Tawfik, X. Yao, Z. Liu, and J. Noble, “Min-max similarity: A contrastive semi-supervised deep learning network for surgical tools segmentation,” *IEEE Transactions on Medical Imaging*, vol. 42, no. 10, pp. 2832–2841, 2023.
- [103] A. Lou, X. Yao, Z. Liu, J. Han, and J. Noble, “Self-supervised surgical instrument 3d reconstruction from a single camera image,” in *Medical Imaging 2023: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 12466, pp. 102–107, SPIE, 2023.
- [104] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [105] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, 2012.
- [106] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [107] T. Müller, B. McWilliams, F. Rousselle, M. Gross, and J. Novák, “Neural importance sampling,” *ACM Transactions on Graphics (ToG)*, vol. 38, no. 5, pp. 1–19, 2019.
- [108] Y. Li, A. Lou, Z. Xu, S. Zhang, S. Wang, D. J. Englot, S. Kolouri, D. Moyer, R. G. Bayrak, and C. Chang, “Neurobolt: Resting-state eeg-to-fmri synthesis with multi-dimensional feature mapping,” *arXiv preprint arXiv:2410.05341*, 2024.
- [109] Y. Li, A. Lou, Z. Xu, S. Wang, and C. Chang, “Leveraging sinusoidal representation networks to predict fmri signals from eeg,” in *Medical Imaging 2024: Image Processing*, vol. 12926, pp. 795–800, SPIE, 2024.
- [110] A. Lou, S. Guan, H. Ko, and M. H. Loew, “Caranet: context axial reverse attention network for segmentation of small medical objects,” in *Medical Imaging 2022: Image Processing*, vol. 12032, pp. 81–92, SPIE, 2022.
- [111] A. Lou, S. Guan, and M. Loew, “Dc-unet: rethinking the u-net architecture with dual channel efficient cnn for medical image segmentation,” in *Medical Imaging 2021: Image Processing*, vol. 11596, pp. 758–768, SPIE, 2021.
- [112] A. Lou, S. Guan, and M. Loew, “Caranet: context axial reverse attention network for segmentation of small medical objects,” *Journal of Medical Imaging*, vol. 10, no. 1, pp. 014005–014005, 2023.
- [113] A. Lou, Y. Li, Y. Zhang, R. F. Labadie, and J. Noble, “Zero-shot surgical tool segmentation in monocular video using segment anything model 2,” *arXiv preprint arXiv:2408.01648*, 2024.
- [114] A. Lou and J. Noble, “Ws-sfmlearner: self-supervised monocular depth and ego-motion estimation on surgical videos with unknown camera parameters,” in *Medical Imaging 2024: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 12928, pp. 119–127, SPIE, 2024.
- [115] A. Lou, Y. Li, Y. Zhang, and J. Noble, “Surgical depth anything: Depth estimation for surgical scenes using foundation models,” *arXiv preprint arXiv:2410.07434*, 2024.

VI. SUPPLEMENTARY MATERIAL

In this supplementary material, we provide further methodological context and implementation details to facilitate reproducibility of our framework DaReNeRF. We also showcase additional quantitative and qualitative results to further highlight the contributions claimed in the paper.

A. Video Presentation

A video presentation of DaReNeRF and its results can be found online, at <https://www.youtube.com/watch?v=hYQsl6Rbxn4>.

B. Dual-Tree Complex Wavelet Transform

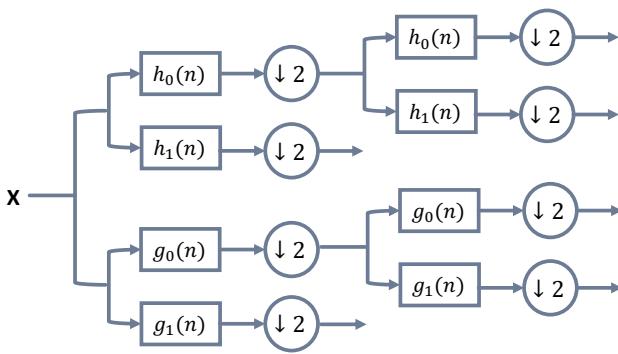


Fig. 9. Analysis filter bank, for the dual tree complex wavelet transfrom.

The idea of dual-tree complex wavelet transform (DTCWT) [30] is quite straightforward. The DTCWT employs two real discrete wavelet transforms (DWTs). The first DWT gives the real part of the transform while the second DWT gives the imaginary part. The analysis filter banks used to implement the DTCWT is illustrated in Figure 9. Here $h_0(n)$, $h_1(n)$ denote the low-pass/high-pass filter pair for upper filter bank, and $g_0(n)$, $g_1(n)$ denote the low-pass/high-pass filter pair for the lower filter bank. The two real wavelets associated with each of the two real wavelet transforms as $\psi_h(t)$ and $\psi_g(t)$. And the complex wavelet can be denoted as $\psi(t) = \psi_h(t) + j\psi_g(t)$. The $\psi_g(t)$ is approximately the Hilbert transform of $\psi_h(t)$. The 2D DTCWT $\psi(x, y) = \psi(x)\psi(y)$ associated with the row-column implementation of the wavelet transform, where $\psi(x)$ is a complex wavelet given by $\psi(x) = \psi_h(x) + j\psi_g(x)$. Then we obtain for $\psi(x, y)$ the expression:

$$\begin{aligned} \psi(x, y) &= [\psi_h(x) + j\psi_g(x)][\psi_h(y) + j\psi_g(y)] \\ &= \psi_h(x)\psi_h(y) - \psi_g(x)\psi_g(y) \\ &\quad + j[\psi_g(x)\psi_h(y) + \psi_h(x)\psi_g(y)] \end{aligned} \quad (10)$$

The spectrum of $\psi_h(x)\psi_h(y) - \psi_g(x)\psi_g(y)$ which corresponds to the real part of $\psi(x, y)$ is supported in two quadrants of the 2D frequency plane and is oriented at -45° . Note that the $\psi_h(x)\psi_h(y)$ is the HH wavelet of a separable 2D real wavelet transform implemented using the filter pair $\{h_0(n), h_1(n)\}$. Similarly, $\psi_g(x)\psi_g(y)$ is the HH wavelet of a real separable wavelet transform, implemented using the filters $\{g_0(n), g_1(n)\}$. To obtain a real 2D wavelet oriented at $+45^\circ$, we consider now the complex 2-D wavelet

$\psi(x, y) = \psi(x)\overline{\psi(y)}$, where $\overline{\psi(y)}$ represents the complex conjugate of $\psi(y)$. This gives us the following expression:

$$\begin{aligned} \psi(x, y) &= [\psi_h(x) + j\psi_g(x)][\overline{\psi_h(y)} + \overline{j\psi_g(y)}] \\ &= \psi_h(x)\psi_h(y) + \psi_g(x)\psi_g(y) \\ &\quad + j[\psi_g(x)\psi_h(y) - \psi_h(x)\psi_g(y)] \end{aligned} \quad (11)$$

The spectrum of $\psi_h(x)\psi_h(y) + \psi_g(x)\psi_g(y)$ is supported in two quadrants of the 2D frequency plane and is oriented at $+45^\circ$. We could obtain four more oriented real 2D wavelets by repeating the above procedure on the following complex 2-D wavelets: $\phi(x)\psi(y)$, $\psi(x)\phi(y)$, $\phi(x)\overline{\psi(y)}$ and $\psi(x)\overline{\phi(y)}$, where $\psi(x) = \psi_h(x) + j\psi_g(y)$ and $\phi(x) = \phi_h(x) + j\phi_g(y)$. By taking the real part of each of these four complex wavelets, we obtain four real oriented 2D wavelets, in addition to the two already obtain in 10 and 11:

$$\psi_i(x, y) = \frac{1}{\sqrt{2}}(\psi_{1,i}(x, y) - \psi_{2,i}(x, y)), \quad (12)$$

$$\psi_{i+3}(x, y) = \frac{1}{\sqrt{2}}(\psi_{1,i}(x, y) + \psi_{2,i}(x, y)) \quad (13)$$

for $i = 1, 2, 3$, where the two separable 2-D wavelet bases are defined in the usual manner:

$$\begin{aligned} \psi_{1,1}(x, y) &= \phi_h(x)\psi_h(y), \psi_{2,1}(x, y) = \phi_g(x)\psi_g(y), \\ \psi_{1,2}(x, y) &= \psi_h(x)\phi_h(y), \psi_{2,2}(x, y) = \psi_g(x)\phi_g(y), \\ \psi_{1,3}(x, y) &= \psi_h(x)\psi_h(y), \psi_{2,3}(x, y) = \psi_g(x)\psi_g(y), \end{aligned} \quad (14)$$

We have used the normalization $\frac{1}{\sqrt{2}}$ only so that the sum and difference operation constitutes an orthonormal operation. From the imaginary parts of $\psi(x)\psi(y)$, $\psi(x)\overline{\psi(y)}$, $\phi(x)\psi(y)$, $\psi(x)\phi(y)$, $\phi(x)\overline{\psi(y)}$ and $\psi(x)\overline{\phi(y)}$, we could obtain six oriented wavelets given by:

$$\psi_i(x, y) = \frac{1}{\sqrt{2}}(\psi_{3,i}(x, y) + \psi_{4,i}(x, y)), \quad (15)$$

$$\psi_{i+3}(x, y) = \frac{1}{\sqrt{2}}(\psi_{3,i}(x, y) - \psi_{4,i}(x, y)) \quad (16)$$

for $i = 1, 2, 3$, where the two separable 2D wavelet bases are defined as:

$$\begin{aligned} \psi_{3,1}(x, y) &= \phi_g(x)\psi_h(y), \psi_{4,1}(x, y) = \phi_h(x)\psi_g(y), \\ \psi_{3,2}(x, y) &= \psi_g(x)\phi_h(y), \psi_{4,2}(x, y) = \psi_h(x)\phi_g(y), \\ \psi_{3,3}(x, y) &= \psi_g(x)\psi_h(y), \psi_{4,3}(x, y) = \psi_h(x)\psi_g(y), \end{aligned} \quad (17)$$

Thus we could obtain six oriented wavelets from both real and imaginary part.

C. Additional Results on Various Datasets

1) *Plenoptic Video Dataset* [7]: The quantitative results for each scene are presented in Table XII, while additional visualizations comparing DaReNeRF with current state-of-the-art methods, HexPlane [16] and K-Planes [17], are provided in Figure 11. Notably, DaReNeRF demonstrates superior recovery of texture details. We also provide an animated qualitative comparison in Figure 10. Furthermore, comprehensive visualizations of DaReNeRF on all six scenes in the Plenoptic dataset are shown in Figure 13 and Figure 14.

Fig. 10. **Visual comparison on dynamic scenes (Plenoptic data).** K-Planes and HexPlane are concurrent decomposition-based methods. As shown in the four zoomed-in patches, our method better reconstruct fine details and captures motion. To see the figure animated, please view the document with compatible software, e.g., *Adobe Acrobat* or *KDE Okular*.

TABLE XII
RESULTS ON PLENOPTIC VIDEO DATASET. WE REPORT RESULTS OF EACH SCENE

Model	Flame Salmon			Cook Spinach			Cut Roasted Beef		
	PSNR ↑	D-SSIM ↓	LPIPS ↓	PSNR ↑	D-SSIM ↓	LPIPS ↓	PSNR ↑	D-SSIM ↓	LPIPS ↓
DaReNeRF-S	30.294	0.015	0.089	32.630	0.013	0.100	33.087	0.013	0.092
DaReNeRF	30.441	0.012	0.084	32.836	0.011	0.090	33.200	0.011	0.091
	Flame Steak			Sear Steak			Coffee Martini		
DaReNeRF-S	33.259	0.011	0.081	33.179	0.011	0.075	30.160	0.016	0.092
DaReNeRF	33.524	0.009	0.077	33.351	0.009	0.072	30.193	0.014	0.089

2) *D-NeRF Dataset* [13]: We provide quantitative results for each scene in Table XIII, while additional visualizations comparing DaReNeRF with current state-of-the-art methods, HexPlane [16] and 4D-GS [76], are shared in Figure 15. We also provide further visualization in a video attached to this supplementary material. Remarkably, although 4D-GS incorporates a deformation field, DaReNeRF still outperforms it in certain cases from the D-NeRF dataset. Furthermore, comprehensive visualizations of DaReNeRF on six scenes in the Plenoptic dataset are shown in Figure 16 and the failure cases are shown in Figure 17.

3) *NeRF Synthetic Dataset*: The quantitative results for each case are presented in Table XIV, while additional visualizations comparing our representation with DWT [22] based representation method, are shown in Figure 18. Furthermore, comprehensive visualizations of eight scenes in the NeRF dataset are shown in Figure 19 and in the attached video.

4) *NSVF Synthetic Dataset*: The quantitative results for each case are presented in Table XV, while additional visualizations comparing our representation with DWT [22] based representation method, are shown in Figure 20. Furthermore, comprehensive visualizations of eight scenes in the NSVF dataset are shown in Figure 21.

5) *LLFF Dataset*: The quantitative results for each case are presented in Table XVI, while additional visualizations comparing our representation with DWT [22] based representation method, are shown in Figure 22. Furthermore, comprehensive visualizations of eight scenes in the NSVF dataset are shown in Figure 23 and in the video.

6) *EndoNeRF Dataset* [2]: The quantitative results for two cases are present in Table XVIII and the visualization results are presented in the main paper.

7) *Hamlyn Dataset* [98]: The quantitative results of 7 cases and average performance are shown in the Table XIX

8) *SCARED Dataset*: The quantitative results of 5 sequences and average performance are shown in the Table XX.

9) *Cochlear Implant Dataset*: The quantitative results of 2 cases are shown in the Table XXI.

D. Additional Ablation Studies

1) *Sparsity Masks*: We evaluate the performance of our direction-aware representation at various sparsity levels controlled by the mask loss weight λ_m . The quantitative and qualitative results on the NSVF dataset with different sparsity levels are presented in Table XVII and Figure 12.

2) *Wavelet Levels*: We investigated the impact of scene reconstruction performance across different wavelet levels, and the results are presented in Table XXII. Interestingly, we observed that increasing the wavelet level did not lead to significant performance improvements. Conversely, we noted a substantial increase in both training time and model size with the increment of wavelet level. As a result, throughout all experiments, we consistently set the wavelet level to 1.

3) *Training Data Sparsity Analysis*: In order to delve deeper into the few-shot capabilities of our proposed direction-aware representation, we conducted experiments with varying levels of training data sparsity. This was achieved by randomly dropping training frames while ensuring sufficient data remained to effectively learn motion on the D-NeRF dataset. The corresponding results are presented in Table XXIII. Remarkably, our proposed DaReNeRF consistently outperforms the baseline across different levels of training data sparsity.

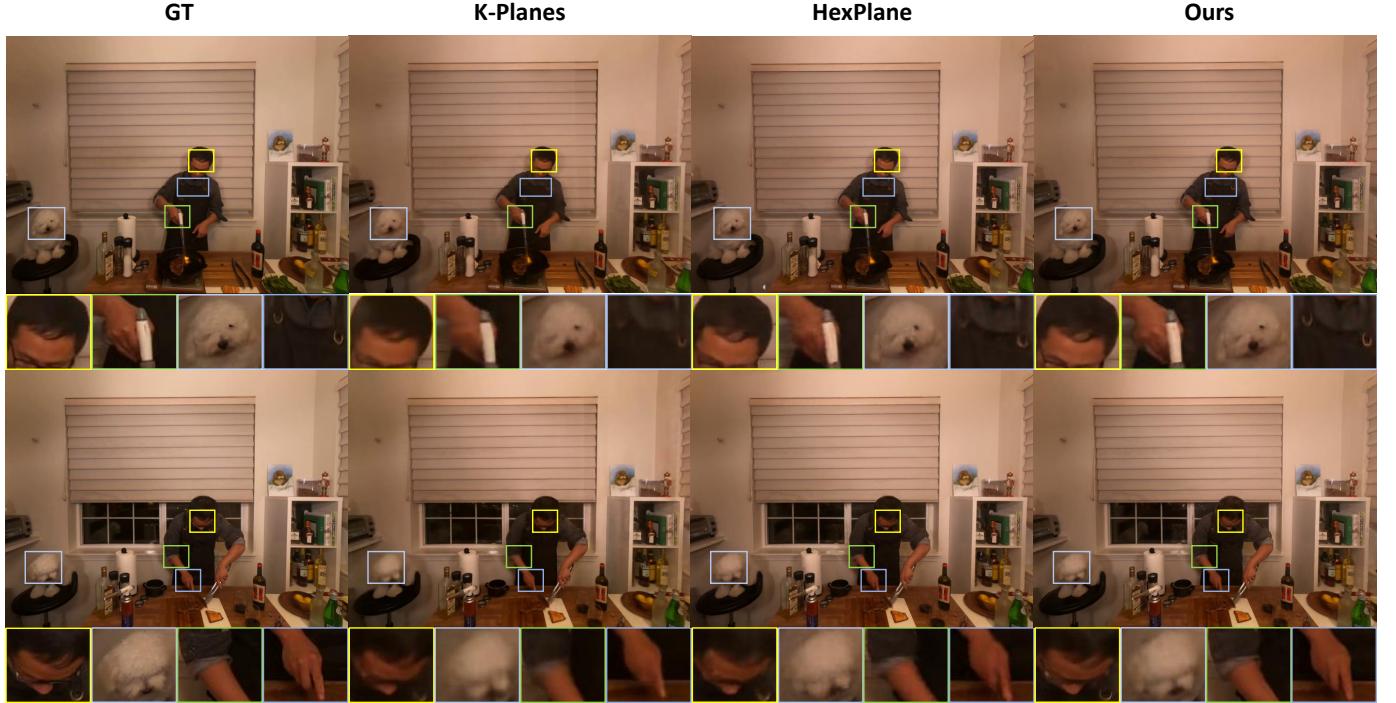


Fig. 11. Visual comparison on dynamic scenes (Plenoptic data). K-Planes and HexPlane are concurrent decomposition-based methods. As shown in the four zoomed-in patches, our method better reconstructs fine details and captures motion.

E. Training Details

1) *Plenoptic Video Dataset* [7]: Plenoptic Video Dataset is a multi-view real-world video dataset, where each video is 10-second long. For training, we set $R_1 = 48$, $R_2 = 48$ and $R_3 = 48$ for appearance, where R_1 , R_2 and R_3 are basis numbers for direction-aware representation of $XY - ZT$, $XZ - YT$ and $YZ - XT$ planes. For opacity, we set $R_1 = 24$, $R_2 = 24$ and $R_3 = 24$. The scene is modeled using normalized device coordinate (NDC) [5] with min boundaries $[-2.5, -2.0, -1.0]$ and max boundaries $[2.5, 2.0, 1.0]$.

During the training, DaReNeRF starts with a space grid size of 64^3 and double its resolution at 20k, 40k and 70k to 512^3 . The emptiness voxel is calculated at 30k, 50k and 80k. The learning rate for representation planes is 0.02 and the learning rate for V^{RF} and neural network is 0.001. All learning rates are exponentially decayed. We use Adam [104] optimization with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. We apply the total variational loss on all representation planes with loss weight $\lambda = 1e - 5$ for spatial planes and $\lambda = 2e - 5$ for spatial-temporal planes. For DaReNeRF-S we set weight of mask loss as $1e - 11$.

We follow the hierarchical training pipeline suggested in [7]. Both DaReNeRF and DaReNeRF-S use 100k iterations, with 10k stage one training, 50k stage two training and 40k stage three training. Stage one is a global-median-based weighted sampling with $\gamma = 0.02$; stage two is also a global-median based weighted sampling with $\gamma = 0.02$; stage three is a temporal-difference-based weighted sampling with $\gamma = 0.2$.

In evaluation, D-SSIM is computed as $\frac{1 - MS - SSIM}{2}$ and LPIPS [85] is calculated using AlexNet [105].

2) *D-NeRF Dataset* [13]: We set $R_1 = 48$, $R_2 = 48$ and $R_3 = 48$ for appearance and $R_1 = 24$, $R_2 = 24$ and $R_3 = 24$ for opacity. The bounding box has max boundaries $[1.5, 1.5, 1.5]$ and min boundaries $[-1.5, -1.5, -1.5]$. During the training, both DaReNeRF and DaReNeRF-S starts with space grid of 32^3 and upsampling its resolution at 3k, 6k and 9k to 200^3 . The emptiness voxel is calculated at 4k, 8k and 10k iterations. Total training iteration is 25k. The learning rate for representation planes are 0.02 and learning rate for V^{RF} and neural network is 0.001. All learning rates are exponentially decayed. We use Adam [104] optimization with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. In evaluation, LPIPS [85] is calculated using VGG-Net [106] following previous works.

For **both** the Plenoptic Video dataset and the D-NeRF dataset, we set the learning rate of the masks in DaReNeRF-S same as their representation planes and we employ a compact MLP for regressing output colors. The MLP consists of 3 layers, with a hidden dimension of 128.

3) *Static Scene*: For three static scene datasets NeRF synthetic dataset, NSVF synthetic dataset and LLFF dataset, we followed the experimental settings of TensoRF [56]. We trained our model for 30000 iterations, each of which is a minibatch of 4096 rays. We used Adam [104] optimization with $\beta_1 = 0.9$ and $\beta_2 = 0.99$ and an exponential learning rate decay scheduler. The initial learning rates of representation-related parameters and neural network (MLP) were set to 0.02 and 0.001. For the **NeRF synthetic** and **NSVF synthetic** datasets, we adopt TensoRF-192 as the baseline and update the alpha masks at the 2k, 4k, 6k, 11k, 16k, and 26k iterations. The

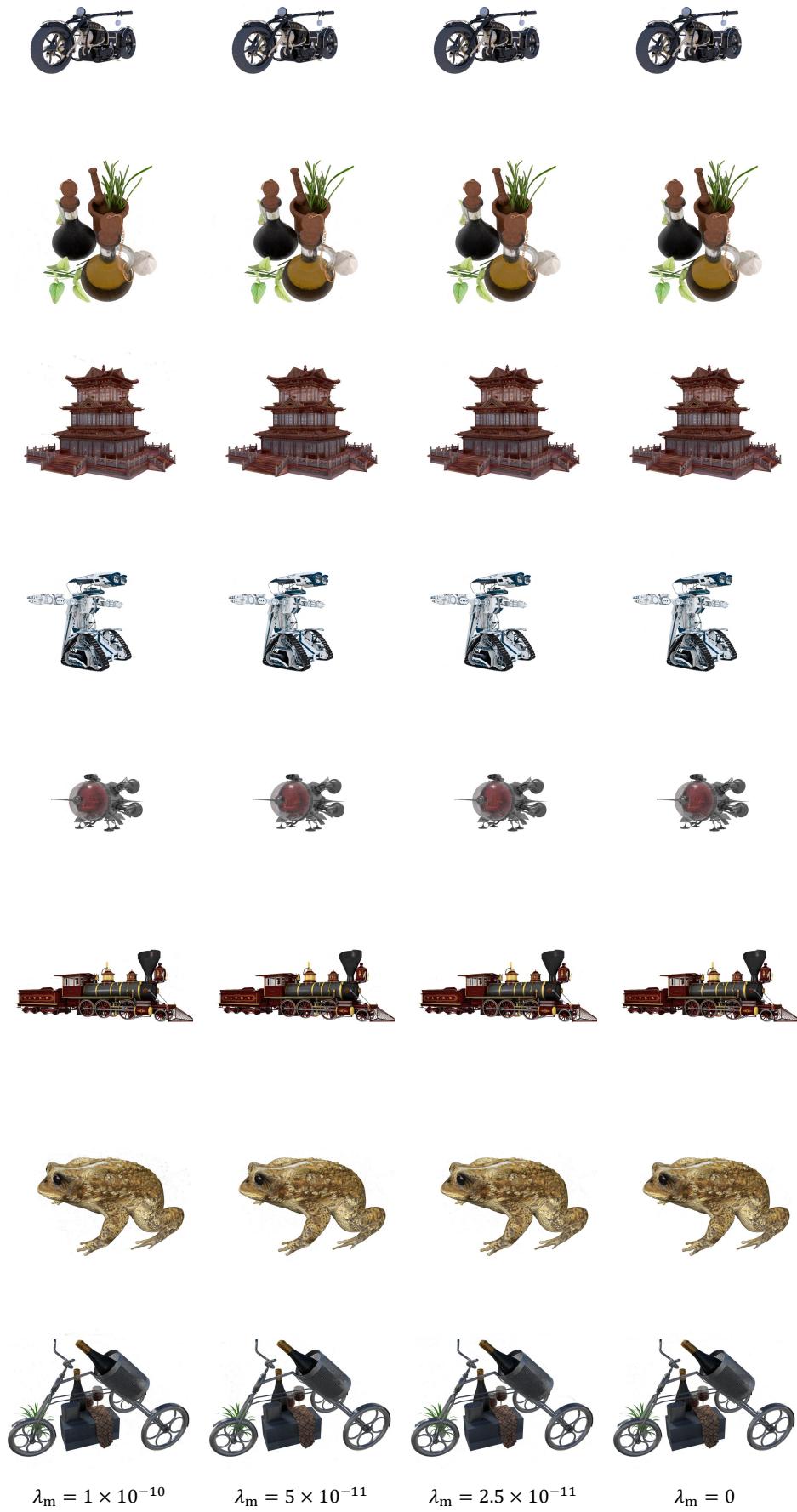


Fig. 12. Qualitative results on NSVF dataset with different sparsity.

TABLE XIII
RESULTS OF D-NeRF DATASET. WE REPORT RESULTS OF EACH SCENE

Model	Hell Warrior			Mutant			Hook		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
T-NeRF	23.19	0.93	0.08	30.56	0.96	0.04	27.21	0.94	0.06
D-NeRF	25.02	0.95	0.06	31.29	0.97	0.02	29.25	0.96	0.11
TiNeuVox-S	27.00	0.95	0.09	31.09	0.96	0.05	29.30	0.95	0.07
TiNeuVox-B	28.17	0.97	0.07	33.61	0.98	0.03	31.45	0.97	0.05
HexPlane	24.24	0.94	0.07	33.79	0.98	0.03	28.71	0.96	0.05
DaReNeRF-S	25.71	0.95	0.04	34.08	0.98	0.02	29.04	0.96	0.04
DaReNeRF	25.82	0.95	0.04	34.17	0.98	0.01	28.96	0.96	0.04
Bouncing Balls									
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
	T-NeRF	37.81	0.98	0.12	23.82	0.90	0.15	30.19	0.96
D-NeRF	38.93	0.98	0.10	21.64	0.83	0.16	31.75	0.97	0.03
TiNeuVox-S	39.05	0.99	0.06	24.35	0.88	0.13	29.95	0.96	0.06
TiNeuVox-B	40.73	0.99	0.04	25.02	0.92	0.07	32.70	0.98	0.03
HexPlane	39.69	0.99	0.03	25.22	0.94	0.04	30.67	0.98	0.03
DaReNeRF-S	42.24	0.99	0.01	25.24	0.94	0.03	31.75	0.98	0.03
DaReNeRF	42.26	0.99	0.01	25.44	0.95	0.03	32.21	0.98	0.02
Stand Up									
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
	T-NeRF	31.24	0.97	0.02	32.01	0.97	0.03	29.51	0.95
D-NeRF	32.79	0.98	0.02	32.80	0.98	0.03	30.50	0.95	0.07
TiNeuVox-S	32.89	0.98	0.03	32.33	0.97	0.04	30.75	0.96	0.07
TiNeuVox-B	35.43	0.99	0.02	34.23	0.98	0.03	32.64	0.97	0.04
HexPlane	34.36	0.98	0.02	31.65	0.97	0.04	31.04	0.94	0.04
DaReNeRF-S	34.47	0.98	0.02	31.99	0.97	0.03	31.82	0.97	0.03
DaReNeRF	34.58	0.98	0.02	32.21	0.97	0.03	31.95	0.97	0.03

initial grid size is set to 128^3 , and we perform upsampling at 2k, 3k, 4k, 5.5k, and 7k iterations, reaching a final resolution of 300^3 . For the LLFF dataset, we adopt TensoRF-96 as the baseline and update the alpha masks at the 2.5k, 4k, 6k, 11k, 16k, and 21k iterations. The initial grid size is set to 128^3 , and we perform upsampling at 2k, 3k, 4k and 5.5k iterations, reaching a final resolution of 640^3 . The learning rates of masks are set same as learning rates of representation-related parameters. We employ a compact MLP for regressing output colors. The MLP consists of 3 layers, with a hidden dimension of 128.

4) *Dynamic Surgical Scene*: For the surgical dynamic scene datasets, we test our DaReNeRF and DaReGS models, following the settings of ForPlane [81] and EndoGaussian [82], respectively.

DaReNeRF. The surgical scene is normalized into normalized device coordinates (NDC), and the video duration is normalized to $[-1, 1]$. The dimensions for the two stages of point sampling by a *Sample – Net* [81] are 128 and 256, respectively. A one-blob [107] encoding is applied to encode the spatiotemporal information. The full model employs a multi-resolution strategy with spatial axis resolutions of 64, 128, 256, 512, while the temporal size is fixed at 100.

The basis number for all spatiotemporal planes is set to 32. An Adam optimizer with an initial learning rate of 0.01 and a batch size of 2048 is selected for training.

DaReGS. We randomly sample 0.1% of the points as initialization points and select Adam as the optimizer with an initial learning rate of 1.6×10^{-3} . A warmup strategy is employed, where the Gaussian is first optimized for 1k iterations, followed by optimization of the entire framework for 3k iterations.

F. Future works

Future work could explore incorporating various frequency-based representations [108], [109] and integrating segmentation masks [102], [110]–[113] with high-quality depth maps [114], [115]. This approach holds potential to significantly accelerate training and improve the accuracy of reconstruction.

TABLE XIV
RESULTS OF NERF SYNTHETIC DATASET.

Bit Precision	Method	Size(MB)	Avg	Chair	Drums	Ficus	Hotdog	Lego	Materials	Mic	Ship
32-bit	KiloNeRF	≤ 100	31.00	32.91	25.25	29.76	35.56	33.02	29.20	33.06	29.23
32-bit	CCNeRF (CP)	4.4	30.55	-	-	-	-	-	-	-	-
8-bit*	NeRF	1.25	31.52	33.82	24.94	30.33	36.70	32.96	29.77	34.41	29.25
8-bit	cNeRF	0.70	30.49	32.28	24.85	30.58	34.95	31.98	29.17	32.21	28.24
8-bit*	PREF	9.88	31.56	34.55	25.15	32.17	35.73	34.59	29.09	32.64	28.58
8-bit*	VM-192	17.93	32.91	35.64	25.98	33.57	37.26	36.04	29.87	34.33	30.64
8-bit*	VM-192 (300) + DWT	0.83	31.95	34.14	25.53	32.87	36.08	34.93	29.42	33.48	29.15
8-bit*	VM-192 (300) + Ours	8.91	32.42	36.05	29.40	35.26	36.37	25.58	33.26	29.82	33.63

TABLE XV
RESULTS OF NSVF SYNTHETIC DATASET.

Bit Precision	Method	Size(MB)	Avg	Bike	Lifestyle	Palace	Robot	Spaceship	Steamtrain	Toad	Wineholder
32-bit	KiloNeRF	≤ 100	33.77	35.49	33.15	34.42	32.93	36.48	33.36	31.41	29.72
8-bit*	VM-192	17.77	36.11	38.69	34.15	37.09	37.99	37.66	37.45	34.66	31.16
8-bit*	VM-48	4.53	34.95	37.55	33.34	35.84	36.60	36.38	36.68	32.97	30.26
8-bit*	CP-384	0.72	33.92	36.29	32.29	35.73	35.63	34.58	35.82	31.24	29.75
8-bit*	VM-192 (300) + DWT	0.87	34.67	37.06	33.44	35.18	35.74	37.01	36.65	32.23	30.08
8-bit*	VM-192 (300) + Ours	8.98	36.24	38.78	34.21	37.22	38.02	38.61	37.79	34.39	30.97

TABLE XVI
RESULTS OF LLFF DATASET.

Bit Precision	Method	Size(MB)	Avg	Fern	Flower	Fortress	Horns	Leaves	Orchids	Room	T-Rex
8-bit	cNeRF	0.96	26.15	25.17	27.21	31.15	27.28	20.95	20.09	30.65	26.72
8-bit*	PREF	9.34	24.50	23.32	26.37	29.71	25.24	20.21	19.02	28.45	23.67
8-bit*	VM-96	44.72	26.66	25.22	28.55	31.23	28.10	21.28	19.87	32.17	26.89
8-bit*	VM-48	22.40	26.46	25.27	28.19	31.06	27.59	21.33	20.03	31.70	26.54
8-bit*	CP-384	0.64	25.51	24.30	26.88	30.17	26.46	20.38	19.95	30.61	25.35
8-bit*	VM-96 (640) + DWT	1.34	25.88	24.98	27.19	30.28	26.96	21.21	19.93	30.03	26.45
8-bit*	VM-96 (640) + Ours	13.67	26.48	25.02	28.23	31.07	27.81	21.24	19.68	31.82	26.97

TABLE XVII
QUANTITATIVE RESULTS ON NSVF DATASET WITH DIFFERENT SPARSITY.

Sparsity	λ_m	Size(MB)	Avg	Bike	Lifestyle	Palace	Robot	Spaceship	Steamtrain	Toad	Wineholder
99.2%	1.0×10^{-10}	1.16	35.36	38.01	33.69	35.70	37.23	37.83	37.26	32.58	30.56
97.3%	5.0×10^{-11}	3.18	35.81	38.52	34.01	36.33	37.79	38.22	37.46	33.33	30.82
94.2%	2.5×10^{-11}	8.98	36.24	38.78	34.21	37.22	38.02	38.61	37.79	34.39	30.97
-	0	135	36.34	38.86	34.37	37.25	38.06	38.72	37.89	34.46	31.09

TABLE XVIII
QUANTITATIVE RESULTS ON ENDONERF DATASET.

Model	Cutting			Training Time (min) ↓	PSNR ↑	SSIM ↑	Pulling		
	PSNR ↑	SSIM ↑	LPIPS ↓				LPIPS ↓	Training Time (min) ↓	
ForPlane	33.68	0.900	0.113	4	36.26	0.936	0.085	4	
DaReNeRF	35.34	0.922	0.096	5	38.03	0.947	0.064	5	
EndoGaussian	38.10	0.962	0.047	2.5	37.00	0.957	0.070	2.5	
DaReGS	38.38	0.965	0.031	3.5	38.32	0.967	0.049	3.5	

TABLE XIX
QUANTITATIVE RESULTS ON HAMLYN DATASET.

Model	Sequence 1			Sequence 2			Sequence 3			
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	
ForPlane	32.86	0.919	0.124	33.88	0.932	0.125	33.66	0.933	0.123	
DaReNeRF	33.12	0.926	0.121	34.39	0.941	0.113	34.01	0.938	0.119	
Sequence 4		Sequence 5		Sequence 6						
Model	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	
ForPlane	37.89	0.963	0.075	38.90	0.971	0.041	35.24	0.945	0.077	
DaReNeRF	38.65	0.971	0.058	39.29	0.973	0.038	35.90	0.959	0.065	
Sequence 7		Averge								
Model	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓				
ForPlane	34.67	0.949	0.089	35.30	0.945	0.093				
DaReNeRF	35.12	0.956	0.085	35.64	0.952	0.085				

TABLE XX
QUANTITATIVE RESULTS ON SCARED DATASET.

Model	PSNR ↑	SSIM ↑	Sequence 1		Sequence 2				
			LPIPS ↓	Training Time (min) ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Training Time (min) ↓	
EndoGaussian	30.212	0.870	0.154	2.5	32.266	0.897	0.126	2.5	
DaReGS	30.384	0.876	0.114	3.5	33.213	0.911	0.082	3.5	
Sequence 3		Sequence 4							
Model	PSNR ↑	SSIM ↑	LPIPS ↓	Training Time (min) ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Training Time (min) ↓	
EndoGaussian	19.523	0.627	0.533	2.5	25.819	0.868	0.373	2.5	
DaReGS	20.551	0.646	0.468	3.5	26.336	0.870	0.337	3.5	
Sequence 5		Average							
Model	PSNR ↑	SSIM ↑	LPIPS ↓	Training Time (min) ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Training Time (min) ↓	
EndoGaussian	26.925	0.874	0.204	2.5	26.949	0.827	0.278	2.5	
DaReGS	27.050	0.876	0.188	3.5	27.500	0.836	0.238	3.5	

TABLE XXI
QUANTITATIVE RESULTS ON COCHLEAR IMPLANT DATASET.

Model	PSNR ↑	SSIM ↑	Case 1		Case 2			
			LPIPS ↓	Training Time (min) ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Training Time (min) ↓
EndoGaussian	33.985	0.945	0.057	2.5	34.063	0.950	0.072	2.5
DaReGS	34.422	0.953	0.072	3.5	34.350	0.951	0.065	3.5

TABLE XXII
WAVELET-LEVEL ANALYSIS OF DIRECTION-AWARE REPRESENTATION,
EVALUATED ON NVSF DATA.

Level	PSNR \uparrow	Model Size (MB) \downarrow	Training Time (min) \downarrow
1	36.34	135	23
2	36.45	152	41
3	36.49	163	55

TABLE XXIII
EVALUATION ON D-NERF WITH VARIOUS TRAINING SET SPARSITY.

Model	75% training set (average)			50% training set (average)		
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
HexPlane	29.85	0.95	0.05	28.03	0.94	0.06
DaReNeRF	30.95	0.96	0.04	29.28	0.96	0.05

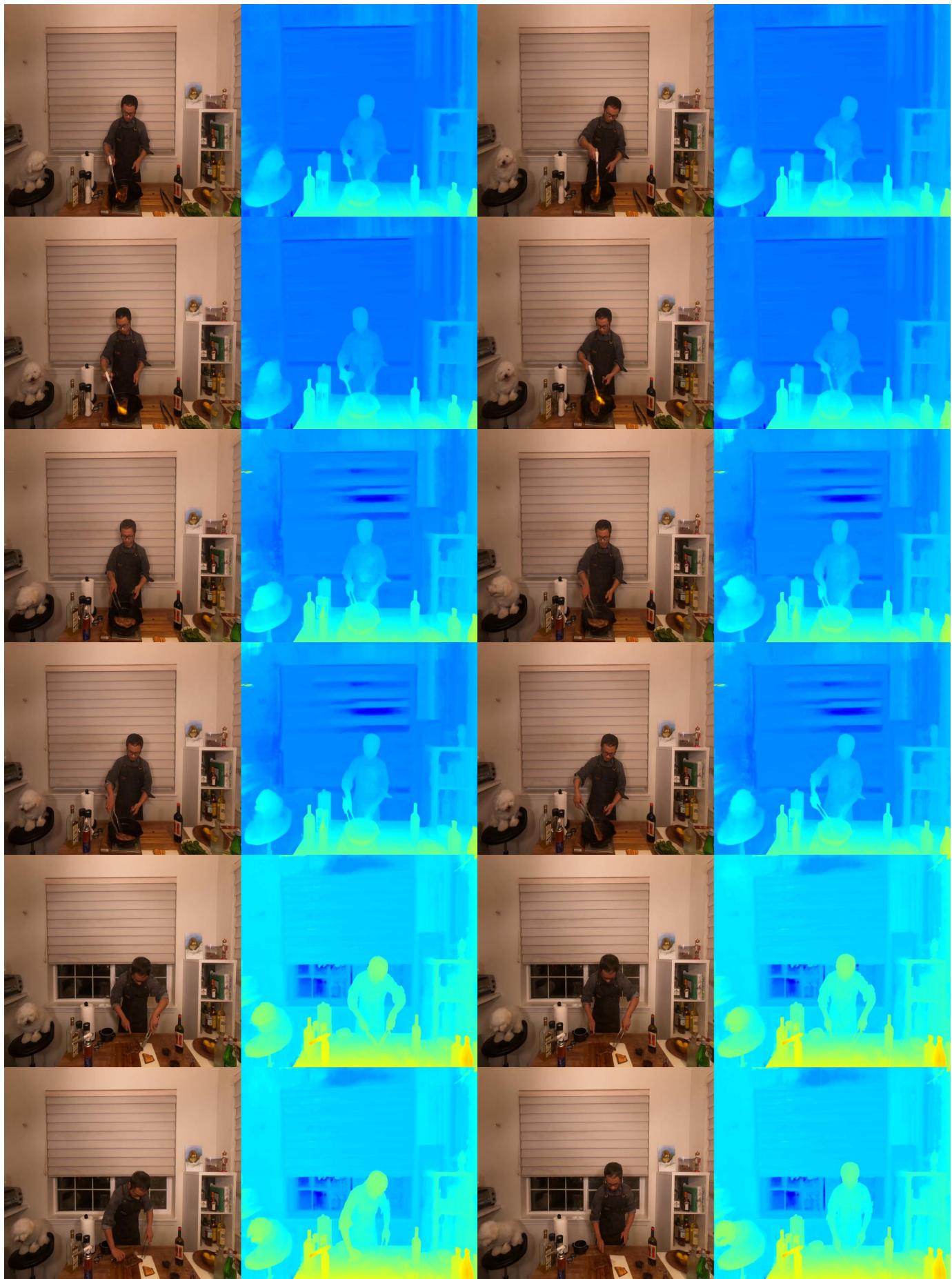


Fig. 13. Visualizations on flame steak, sear steak and cut roasted beef scenes.

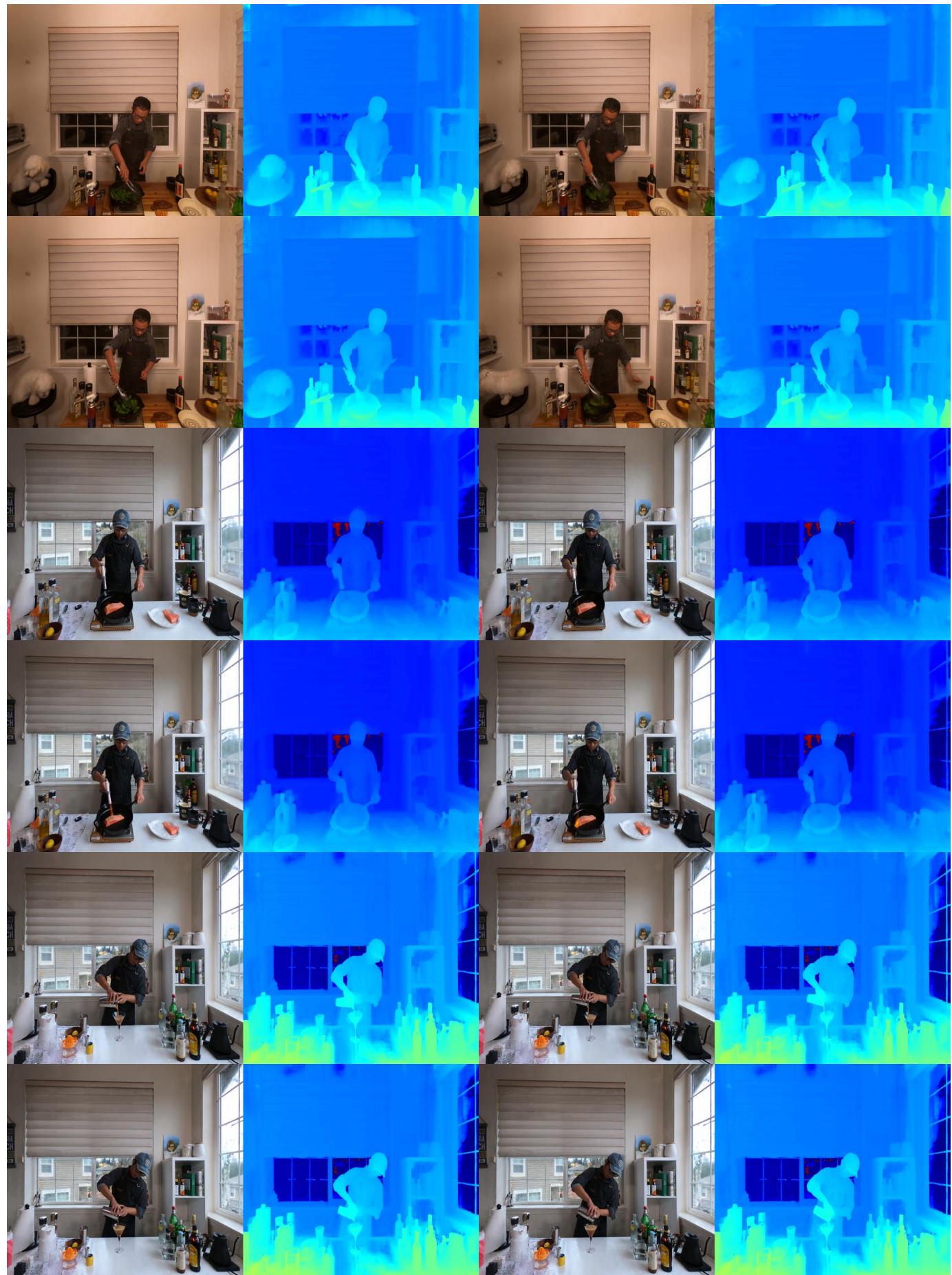


Fig. 14. Visualizations on cook spinach, flame salmon and coffee martini scenes.

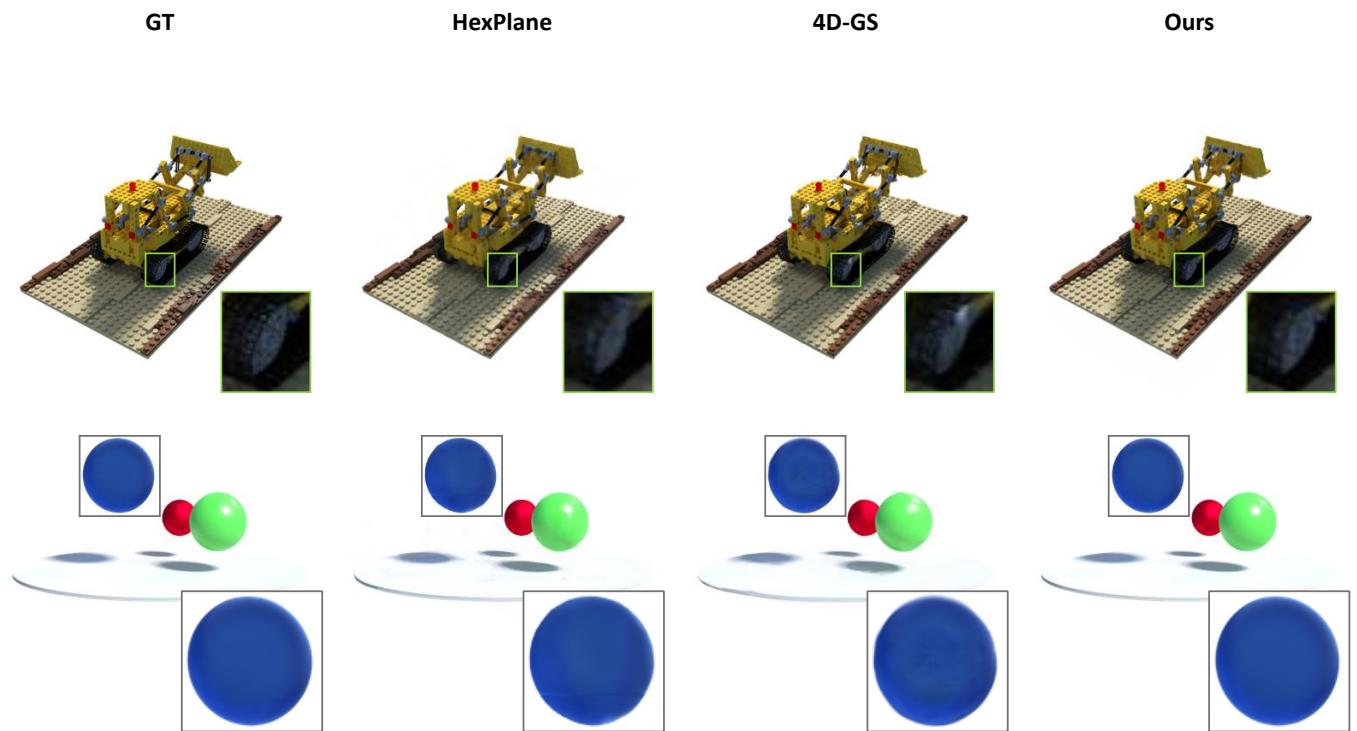


Fig. 15. Visual Comparison on Dynamic Scenes (D-NeRF Data). 4D-GS and HexPlane are decomposition-based and deformation-based methods.



Fig. 16. Visualizations on D-NeRF dataset.



Fig. 17. Visualizations on failure cases from D-NeRF dataset.

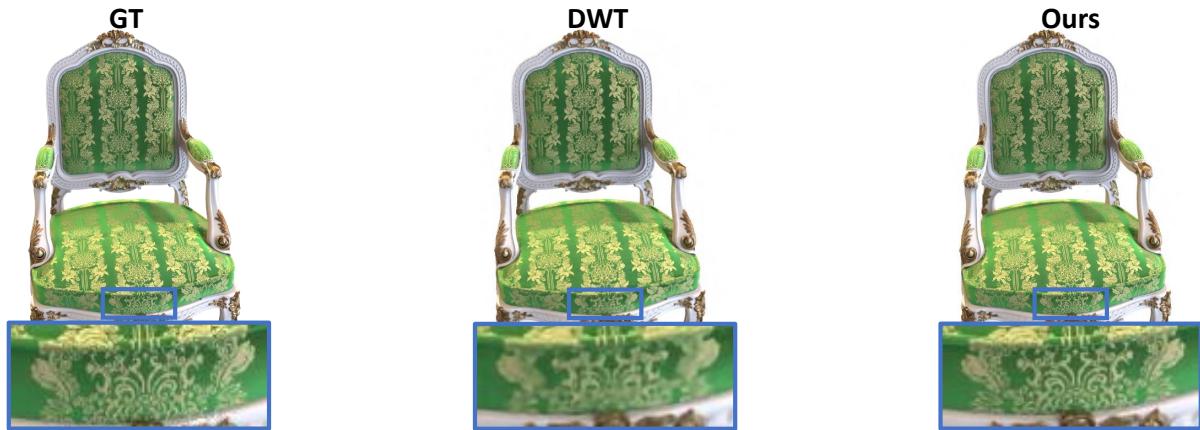


Fig. 18. Visual comparison on NeRF synthetic dataset.



Fig. 19. Visualizations on NeRF synthetic dataset.

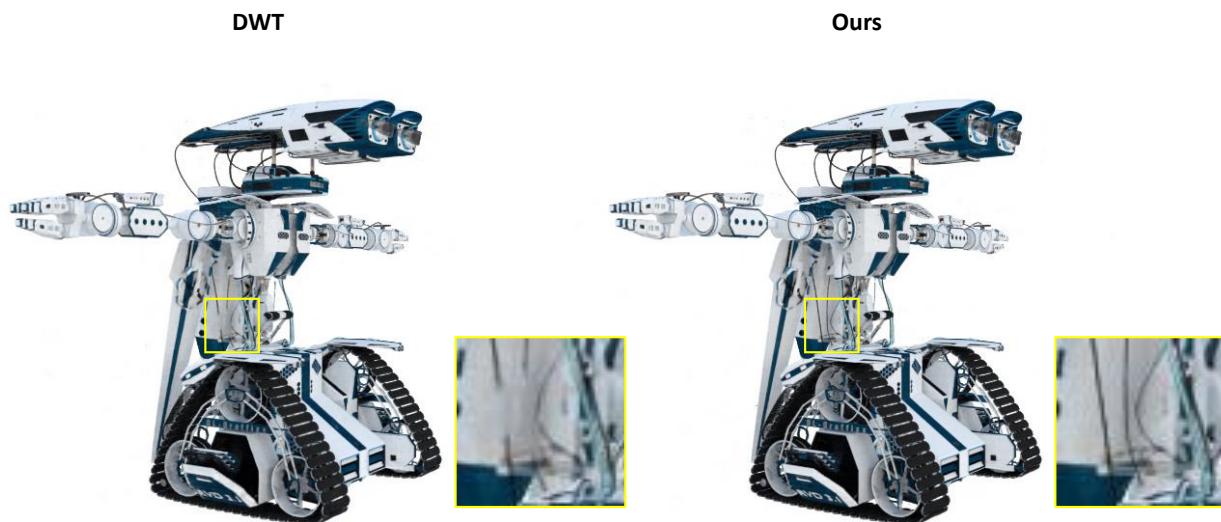


Fig. 20. Visual comparison on NSVF synthetic dataset.



Fig. 21. Visualizations on NSVF synthetic dataset.



Fig. 22. Visual comparison on LLFF synthetic dataset.

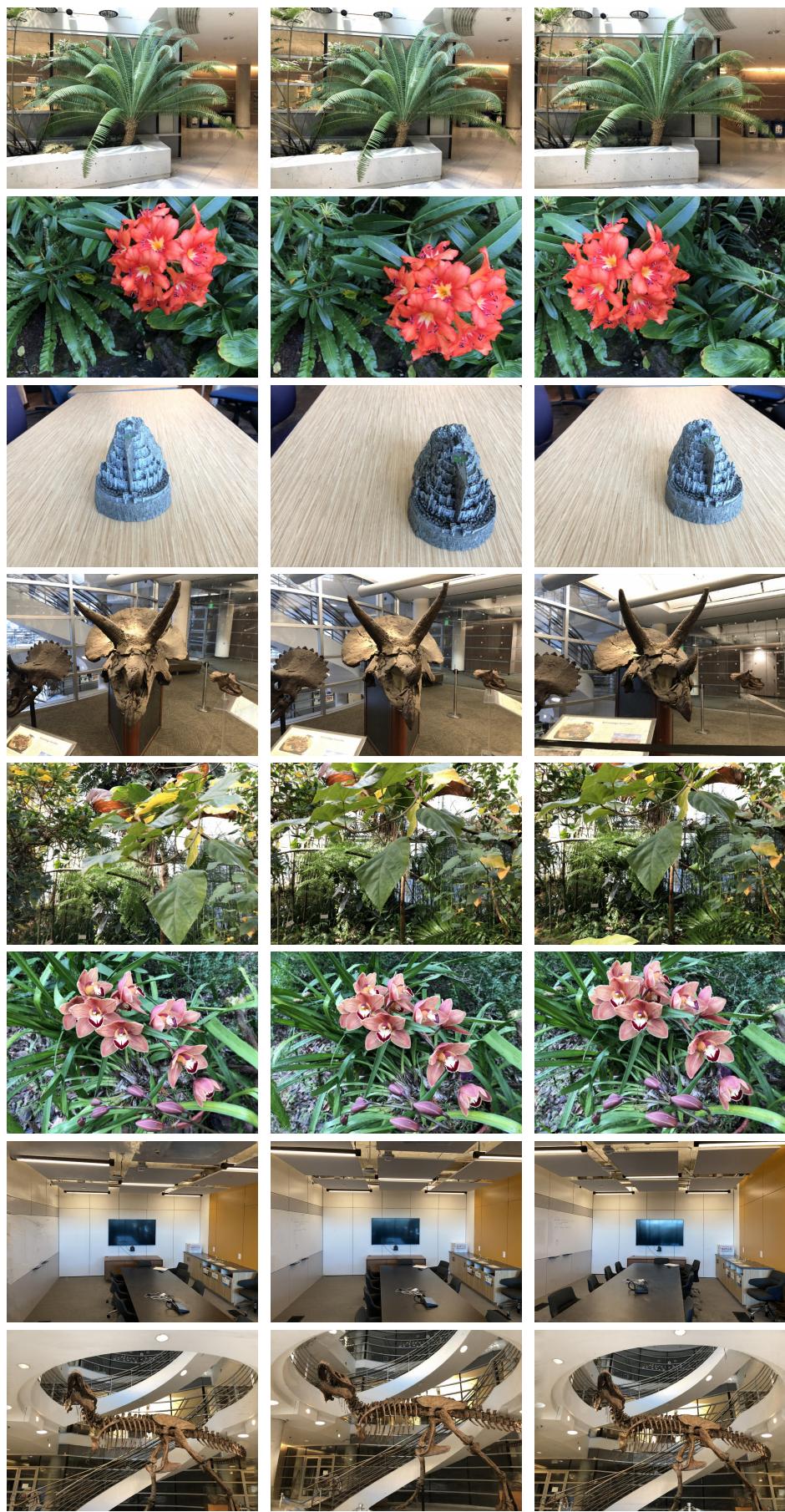


Fig. 23. Visualizations on LLFF synthetic dataset.