ST-4DGS: Spatial-Temporally Consistent 4D Gaussian Splatting for Efficient Dynamic Scene Rendering

Deqi Li
dqli@mail.bnu.edu.cn
Beijing Normal University
Beijing, China

Shi-Sheng Huang
huangss@bnu.edu.cn
Beijing Normal University
Beijing, China

Zhiyuan Lu
zylu@mail.bnu.edu.cn
Beijing Normal University
Beijing, China

Xinran Duan
202011081033@mail.bnu.edu.cn
Beijing Normal University
Beijing, China

Hua Huang*
huahuang@bnu.edu.cn
Beijing Normal University
Beijing, China

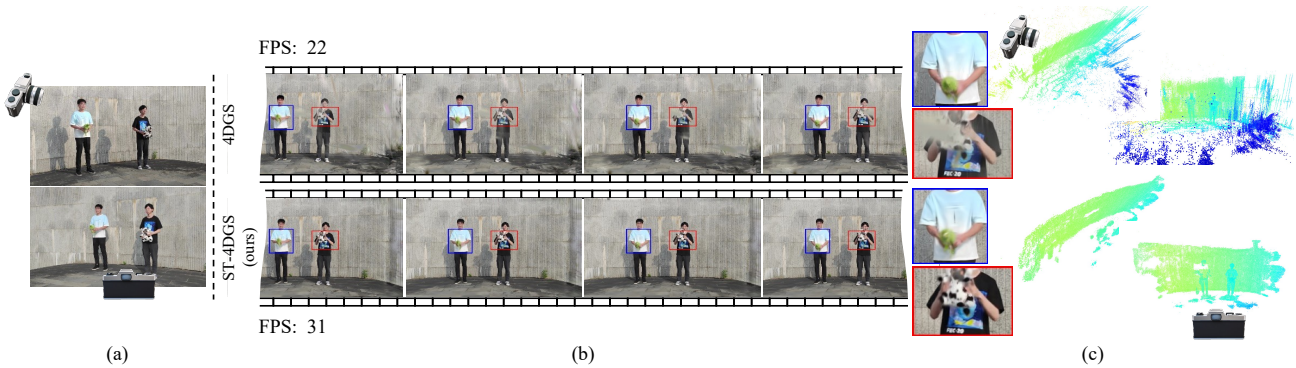

Figure 1: This paper proposes a novel spatial-temporally consistent 4D Gaussian Splatting, i.e., ST-4DGS. Given multi-view images captured from a dynamic scene (a), the key benefit of ST-4DGS is to learn the *consistently compact* 4D Gaussians representation for the dynamic scene, thus enabling persistent dynamic novel view synthesis (b). Note that the 4D Gaussian learned by ST-4DGS is more compact than 4DGS [Wu et al. 2023] with significantly fewer Gaussian floaters in 3D space (blue points shown in (c)), which serves as the key factor for achieving high-fidelity dynamic scene rendering with a very efficient rendering speed.

ABSTRACT

Dynamic scene rendering at any novel view continues to be a difficult but important task, especially for high-fidelity rendering quality with efficient rendering speed. The recent 3D Gaussian Splatting, i.e., 3DGS, shows great success for static scene rendering with impressive quality at a very efficient speed. However, the extension of 3DGS from static scene to dynamic 4DGS is still challenging, even for scenes with modest amounts of foreground object movement (such as a human moving an object). This paper proposes a novel spatial-temporally 4D Gaussian Splatting, i.e., ST-4DGS, which aims at the spatial-temporally persistent dynamic rendering quality and maintains real-time rendering efficiency. The key ideas of ST-4DGS are two novel mechanisms: (1) a novel spatial-temporal 4D Gaussian Splatting with a motion-aware shape regularization, and (2) a spatial-temporal joint density control mechanism. The proposed mechanisms efficiently prevent the *compactness degeneration* of the 4D Gaussian representation during dynamic scene learning, thus leading to spatial-temporally consistent dynamic rendering quality. With extensive evaluation on public datasets, our ST-4DGS can achieve much better dynamic rendering quality than previous approaches, such as 4DGS, HexPlane, K-Plane, 4K4D, etc, and in a more efficient rendering speed for persistent dynamic rendering. To our best knowledge, ST-4DGS is a new state-of-the-art 4D Gaussian Splatting for high-fidelity dynamic rendering, especially ensuring the spatial-temporally consistent rendering quality in scenes with modest movement. The code is available at https://github.com/wanglids/ST-4DGS.

CCS CONCEPTS

• **Computing methodologies** → **Rendering**; **Image-based rendering**.

*Corresponding author

SIGGRAPH Conference Papers '24, July 27-August 1, 2024, Denver, CO, USA

ACM ISBN 979-8-4007-0525-0/24/07…$15.00
https://doi.org/10.1145/3641519.3657520

KEYWORDS

Dynamic Scene Rendering, 4D Gaussian Splatting, Spatial-Temporally Conistent

ACM Reference Format:

Deqi Li, Shi-Sheng Huang, Zhiyuan Lu, Xinran Duan, and Hua Huang. 2024. ST-4DGS: Spatial-Temporally Consistent 4D Gaussian Splatting for Efficient Dynamic Scene Rendering. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers '24 (SIGGRAPH Conference Papers '24), July 27-August 1, 2024, Denver, CO, USA*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3641519.3657520

1 INTRODUCTION

Dynamic scene rendering [Collet et al. 2015; De Aguiar et al. 2008; Gao et al. 2021; Hedman et al. 2018; Zitnick et al. 2004] aims at reconstructing dynamic 3D scenes from videos and enabling novel view synthesis at any viewpoint for immersive virtual display, which continues to be an important topic in the computer graphics and computer vision community. The essential requirement of dynamic scene rendering is to efficiently perform high-fidelity rendering of dynamic scenes, which has a wide range of applications such as VR/AR, sports broadcasting, and movie production.

The success of Neural Radiance Fields (NeRF) [Mildenhall et al. 2021] and its variants [Chen et al. 2021; Guo et al. 2023; Wang et al. 2021] have shown impressive novel view synthesis results via utilizing neural implicit representation with volume rendering. However, these methods are limited to static scenes. The subsequent works introduce extra-temporal deformation [Li et al. 2022a, 2021; Park et al. 2021; Pumarola et al. 2021; Song et al. 2023; Wu et al. 2020; Zhang et al. 2021] to expand NeRF's boundary of novel view synthesis for dynamic scenes. However, they still suffer from significant training and rendering costs. Although some recent works [Cao and Johnson 2023; Fang et al. 2022; Fridovich-Keil et al. 2023; Lin et al. 2022; Müller et al. 2022; Shao et al. 2023; Shuai et al. 2022] have proposed strategies to reduce the training time from days to hours, their processing based on volume rendering still bears a non-negligible latency, which limits their applications to lightweight application scenarios.

On the other hand, some recent works have proposed explicit scene representation schemes, such as point cloud [Cao et al. 2022; Xu et al. 2023] or 3D Gaussians [Kerbl et al. 2023], which significantly boost the rendering speed by utilizing the benefit of custom rasterization framework. Meanwhile, those point-based renderings often have limited rendering quality [Cao et al. 2022]. 4K4D [Xu et al. 2023] proposed to improve the rendering quality using a hybrid appearance model, but needs huge memory storage due to its appearance optimization for every individual frame. The 3D Gaussian Splatting [Kerbl et al. 2023], i.e., 3DGS, significantly boosts the rendering quality by differentiable splatting 3D Gaussians in a very efficient manner, which is viewed as one of the most promising rendering frameworks. However, one major issue when extending 3DGS from static to dynamic scene as 4DGS [Wu et al. 2023] is how to effectively maintain the *compactness* of the 3D Gaussians during the dynamic learning, thus enabling persistent dynamic rendering with spatial-temporal consistent quality [Li et al. 2023].

In this paper, we introduce a novel Spatial-temporally Consistent 4D Gaussian Splatting, i.e., ST-4DGS, which inherits the benefit of 3DGS with high-fidelity rendering quality and efficient rendering speed, and learns *consistently compact* 4D Gaussians for the dynamic scene to enable persistent dynamic scene rendering simultaneously. Following 3DGS, we represent the dynamic scene using a set of 3D Gaussians, deform the 3D Gaussians with a spatial-temporal deformation field, and perform the dynamic rendering using Gaussian splatting, thus building up a 4D Gaussian Splatting. More importantly, we provide two key mechanisms to prevent the *compactness degeneration* for the 4D Gaussians during training: (1) a motion-aware shape regularization strategy for the 4D Gaussians learning, which effectively decouples the 4D Gaussians' motion part and shape part for accurate 4D Gaussian deformation estimation, (2) a spatial-temporally joint density control, which incorporates the joint cues in both spatial and temporal domain together to adaptively control the 4D Gaussians' pruning and splitting. Using such two mechanisms, we can effectively preserve the *compactness* of the 4D Gaussians, thus enabling spatial-temporally consistent dynamic rendering in high-fidelity quality as shown in Fig. 1.

To evaluate the effectiveness of our ST-4DGS, we perform extensive evaluation on public dynamic datasets, such as DyNeRF [Li et al. 2022a], ENeRF-Outdoor [Lin et al. 2022], and Dynamic Scene [Yoon et al. 2020] datasets, by comparing with state-of-the-art dynamic rendering approaches like ENeRF [Lin et al. 2022], HexPlane [Cao and Johnson 2023], K-Plane [Fridovich-Keil et al. 2023], 4DGS [Wu et al. 2023], and 4K4D [Xu et al. 2023]. Our ST-4DGS can achieve much better persistent dynamic rendering quality than those previous approaches. For 4K4D [Xu et al. 2023], our ST-4DGS achieves comparable rendering accuracy in quantitative evaluation, but with more persistent rendering quality (see the accompanying video) thanks to the spatial-temporal consistency strategy in our ST-4DGS. Besides, since the number of 4D Gaussians in our ST-4DGS is very compact and far less than the dense points used in 4K4D [Xu et al. 2023], the rendering speed of our ST-4DGS can be about 2× faster than 4K4D with the similar rendering processing, but with much more efficient training time (about 5× faster than 4K4D) and much less memory storage costs. Although the evaluation datasets are mostly dynamic scenes with modest movement, our strategy for ST-4DGS learning takes effects to improve the output quality of 4DGS [Wu et al. 2023], and can serve as general and effective prior for better dynamic scene learning. To the best of our knowledge, our ST-4DGS is a new state-of-the-art 4D Gaussian Splatting for high-fidelity dynamic rendering quality with efficient rendering speed, especially ensuring the spatial-temporally consistent rendering quality.

2 RELATED WORK

2.1 Neural Scene Representation

Unlike the early novel view synthesis methods that take special effects to reconstruct the 3D scenes' appearance and illumination [Nam et al. 2018; Xia et al. 2016], the recent progress use neural scene representations, such as multi-plane images (MPI) [Han et al. 2022; Ouyang et al. 2022; Tucker and Snavely 2020], implicit neural radiance fields (NeRF) [Chen et al. 2021; Deng et al. 2022; Guo et al. 2023; Mildenhall et al. 2021; Wang et al. 2021; Zhang et al. 2022], and explicit representations [Cao et al. 2022; Choi et al. 2019; Luo et al.

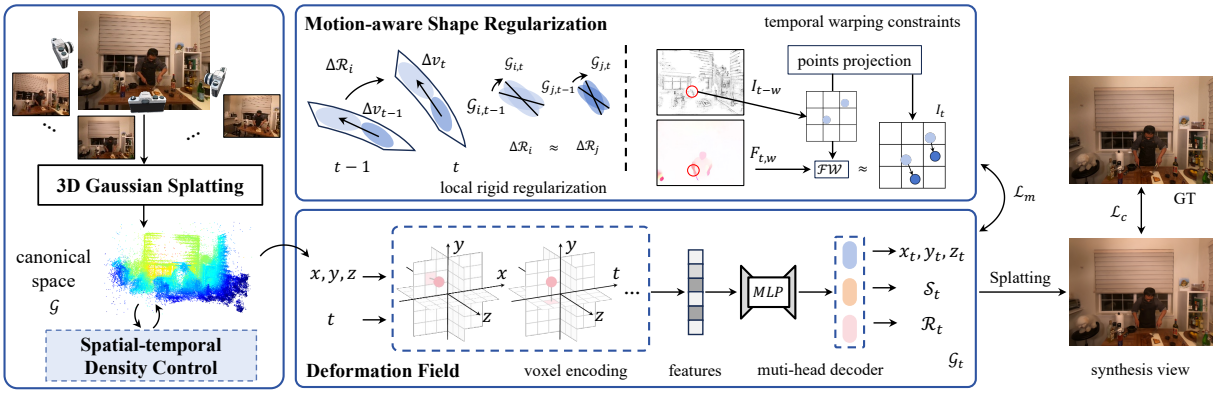

Figure 2: The overview of ST-4DGS. Based on the 3DGS, ST-4DGS represents the dynamic scene with a 4D Gaussian Splatting, which contains a deformation field that warps dynamic 3D Gaussians using a spatial-temporal voxel encoder. More importantly, our ST-4DGS uses a motion-aware shape regularization and spatial-temporal density control to learn much better *compact* 4D Gassuians for high-fidelity dynamic rendering.

2019; Nguyen-Ha et al. 2022] to perform photo-realistic rendering quality.

With adaptive depth sampling [Han et al. 2022; Tucker and Snavely 2020], the MPI has shown high-quality novel view synthesis results from only single images. However, when facing complex scenes or multi-view inputs, MPI often fails to represent scenes' fine-grained geometry thus easily leading to visual artifacts in the synthesized novel view. NeRF [Mildenhall et al. 2021] and its variants [Chen et al. 2021; Guo et al. 2023; Wang et al. 2021] adopt to represent the 3D scene as an implicit radiance field and achieve impressive novel view synthesis results via volume rendering. Some works proposed to use explicit representation, such as point cloud [Cao et al. 2022] based on the custom rasterization framework for efficient scene rendering. Nevertheless, the rendering quality is still limited. Recently, 3D Gaussian Splatting [Kerbl et al. 2023] significantly boosts the rendering quality and speed using an explicit representation, i.e., 3DGS, which is regarded as the most promising scene rendering approach for novel view synthesis. However, all of these approaches are limited to static scenes and would fail to perform realistic rendering for dynamic scenes.

Our ST-4DGS is inspired by those previous neural scene representations for static scenes but aims to promote rendering quality for dynamic scenes.

2.2 Dynamic Scene Rendering

Traditional dynamic scene rendering approaches often rely on depth-image-based rendering [Li et al. 2022b; Zitnick et al. 2004] for multi-view warping and blending, but the dynamic rendering quality is limited by accurate depth. The success of NeRF has inspired many subsequent works to extend it for dynamic scene rendering using an extra deformation field [Li et al. 2022a, 2021; Park et al. 2021; Pumarola et al. 2021; Song et al. 2023; Wu et al. 2020; Zhang et al. 2021]. However, these approaches are too time-consuming that unable to support fast dynamic scene rendering. Although the subsequent works such as ENeRF [Lin et al. 2022], NeuralVoxels [Fang et al. 2022], HexPlane [Cao and Johnson 2023], K-Plane [Fridovich-Keil et al. 2023], Im4D [Lin et al. 2023] etc, have proposed various acceleration techniques to reduce training time. However, volume-based rendering processing makes real-time dynamic scene rendering difficult.

Recently, inspired by the success of 3DGS [Kerbl et al. 2023], 4DGS [Wu et al. 2023] proposed to encode 3D Gaussians using a temporal conditioned deformation field. But 4DGS learns the 3D Gaussian deformation (including the motion deformation and shape deformation) using only one total loss in a deformation field, the learnable 3D Gaussians can easily degenerate during the dynamic scene learning, thus often limited to achieving persistent rendering quality. 4K4D [Xu et al. 2023] proposed a novel new explicit rendering based on dense point cloud representation, which achieves impressive rendering quality and speed. But its dynamic rendering is not spatial-temporally consistent, and often gets obvious rendering artifacts such as point flickers, especially for complex dynamic scenes.

In contrast, our ST-4DGS focuses on preserving the spatial-temporally consistent dynamic rendering quality, which can achieve much better rendering quality than those previous approaches.

3 METHODS

Given multi-view images captured from a dynamic scene as input, our approach aims to learn the 4D Gaussian representation $\mathcal{G}_t$ for the dynamic scene, which enables dynamic novel view synthesis at any timestamp t. We first introduce a 4D Gaussian Splatting (Sec. 3.1) to obtain $\mathcal{G}_t$ via a spatial-temporal deformation field $\mathcal{F}$. More importantly, as shown in Fig. 2, we propose two key components to learn a compact $\mathcal{G}_t$, including a motion-aware shape regularization (Sec. 3.2) and a spatial-temporal density control strategy (Sec. 3.3).

Compact Gaussian Representation. We mean a Gaussian representation as compact when it tightly adheres to the surface of objects in 3D space. Otherwise, when there exist lots of floaters like those of 4DGS [Wu et al. 2023] as shown in Fig. 8, we call the compactness of those 3D Gaussians being degraded, which would lead to obvious artifacts when performing dynamic scene rendering.

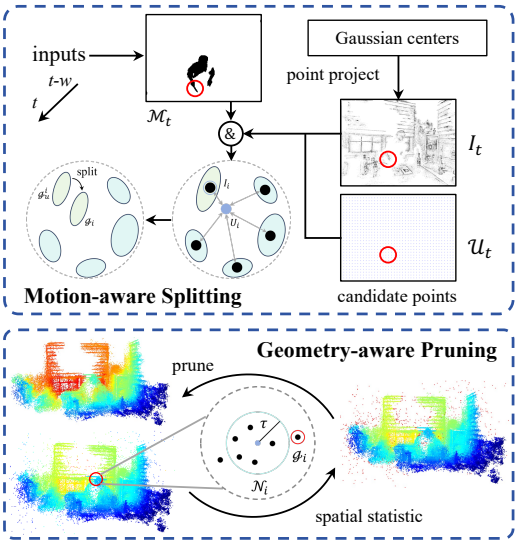

Figure 3: Illustration of Spatial-temporal Density Control strategy.

3.1 4D Gaussian Splatting

Following 3DGS [Kerbl et al. 2023], we first reconstruct a coarse 3D Gaussian representation in a canonical space $\mathcal{G} = \{\mathcal{X}, \mathcal{S}, \mathcal{R}, \alpha, C\}$, where $\mathcal{X}$,$\mathcal{S}$,$\mathcal{R}$, α, C represent the 3D Gaussians' spatial position, scale, rotation, alpha and spherical harmonic coefficients (SH) respectively. As shown in Fig 2, we adopt the spatial-temporal encoder framework like HexPlane [Cao and Johnson 2023] to construct the deformation field $\mathcal{F}$ for $\mathcal{G}$ at any timestamp t. Specifically, we set the deformation field $\mathcal{F}(\mathcal{X}, t)$ as parameter offset $\delta\mathcal{X}$,$\delta\mathcal{S}$, $\delta\mathcal{R}$, i.e., $\mathcal{F}(\mathcal{X}, t) = (\delta\mathcal{X}, \delta\mathcal{S}, \delta\mathcal{R})$, and deform $\mathcal{G}$ to $\mathcal{G}_t$ following

$$\mathcal{G}_t = \{\mathcal{X}_t, \mathcal{S}_t, \mathcal{R}_t, \alpha_t, C_t\},$$

$$(\mathcal{X}_t, \mathcal{S}_t, \mathcal{R}_t) = \mathcal{F}(\mathcal{X}, t) + (\mathcal{X}, \mathcal{S}, \mathcal{R}), \tag{1}$$

where $\alpha_t = \alpha$, $C_t = C$ as fixed. When performing rendering at any given camera position $[R', T']$, we splat the deformed 3D Gaussians $\mathcal{G}_t$ following the splatting [Kerbl et al. 2023] (written as Q) to obtain the rendering image $\mathcal{I}_t$ as:

$$\mathcal{I}_t = Q(\mathcal{G}_t | [R', T']). \tag{2}$$

3.2 Motion-aware Shape Regularization

In 4DGS [Wu et al. 2023], one major drawback of deformation field $\mathcal{F}$ is that the 4D Gaussians' motion parameter $\mathcal{X}$ and shape parameters $(\mathcal{S}, \mathcal{R})$ are tightly coupled together during training. Without an effective decoupling strategy, the compactness of 4D Gaussians would easily degenerate, which significantly decreases the quality of dynamic rendering.

To alleviate this problem, we propose motion-aware shape regularity for the dynamic 4D Gaussians $\mathcal{G}_t$ learning. It effectively decouples the entanglement between motion and shape parameters of dynamic Gaussians via local rigid regularization and temporal warping constraint.

Local Rigid Regularization. As shown in Fig. 2, for a pair of consecutive 3D Gaussians $(\mathcal{G}_{t-1}, \mathcal{G}_t)$, our key observation is that their shape deformation can be instantaneous *locally* rigid, which can be used to regularize the 3D Gaussians' motion during training effectively. Specifically, for any 3D Gaussian $g_{i,t}$, its neighbor 3D Gaussian $g_{j,t}$ should move following a rigid transform as much as possible. Following D3G [Luiten et al. 2023], we use the K-nearest-neighbor (KNN) Euclidean distance to construct a local neighbor subset with Gaussians' index denoted as $\mathcal{N}_i \in \mathbb{R}^{20}$ (20 neighbors) for each $g_{i,t}$. For a pair of neighboring Gaussians $(g_{i,t}, g_{j,t})$, we assume they have similar rotational variation, $\mathcal{L}_{rot}$ (Eq. 5). Similarly, for their spatial position displacement vector $\Delta v_t = (\mathcal{X}_{j,t} - \mathcal{X}_{i,t})$ ($\mathcal{X}_{i,t}$,$\mathcal{X}_{j,t}$ are the spatial position of $g_{i,t}$, $g_{j,t}$ respectively), we also assume Δv_t can be estimated from Δv_{t-1} by the rotation transformation $\Delta\mathcal{R}_i = \mathcal{R}_{i,t-1}\mathcal{R}_{i,t}^{-1}$, i.e., $\Delta v_{t-1} \sim \Delta\mathcal{R}_i \Delta v_t$. Therefore, the local rigid regularization $\mathcal{L}_{loc}$ can be formulated as

$$\mathcal{L}_{loc} = \lambda_{rig}\mathcal{L}_{rig} + \lambda_{rot}\mathcal{L}_{rot}, \tag{3}$$

$$\mathcal{L}_{rig} = \frac{1}{k|\mathcal{G}|}\sum_{g_i \in \mathcal{G}}\sum_{j \in \mathcal{N}_i} w_{i,j} \left\| \Delta v_{t-1} - \Delta\mathcal{R}_i \Delta v_t \right\|_2, \tag{4}$$

$$\mathcal{L}_{rot} = \frac{1}{k|\mathcal{G}|}\sum_{g_i \in \mathcal{G}}\sum_{j \in \mathcal{N}_i} w_{i,j} \left\| \mathcal{R}_{i,t-1}\mathcal{R}_{i,t}^{-1} - \mathcal{R}_{j,t-1}\mathcal{R}_{j,t}^{-1} \right\|_2, \tag{5}$$

$$w_{i,j} = exp\left(-\lambda_w \left\| \mathcal{X}_{j,t-1} - \mathcal{X}_{i,t-1} \right\|_2^2\right), \tag{6}$$

λ_{rig} and λ_{rot} are all set to 0.01, λ_w is set to -2000.

Temporal Warping Constraints. Similarly, our other observation is that when projecting the position center of a 3D Gaussian pair $(\mathcal{G}_{t-w}, \mathcal{G}_t)$ to their corresponding image planes, the 2D position warping between those projection pixels should be similar to the warping detected via optical flows.

Specifically, given a time window w, we first project the 3D Gaussian center $\mathcal{X}_{t-w}$ and $\mathcal{X}_t$ to their image planes as 2D coordinates I_{t-w} and I_t respectively. Then we use RAFT [Teed and Deng 2020] to estimate the optical flow $F_{t,w}$ from the image domain. Finally, by warping I_{t-w} to I_t with a warping operation $\mathcal{FW}$ (pixel translation), we calculate the L_1 distance between the pixel offsets and obtain a temporal warping loss $\mathcal{L}_{tem}$ as

$$\mathcal{L}_{tem} = \left\| I_t - \mathcal{FW}\left(I_{t-w}; F_{t,w}\right) \right\|_1. \tag{7}$$

In addition, inspired by PhyGaussian [Xie et al. 2023], we also introduce an extra anisotropic regularity term $\mathcal{L}_{ani}$ to improve the 3D Gaussians shape of thin objects in 3D scenes following

$$\mathcal{L}_{ani} = \frac{1}{|\mathcal{S}|}\sum_{s_i \in \mathcal{S}} \max\left\{\frac{\max(\mathcal{S}_i)}{\min(\mathcal{S}_i)}, \tau\right\} - \tau, \tag{8}$$

where $\tau = 10$ is a scale threshold to constrain Gaussian shapes. The overall loss function for the motion-aware regularization term is defined as

$$\mathcal{L}_m = \lambda_{tem}\mathcal{L}_{tem} + \lambda_{ani}\mathcal{L}_{ani} + \mathcal{L}_{loc}, \tag{9}$$

where λ_{tem} is set to 0.001 and λ_{ani} is set to 0.2.

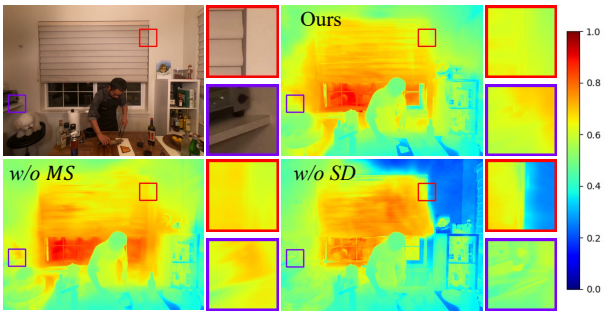

Figure 4: Visual comparison of relative depth images. *w/o MS* **represents ours without motion-aware shape regularization.** *w/o SD* **represents ours without spatial-temporal density control. Our depth image has fewer plush edges than** *w/o MS* **(purple box), and less incorrect depth affected by floaters in the background wall (red box blue area) than** *w/o SD*.

3.3 Spatial-temporal Density Control

For the original 3DGS [Luiten et al. 2023; Wu et al. 2023], the 3D Gaussians' density control is important to ensure the final rendering quality. However, the proposed adaptive density control of 3DGS is only feasible in static scenes, which would easily lead to 3D Gaussians degeneration as shown in Fig. 1 (c) and Fig. 8.

To mitigate this problem, we propose a spatial-temporal density control to optimize the 3D Gaussian distribution during training. It incorporates the spatial and temporal cues to perform geometry-aware pruning and motion-aware splitting respectively.

Geometry-aware Pruning. To effectively prune the floaters during the dynamic 3D Gaussian learning, we propose a geometry-aware pruning based on the 3D Gaussians' spatial distribution. As shown in Fig. 3, for any Gaussian g_j with the spatial position X_j, we use KNN to construct a local neighbor subset $\mathcal{N}_i$ and then calculate the spatial center $\bar{X}_i$ for all of 3D Gaussians in $\mathcal{N}_i$, i.e., $\bar{X}_i = \frac{1}{|\mathcal{N}_i|}\sum_{g_j \in \mathcal{N}_i} X_j$. Here we assume the distance for each 3D Gaussian g_j's spatial position X_j to spatial center $\bar{X}_i$ follow a Gaussian distribution, i.e. $X_j \sim \mathcal{N}(\bar{X}_i, \sigma)$. Thereafter, if the distance $d_i = |X_i - \bar{X}_i|$ is larger than 3σ, we determine g_i is a floater, otherwise is a compact Gaussian. Finally, we filter out all of the Gaussian floaters. The proposed geometry-aware pruning strategy is effective in removing floaters and achieving a compact reconstruction that Gaussians fit the geometric surface (Fig. 1 (c) and Fig. 8).

Motion-aware Splitting. We further introduce a motion-aware splitting strategy to densify the 3D Gaussians during the dynamic scene learning.

As shown in Fig. 3, we first compute a motion mask $\mathcal{M}_t$ via a threshold operation ($thr = 0.5$) on the optical flow $F_{t,w}$, and further improve it with a morphological operation. Here the motion mask $\mathcal{M}_t$ is used to identify dynamic 3D Gaussians. On the other hand, we project the 3D Gaussians' center $\mathcal{X}_t$ to the image plane and get a Gaussian projection image I_t. Also, we uniformly sample 2D position in the image plane to obtain a candidate point image $\mathcal{U}_t$. Then we use motion mask $\mathcal{M}_t$ to filter out dynamic Guassian projection image I'_t from I_t and dynamic candidate point image $\mathcal{U}'_t$ from $\mathcal{U}_t$ respectively. Since all of the positions in I'_t are located in the dynamic region, we add new 3D Gaussians according to the correlation between I'_t and $\mathcal{U}'_t$. Specifically, for each candidate point $U_i \in \mathcal{U}'_t$, we search the nearest Guassian projection point $I_i \in I'_t$, and finally add a new 3D Gaussian g_i by a Gaussian splitting from the original 3D Gaussian g^i_u, which is corresponding to U_i.

A naive densification strategy is to register $\mathcal{U}'_t$ as a point cloud with rendering depth (position) and clone the other Gaussian parameters of $\mathcal{G}$. However, the real projected coordinates of $\mathcal{G}$ may be far from $\mathcal{U}'_t$ in the dynamic scene, which will add incorrect floaters that degrade the compact geometric representation. Therefore, we use the split operation like 3DGS [Kerbl et al. 2023]to densify dynamic Gaussians which are identified by the motion information form $\mathcal{U}'_t$ conjugate in $\mathcal{G}$, following a motion-aware splitting manner. It is worth noting that we allow for perspective projection of 3D position, which is beneficial for optimizing 3D Gaussians located in the occluded background region. Please refer to our supplementary materials for more details.

Fig. 4 shows a tiny example of depth image rendering from different variants of ST-4DGS. It could be seen that the depth of our full system is more compact and accurate than other variants.

3.4 Coarse-to-Fine Learning

A coarse-to-fine learning scheme is applied to train the ST-4DGS. The coarse stage optimizes a canonical 3D Gaussian $\mathcal{G}$, and the fine stage learns the accurate spatial-temporal deformation field $\mathcal{F}$.

Specifically, we formulate the overall loss function $\mathcal{L}$ to train the ST-4DGS by combining a view synthesis term $\mathcal{L}_c$, a plane decoupling regularization term $\mathcal{L}_{TV}$ [Cao and Johnson 2023], and a motion-aware regularization term $\mathcal{L}_m$. $\mathcal{L}_c$ encourages the rendering image to match the ground truth. $\mathcal{L}_{TV}$ is a grid-based total-variational loss which learns the time dependence. By physically modeling Gaussian motion, $\mathcal{L}_m$ decouples the learning of motion and shape in the deformation field, encouraging force the spatial-temporal consistency. The overall loss is formulated as

$$\mathcal{L} = \lambda_c \mathcal{L}_c + \lambda_{TV}\mathcal{L}_{TV} + \mathcal{L}_m, \tag{10}$$

where λ_c is set to 1 and λ_{TV} is set to 2e-4.

4 EXPERIMENTS

4.1 Experimental Setup

Datasets. We evaluate the proposed ST-4DGS on three publicly available datasets of dynamic scenes, namely DyNeRF[Li et al. 2022a], ENeRF-Outdoor[Lin et al. 2022], and Dynamic Scenes[Yoon et al. 2020]. Following previous works, all images are resized with a ratio of 0.5. DyNeRF records cooking actions in a kitchen using 21 cameras. Each sequence of the DyNeRF contains 300 frames and the resolution is 1352×1014. ENeRF-Outdoor records long sequence videos (1200 frames) of multiple dynamic humans with objects in an outdoor scene using 18 cameras, and the resolution is 960×540. We select 100 frames from ENeRF-Outdoor for the experiment. Dynamic Scene records dynamic humans in rich-textured outdoor scenes using 12 cameras. Each sequence of the Dynamic Scene contains about 100-200 frames and the resolution is 960×506.

Training details. During training, we select one camera as the test view, and the leftover views are inputs for dynamic scene

Table 1: Quantitative comparison on the *Spinach, Beef, and Steak scenes* of the DyNeRF dataset. (·) is the rendering speed of our ST-4DGS using the same rendering acceleration technique as 4K4D. *Time* is the training time. Results from 4K4D and NeRFPlayer are from their original papers.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	FPS ↑	*Time* ↓
NeRFPlayer	32.07	0.931	0.281	0.09	5.5h
4K4D	32.85	0.973	0.167	110	∼24h
K-Plane	31.63	0.935	0.248	0.15	4.1h
HexPlane	31.70	0.943	0.216	0.21	12h
4DGS	31.03	0.938	0.167	36	2h
Ours	32.67	0.946	0.166	37 (335)	2.7h

Table 2: Quantitative comparison on *actor1-4, actor2-3*, and *actor5-6* of the ENeRF-Outdoor dataset. (·) is the rendering speed of our ST-4DGS using the same rendering acceleration technique as 4K4D. Results from ENeRF and 4K4D are from their original papers.

Method	PSNR ↑	SIMM ↑	LPIPS ↓	FPS ↑	*Time* ↓
ENeRF	25.45	0.809	0.273	11.3	-
4K4D	25.81	0.898	0.147	141	∼24h
K-Plane	23.00	0.818	0.194	0.25	2.7h
HexPlane	14.85	0.332	0.609	0.33	12.1h
4DGS	22.97	0.718	0.275	22	1.7h
Ours	26.74	0.856	0.128	31 (218)	2.3h

reconstruction. We use COLMAP [Schonberger and Frahm 2016] to initialize 3D Gaussians of the scene. Our proposed ST-4DGS executes 3K iterations for the coarse stage and 30K for the fine stage in training. The geometry-aware pruning is conducted every 500 iterations in the coarse stage ($\sigma = 1$), and 1K in the fine stage. Motion-aware splitting is only performed in the fine stage ($\sigma = 2$). The motion-aware shape regularization is applied after warm-up learning (15K), to prevent learning incorrect dynamic information for a better convergence of the deformation field learning. All the experiments are trained end-to-end in a single RTX 3090 GPU.

Metrics. We measure the view synthesis quality by using standard metrics, such as PSNR, Structural Similarity Index (SSIM), and perceptual similarity (LPIPS)[Zhang et al. 2018]. We also use the storage overhead (Mb) and rendering speed (FPS) to evaluate the efficiency of the proposed model.

4.2 Comparison with Previous Methods

We compare our method with previous state-of-the-art dynamic rendering approaches, including ENeRF [Lin et al. 2022], 4DGS [Wu et al. 2023], Hexplane [Cao and Johnson 2023], K-Plane [Fridovich-Keil et al. 2023], and 4K4D [Xu et al. 2023] respectively.

Table 3: Quantitative comparison on the *Ballon1, Ballon2* and *Skating* of the Dynamic Scene dataset. (·) is the rendering speed of our ST-4DGS using the same rendering acceleration technique as 4K4D.

Method	PSNR ↑	SIMM ↑	LPIPS ↓	FPS ↑	*Time* ↓
K-Plane	26.20	0.848	0.226	0.21	2.9 h
HexPlane	17.64	0.357	0.602	0.21	11.5 h
4DGS	17.85	0.604	0.409	25	1.9 h
Ours	28.66	0.897	0.121	29 (169)	2.9 h

Quantitative comparison. As shown in Table 1-Table 3, we perform a quantitative comparison between our ST-4DGS and previous approaches on the 3 public datasets. Compared to HexPlane [Cao and Johnson 2023], K-Plane [Fridovich-Keil et al. 2023] and 4DGS [Wu et al. 2023], our method can achieve much better accuracy values in every metric, which indicates that our ST-4DGS can achieve consistent rendering quality than those approaches. Our ST-4DGS also takes much less training time than HexPlane and K-Plane, with about 1/2 of training time for K-Plane and 1/4 for HexPlane. The rendering speed of our ST-4DGS is significantly faster than HexPlane [Cao and Johnson 2023] and K-Plane [Fridovich-Keil et al. 2023] with more than 100× faster efficiency.

Compared to 4DGS [Wu et al. 2023], our ST-4DGS also achieves faster rendering speed than 4DGS. One main reason would be that the dynamic 3D Gaussians learned by our ST-4DGS are more *compact* than 4DGS, which is beneficial for more efficient rendering. Our ST-4DGS's training time will be slightly higher than 4DGS. It is mainly caused by the extra motion-aware shape regularization for dynamic 3D Gaussian learning. But our ST-4DGS can achieve much better rendering quality via learning more compact 3D Gaussians. This rendering quality difference is much more significant in the Dynamic Scene dataset (Table 3). One main reason would be that scenes in the Dynamic Scene dataset have rich textures that require more 3D Gaussians to reconstruct. It leads to more compactness degeneration of 3D Gaussians in 4DGS than our ST-4DGS (Fig. 8).

Compared to 4K4D [Xu et al. 2023], our ST-4DGS achieves better accuracy results in the ENeRF-Outdoor dataset (e.g. PSNR is up to 26.74) and comparable accuracy in the DyNeRF dataset. If ST-4DGS uses the same rendering acceleration technology as 4K4D, i.e., each dynamic 3D Gaussian for frames is stored in the main memory before splatting inference, our ST-4DGS can achieve much faster rendering speed. The rendering speed of our ST-4DGS is about 2× faster than 4K4D, with 335fps (ST-4DGS, marked as (·)) v.s. 110fps (4K4D) in Table 1, 218fps (ST-4DGS, marked as (·)) v.s. 141fps (4K4D) in Table 2. Besides, since 4K4D performs individual point cloud learning for each frame without using a lightweight deformation field like ST-4DGS, the resource consumption for 4K4D is also significantly higher than our ST-4DGS.

Regarding memory storage, since we reconstruct a compact representation for dynamic scenes with fewer Gaussians, our ST-4DGS will cost less memory storage while maintaining faster rendering speed. Table 4 reports the comparison results on memory storage for different approaches. We can see that our ST-4DGS is still superior with smaller storage and more efficient rendering.

Table 4: Storage analysis on the 192-frame *Balloon1* of the Dynamic Scene dataset. The first three columns represent the storage consumption of the trained model, explicit Gaussians, and total (*Mb*). S/F is "Storage / Frame" and F/S is "FPS / Storage" which indicate the efficiency of models.

Method	Model	Data	Total	FPS↑	$S/F\downarrow$	$F/S\uparrow$
K-Plane	308.9	-	308.9	0.20	1.61	0.00064
HexPlane	327.3	-	327.3	0.22	1.76	0.00067
4DGS	91.5	126.4	217.9	21	1.13	0.096
Ours	91.5	96.7	188.2	28	0.98	0.148

Table 5: Ablation study on regularization terms and framework design on the DyNeRF dataset. *SP* is the spatial statistic pruning and *MS* is the motion-aware splitting.

Method	PSNR ↑	SIMM ↑	LPIPS ↓	*Time* ↓
w/o $\mathcal{L}_{ani}$	31.03	0.939	0.178	2.7 h
w/o $\mathcal{L}_{loc}$	30.32	0.928	0.187	2.3 h
w/o $\mathcal{L}_{tem}$	31.84	0.943	0.172	2.4 h
w/o *SP*	31.77	0.941	0.173	2.9 h
w/o *MS*	31.13	0.931	0.174	2.6 h
Ours	32.67	0.945	0.166	2.7 h

Qualitative Comparison. We also perform a qualitative comparison between our ST-4DGS and those previous approaches. As shown in Fig. 6, we can see that the results of HexPlane [Cao and Johnson 2023] and K-Plane [Fridovich-Keil et al. 2023] methods tend to be smooth without fine-grained details, and have many motion blurs, especially in the dynamic occlusion regions. The synthesis results of 4DGS [Wu et al. 2023] often exhibit visual artifacts and perspective distortions in the motion occlusion region. This condition is even worse for background regions with rich textures.

In contrast, our ST-4DGS can synthesize views with higher quality. Compared with HexPlane and K-Plane, our method ST-4DGS has better modeling ability for the appearance changes of motion occluded areas. Compared to 4DGS, our method reconstructs the scene with the correct layering. Even in the challenging multi-actor ENeRF-Outdoor dataset as shown in Fig. 7, our method ST-4DGS can recover the highly detailed appearance and geometry of the 4D dynamic scene (e.g. fingers and T-shirt patterns). Fig. 8 shows Gaussian spatial distributions of several scenes, where our ST-4DGS is much better than 4DGS with fewer floaters and compacter distribution. Please see more visual comparison results in the supplementary material and videos, where our ST-4DGS achieves better spatial-temporal consistent dynamic rendering quality than those previous approaches.

Results in long Sequence. To demonstrate the ability of ST-4DGS to reconstruct long-sequence dynamic scenes, we conduct comparative experiments on three ENeRF-Outdoor [Lin et al. 2022] sequences as long as 1200 frames. As shown in Table 6, although the accuracy results have decreased compared to the short frames training in Table 2, our ST-4DGS also achieves better accuracy results than 4DGS [Wu et al. 2023] and K-Plane[Fridovich-Keil et al. 2023]. Our ST-4DGS has about 2.1 dB PSNR accuracy improvement than 4DGS, which shows that the spatial-temporal consistent design of ST-4DGS also takes effective for long sequence dynamic view synthesis. Please refer to the supplementary materials and video for more details.

Table 6: Quantitative comparison between our method and other approaches on the long sequence ENeRF-Outdoor dataset (1200-frame).

Method	PSNR ↑	SIMM ↑	LPIPS ↓
K-Plane	22.98	0.769	0.249
4DGS	22.72	0.706	0.248
Ours	24.81	0.746	0.223

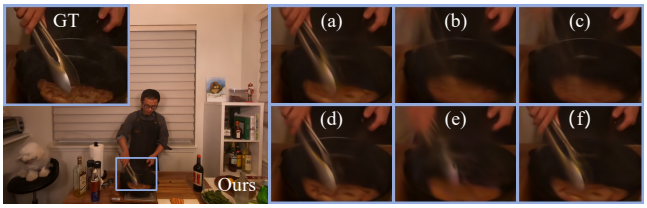

Figure 5: Ablation studies by visualization. (a) w/o $\mathcal{L}_{ani}$, (b) w/o $\mathcal{L}_{loc}$, (c) w/o $\mathcal{L}_{tem}$, (d) w/o *GP*, (e) w/o *MS*, and (f) Ours.

4.3 Ablution Study

We conduct ablation studies to validate the design choices in our ST-4DGS on the DyNeRF dataset[Li et al. 2022a]. The components related to view synthesis are removed in the ablation and the results are reported in Table 5, Fig. 4 and Fig. 5. It can be observed that deleting each component will reduce the ability of the framework to synthesize realistic views, resulting in poor metric values.

Effect of motion-aware regularization. The quantitative comparisons are conducted on components $\mathcal{L}_{ani}$, $\mathcal{L}_{loc}$, and $\mathcal{L}_{tem}$, respectively. If we discard $\mathcal{L}_{ani}$, the edge area of the synthetic view shows the plush products (Fig. 5 (a)), influenced by the narrow Gaussians. It demonstrates that $\mathcal{L}_{ani}$ can effectively constrain the scale shape of Gaussians. Fig. 5 (b) and (c) show that $\mathcal{L}_{loc}$ and $\mathcal{L}_{tem}$ can perceive changes in motion. The motion areas will synthesize artifacts when $\mathcal{L}_{loc}$ and $\mathcal{L}_{tem}$ are not both considered. Specifically, the performance drops if we discard the $\mathcal{L}_{loc}$ term (e.g., 2.35dB drop in PSNR), demonstrating the effectiveness of our local rigid regularization for perceiving motion in a compact dynamic Gaussian.

Effect of spatial-temporal density control. Fig. 4 *w/o SD* shows an error in the scene geometry without spatial-temporal density control, where depth values in the far background areas are small. This error is affected by floaters. If we discard geometry-aware pruning, the synthesized view is ambiguous (Fig. 5 (d)). The PSNR drops from 32.67 to 31.77. Motion-aware splitting can increase the Gaussian number, effectively improving the synthesized quality for motion regions. Without this component, the synthesized view

has significant ghosting on the motion region, especially in thin structures (Fig. 5 (e)). The PSNR and SIMM also decrease by 1.54dB and 0.0139.

Analyzing the splatting Speed. We also analyzed the main factors affecting rendering speed from the perspectives of rendering resolution and Gaussian number. We found that a compact Gaussian with few numbers is a key factor of ST-4DGS to achieve real-time view synthesis.

4.4 Limitations

Our ST-4DGS still has several limitations. Our proposed motion-aware shape regularization is based on the physical mechanism of short-term local rigidity consistency, so ST-4DGS would fail when dynamic objects move following long-term or non-rigid deformation. Another major limitation of our ST-4DGS is that the deformation field cannot accurately learn challenging 3D Gaussian deformations in motion. One interesting future work is to combine more effective regularization of long-term or non-rigid movements, enabling more accurate motion predictions even in challenging movements.

5 CONCLUSION

This paper presents a novel ST-4DGS for dynamic view synthesis with high-fidelity rendering quality and efficient rendering speed. The key components of ST-4DGS are motion-aware shape regularization and spatial-temporal density control, to learn consistently compact 4D Gaussians. We conduct extensive experiments on multiple dynamic datasets. Both quantitative and qualitative results demonstrate the effectiveness of our approach for realistic dynamic view synthesis with real-time speed. In the future, we hope this work can inspire subsequent works for more persistent and efficient dynamic scene rendering.

ACKNOWLEDGMENTS

We thank the reviewers for their constructive suggestions. This project was supported by the Natural Science Foundation of China (Project Number 62202057).

REFERENCES

Ang Cao and Justin Johnson. 2023. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 130–141.

Ang Cao, Chris Rockwell, and Justin Johnson. 2022. Fwd: Real-time novel view synthesis with forward warping and depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15713–15724.

Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. 2021. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF international conference on computer vision*. 14124–14133.

Inchang Choi, Orazio Gallo, Alejandro Troccoli, Min H Kim, and Jan Kautz. 2019. Extreme view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7781–7790.

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

Edilson De Aguiar, Carsten Stoll, Christian Theobalt, Naveed Ahmed, Hans-Peter Seidel, and Sebastian Thrun. 2008. Performance capture from sparse multi-view video. In *ACM SIGGRAPH 2008 papers*. 1–10.

Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. 2022. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12882–12891.

Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. 2022. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia 2022 Conference Papers*. 1–9.

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. 2023. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12479–12488.

Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. 2021. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5712–5721.

Mantang Guo, Junhui Hou, Jing Jin, Hui Liu, Huanqiang Zeng, and Jiwen Lu. 2023. Content-aware warping for view synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).

Yuxuan Han, Ruicheng Wang, and Jiaolong Yang. 2022. Single-view view synthesis in the wild with learned adaptive multiplane images. In *ACM SIGGRAPH 2022 Conference Proceedings*. 1–8.

Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. 2018. Deep blending for free-viewpoint image-based rendering. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–15.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–14.

Deqi Li, Shi-Sheng Huang, Tianyu Shen, and Hua Huang. 2023. Dynamic View Synthesis with Spatio-Temporal Feature Warping from Sparse Views. In *Proceedings of the 31st ACM International Conference on Multimedia*. 1565–1576.

Shuai Li, Kaixin Wang, Yanbo Gao, Xun Cai, and Mao Ye. 2022b. Geometric warping error aware CNN for DIBR oriented view synthesis. In *Proceedings of the 30th ACM International Conference on Multimedia*. 1512–1521.

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

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

Haotong Lin, Sida Peng, Zhen Xu, Tao Xie, Xingyi He, Hujun Bao, and Xiaowei Zhou. 2023. High-fidelity and real-time novel view synthesis for dynamic scenes. In *SIGGRAPH Asia 2023 Conference Papers*. 1–9.

Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. 2022. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia 2022 Conference Papers*. 1–9.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. 2023. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. *arXiv preprint arXiv:2308.09713* (2023).

Guibo Luo, Yuesheng Zhu, Zhenyu Weng, and Zhaotian Li. 2019. A disocclusion inpainting framework for depth-based view synthesis. *IEEE transactions on pattern analysis and machine intelligence* 42, 6 (2019), 1289–1302.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)* 41, 4 (2022), 1–15.

Giljoo Nam, Joo Ho Lee, Diego Gutierrez, and Min H Kim. 2018. Practical svbrdf acquisition of 3d objects with unstructured flash photography. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–12.

Phong Nguyen-Ha, Nikolaos Sarafianos, Christoph Lassner, Janne Heikkilä, and Tony Tung. 2022. Free-viewpoint rgb-d human performance capture and rendering. In *European Conference on Computer Vision*. Springer, 473–491.

Hao Ouyang, Bo Zhang, Pan Zhang, Hao Yang, Jiaolong Yang, Dong Chen, Qifeng Chen, and Fang Wen. 2022. Real-time neural character rendering with pose-guided multiplane images. In *European Conference on Computer Vision*. Springer, 192–209.

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

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

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

Ruizhi Shao, Zerong Zheng, Hanzhang Tu, Boning Liu, Hongwen Zhang, and Yebin Liu. 2023. 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*. 16632–16642.

Qing Shuai, Chen Geng, Qi Fang, Sida Peng, Wenhao Shen, Xiaowei Zhou, and Hujun Bao. 2022. Novel view synthesis of human interactions from sparse multi-view

videos. In *ACM SIGGRAPH 2022 Conference Proceedings*. 1–10.

Liangchen Song, Anpei Chen, Zhong Li, Zhang Chen, Lele Chen, Junsong Yuan, Yi Xu, and Andreas Geiger. 2023. Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics* 29, 5 (2023), 2732–2742.

Zachary Teed and Jia Deng. 2020. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 402–419.

Richard Tucker and Noah Snavely. 2020. Single-view view synthesis with multiplane images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 551–560.

Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. 2021. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4690–4699.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 2023. 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. *arXiv e-prints* (2023), arXiv–2310.

Minye Wu, Yuehao Wang, Qiang Hu, and Jingyi Yu. 2020. Multi-view neural human rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1682–1691.

Rui Xia, Yue Dong, Pieter Peers, and Xin Tong. 2016. Recovering shape and spatially-varying surface reflectance under unknown illumination. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 1–12.

Tianyi Xie, Zeshun Zong, Yuxin Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. 2023. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. *arXiv preprint arXiv:2311.12198* (2023).

Zhen Xu, Sida Peng, Haotong Lin, Guangzhao He, Jiaming Sun, Yujun Shen, Hujun Bao, and Xiaowei Zhou. 2023. 4k4d: Real-time 4d view synthesis at 4k resolution. *arXiv preprint arXiv:2310.11448* (2023).

Jae Shin Yoon, Kihwan Kim, Orazio Gallo, Hyun Soo Park, and Jan Kautz. 2020. Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5336–5345.

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

Jiahui Zhang, Fangneng Zhan, Rongliang Wu, Yingchen Yu, Wenqing Zhang, Bai Song, Xiaoqin Zhang, and Shijian Lu. 2022. Vmrf: View matching neural radiance fields. In *Proceedings of the 30th ACM International Conference on Multimedia*. 6579–6587.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.

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

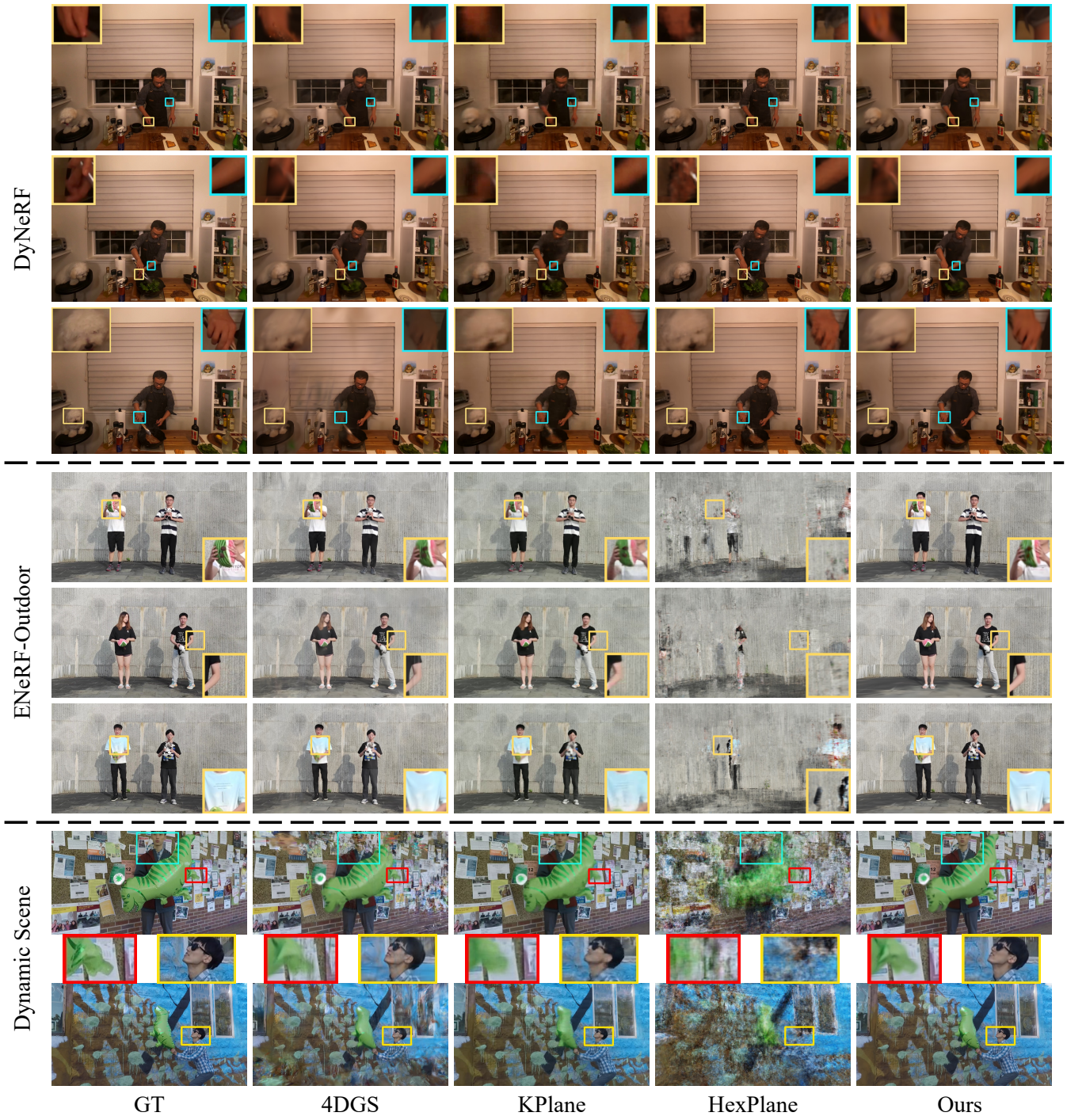

Figure 6: Some qualitative comparison results between our ST-4DGS and other previous approaches on different datasets. (Best viewed with zoom-in.)

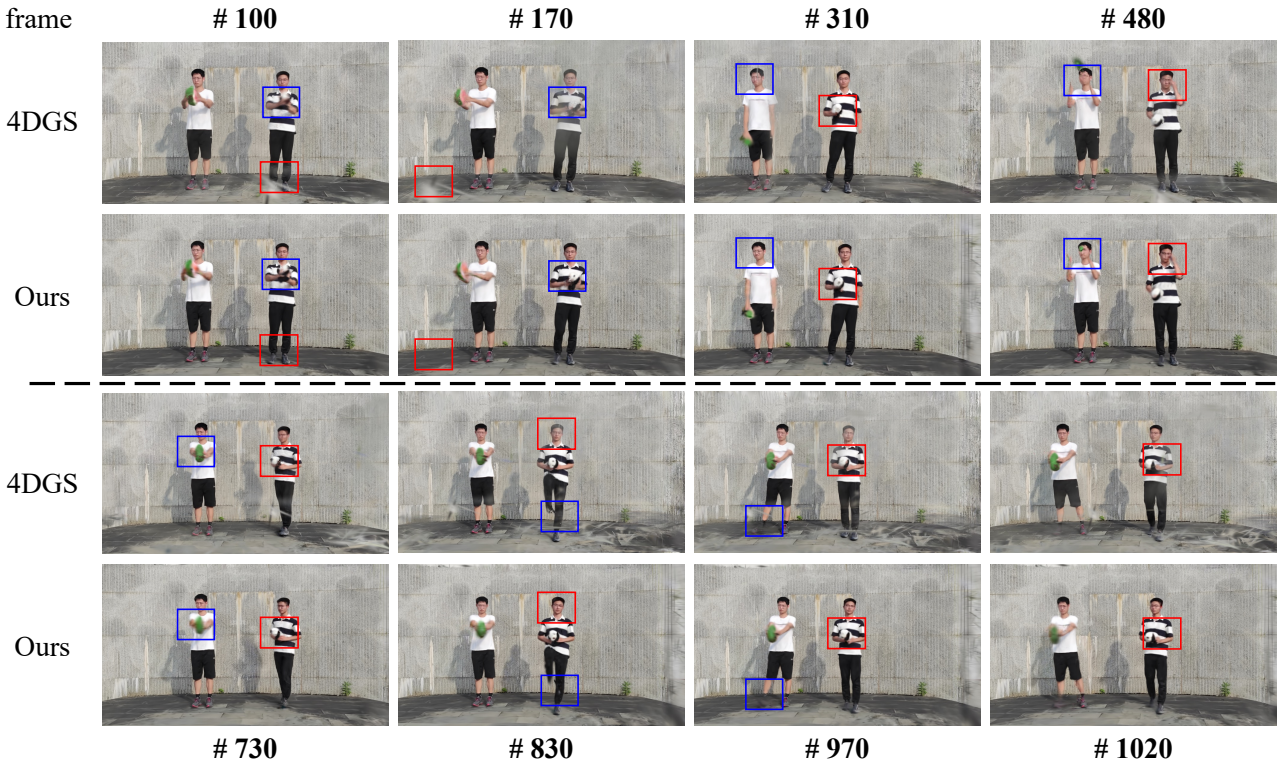

Figure 7: Some qualitative comparison results between our ST-4DGS and 4DGS in a long sequence of ENeRF-Outdoor (1200-frame), with rendering at free viewpoints.

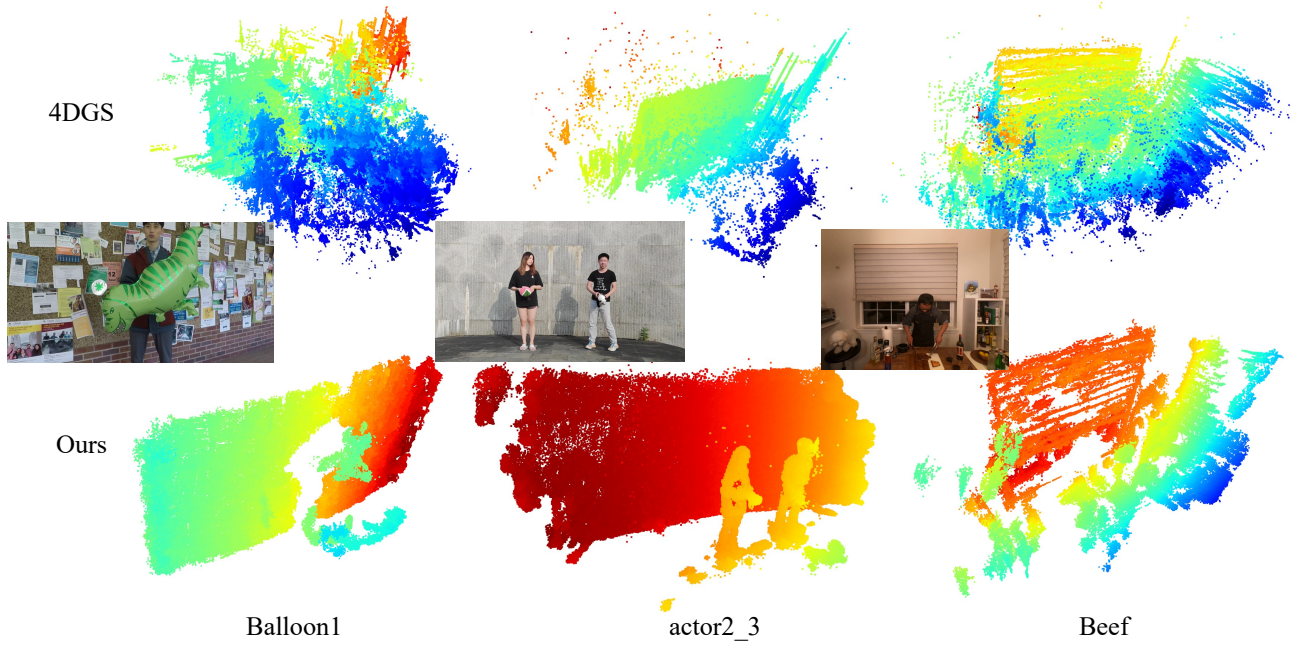

Figure 8: Several visual comparison results for the 4D Gaussians learned by our ST-4DGS and 4DGS, where points colored blue are Gaussian floaters.