Rigidity-Aware 3D Gaussian Deformation from a Single Image

JINHYEOK KIM, Ulsan National Institute of Science and Technology, Republic of Korea
JAEHUN BANG, Ulsan National Institute of Science and Technology, Republic of Korea
SEUNGHYUN SEO, Ulsan National Institute of Science and Technology, Republic of Korea
KYUNGDON JOO, Ulsan National Institute of Science and Technology, Republic of Korea

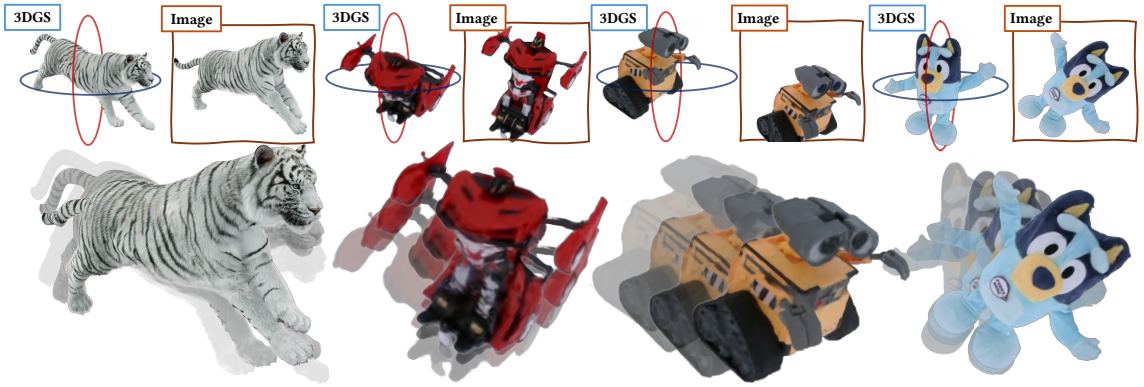

Fig. 1. *Overview of our task.* Given a single target image and an initial 3D Gaussian, `DeformSplat` deforms the Gaussian to match the target image while preserving geometry. The motion is represented by varying the transparency of the images over time.

Reconstructing object deformation from a single image remains a significant challenge in computer vision and graphics. Existing methods typically rely on multi-view video to recover deformation, limiting their applicability under constrained scenarios. To address this, we propose `DeformSplat`, a novel framework that effectively guides 3D Gaussian deformation from only a single image. Our method introduces two main technical contributions. First, we present Gaussian-to-Pixel Matching which bridges the domain gap between 3D Gaussian representations and 2D pixel observations. This enables robust deformation guidance from sparse visual cues. Second, we propose Rigid Part Segmentation consisting of initialization and refinement. This segmentation explicitly identifies rigid regions, crucial for maintaining geometric coherence during deformation. By combining these two techniques, our approach can reconstruct consistent deformations from a single image. Extensive experiments demonstrate that our approach significantly outperforms existing methods and naturally extends to various applications, such as frame interpolation and interactive object manipulation. Project page : https://vision3d-lab.github.io/deformsplat

† Corresponding author.
Authors' Contact Information: Jinhyeok Kim, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea, jinhyeok@unist.ac.kr; Jaehun Bang, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea, devappendcbangj@unist.ac.kr; Seunghyun Seo, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea, gogogo0312@unist.ac.kr; Kyungdon Joo, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea, kdjoo369@gmail.com.

SA Conference Papers '25, Hong Kong, Hong Kong

ACM ISBN 979-8-4007-2137-3/25/12
https://doi.org/10.1145/3757377.3763937

CCS Concepts: • **Computing methodologies** → *Rendering*; **Shape modeling**; *Shape analysis.*

Additional Key Words and Phrases: Deformation, Dynamic, Reconstruction, Gaussian Splatting, Single image

ACM Reference Format:
Jinhyeok Kim, Jaehun Bang, Seunghyun Seo, and Kyungdon Joo. 2025. Rigidity-Aware 3D Gaussian Deformation from a Single Image. In *SIGGRAPH Asia 2025 Conference Papers (SA Conference Papers '25), December 15–18, 2025, Hong Kong, Hong Kong.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3757377.3763937

1 Introduction

Reconstructing object deformation from visual data is essential for creating realistic and immersive content in various media fields, such as virtual reality (VR), film, and gaming [Shuai et al. 2022]. As part of this effort, recent works have explored photorealistic rendering of deformable objects, aiming to better capture their appearance and motion over time [Lu et al. 2024a; Wu et al. 2024b]. Although these methods have demonstrated strong capabilities for dynamic scene reconstruction, they often rely on multi-view or temporally continuous video data. Such data can be difficult to capture in real-world settings, which limits their practical applicability. This motivates the development of methods that reconstruct deformation from minimal visual inputs.

To achieve high-quality and efficient reconstruction of deformable objects, recent research has focused on scene representations that support both photorealism and editability. Among these, 3D Gaussian Splatting (3DGS) [Kerbl et al. 2023] has gained attention for its high-quality rendering and fast inference. Unlike implicit neural representations, such as Neural Radiance Fields (NeRF) [Mildenhall

et al. 2021], 3DGS explicitly models geometry, making it both interpretable and easy to manipulate. Motivated by these advantages, several subsequent works have explored Gaussian-based scene editing. For example, several approaches leveraging diffusion models enable intuitive editing guided by text prompts [Wu et al. 2024a] or reference images [Mei et al. 2024]. While these approaches enable intuitive edits, diffusion models often produce inconsistent results, as ambiguous text prompts lead to varied interpretations and limited geometric control. Another recent method, GESI [Luo et al. 2024], aims to address detailed geometric editing directly using a single reference image. However, it encounters difficulties in preserving intricate geometry under long-range deformations, often altering the original structure significantly. These limitations motivate our key research question: *Can 3D Gaussians be deformed from a single image while preserving the original geometry?*

In this work, we aim to deform a pre-reconstructed 3D Gaussian representation using only a single target image depicting a deformation, as illustrated in Fig. 1. Our setting is challenging, as we aim to deform 3D Gaussians using only a single RGB image, unlike conventional methods that rely on richer inputs such as multiple views or video sequences. The absence of depth and camera pose information further complicates the deformation process in our case. This constrained setting gives rise to two key challenges. The first challenge is determining how and in which direction the Gaussians should deform when only a single viewpoint is available. This is difficult because a single image provides only partial observations of the 3D structure, making it hard to infer meaningful deformation cues. To extract meaningful deformation cues under this constraint, it is essential to establish reliable correspondences between the 2D image and the 3D Gaussians. The second challenge is to prevent overfitting to the single input image, which can result in unwanted geometric distortions due to the lack of depth or multi-view constraints. Thus, preserving original geometry is crucial, especially in rigid regions that should remain unchanged during deformation.

To address these challenges, we propose a novel framework called `DeformSplat`, consisting of two main components: Gaussian-to-Pixel Matching and Rigid Part Segmentation. Gaussian-to-Pixel Matching aims to guide the deformation by linking visually similar regions between the 3D Gaussians and the target image. Specifically, we render multi-view images from the 3D Gaussians and compute pixel-wise correspondences between each rendered image and the target image using an image matcher. Based on the pixel-to-pixel matching, we select the viewpoint with the largest visual overlap, and translate its correspondences into Gaussian-to-Pixel mappings. This step provides an essential basis for deformation, enabling the Gaussians to reflect the geometry depicted in the target image.

To further ensure geometric consistency, we propose Rigid Part Segmentation that explicitly identifies rigid regions within the Gaussian representation. To achieve this, our method first initializes rigid groups based on Gaussian-to-Pixel correspondences and then iteratively refines these groups during optimization. The segmentation is used in a rigidity-aware optimization that regularizes rigid and non-rigid regions differently to preserve geometry. Consequently, we achieve superior performance than previous SOTA and generalize to applications such as frame interpolation and interactive object manipulation.

Our contributions can be summarized as follows:

- We propose `DeformSplat`, a novel framework for rigidity-aware deformation of 3D Gaussians using only a single target image.
- We present Gaussian-to-Pixel Matching strategy that connects the 3D Gaussian representation with the 2D target image to guide deformation.
- We propose Rigid Part Segmentation method, which preserves the original geometry by detecting rigid regions and constraining their deformation.
- Our method shows superior performance in single image Gaussian deformation and extends to applications such as frame interpolation and interactive manipulation.

2 Related Work

2.1 Dynamic Reconstruction

Dynamic reconstruction is the task of recovering time-varying 3D geometry, including motion and non-rigid deformations, in real-world. One influential line of work [Guo et al. 2023; Li et al. 2022; Park et al. 2021, 2023; Pumarola et al. 2021] is based on NeRF [Mildenhall et al. 2021]. D-NeRF [Pumarola et al. 2021] is a representative extension of NeRF that introduces time as an additional input, enabling dynamic reconstruction and partial modeling of non-rigid motion. However, its MLP-based implicit representation results in slow processing and limited control over localized dynamics.

Recent works [Huang et al. 2024; Lu et al. 2024a; Luiten et al. 2024; Wu et al. 2024b; Yang et al. 2024, 2023] have focused on 3DGS [Kerbl et al. 2023], an explicit representation using 3D Gaussians that enables fast training, real-time For instance, 4D-GS [Wu et al. 2024b] deforms a fixed set of canonical Gaussians over time via a learned deformation field, enabling real-time rendering. SC-GS [Huang et al. 2024] controls motion using a small number of sparse control points, allowing efficient and editable deformation with fewer parameters. These methods leverage the strengths of explicit representations in terms of editability and computational speed. However, their reliance on continuous multi-view video limits their practicality in real-world settings, in which such data is often difficult to obtain.

To mitigate the difficulty of real-world data acquisition, recent work has investigated few-view and single-view dynamic reconstruction. Few-view methods such as NPG [Das et al. 2024] and MAGS [Guo et al. 2024] adopt low-rank bases or optical flow to better capture motion under limited viewpoints. In the single-view case, approaches like Shape of Motion [Wang et al. 2024] and MoSCA [Lei et al. 2025] leverage priors such as depth and tracking models, while CUT3R [Wang et al. 2025] reconstructs camera pose and dynamics in a feed-forward manner. More recent methods, including MegaSAM [Li et al. 2025] and ViPE [Huang et al. 2025], combine video depth prediction with bundle adjustment for fast dynamic reconstruction. However, one major drawback of these methods is that they depend on continuous video. In particular, limited frame-to-frame continuity, such as with low or inconsistent frame rates, leads to a significant decline in reconstruction performance. These challenges collectively point to the need for a framework that can explicitly perform 3D Gaussian deformation from a single image.

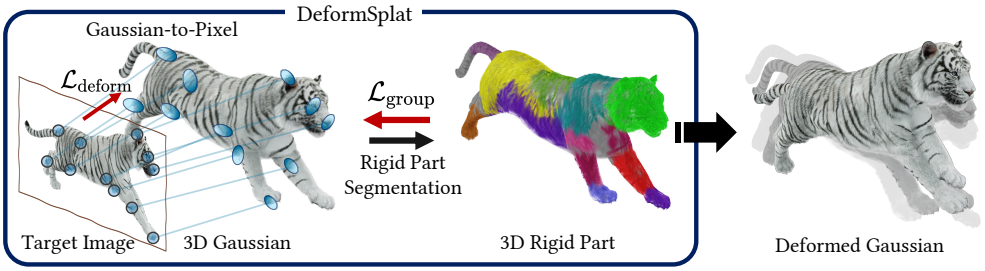

Fig. 2. *Overview of* `DeformSplat`. It first establishes correspondences between the 2D target image and 3D Gaussian. Then, Rigid Part Segmentation explicitly identifies rigid regions. By combining two methods, our approach reconstructs the deformed Gaussian, which can be rendered from novel views.

2.2 3D Gaussian Editing

3D editing refers to the interactive modification of 3D representations based on user input. Recent research highlights 3DGS for enabling fast, photorealistic rendering and intuitive editing through its explicit representation.

Text-driven 3D editing has emerged as a promising direction within 3D Gaussian editing, where user-provided text prompts are used to modify 3DGS representations. Recent works, such as GaussCtrl [Wu et al. 2024a], GaussianEditor [Chen et al. 2024], GSEdit [Palandra et al. 2024], and GSEditPro [Sun et al. 2024b], utilize 2D diffusion models conditioned on text prompts to modify the appearance of 3D Gaussians. To enhance editing fidelity and usability, these methods incorporate additional techniques, such as depth-aware consistency [Wu et al. 2024a], semantic region tracking [Chen et al. 2024], fast object-level modification [Palandra et al. 2024], and attention-guided localization [Sun et al. 2024b]. This research direction enables intuitive and diverse editing without requiring expert skills. However, the ambiguity of natural language can lead to varied interpretations, causing inconsistent results due to the limitations of 2D diffusion models.

Complementary to text-based methods, image-driven 3D editing enables intuitive modification of 3DGS through visual inputs, allowing users to express their intent more clearly. Representative works, such as ReGS [Mei et al. 2024], ICE-G [Jaganathan et al. 2024], and ZDySS [Saroha et al. 2025], enable appearance editing based on reference images. Specifically, they address texture underfitting [Mei et al. 2024], enable localized appearance transfer [Jaganathan et al. 2024], and support zero-shot stylization [Saroha et al. 2025]. These works allow intuitive editing by directly linking 2D inputs to 3D output, but current methods focus only on appearance, highlighting the need for geometry-level control. GESI [Luo et al. 2024] addresses this by modifying 3D Gaussians based on a reference image and camera pose, following a principle of "what you see is what you get". However, the lack of explicit separation between rigid and deformable regions makes it difficult to preserve structural integrity and geometric consistency during deformation. To address previous limitations, we propose a single-image deformation framework that preserves the object's structural integrity by explicitly separating rigid and non-rigid components. This enables stable dynamic reconstruction without relying on multi-view or video input, making it practical for real-world use.

3 Method

3.1 Task Overview

Given a pre-reconstructed 3D Gaussian and a single target image depicting a deformed object, our goal is to deform the 3D Gaussian to accurately match the deformation observed in the target image. At the same time, we aim to preserve the original geometry of the initial 3D Gaussian. An overview of our method is shown in Fig. 2. Formally, let the 3D Gaussian $\mathcal{G} = \{\mu_i, q_i, s_i, \alpha_i, sh_i\}$ denote a set of Gaussians, each defined by its center position $\mu_i \in \mathbb{R}^3$, quaternion $q_i \in \mathbb{R}^4$, scale $s_i \in \mathbb{R}^3$, opacity $\alpha_i \in \mathbb{R}^1$, and spherical harmonic coefficients $sh_i \in \mathbb{R}^{48}$. This Gaussian representation is initially reconstructed from multiple views captured at an earlier time step. The target deformation is provided as a single RGB image $\mathcal{I}_{\text{target}}$, without any explicit 3D information, such as depth or camera pose. To efficiently find the underlying deformation, we only optimize location μ_i and rotations q_i in order to align with the deformation depicted in the target image $\mathcal{I}_{\text{target}}$.

Our task is particularly difficult due to limited input conditions. Conventional dynamic reconstruction approaches [Huang et al. 2024; Lu et al. 2024a; Wu et al. 2024b] typically rely on abundant multi-view images or continuous temporal data to robustly model deformation. In contrast, our method is restricted to supervision from just a single image, complicating accurate deformation guidance. Furthermore, even though multiple camera poses are known from the initial Gaussian reconstruction, the exact viewpoint corresponding to the target image remains unknown. This ambiguity poses additional difficulties in precisely aligning the Gaussian with the observed 2D deformation.

Under these constraints, we face two significant challenges. First, it is difficult to determine how each Gaussian should deform from only a single image, since this requires reliable correspondences between 3D Gaussians and 2D image. Second, without depth or multi-view constraints, deformation can easily overfit the target image and cause distortions, even in rigid regions that should remain unchanged. To address these challenges, we propose two key components: (1) Gaussian-to-Pixel Matching that selects the most overlapping viewpoint and establishes 3D-to-2D correspondences for deformation guidance, and (2) Rigid Part Segmentation that detects and preserves rigid regions through initialization and refinement. Together, these components enable accurate single-image Gaussian deformation while maintaining geometric consistency.

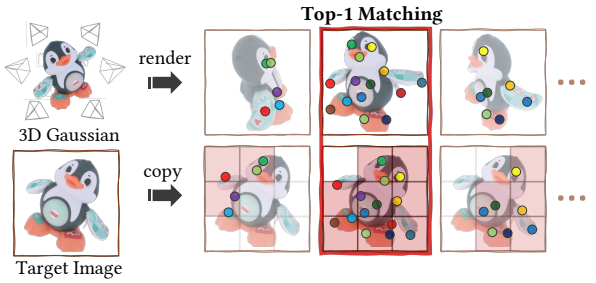

Fig. 3. *High-overlap image selection.* We render images from multiple cameras used initial Gaussian reconstruction. Each rendered image is matched with the target image to measure the overlap.

3.2 Gaussian-to-Pixel Matching

Given a reconstructed 3D Gaussian and an unposed target image, a naive approach is to randomly select a camera viewpoint from those used in the initial Gaussian reconstruction. Then, the Gaussian can be rendered and optimized using pixel-wise loss. However, this strategy often fails. The random view may have minimal visual overlap with the target image, making pixel-wise guidance ineffective. Even with a large overlap, pixel-wise loss alone cannot accurately handle long-range deformation because the target image represents a deformed shape compared to the initial Gaussian. Thus, a more precise method is required to guide deformation robustly.

To address this challenge, we propose Gaussian-to-Pixel Matching approach. We start by rendering multiple images from the original set of camera viewpoints used for the initial Gaussian reconstruction. We then apply an image matcher, RoMA [Edstedt et al. 2024], between each rendered image and the target image, obtaining corresponding pixel pairs (x_p, x'_p), where x_p represents pixels from rendered images and x'_p represents corresponding pixels in the target image. In order to select the best viewpoint, we partition the target image into evenly spaced grids. For each rendered viewpoint, we count how many of these grids contain matched pixels x'_p. We then select the camera viewpoint with the maximum number of matched grids, ensuring visual alignment with the target deformation. Fig. 3 illustrates this selection procedure, clearly depicting the manner in which visual overlap across multiple camera viewpoints can be quantified.

After selecting the viewpoint, we convert the pixel-to-pixel correspondences into Gaussian-to-Pixel correspondences. Specifically, we first evaluate the visibility of Gaussians using alpha-blended opacity $\alpha_i \prod(1 - \alpha_j)$ to exclude invisible Gaussians from the selected viewpoint. Among visible Gaussians, we associate each matched pixel x_p with the nearest projected Gaussian center $\mu_i^{2D} = P\mu_i$, where P denotes the camera projection matrix. Pixels sufficiently close to Gaussian projections are then replaced by the corresponding Gaussian centers μ_i, establishing Gaussian-to-Pixel correspondences (μ_i, x'_p). The derived 3D-to-2D matches inherently capture the necessary directional information for guiding the Gaussians' movements. Leveraging this matching, we effectively guide the deformation process at a detailed level.

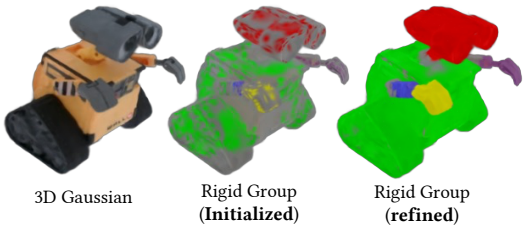

Fig. 4. *Example of Initialization and Refinement.* Rigid groups are initialized from Gaussian-to-Pixel correspondences (μ_i, x'_p). Then, refinement step further expands these groups to cover broader regions. Each color denotes an independent rigid group, while grey indicates ungrouped Gaussians.

3.3 Rigid Part Segmentation

Although Gaussian-to-Pixel Matching strategy provides effective guidance for Gaussian deformation, it does not inherently guarantee preservation of the original geometry. To preserve the geometric of the reconstructed Gaussian during deformation, we propose two-stage rigid segmentation composed of a region-growing initialization step followed by a refinement step.

Rigid regions typically share two key properties: (1) they undergo the same rigid transformations, and (2) they exhibit strong spatial connection. The proposed rigid segmentation leverages these properties to robustly identify coherent rigid regions, significantly enhancing geometry preservation during deformation.

Rigid Part Initialization. In the rigid initialization stage, we first utilize Perspective-n-Point (PnP) algorithm [Lepetit et al. 2009], which estimates a rigid transformation from 3D-to-2D correspondences. To identify subset Gaussians sharing similar rigid transformations, we combine the PnP algorithm with RANSAC [Fischler and Bolles 1981]. Specifically, PnP estimates rigid transformations from Gaussian-to-Pixel correspondences, while RANSAC robustly selects the most consistent subset of correspondences. For simplicity, we refer to this combination as PnP-RANSAC. Given Gaussian-to-Pixel correspondences (μ_i, x'_p) obtained previously, PnP-RANSAC identifies subsets of Gaussians sharing similar rigid transformations. However, although sharing similar transformations is a necessary condition for rigid grouping, it alone does not guarantee spatial coherence among Gaussians. A meaningful rigid group should consist of Gaussians that are spatially connected. If spatial connectivity is not enforced, distant and unrelated Gaussians may coincidentally share similar transformations. For instance, left and right hands might exhibit similar transformations by chance, yet they clearly should not belong to the same rigid region.

To guarantee spatial connectivity, we propose a region-growing strategy for rigid group initialization. We begin this process by forming an initial rigid group G from a randomly selected single Gaussian. This group is then iteratively expanded to include neighboring Gaussians using ball queries. After each expansion, we apply PnP-RANSAC to identify inliers G_{inlier} sharing a consistent rigid transformation. This iterative expansion and filtering continue until convergence, ensuring the rigid groups are spatially coherent and transformation-consistent. For detailed rigid initialization, please refer to the supplementary material.

While the rigid initialization produces spatially coherent groups, its scope is restricted to Gaussians derived from Gaussian-to-Pixel correspondences. This means that only small subsets of the rigid parts are actually initialized, as shown in Fig. 4. To complete the rigid segmentation, we introduce a refinement stage. This process extends the segmentation to most of the Gaussians.

Rigid Part Refinement. The refinement stage leverages the same characteristics of rigid regions: consistent rigid transformations and spatial connectivity. However, the key difference from the initialization is that refinement is independent of Gaussian-to-Pixel Matching and utilizes continuously updated Gaussian parameters (positions μ_i' and rotations q_i') obtained during optimization. Note that the refinement is an iterative process performed jointly with rigid-aware optimization (cf. Sec. 3.4).

In particular, we iteratively refine initially identified rigid group by enlarging it with neighboring Gaussians found via local ball queries. To evaluate whether newly added candidate Gaussians adhere consistently to the group transformation, we propose a rigidity score inspired by the As-Rigid-As-Possible (ARAP) principle [Sorkine and Alexa 2007]. Given a candidate Gaussian μ_i, the rigidity score relative to an existing rigid group G is computed as:

$$S_{\text{rigid}}(\mu_i, G) = \frac{1}{|G|} \sum_{\mu_j \in G} ||R_i^{-1}(\mu_i - \mu_j) - R_i'^{-1}(\mu_i' - \mu_j')||^2, \quad (1)$$

where (μ_i, q_i) and (μ_j', q_j') denote the initial and optimized Gaussian positions and rotations, respectively. R_i and R_i' are the rotation matrices derived from quaternion q_i and q_i', respectively. $|G|$ is the number of Gaussians in the group. A small rigidity score indicates strong consistency, thus justifying the inclusion of the candidate Gaussian into the rigid group.

During each iteration, we first expand the current rigid group G by identifying candidate Gaussians $\mu_i \in G_{\text{expand}}$ within a local ball-query radius. Each candidate Gaussian is evaluated based on its rigidity score $S_{\text{rigid}}(\mu_i, G)$. Candidates with rigidity scores below a lower threshold τ_{low} are included in the rigid group G, whereas those exceeding an upper threshold τ_{high} are excluded if previously part of the group. Through iterative inclusion and exclusion, this refinement procedure progressively corrects and expands rigid groups, ensuring robust geometry preservation throughout deformation optimization. For the detailed procedure, please refer to the supplementary material.

By the combined rigid initialization and refinement steps, our Rigid Part Segmentation robustly identifies spatially coherent rigid regions, ensuring strong geometric consistency throughout the deformation process.

3.4 Rigid-Aware Optimization

Directly optimizing Gaussian parameters independently often leads to excessive flexibility, potentially disrupting the deformation quality. To effectively mitigate this issue, we adopt an anchor-based deformation representation following previous works [Huang et al. 2024; Sumner et al. 2007]. Specifically, the deformation is represented using a sparse set of anchors, each parameterized by its position $a_k \in \mathbb{R}^3$, rotation $R_k^a \in SO(3)$ (equivalently as quaternion q_k^a), and translation $T_k \in \mathbb{R}^3$. Anchor positions are initialized by voxelizing the space and computing the average Gaussian positions within each voxel. Using these anchors, updated Gaussian positions μ_i' and rotations q_i' are computed by interpolating transformations of neighboring anchors $\mathcal{N}$ as follows:

$$\mu_i' = \sum_{k \in \mathcal{N}} w_{ik} \left(R_k^a(\mu_i - a_k) + a_k + T_k \right), \quad q_i' = q_i \otimes \sum_{k \in \mathcal{N}} w_{ik} q_k^a, \quad (2)$$

where $\otimes$ is the production operation of quaternions and w_{ik} is interpolation weight inversely proportional to distances between Gaussian μ_i and anchors a_k. This sparse anchor representation significantly reduces deformation complexity, promoting smoothness and geometric coherence through localized transformations.

Deformation Loss. Using the Gaussian-to-Pixel correspondences, we define a deformation loss as follows:

$$\mathcal{L}_{\text{deform}} = \sum_i ||\mu_i^{2D} - x_p'||^2, \quad (3)$$

which encourages Gaussian centers to move toward matched pixel locations. This approach effectively guides the deformation based on structurally meaningful matches.

Rigid Group Regularization. Using the rigid groups from Sec. 3.3, we introduce a group-based rigidity loss $\mathcal{L}_{\text{group}}$ to explicitly preserve geometric consistency within rigid regions. Specifically, within each rigid group G, we enforce consistency between the original and transformed Gaussian structures as:

$$\mathcal{L}_{\text{group}} = \sum_{G_k} \sum_{\mu_i, \mu_j \in G_k} ||R_i^{-1}(\mu_i - \mu_j) - R_i'^{-1}(\mu_i' - \mu_j')||^2, \quad (4)$$

where R_i and R_i' denote rotation matrices derived from original and updated Gaussian rotations q_i and q_i', respectively.

ARAP Regularization. While $\mathcal{L}_{\text{group}}$ explicitly preserves geometry within rigid regions, non-rigid regions also require regularization to ensure smooth and natural deformation. To achieve this, we apply ARAP regularization between neighboring anchors as follows:

$$\mathcal{L}_{\text{arap}} = \sum_{a_i} \sum_{k \in \mathcal{N}} ||R_i^a(a_i - a_k) - (a_i' - a_k')||^2, \quad (5)$$

where $a_i' = a_i + T_i$ is the updated anchor position, and R_i^a is the rotation at anchor a_i. This ARAP loss promotes local rigidity among anchors, resulting in coherent deformation transitions.

RGB Loss. To ensure visual alignment with the target deformation, we employ a photometric RGB loss aligning the rendered image $\mathcal{I}_{\text{render}}$ with the target image $\mathcal{I}_{\text{target}}$:

$$\mathcal{L}_{\text{rgb}} = ||\mathcal{I}_{\text{render}} - \mathcal{I}_{\text{target}}||^2. \quad (6)$$

Total Optimization Loss. Combining these terms, we obtain our total optimization objective:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{deform}} \mathcal{L}_{\text{deform}} + \lambda_{\text{group}} \mathcal{L}_{\text{group}} + \lambda_{\text{arap}} \mathcal{L}_{\text{arap}} + \lambda_{\text{rgb}} \mathcal{L}_{\text{rgb}}. \quad (7)$$

Here, each λ denotes a hyperparameter that balances the corresponding loss term. The unified optimization scheme simultaneously guides accurate deformation, preserves rigid region geometry, and ensures visual consistency.

Table 1. *Quantitative result on the Diva360 and DFA datasets.* Best results are indicated in **bold** and second-best results are underlined.

Method	Diva360			DFA		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3DGS [Kerbl et al. 2023]	21.28	0.900	0.098	19.48	0.866	0.118
DROT [Xing et al. 2022]	21.08	0.914	0.086	17.64	0.872	0.119
SC-GS [Huang et al. 2024]	22.20	0.910	0.097	19.49	0.867	0.116
4DGS [Wu et al. 2024b]	19.93	0.913	0.100	14.56	0.856	0.204
3DGStream [Wu et al. 2024b]	22.57	0.928	0.088	20.16	0.886	0.100
GESI [Luo et al. 2024]	22.71	0.897	0.086	18.54	0.876	0.127
GESI (μ, q) [Luo et al. 2024]	22.53	0.924	0.078	20.05	0.888	0.100
Ours	**26.84**	**0.955**	**0.050**	**21.81**	**0.897**	**0.091**

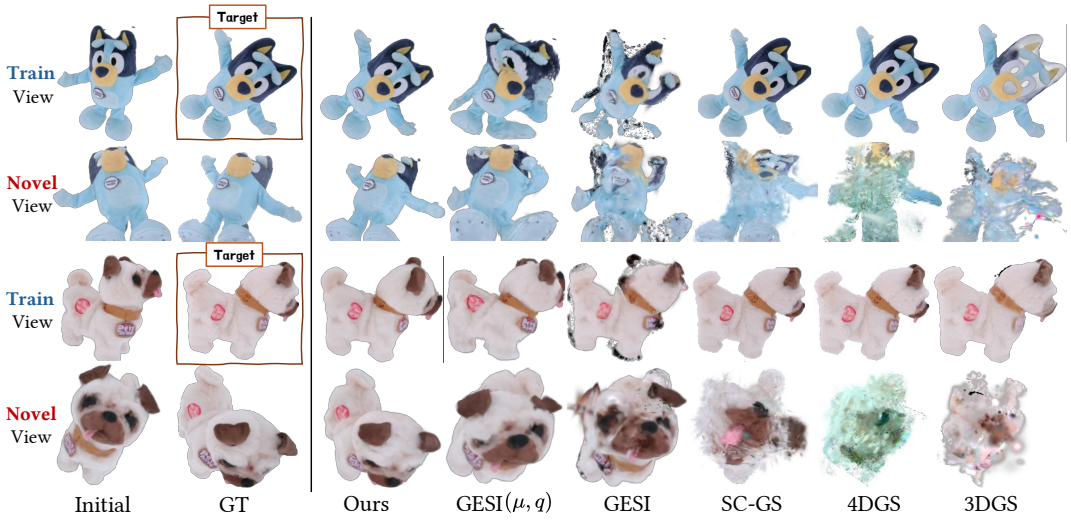

Fig. 5. *Qualitative comparison on the Diva360 dataset.* The target images are highlighted with brown boxes in the second column.

Smooth Motion Interpolation. After optimization, we further introduce post-processing for smooth interpolation. Specifically, we define an interpolation loss as follows:

$$\mathcal{L}_{\text{inter}} = \sum_i \left(||R_i^a - \hat{R}_i^a|| + ||T_i^a - \hat{T}_i^a|| \right) + \lambda_{\text{inter}} (\mathcal{L}_{\text{group}} + \mathcal{L}_{\text{arap}}), \tag{8}$$

where (R_i^a, T_i^a) and $(\hat{R}_i^a, \hat{T}_i^a)$ denote initial and optimized anchor transformations. λ_{inter} is decaying hyperparameter to ensure convergence. As (R_i^a, T_i^a) gradually approaches $(\hat{R}_i^a, \hat{T}_i^a)$, we achieve smooth motion transitions that preserve geometric consistency, resulting in visually pleasing deformation outcomes.

4 Experiment

4.1 Experiment Setting

Datasets. We evaluate ours on two multi-view video datasets: diverse moving object sequences in the Diva360 dataset [Lu et al. 2024b] and the synthetic Dynamic Furry Animal (DFA) dataset [Luo et al. 2022]. The Diva360 dataset captures various dynamic objects from multiple views in a 360° configuration and comprises 21 sequences. Among them, we exclude two "Plasma Ball" sequences since they show only light changes and do not exhibit any deformation. The synthetic DFA dataset, generated from motion capture data, includes 25 sequences depicting animated animal movements.

For each of the N video sequences in each dataset, we select two distinct timesteps. For the first timestep, we select a moment where the object is fully visible without occlusions. This enables accurate initial reconstruction of the Gaussian model using images from multiple views. The second timestep, chosen to represent the target deformation state, contains noticeable deformation. The target deformation is supervised using only a single viewpoint image from the second timestep. The remaining viewpoint images from the second timestep are used as ground-truth images for evaluation. These two timesteps are manually selected to ensure noticeable deformation, minimal occlusion, and sufficient visual consistency. The selected data samples can be founded through our released code.

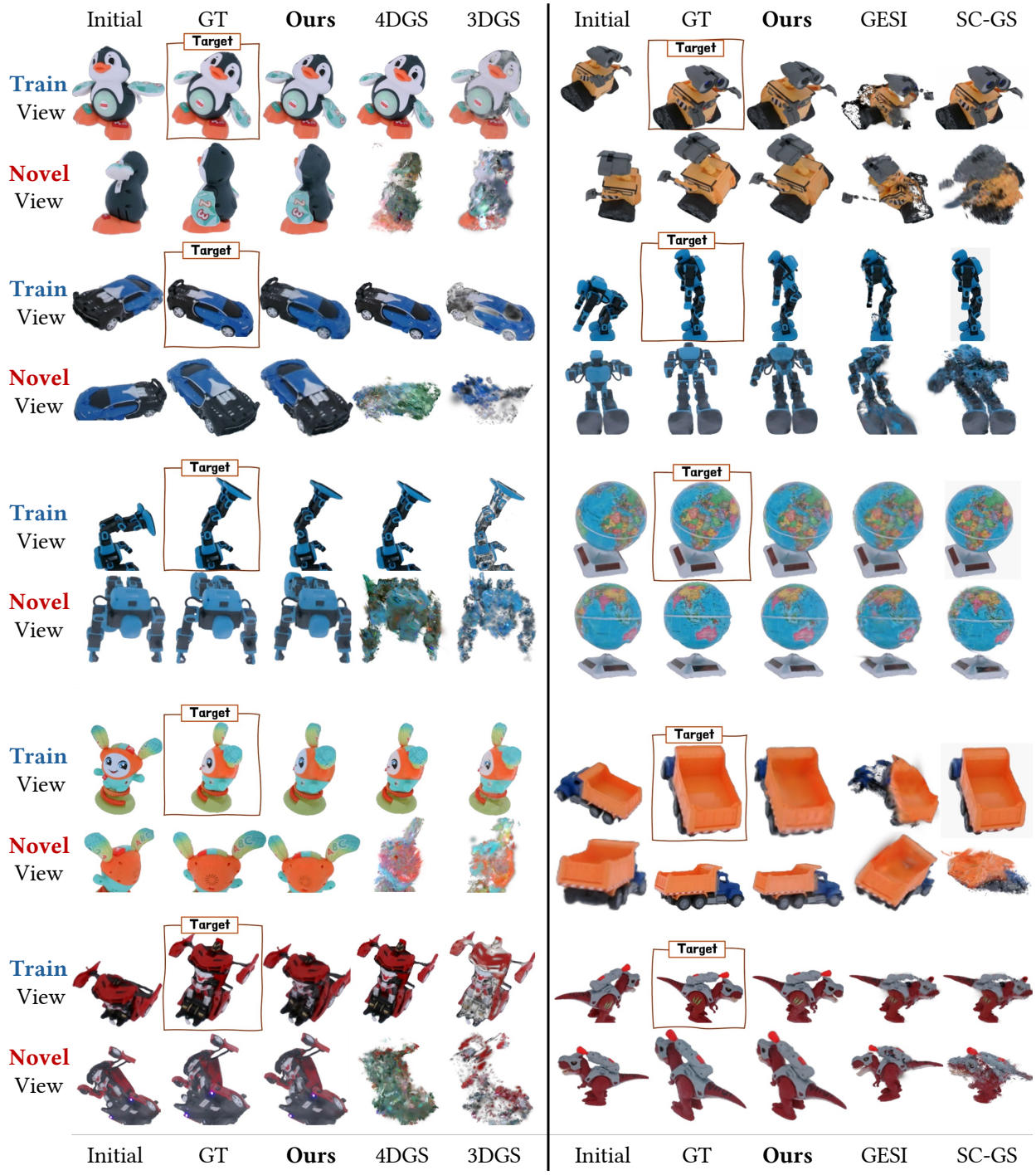

Fig. 6. *Diverse result on the Diva360 dataset.* The target images are highlighted with brown boxes in the second column.

Fig. 7. *Rigid group visualization on the DFA dataset.* Grey region refers ungrouped Gaussian.

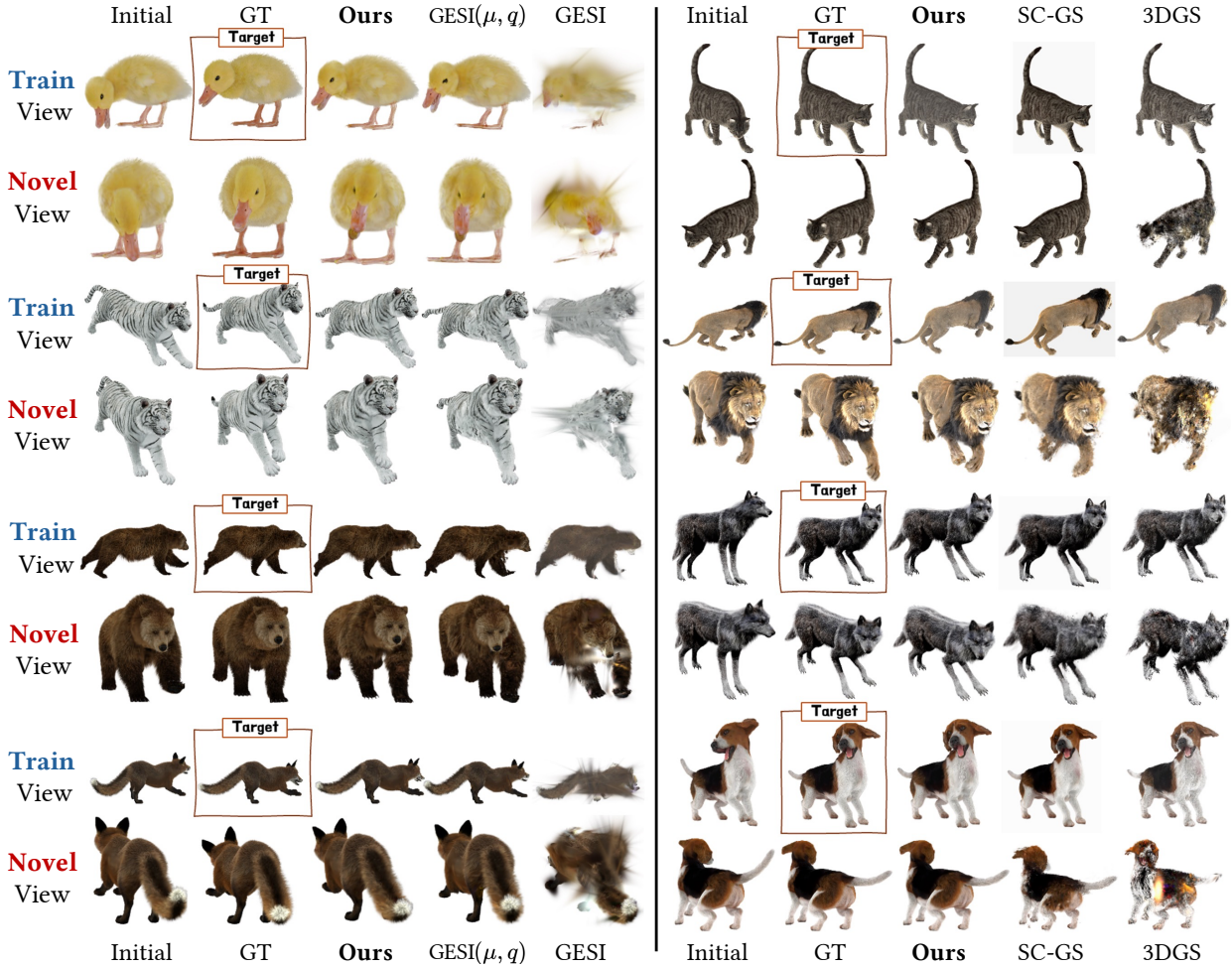

Fig. 8. *Diverse result on the DFA dataset.* The target images are highlighted with brown boxes in the second column.

Baselines. To validate our approach, we compare DeformSplat with established baseline methods. Specifically, we compare ours with 3DGS [Kerbl et al. 2023], which directly optimizes reconstructed Gaussians using pixel-wise RGB losses (L1 and SSIM) from a single target image. We also include a comparison with 3DGS that is optimized using the optical transport RGB loss, called DROT [Xing et al. 2022], which enables more robust, long-range comparisons rather than simple pixel-wise differences. Additionally, we compare against two dynamic Gaussian reconstruction methods (4DGS [Wu et al. 2024b], SC-GS [Huang et al. 2024]) and a streamable Gaussian method (3DGStream [Sun et al. 2024a]). Lastly, we evaluate GESI [Luo et al. 2024], a Gaussian editing method designed for single-image input, in two variants: one optimizing all Gaussian parameters, and another selectively optimizing only Gaussian positions μ and rotations q. The selective tuning of (μ, q) parameters aims to represent deformation more explicitly, ensuring a fairer comparison with our method. Since there is no publicly available implementation for GESI, we implement it following the details provided in the original paper.

Evaluation Metrics and Camera Alignment. We quantitatively evaluate deformation accuracy using three metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS) [Zhang et al. 2018]. Since our approach selects one camera viewpoint based on visual overlap, the camera pose might not be aligned with the global coordinate. Therefore, after optimization, the camera pose and final Gaussian are rotated and translated to align with the ground-truth camera poses. This enables fair quantitative and qualitative evaluation. All baseline methods directly use these ground-truth camera poses, ensuring consistency across comparisons.

4.2 Comparison with Baselines

Qualitative Comparison. Fig. 5, Fig. 6 and Fig. 8 present qualitative results comparing our method with the baseline approaches on the Diva360 and DFA datasets, respectively. DeformSplat successfully reconstructs accurate and visually consistent deformation from the single-target-image input, achieving high visual similarity to the ground-truth reference images. In contrast, baseline methods demonstrate notable visual artifacts and geometric distortions when rendered from viewpoints not observed during optimization. Specifically, 3DGS, 4DGS, and SC-GS overly rely on pixel-level color losses, leading to insufficiently accurate deformations. GESI, on the other hand, fails to produce accurate deformations due to occasional inaccuracies in its long-range matching via DROT.

Quantitative Comparison. Table 1 summarizes quantitative evaluations. DeformSplat significantly outperforms baseline methods, setting a new SOTA performance standard. On Diva360, DeformSplat achieves an average PSNR increase of 4.1 compared to the next best-performing baseline. On the DFA dataset, we similarly observe a PSNR improvement of 1.8. These results confirm the robust capability of DeformSplat to reconstruct detailed object deformations from minimal supervision accurately.

Regarding the performance of other baselines, GESI demonstrates the second-best performance on the Diva360 dataset, following our method, largely due to its ability to provide long-range guidance. On the DFA dataset, 3DGStream ranks just below our method, benefiting from its efficient optimization of Gaussians with fewer iterations compared to other baselines. In contrast, 4DGS shows the lowest performance in both dataset. This is because 4DGS encodes both geometry and color into the same implicit function, causing the color to change significantly when the geometry is updated during the optimization.

Table 2. *Ablation study of each component on Diva360 dataset.*

Method Variant	PSNR↑	SSIM↑	LPIPS↓
w/o L_{deform}, w/o L_{group}	21.73	0.917	0.082
w/o L_{group}	24.36	0.942	0.071
w/o region-growing initialize	25.25	0.946	0.067
w/o rigid refinement	25.79	0.949	0.061
full pipeline (ours)	**26.84**	**0.955**	**0.050**

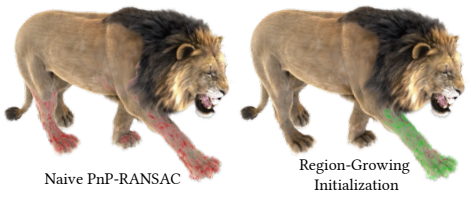

Fig. 9. *Ablation on Region-Growing Rigid Clustering.* Naive PnP-RANSAC yields a disconnected rigid group, while our region-growing method produces a spatially coherent group.

4.3 Ablation Study

We conduct an ablation study to evaluate the contribution of each component in our framework. As summarized in Table 2, removing or replacing individual modules leads to noticeable performance drops, confirming the importance of each design choice.

Deformation Loss. The absence of our deformation loss significantly degrades deformation quality. Without this structural guidance, deformation relies solely on pixel-wise color information, resulting in substantial performance degradation.

Rigid Group Loss. Removing the rigid group loss notably reduces deformation quality. This loss explicitly preserves geometry within rigid regions, highlighting its crucial role in maintaining geometry during deformation.

Region-Growing Initialization. Substituting our region-growing initialization with naive PnP-RANSAC initialization decreases deformation quality. Qualitatively, as shown in Fig. 9, naive PnP-RANSAC produces disconnected rigid groups, while our region-growing method effectively enforces spatial coherence.

Rigid Refinement. Removing the rigid refinement reduces deformation accuracy. This refinement iteratively updates rigid segmentation using optimized Gaussian parameters, correcting initial segmentation errors. Without it, the initial segmentation can be biased or have some errors.

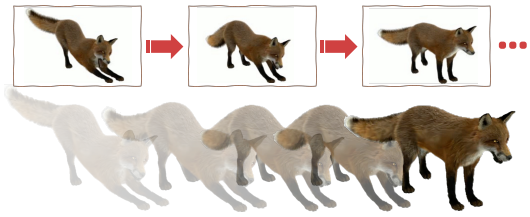

Fig. 10. *Multi-Frame Interpolation.* DeformSplat naturally extends to sequential tasks, enabling smooth frame interpolation. Please refer supplementary video for detailed examples.

Fig. 11. *Interactive Manipulation.* Our rigidity-aware group loss effectively preserves underlying geometry, facilitating more natural and intuitive interactive manipulation of 3D objects.

5 Application

5.1 Multi-Frame Interpolation

Given the capability of DeformSplat to reconstruct deformation from a single image, our framework naturally generalizes to frame interpolation for generating smooth video sequences. Specifically, when provided with an initial Gaussian reconstruction and multiple target frames, DeformSplat sequentially applies deformation in an autoregressive manner, using the deformed Gaussian from the previous frame as the input for the next.

As illustrated in Fig. 10, the resulting interpolated frames exhibit smooth transitions, accurately preserving temporal coherence and dynamic realism for various objects. These interpolation results highlight DeformSplat's capability beyond single-frame deformation, suggesting promising extensions into practical applications such as video content creation and dynamic scene generation.

5.2 Rigid-Aware Manipulation

Leveraging the explicit rigid segmentation and Gaussian-to-Pixel correspondences established by our framework, DeformSplat enables intuitive and precise interactive manipulation of 3D Gaussian objects. Specifically, users can perform direct manipulation by dragging pixels on a target image, guiding corresponding Gaussians effectively through our deformation loss. Simultaneously, our rigid group loss and ARAP regularization preserve structural integrity, ensuring natural and physically plausible transformations.

As shown in Fig. 11, user-defined manipulations produce smooth, coherent deformations. Notably, rigid regions maintain their structural integrity with minimal distortion, while adjacent non-rigid regions deform flexibly and naturally. This rigid-aware characteristic makes DeformSplat highly suitable for various interactive applications, including video content editing and object manipulation in virtual reality environments.

6 Limitations and Future Work

While DeformSplat achieves SOTA results for single image-guided 3D Gaussian deformation, it still has three key limitations that we aim to address in future work.

Robust Matching for Dynamic Object. The performance of our method is highly dependent on the quality of image matching. The image matcher we use, RoMA, is trained on static datasets, which sometimes leads to inaccurate correspondences when applied to dynamic objects. This can result in incorrect deformations. Improving the robustness of image matching for dynamic content is an important direction for future research.

Handling Fully Flexible Object. While our approach performs well on semi-rigid deformations, it struggles with highly non-rigid objects such as clothing or fluids. This limitation arises from our assumption that the target object contains rigid components. Developing alternative strategies tailored to fully flexible objects will be essential for broadening the applicability of our method.

Handling Color Change. Since DeformSplat only optimizes μ, q parameters, it does not handle color changes during deformation. When we attempted to optimize color jointly with geometry, color of 3D Gaussians tended to overfit, resulting in unrealistic appearances. Future work will focus on incorporating regularization strategies that enable consistent and natural color adaptation throughout deformation, without overfitting.

7 Conclusion

In this work, we introduced DeformSplat, a novel framework that reconstructs deformations of 3D Gaussians from a single image. Specifically, we proposed Gaussian-to-Pixel Matching that bridges two distinct data representations to accurately guide deformation. Additionally, we introduced Rigid Part Segmentation, a two-stage method consisting of initialization and refinement, designed to preserve original geometric structures. These techniques effectively handle long-range deformation and geometric preserving inherent in single image deformation. The experiment shows that our method significantly outperforms existing approaches and extends to diverse applications.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2025-25442149, LG AI STAR Talent Development Program for Leading Large-Scale Generative AI Models in the Physical AI Domain, No.RS-2025-25442824, AI Star Fellowship Program (UNIST) and No.RS-2020-II201336, Artificial Intelligence Graduate School Program (UNIST)).

References

Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. 2024. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 21476–21485.

Devikalyan Das, Christopher Wewer, Raza Yunus, Eddy Ilg, and Jan Eric Lenssen. 2024. Neural parametric gaussians for monocular non-rigid object reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10715–10725.

Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. 2024. RoMa: Robust dense feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19790–19800.

Martin A Fischler and Robert C Bolles. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* 24, 6 (1981), 381–395.

Xiang Guo, Jiadai Sun, Yuchao Dai, Guanying Chen, Xiaoqing Ye, Xiao Tan, Errui Ding, Yumeng Zhang, and Jingdong Wang. 2023. Forward flow for novel view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 16022–16033.

Zhiyang Guo, Wengang Zhou, Li Li, Min Wang, and Houqiang Li. 2024. Motion-aware 3d gaussian splatting for efficient dynamic scene reconstruction. *IEEE Transactions on Circuits and Systems for Video Technology* (2024).

Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, et al. 2025. Vipe: Video pose engine for 3d geometric perception. *arXiv preprint arXiv:2508.10934* (2025).

Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. 2024. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4220–4230.

Vishnu Jaganathan, Hannah Hanyun Huang, Muhammad Zubair Irshad, Varun Jampani, Amit Raj, and Zsolt Kira. 2024. Ice-g: Image conditional editing of 3d gaussian splats. *arXiv preprint arXiv:2406.08488* (2024).

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.* 42, 4 (2023), 139–1.

Jiahui Lei, Yijia Weng, Adam W Harley, Leonidas Guibas, and Kostas Daniilidis. 2025. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 6165–6177.

Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. 2009. EP n P: An accurate O (n) solution to the P n P problem. *International journal of computer vision* 81 (2009), 155–166.

Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. 2022. 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, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. 2025. MegaSaM: Accurate, fast and robust structure and motion from casual dynamic videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 10486–10496.

Cheng-You Lu, Peisen Zhou, Angela Xing, Chandradeep Pokhariya, Arnab Dey, Ishaan Nikhil Shah, Rugved Mavidipalli, Dylan Hu, Andrew I Comport, Kefan Chen, et al. 2024b. Diva-360: The dynamic visual dataset for immersive neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22466–22476.

Zhicheng Lu, Xiang Guo, Le Hui, Tianrui Chen, Min Yang, Xiao Tang, Feng Zhu, and Yuchao Dai. 2024a. 3d geometry-aware deformable gaussian splatting for dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8900–8910.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. 2024. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*. IEEE, 800–809.

Guan Luo, Tian-Xing Xu, Ying-Tian Liu, Xiao-Xiong Fan, Fang-Lue Zhang, and Song-Hai Zhang. 2024. 3D Gaussian Editing with A Single Image. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 6627–6636.

Haimin Luo, Teng Xu, Yuheng Jiang, Chenglin Zhou, Qiwei Qiu, Yingliang Zhang, Wei Yang, Lan Xu, and Jingyi Yu. 2022. Artemis: Articulated neural pets with appearance and motion synthesis. *arXiv preprint arXiv:2202.05628* (2022).

Yiqun Mei, Jiacong Xu, and Vishal M Patel. 2024. Reference-based Controllable Scene Stylization with Gaussian Splatting. *arXiv preprint arXiv:2407.07220* (2024).

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.

Francesco Palandra, Andrea Sanchietti, Daniele Baieri, and Emanuele Rodolà. 2024. Gsedit: Efficient text-guided editing of 3d objects via gaussian splatting. *arXiv preprint arXiv:2403.05154* (2024).

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. 2021. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228* (2021).

Sungheon Park, Minjung Son, Seokhwan Jang, Young Chun Ahn, Ji-Yeon Kim, and Nahyup Kang. 2023. Temporal interpolation is all you need for dynamic neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4212–4221.

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.

Abhishek Saroha, Florian Hofherr, Mariia Gladkova, Cecilia Curreli, Or Litany, and Daniel Cremers. 2025. ZDySS–Zero-Shot Dynamic Scene Stylization using Gaussian Splatting. *arXiv preprint arXiv:2501.03875* (2025).

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.

Olga Sorkine and Marc Alexa. 2007. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, Vol. 4. Citeseer, 109–116.

Robert W Sumner, Johannes Schmid, and Mark Pauly. 2007. Embedded deformation for shape manipulation. In *ACM siggraph 2007 papers*. 80–es.

Jiakai Sun, Han Jiao, Guangyuan Li, Zhanjie Zhang, Lei Zhao, and Wei Xing. 2024a. 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*. 20675–20685.

Yanhao Sun, Runze Tian, Xiao Han, XinYao Liu, Yan Zhang, and Kai Xu. 2024b. GSEdit-Pro: 3D Gaussian Splatting Editing with Attention-based Progressive Localization. In *Computer Graphics Forum*, Vol. 43. Wiley Online Library, e15215.

Qianqian Wang, Vickie Ye, Hang Gao, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. 2024. Shape of motion: 4d reconstruction from a single video. *arXiv preprint arXiv:2407.13764* (2024).

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. 2025. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 10510–10522.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 2024b. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 20310–20320.

Jing Wu, Jia-Wang Bian, Xinghui Li, Guangrun Wang, Ian Reid, Philip Torr, and Victor Adrian Prisacariu. 2024a. Gaussctrl: Multi-view consistent text-driven 3d gaussian splatting editing. In *European Conference on Computer Vision*. Springer, 55–71.

Jiankai Xing, Fujun Luan, Ling-Qi Yan, Xuejun Hu, Houde Qian, and Kun Xu. 2022. Differentiable rendering using rgbxy derivatives and optimal transport. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–13.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. 2024. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 20331–20341.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. 2023. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642* (2023).

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.