MONO4D: LIFTING SINGLE-IMAGE 3D GAUSSIAN RECONSTRUCTION TO DYNAMIC DECOMPOSED 4D SCENES FROM MONOCULAR VIDEOS

Anonymous authors
Paper under double-blind review

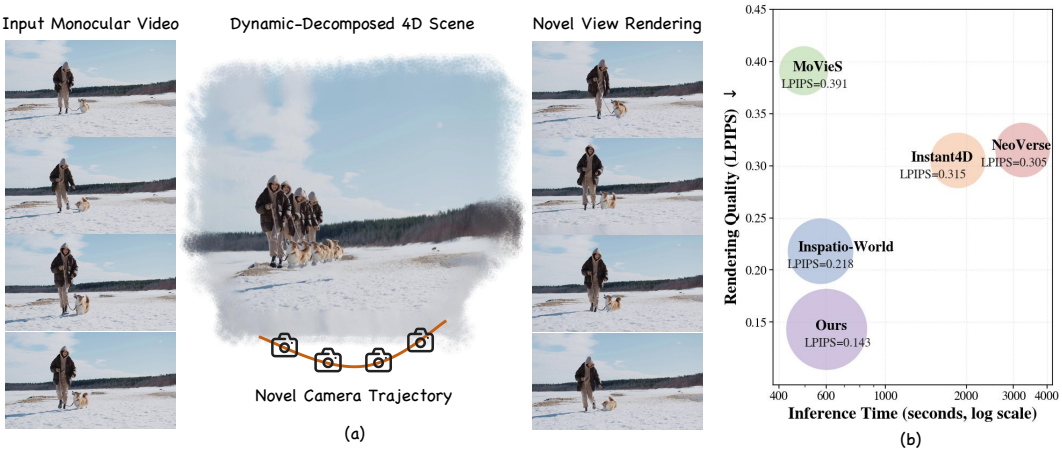


Figure 1: **Part(a): Monocular 4D Scene Reconstruction.** Given a monocular video input (left), Mono4D infers a static background Gaussian Splatting and a temporally evolving dynamic 3DGS sequence via static-dynamic scene decoupling (middle), enabling real-time nearby novel view synthesis for dynamic scenes (right). **Part(b):** Bubble chart for comparison with recent state-of-the-art approaches. Note that bubble size is inversely proportional to the LPIPS value.

ABSTRACT

Monocular 4D scene reconstruction is a challenging yet important task for immersive 3D applications. Existing optimization-based methods require multi-view data and costly per-scene optimization, while camera-controlled video generation methods lack explicit 3D representations and demand repeated inference for new camera trajectories; yet recent feedforward 3D reconstruction models perform well on static scenes but suffer from severe geometric and inter-frame inconsistencies when handling dynamic scenes. To address these challenges, we introduce **Mono4D**, a framework built upon a **geometry-aligned static–dynamic decoupled reconstruction paradigm** that lifts single-image feedforward 3DGS models to monocular videos of dynamic scenes without additional 4D training or per-scene optimization, enabling efficient nearby novel view rendering. Given a monocular video, Mono4D first estimates camera poses and dynamic masks, then aligns frame-wise 3DGS predictions to a shared world coordinate system. Reliable cross-frame background Gaussians are fused into a globally consistent static representation, while dynamic objects are modeled as temporally indexed Gaussian sequences. To enhance static-dynamic separation, we introduce a boundary-aware mask refinement and Gaussian assignment strategy to mitigate static-dynamic leakage and edge artifacts near fine structures. We further propose a spatiotemporal Gaussian compression scheme leveraging spatial sparsity and temporal redundancy through Gaussian pruning, cluster-level motion encoding, and residual coding for a compact 4D scene representation. Extensive experiments on novel view synthesis benchmarks demonstrate that Mono4D achieves superior performance on monocular 4D scene reconstruction compared with existing approaches, in terms of both visual fidelity and inference efficiency.

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.

1 INTRODUCTION

Monocular 4D scene reconstruction enables free-viewpoint interactive exploration of dynamic scenes and supports the generation of spatiotemporally consistent content along novel camera trajectories. It serves as a critical technology for immersive virtual reality and 3D content creation.

Early advances in 4D dynamic scene reconstruction are driven by per-scene optimization-based neural rendering methods Park et al. (2021); Feng et al. (2024); Yang et al. (2023); Wu et al. (2024); Lee et al. (2024); Sun et al. (2024); Liu et al. (2025a); Park et al. (2025b), which are built primarily upon NeRF Mildenhall et al. (2021) or 3D Gaussian Splatting Kerbl et al. (2023), achieving decent rendering quality on standard benchmarks. These methods optimize geometry and motion fields for each captured scene. Video diffusion models then emerge as another promising paradigm for novel view synthesis: one branch reconstructs 3D geometry and uses video diffusion models to inpaint occluded regions Yang et al. (2026); Zhou et al. (2026); Cao et al. (2026); Team (2026); Song et al. (2026); Lin et al. (2026b), while another conditions diffusion models directly on camera poses for view synthesis Bai et al. (2025); Li et al. (2026b); Park et al. (2025a); Paliwal et al. (2026). Recently, feedforward 3D reconstruction models enable single-pass 3D prediction from images or videos Lin et al. (2025); Yu et al. (2026); Mescheder et al. (2026); Li et al. (2026a); Xu et al. (2025); Chen et al. (2025), and they can rapidly output explicit 3D scene representations, inspiring feedforward 4D methods including 4DGT Xu et al. (2025) and MoVieS Lin et al. (2026a) that aim to predict scene geometry and motion directly from monocular video inputs.

Despite these developments, existing paradigms still suffer from notable limitations. Optimization-based 4D reconstruction requires dense synchronized multi-view video inputs Yoon et al. (2020); Li et al. (2022); under the monocular inputs setting, under-constrained optimization causes visible artifacts such as floating Gaussians and motion distortions. Diffusion-based methods lack explicit 3D scene representations and require separate inference for each new camera trajectory, leading to temporal appearance inconsistency and high computational overhead. Current feedforward approaches also struggle with dynamic scenes: transferring image-based 3D feedforward models to videos yields severe inter-frame inconsistency; some approaches Liu et al. (2025b); Chen et al. (2025) are restricted to generating 4D scenes only from image and text prompts. Other feedforward 4D methods cannot deliver continuous novel view rendering synchronized with scene motion. These challenges motivate us to pursue static-dynamic separation for monocular 4D reconstruction. We summarize the key properties of representative 4D novel view synthesis paradigms in Tab. 1.

To address these limitations, we propose Mono4D, a monocular 4D reconstruction framework for dynamic novel view rendering. Given a feedforward 3D model that predicts single-image 3D Gaussians, Mono4D applies this backbone to video inputs, lifting single-image 3D reconstruction to 4D dynamic scene modeling. The core of Mono4D is a static-dynamic decoupled reconstruction paradigm. We first extract dynamic masks from the input video using an off-the-shelf model, and decompose the scene into moving-object sequences and a stationary background. For the static background component, we fuse background Gaussians from each frame and reconstruct a single unified Gaussian point cloud that maintains geometric consistency; for the dynamic objects, we model a time-varying Gaussian sequence to faithfully capture continuous object motion. During reconstruction, we introduce a dedicated dynamic mask refinement strategy to cleanly separate static and dynamic regions, effectively eliminating artifacts along region boundaries. In addition, we design a spatiotemporal Gaussian compression scheme to efficiently compress dynamic Gaussian sequences by leveraging inherent spatial sparsity and temporal redundancy. This produces a highly compact 4D scene that enables lightweight and efficient real-time novel view rendering.

We perform extensive experiments on multi-view dynamic video benchmarks spanning both real-world and synthetic data, demonstrating that Mono4D outperforms existing state-of-the-art approaches in visual fidelity, temporal consistency, and inference efficiency, as illustrated in Fig. 1. Mono4D also differs fundamentally from monocular dynamic 3DGS methods Park et al. (2025b); Liu et al. (2025a); Luo et al. (2026): these approaches fit each scene through iterative per-scene optimization, which is severely under-constrained under monocular input and prone to converging to degenerate geometry, whereas Mono4D reconstructs in a single feedforward pass. Our framework is compatible with any feedforward 3DGS backbone Mescheder et al. (2026); Yu et al. (2026); Li et al. (2026a), and 4D reconstruction performance can be readily improved as the underlying feedforward 3DGS model advances, highlighting the generality of our static-dynamic decoupling paradigm.

Table 1: **Capability comparison.** Core feature comparison across existing 4D NVS paradigms.

Capability	Camera-controlled video generation	Per-scene optimization	Feedforward 3D/4D model	**Ours**
Reconstructed 3D/4D representation	×	✓	✓	✓
Monocular video input only	✓	×	✓	✓
Dynamic novel view rendering	×	✓	×	✓

Overall, our work makes the following contributions:

- We propose Mono4D, a novel framework that lifts a pretrained single-image 3D reconstruction model to produce temporally coherent 4D reconstructions without additional training. Our static-dynamic decoupled paradigm effectively resolves the temporal inconsistencies and geometric distortions that arise when transferring feedforward 3D models to dynamic scenarios.
- We introduce a dynamic mask refinement strategy tailored for decoupled Gaussian reconstruction, which achieves clean separation between static and dynamic scene components and effectively mitigates boundary artifacts.
- We develop a spatiotemporal Gaussian compression scheme for 3DGS sequences, which yields a compact 4D scene representation and enables efficient real-time novel view rendering. Extensive experiments demonstrate that our method achieves superior performance in visual fidelity, temporal consistency, and inference efficiency compared with existing state-of-the-art methods.

2 Related Work

Optimization-based Novel View Synthesis. 3D Gaussian Splatting (3DGS) Kerbl et al. (2023) enables fast rendering with its explicit representation, spurring widespread research into dynamic 3DGS modeling, such as deformation-based modeling Yang et al. (2023); Wu et al. (2024); Shao et al. (2023); Bae et al. (2024), basis motion encoding Li et al. (2023); Lin et al. (2024), and motion clustering for compact representations Feng et al. (2024); Huang et al. (2024). Instant4D Luo et al. (2026) achieves rapid 4D Gaussian reconstruction, SplineGS Park et al. (2025b) models dynamic Gaussians via a motion-adaptive spline mechanism, and MoDGS Liu et al. (2025a) integrates depth priors into Gaussian reconstruction. However, most of these methods require multi-view inputs and perform poorly on in-the-wild monocular videos, with limited generalization.

Camera-conditioned video generation. Camera-controlled video generation has two main routes. The first route Ren et al. (2025); Yang et al. (2026); Cao et al. (2026); Team (2026); Yu et al. (2025); Lin et al. (2026b) uses a 3D foundation model Wang et al. (2025; 2026a); Lin et al. (2025); Keetha et al. (2026) to perform 3D reconstruction of input video, then renders along target camera trajectories and fills missing content via video generation models Blattmann et al. (2023); Yang et al. (2024); Kong et al. (2024); Wan et al. (2025). The second route Bai et al. (2025); Li et al. (2026b); He et al. (2025); Park et al. (2025a) feeds camera parameters into diffusion models as control signals via implicit embeddings, such as Plücker embeddings Sitzmann et al. (2021) or ControlNet Zhang et al. (2023). However, these methods ultimately lack explicit 3D representations, thus requiring separate inference for each new camera trajectory and producing inconsistent outputs.

Feedforward 3D reconstruction. Feedforward 3D reconstruction models represented by VGGT Wang et al. (2025) can estimate depth, point maps, and camera parameters, driving the transition of 3DGS from per-scene optimization to the feedforward paradigm. These methods Wang et al. (2026a); Lin et al. (2025); Wang et al. (2026b); Keetha et al. (2026); Liu et al. (2025c) achieve decent reconstruction quality on static scenes but cannot handle dynamic scenarios. Some approaches Mescheder et al. (2026); Yu et al. (2026); Li et al. (2026a) support small-angle view wandering from a single image, yet directly transferring them to dynamic scenes yields inter-frame inconsistency. Some 4D reconstruction works Lu et al. (2026); Hu et al. (2025); Zhang et al. (2026); Lin et al. (2026c); Gao et al. (2026); Huang et al. (2026) jointly model geometry and motion for dynamic scenes. Lyra Bahmani et al. (2026) achieves 3DGS generation via self-distillation from a video diffusion model. 4DGT Xu et al. (2025) trains a 4D Gaussian Transformer on monocular videos; MoVieS Lin et al. (2026a) adds motion modeling to 4DGS generation, and BulletTimer Ren et al. (2026) delivers feedforward bullet-time reconstruction for dynamic scenes. Compared with

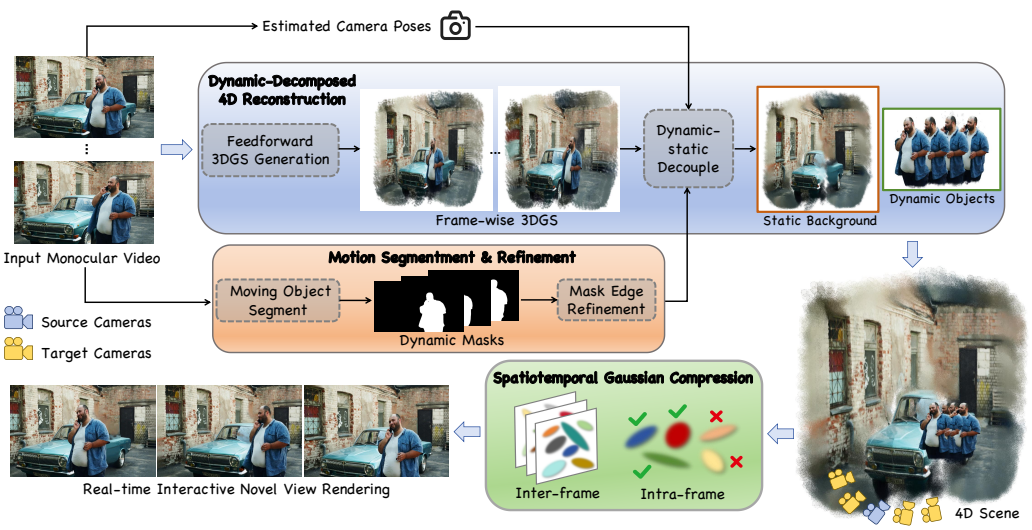


Figure 2: **Overview of Mono4D.** Starting with a monocular video, we first estimate dynamic masks and camera parameters. Our Dynamic-Decomposed 4D Reconstruction generates per-frame 3DGS via feedforward inference, then separates static background and dynamic objects by back-projecting Gaussians onto dynamic masks, and fuses static 3DGS across all frames into a unified background. The dynamic mask refinement strategy further calibrates the Gaussian points near mask edge. We further design Spatiotemporal Gaussian Compression to reduce intra-frame and inter-frame redundancy within the dynamic Gaussian sequence, enabling novel view rendering of the 4D scene.

our method, these approaches only support novel view rendering at a fixed frame and cannot render novel views as the scene evolves over time.

3 METHOD

An overview of Mono4D is provided in Fig. 2. Our goal is to reconstruct a 4D dynamic scene from monocular video input. We first introduce the feedforward 3DGS framework that serves as our backbone in Sec. 3.1, then present our dynamic-decomposition reconstruction in Sec. 3.2, mask edge refinement strategy in Sec. 3.3, and spatiotemporal Gaussian compression in Sec. 3.4.

3.1 PRELIMINARY: FEEDFORWARD 3D GAUSSIAN SPLATTING

Our method builds upon Sharp Mescheder et al. (2026), a representative feed-forward 3DGS backbone. It takes an RGB image $\mathbf{I} \in \mathbb{R}^{3\times H\times W}$ as input and outputs a set of 3D Gaussian primitives $\mathbf{G} \in \mathbb{R}^{K\times N}$. First, the monodepth module performs feature extraction and depth prediction. The image encoder produces intermediate feature maps, and the depth decoder predicts the depth map:

$$(\mathbf{f}_i)_{i\in\{1,\dots,4\}} = \varphi_{\text{enc}}(\mathbf{I}), \quad \hat{\mathbf{D}} = \varphi_{\text{dec}}\big((\mathbf{f}_i)_{i\in\{1,\dots,4\}}\big) \tag{1}$$

Sharp employs a two-layer depth representation $\hat{\mathbf{D}} \in \mathbb{R}^{2\times H\times W}$, where the first layer models visible surfaces and the second represents occluded regions. The predicted depth map is fed into a depth adjustment module, which yields the refined depth $\bar{\mathbf{D}}$ by alleviating the scale ambiguity. Then a set of base Gaussians $\mathbf{G}_0$ is initialized using the adjusted two-layer depth map $\bar{\mathbf{D}} \in \mathbb{R}^{2\times H\times W}$ together with the input image $\mathbf{I} \in \mathbb{R}^{3\times H\times W}$. A Gaussian decoder further refines all Gaussian attributes:

$$\Delta\mathbf{G} = \varphi_{\text{gauss}}\left((\mathbf{f}_i)_{i\in\{1,\dots,4\}}, \mathbf{I}\right). \tag{2}$$

The final Gaussian attributes $\mathbf{G} \in \mathbb{R}^{K\times 2\times H'\times W'}$ are produced by the Gaussian composer, which takes the base Gaussians $\mathbf{G}_0 \in \mathbb{R}^{K\times 2\times H'\times W'}$ and the Gaussian refinements $\Delta\mathbf{G} \in \mathbb{R}^{K\times 2\times H'\times W'}$ as input. The composition is performed via an attribute-specific activation function with a corresponding scale factor. Benefiting from the Gaussian attribute regression pipeline, high-resolution novel view synthesis is feasible in this purely feedforward framework.

3.2 DYNAMIC-DECOMPOSED 4D RECONSTRUCTION

Given a monocular video $\{\mathbf{I}_t\}_{t=1}^{T}$, we decompose the scene into static background and dynamic objects. We employ VGGT-Ω Wang et al. (2026a) as our camera predictor to estimate camera intrinsics and extrinsics $\{\mathbf{K}_t, \mathbf{T}_t\}_{t=1}^{T}$, providing camera poses for temporal alignment. Sharp Mescheder et al. (2026) is adopted as the feedforward 3DGS backbone to reconstruct per-frame Gaussian primitives. GMOS Xie et al. (2026) is applied for motion segmentation to obtain dynamic masks $\{\mathbf{M}_t^{\text{raw}}\}_{t=1}^{T}$, where $\mathbf{M}_t^{\text{raw}} \in \{0,1\}^{H\times W}$ indicates the pixel-wise motion label. The initial masks are refined through the strategy described in Sec. 3.3, yielding refined masks $\{\mathbf{M}_t\}_{t=1}^{T}$. Using these masks, we split each frame into foreground $\mathbf{I}_t^f = \mathbf{I}_t \odot \mathbf{M}_t$ and background $\mathbf{I}_t^b = \mathbf{I}_t \odot (1-\mathbf{M}_t)$, where $\odot$ denotes element-wise multiplication. For each frame t, we feed the monocular image into the feedforward 3DGS backbone to generate 3D Gaussians $\mathbf{G}_t = \{\mathbf{g}_i\}$, where each Gaussian $\mathbf{g}_i = (\mathbf{p}_i, \mathbf{c}_i, \mathbf{s}_i, \mathbf{r}_i, \alpha_i)$ is parameterized by 3D position $\mathbf{p}_i \in \mathbb{R}^3$, color $\mathbf{c}_i$, scale $\mathbf{s}_i$, rotation $\mathbf{r}_i$, and opacity α_i.

3D Gaussian assignment. To partition Gaussians into static and dynamic sets, we project each Gaussian's 3D center $\mathbf{p}_i$ onto the image plane using the estimated camera parameters:

$$(u_i, v_i) = \pi(\mathbf{K}_t, \mathbf{T}_t, \mathbf{p}_i) \tag{3}$$

where π denotes the perspective projection function. We then assign Gaussians based on the refined mask value at the projected location:

$$\mathcal{G}_d^t = \{\mathbf{g}_i \mid \mathbf{M}_t(u_i, v_i) > \tau_m\}, \quad \mathcal{G}_s^t = \{\mathbf{g}_i \mid \mathbf{M}_t(u_i, v_i) \le \tau_m\} \tag{4}$$

where $\tau_m = 0.5$ is the binary threshold. The foreground Gaussians $\mathcal{G}_d^t$ correspond to the image region $\mathbf{I}_t^f$, while the static Gaussians $\mathcal{G}_s^t$ correspond to $\mathbf{I}_t^b$.

World-coordinate alignment and static background fusion. Per-frame outputs from the 3DGS backbone lie in their respective local camera coordinates with inconsistent scales relative to the camera poses estimated by the camera predictor. To fix this scale mismatch, we compute a sequence-level $\mathrm{Sim}(3)$ transform to map the local 3DGS into the global coordinate. We solve for the similarity transform $(\mathbf{s}, \mathbf{R}_{\text{sim}}, \mathbf{t}_{\text{sim}})$ with static 3D-2D correspondences $\{\mathbf{p}_i^{\text{GS}}, \mathbf{x}_i\}_{i=1}^{N_p}$ derived from the back-projected depth of the 3DGS backbone:

$$(\mathbf{s}, \mathbf{R}_{\text{sim}}, \mathbf{t}_{\text{sim}}) = \arg\min_{\mathbf{s},\mathbf{R},\mathbf{t}} \sum_{i=1}^{N_p} \left\| \mathbf{s}\mathbf{R}\mathbf{p}_i^{\text{GS}} + \mathbf{t} - \mathbf{p}_i^{\text{CP}} \right\|_2^2 \tag{5}$$

where $\mathbf{p}_i^{\text{CP}} = \mathbf{T}_{\text{vcam},t}\, \mathbf{p}_i^{\text{cam}}$ denotes the world space 3D point transformed from camera coordinate point $\mathbf{p}_i^{\text{cam}}$ via the estimated extrinsics $\mathbf{T}_{\text{vcam},t}$. The solved $\mathrm{Sim}(3)$ transform reduces to the scale s alone ($\mathbf{R}_{\text{sim}} = \mathbf{I}$, $\mathbf{t}_{\text{sim}} = \mathbf{0}$), obtained in closed form as a median of depth ratios, which needs no initialization and is insensitive to depth-prediction failures and far-plane degeneracies. We transform background Gaussians outside the dynamic mask into the global world coordinate:

$$\mathbf{p}_{t,i}^{w} = \mathbf{s}\, \mathbf{R}_{\text{sim}}\, \mathbf{R}_{\text{cam},t}\, \mathbf{p}_{t,i}^{c} + \mathbf{t}_{\text{sim}} + \mathbf{t}_{\text{cam},t} \tag{6}$$

where $\mathbf{R}_{\text{cam},t}$ and $\mathbf{t}_{\text{cam},t}$ represent the estimated extrinsics rotation and translation. The transformed Gaussian covariance is computed as $\mathbf{\Sigma}_i^{w} = \mathbf{s}^2\, \mathbf{R}_{\text{sim}}\, \mathbf{R}_{\text{cam},t}\, \mathbf{\Sigma}_i^{c}\, \mathbf{R}_{\text{cam},t}^{\top} \mathbf{R}_{\text{sim}}^{\top}$.

To establish cross-frame correspondences and aggregate appearance from multiple views, we adopt LoMa Nordström et al. (2026) as a feature matching module. Given frame pairs (t, t'), LoMa establishes dense keypoint correspondences between frames:

$$\{\mathbf{k}_t^{(m)}, \mathbf{k}_{t'}^{(m)}\}_{m=1}^{M} = \text{LoMa}\,(\mathbf{I}_t, \mathbf{I}_{t'}; \theta_{\text{loma}}) \tag{7}$$

For each Gaussian $\mathbf{g}_{t,i} \in \mathcal{G}_s^t$ at frame t, we find its nearest keypoint $\mathbf{k}_t^{(m)}$ and fetch the paired $\mathbf{k}_{t'}^{(m)}$ from frame t'. This keypoint is back-projected to world coordinates, and we construct the full static background by aggregating background Gaussians over the video via voxel-based spatial fusion:

$$\mathcal{G}_s = \text{VoxelFusion}\left(\bigcup_{t=1}^{T} \left\{ \mathbf{g}_{t,i}^{w} \mid \mathbf{M}_t(u_i, v_i) \le \tau_m \right\} \right) \tag{8}$$

where VoxelFusion$(\cdot)$ performs duplicate association and color aggregation. Within each voxel, redundant cross-frame Gaussians are merged using maximum scale and confidence-weighted averaging for color and opacity, where voxels appearing in at least $n_{\min} = 2$ frames are kept. This geometry-aligned fusion produces a complete, temporally consistent static background from all visible regions of the sequence. More details are provided in the Appendix.

3.3 MASK EDGE REFINEMENT

Motion segmentation masks produced by off-the-shelf models often have coarse boundaries, which degrade reconstruction quality. These coarse edges misassign dynamic Gaussian primitives to static background regions and cause floaters along static-dynamic boundaries.

Motion mask matting. We first employ VideoMaMa Lim et al. (2026), a mask-guided video matting model robust to coarse input masks, for mask sequence matting to refine initial dynamic masks:

$$\mathbf{M}_t^{\text{refine}} = \text{VideoMaMa}\left(\mathbf{I}_t, \mathbf{M}_t^{\text{raw}}\right), \quad \mathbf{M}_t^{\text{refine}} \in [0,1]^{H\times W} \tag{9}$$

VideoMaMa produces alpha mattes with fine-grained boundary details by exploiting video matting priors, where $\mathbf{M}_t^{\text{refine}}$ is the output alpha matte with improved boundary precision, enabling more accurate static-dynamic separation.

Adaptive dynamic mask processing. Instead of hard binary thresholding, we retain continuous alpha values for smooth boundary transitions. We apply $k \times k$ max-pooling to dilate the mask and recover motion-blurred or occluded edge pixels: $\mathbf{M}_t^{\text{pool}}(u,v) = \max_{(i,j)\in\mathcal{N}_k(u,v)} \mathbf{M}_t^{\text{refine}}(i,j)$. Each 3D Gaussian center $\mathbf{p}_i$ is projected to the 2D plane: $\boldsymbol{\pi}_i = \pi(\mathbf{K}_t, \mathbf{T}_t, \mathbf{p}_i)$, and we compute the soft mask weight by integrating $\mathbf{M}_t^{\text{pool}}$ over the Gaussian's projected region Ω_i:

$$w_i = \frac{\int_{\Omega_i} \mathbf{M}_t^{\text{pool}}(\mathbf{x})\, G_i(\mathbf{x})\, d\mathbf{x}}{\int_{\Omega_i} G_i(\mathbf{x})\, d\mathbf{x}} \tag{10}$$

where $G_i(\mathbf{x})$ is the projected 2D footprint of the 3D Gaussian at pixel $\mathbf{x}$. Background Gaussians near mask boundaries are prone to foreground artifacts, so we filter unreliable background Gaussians whose projected center distance to the dynamic boundary $\partial\mathcal{M}_t$ is below the threshold $\delta_b = 5$ pixels:

$$\mathcal{G}_s^{t,\text{filt}} = \left\{\mathbf{g}_i \in \mathcal{G}_s^t \mid d(\boldsymbol{\pi}_i, \partial\mathcal{M}_t) > \delta_b\right\} \tag{11}$$

Finally, the refined motion mask is adopted for static-dynamic scene decomposition in Sec. 3.2 to partition Gaussians into dynamic and static subsets.

3.4 SPATIOTEMPORAL GAUSSIAN COMPRESSION

For compact dynamic scene representation, we introduce a two-stage compression: intra-frame pruning and inter-frame canonical encoding.

Intra-frame compression. For each frame t, we compute an importance score for each Gaussian by combining opacity and spatial scale. Specifically, the importance of Gaussian i is defined as:

$$\eta_i = \alpha_i \cdot \bar{\mathbf{s}}_i, \quad \bar{\mathbf{s}}_i = \frac{1}{3}\sum_{j=1}^{3} e^{s_{i,j}} \tag{12}$$

where α_i is the sigmoid-activated opacity and $e^{s_{i,j}}$ is the j-th scale axis after exponentiation. We retain the top-r fraction of Gaussians ranked by importance, and r is the Gaussian retention ratio:

$$\mathcal{G}_d^{t,\text{pruned}} = \left\{\mathbf{g}_i \in \mathcal{G}_d^t \mid \eta_i \geq \tau_r\right\} \tag{13}$$

where τ_r is the $(1-r)$-th quantile of importance scores. We set $r = 0.6$ for large storage reduction with negligible perceptual loss, with full ablation on the retention ratio in the Appendix.

Inter-frame compression via canonical clustering. To exploit temporal redundancy across frames, we construct a canonical Gaussian representation through spatial clustering. We designate frame 0 as the canonical frame and perform K-means clustering on the positions of its foreground Gaussians $\{\mathbf{g}_i^0 \in \mathcal{G}_d^{0,\text{pruned}}\}$, with centroids initialized by uniform sampling, to obtain $K = 64$ canonical clusters. Frame 0 acts only as a spatial anchor, so selecting a non-optimal canonical frame increases the residual magnitude but does not invalidate the reconstruction:

$$\mathcal{C} = \{C_k\}_{k=1}^{K}, \quad C_k = \left\{\mathbf{g}_i^0 \mid \|\mathbf{p}_i^0 - \boldsymbol{\mu}_k\|_2 < \|\mathbf{p}_i^0 - \boldsymbol{\mu}_j\|_2, \forall j \neq k\right\} \tag{14}$$

where $\mathcal{C}, C_k, \mathbf{g}_i^0, \mathbf{p}_i^0, \boldsymbol{\mu}_k, j$ denote the set of all canonical clusters, the k-th canonical cluster, the i-th Gaussian in the canonical frame, its 3D position, the centroid of k-th cluster, and cluster index,

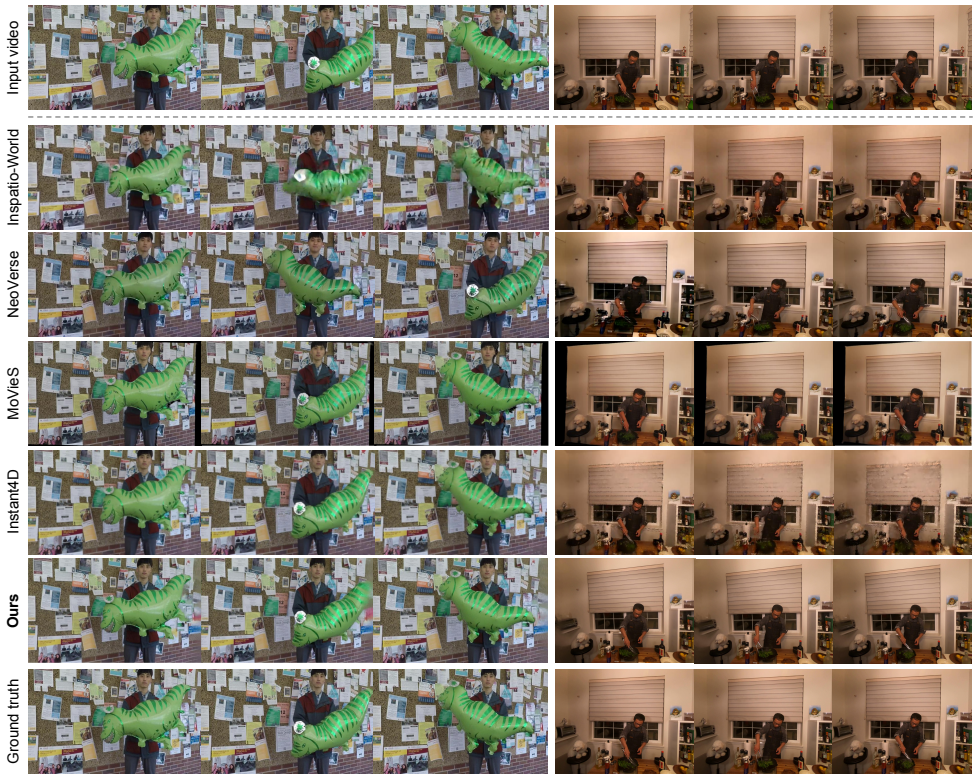


Figure 3: Qualitative Comparison on Neural-3D-Video and NVIDIA-Dynamic-Scene datasets.

respectively. We define the temporal association set $\mathcal{A}_t = \{(i,j) \mid \mathbf{g}_i^0 \leftrightarrow \mathbf{g}_j^t\}$, where (i,j) matches canonical Gaussian $\mathbf{g}_i^0$ (frame 0) to Gaussian $\mathbf{g}_j^t$ (frame t) built via LoMa. Cluster-k matched pairs are $\mathcal{A}_{t,k} = \{(i,j) \in \mathcal{A}_t \mid k(i) = k\}$, where $k(i)$ gives the cluster index of canonical Gaussian i. We encode inter-frame motion using cluster-level rigid transformations; for frame t and cluster k we solve the weighted orthogonal Procrustes problem over matched pairs to estimate rigid alignment:

$$\mathbf{T}_k^t = \arg\min_{\mathbf{R}_k^t, \mathbf{t}_k^t} \sum_{(i,j)\in\mathcal{A}_{t,k}} w_{ij} \left\| \mathbf{p}_j^t - \mathbf{R}_k^t \mathbf{p}_i^0 - \mathbf{t}_k^t \right\|_2^2 \tag{15}$$

where w_{ij} is the correspondence confidence weight, $k(i)$ is the cluster index of canonical Gaussian i, $\mathbf{R}_k^t \in \mathrm{SO}(3)$ and $\mathbf{t}_k^t \in \mathbb{R}^3$. Gaussians in frame t are then reconstructed as

$$\mathbf{p}_i^t = \mathbf{R}_{k(i)}^t \mathbf{p}_i^0 + \mathbf{t}_{k(i)}^t + \boldsymbol{\delta}_i^t,$$

where $\boldsymbol{\delta}_i^t$ is the per-Gaussian offset from transformed canonical positions. Before compression, all dynamic Gaussians are transformed into the world coordinate system, ensuring that cluster-level rigid motion $\mathbf{T}_k^t$ encodes only object motion, not camera motion.

Frame-wise decompression for rendering. At inference, we decompress and render frame by frame. For frame t, we fetch canonical Gaussians from clusters with valid activation $a_t^k = 1$, apply cluster-level transformations $\{\mathbf{T}_k^t\}$, and recover positions with per-Gaussian residuals. This frame-wise decompression scheme avoids loading the full dynamic sequence all at once, further lowering the memory consumption. Additional details of Gaussian compression are shown in the Appendix.

4 Experiments

4.1 Datasets and evaluation metrics

We conduct comprehensive experiments on real-world and synthetic multi-view video benchmarks for novel view synthesis (NVS). Specifically, real-world test datasets include NVIDIA-Dynamic-Scene Yoon et al. (2020) and Neural 3D Video Li et al. (2022), while synthetic benchmark consists

Table 2: Quantitative results on Neural-3D-Video and NVIDIA-Dynamic-Scene datasets. Best and second-best in each column are highlighted.

Method	Neural-3D-Video					NVIDIA-Dynamic-Scene				
	PSNR ↑	SSIM ↑	LPIPS ↓	T-LPIPS ↓	DISTS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	T-LPIPS ↓	DISTS ↓
Inspatio-World	18.28	0.667	0.218	0.106	0.143	15.58	0.469	0.374	0.282	0.128
NeoVerse	17.05	0.549	0.305	0.336	0.196	19.52	0.689	0.116	0.147	0.099
MoVieS	16.55	0.437	0.391	0.179	0.228	18.18	0.647	0.079	0.129	0.062
Instant4D	19.86	0.684	0.315	0.196	0.246	20.48	0.766	0.128	0.159	0.084
Ours	21.30	0.719	0.143	0.068	0.051	20.19	0.697	0.068	0.051	0.054

Table 4: Storage comparison with existing 4D reconstruction methods.

Method	SplineGS	Lyra	Instant4D	MoVieS	Ours
Size of 4D Scene	135 MB	2.70 GB	174 MB	641 MB	152 MB

of StereoDataset Cui et al. (2026). We adopt five quantitative metrics for evaluation: PSNR, SSIM, LPIPS Zhang et al. (2018), T-LPIPS Chu et al. (2020), and DISTS Ding et al. (2020). PSNR and SSIM are handcrafted pixel and structural metrics, while LPIPS and DISTS are perceptual metrics, and T-LPIPS further evaluates temporal consistency across video frames.

4.2 Baseline Comparisons

We compare Mono4D against three types of baseline approaches. The first is per-scene optimization methods, and we select Instant4D Luo et al. (2026). The second refers to feedforward 4D reconstruction and we adopt MoVieS Lin et al. (2026a). The third includes camera-controlled video generation, with NeoVerse Yang et al. (2026) and Inspatio-World Team (2026) selected for evaluation.

4.3 Qualitative Comparison

We visualize novel view results in Fig. 3. The first row shows input frames, middle rows present novel view outputs of different methods, and the last row is ground truth. NeoVerse and Inspatio-World suffer from degraded textures and inconsistencies against the input video owing to generation randomness. MoVieS fails to recover textures within occluded regions and produces distorted moving objects. Instant4D struggles to reconstruct scene textures correctly. In contrast, Mono4D properly reconstructs both static backgrounds and dynamic objects, and produces superior fidelity renderings while maintaining temporal consistency for nearby novel views in dynamic scenes. More visualizations are provided in the Appendix. We refer readers to the supplementary materials for video results.

Table 3: Quantitative results on StereoDataset.

Method	StereoDataset				
	PSNR↑	SSIM↑	LPIPS↓	T-LPIPS↓	DISTS↓
Inspatio-World	17.19	0.625	0.390	0.235	0.244
MoVieS	18.74	0.708	0.235	0.178	0.154
NeoVerse	14.25	0.654	0.527	0.270	0.186
Instant4D	20.97	0.720	0.240	0.160	0.212
Ours	23.68	0.735	0.059	0.054	0.046

4.4 Quantitative Comparison

Quantitative results are presented in Tab. 2 and Tab. 3. PSNR and SSIM are sensitive to subtle spatial misalignments and yield inflated scores for over-smoothed outputs. This explains why Instant4D achieves strong PSNR and SSIM despite its heavily blurred textures. Perceptual metrics including LPIPS and DISTS tolerate minor geometric shifts and align better with human visual perception, as validated in Sharp Mescheder et al. (2026). We further compare the storage of reconstructed 4D scenes in Tab. 4. This evaluation is conducted on the Vista4D evaluation dataset Lin et al. (2026b) with 20 sampled scenes; each video contains 49 frames. Mono4D achieves favorable storage efficiency; its scene size is slightly larger than SplineGS while outperforming the remaining methods. Overall, our method outperforms baseline approaches on most evaluation metrics.



Figure 4: Quantitative ablation results on Vista4D evaluation dataset.

4.5 Ablation Study

We conduct quantitative ablation evaluations in Tab. 6, and further present qualitative ablation in Fig. 4. Without our dynamic-decomposed reconstruction scheme, independently running a feedforward 3D model frame by frame introduces noticeable inter-frame artifacts at static-dynamic boundaries. Such temporal artifacts degrade perceptual metrics substantially relative to our full method, which preserves favorable inter-frame consistency. When the edge mask refinement module is removed, boundaries of moving objects such as hair strands are misassigned to the background. Without background fusion, blurry textures appear in static background regions, while our full method preserves consistent reconstruction of background textures. We also perform ablation studies for our spatiotemporal Gaussian compression in Tab. 5. We select the first 50 frames of each scene from Neural-3D-Video. Per-frame 3DGS inference yields the worst perceptual metrics, consuming 3.20 GB of storage. The uncompressed baseline occupies 767 MB, achieving LPIPS of 0.139, T-LPIPS of 0.059, and DISTS of 0.046. In comparison, our full method reduces the storage to 152 MB at LPIPS 0.143, T-LPIPS 0.068, and DISTS 0.051. This demonstrates that our compression strategy drastically cuts scene storage with a modest perceptual-quality trade-off. Additional ablation studies are provided in the Appendix.

Table 5: Ablation study on compression.

Method	Scene Size	LPIPS↓	T-LPIPS↓	DISTS↓
Per-frame 3DGS	3.20 GB	0.243	0.315	0.176
w/o Compression	767 MB	0.139	0.059	0.046
Ours-full	152 MB	0.143	0.068	0.051

5 Conclusion

We present Mono4D, a framework that lifts a pretrained 3D reconstruction model into a temporally coherent 4D reconstruction framework for monocular video inputs without additional training or per-scene optimization. Unlike existing methods that either demand per-scene optimization or suffer from inherent 2D constraints within video generation models, Mono4D can build upon any feedforward 3DGS backbone and adopts a static-dynamic decoupled paradigm. By modeling stationary backgrounds and moving foreground objects separately, our method effectively mitigates the temporal inconsistency and geometric drift that arise when feedforward 3D models are directly applied to dynamic scenarios. We further incorporate an edge-aware mask refinement strategy to correct coarse segmentation boundaries and suppress edge artifacts around fine structures such as hair and clothing. We also propose a spatiotemporal Gaussian compression scheme that leverages spatial sparsity and temporal redundancy within dynamic sequences, producing a compact 4D scene representation for efficient rendering with low memory consumption. Extensive experiments on NVS benchmarks show that Mono4D outperforms existing approaches regarding visual fidelity, temporal coherence and inference efficiency. Our framework is compatible with any feedforward 3DGS backbone, demonstrating the potential applicability of the static-dynamic decoupling paradigm beyond a single model. We hope that Mono4D can serve as a practical foundation for immersive content-creation pipelines, as well as for future research on dynamic 4D scene modeling and monocular novel view synthesis.

Table 6: Ablation study on Neural-3D-Video.

Method	LPIPS ↓	T-LPIPS ↓	DISTS ↓
w/o dynamic-decomposed	0.243	0.315	0.176
w/o edge mask refinement	0.189	0.176	0.082
w/o LoMa completion	0.205	0.178	0.134
Ours-full	0.143	0.068	0.051

AI-Assisted Statement

During this work, large language models assisted in experimental design and code prototyping. Large language models were also used for manuscript wording refinement, readability improvement, and research narrative structuring. However, none of the core algorithmic designs, technical insights, experimental validations, or final conclusions were directly produced by large language models. All AI-suggested code drafts and text outputs were thoroughly inspected, modified, and validated by the authors, who take full responsibility for all claims and content in this paper.

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 (ECCV)*, 2024.

Sherwin Bahmani, Tianchang Shen, Jiawei Ren, Jiahui Huang, Yifeng Jiang, Haithem Turki, Andrea Tagliasacchi, David Lindell, Zan Gojcic, Sanja Fidler, et al. Lyra: Generative 3d scene reconstruction via video diffusion model self-distillation. In *International Conference on Learning Representations*, volume 2026, pp. 82850–82880, 2026.

Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. Recammaster: Camera-controlled generative rendering from a single video. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14834–14844. IEEE, 2025.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets, 2023. URL https://arxiv.org/abs/2311.15127.

Wei Cao, Hao Zhang, Fengrui Tian, Yulun Wu, Yingying Li, Shenlong Wang, Ning Yu, and Yaoyao Liu. Freeorbit4d: Training-free arbitrary camera redirection for monocular videos via foreground-complete 4d reconstruction. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–12, 2026.

Zhaoxi Chen, Tianqi Liu, Long Zhuo, Jiawei Ren, Zeng Tao, He Zhu, Fangzhou Hong, Liang Pan, and Ziwei Liu. 4dnex: Feed-forward 4d generative modeling made easy. *arXiv preprint arXiv:2508.13154*, 2025.

Mengyu Chu, You Xie, Jonas Mayer, Laura Leal-Taixé, and Nils Thuerey. Learning temporal coherence via self-supervision for gan-based video generation. *ACM Transactions on Graphics (TOG)*, 39(4):75–1, 2020.

Yangzhi Cui, Feng Qiao, and Nathan Jacobs. Stereogenbench: A synthetic multi-camera benchmark for stereo generation under controlled baseline regimes. *arXiv preprint arXiv:2605.23237*, 2026.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *CoRR*, abs/2004.07728, 2020. URL https://arxiv.org/abs/2004.07728.

Qi-Yuan Feng, Hao-Xiang Chen, Qun-Ce Xu, and Tai-Jiang Mu. Sls4d: Sparse latent space for 4d novel view synthesis. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–14, 2024. doi: 10.1109/TVCG.2024.3429421.

Rongxin Gao, Yuzhi Huang, Dongxuan Liu, Chu Li, Zhenye Wang, Jie Wu, Shuzhao Xie, Jingyan Jiang, Xinghao Ding, Xiaotong Tu, et al. Dyntrace: Tracking dynamic object evidence for 4d spatio-temporal reasoning in mllms. *arXiv preprint arXiv:2607.12503*, 2026.

Hao He, Ceyuan Yang, Shanchuan Lin, Yinghao Xu, Meng Wei, Liangke Gui, Qi Zhao, Gordon Wetzstein, Lu Jiang, and Hongsheng Li. Cameractrl ii: Dynamic scene exploration via camera-controlled video diffusion models, 2025. URL https://arxiv.org/abs/2503.10592.

Yu Hu, Chong Cheng, Sicheng Yu, Xiaoyang Guo, and Hao Wang. Vggt4d: Mining motion cues in visual geometry transformers for 4d scene reconstruction, 2025. URL https://arxiv.org/abs/2511.19971.

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.

Yuzhi Huang, Kairun Wen, Rongxin Gao, Dongxuan Liu, Yibin Lou, Jie Wu, Jing Xu, Jian Zhang, Zheng Yang, Yunlong Lin, et al. Thinking in dynamics: How multimodal large language models perceive, track, and reason dynamics in physical 4d world. *arXiv preprint arXiv:2603.12746*, 2026.

Yudong Jin, Tao Xie, Qihang Zhang, Zehong Shen, Zhen Xu, Yujun Shen, Hujun Bao, Xiaowei Zhou, and Yinghao Xu. 4danyone: Create anyone in 4d from a casual monocular video. *arXiv preprint arXiv:2608.20335*, 2026.

Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. MapAnything: Universal feed-forward metric 3D reconstruction. In *International Conference on 3D Vision (3DV)*. IEEE, 2026.

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

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

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*, 2024.

Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5511–5521. IEEE, 2022.

Yinglong Li, Donghui Shen, Xiaoyu Zhang, Zhichao Ye, Hongyu Wu, Aimin Hao, Guofeng Zhang, and Haomin Liu. Querysplat: Decoupling geometry and appearance representations in 3dgs prediction. *arXiv preprint arXiv:2608.01186*, 2026a.

Yixuan Li, Yanhong Zeng, Ka Leong Cheng, Jiayi Zhu, Hanlin Wang, Wen Wang, Yihao Meng, Hao Ouyang, Qiuyu Wang, Yue Yu, ZiDong Wang, Yiyuan Zhang, Yujun Shen, and Dahua Lin. Cameraanything: Refilming videos with arbitrary camera control. In *European Conference on Computer Vision (ECCV)*, 2026b.

Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. *arXiv preprint arXiv:2312.16812*, 2023.

Sangbeom Lim, Seoung Wug Oh, Jiahui Huang, Heeji Yoon, Seungryong Kim, and Joon-Young Lee. Videomama: Mask-guided video matting via generative prior. *arXiv preprint arXiv:2601.14255*, 2026.

Chenguo Lin, Yuchen Lin, Panwang Pan, Yifan Yu, Tao Hu, Honglei Yan, Katerina Fragkiadaki, and Yadong Mu. Movies: Motion-aware 4d dynamic view synthesis in one second. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, 2026a.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Kuan Heng Lin, Zhizheng Liu, Pablo Salamanca, Yash Kant, Ryan Burgert, Yuancheng Xu, Koichi Namekata, Yiwei Zhao, Bolei Zhou, Micah Goldblum, et al. Vista4d: Video reshooting with 4d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 32671–32682, 2026b.

Shing Ho J Lin, Wenzhao Zheng, Dong Zhuo, Yuqi Wu, Jie Zhou, and Jiwen Lu. Sm4rt: Learning structured motion geometry for 4d reconstruction. *arXiv preprint arXiv:2607.22534*, 2026c.

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

Qingming Liu, Yuan Liu, Jiepeng Wang, Xianqiang Lyu, Peng Wang, Wenping Wang, and Junhui Hou. Modgs: Dynamic gaussian splatting from casually-captured monocular videos with depth priors. In *International Conference on Learning Representations*, volume 2025, pp. 97048–97074, 2025a.

Tianqi Liu, Zihao Huang, Zhaoxi Chen, Guangcong Wang, Shoukang Hu, Liao Shen, Huiqiang Sun, Zhiguo Cao, Wei Li, and Ziwei Liu. Free4d: Tuning-free 4d scene generation with spatial-temporal consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25571–25582, October 2025b.

Yifan Liu, Zhiyuan Min, Zhenwei Wang, Junta Wu, Tengfei Wang, Yixuan Yuan, Yawei Luo, and Chunchao Guo. Worldmirror: Universal 3d world reconstruction with any-prior prompting. *arXiv preprint arXiv:2510.10726*, 2025c.

Jiahao Lu, Jiayi Xu, Wenbo Hu, Ruijie Zhu, Chengfeng Zhao, Sai-Kit Yeung, Ying Shan, and Yuan Liu. Track4world: Feedforward world-centric dense 3d tracking of all pixels. *arXiv preprint arXiv:2603.02573*, 2026.

Zhanpeng Luo, Haoxi Ran, and Li Lu. Instant4d: 4d gaussian splatting in minutes. *Advances in neural information processing systems*, 38:29953–29972, 2026.

Lars Mescheder, Wei Dong, Shiwei Li, Xuyang Bai, Marcel Santos, Peiyun Hu, Bruno Lecouat, Mingmin Zhen, Amaël Delaunoy, Tian Fang, et al. Sharp monocular view synthesis in less than a second. In *International Conference on Learning Representations*, volume 2026, pp. 24192–24230, 2026.

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

David Nordström, Johan Edstedt, Georg Bökman, Jonathan Astermark, Anders Heyden, Viktor Larsson, Mårten Wadenbäck, Michael Felsberg, and Fredrik Kahl. Loma: Local feature matching revisited. *arXiv preprint arXiv:2604.04931*, 2026.

Avinash Paliwal, Adithya Iyer, Shivin Yadav, Muhammad Afridi, and Midhun Harikumar. Reshoot-anything: A self-supervised model for in-the-wild video reshooting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11596–11606, 2026.

Byeongjun Park, Byung-Hoon Kim, Hyungjin Chung, and Jong Chul Ye. Redirector: Creating any-length video retakes with rotary camera encoding. *arXiv preprint arXiv:2511.19827*, 2025a.

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. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26866–26875. IEEE, 2025b.

Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. *ICCV*, 2021.

Jiawei Ren, Ashkan Mirzaei, Antonio Torralba, Ziwei Liu, Igor Gilitschenski, Sanja Fidler, Cengiz Oztireli, Huan Ling, Zan Gojcic, Jiahui Huang, et al. Feed-forward bullet-time reconstruction of dynamic scenes from monocular videos. *Advances in Neural Information Processing Systems*, 38:151718–151746, 2026.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Ruizhi Shao, Zerong Zheng, Hanzhang Tu, Boning Liu, Hongwen Zhang, and Yebin Liu. Tensor4d: Efficient neural 4d decomposition for high-fidelity dynamic reconstruction and rendering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

Vincent Sitzmann, Semon Rezchikov, William T. Freeman, Joshua B. Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. In *Proc. NeurIPS*, 2021.

Chenxi Song, Yanming Yang, Tong Zhao, Ruibo Li, and Chi Zhang. Taming video models for 3d and 4d generation via zero-shot camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40352–40363, 2026.

Jiakai Sun, Han Jiao, Guangyuan Li, Zhanjie Zhang, Lei Zhao, and Wei Xing. 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 (CVPR)*, pp. 20675–20685, June 2024.

InSpatio Team. Inspatio-world: A real-time 4d world simulator via spatiotemporal autoregressive modeling, 2026.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306. IEEE, 2025.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-ω. *arXiv preprint arXiv:2605.15195*, 2026a.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *International Conference on Learning Representations*, volume 2026, pp. 10481–10497, 2026b.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20310–20320. IEEE, 2024.

Junyu Xie, Tengda Han, Weidi Xie, and Andrew Zisserman. Gmos: Grounding moving object segmentation in 3d space and time. *arXiv preprint arXiv:2605.30352*, 2026.

Zhen Xu, Zhengqin Li, Zhao Dong, Xiaowei Zhou, Richard Newcombe, and Zhaoyang Lv. 4dgt: Learning a 4d gaussian transformer using real-world monocular videos. *arXiv preprint arXiv:2506.08015*, 2025.

Yuxue Yang, Lue Fan, Ziqi Shi, Junran Peng, Feng Wang, and Zhaoxiang Zhang. Neoverse: Enhancing 4d world model with in-the-wild monocular videos. *arXiv preprint arXiv:2601.00393*, 2026.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. *arXiv preprint arXiv:2309.13101*, 2023.

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, 2020. URL https://arxiv.org/abs/2004.01294.

Hao Yu, Haotong Lin, Jiawei Wang, Jiaxin Li, Yida Wang, Xueyang Zhang, Yue Wang, Xiaowei Zhou, Ruizhen Hu, and Sida Peng. Infinidepth: Arbitrary-resolution and fine-grained depth estimation with neural implicit fields. *arXiv preprint arXiv:2601.03252*, 2026.

Mark Yu, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 100–111. IEEE, 2025.

Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K Barral, Raia Hadsell, et al. Efficiently reconstructing dynamic scenes one d4rt at a time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7382–7392, 2026.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3813–3824. IEEE, 2023.

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

Haiyang Zhou, Wangbo Yu, Chaoran Feng, Xunyu Zhou, Yonghong Tian, and Li Yuan. Uniworld-view: Large-baseline view synthesis via video diffusion models. *arXiv preprint arXiv:2608.04701*, 2026.

A Supplementary Video Demo

Supplementary video demos are provided to visually present qualitative results and demonstrate the superior viewing experience. The demo input monocular videos are collected from NeoVerse Yang et al. (2026), ReCamMaster Bai et al. (2025), Vista4D evaluation dataset Lin et al. (2026b), 4DAnyone Jin et al. (2026), AIGC-generated videos, and online-sourced data. Our supplementary videos include qualitative comparisons against baseline approaches, as well as dynamic novel view rendering results of Mono4D under diverse in-the-wild monocular video inputs. These videos showcase our static-dynamic decomposition and novel view synthesis results, demonstrating the superior temporal consistency and rendering quality of our method compared to baseline approaches.

B Additional Baseline Details

We also incorporate additional baselines across three categories for a comprehensive evaluation: Lyra represents feedforward 4D reconstruction, which distills implicit 3D knowledge from video diffusion into explicit 3D Gaussian representations and supports static and dynamic scene generation. ReCamMaster is a camera-controlled video generation model, which adopts a distinctive frame-dimension conditioning scheme to preserve cross-frame appearance and temporal-dynamic synchronization. SplineGS adopts per-scene 4DGS optimization, employing motion-adaptive splines to model continuous Gaussian motion trajectories for dynamic scenes.

C Additional Evaluation

Qualitative comparison on the NVS benchmark. Additional qualitative results on Neural-3D-Video and NVIDIA-Dynamic-Scene are provided in Fig. 5, while results on StereoDataset are shown in Fig. 6. As shown in these figures, our approach consistently yields high-fidelity textures and temporally coherent novel-view renderings compared to baseline methods.

Qualitative and quantitative comparisons on the Vista4D evaluation dataset. We also provide qualitative comparisons on the in-the-wild monocular Vista4D evaluation dataset, as shown in Fig. 7. This dataset contains 110 captured monocular videos without synchronized multi-view data, along with camera parameters and precomputed dynamic masks, though we only utilize the RGB videos as input for evaluation. We incorporate additional baselines for this evaluation, and our approach still achieves substantially better visual quality than baseline methods. We further conduct inference speed comparisons on the Vista4D evaluation dataset in Tab. 7, where each video consists of exactly 49 frames. Even with preprocessing for dynamic masks and camera parameters included, our method completes inference in approximately 10 minutes, which is comparable to MoVieS, the fastest feedforward 4D reconstruction baseline.

Table 7: Inference time comparison.

Method	Inspatio-World	NeoVerse	Lyra	MoVieS	Instant4D	SplineGS	RecamMaster	Ours
Inference Time	572s	1862s	3.1h	**495s**	3244s	20.5h	1340s	603s

Qualitative ablation on spatiotemporal Gaussian compression. We present visual comparisons of spatiotemporal Gaussian compression (with and without it) on the Vista4D evaluation dataset, as shown in Fig. 8. Our compression strategy introduces a modest perceptual quality trade-off in rendering, with quality differences barely perceptible to the human eye and only visible in fine structures such as hair strands, while reducing the storage footprint of the dynamic Gaussian sequence to roughly one-fifth of the original size.

D Evaluation Details

In evaluations on the NVS benchmarks including Neural-3D-Video, NVIDIA-Dynamic-Scene, and StereoDataset, we use the synchronized multi-view videos and camera parameters provided by each

dataset. All evaluations are conducted on a single NVIDIA H20 GPU. The original pipelines of Instant4D and MoVieS only support novel view synthesis on isolated static frames, so we extend them to dynamic settings to achieve novel view rendering with video playing. For each scene in our test datasets, we select one viewpoint as input and evaluate rendering quality on the four nearest neighboring viewpoints. For Neural-3D-Video, we evaluate on all 6 scenes, using the first 100 frames from each 300-frame video. We take cam07 among its 21 available views as the input view, and use cam05, cam06, cam08, and cam09 for evaluation. For NVIDIA-Dynamic-Scene, we use all 9 scenes and their full video frames. We select cam03 from its 12 views as the input, and evaluate on cam01, cam02, cam04, and cam05. For StereoDataset, which contains over 7000 scenes, we sample 20 scenes and use all video frames for evaluation. We set cam02 out of its 6 views as the input view, with cam00, cam01, cam03, and cam04 serving as evaluation views.

Beyond conventional pixel-wise metrics such as PSNR and SSIM, which are sensitive to subtle spatial misalignments, we mainly adopt three perceptual metrics for quantitative evaluation: LPIPS Zhang et al. (2018), T-LPIPS Chu et al. (2020), and DISTS Ding et al. (2020). LPIPS measures the feature-space distance between the predicted image $\hat{I}$ and ground truth image I:

$$\text{LPIPS}(\hat{I}, I) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} \left| w_l \odot \left(\phi_l(\hat{I}) * h, w - \phi_l(I) * h, w \right) \right|_2^2, \tag{16}$$

where ϕ_l denotes the feature map extracted from the l-th layer of a pre-trained network, w_l is the learned per-channel weight, and H_l, W_l are the spatial dimensions of the feature map.

T-LPIPS further extends LPIPS to quantify temporal consistency between adjacent frames, capturing inter-frame temporal artifacts. Given consecutive predicted frames $\hat{I}_t, \hat{I}_{t+1}$ and corresponding ground truth frames I_t, I_{t+1}, it computes the distance between frame-wise feature differences and averages over all adjacent frame pairs in a video:

$$\text{T-LPIPS} = \sum_l \frac{1}{H_l W_l} \sum_{h,w} \left| w_l \odot \left[\left(\phi_l(\hat{I}_{t+1}) - \phi_l(\hat{I}_t) \right) - \left(\phi_l(I_{t+1}) - \phi_l(I_t) \right) \right] \right|_2^2. \tag{17}$$

DISTS jointly evaluates structural and textural similarity by combining Pearson correlations between local mean and standard deviation maps of deep features extracted within spatial sliding windows:

$$\text{DISTS}(\hat{I}, I) = \sum_l \left(\alpha_l \cdot \left(1 - \text{corr}(\mu_{\phi_l(\hat{I})}, \mu_{\phi_l(I)}) \right) + \beta_l \cdot \left(1 - \text{corr}(\sigma_{\phi_l(\hat{I})}, \sigma_{\phi_l(I)}) \right) \right), \tag{18}$$

where $\mu_{\phi_l}, \sigma_{\phi_l}$ denote the local mean and standard deviation maps of layer-l deep features, $\text{corr}(\cdot, \cdot)$ denotes Pearson correlation, and α_l, β_l are learned weights.

E VOXEL-BASED BACKGROUND FUSION

Our background completion pipeline employs a voxel-grid-based fusion strategy to aggregate multi-frame background Gaussians into a unified static representation while eliminating spatial redundancy and temporal inconsistencies.

Spatial discretization. We partition the 3D space into uniform voxel grids with edge length $\delta_v = 0.05$ meters (5 cm). Each background Gaussian $\mathbf{g}_t^{(i)}$ from frame t is assigned to voxel bin $\mathbf{v}_{xyz} = \lfloor \mathbf{p}_t^{(i)} / \delta_v \rfloor$, where $\mathbf{p}_t^{(i)} \in \mathbb{R}^3$ denotes the 3D position of the Gaussian center.

Multi-view aggregation. For each non-empty voxel $\mathbf{v}_{xyz}$, we collect all assigned Gaussians $\{\mathbf{g}_t^{(i)} \mid \mathbf{v}_{xyz} = \lfloor \mathbf{p}_t^{(i)} / \delta_v \rfloor\}$ and aggregate their attributes as follows:

$$\bar{\mathbf{c}} = \frac{1}{N_v} \sum_{i \in \mathcal{I}_v} \mathbf{c}_t^{(i)}, \quad \bar{\mathbf{o}} = \frac{1}{N_v} \sum_{i \in \mathcal{I}_v} \mathbf{o}_t^{(i)}, \quad \bar{\mathbf{s}} = \max_{i \in \mathcal{I}_v} \mathbf{s}_t^{(i)} \tag{19}$$

where $\mathcal{I}_v$ denotes the index set of Gaussians within voxel $\mathbf{v}_{xyz}$, $N_v = |\mathcal{I}_v|$ is the cardinality of this set, and color $\bar{\mathbf{c}}$ and opacity $\bar{\mathbf{o}}$ are averaged while scale $\bar{\mathbf{s}}$ takes the maximum value to preserve fine-grained geometric details. The centroid position is computed as the mean: $\bar{\mathbf{p}} = \frac{1}{N_v} \sum_{i \in \mathcal{I}_v} \mathbf{p}_t^{(i)}$.

Outlier filtering. We apply three criteria to filter unreliable voxels:

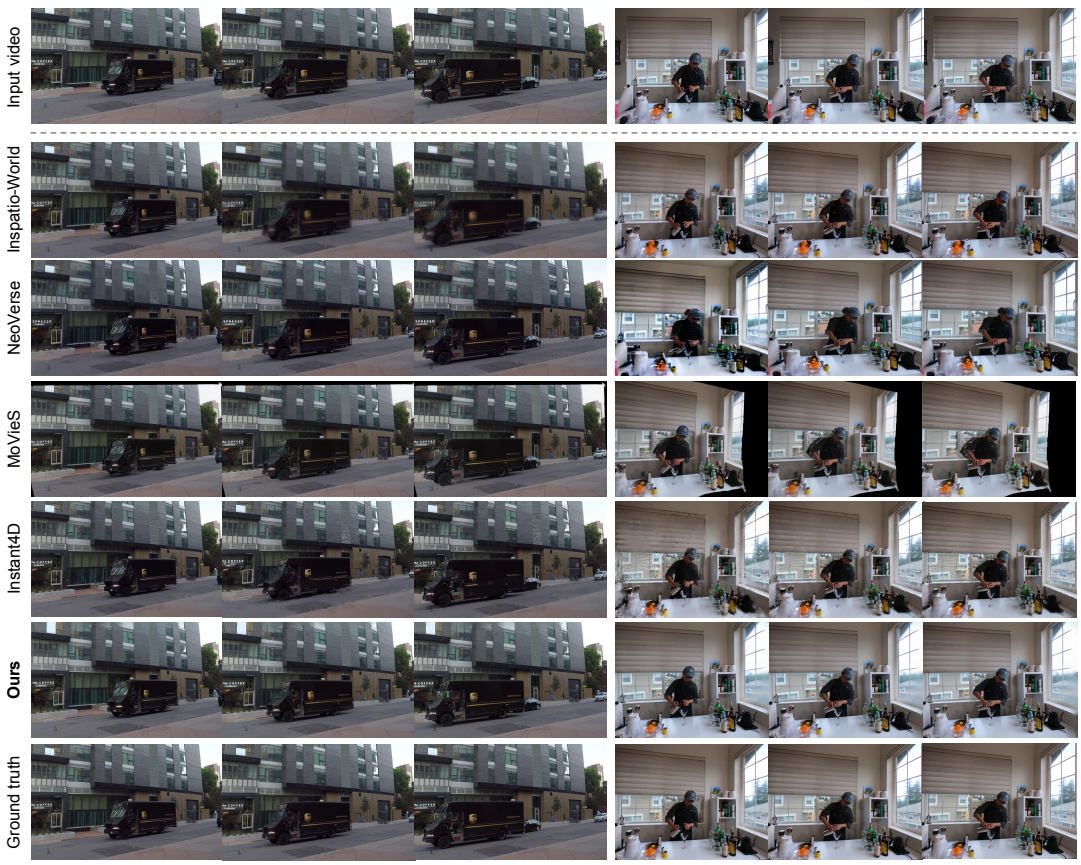


Figure 5: Qualitative Comparison on Neural-3D-Video and NVIDIA-Dynamic-Scene datasets.

1. **Minimum view count**: Retain only voxels observed in at least $n_{\text{min}} = 2$ frames to ensure multi-view consistency.
2. **Opacity threshold**: Discard voxels with average opacity $\bar{\mathbf{o}} < \tau_{\text{opacity}} = 0.1$, as low-opacity voxels typically correspond to noise or transient artifacts.
3. **Scale bound**: Exclude voxels with maximum scale $\bar{\mathbf{s}} > \tau_{\text{scale}} = 1.0$ meters to suppress overly large splats, which cause rendering artifacts that may result from depth estimation errors or foreground contamination.

The final fused background $\mathcal{G}_s^{\text{complete}}$ consists of one representative Gaussian per valid voxel, with aggregated attributes $\{\bar{\mathbf{p}}, \bar{\mathbf{c}}, \bar{\mathbf{o}}, \bar{\mathbf{s}}, \bar{\mathbf{r}}\}$, where rotation $\bar{\mathbf{r}}$ is inherited from the Gaussian with maximum opacity within the voxel.

F Spatiotemporal Gaussian Compression

Intra-frame Importance-based pruning. We compute the Gaussian opacity-weighted scale importance score as $\eta_i = \alpha_i \cdot \frac{1}{3}\sum_{j=1}^{3} e^{s_{i,j}}$, and we retain the top-$r$ fraction by computing the $(1-r)$-th quantile threshold τ_r among all foreground Gaussians in frame t. A compression-ratio ablation on the Neural-3D-Video dataset is shown in Tab. 8. We select $r = 0.6$ since the visual quality barely degrades while the storage size is considerably reduced.

Inter-frame rigid alignment via canonical-frame correspondence. Frame 0 uses K-means clustering to establish $K = 64$ canonical clusters $\{C_k\}$ with centers $\{\boldsymbol{\mu}_k\}$, the cluster index $k(i)$ denotes the precomputed cluster membership of the canonical Gaussian $\mathbf{g}_i^0$ obtained from frame-0 K-means clustering. We define the temporal association set $\mathcal{A}_t = \{(i,j) \mid \mathbf{g}_i^0 \leftrightarrow \mathbf{g}_j^t\}$ and the matched pairs

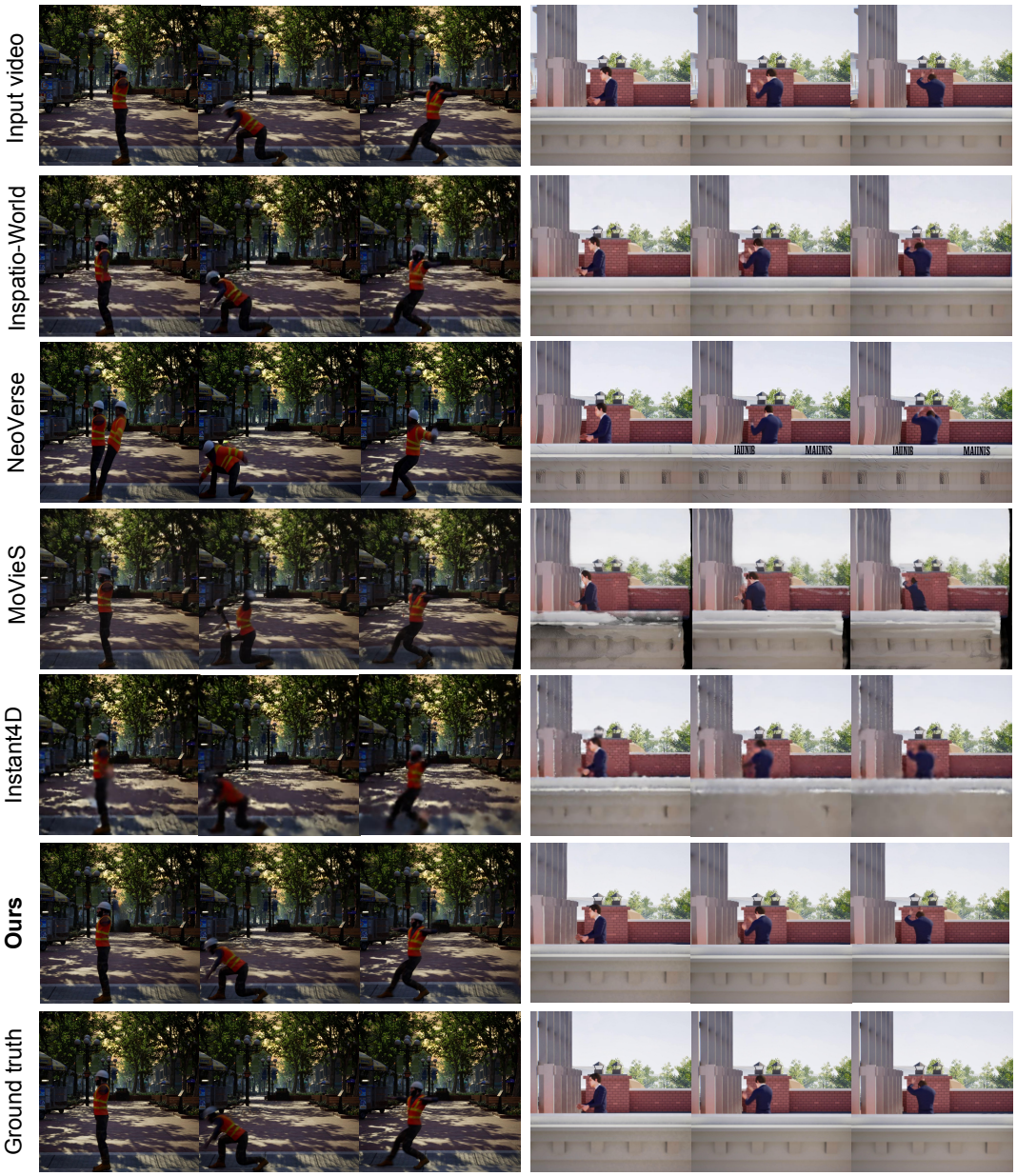


Figure 6: Qualitative Comparison on StereoDataset.

Table 8: Ablation study of Gaussian retention-ratio r on the Neural-3D-Video dataset.

Retention ratio r	**LPIPS**↓	**T-LPIPS**↓	**DISTS**↓	**Storage**
r=1.0	0.139	0.054	0.046	100%
r=0.6	0.143	0.068	0.051	60%
r=0.4	0.172	0.085	0.096	40%
r=0.2	0.248	0.162	0.173	20%

within cluster k as $\mathcal{A}_{t,k} = \{(i,j) \in \mathcal{A}_t \mid k(i) = k\}$, where w_{ij} is the correspondence confidence weight. For each cluster k, we estimate the rigid transformation $(\mathbf{R}_k^t, \mathbf{t}_k^t)$ by solving the weighted



Figure 7: Qualitative Comparison on Vista4D evaluation dataset.

orthogonal Procrustes problem over matched pairs:

$$(\mathbf{R}_k^t, \mathbf{t}_k^t) = \arg\min_{\mathbf{R},\mathbf{t}} \sum_{(i,j)\in\mathcal{A}_{t,k}} w_{ij} \left\| \mathbf{p}_j^t - \mathbf{R}\,\mathbf{p}_i^0 - \mathbf{t} \right\|_2^2, \tag{20}$$

which we solve in closed form via the Kabsch algorithm: (1) center the two matched point subsets of cluster k by subtracting their cluster-wise means $\bar{\mathbf{p}}_k^t$ and $\bar{\mathbf{p}}^{0,k}$, (2) compute the cross-covariance matrix $\mathbf{H} = \mathbf{P}_t\mathbf{P}_0^T$ where $\mathbf{P}_t, \mathbf{P}_0$ are the $3\times N_k$ centered coordinate matrices, (3) apply SVD to obtain $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, (4) compute rotation as $\mathbf{R}_k^t = \mathbf{V}\mathbf{U}^T$ with reflection correction, and (5) compute translation as $\mathbf{t}_k^t = \bar{\mathbf{p}}_k^t - \mathbf{R}_k^t\bar{\mathbf{p}}^{0,k}$. This point-wise correspondence ensures a well-conditioned rigid alignment that respects the canonical cluster structure.

Cluster activation. Not all canonical clusters are active in every frame. We define $a_t^k \in \{0, 1\}$ as the activation indicator for canonical cluster k in frame t, where $a_t^k = 1$ if at least one matched Gaussian pair exists in $\mathcal{A}_{t,k}$, and $a_t^k = 0$ otherwise. During decompression, only active clusters ($a_t^k = 1$) contribute Gaussians. New Gaussians that appear in frame t without a canonical counterpart in frame 0 are assigned to their nearest canonical cluster $k' = \arg\min_k \|\mathbf{p}_j^t - \boldsymbol{\mu}_k\|_2$ and treated as additional residuals. The set of canonical clusters $\mathcal{C}$ remains fixed starting from frame 0 and is never augmented with newly-appearing Gaussians. Conversely, canonical Gaussians that become occluded or leave the scene in frame t have no valid temporal matches and are omitted during decompression.

K-sensitivity. We report the effect of the number of canonical clusters K on scene size, LPIPS, T-LPIPS, and DISTS on the Neural-3D-Video dataset in Tab. 9. We select $K = 64$ as it achieves a good trade-off between reconstruction quality and storage overhead.



Figure 8: Qualitative ablation of spatiotemporal Gaussian compression.

Table 9: Ablation study of canonical cluster number K on reconstruction quality and storage.

Canonical clusters	Size of 4D Scene	LPIPS↓	T-LPIPS↓	DISTS↓
K=16	134 MB	0.155	0.072	0.054
K=32	128 MB	0.150	0.068	0.050
K=64	152 MB	0.143	0.068	0.051
K=128	176 MB	0.146	0.065	0.049
K=256	230 MB	0.141	0.062	0.045

Mixed-precision Attribute Quantization. We adopt mixed-precision quantization for compact Gaussian attribute storage. Specifically, temporal position residuals $\boldsymbol{\delta}_i^t$ are quantized with 10-bit precision to preserve subtle geometric motion details, while appearance attributes including color, scale and opacity employ 8-bit quantization. Since cluster-level rigid transformations only model geometric motion, they cannot capture appearance variations across frames, making direct per-frame quantization feasible for these visual properties. During reconstruction, cluster-level rotation updates Gaussian orientations, whereas scale, color and opacity are quantized frame-by-frame without canonical Gaussian prediction or residual compensation.

G LIMITATION

Our method is constrained by the performance of the adopted dynamic mask segmentation model Xie et al. (2026). As shown in Fig. 9, if this model fails to fully detect and segment all moving subjects in the video, it will incorrectly model undetected dynamic objects as static-background regions in the reconstructed 4D scene. For instance, the cigarette held by the man in the first case

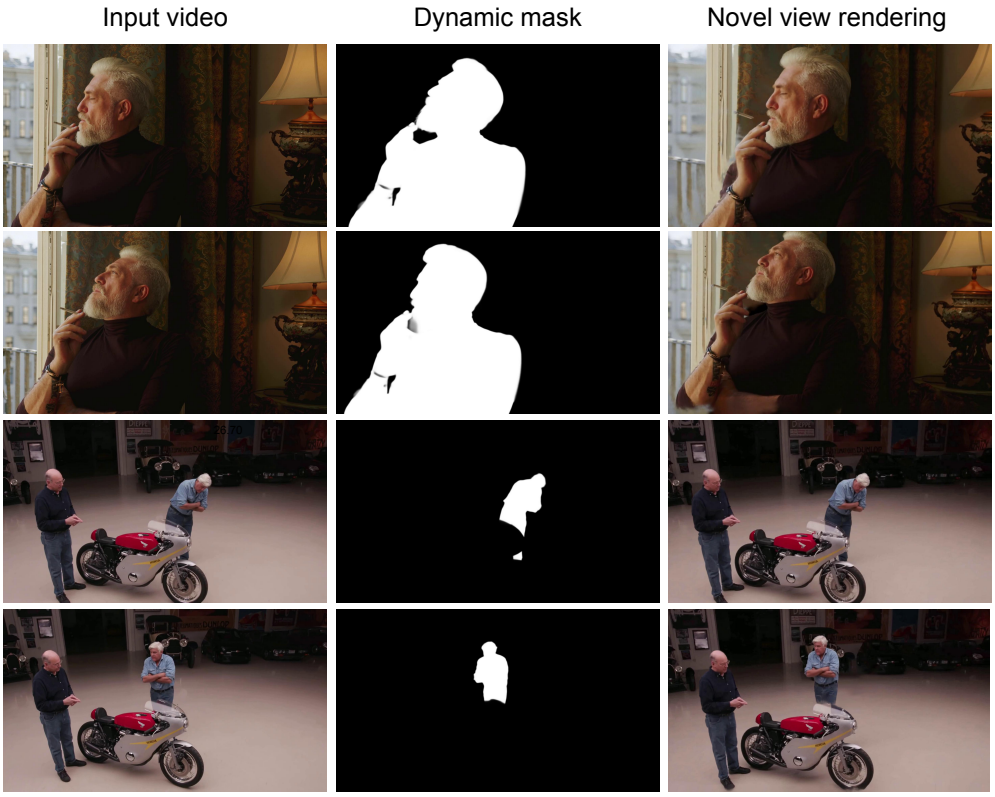


Figure 9: Failure cases. Reconstruction artifacts arise from imperfect dynamic mask segmentation and inaccurate depth prediction in challenging regions.

and the man at the bottom left corner in the second case are not recognized as dynamic objects. Our method is also limited by the capability of the underlying 3D reconstruction model Mescheder et al. (2026), which suffers from long-tail depth prediction challenges. In the second case, inaccurate depth estimation for the motorcycle windshield leads to failed reconstruction of the legs of the man in the blue shirt. Furthermore, for long video sequences spanning hundreds of frames, the reconstructed 4D scene still occupies considerable storage space even with our spatiotemporal Gaussian compression scheme, which motivates the further development of more efficient compression algorithms tailored for 4D Gaussian sequences.