DLGStream: Dynamic Language-embedded Guassian Splatting for Open-vocabulary Enabled Free-viewpoint Video Streaming

Zhihui Ke, Yuyang Liu, Xiaobo Zhou*, and Tie Qiu

School of Computer Science and Technology, Tianjin University, China
{kezhihui,yvyang,xiaobo.zhou,qiutie}@tju.edu.cn

Abstract. 3D Gaussian Splatting (3DGS) has emerged as a promising paradigm for reconstructing streamable free-viewpoint video (FVV) from multi-view videos. However, 3DGS-based FVVs typically lack user interaction and editing capabilities, which diminishes the immersive experience. Recent research has integrated language features from CLIP into 3DGS via distillation, enabling open-vocabulary queries and supporting many downstream applications. Nevertheless, the stringent requirements of FVV, low frame size and high FPS, make current language Gaussian representations unsuitable for language-embedded FVV. In this paper, we propose DLGStream, a novel language-embedded FVV representation that streams time-varying language features alongside Gaussian attributes to support 4D environment interaction, scene editing, and spatial intelligence. Specifically, we propose a dual-opacity dynamic language Gaussian representation, which maintains two opacity attributes for color and language features to deal with performance degradation that occurs when colors and features are jointly optimized. Furthermore, we introduce an interpolation-based deformation field to reduce temporal redundancy. This deformation field can also be used for 4D frame interpolation, boosting FVV sequences from low to high FPS. Experimental results demonstrate that DLGStream achieves superior performance in both on open-vocabulary segmentation and reconstruction quality with an average frame size of merely 43 KB. The code is available on https://github.com/kkkzh/DLGStream.

Keywords: Language Guassian Splatting · Dynamic Scene Reconstruction · Free-viewpoint Video

1 Introduction

The groundbreaking development of 3D Gaussian Splatting (3DGS) [29] for static scene reconstruction has prompted the creation of 3DGS-based Free-Viewpoint Video (FVV) methods. These methods extend 3DGS to dynamic scenes and support frame streaming just like traditional video. Due to their superior reconstruction quality and FPS, 3DGS-based FVV methods have rapidly

* Corresponding author.

supplanted those based on Neural Radiance Field (NeRF). However, these FVV representations are fundamentally composed of neural network parameters, which precludes users from performing tasks such as 3D scene editing, environment interaction, and so on.

Recently, LangSplat [62] uses Segment Anything Model (SAM) [32] to segment large, medium, and small objects in multi-view images and extracts their language features from CLIP [63]. Then, these multi-level language features are treated as pseudo ground-truth and distilled into the original 3DGS representation by incorporating language feature attributes. Through language-embedded Gaussian splatting, open-vocabulary queries can be achieved and naturally support various downstream tasks, including 4D environment interaction, scene editing, and embodied and spatial intelligence.

Based on LangSplat, subsequent research has further enhanced query accuracy and expanded its application scope. For example, 4DLangSplat [40] introduces language feature distillation into a dynamic 3D Gaussian representation to enable time-sensitive queries. 4DLangSplat adheres to existing training procedures for language Gaussian representation, resulting in three separate language-embedded dynamic 3D Gaussian models, one for each level of language features. The training process first fixes the parameters of the dynamic 3D Gaussians, and then distills language features. However, this scheme is ill-suited for language-embedded FVV. The aims of FVV reconstruction are high reconstruction quality and low frame size, yet 4DLangSplat have to transmit three distinct models triples the frame size. Moreover, executing open-vocabulary queries necessitates multiple rendering passes to obtain the multi-level language features, substantially decreasing FPS. The separate training of language features also prolongs the total training time. Crucially, when we try to jointly train language feature and color, the reconstruction performance degrades [20], especially for color, which is unacceptable for FVV.

To address the aforementioned challenges of language-embedded FVV, we propose DLGStream, a novel language-embedded FVV representation that enables many downstream tasks through streaming time-related language features while preserving the high reconstruction quality and low frame size essential for FVV. To solve the performance degradation when jointly training language features and color, we propose a dual-opacity dynamic language Gaussian representation. This is based on a key insight that color and language feature have different physical properties. While attributes such as position, rotation, and scaling solely define the geometric distribution of the scene. Opacity, however, is integrally involved in the rendering of both color and language feature. Consequently, the discrepancy between color and language features drives opacity optimization in diverging directions. Thus, we use two separate opacity attributes to mitigate the interference between language features and color.

Furthermore, existing FVV representations learn independent temporal features or Gaussian residuals per frame, which results in significant temporal redundancy. This redundancy is further exacerbated by the incorporation of language features. In this paper, we propose a novel deformation field based on

time feature interpolation, inspired by video interpolation. The interpolation-based deformation field maintains key time features at fixed intervals and derives non-key time features via interpolation. This strategy substantially reduces temporal redundancy and frame size. The interpolated time features are then fed into attribute-specific MLPs to predict Gaussian attribute offsets. By modeling continuous temporal changes through interpolation, our deformation field enables 4D frame interpolation for FVV, effectively enhancing a model trained at a low FPS to a higher one. This low FPS training also reduces the cost of extracting language feature labels. Moreover, our proposed interpolation-based deformation field is decoupled from other Gaussian attributes, making it applicable to a wide range of Gaussian-based representations. Finally, we design a GOP-by-GOP training strategy to address background flickering across different GOPs. The contributions are summarized as follows:

- We propose a novel language-embedded FVV representation, DLGStream, for real-time FVV streaming that supports various downstream tasks via 4D open vocabulary queries.
- We introduce a dual-opacity dynamic language Gaussian representation to solve the performance degradation during the joint optimization of language features and color.
- We develop an interpolation-based deformation field to reduce temporal redundancy and frame size, which also supports 4D frame interpolation.
- Our method is applicable to various Gaussian representations. Experiments conducted on 3DGS and Scaffold-GS demonstrate that our method achieves a 9% mIOU improvement on 4D open-vocabulary query tasks while maintaining state-of-the-art (SoTA) reconstruction quality, with an average frame size of only 43KB.

2 Related Work

2.1 3DGS-based Dynamic Scene Reconstruction

NeRF-based dynamic scene reconstruction methods [1,3,7,11,12,14,18,38,39,42, 43,47,48,55,57–61,64,67,71,73,75–77,81,91,95,96,102,105] struggle to achieve real-time performance even on high-performance GPUs. Benefiting from fast rendering performance of 3DGS [29], dynamic 3DGS has rapidly emerged and been applied to dynamic scene reconstruction. Four primary categories exist: (1) Deformable 3DGS [2, 5, 9, 23, 27, 31, 33, 41, 44, 45, 49, 51–53, 65, 74, 83, 87–89, 92, 93, 99, 101, 109, 113], which constructs canonical 3D Gaussians and employs a deformation field (e.g., MLPs, HexPlanes, or Hash Grids) to predict attribute residuals for each canonical Gaussian based on its position and timestamps. (2) 4D Gaussian representation [8, 98, 103], which incorporates a temporal dimension into 3D Gaussian attributes to create 4D Gaussian primitives. (3) Other foundation representations [35,37,84,101]. (4) Sparse dynamic scene reconstruction [86]. However, these methods utilize a single model to represent an entire dynamic scene, making them unsuitable for streaming applications.

2.2 Free-viewpoint Video Reconstruction

Implicit neural representations become a new paradigm for reconstructing FVV from multi-view videos. StreamRF [38] proposes a frame-by-frame training strategy, where an initial NeRF model is trained using the first frame, following by incremental training of sequential model residuals based on previous NeRF model. This method enables the streaming of model residuals rather than the entire NeRF model. Subsequent works, such as ReRF [79], VideoRF [80], HPC [111], [108], FSVFG [100], and VRVVC [22] further developed this training strategy, primarily to reduce the storage requirements. With the advent of 3DGS, methods like 3DGStream [68], HiCoM [15], QUEEN [17], OR2 [104], iFVC [70], and 4DGC [21] extended the frame-by-frame training strategy to 3DGS for FVV reconstruction. 3DGStream [68], HiCoM [15], OR2 [104] and 4DGC [21] employ deformation fields to predict position and rotation residuals and deform previous Gaussians to the current timestamp, subsequently streaming deformation fields. IGS [94] utilize a feed-forward model to predict position and rotation residuals. However, since only position and rotation attributes are varied per frame, they struggle to handle non-rigid deformation and the appearance of new objects in the following frames. In contrast, QUEEN [17], V^3 [82], H3D-DGS [19], and iFVC [70] learn all Gaussian attribute residuals but suffer from high frame size. Moreover, frame-by-frame training strategy inevitably leads to a continuous decline in performance due to accumulation errors. To mitigate this problem, TeTriRF [90], V^3 [82], GIFStream [36], 4DGCPro [110], and StreamSTGS [28] divide a video into multiple GOPs for training within each GOP to limit error accumulation. However, these methods require learning the corresponding temporal features per frame, resulting in significant temporal redundancy. Moreover, the above FVV representations do not support environment interaction, which diminishes the user's immersive experience. DLGStream distills language features into FVV to support many download tasks.

2.3 3D Language Gaussian Field

Current 3D language Gaussian representations [25, 26, 62, 72, 112] distill features from 2D foundation model into 3D Gaussians to enable 3D open-vocabulary queries. These methods typically use SAM to obtain objects, parts, and subparts within multi-view images, and then extract multi-level language features from CLIP [63]. Subsequently, they train separate language Gaussian fields for each feature level. 4DLangSplat [40] adopts this training method and further extends language Gaussian field into dynamic scene, thereby supporting time-related 4D open-vocabulary queries. However, 4DLangSplat is not directly suitable for language-embedded FVV, due to it greatly reduces FPS, increases frame size, and adversely affects reconstruction quality and training time. In contrast, DLGStream successfully overcomes these challenges through dual-opacity language Gaussian representation and interpolation-based deformation field.

2.4 3D Gaussian Compression

The 3DGS Compression methods comprises two main categories: (1) Constructing sparse 3DGS representation [10, 34, 54, 56, 69, 107] through pruning less important Gaussians, aided by attribute quantization and codebook compression. (2) Employing rate-distortion optimized entropy encoding models to encode the probability distributions of the features and attributes end-to-end during training [6,16,46,78,85,106]. These studies are design for compressing static 3D Gaussians. DLGStream's compression of canonical Gaussian attribute is independent of its time feature compression, allowing canonical Gaussians to be compressed using different methods. This decoupled design means that advancements in static Gaussian compression can be integrated into DLGStream framework.

3 Method

Given multi-view videos, we split them into multiple GOPs to address the performance degradation from accumulated errors inherent in frame-by-frame training. Typically, real-world dynamic scenes captured by multiple cameras contain many static elements. Deforming these static elements is computationally inefficient and resource-intensive. Thus, we perform a static-dynamic 3D Gaussian decomposition, following Swift4D [89] (More details are provided in Appendix). This decomposition yields a static 3D Gaussian representation $\mathcal{G}^s$ for static elements, ensuring that only the dynamic 3D Gaussians $\mathcal{G}^d$ learn deformation. Then, we propose a dual-opacity language Gaussian representation in Section 3.1 for both static and dynamic Gaussians to facilitate the learning of language features for downstream tasks as shown in Fig. 1. To reduce the temporal redundancy of existing FVV representations, we introduce an interpolation-based deformation field in Section 3.2, which maintains only key time features for dynamic Gaussians while non-key time features are obtained via interpolation. Finally, we design a GOP-by-GOP training and compression strategy in Section 3.3 to further reduces frame size and eliminate inter-GOP flickering.

3.1 Dual-Opacity Language Gaussian Splatting

4DLangSplat [40] constructs separate models for multi-level language features, which significantly impacts training time, frame size, and FPS, thereby degrading FVV performance. To address these limitations, we constructed a unified language field. Specifically, we employ 9-dimensional Gaussian language features $\mathcal{F}_l^s \in \mathbb{R}^9$ and $\mathcal{F}_l^d \in \mathbb{R}^9$ to learn all three language feature levels. The feature vector is partitioned such that the first, middle, and final three dimensions correspond to the three distinct language levels, respectively. This design needs only a single language Gaussian splatting operation, bypassing the need for three separate rendering processes and consequently achieving a higher FPS.

To reduce training time, we initially attempted to optimize language features jointly with color reconstruction. However, the physical attribute differences between colors and language features cause shared attributes to be optimized in

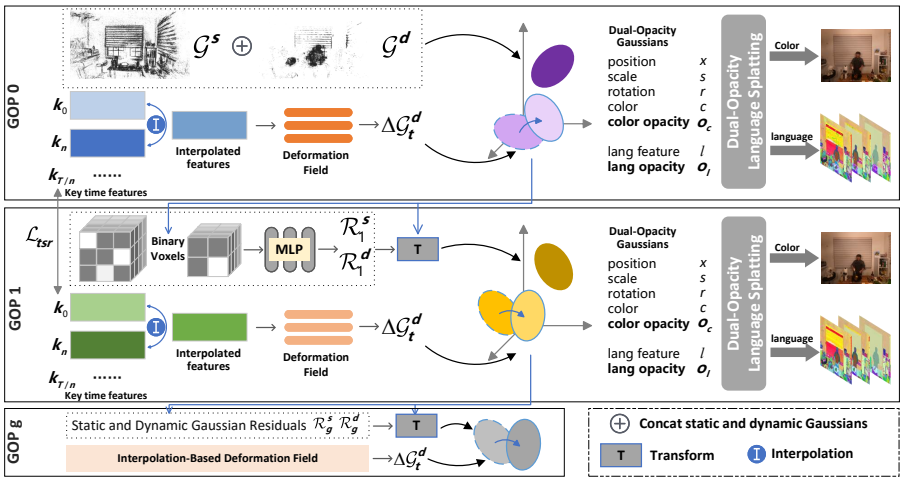


Fig. 1: Overview of the DLGStream framework. We adopt a GOP-by-GOP training strategy. In the first GOP, static and dynamic Gaussians are decoupled, while in subsequent GOPs, the residuals of these Gaussians are learned from the previous GOP. Besides, we interpolate key time features and use a deformation field to predict deformed dynamic Gaussian attributes. Finally, dual-opacity language Gaussian splatting is applied to obtain color and language features in one rendering.

different directions, leading to performance degradation both in open-vocabulary query accuracy and color rendering quality. Considering that position, rotation, and scaling only define the geometric distribution of the scene. However, opacity plays a decisive role in the rendering of color and linguistic features. Therefore, we learn two independent opacity attributes, $o^c \in \mathcal{O}^c$ for color and $o^l \in \mathcal{O}^l$ for language features, which decouples the color and language feature computations during the alpha-blending stage of splatting, as defined below:

$$R_c(x) = \sum_{m \in N} c_m o_m^c \prod_{j=1}^{m-1} (1 - o_j^c), \tag{1}$$

$$R_l(x) = \sum_{m \in N} f_m o_m^l \prod_{j=1}^{m-1} (1 - o_j^l), \tag{2}$$

where R_c and R_l is rendered color and language feature images. c_m and f_m represent the color and language feature of Gaussian m, respectively. To prevent surface localization mismatches between color and language features, we use the color transparency accumulation truncation for the language feature computation. Moreover, the color and language features are optimized with separate gradients during backpropagation.

The language feature loss can be expressed as:

$$\mathcal{L}_l = L1(R_l, concate(I_l^{small}, I_l^{mid}, I_l^{large})) \tag{3}$$

where *concate* is a tensor concatenation operation where we concatenate language feature labels of three levels together.

After training, we follow 4D open-vocabulary query method of 4DLangSplat, which first renders multi-level language feature images for each frames and then calculates the relevance score [30] between these features images and a given text query to generate a segmentation mask for each frame.

3.2 Interpolation-based Deformation Field

Existing FVV reconstruction methods require learning corresponding time features for every frame. The incorporation of additional language feature attributes further escalates the frame size required per frame. Considering that objects in real-world videos exhibit continuous motion, video interpolation [24] utilizes this characteristic to reduce temporal redundancy. Inspired by video interpolation, we interpolate time features, thereby eliminating the need to learn time features for every individual frame and substantially reducing frame size.

Our interpolation-based deformation field maintains several key time features $\mathcal{K} = \{k_0, k_1, \cdots, k_{T/n}\}$, where T represents the total number of frames in a GOP and n denotes the interval. Given timestamp t, adjacent key time features k_i and k_{i+1} are obtained using the index $i = \lfloor \frac{t}{n} \rfloor$. Then, we utilize linear interpolator to obtain non-key time feature $k_t = Lerp(k_i, k_{i+1}, t)$. For each dynamic Gaussian attribute $a^d \in \mathcal{G}^d$[1], we utilize Gaussian attribute predictors $\mathcal{D}_a$ to predict corresponding attribute offset $\Delta a_t^d \in \Delta\mathcal{G}_t^d$:

$$\Delta a_T^d = \mathcal{D}_a(k_t), \quad a_t^d = a^d + \Delta a_t^d. \tag{4}$$

As a result, our interpolation-based deformation field significantly reduces storage requirements by storing T/n key temporal features instead of the full T.

To enable real-time streaming, an effective time feature compression method is essential. Although several compression frameworks have been developed for encoding static 3D Gaussians, real-time streaming requires the consideration of both frame size and decoding latency. Recent context-aware entropy encoding methods (e.g., HAC [6]) offer remarkable compression efficiency, but their long decoding latency makes them unsuitable for this application. Thus, we propose to utilize video codec to encode time features within GOPs.

We organize the time features $\mathcal{K}$ into a sequence of 2D images and encode them as a feature video using video codecs (e.g., H.264 and HEVC). These video codecs are highly optimized, facilitating real-time encoding and decoding on a wide range of devices. The frame size of the feature video is highly determined by the similarity between adjacent time features. Because of the temporal continuity of object motion in real-world videos, time features should also have similar continuity. To enforce this, we introduce temporal regularization $\mathcal{L}_{tsr}$ as follows:

$$\mathcal{L}_{tsr} = L1(k_i, k_{i+1}),\ i = \lfloor t/n \rfloor . \tag{5}$$

[1] For 3DGS, $\mathcal{G}$={Position, Scale, Rotation, Opacity, and Color}, while for Scaffold-GS, $\mathcal{G}$={Position, Scale, Feature, Offset}.

To balance storage and performance, the GOP size is set to 60, and the key time feature interval is set to $n = 10$. Moreover, time features are encoded as a video using libx265 with a quality parameter $crf = 6$.

3.3 GOP-by-GOP training and Compression

We observe that previous GOP-based FVV training methods [28, 36, 82, 90] exhibit noticeable flickering across different GOPs. This artifact arises because the Gaussian color parameters learned by these methods represent the average color within a single GOP. Since these average values differ across GOPs and no temporal correlation constraints are enforced between them, inconsistent brightness becomes apparent. To solve this problem, we leverage the Gaussian attributes of the previous GOP $g-1$ as initialization rather than training from scratch. We optimize Gaussian attribute residuals of the static and dynamic Gaussians from the previous GOP $g-1$:

$$\mathcal{G}_g^s = \mathcal{G}_{g-1}^s + \mathcal{R}_g^s, \quad \mathcal{G}_g^d = \mathcal{G}_{g-1}^d + \mathcal{R}_g^d, \tag{6}$$

where $\mathcal{R}_g^s$ and $\mathcal{R}_g^s$ are predicted residuals for each attribute. The interpolation-based deformation field requires training from initialization. This GOP-by-GOP training strategy stores not the complete set of Gaussian attributes, but only their residuals, which further reduces the frame size. The core challenge, therefore, becomes the effective compression of these residual values.

The residuals of Gaussian attributes are sensitive to numerical precision and quantization operation that can significantly degrade reconstruction quality. Motivated by BiRF [66] that utilizes binary voxels with a tiny MLP to model static scene. The parameters of these binary voxels are constrained to values of either $+1$ or -1, rather than floating-point numbers, which achieves a substantial reduction in storage. Hence, we propose employing binary voxels to model the Gaussian attribute residuals for both static and dynamic canonical 3D Gaussians. Specifically, for a Gaussian at position $\mathcal{X}$, we obtain spatial feature f_g from multi-level binary voxels $\mathcal{V}$:

$$f_g = Sigmoid(interp(\mathcal{X}, \mathcal{V})), \tag{7}$$

Then, a tiny MLP is utilized to predict the Gaussian attribute residuals $\mathcal{R}_g^s$ and $\mathcal{R}_g^d$ as follows:

$$\mathcal{R}_g^s, \mathcal{R}_g^d = MLP(f_g). \tag{8}$$

To reduce the size of binary voxels, we follow previous work [6] to estimate bit consumption and minimize it:

$$\mathcal{L}_e = M_+(-log(v_f)) + M_-(-log(1 - v_f)), \tag{9}$$

where, v_f is the occurrence frequency of the parameter $+1$ in binary voxels. M_+ and M_- is the total number of $+1$ and -1, respectively.

Since the uniform number of dynamic Gaussians across GOPs, we introduce an inter-GOP temporal regularization, which enables us to encode temporal features from different GOPs into a unified feature video, thereby further reducing data size.

$$\mathcal{L}_{tsr} = L1(k^g_{T/n}, k^{g-1}_0). \tag{10}$$

Parallel GOP Training. To accelerate training speed, the GOP-by-GOP training strategy can be adapted to facilitate parallel training. Specifically, after the first GOP is trained, subsequent GOPs can be trained based on the first GOP without waiting for the previous GOP to finish training, i.e., $\mathcal{G}^s_g = \mathcal{G}^s_0 + \mathcal{R}^s_g$, $\mathcal{G}^d_g = \mathcal{G}^d_0 + \mathcal{R}^d_g$.

3.4 Implementation

The dual-opacity language Gaussian representation, interpolation-based deformation field, and GOP-by-GOP training strategy are agnostic to the fundamental 3D Gaussians representation and can therefore be integrated into various frameworks. To demonstrate this broad applicability, we implement our method using two prevalent representations: 3DGS [29] and Scaffold-GS [50]. We empoly SOG [24] to compress 3DGS-based static and dynamic Gaussians, and HAC [6] to compress Scaffold-GS based static and dynamic anchors.

Since 3DGS representation differs from Scaffold-GS representation, our dual-opacity language Gaussian implementation is tailored to each. For 3DGS, we augment both static and dynamic Gaussians by adding learnable language feature attributes $\mathcal{F}^s_l \in \mathbb{R}^9$ and $\mathcal{F}^d_l \in \mathbb{R}^9$ alongside their corresponding opacity attributes $\mathcal{O}^s_l$ and $\mathcal{O}^d_l$. Furthermore, the deformation field is extended with a tiny MLP to predict residuals for these language features. As for Scaffold-GS, we simply incorporate an additional attribute decoding MLP that takes anchor features as input and predicts Gaussian language features of dimension $[N \times k, 9]$. Simultaneously, the opacity decoding MLP is modified to output $[N \times k, 2]$ opacity attributes.

The implementation of the interpolation-based deformation field is the same for both 3DGS and Scaffold-GS. It is worth noting that deformation is not applied to the dynamic Gaussian attributes decoded from Scaffold-GS, but rather directly to the positions, features, scales, and offsets of the dynamic anchors . We have provided more details in the Appendix.

Loss. When training the first GOP, the total loss is $\mathcal{L} = \alpha\mathcal{L}_{rgb} + (1-\alpha)\mathcal{L}_{ssim} + \beta\mathcal{L}_{tsr} + \psi\mathcal{L}_l$. For subsequent GOPs, the binary regularization term $\mathcal{L}_e$ needs to be incorporated:

$$\mathcal{L} = \alpha\mathcal{L}_{rgb} + (1-\alpha)\mathcal{L}_{ssim} + \beta\mathcal{L}_{tsr} + \psi\mathcal{L}_l + \gamma\mathcal{L}_e, \tag{11}$$

where α, β, ψ, and γ are hyper-parameters. We have not shown the regularization terms introduced by SOG or HAC. They are only used to train the first GOP and are not needed for training subsequent GOPs.

Table 1: mIoU(%)/mAcc(%) comparison of 4D open-vocabulary querying on the N3DV dataset.

Method	Coffee Martini	Cook Spinach	Cut Beef	Flame Salmon	Flame Steak	Sear Steak	Average
LangSplat [62]	68.0/**98.5**	78.3/**98.6**	36.5/97.0	66.0/82.2	64.1/97.8	78.3/98.6	61.49/91.9
4DLangSplat [40]	76.0/80.0	77.4/62.7	67.4/100	84.5/100	74.9/100	72.9/100	75.5/90.4
Ours(3DGS)	**87.8/80.0**	**88.5/97.3**	85.0/100	**86.8/100**	**87.0/100**	**81.7/100**	**85.8/96.2**
Ours(ScaffoldGS)	87.4/80.0	88.3/96.0	**86.3/100**	84.6/100	84.3/100	77.3/100	84.7/ 96.0

Table 2: Quantitative comparison of joint optimizing color and language features on N3DV dataset.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	Storage (KB) ↓	FPS ↑
4DLangSplat [40]	30.99	0.9351	0.1571	431	15
Ours(3DGS)	**32.15**	0.9442	**0.1413**	97.11	**83**
Ours(ScaffoldGS)	32.04	**0.9446**	0.1419	**46.33**	48

4 Experiment

4.1 Experimental Setup

Datasets. We evaluate our method on three multi-view dynamic scene datasets: (1) **N3DV** [39] dataset have six scenes featuring large motions, dynamic illuminations, and specular highlights. These multi-view videos were captured by 18 to 21 cameras at a resolution of 2704×2028 with 30 FPS. Following prior work, we downsample the resolution to 1352×1014. (2) **MeetRoom** [38] dataset, which includes three scenes captured by 12 cameras at a resolution of 1280×720 and a frame rate of 30. (3) **WideRange4D** [97] dataset is different from the two previous forward-facing datasets, as it provides a 360-degree capture of dynamic scenes with less overlap between camera views, thereby establishing a new evaluation benchmark. We selected two outdoor scenes, each comprising over 240 frames. For each scene, 60 camera views were captured, of which 59 were used for training and one for testing.

Baseline. For 4D open-vocabulary querying task, we select 4DLangSplat [40] and LangSplat [62] as baselines for evaluation. As for FVV reconstruction task, we use two different 3D Gaussian representations (i.e., 3DGS and Scaffold-GS), so we also divided the current benchmark into two categories. 3DGS-based FVV representations include TeTriRF [90], 3DGStream [68], HiCoM [15], QUEEN [17], OR2 [104], 4DGC [21], and StreamSTGS [28]; Scaffold-GS-based FVV representations include GIFStream [36] and iFVC [70]. Considering that some methods utilize de-distorted images for training, while others do not. We follow the dataset preprocessing pipeline established by 4DGaussian [87] and re-trained several benchmarks using this pipeline to ensure a fair comparison.

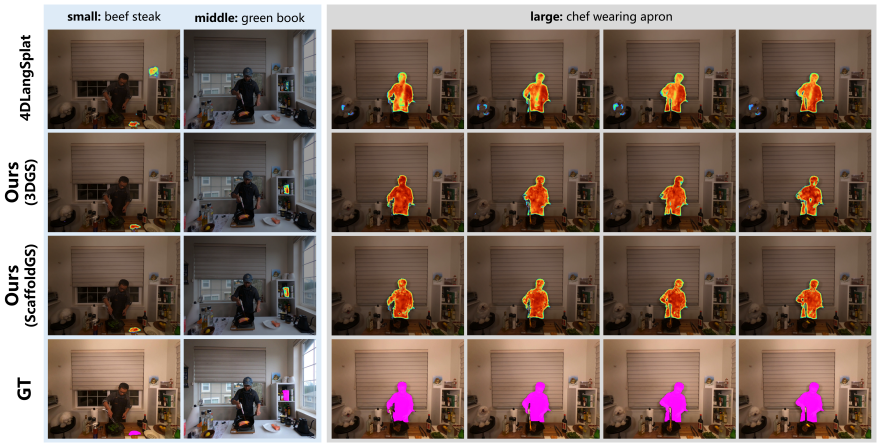


Fig. 2: Qualitative comparisons of 4D open-vocabulary querying.

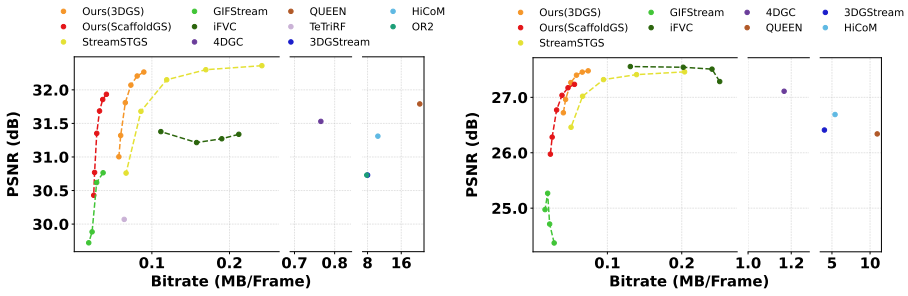
4.2 4D Open-vocabulary Querying Results

Quantitative Comparisons . We perform 4D open-vocabulary querying on N3DV dataset, with mIoU(%) and mAcc(%) results detailed in Table 1. Compared to 4DLangSplat, our 3DGS-based DLGStream improves mIoU by 10.3%, and Scaffold-GS-based DLGStream achieves a 9.2% improvement. As for mAcc metric, our DLGStream achieves a 6% improvement. As shown in Table 2, our method reduces the frame size by 10X and increases the FPS by 5X compared to 4DLangSplat. These enhancements primarily stem from our proposed dual-opacity language Gaussian representation, which effectively eliminates the interference between color and language features during joint training. Moreover, this representation enables the simultaneous learning of multi-level language features, thereby acquiring all feature maps in a single rendering pass.

Qualitative Comparisons . Fig. 2 shows the segmentation results for the 4D open-vocabulary querying task. We queried objects of small, medium and large sizes across different dynamic scenarios. According to the qualitative results in Fig. 2, our method accurately segments the target objects, whereas 4DLangSplat either identifies non-target objects or fails to segment the target completely. More quantitative and qualitative comparison results are provided in the **Appendix**.

4.3 FVV Reconstruction Results

As presented results in Table 3, our method achieves superior FVV reconstruction performance, requiring only 92KB and 42KB for 3DGS-based and Scaffold-GS-based representations, respectively. This performance is primarily attributed to our interpolation-based deformation field, which effectively reduces temporal redundancy, and GOP-by-GOP training strategy further reduces the redundancy of Gaussian spatial attributes. Although GIFStream has a smaller frame size



(a) N3DV **(b)** MeetRoom

Fig. 3: RD-Cave results of N3DV and MeetRoom datasets.

Table 3: Quantitative comparison of FVV reconstruction on N3DV dataset.

Method	N3DV					MeetRoom				
	PSNR ↑	SSIM ↑	LPIPS ↓	Storage ↓	FPS ↑	PSNR ↑	SSIM ↑	LPIPS ↓	Storage ↓	FPS ↑
TeTriRF [90]	30.07	0.9003	0.2986	65.89	1.53	-	-	-	-	-
3DGS-based representations										
3DGStream [68]	30.73	0.9348	0.1470	8205	72	26.41	0.8990	0.2370	4108	121.44
HiCoM [15]	31.31	0.9390	0.1475	10704	163	26.69	0.9035	0.2315	5535	275
QUEEN [17]	31.79	0.9449	0.1415	700	251	26.34	0.9185	0.2164	11201	497
4DGC [21]	31.53	0.9412	0.1431	784	79	27.11	0.9093	0.2306	1195	110
OR2 [104]	30.73	0.9336	0.1577	8005	134	-	-	-	-	-
4DGCPro [110]	31.64	0.9440	-	655.4	-	-	-	-	-	-
Motion Matters [4]	32.12	0.9450	0.1290	109.6	-	-	-	-	-	-
ReCon-GS [13]	31.89	0.9450	0.1410	450.6	-	-	-	-	-	-
StreamSTGS [28]	32.30	0.9436	0.1474	174	100	27.41	0.9181	0.2157	143	100
Ours(3DGS)	32.26	0.9452	0.1388	92.04	107	27.48	0.9179	0.2080	75.92	166
Scaffold-GS-based representations										
GIFStream [36]	30.76	0.9368	0.1470	38.04	97	24.37	0.8807	0.2387	29.11	126
iFVC [70]	31.38	0.9419	0.1357	114	14.65	27.29	0.9120	0.2036	170	17.90
Ours(ScaffoldGS)	31.93	0.9448	0.1408	42.42	88.27	27.28	0.9172	0.2059	49.31	123

Table 4: Quantitative comparison on WideRange4D dataset.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	Storage (KB) ↓	FPS ↑
iFVC [70]	27.06	0.7960	0.2647	263.80	37.10
Ours	27.58	0.7975	0.2814	226.66	36.68

than our method, the rate-distortion (RD) curves in Fig. 3 demonstrate that our method achieves superior quality at similar bitrates. Moreover, GIFStream and iFVC use entropy encoding loss to control frame size, but it is not stable. Our method controls frame size through video codecs, resulting in better RD performance. We also evaluated our method on an outdoor dataset WideRange4D with 360-degree captures. As shown in Table 4, our method maintains SoTA FVV reconstruction performance. More results are provided in the **Appendix**.

Table 5: Quantitative results of frame interpolation on N3DV dataset. We skip frames during training but evaluate all frames to simulate frame interpolation.

Method	Skipped frames	mIoU↑	mAcc↑	PSNR↑	SSIM↑	Storage↓
Ours(3DGS)	1 (15 FPS)	83.51	94.63	32.04	0.943	79.19
	2 (10 FPS)	84.72	94.54	32.24	0.944	100.8
	4 (6 FPS)	83.79	95.11	31.62	0.942	92.87
	9 (3 FPS)	82.64	95.58	30.94	0.936	90.01
	0 (30 FPS)	85.78	96.22	32.15	0.944	97.11
Ours(ScaffoldGS)	1 (15 FPS)	85.49	96.67	31.79	0.945	59.81
	2 (10 FPS)	85.29	96.00	31.90	0.946	65.6
	4 (6 FPS)	85.18	93.33	31.80	0.945	64.08
	9 (3 FPS)	82.44	94.74	31.26	0.941	59.32
	0 (30 FPS)	84.69	96.00	32.04	0.945	46.33

4.4 4D Frame Interpolation

Based on our proposed interpolation-based deformation field, we investigate the 4D frame interpolation task. This task is simulated by excluding specific frames during the training. For example, when one frame is skipped, only odd-numbered frames are used for training. As presented in Table 5, our method achieves promising results when one or two frames are interpolated. Even with 9 frames skipped, namely conducting training at a rate of 3 FPS, competitive results were still attained. This outcome indicates that our interpolation-based deformation field effectively captures the temporal correlations. As a result, our method can be trained on sequences with a lower frame rate (e.g., 10 FPS), while higher frame rates (e.g., 30 or 60 FPS) can be achieved via interpolation during rendering. This approach also significantly reduces the costs associated with preparing pseudo language labels.

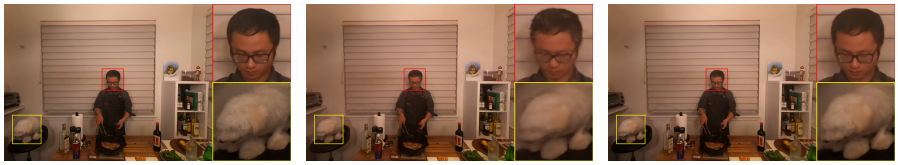
4.5 Ablation Study

Dual-opacity Gaussian representation. We conducted an ablation study by removing the dual-opacity design, thereby sharing a single opacity between color and language features. The results, presented in Table 6, show a significant decrease in mIoU, mAcc, PSNR, and SSIM. Fig. 4 shows noticeable artifacts in the dynamic regions when color and feature share the same opacity. This finding confirms that using separate opacity attributes for color and language features is essential for effective joint optimization.

Quantitative analysis of dual-opacity. Fig. 5(a)-(c) illustrate a distinct distributional discrepancy: feature opacity follows a Gaussian distribution centered at 0.5, whereas color opacity is centered at 0.3. This is because color requires lower opacity to represent the rich texture details of objects, while semantics of an object are consistent and do not exhibit translucency. The dynamic object in Fig. 4(b) appears overly smooth and lacks detailed textures, which confirms that

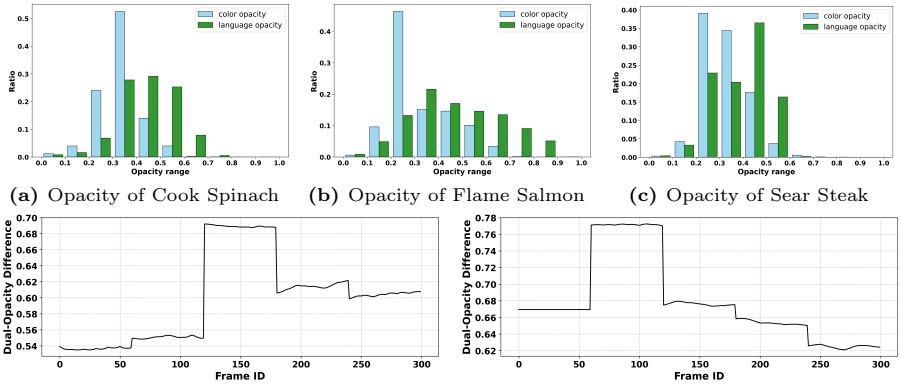
Table 6: Ablation results of dual-opacity Gaussian representation on the N3DV dataset. 'SO' indicates single opacity.

Method	mIoU↑	mAcc↑	PSNR↑	SSIM↑	Storage↓
Ours(3DGS)-SO	78.99	92.54	31.73	0.9423	**96.25**
Ours(3DGS)	**85.78**	**96.22**	**32.15**	**0.9442**	97.11
Ours(ScaffoldGS)-SO	79.62	94.67	31.88	0.9440	46.80
Ours(ScaffoldGS)	**84.69**	**96.00**	**32.04**	**0.9446**	**46.33**



(a) GT **(b)** Single Opacity **(c)** Dual Opacity

Fig. 4: Qualitative comparison of opacity ablation.



(a) Opacity of Cook Spinach **(b)** Opacity of Flame Salmon **(c)** Opacity of Sear Steak

(d) Opacity time difference of Cook Spinach **(e)** Opacity time difference of Flame Salmon

Fig. 5: Opacity statistical distributions of color and language feature.

sharing opacity can interfere with the learning of color and semantics. Fig. 5(d) and (e) further depict the temporal evolution of per-Gaussian opacity differences, revealing substantial variations over time. These observations prove a significant physical distinction between color opacity and language feature opacity.

Details of storage. We present detailed information on the storage in Table 7. From the table, it can be seen that the temporal feature is only 20 to 30 KB, thanks to our proposed interpolation-based deformation field, which greatly reduces time redundancy. In addition, the Gaussian residual R_g^s and R_g^d between GOPs is also very small, only about 2KB. Note that Gaussian attributes still occupy a high proportion, and the advancement of static Gaussian compression methods can further reduce the storage.

Table 7: Details of average frame size (KB) in N3DV dataset.

Method	Attributes	Temporal features	MLPs	R_g^s & R_g^d
Ours(3DGS)	64.03	29.33	0.3854	3.3595
Ours(ScaffoldGS)	22.53	21.18	0.7077	2.5405

Table 8: Ablation of interpolation method in N3DV dataset.

Interpolation method	PSNR↑	SSIM↑	LPIPS↓	Storage↓	Size of Temp. features↓
Nearest	31.35	0.943	0.143	66.48	37.59
Bicubic	32.20	0.946	0.136	80.24	53.12
Bilinear(Ours)	31.93	0.945	0.141	42.42	19.52

Table 9: Ablation of intra-GOP Guassian residual learning.

Configure	PSNR↑	SSIM↑	LPIPS↓	Storage↓	Size of R_g^s&R_g^d↓
$\text{MLP}_{32\times2}$	31.92	0.945	0.141	42.15	2.18
$\text{MLP}_{128\times2}$	31.93	0.945	0.141	42.74	2.85
$\text{Voxel}_{256to2048}$	31.93	0.945	0.141	42.12	2.12
$\text{Voxel}_{1024to8192}$	31.94	0.945	0.141	42.62	2.85
Default/Ours(ScaffoldGS)	31.93	0.945	0.141	42.42	2.54

Ablation of interpolation method. Table 8 shows bilinear interpolation is sufficient for real-world dynamic scenes. The complex bicubic yield a minimal gains, whereas nearest-neighbor performs poorly.

Ablation of Gaussian residual learning. The default voxels setting are [512, 1024, 2048, 4096] alongside 2-layers MLP with 64 neurons per layer. Table 9 demonstrates that scaling the voxel grids or MLP architecture yields only marginal differences in performance.

5 Conclusion

We propose DLGStream, a novel language-embedded streamable FVV representation that supports 4D open-vocabulary querying, 4D video interpolation, and so on. Our dual-opacity dynamic language Gaussian splatting efficiently solve the performance degradation during the joint optimization of color and language features. Moreover, we introduce an interpolation-based deformation field that stores only key time features and interpolates non-key time features, thereby significantly reducing the frame size. Finally, we develop GOP-by-GOP training and compression to further reduce frame size while simultaneously eliminate flickering artifacts across GOPs.

Acknowledgements

This work is supported by the Beijing-Tianjin-Hebei Basic Research Cooperation Project (No. 24JCZXJC00050).

References

1. Attal, B., Huang, J.B., Richardt, C., Zollhoefer, M., Kopf, J., O'Toole, M., Kim, C.: 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)
2. Bae, J., Kim, S., Yun, Y., Lee, H., Bang, G., Uh, Y.: Per-gaussian embedding-based deformation for deformable 3d gaussian splatting. In: European Conference on Computer Vision. pp. 321–335. Springer (2025)
3. Cao, A., Johnson, J.: Hexplane: A fast representation for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 130–141 (2023)
4. Chen, J., Mao, Q., Bao, Y., Meng, X., Meng, F., Wang, R., Liang, Y.: Motion matters: Compact gaussian streaming for free-viewpoint video reconstruction. Advances in Neural Information Processing Systems **38**, 120385–120409 (2026)
5. Chen, J., Hu, Z., Wu, P., Zhu, H., Li, H., Sun, X.: Dash: 4d hash encoding with self-supervised decomposition for real-time dynamic scene rendering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 26349–26359 (2025)
6. Chen, Y., Wu, Q., Lin, W., Harandi, M., Cai, J.: Hac: Hash-grid assisted context for 3d gaussian splatting compression. In: European Conference on Computer Vision. pp. 422–438. Springer (2024)
7. Du, Y., Zhang, Y., Yu, H.X., Tenenbaum, J.B., Wu, J.: Neural radiance flow for 4d view synthesis and video processing. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 14304–14314. IEEE Computer Society (2021)
8. Duan, Y., Wei, F., Dai, Q., He, Y., Chen, W., Chen, B.: 4d-rotor gaussian splatting: towards efficient novel view synthesis for dynamic scenes. In: ACM SIGGRAPH 2024 Conference Papers. pp. 1–11 (2024)
9. Duisterhof, B.P., Mandi, Z., Yao, Y., Liu, J.W., Shou, M.Z., Song, S., Ichnowski, J.: Md-splatting: Learning metric deformation from 4d gaussians in highly deformable scenes. arXiv preprint arXiv:2312.00583 (2023)
10. Fan, Z., Wang, K., Wen, K., Zhu, Z., Xu, D., Wang, Z., et al.: Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps. Advances in neural information processing systems **37**, 140138–140158 (2024)
11. Fang, J., Yi, T., Wang, X., Xie, L., Zhang, X., Liu, W., Nießner, M., Tian, Q.: Fast dynamic radiance fields with time-aware neural voxels. In: SIGGRAPH Asia 2022 Conference Papers. pp. 1–9 (2022)
12. Fridovich-Keil, S., Meanti, G., Warburg, F.R., Recht, B., Kanazawa, A.: 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)
13. Fu, J., Gao, Q., Wen, C., Wu, Y., Ma, S., Zhang, J., Zhang, J.: Recon-gs: Continuum-preserved gaussian streaming for fast and compact reconstruction of dynamic scenes. Advances in Neural Information Processing Systems **38**, 134751–134778 (2026)
14. Gao, C., Saraf, A., Kopf, J., Huang, J.B.: Dynamic view synthesis from dynamic monocular video. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5712–5721 (2021)

15. Gao, Q., Meng, J., Wen, C., Chen, J., Zhang, J.: Hicom: Hierarchical coherent motion for streamable dynamic scene with 3d gaussian splatting. Advances in Neural Information Processing Systems (2024)
16. Girish, S., Gupta, K., Shrivastava, A.: Eagles: Efficient accelerated 3d gaussians with lightweight encodings. In: European Conference on Computer Vision. pp. 54–71. Springer (2024)
17. Girish, S., Li, T., Mazumdar, A., Shrivastava, A., Luebke, D., De Mello, S.: Queen: Quantized efficient encoding of dynamic gaussians for streaming free-viewpoint videos. Advances in Neural Information Processing Systems (2024)
18. Guo, X., Sun, J., Dai, Y., Chen, G., Ye, X., Tan, X., Ding, E., Zhang, Y., Wang, J.: Forward flow for novel view synthesis of dynamic scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16022–16033 (2023)
19. He, B., Chen, Y., Lu, G., Wang, Q., Gu, Q., Xie, R., Song, L., Zhang, W.: H3d-dgs: Exploring heterogeneous 3d motion representation for deformable 3d gaussian splatting. Advances in Neural Information Processing Systems **38**, 114028–114053 (2026)
20. He, J., Li, C., Wang, S., Kwong, S.: Joint semantic and rendering enhancements in 3d gaussian modeling with anisotropic local encoding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 28354–28363 (2025)
21. Hu, Q., Zheng, Z., Zhong, H., Fu, S., Song, L., Zhai, G., Wang, Y., et al.: 4dgc: Rate-aware 4d gaussian compression for efficient streamable free-viewpoint video. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)
22. Hu, Q., Zhong, H., Zheng, Z., Zhang, X., Cheng, Z., Song, L., Zhai, G., Wang, Y.: Vrvvc: Variable-rate nerf-based volumetric video compression. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 3563–3571 (2025)
23. Huang, Y.H., Sun, Y.T., Yang, Z., Lyu, X., Cao, Y.P., Qi, X.: Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4220–4230 (2024)
24. Jain, S., Watson, D., Tabellion, E., Poole, B., Kontkanen, J., et al.: Video interpolation with diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7341–7351 (2024)
25. Jiao, S., Dong, H., Yin, Y., Jie, Z., Qian, Y., Zhao, Y., Shi, H., Wei, Y.: Clip-gs: Unifying vision-language representation with 3d gaussian splatting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4670–4680 (2025)
26. Jun-Seong, K., Kim, G., Yu-Ji, K., Wang, Y.C.F., Choe, J., Oh, T.H.: Dr. splat: Directly referring 3d gaussian splatting via direct language embedding registration. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 14137–14146 (2025)
27. Katsumata, K., Vo, D.M., Nakayama, H.: A compact dynamic 3d gaussian representation for real-time dynamic view synthesis. In: European Conference on Computer Vision. pp. 394–412. Springer (2025)
28. Ke, Z., Liu, Y., Zhou, X., Qiu, T.: Streamstgs: Streaming spatial and temporal gaussian grids for real-time free-viewpoint video (2025)
29. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM transactions on graphics (TOG) **42**(4), 139–1 (2023)

30. Kerr, J., Kim, C.M., Goldberg, K., Kanazawa, A., Tancik, M.: Lerf: Language embedded radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 19729–19739 (2023)
31. Kim, M., Lim, J., Han, B.: 4d gaussian splatting in the wild with uncertainty-aware regularization. Advances in Neural Information Processing Systems (2024)
32. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4015–4026 (2023)
33. Kong, H., Yang, X., Wang, X.: Efficient gaussian splatting for monocular dynamic scene rendering via sparse time-variant attribute modeling. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 4374–4382 (2025)
34. Lee, J.C., Rho, D., Sun, X., Ko, J.H., Park, E.: Compact 3d gaussian representation for radiance field. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21719–21728 (2024)
35. Lee, J., Won, C.Y., Jung, H., Bae, I., Jeon, H.G.: Fully explicit dynamic gaussian splatting. Advances in Neural Information Processing Systems (2024)
36. Li, H., Li, S., Gao, X., Batuer, A., Yu, L., Liao, Y.: Gifstream: 4d gaussian-based immersive video with feature stream. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 21761–21770 (2025)
37. Li, J., Song, Z., Yang, B.: Trace: Learning 3d gaussian physical dynamics from multi-view videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8820–8829 (2025)
38. Li, L., Shen, Z., Wang, Z., Shen, L., Tan, P.: Streaming radiance fields for 3d video synthesis. Advances in Neural Information Processing Systems **35**, 13485–13498 (2022)
39. Li, T., Slavcheva, M., Zollhoefer, M., Green, S., Lassner, C., Kim, C., Schmidt, T., Lovegrove, S., Goesele, M., Newcombe, R., 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)
40. Li, W., Zhou, R., Zhou, J., Song, Y., Herter, J., Qin, M., Huang, G., Pfister, H.: 4d langsplat: 4d language gaussian splatting via multimodal large language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 22001–22011 (2025)
41. Li, Z., Chen, Z., Li, Z., Xu, Y.: Spacetime gaussian feature splatting for real-time dynamic view synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8508–8520 (2024)
42. Li, Z., Niklaus, S., Snavely, N., Wang, O.: 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)
43. Li, Z., Wang, Q., Cole, F., Tucker, R., Snavely, N.: Dynibar: Neural dynamic image-based rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4273–4284 (2023)
44. Liang, Y., Xu, T., Kikuchi, Y.: Himor: Monocular deformable gaussian reconstruction with hierarchical motion representation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)
45. Lin, Y., Dai, Z., Zhu, S., Yao, Y.: Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21136–21145 (2024)

46. Liu, X., Wu, X., Zhang, P., Wang, S., Li, Z., Kwong, S.: Compgs: Efficient 3d scene representation via compressed gaussian splatting. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 2936–2944 (2024)
47. Liu, Y.L., Gao, C., Meuleman, A., Tseng, H.Y., Saraf, A., Kim, C., Chuang, Y.Y., Kopf, J., Huang, J.B.: Robust dynamic radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13–23 (2023)
48. Lou, A., Planche, B., Gao, Z., Li, Y., Luan, T., Ding, H., Chen, T., Noble, J., Wu, Z.: Darenerf: Direction-aware representation for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5031–5042 (2024)
49. Lu, J., Deng, J., Zhu, R., Liang, Y., Yang, W., Zhang, T., Zhou, X.: Dn-4dgs: Denoised deformable network with temporal-spatial aggregation for dynamic scene rendering. Advances in Neural Information Processing Systems (2024)
50. Lu, T., Yu, M., Xu, L., Xiangli, Y., Wang, L., Lin, D., Dai, B.: Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20654–20664 (2024)
51. Lu, Y., Zhou, Y., Liu, D., Liang, T., Yin, Y.: Bard-gs: Blur-aware reconstruction of dynamic scenes via gaussian splatting. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)
52. Lu, Z., Guo, X., Hui, L., Chen, T., Yang, M., Tang, X., Zhu, F., Dai, Y.: 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)
53. Luiten, J., Kopanas, G., Leibe, B., Ramanan, D.: Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In: 2024 International Conference on 3D Vision (3DV). pp. 800–809. IEEE (2024)
54. Mallick, S.S., Goel, R., Kerbl, B., Steinberger, M., Carrasco, F.V., De La Torre, F.: Taming 3dgs: High-quality radiance fields with limited resources. In: SIGGRAPH Asia 2024 Conference Papers. pp. 1–11 (2024)
55. Mihajlovic, M., Prokudin, S., Pollefeys, M., Tang, S.: Resfields: Residual neural fields for spatiotemporal signals. arXiv preprint arXiv:2309.03160 (2023)
56. Niedermayr, S., Stumpfegger, J., Westermann, R.: Compressed 3d gaussian splatting for accelerated novel view synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10349–10358 (2024)
57. Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: Nerfies: Deformable neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5865–5874 (2021)
58. Park, K., Sinha, U., Hedman, P., Barron, J.T., Bouaziz, S., Goldman, D.B., Martin-Brualla, R., Seitz, S.M.: Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. ACM transactions on graphics (TOG) **40**(6) (dec 2021)
59. Park, S., Son, M., Jang, S., Ahn, Y.C., Kim, J.Y., Kang, N.: Temporal interpolation is all you need for dynamic neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4212–4221 (2023)
60. Peng, S., Yan, Y., Shuai, Q., Bao, H., Zhou, X.: Representing volumetric videos as dynamic mlp maps. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4252–4262 (2023)

61. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: 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)
62. Qin, M., Li, W., Zhou, J., Wang, H., Pfister, H.: Langsplat: 3d language gaussian splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20051–20060 (2024)
63. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
64. Shao, R., Zheng, Z., Tu, H., Liu, B., Zhang, H., Liu, Y.: 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)
65. Shaw, R., Nazarczuk, M., Song, J., Moreau, A., Catley-Chandar, S., Dhamo, H., Pérez-Pellitero, E.: Swings: sliding windows for dynamic 3d gaussian splatting. In: European Conference on Computer Vision. Springer (2024)
66. Shin, S., Park, J.: Binary radiance fields. Advances in neural information processing systems **36**, 55919–55931 (2023)
67. Song, L., Chen, A., Li, Z., Chen, Z., Chen, L., Yuan, J., Xu, Y., Geiger, A.: Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. IEEE Transactions on Visualization and Computer Graphics **29**(5), 2732–2742 (2023)
68. Sun, J., Jiao, H., Li, G., Zhang, Z., Zhao, L., Xing, W.: 3dgstream: On-the-fly training of 3d gaussians for efficient streaming of photo-realistic free-viewpoint videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20675–20685 (2024)
69. Sun, X., Lee, J.C., Rho, D., Ko, J.H., Ali, U., Park, E.: F-3dgs: Factorized coordinates and representations for 3d gaussian splatting. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 7957–7965 (2024)
70. Tang, L., Yang, J., Peng, R., Zhai, Y., Shen, S., Wang, R.: Compressing streamable free-viewpoint videos to 0.1 mb per frame. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 7257–7265 (2025)
71. Tian, F., Du, S., Duan, Y.: Mononerf: Learning a generalizable dynamic radiance field from monocular videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17903–17913 (2023)
72. Tian, L., Li, X., Ma, L., Yin, H., Zheng, Z., Huang, H., Li, T., Lu, H., Jia, X.: Ccl-lgs: Contrastive codebook learning for 3d language gaussian splatting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9855–9864 (2025)
73. Tretschk, E., Tewari, A., Golyanik, V., Zollhöfer, M., Lassner, C., Theobalt, C.: Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12959–12970 (2021)
74. Wan, D., Lu, R., Zeng, G.: Superpoint gaussian splatting for real-time high-fidelity dynamic scene reconstruction. In: Forty-first International Conference on Machine Learning (2024)
75. Wang, C., MacDonald, L.E., Jeni, L.A., Lucey, S.: Flow supervision for deformable nerf. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21128–21137 (2023)

76. Wang, F., Chen, Z., Wang, G., Song, Y., Liu, H.: Masked space-time hash encoding for efficient dynamic scene reconstruction. Advances in Neural Information Processing Systems **36** (2024)
77. Wang, F., Tan, S., Li, X., Tian, Z., Song, Y., Liu, H.: Mixed neural voxels for fast multi-view video synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19706–19716 (2023)
78. Wang, H., Zhu, H., He, T., Feng, R., Deng, J., Bian, J., Chen, Z.: End-to-end rate-distortion optimized 3d gaussian representation. In: European Conference on Computer Vision. pp. 76–92. Springer (2024)
79. Wang, L., Hu, Q., He, Q., Wang, Z., Yu, J., Tuytelaars, T., Xu, L., Wu, M.: Neural residual radiance fields for streamably free-viewpoint videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 76–87 (2023)
80. Wang, L., Yao, K., Guo, C., Zhang, Z., Hu, Q., Yu, J., Xu, L., Wu, M.: Videorf: Rendering dynamic radiance fields as 2d feature video streams. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 470–481 (2024)
81. Wang, P., Liu, Y., Chen, Z., Liu, L., Liu, Z., Komura, T., Theobalt, C., Wang, W.: 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)
82. Wang, P., Zhang, Z., Wang, L., Yao, K., Xie, S., Yu, J., Wu, M., Xu, L.: V^ 3: Viewing volumetric videos on mobiles via streamable 2d dynamic gaussians. ACM Transactions on Graphics (TOG) **43**(6), 1–13 (2024)
83. Wang, R., Lohmeyer, Q., Meboldt, M., Tang, S.: Degauss: Dynamic-static decomposition with gaussian splatting for distractor-free 3d reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6294–6303 (2025)
84. Wang, Y., Yang, P., Xu, Z., Sun, J., Zhang, Z., Chen, Y., Bao, H., Peng, S., Zhou, X.: Freetimegs: Free gaussian primitives at anytime anywhere for dynamic scene reconstruction. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 21750–21760 (2025)
85. Wang, Y., Li, Z., Guo, L., Yang, W., Kot, A., Wen, B.: Contextgs: Compact 3d gaussian splatting with anchor level context model. Advances in neural information processing systems **37**, 51532–51551 (2024)
86. Wang, Z., Tan, J., Khurana, T., Peri, N., Ramanan, D.: Monofusion: Sparse-view 4d reconstruction via monocular fusion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8252–8263 (2025)
87. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Wang, X.: 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)
88. Wu, J., Peng, R., Jiao, J., Yang, J., Tang, L., Xiong, K., Liang, J., Yan, J., Liu, R., Wang, R.: Localdygs: Multi-view global dynamic scene modeling via adaptive local implicit feature decoupling. arXiv preprint arXiv:2507.02363 (2025)
89. Wu, J., Peng, R., Wang, Z., Xiao, L., Tang, L., Yan, J., Xiong, K., Wang, R.: Swift4d: Adaptive divide-and-conquer gaussian splatting for compact and efficient reconstruction of dynamic scene. International Conference on Learning Representations (ICLR) (2025)

90. Wu, M., Wang, Z., Kouros, G., Tuytelaars, T.: Tetrirf: Temporal tri-plane radiance fields for efficient free-viewpoint video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6487–6496 (2024)
91. Xian, W., Huang, J.B., Kopf, J., Kim, C.: Space-time neural irradiance fields for free-viewpoint video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9421–9431 (2021)
92. Xu, J., Fan, Z., Yang, J., Xie, J.: Grid4d: 4d decomposed hash encoding for high-fidelity dynamic gaussian splatting. Advances in Neural Information Processing Systems (2024)
93. Yan, J., Peng, R., Tang, L., Wang, R.: 4d gaussian splatting with scale-aware residual field and adaptive optimization for real-time rendering of temporally complex dynamic scenes. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 7871–7880 (2024)
94. Yan, J., Peng, R., Wang, Z., Tang, L., Yang, J., Liang, J., Wu, J., Wang, R.: Instant gaussian stream: Fast and generalizable streaming of dynamic scene reconstruction via gaussian splatting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16520–16531 (2025)
95. Yan, Z., Li, C., Lee, G.H.: 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)
96. Yang, J.W., Sun, J.M., Yang, Y.L., Yang, J., Shan, Y., Cao, Y.P., Gao, L.: Dmit: Deformable mipmapped tri-plane representation for dynamic scenes. In: European Conference on Computer Vision. pp. 436–453. Springer (2025)
97. Yang, L., Zhu, K., Tian, J., Zeng, B., Lin, M., Pei, H., Zhang, W., Yan, S.: Widerange4d: Enabling high-quality 4d reconstruction with wide-range movements and scenes. arXiv preprint arXiv:2503.13435 (2025)
98. Yang, Z., Yang, H., Pan, Z., Zhang, L.: Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In: International Conference on Learning Representations (ICLR) (2024)
99. Yang, Z., Gao, X., Zhou, W., Jiao, S., Zhang, Y., Jin, X.: Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20331–20341 (2024)
100. Yin, D., Shi, J., Zhang, M., Huang, Z., Liu, J., Dong, F.: Fsvfg: Towards immersive full-scene volumetric video streaming with adaptive feature grid. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 11089–11098 (2024)
101. Yoon, J., Han, S., Oh, J., Lee, M.: Splinegs: Learning smooth trajectories in gaussian splatting for dynamic scene reconstruction. In: The Thirteenth International Conference on Learning Representations (2025)
102. Yu, H., Julin, J., Milacski, Z.A., Niinuma, K., Jeni, L.A.: Dylin: Making light field networks dynamic. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12397–12406 (2023)
103. Yuan, Y., Shen, Q., Yang, X., Wang, X.: 1000+ fps 4d gaussian splatting for dynamic scene rendering. arXiv preprint arXiv:2503.16422 (2025)
104. Yun, Y., Bae, J., Son, H., Kim, S., Lee, H., Bang, G., Uh, Y.: Compensating spatiotemporally inconsistent observations for online dynamic 3d gaussian splatting. In: Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers. pp. 1–9 (2025)

105. Zhan, Y., Li, Z., Niu, M., Zhong, Z., Nobuhara, S., Nishino, K., Zheng, Y.: Kfd-nerf: Rethinking dynamic nerf with kalman filter. In: European Conference on Computer Vision. Springer (2024)
106. Zhan, Y.T., Ho, C.Y., Yang, H., Chen, Y.H., Chiang, J.C., Liu, Y.L., Peng, W.H.: Cat-3dgs: A context-adaptive triplane approach to rate-distortion-optimized 3dgs compression. International Conference on Learning Representations (ICLR) (2025)
107. Zhang, Z., Song, T., Lee, Y., Yang, L., Peng, C., Chellappa, R., Fan, D.: Lp-3dgs: Learning to prune 3d gaussian splatting. In: Advances in neural information processing systems (2024)
108. Zhang, Z., Lu, G., Liang, H., Cheng, Z., Tang, A., Song, L.: Rate-aware compression for nerf-based volumetric video. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 3974–3983 (2024)
109. Zhao, B., Li, Y., Sun, Z., Zeng, L., Shen, Y., Ma, R., Zhang, Y., Bao, H., Cui, Z.: Gaussianprediction: Dynamic 3d gaussian prediction for motion extrapolation and free view synthesis. In: ACM SIGGRAPH 2024 Conference Papers. pp. 1–12 (2024)
110. Zheng, Z., Wu, Z., Zhong, H., Tian, Y., Cao, N., Xu, L., Yao, J., Zhang, X., Hu, Q., Zhang, W.: 4dgcpro: Efficient hierarchical 4d gaussian compression for progressive volumetric video streaming. arXiv preprint arXiv:2509.17513 (2025)
111. Zheng, Z., Zhong, H., Hu, Q., Zhang, X., Song, L., Zhang, Y., Wang, Y.: Hpc: Hierarchical progressive coding framework for volumetric video. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 7937–7946 (2024)
112. Zhou, S., Chang, H., Jiang, S., Fan, Z., Zhu, Z., Xu, D., Chari, P., You, S., Wang, Z., Kadambi, A.: Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21676–21685 (2024)
113. Zhu, R., Liang, Y., Chang, H., Deng, J., Lu, J., Yang, W., Zhang, T., Zhang, Y.: Motiongs: Exploring explicit motion guidance for deformable 3d gaussian splatting. Advances in Neural Information Processing Systems (2024)