SD4R: DECOUPLING MOTION AND CORRESPONDENCE FOR 4D GAUSSIAN SCENE RECONSTRUCTION AND EDITING FROM UNPOSED VIDEOS

Anonymous authors
Paper under double-blind review

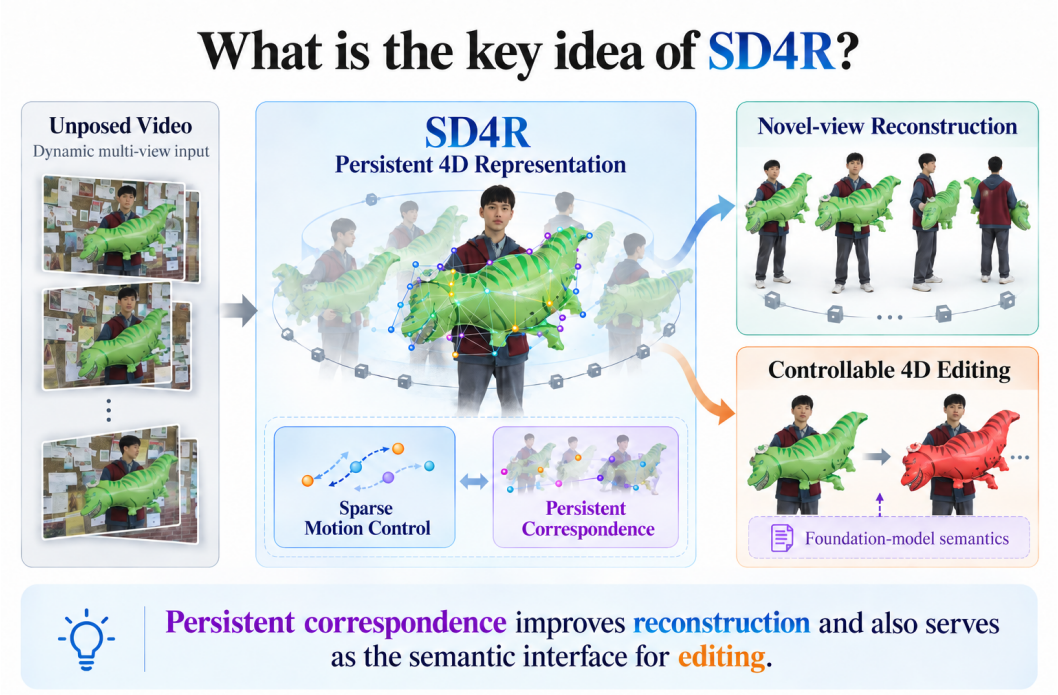


Figure 1: Key idea of SD4R. SD4R converts an unposed video into a persistent 4D Gaussian representation by decoupling motion control from correspondence persistence. This shared representation supports both novel-view reconstruction and controllable 4D editing.

ABSTRACT

Unposed dynamic reconstruction from casually captured videos is fundamentally underconstrained, as camera motion, non-rigid dynamics, and temporal correspondence must be inferred jointly from image observations. Beyond the well-known ambiguity between camera and scene motion, we identify *correspondence instability* as another source of optimization failure: a representation may explain each frame well while gradually losing the identity of the physical elements it is intended to track. We present SD4R, a unified framework that treats unposed dynamic reconstruction as the joint estimation of *geometry, motion, and persistent correspondence*. The central idea is to decouple motion representation from identity preservation, allowing scene dynamics to remain flexible while correspondence provides a stable temporal reference. We further introduce a compact sparse motion representation that captures coherent non-rigid dynamics through shared continuous structure. Importantly, the resulting persistent representation naturally extends beyond reconstruction. By linking image-space semantics to temporally persistent scene primitives, SD4R enables controllable object-level 4D editing within the same

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

representation. Experiments demonstrate that this formulation improves dynamic reconstruction consistency while supporting localized scene editing.

1 INTRODUCTION

Reconstructing dynamic 3D scenes from casually captured videos is important for novel-view synthesis, robotic perception, and editable visual content. Recent advances in 3D Gaussian Splatting (3DGS) Kerbl et al. (2023) provide an efficient explicit representation for static scenes, while dynamic extensions model non-rigid motion through deformation fields, explicit trajectories, or continuous temporal parameterizations Yang et al. (2024); Wu et al. (2024); Huang et al. (2024). However, most existing methods either rely on externally estimated camera poses or optimize scene deformation under imperfect camera geometry. In the *unposed* setting, camera motion and non-rigid deformation must be recovered jointly from uncalibrated video, creating an inherent explanation ambiguity: pose errors may be absorbed into deformation, while inaccurate scene motion can in turn interfere with camera refinement.

Beyond this pose–motion ambiguity, we identify a second source of instability that is largely overlooked: *correspondence instability*. A dynamic Gaussian Hu et al. (2025); Park et al. (2024) should not only reproduce the correct appearance at individual frames, but also preserve its association with the same physical scene element over time. Photometric reconstruction alone does not guarantee such consistency. As camera poses, geometry, and deformation are jointly updated, the projected locations of Gaussian primitives continuously change Liu et al. (2023); Lei et al. (2024); Wang et al. (2025). Consequently, repeatedly rematching Gaussians to image observations or 2D tracks can cause their supervision identities to switch during optimization, even when the rendering loss remains low. Such *correspondence drift* entangles motion estimation with identity assignment and can lead to temporally inconsistent trajectories.

We argue that this instability arises from an implicit coupling between *motion representation* and *correspondence assignment*. When the supervision identity of a Gaussian is determined by its current position, changing its motion can also change which observation supervises it. In other words, *how a Gaussian moves* and *which scene element it represents* become entangled. Our key principle is therefore to decouple motion representation from correspondence persistence: the motion state remains continuously optimizable, while the correspondence identity provides a stable temporal reference throughout optimization. This yields a simple but important property,

$$\text{motion update} \not\Rightarrow \text{correspondence reassignment}.$$

Based on this principle, we introduce **SD4R**, as shown in 1, a correspondence-persistent framework for unposed dynamic Gaussian reconstruction. SD4R first performs a camera warm-up with non-rigid deformation suppressed to obtain a stable geometric initialization. Long-range 2D trajectories are then lifted into 3D only to initialize scene motion. Rather than assigning an independent trajectory to every Gaussian, their motion is represented by a compact set of learnable sparse control nodes whose continuous trajectories drive nearby Gaussians through local interpolation.

Persistent correspondence also provides a natural interface between reconstruction and downstream scene manipulation. Image-space semantic masks alone do not directly specify which dynamic Gaussian primitives should be edited, particularly when the object moves across frames and viewpoints. We therefore transfer object semantics from 2D observations to persistent Gaussian primitives. Specifically, foundation models provide object localization, temporal masks, and candidate edited observations, while a semantic mask–trajectory–Gaussian association determines the editable primitives. Gaussian-level selection and parameter-subspace gating then restrict optimization to the target object and task-relevant attributes. This enables localized object-level 4D editing without requiring explicit 3D object annotations or manually defined object trajectories.

Our contributions are summarized as follows:

- We identify *correspondence instability* as a distinct failure mode in unposed dynamic reconstruction, and formulate the problem as jointly estimating geometry, motion, and persistent correspondence.

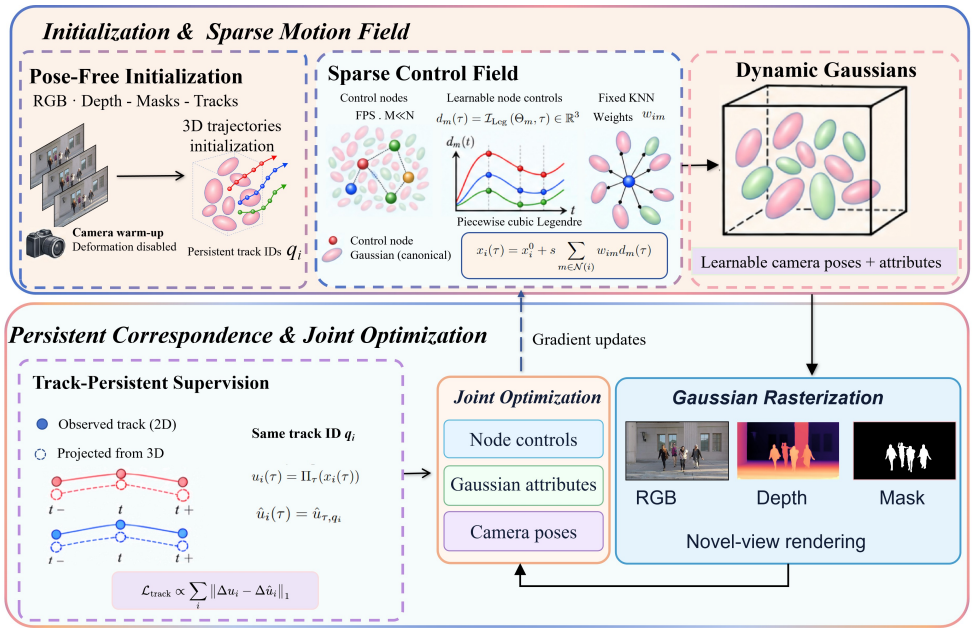


Figure 2: Overview of SD4R. We decouple motion representation from correspondence persistence in unposed dynamic reconstruction. Sparse control trajectories model coherent Gaussian motion, while persistent track identities provide stable long-range supervision during joint optimization of camera poses, Gaussian attributes, and scene dynamics.

- We introduce a correspondence-persistent dynamic Gaussian representation that explicitly decouples motion modeling from identity preservation, enabling stable long-range supervision during joint camera–scene optimization.
- We introduce a foundation-model-assisted object-level 4D editing framework with a principled semantic mask–trajectory–Gaussian association, enabling localized editing through Gaussian selection and parameter-subspace gating without explicit 3D object annotations or manually defined trajectories.

2 Related Work

Dynamic Gaussian Splatting. Following the success of 3D Gaussian Splatting (3DGS) Kerbl et al. (2023), an increasing number of studies have extended explicit Gaussian representations to dynamic scenes. D3DGS Yang et al. (2024) associates Gaussians with deformation parameters to capture non-rigid motion, while 4DGS Wu et al. (2024) introduces a spatio-temporal deformation representation for efficient dynamic rendering. E-D3DGS Bae et al. (2024) further assigns latent embeddings to individual Gaussians to improve deformation modeling. Beyond deformation fields, another line of approaches explicitly parameterizes temporal evolution using Gaussian trajectories, persistent scene primitives, or continuous temporal functions Li et al. (2024); Luiten et al. (2024); Lin et al. (2024); Hu et al. (2025). Such representations provide more direct access to object motion and are particularly suitable for temporal interpolation and scene editing.

Unposed Dynamic Reconstruction. Most dynamic novel-view synthesis methods assume that camera poses are obtained from calibrated capture systems or offline SfM preprocessing. Removing this assumption is substantially more challenging, as camera motion, scene geometry, and non-rigid deformation must be jointly inferred from the same image observations. Early monocular dynamic NeRF approaches therefore combine photometric reconstruction with auxiliary geometric cues, such as monocular depth, optical flow, and motion segmentation. For example, RoDynRF Liu et al. (2023) jointly estimates camera motion and dynamic scene content from casually captured videos, while DynPoint Zhou et al. (2023) represents dynamic scenes using explicit point-based geometry. Related

unposed and COLMAP-free Gaussian approaches Hong et al. (2024); Ji & Yao (2025); Chen et al. (2024) further demonstrate that camera and scene geometry can be recovered without externally provided SfM poses, although most of them primarily focus on static or sparse-view reconstruction. More recent approaches directly address unposed dynamic scenes by incorporating stronger temporal priors.

3 METHOD

3.1 OVERVIEW

Given an uncalibrated RGB sequence $\{\mathbf{I}_t\}_{t=1}^{T}$, SD4R reconstructs a dynamic Gaussian scene without external camera poses or SfM geometry, using automatically extracted depth, dynamic masks, and long-range 2D tracks as auxiliary cues. As illustrated in Fig. 2, SD4R first performs an unposed camera warm-up with non-rigid deformation disabled, then lifts reliable 2D tracks into 3D trajectories to initialize dynamic Gaussians and sparse motion controls. Scene dynamics are subsequently refined by learnable sparse control trajectories that drive dense Gaussians through local interpolation. The key principle is to decouple *motion* from *correspondence*: sparse controls determine how Gaussians move, while persistent track identities determine which observations supervise them. The same persistent correspondence further transfers image-space semantics to Gaussian primitives through a semantic mask–trajectory–Gaussian association, enabling localized object-level 4D editing via Gaussian selection and parameter gating.

3.2 UNPOSED CAMERA WARM-UP

Unposed dynamic reconstruction suffers from a coupled explanation ambiguity: similar image-space variations can be explained by either camera motion or non-rigid scene deformation. This ambiguity is particularly severe at the beginning of optimization, when geometry is still inaccurate and a high-capacity deformation model can reduce the reconstruction loss by absorbing motion that should instead be attributed to the camera. SD4R therefore adopts a camera warm-up stage that temporarily suppresses non-rigid deformation, biasing early optimization toward recovering the global rigid motion induced by the camera. The recovered poses are not fixed afterwards, but serve as a stable geometric initialization for subsequent joint camera–scene refinement and 3D track lifting.

Specifically, we use a lightweight time-conditioned pose network f_θ to predict a 6-DoF camera parameter for each frame,

$$\boldsymbol{\xi}_t = f_\theta(t) = [\boldsymbol{\omega}_t, \mathbf{v}_t], \tag{1}$$

where $\boldsymbol{\omega}_t$ and $\mathbf{v}_t$ denote the rotational and translational components, respectively. The resulting camera poses are optimized during warm-up using photometric and geometric supervision, and are further jointly refined after the dynamic model is activated.

3.3 TRACK-GUIDED INITIALIZATION AND SPARSE MOTION FIELD

Track-Guided Dynamic Gaussian Initialization. Given a video of length T, we first extract a set of 2D point trajectories :

$$\mathcal{P}_j = \{(\mathbf{p}_j^t, v_j^t)\}_{t=0}^{T-1}, t \in 0, ..., T-1. \tag{2}$$

where $\mathbf{p}_j^t \in \mathbb{R}^2$ denotes the image position of the j-th trajectory at frame t, and v_j^t denotes its visibility confidence. We retain trajectories satisfying:

$$\mathcal{J}_{\text{valid}} = \left\{ j \;\middle|\; \sum_{t=0}^{T-1} \mathbb{I}[v_j^t > \tau_v] \geq V_{\min} \right\}. \tag{3}$$

We use $V_{\min} = T$ for conservative initialization in our main experiments, while the implementation also supports partially visible trajectories.

We sample dynamic pixels $\mathbf{u}_i$ from dynamic candidate regions Ω_{dyn} and associate each sampled point with the nearest valid 2D trajectory in the canonical frame. The corresponding persistent track identity of Gaussian i is defined as:

$$q_i = \arg\min_{j \in \mathcal{J}_{\text{valid}}} \left\| \mathbf{u}_i - \mathbf{p}_j^0 \right\|_2^2. \tag{4}$$

Given the per-frame depth map D_t and camera projection model π_t, we lift the associated 2D trajectory into 3D as:

$$\mathbf{X}_i^t = \pi_t^{-1}\left(\mathbf{p}_{q_i}^t, D_t(\mathbf{p}_{q_i}^t)\right), \tag{5}$$

yielding an initial 3D trajectory:

$$\mathcal{X}_i = \{\mathbf{X}_i^t\}_{t=0}^{T-1}. \tag{6}$$

After filtering the lifted samples using the per-frame dynamic masks and valid depth ranges, we initialize the dynamic Gaussian from the sampled canonical-frame pixel:

$$\mathbf{x}_i^0 = \pi_0^{-1}(\mathbf{u}_i, D_0(\mathbf{u}_i)), \tag{7}$$

and retain q_i as its persistent track identity.

Sparse Control Field. Although trajectory lifting provides an initial 3D motion estimate for each dynamic Gaussian, independently parameterizing the motion of all Gaussians would introduce substantial parameter overhead and make the representation sensitive to noise in individual tracks and monocular depth. We therefore construct a shared continuous transformation field using a compact set of sparse control nodes.

Let the canonical centers of N dynamic Gaussians be

$$\mathcal{G} = \{\mathbf{x}_i^0\}_{i=1}^N. \tag{8}$$

We use farthest point sampling to select M spatially distributed Gaussians as control nodes Huang et al. (2024). Denoting the selected Gaussian index for the m-th control node by $i(m)$, we define $\mathbf{c}_m = \mathbf{x}_{i(m)}^0, M \ll N$.

Each control node directly inherits the initial 3D trajectory of its corresponding Gaussian. Its normalized displacement at discrete frame t is initialized as

$$\mathbf{d}_m^t = \frac{\mathbf{X}_{i(m)}^t - \mathbf{c}_m}{s}, \tag{9}$$

where s denotes the scene spatial scale. From these discrete displacements $\{\mathbf{d}_m^t\}_{t=1}^T$, we initialize C learnable temporal controls for each node, $\mathbf{\Theta}_m = \{\boldsymbol{\theta}_{m,k}\}_{k=1}^C, \boldsymbol{\theta}_{m,k} \in \mathbb{R}^3$. For an arbitrary continuous time $\tau \in [0,1]$, the node displacement is evaluated as

$$\mathbf{d}_m(\tau) = \mathcal{I}_{\text{Leg}}\left(\mathbf{\Theta}_m, \tau\right), \tag{10}$$

where $\mathcal{I}_{\text{Leg}}$ denotes a piecewise cubic temporal interpolator expressed in a Legendre basis. For each query time, it evaluates the displacement using only a fixed local set of neighboring temporal controls, yielding continuous motion without introducing per-frame motion parameters.

For each dynamic Gaussian $\mathbf{x}_i^0$, we identify its K nearest control nodes in canonical space, denoted by $\mathcal{N}_i$, and compute normalized inverse-distance interpolation weights:

$$w_{im} = \frac{\left(\|\mathbf{x}_i^0 - \mathbf{c}_m\|_2 + \epsilon\right)^{-1}}{\sum_{n\in\mathcal{N}_i}\left(\|\mathbf{x}_i^0 - \mathbf{c}_n\|_2 + \epsilon\right)^{-1}}, \qquad m \in \mathcal{N}_i. \tag{11}$$

The position of dynamic Gaussian i at time τ is then obtained by locally blending the displacements of its neighboring control nodes:

$$\mathbf{x}_i(\tau) = \mathbf{x}_i^0 + s\sum_{m\in\mathcal{N}_i} w_{im}\mathbf{d}_m(\tau). \tag{12}$$

By representing the continuous motion of dense dynamic Gaussians with a small number of shared control nodes, the sparse control field reduces the number of motion parameters while encouraging locally coherent deformation through shared motion support.

Efficiency. Our sparse control field shares motion among nearby Gaussians instead of assigning an independent temporal motion model to every Gaussian. The learnable motion parameters scale as $O(MC)$ for M control nodes and C temporal controls, rather than $O(NC)$ for N Gaussians. Evaluating the field at one time step costs $O(M + NK)$: node controls are interpolated once, and each Gaussian blends the motion of its K neighboring nodes. When split creates S new Gaussians, their locality state is rebuilt once with $O(SM)$ distance computations. Since $M \ll N$ and K is small, the method reduces motion-parameter optimization cost while keeping field evaluation linear in the number of Gaussians.

3.4 PERSISTENT TRACK SUPERVISION

The sparse motion field determines how each Gaussian moves in 3D, but does not by itself determine which image observation should consistently supervise that motion. During joint pose and motion refinement, the projected position of a Gaussian continuously changes. If the supervision target is recomputed by nearest-neighbor matching at every iteration, the assigned track may switch as the Gaussian moves or the camera pose is updated, resulting in correspondence drift. We therefore retain the track identity q_i established during initialization and use it as persistent supervision state.

Let τ_t denote the continuous time corresponding to frame t. The current image-space position of Gaussian i is obtained by projecting its dynamically updated 3D position:

$$\hat{\mathbf{p}}_i^t = \Pi_t(\mathbf{x}_i(\tau_t)), \tag{13}$$

where Π_t denotes projection under the current camera parameters. In contrast to the projected Gaussian position, which changes throughout optimization, its supervision target is always taken from the stored track q_i: $\mathbf{p}_{q_i}^t$. Thus, the Gaussian trajectory remains fully optimizable, while the correspondence used to supervise it is kept persistent.

Rather than enforcing absolute pixel alignment, we constrain the forward and backward image-space displacements. For $\sigma \in \{-1, +1\}$, we define

$$\Delta\hat{\mathbf{p}}_i^{t,\sigma} = \hat{\mathbf{p}}_i^{t+\sigma} - \hat{\mathbf{p}}_i^t, \tag{14}$$

and the corresponding track displacement as

$$\Delta\mathbf{p}_{q_i}^{t,\sigma} = \mathbf{p}_{q_i}^{t+\sigma} - \mathbf{p}_{q_i}^t. \tag{15}$$

The persistent track loss is then

$$\mathcal{L}_{\text{track}} = \frac{1}{2|\mathcal{V}|} \sum_{(i,t)\in\mathcal{V}} \sum_{\sigma\in\{-1,+1\}} \left\| \frac{\Delta\hat{\mathbf{p}}_i^{t,\sigma}}{(W,H)} - \frac{\Delta\mathbf{p}_{q_i}^{t,\sigma}}{(W,H)} \right\|_{1,\text{mean}}, \tag{16}$$

where (W, H) normalizes the horizontal and vertical displacements, and $\mathcal{V}$ contains valid Gaussian–track pairs whose current and adjacent projections lie inside the image.

This displacement-based formulation constrains the temporal motion pattern rather than the absolute projected position, making the supervision less sensitive to residual camera-pose errors and imperfections in the lifted geometry.

3.5 OBJECT-LEVEL CONTROLLABLE 4D SCENE EDITING

Given a reconstructed unposed 4D Gaussian scene, directly applying image-level editing supervision may modify background regions or neighboring instances. We therefore introduce an object-level editing pipeline that controls *where* editing supervision is valid, *which Gaussians* receive it, and *which parameters* can be updated. We use Qwen-Image-Edit to generate candidate edited observations, and further refine them with our unposed dynamic Gaussian model equipped with object-aware Gaussian primitive selection and parameter gating.

Object-Aware Mask Construction. Given a text instruction, we specify a target class c_{tar} and optionally protected classes $\{c_{\text{pro}}^k\}$. GroundingDINO localizes each class, SAM2 refines the detections into instance masks and propagates them temporally. To reduce missed detections and long-term drift, we fuse SAM2 propagation with frame-wise detection:

$$M_t^c = M_{t,\text{SAM2}}^c \vee M_{t,\text{frame}}^c. \tag{17}$$

The editable region is then defined as

$$M_t = D_{r_t}\left(M_t^{\text{tar}}\right) \wedge \neg\left(\bigvee_k D_{r_p}\left(M_{t,k}^{\text{pro}}\right)\right), \tag{18}$$

where $D_r(\cdot)$ denotes morphological dilation. A larger protection radius creates a safety margin around neighboring objects and suppresses edit leakage. We first fuse temporally propagated and

frame-wise detected masks to improve mask completeness. The final editable region is obtained by dilating the target mask and subtracting the union of dilated protected-object masks, thereby preventing edit leakage to neighboring instances. For global scene-level instructions, the spatial restriction is relaxed by setting $M_t = \mathbf{1}$ and activating all Gaussian gates $g_i = 1$. The same editing objective is then applied over the full reconstructed representation. Thus, the proposed formulation covers both localized object-level editing and global scene-level transformation.

From 2D Masks to Editable Gaussians. The image-space instance identities are not directly shared with the persistent track identities of the reconstructed Gaussian scene. We therefore bridge the 2D semantic masks and 3D Gaussians through the precomputed point trajectories $\mathcal{P}_j = \{(\mathbf{p}_j^t, v_j^t)\}_{t=0}^{T-1}$, where $\mathbf{p}_j^t$ and v_j^t denote the image position and visibility of track j at frame t.

For each trajectory, we evaluate its multi-frame agreement with the target masks over visible observations:

$$c_j = \frac{\sum_t v_j^t M_t(\mathbf{p}_j^t)}{\sum_t v_j^t + \epsilon}. \tag{19}$$

Tracks with sufficient visibility and consistent support from the target masks are retained as editable trajectories. Let $\mathcal{J}_{\text{edit}}$ denote the set of editable track indices selected by mask consistency. The Gaussian edit gate for primitive i is defined as $g_i = \mathbb{I}[\, q_i \in \mathcal{J}_{\text{edit}}\,]$. This establishes the correspondence semantic mask $\rightarrow$ 2D trajectory $\rightarrow$ Gaussian primitive, allowing image-space semantic regions to be transferred to persistent Gaussian primitives across time.

Masked Supervision and Parameter Optimization. Candidate edited frames $\widehat{I}_t$ are composited with the original observations I_t using the target mask:

$$I_t^* = M_t \odot \widehat{I}_t + (1 - M_t) \odot I_t. \tag{20}$$

To preserve unrelated scene content, we optimize only the Gaussian parameters relevant to the requested edit while keeping the remaining scene parameters fixed. Let Θ denote all optimizable Gaussian parameters, and let $\Theta_{\text{edit}} \subseteq \Theta$ be the task-dependent editable parameter subset. The optimization is restricted to the selected Gaussians through

$$\nabla_{\theta_i} \mathcal{L} \leftarrow g_i \nabla_{\theta_i} \mathcal{L}, \qquad \theta_i \in \Theta_{\text{edit}}, \tag{21}$$

where g_i is the Gaussian-level edit gate. This design confines the optimization to both the target primitives and the parameter subspace required by the specific editing operation.

Let $\hat{I}_t^r$ denote the image rendered from the current edited 4D Gaussian scene at frame t. The overall editing objective is

$$\mathcal{L}_{\text{edit}}^{\text{total}} = \mathcal{L}_{\text{edit}} + \mathcal{L}_{\text{preserve}} + \big(1 - \text{SSIM}(\hat{I}_t^r, I_t^*)\big), \tag{22}$$

where $\mathcal{L}_{\text{edit}}$ enforces the desired appearance inside the target region, and $\mathcal{L}_{\text{preserve}}$ suppresses changes outside the target region. Only training cameras participate in optimization; test views are used solely for validation and evaluation.

4 Experiments

4.1 Experimental setup

Datasets and protocol. We conduct quantitative evaluation for novel-view synthesis on the NVIDIA Dynamic Scenes benchmark (Yoon et al., 2020), and use the DAVIS dataset (Pont-Tuset et al., 2017), which contains real-world in-the-wild scenes, for qualitative stress testing. All experiments follow the standard training/test splits, and evaluation protocols. For the main comparisons, we report Peak Signal-to-Noise Ratio (PSNR) and Learned Perceptual Image Patch Similarity (LPIPS) Zhang et al. (2018), which are among the most commonly used metrics in prior work. We separate methods that rely on SfM preprocessing from SfM-free methods, since pose availability changes the reconstruction problem. The comparison covers dynamic radiance fields (Gao et al., 2021; Tian et al., 2023), dynamic Gaussian deformation models(Huang et al., 2024), robust dynamic radiance fields (Liu et al., 2023), motion scaffolds (Lei et al., 2024), and track-aware dynamic point methods (Wang et al., 2025).

Table 1: Experimental comparison on the NVIDIA Dynamic Scenes subset (Yoon et al., 2020). Each scene reports PSNR↑/LPIPS↓, and averages are recomputed over the shown scenes. Across all listed methods, best and second denote the best and second-best values for each metric, respectively.

	Methods	Jumping	Skating	Truck	Umbrella	Balloon1	Balloon2	Playground	Average
SfM preprocessing	DynNeRF (ICCV'21) (Gao et al., 2021)	24.68 / 0.090	32.66 / 0.035	28.56 / 0.082	23.26 / 0.137	22.36 / 0.104	27.06 / 0.049	24.15 / 0.080	26.10 / 0.082
	MonoNeRF (ICCV'23) (Tian et al., 2023)	24.26 / 0.091	32.06 / 0.044	27.56 / 0.115	23.62 / 0.180	21.89 / 0.129	27.36 / 0.052	22.61 / 0.130	25.62 / 0.106
	STGS (CVPR'24) (Li et al., 2024)	20.82 / 0.187	24.80 / 0.109	25.01 / 0.103	21.88 / 0.195	20.36 / 0.196	23.12 / 0.124	19.23 / 0.151	22.17 / 0.152
	SC-GS (CVPR'24) (Huang et al., 2024)	15.68 / 0.920	14.88 / 0.908	23.81 / 0.140	21.84 / 0.160	20.17 / 0.179	21.07 / 0.149	20.71 / 0.115	19.74 / 0.367
	D3DGS (CVPR'24) (Yang et al., 2024)	22.02 / 0.266	24.06 / 0.227	23.04 / 0.247	22.67 / 0.192	21.22 / 0.202	25.86 / 0.118	22.30 / 0.111	23.02 / 0.195
	4DGS (CVPR'24) (Wu et al., 2024)	22.37 / 0.178	26.72 / 0.084	25.93 / 0.097	22.36 / 0.178	21.89 / 0.153	24.85 / 0.081	21.36 / 0.089	23.64 / 0.123
	SP-GS (ICML'24) (Wan et al., 2024)	22.13 / 0.468	29.21 / 0.236	27.38 / 0.190	24.88 / 0.323	24.36 / 0.180	29.65 / 0.097	22.29 / 0.234	25.70 / 0.247
	Casual-FVS (ECCV'24) (Lee et al., 2024b)	23.45 / 0.100	29.98 / 0.045	25.22 / 0.090	23.24 / 0.096	23.76 / 0.079	24.15 / 0.081	22.19 / 0.074	24.57 / 0.081
	Kinematic Fields (ECCV'24) (Im et al., 2024)	23.74 / 0.102	31.89 / 0.035	28.34 / 0.074	25.46 / 0.098	23.72 / 0.079	26.49 / 0.055	24.64 / 0.053	26.33 / 0.071
	E-D3DGS (ECCV'24) (Bae et al., 2024)	22.39 / 0.180	24.62 / 0.137	26.85 / 0.081	23.60 / 0.145	20.87 / 0.161	24.39 / 0.073	20.79 / 0.124	23.36 / 0.129
	CTNeRF (PR'24) (Miao et al., 2024)	24.35 / 0.094	33.51 / 0.034	28.27 / 0.084	23.48 / 0.129	22.19 / 0.111	26.86 / 0.048	24.28 / 0.077	26.13 / 0.082
	D-NPC (Eurographics'25) (Kappel et al., 2025)	24.51 / 0.116	30.22 / 0.061	28.92 / 0.084	24.15 / 0.136	22.26 / 0.161	25.84 / 0.107	23.59 / 0.099	25.64 / 0.109
	Shape-of-Motion (arXiv'24) (Wang et al., 2025)	22.74 / 0.128	29.12 / 0.048	24.63 / 0.126	23.79 / 0.089	22.74 / 0.120	25.83 / 0.072	20.61 / 0.130	24.21 / 0.102
	Ex4DGS (NeurIPS'24) (Lee et al., 2024a)	18.93 / 0.321	21.92 / 0.233	19.04 / 0.308	19.03 / 0.340	14.69 / 0.503	16.29 / 0.457	14.16 / 0.437	17.72 / 0.371
	DynPoint (NeurIPS'23) (Zhou et al., 2023)	24.69 / 0.097	31.34 / 0.045	29.30 / 0.061	24.59 / 0.086	22.77 / 0.099	27.63 / 0.049	25.37 / 0.039	26.53 / 0.068
SfM-Free	GaussianVideo (ICCV'25) Bond et al. (2025)	20.38 / 0.217	23.81 / 0.203	21.19 / 0.269	23.54 / 0.111	19.84 / 0.216	22.64 / 0.162	20.83 / 0.118	21.75 / 0.185
	RoDynRF (CVPR'23) (Liu et al., 2023)	24.27 / 0.100	28.71 / 0.046	28.85 / 0.066	23.25 / 0.104	21.81 / 0.122	25.58 / 0.064	25.20 / 0.052	25.38 / 0.079
	MoSca (arXiv'24) Lei et al. (2024)	25.16 / 0.083	32.08 / 0.045	27.80 / 0.080	25.49 / 0.084	23.63 / 0.097	27.66 / 0.055	24.37 / 0.054	26.60 / 0.071
	SD4R	25.07 / 0.070	34.02 / 0.029	28.81 / 0.043	24.72 / 0.072	25.03 / 0.066	29.40 / 0.032	25.40 / 0.041	27.49 / 0.050

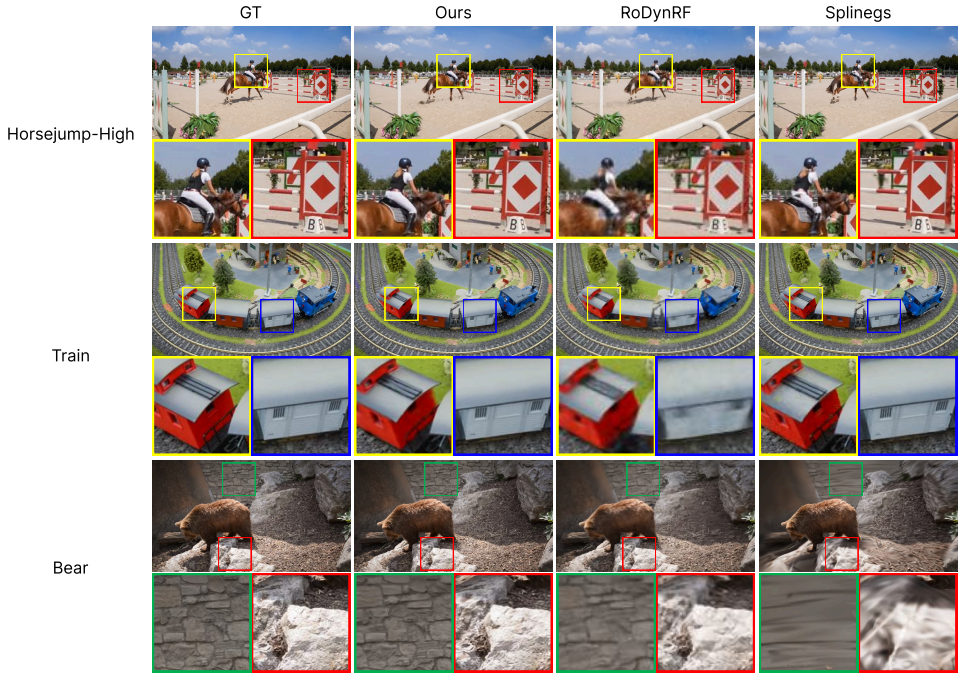


Figure 3: Qualitative DAVIS (Pont-Tuset et al., 2017) results.

4.2 Novel-view synthesis

Quantitative analysis. Table 1 compares novel-view synthesis quality on the NVIDIA subset. Across all listed methods, SD4R obtains the best average for both PSNR and LPIPS, reaching 27.49 dB and 0.050 LPIPS. The method is strongest on scenes with large foreground motion: it achieves the best PSNR on Skating and Balloon1, the second-best PSNR on Jumping and Balloon2, and the best LPIPS on Skating, Truck, Balloon2, and the average. Truck remains a case where methods using SfM preprocessing obtain higher PSNR, but SD4R still gives the lowest LPIPS, suggesting that the unposed persistent-control representation preserves perceptual structure even when pixel-aligned reconstruction is difficult. Overall, these results demonstrate that SD4R achieves a favorable balance between pixel-level fidelity and perceptual quality, while remaining robust to challenging foreground dynamics under the unposed setting. **Due to space limitations, we provide additional editing results and detailed analysis in the appendix.**

Qualitative analysis on in-the-wild scenes. We further evaluate SD4R on real-world DAVIS (Pont-Tuset et al., 2017) sequences to assess its robustness beyond controlled benchmarks. Such in-the-wild videos are particularly valuable because they introduce factors that are difficult to reproduce in curated datasets, including irregular camera motion, large non-rigid deformation, partial occlusion, motion blur, and complex foreground–background interactions. As shown in Fig. 3, SD4R preserves sharper boundaries and finer structures across Horsejump-High, Train, and Bear. Compared with RoDynRF (Liu et al., 2023) and SplineGS Park et al. (2024) , our results exhibit less local blur and structure over-smoothing in the highlighted regions. These results demonstrate that SD4R remains robust under complex real-world dynamics, highlighting the importance of in-the-wild evaluation beyond controlled settings.

4.3 Ablations

Table 2 evaluates the contribution of each component on five selected scenes. Removing the sparse control field causes the largest performance degradation, reducing PSNR from 28.47 dB to 18.27 dB and increasing LPIPS from 0.048 to 0.272, demonstrating the importance of structured motion sharing. Among the Gaussian attributes, scale modeling is particularly important, while rotation, opacity, and color provide complementary improvements. For correspondence modeling, removing the persistent track loss or identity propagation consistently degrades all metrics. Shuffling track identities further reduces performance, confirming that the benefit stems from maintaining correct long-range correspondences rather than merely introducing track-based regularization. Overall, sparse motion control and persistent track supervision are complementary: the former models how Gaussian primitives move, while the latter determines which observations consistently supervise that motion.

Table 2: Ablation study on five selected scenes. We report the average PSNR↑, SSIM↑, and LPIPS↓ over these scenes for comparison.

Ablation setting	PSNR ↑	SSIM ↑	LPIPS ↓
Motion and Gaussian parameterization			
w/o sparse control field	18.27	0.722	0.272
w/o rotations	27.27	0.886	0.061
w/o scales	19.37	0.574	0.599
w/o opacity	27.32	0.871	0.078
w/o color	27.17	0.884	0.065
Persistent correspondence			
w/o persistent track loss	27.44	0.888	0.060
w/o identity propagation	27.33	0.887	0.062
w/ shuffled track id	27.05	0.883	0.067
SD4R (Ours)	**28.47**	**0.898**	**0.048**

5 Conclusion

We presented SD4R, a unposed framework for dynamic Gaussian reconstruction and object-level 4D editing. Beyond camera–deformation ambiguity, we identify *correspondence instability* as a key failure mode and formulate unposed reconstruction as jointly estimating geometry, motion, and persistent correspondence. Our core design decouples motion representation from correspondence persistence, allowing scene dynamics to remain optimizable while maintaining stable long-range supervision. A compact sparse control field models coherent non-rigid motion, while the same persistent correspondence further bridges image-space semantics and dynamic Gaussian primitives for localized 4D editing. Overall, our results show that explicitly maintaining correspondence as a persistent state benefits both reconstruction stability and controllable scene manipulation.

AI use statement

We used generative AI tools to assist with literature review, code implementation and debugging, and language polishing. All AI-assisted outputs were manually reviewed, verified, and modified by the authors. We take full responsibility for the research ideas, experiments, claims, and final content of the submission.

References

Jeongmin Bae, Seoha Kim, Youngsik Yun, Hahyun Lee, Gun Bang, and Youngjung Uh. Per-gaussian embedding-based deformation for deformable 3d gaussian splatting. In *European Conference on Computer Vision*, pp. 321–335. Springer, 2024.

Andrew Bond, Jui-Hsien Wang, Long Mai, Erkut Erdem, and Aykut Erdem. Gaussianvideo: Efficient video representation via hierarchical gaussian splatting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7187–7196, 2025.

Yu Chen, Rolandos Alexandros Potamias, Evangelos Ververas, Jifei Song, Jiankang Deng, and Gim Hee Lee. Zerogs: Training 3d gaussian splatting from unposed images. *arXiv preprint arXiv:2411.15779*, 2024.

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

Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jisang Han, Jiaolong Yang, Chong Luo, and Seungryong Kim. Pf3plat: Pose-free feed-forward 3d gaussian splatting. *arXiv preprint arXiv:2410.22128*, 2024.

Bingbing Hu, Yanyan Li, Rui Xie, Bo Xu, Haoye Dong, Junfeng Yao, and Gim Hee Lee. Learnable infinite taylor gaussian for dynamic view rendering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26844–26854, 2025.

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

Woobin Im, Geonho Cha, Sebin Lee, Jumin Lee, Juhyeong Seon, Dongyoon Wee, and Sung-Eui Yoon. Regularizing dynamic radiance fields with kinematic fields. In *European Conference on Computer Vision*, 2024.

Bo Ji and Angela Yao. Sfm-free 3d gaussian splatting via hierarchical training. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 21654–21663, 2025.

Moritz Kappel, Florian Hahlbohm, Timon Scholz, Susana Castillo, Christian Theobalt, Martin Eisemann, Vladislav Golyanik, and Marcus A. Magnor. D-NPC: dynamic neural point clouds for non-rigid view synthesis from monocular video. *Comput. Graph. Forum*, 44(2), 2025.

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

Junoh Lee, ChangYeon Won, Hyunjun Jung, Inhwan Bae, and Hae-Gon Jeon. Fully explicit dynamic guassian splatting. In *Proceedings of the Neural Information Processing Systems*, 2024a.

Yao-Chih Lee, Zhoutong Zhang, Kevin Blackburn-Matzen, Simon Niklaus, Jianming Zhang, Jia-Bin Huang, and Feng Liu. Fast view synthesis of casual videos with soup-of-planes. In *European Conference on Computer Vision (ECCV)*, 2024b.

Jiahui Lei, Yijia Weng, Adam Harley, Leonidas Guibas, and Kostas Daniilidis. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. *arXiv preprint arXiv:2405.17421*, 2024.

Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8508–8520, 2024.

Youtian Lin, Zuozhuo Dai, Siyu Zhu, and Yao Yao. Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21136–21145, 2024.

Yu-Lun Liu, Chen Gao, Andreas Meuleman, Hung-Yu Tseng, Ayush Saraf, Changil Kim, Yung-Yu Chuang, Johannes Kopf, and Jia-Bin Huang. Robust dynamic radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *3DV*, 2024.

Xingyu Miao, Yang Bai, Haoran Duan, Fan Wan, Yawen Huang, Yang Long, and Yefeng Zheng. Ctnerf: Cross-time transformer for dynamic neural radiance field from monocular video. *Pattern Recognition*, pp. 110729, 2024. ISSN 0031-3203.

Jongmin Park, Minh-Quan Viet Bui, Juan Luis Gonzalez Bello, Jaeho Moon, Jihyong Oh, and Munchurl Kim. Splinegs: Robust motion-adaptive spline for real-time dynamic 3d gaussians from monocular video. *CoRR*, abs/2412.09982, 2024.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Fengrui Tian, Shaoyi Du, and Yueqi Duan. MonoNeRF: Learning a generalizable dynamic radiance field from monocular videos. In *Proceedings of the International Conference on Computer Vision (ICCV)*, October 2023.

Diwen Wan, Ruijie Lu, and Gang Zeng. Superpoint gaussian splatting for real-time high-fidelity dynamic scene reconstruction. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 49957–49972, 2024.

Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *International Conference on Computer Vision (ICCV)*, 2025.

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

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

Jae Shin Yoon, Kihwan Kim, Orazio Gallo, Hyun Soo Park, and Jan Kautz. Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

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

Kaichen Zhou, Jia-Xing Zhong, Sangyun Shin, Kai Lu, Yiyuan Yang, Andrew Markham, and Niki Trigoni. Dynpoint: Dynamic neural point for view synthesis. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

A Appendix

A.1 Qualitative Results on 4D Scene Editing

Fig. 4 presents representative examples covering both localized object manipulation and global scene transformation. These examples span two complementary axes of editing difficulty: *local vs. global* spatial scope and *appearance vs. structural/semantic* modification. For each example, an image-editing foundation model first produces a reference observation according to the semantic instruction, while our framework determines how this image-space edit is transferred into the reconstructed 4D Gaussian representation. The examples cover substantially different editing scopes, ranging from local appearance modification and object removal to full-scene stylization and environmental transformation.

Object recoloring. In the first row, the instruction changes the inflatable dinosaur-shaped balloon from green to bright red. The editable mask isolates the balloon from the person and background, providing a clear object-level constraint. Our result reproduces the desired red appearance over

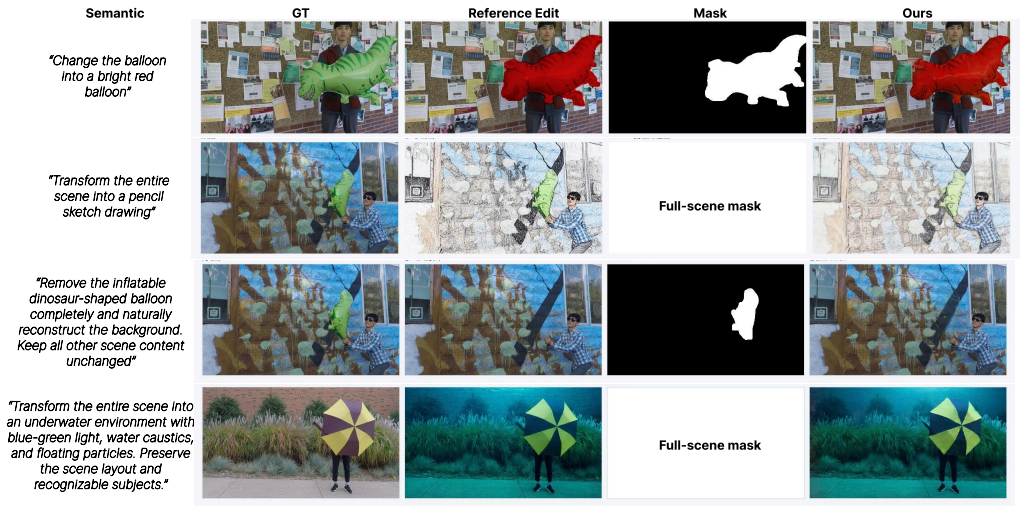


Figure 4: Qualitative results of controllable 4D scene editing. Given a semantic instruction, our method supports both localized object-level edits and global scene-level transformations. For localized edits, semantic masks restrict modifications to the target object while preserving surrounding content; for global edits, the full scene is transformed consistently with the instruction. From top to bottom, we show object recoloring, global style transfer, object removal, and global environment transformation.

the target object while largely preserving the person's appearance, clothing, background wall, and surrounding scene content. Importantly, the modification follows the complete spatial extent of the balloon rather than affecting only isolated image regions.

Global style transformation. The second row considers a substantially different setting in which the entire scene is transformed into a pencil-sketch drawing. Unlike object-level editing, no localized object mask is required because the semantic instruction applies globally. Our reconstruction follows the reference sketch appearance across the wall, person, and foreground objects while preserving the overall scene composition and recognizable geometry. This example shows that the proposed editing framework is not restricted to local object attributes, but can also accommodate scene-level appearance transformations when the editable support is extended to the full representation.

Object removal. The third row presents a more challenging structural edit that removes the inflatable balloon and reconstructs the previously occluded background. Compared with recoloring, removal requires not only suppressing the target object's appearance but also preventing its Gaussian primitives from leaving residual artifacts in the edited rendering. The target mask localizes the balloon, and our result removes the corresponding object while keeping the person and surrounding background largely unchanged. The reconstructed region follows the appearance of the edited reference without introducing substantial changes to unrelated scene elements. This experiment highlights the importance of Gaussian-level selection: the edit is injected into the primitives associated with the target trajectory rather than indiscriminately updating nearby Gaussians.

Global environment transformation. The last row demonstrates a scene-level semantic transformation from the original outdoor environment to an underwater appearance. The reference image introduces strong global changes in color, illumination, vegetation appearance, and atmospheric characteristics. Despite this large domain shift, our result reproduces the dominant underwater appearance while preserving the original scene layout and recognizable foreground subject. Compared with localized edits, this case requires coordinated modification over a much larger set of Gaussian primitives and therefore evaluates the flexibility of the reconstructed representation under global semantic supervision.

Taken together, these examples demonstrate two complementary editing regimes. For object-level instructions, persistent correspondences provide a bridge from semantic masks to temporally associated Gaussian primitives, allowing the edit to remain localized to the intended object. For global instructions, the same reconstruction can relax the spatial restriction and propagate the desired appear-

ance over the entire scene. The results therefore suggest that the proposed representation supports both *localized semantic manipulation* and *scene-level transformation* within a unified 4D editing framework.