Restage4D: Reanimating Deformable 3D Reconstruction from a Single Video

Jixuan He[1] , Chieh Hubert Lin[2], Lu Qi[3],*Ming-Hsuan Yang[2]
[1]Cornell Tech, [2] University of California, Merced, [3] Wuhan University
jh2926@cornell.edu

Abstract

Creating deformable 3D content has gained increasing attention with the rise of text-to-image and image-to-video generative models. While these models provide rich semantic priors for appearance, they struggle to capture the physical realism and motion dynamics needed for authentic 4D scene synthesis. In contrast, real-world videos can provide physically grounded geometry and articulation cues that are difficult to hallucinate. One question is raised: *Can we generate physically consistent 4D content by leveraging the motion priors of the real-world video*? In this work, we explore the task of reanimating deformable 3D scenes from a single video, using the original sequence as a supervisory signal to correct artifacts from synthetic motion. We introduce **Restage4D**, a geometry-preserving pipeline for video-conditioned 4D restaging. Our approach uses a video-rewinding training strategy to temporally bridge a real base video and a synthetic driving video via a shared motion representation. We further incorporate an occlusion-aware rigidity loss and a disocclusion backtracing mechanism to improve structural and geometry consistency under challenging motion. We validate Restage4D on DAVIS and PointOdyssey, demonstrating improved geometry consistency, motion quality, and 3D tracking performance. Our method not only preserves deformable structure under novel motion, but also automatically corrects errors introduced by generative models, revealing the potential of video prior in 4D restaging task. Source code and trained models will be released.

1 Introduction

Recent advances in foundational generative models in images [1, 2], videos [3], and 3D scenes [4, 5] drive growing interest in the creation of 4D content, which brings time-varying dynamics to 3D. In practical applications, controllability remains a valuable and continuing challenge in all generative frameworks. Inspired by video generative models, early controllable 4D generative frameworks condition on text prompts [6] or an image [7]. This intrigues us to explore other conditioning sources that are beneficial and intuitive for 4D content creation. In particular, we are interested in video conditioning, which is easily accessible on the Internet, exhibits physical properties of objects, and maintains visible regions that are broader than a single image. Meanwhile, existing works in 4D synthesis [8] mostly source the dynamics from video foundation models. Despite these models being able to synthesize visually appealing results with seemingly correct dynamics, they struggle to ensure physical plausibility, such as inconsistent appearance (*e.g.*, geometry modified after occlusion or out-of-view) and infeasible deformation (*e.g.*, limbs swapping during intersection). In Section 4.3, we show that extracting explicit 3D deformation from infeasible physics leads to floaters and broken geometry. These observations motivate us to develop a new paradigm for creating 4D content,
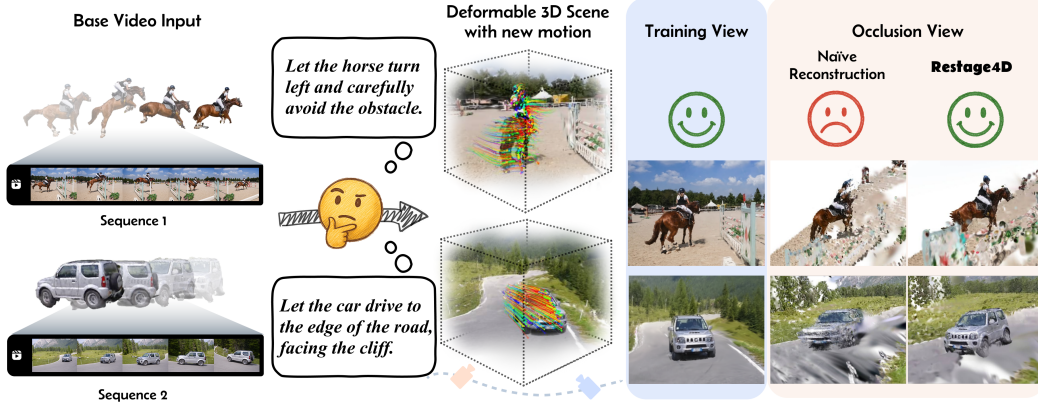
---

*Corresponding author: Lu Qi

Figure 1: We show the input and output for the Restage4D in the left. Given a base video and a text prompt, we reanimate the scene according to the specified motion, while keeping the geometry consistency for occlusion part and motion coherent for the restaged 4D scene as shown in the right.

leveraging two promising information sources: the geometry and physical constraints distilled from a real-world video and the diverse pseudo-motion synthesized by video foundation models.

One intuitive way to build a 4D representation is using single-video deformable 3D reconstruction [9, 10], which achieves photorealistic results with temporally coherent and physically plausible deformations. Such methods commonly model dense 3D deformation using low-rank motion bases, then linearly combine via motion coefficients. These motion coefficients determine how the bases are weighted to produce the final deformation. Their similarity can be interpreted as an indicator of how tightly coupled the two spatial points are in the object's motion. We observe that such a design is conceptually similar to the skinning weights of articulated meshes [11]. In graphics workflow, the artists assign deformations to only a few key points, and per-vertex skinning weights map those deformations to the full mesh and drive the articulated meshes to perform desired motions. In this view, the input base video provides informed skinning weights in the real world, effectively turning the reconstructed object into a *puppet*, whose strings can be manipulated by altering the low-rank motion components. This intrigues us to explore controlling the puppet with dynamics generated by a video diffusion model.

We propose **Restage4D** towards a new *4D restaging* task to create video-conditioned 4D content. Given a monocular video, Restage4D aims to synthesize a new 4D motion sequence that maintains the original appearance while generating novel and physically plausible deformations. The restaged 4D sequence is driven by a synthetic video generated by an image-conditioned video diffusion model based on the first frame of the base video. More specifically, the base video provides the shape and articulation model (*i.e.*, the puppet to control), while the driving video supplies new dynamics (*i.e.*, manipulate the puppet) and the appearance of the additional dis-occlusion revealed by novel motions.

The first challenge lies in sharing a consistent shape and motion representation across both videos. We observe that the deformable 3D reconstruction frameworks are invariant to temporal direction, reconstructing from a video playing in a reversed temporal order (*i.e.*, *rewind*) would result in the same reconstruction. In combination with the assumption that our base video and driving video share the first frame, we propose a video-rewinding joint-training scheme. We rewind the base video into the reversed temporal order, temporally concatenate with the driving video, and then jointly reconstruct both sequences as a single video. This enables smooth motion transition across videos when initializing deformable 3D reconstruction frameworks with smooth tracking, and allows intuitive sharing of motion coefficients across two videos. However, the simple joint optimization does not sufficiently leverage all the available information. As the synthetic video presents novel motions, certain geometries visible in the input video become partially or even completely invisible throughout the synthetic video. Lacking visibility makes these geometries unable to receive gradients to preserve the smoothness and continuity of the appearance. To leverage the geometry recovered in the input video, we introduce an occlusion-aware rigidity regularization to preserve the local rigidity of the less visible geometry. In addition, we use a disocclusion backtracing mechanism to recover missing canonical geometry by tracing observed points from the driving video back to the base video.

We validate our method on two datasets with monocular video settings and complicated motions, DAVIS [12] and PointOdyssey [13], as well as several self-collected Internet videos to test the generalization. Our experimental results show that our Restage4D can correct both infeasible motion and inconsistent geometries created by video diffusion models under complex deformations. Under rigorous ablation and physically-based metrics evaluation, we show that our method achieves consistent improvements across reconstruction quality, appearance consistency, and physical consistency.

The main contributions of this work are:

- We introduce a new task, *4D restaging*, which aims to reanimate deformable 4D scenes from video-driven motion.
- We propose a geometry-preserving pipeline that leverages a video-rewinding joint training scheme, combined with occlusion-aware ARAP and disocclusion backtracing.
- We demonstrate how real-world supervision can correct artifacts from generative videos, bridging controllability and geometric fidelity in 4D reconstruction.

2 Related Work

Deformable 3D Neural Rendering. Neural rendering techniques based on NeRF [14, 15, 16, 17, 18, 19, 20], and Gaussian Splatting [21, 22] have demonstrated impressive performance in reconstructing high-fidelity static 3D scenes. Building upon this success, recent works have explored how to extend these representations to handle deformable 3D scenes with complex motion. A prominent line of work, including D-NeRF [23], Nerfies [24], and HyperNeRF [25], learns a time-conditioned deformation field via MLPs, enabling per-frame warping from a canonical space. However, because of the implicit nature of NeRF, maintaining coherent motion and preserving geometry over time remains challenging. To address this, recent Gaussian Splatting-based approaches [26, 27, 28, 29] adopt similar deformation-field formulations to model temporal variation, using multi-view videos as input. In contrast, methods like Shape-of-Motion [9], Mosca [10] and DGS-LRM [30] explicitly represent motion via rigid-body transformations and shared motion bases by utilizing depth [31], tracking [32] and camera-pos [33] priors. Although occlusion remains a significant challenge for such approaches, the explicit nature of the representation enables finer-grained control over motion and facilitates geometric reasoning. However, existing monocular approaches suffer from preserving geometric information under occlusion introduced by the complex motion. Therefore, our work focuses on maintaining geometric realism and motion coherence during monocular video-conditioned 4D content creation, which requires strong regularization and guidance from real observations.

Motion Retargetting. Motion retargeting is a fundamental problem in human-centric motion analysis and plays a central role in motion capture (MoCap) [34], scene animation [35], and motion transfer [36]. The goal of retargeting is to animate the static object using a reference input. Early works formulate retargeting as an optimization problem with kinematic constraints on articulationd body models [37], often solved via inverse kinematics (IK) [38, 39]. Recent approaches leverage deep learning by incorporating skeleton-aware modules [40, 41], enabling high-quality real-time motion retargetting in a variety of subjects. Beyond human motion, retargeting deformable 3D scenes presents new challenges. Under the Gaussian Splatting framework, methods such as SC-GS [42] and D-MiSo [43] introduce keypoint-based control to transfer motion from one 4D scene to another. Although effective in constrained scenarios, these methods rely on carefully designed control points and often require meticulous tuning to produce natural motion, especially when handling complex or long-range deformations. In contrast, our approach adopts an example-driven paradigm. Instead of designing explicit keypoint trajectories, we reanimate the scene by conditioning on a generated video sequence. This allows for retargeting complex and subtle motions directly from video while preserving geometric and temporal coherence.

4D generation. Recent advances in large-scale generative models such as Stable Diffusion [1], Flux [44], and Sora [45] have catalyzed significant progress in image, video and 3D generation. Naturally, this momentum has extended to 4D content generation, where the goal is to synthesize dynamic 3D scenes with time-varying geometry and appearance. One line of work directly uses generative models to create multiview observations or spatiotemporal sequences for the object condition on image or text , such as Zero-1-to-3 [46] and 4D-Diffusion [47]. Another line of research integrates synthetic supervision via score distillation sampling (SDS): methods like DreamGaussian [48] and
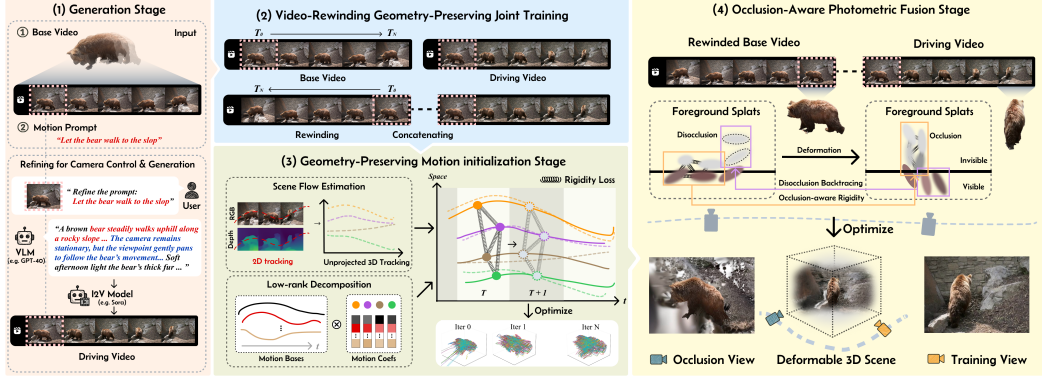
Figure 2: Pipeline of Restage4D. To perform a 4D restaging task, a base video and a text prompt are utilized to generate a driving video in the generation stage. Then, in video-rewinding stage, the base video is back-played and concatenate with the driving video to perform joint optimization. A geometry-preserving rigidity loss is applied in the motion initialization stage. Then, in photometric fusion stage, a occlusion-aware rigidity loss and a backtracing mechanism is incorporated.

4D-FY[8] use SDS to guide 3D or 4D reconstruction from text or reference images. Style-conditioned generation and editing also benefit from generative models, as demonstrated by works such as Instruct-NeRF2NeRF [49] and DreamBooth3D [50], where image-generated priors are used to control object appearance, geometry, or identity across views. However, few works explore using video as input for the objects' articulation and using text as motion source to generate realistic 4D scene. By using semantic prior in the video, our work tackles the problem of inconsistent appearance and motion in 4D restaging, helping to preserve physical plausibility and temporal coherence.

3 Method

With an input base video providing the context and a text prompt providing the description of motion, our goal is to create a 4D scene in which the object follows the specified motion. Since we use a monocular video as reference for the objects' motion articulation and appearance and a text prompt as the source of motion, it is very likely that the novel motion introduces occlusion and artifacts, leading to wrong geometry. To have a better geometry in the restaged 4D scene, the key is to utilize the prior knowledge in the base video. Therefore, we use a low-rank decomposed representation combined with a geometry-preserving training pipeline to obtain authentic 4D content.

3.1 Preliminaries: Low-rank Decomposed 4D Gaussian Splatting Representation

We adopt a motion representation similar to *Shape-of-Motion* [9] to model dynamic 4D Gaussian Splatting. Specifically, the scene is represented by two components: a static background point set $\mathbf{B} = \{b_0, b_1, ..., b_{N_b-1}\}$ and a dynamic foreground point set $\mathbf{X} = \{x_0, x_1, ..., x_{N_f-1}\}$, where each point is associated with a 3D Gaussian (mean, scale, rotation, opacity and color).

To capture temporal motion, each foreground point x_i is associated with a canonical position $\mu_i \in \mathbb{R}^3$, a shared time-independent motion coefficient $\beta_i \in \mathbb{R}^K$, and a set of time-varying motion bases $\mathbf{M} = \{M_k(t)\}_{k=1}^K$, where each $M_k(t) \in SE(3)$ is a smooth rigid body transformation over time.

The deformation of each foreground point at time t is obtained by composing the motion bases weighted by its coefficients:

$$T_i(t) = \gamma(\sum_{k=1}^K \beta_{ik} \cdot M_k(t)), \qquad x_i(t) = T_i(t) \cdot \mu_i \tag{1}$$

where the transformation is applied in SE(3), and $x_i(t)$ denotes the 3D position of point x_i at time t. An orthogonalization function γ is applied to ensure the legality of T_i. During training, the parameters of the Gaussians and the motion bases $M_k(t)$ are optimized to match multi-view input observations. The training procedure in Shape-of-Motion consists of two stages. In the motion initialization phase, a pre-trained tracking model is applied to detect dynamic foreground points across frames. These

4

tracked 2D points are then unprojected into 3D to estimate a coarse scene flow, providing supervision for the initial motion. Based on this, the motion coefficients β_{ik} and the time-varying motion bases $M_k(t)$ are initialized and optimized to best fit the scene flow. In the photometric fusion phase, the model is fine-tuned using a standard Gaussian splatting-style inverse rendering pipeline, which jointly refines motion, appearance, and structure using photometric losses and 2D tracking consistency.

3.2 Driving Video Generation & Rewinding

To achieve 4D scene creation with novel motion, we propose an example-driven pipeline that incorporates the deformable object's geometry and motion articulation from the base video and the motion dynamics from the text prompt. As shown in the first stage of Figure 2, the input of the pipeline is a base video and a motion prompt describing the desired motion. To create the dynamics from the text prompt to drive the object, instead of using some key points to simulate the proposed motion, we leverage an Image-to-Video (I2V) diffusion model such as Sora [45] to synthesize a driving video for complex and plausible motion as the source of dynamics. To be specific, the first frame of the base video is extracted as the input for the I2V models. Moreover, to obtain a stable source of dynamics, a smooth and steady camera motion in the driving video is desired. Therefore, another Vision Language Model (VLM) such as ChatGPT-4o is utilized to refine the text prompt, especially for explicit control of camera motion. After that, the refined text prompt, along with the extracted frame, is sent to the I2V models to generate a relatively high quality driving video.

However, driving video may introduce additional occlusion, disocclusion, and artifacts. To address these issues, we incorporate geometry and articulation priors from the base video as additional supervision. This helps stabilize the reconstruction, preserve geometry, and improve motion consistency in the final 4D scene. A key insight is that the original and generated sequences typically share the same first frame, enabling us to connect them through a shared canonical scene. As such, we rewind the base video (i.e., play it backward), concatenate it with the driving sequence, and perform joint optimization over the entire temporal span as shown in the second of Figure 2 . After optimization, we can truncate the reconstructed scene to get the final result.

This joint training strategy brings in strong geometry and motion constraints from the original clip and allows them to propagate into the edited segment via shared motion coefficients. We formally state this motivation in the following lemma and defer the detailed proof to Appendix A.

Lemma 1. *Let $d(t)$ denote the distance between two 3D points $x_A(t), x_B(t)$ defined via shared motion coefficients and time-dependent bases, as in Shape of Motion. Suppose that the motion bases are trained jointly on an edited sequence $[0, t_1]$ (without supervision) and an original sequence $[t_1, t_2]$ (with supervision), with the temporal smoothness regularization applied.*

Then, the variance of $d(t)$ over $[0, t_1]$ is lower than if the model were trained on $[0, t_1]$ alone:

$$Var_{t \in [0, t_1]}(d(t)) < \sigma_0$$

where σ_0 is the variance from training without supervision.

3.3 Occlusion-Aware Rigidity Regularization

During the Motion Initialization Stage, we achieve geometry preservation by effectively propagating geometry supervision throughout the sequence to achieve better initialization. Following the shape of motion [9], we use a low-rank decomposition of the motion bases $\{M_k(t)\}$ and the motion coefficients β_i to fit the estimated flow of the scene. This scene flow is often noisy, especially in the driving video due to occlusions or motion artifacts. Inspired by physical-based regularization such as As-Rigid-As-Possible (ARAP) [51], we use a global Rigidity Loss to regularize the deformation. Specifically, we first construct a k-NN graph $\Omega = \{(x_i, x_j)\}$ among foreground Gaussians, and define the loss as:

$$\mathcal{L}_{Rigidity\_init} = \sum_{t=1}^{T} \sum_{\Omega_{i,j}} s_{ij} \|d(x_i(t), x_j(t)) - d(\mu_i, \mu_j)\|_1 \tag{2}$$

Here, we use the frame with the most visible points as a canonical frame. μ_i is the canonical position of the point x_i. s_{ij} measures motion similarity based on the cosine similarity of motion coefficients

β_i:

$$s_{ij} = \frac{\boldsymbol{\beta}^\top \boldsymbol{\beta}_j}{\|\boldsymbol{\beta}_i\|\|\boldsymbol{\beta}_j\|} \tag{3}$$

Once the motion bases and coefficients are reasonably initialized, the object's appearance will be refined during the Photometric Fusion Stage. The photometric and tracking losses are used to further optimize the Gaussian parameters. To effectively propagate geometry supervision from the base sequence to occluded and inconsistent regions in the driving sequence, while avoid oversmoothing visible regions, we adapt the Rigidity Loss to this occlusion-aware scenario. The intuition is that the occluded regions in the driving video should correspond to visible regions in the base sequence under a rigid transformation, while the visible part should rely more on the photometric information.

We first estimate the invisibility score $\zeta_i(t) \in [0,1]$ for each point $x_i(t)$, based on its depth difference from the rendered depth buffer $\hat{D}(t)$. The depth buffer is computed similar to color rendering as:

$$\hat{D}(t) = \sum_{i \in N} d_i(t)\alpha_i T_i, \text{ where } T_i = \prod_{j=1}^{i-1}(1 - \alpha_i) \tag{4}$$

where $d_i(t)$ is the depth of each splat $x_i(t)$ from the camera. We then define a smooth step function over the depth difference $\delta(t) = d_i(t) - \hat{D}(t)$ to compute invisibility

$$\zeta_i(t) = \begin{cases} 0 & \text{if } \delta < \tau_0 \\ 3(\frac{\delta - \tau_0}{\tau_1 - \tau_0})^2 - 2(\frac{\delta - \tau_0}{\tau_1 - \tau_0})^3 & \text{if } \tau_0 \leq \delta < \tau_1 \\ 1 & \text{if } \delta \geq \tau_1 \end{cases} \tag{5}$$

where τ_0 and τ_1 is a pre-defined boundary between visible and invisible part. This formulation allows a soft occlusion mask to guide the selective regularization of rigidity.

Finally, the occlusion-aware Rigidity loss is applied during refinement as:

$$\mathcal{L}_{Regidity\_refine} = \sum_{t=1}^{T} \sum_{\Omega_{i,j}} \zeta_i(t)\zeta_j(t)s_{ij}\|d(x_i(t), x_j(t)) - d(\mu_i, \mu_j)\|_2 \tag{6}$$

This encourages local rigidity only in regions that are not directly supervised, improving geometry propagation in occluded areas without compromising visible-region quality. We observe that overly strong rigidity constraints may encourage motion coefficient sparsity and suppress high-frequency motion. To address this, we increase the number of motion bases to 50–100 in our implementation, which empirically balances rigidity and expressiveness.

3.4 Disocclusion Backtracing

While the Rigidity Loss during initialization helps align the edited region motion with the original sequence, we observe that newly disoccluded parts that appear only in the driving video are hard to recover, since they were never visible in the base video and thus absent from the canonical frame.

To address this, we propose a disocclusion backtracing mechanism to recover such missing geometry by supplementing the canonical representation after initialization. We examine the visibility $\nu_i(t)$ of each tracked point $p_i(t)$, using the output of standard tracking models such as TAP-AIR [32] or Bootstap [52]. A point is considered disoccluded at time $t \in T_{\text{edit}}$ if it becomes visible in the edited video but was occluded $\mathcal{D}$ in the canonical frame:

$$v_i(t) \wedge \neg v_i(t_{cano}) \implies p_i(t) \in \mathcal{D}(t) \tag{7}$$

For each such disoccluded point, we insert a new Gaussian splat $x_i'(t)$ at its 3D location and assign its color directly from the input. We then interpolate its motion coefficients β_i' from nearby Gaussians and backtrace the point to the canonical frame by using the inverting of Equation 1:

$$\mu_i' = \gamma(\sum \beta_{ik}' \cdot M_k(t))^{-1}x_i'(t), \ \forall p_i(t) \in \mathcal{D}(t) \tag{8}$$

This complements the occlusion-aware rigidity regularization by explicitly recovering disocclusion, allowing both occluded and disoccluded regions to be consistently aligned in the canonical space.
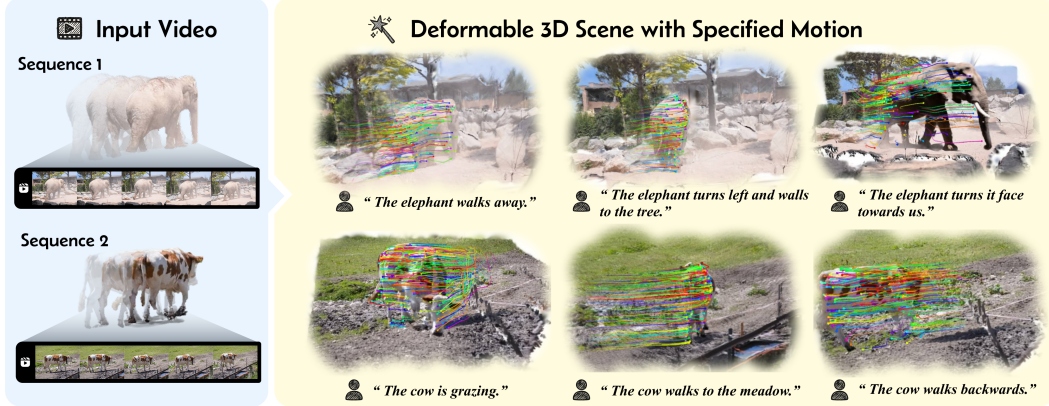
Figure 3: Samples of 4D motion creation. For the same input base video, we can create different motion variation based on the input text prompt, while preserving visual context in the base video.

4 Experiments

We evaluate our pipeline on both curated and in-the-wild video datasets, including DAVIS [12] and PointOdyssey [13]. For 4D restaging evaluation, we use 20 video sequences from the DAVIS dataset. Following the pipeline described in the previous section, we apply text-driven video generation to reanimate the input videos and reconstruct them using our geometry-preserving framework. To evaluate geometric consistency, we further test our method on the PointOdyssey dataset. We apply our full pipeline to reconstruct dynamic scenes and compare the resulting 3D trajectories with the ground-truth object tracking annotations, demonstrating our method's ability to preserve accurate geometry. The DAVIS sequences used in our experiments will be released to facilitate future research. All experiments are conducted on a single NVIDIA A100-SXM4 with 15GB GPU RAM. Each sequence typically takes 40 minutes for a sequence with 100 frames on 500 training epochs.

4.1 Reanimation and Geometry Preserving

We evaluate our pipeline in the DAVIS data set to demonstrate its ability to preserve geometry in the 4D reconstruction task. Given an input video and a text prompt describing a new motion, we use a pre-trained video generation model (e.g., Sora) to reanimate the object and then reconstruct the 4D scene using our pipeline. As shown in Figure 3, our method can generate a variation of motion based on a different text prompt. The restaged video presents novel motion while keeping the context of the input scene. Please refer to the supplementary for the input sequence and motion prompts.

A key challenge in 4D restaging is occlusion, as parts of the object may become invisible due to the new motion trajectory. Figure 4 shows the visualization for the restaged 4D scene in different views and time steps. Significant improvement in quality can be observed for occluded regions.

To assess geometry consistency, we introduce three metrics: (1) CLIP [53]-based object-centered view (OCV) consistency, (2) volume consistency, and (3) edge length stability. Existing 3D quality metrics such as PointSSIM [54] and GraphSSIM [55] are not suitable for our setting, as they are sensitive to spatial deformation and assume static correspondence. Since our pipeline explicitly models deformation, such frame-level metrics fail to capture global geometry stability. For OCV CLIP, we render each scene from a fixed relative viewpoint to the object (*e.g.*, behind or side) and compute the CLIP similarity to a clean reference frame. This measures whether the occluded geometry is preserved and visually plausible. Volume consistency C_v is computed as:

$$C_v = \left(-\log \left(\frac{1}{|T_{\text{driving}}|} \sum_{t \in T_{\text{driving}}} (V_t - \bar{V})^2 \right) \right)^{\gamma}, \quad \gamma = 1.5 \tag{9}$$

where V_t is the volume of the foreground voxelized at time t, and $\bar{V}$ is the mean volume over the driving clip. Higher values indicate lower variance and stronger volume conservation. Similarly, we measure the frame-to-frame consistency of local geometry by sampling 1000 Gaussians and tracking the standard deviation of their edge lengths to neighbors over time. Our proposed metrics better reflect the consistency of deformable 3D reconstructions under the 4D restaging task.
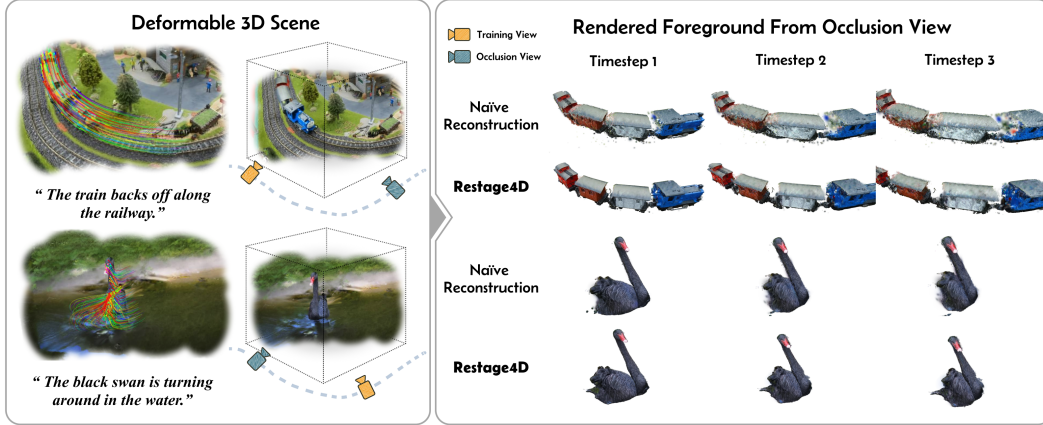
Figure 4: Samples of geometry preserving for occlusion view. For each restaged 4D scene, we visualize the foreground observing from the occlusion view at different time step. The geometry of the object can be better preserved using our method, compared to naive reconstruction.
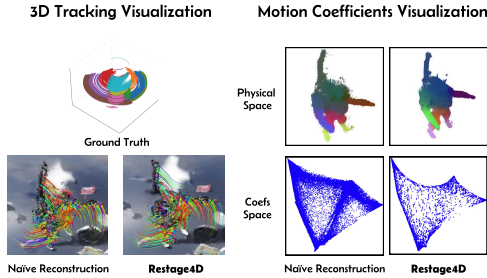

Figure 5: Visualization on PointOdessy dataset.

Table 4: Quantitative measure on PointOdessy Dataset. Restage4D can achieve higher tracking quality with geometry-preserving strategy.

Model	3D Tracking Loss
All	**0.05**
w/o Backtracing	0.06
w/o Rigidity	0.09

Table 1 summarizes the results. These results reflect our core motivation for designing the occlusion-aware ARAP. While the naive ARAP improves geometric consistency metrics (Vol./Edge Consistency) for the occluded part, it applied a strong rigidity constraint globally, which significantly degrades the reconstruction quality (PSNR drops by 5.24). This is especially problematic in regions with high-frequency motion, where excessive regularization results in visually unnatural and frozen movements, leading to incorrect motions. Through empirical observations, we found a trade-off: (1) Enforcing ARAP uniformly helps preserve occluded geometry, but overly constrains visible regions and suppresses natural motion. (2) Reducing regularization improves visible motion, but sacrifices occluded region stability.

By providing a base video, we enable the model to transfer accurate articulations and geometry. Combined with selective ARAP regularization, our approach optimizes the trade-off between geometric consistency and motion fidelity. Our method achieves the highest PSNR on the training views. By leveraging priors from the base video, it also achieves strong consistency scores in terms of CLIP-score and occlusion geometry, demonstrating its ability to maintain perceptual coherence.

In Table 2, we also compare our model with other 4D reconstruction baselines on generated video condition on videos from DAVIS [12]. It's hard to get ground truth for novel view in monocular video, so we conduct additional evaluation using the Neural3DV [56], which provides 20 calibrated camera views of dynamic scenes, from which we can derive ground truth for novel views. Please refer to the appendix for construction details. We integrate other image-to-video(I2V) models into our pipeline, and it works on multiple models. In Table 3, we evaluated CausVid [57] and LTX-Video [58] for video generation and compared them with Sora. We applied our restaging pipeline to two sequences: cat-jump and cows-graze, using each I2V model as the video generation component. These results demonstrate that our Restage4D framework performs effectively across diverse I2V backbones, with consistent reconstruction quality and visual plausibility.

Table 1: Ablation study on 20 DAVIS sequences. We report the mean $\pm$ std of PSNR, SSIM and LPIPS for foregrounds, as well as CLIP similarity for occlusion views (OCV), and geometry consistency metrics. Improvements (Δ) are computed per-sequence relative to the Baseline. We also evaluated the effect of using naive ARAP and occlusion-aware ARAP (o-ARAP).

Method	PSNR ↑	SSIM ↑	LPIPS ↓	OCV CLIP ↑	Vol. Consist ↑	Edge Consist ↑
Baseline	26.71 ± 1.97	0.9765 ± 0.004	0.0994 ± 0.049	81.68 ± 4.91	7.41 ± 3.80	12.66 ± 3.01
w/o Base Video						
+ Naive ARAP	23.67 ± 1.78	0.9581 ± 0.007	0.1733 ± 0.091	84.78 ± 4.39	9.14 ± 3.31	15.91 ± 3.84
Δ from Baseline	-3.04	-0.0175	+0.0739	+3.1	+1.73	+3.25
+ o-ARAP	26.80 ± 2.26	0.9812 ± 0.006	0.0754 ± 0.044	82.65 ± 4.29	7.92 ± 3.97	13.64 ± 3.38
Δ from Baseline	+0.19	+0.0056	-0.024	+0.97	+0.51	+0.98
w/ Base Video						
w/o ARAP	26.58 ± 2.21	0.9614 ± 0.004	0.1429 ± 0.063	82.29 ± 3.58	8.50 ± 3.81	13.24 ± 3.54
Δ from Baseline	−0.13	-0.0152	+0.0435	+0.61	+1.09	+0.58
+ Naive ARAP	21.47 ± 1.94	0.9094 ± 0.005	0.1917 ± 0.083	87.31 ± 3.05	11.43 ± 4.27	17.30 ± 3.95
Δ from Baseline	-5.24	-0.0671	+0.0923	+5.63	+4.02	+4.64
+ o-ARAP	27.12 ± 2.27	0.9832 ± 0.006	0.0632 ± 0.042	85.40 ± 4.11	10.32 ± 4.23	16.28 ± 3.10
Δ from Baseline	+0.41	+0.0067	-0.0362	+3.72	+2.91	+3.62

Table 2: Quantitative measure on PointOdessy Dataset. Restage4D can achieve higher tracking quality with geometry-preserving strategy.

Method	DAVIS [12]	Neural3DV [56]
Vidu4D [59]	19.77	14.74
SC-GS [42]	22.69	16.71
Ours	**27.38**	**17.78**

Table 3: Reconstruction PSNR using different video generation models. Restage4D can create content of similar quality using different models.

I2V Model	cat-jump	cows-graze
Sora [59]	26.74	26.25
CausVid [42]	23.68	26.45
LTX-video	25.08	28.05

4.2 3D Tracking

The advantages of geometry preservation are also demonstrated in 3D tracking. We evaluate our pipeline on DAVIS and PointOdyssey to show that our occlusion-aware rigidity constraint improves the tracking quality of occluded regions, As shown in the left part of Figure 5, our method can faithfully track points that become occluded during motion, as well as obtain smoother tracking trajectories, while the baseline without rigidity loss fails to maintain consistent trajectories and geometry for those regions. PointOdyssey provides ground-truth 3D point tracks, allowing us to quantitatively evaluate tracking accuracy. We compute the L1 error between the predicted and ground-truth 3D trajectories. As shown in Table 4, incorporating the occlusion-aware rigidity loss and back-tracing mechanism leads to lower tracking error, particularly in previously occluded areas, confirming its effectiveness in maintaining geometric consistency over time. We further visualize the learned motion coefficients in the right part of Figure 5. With loss of occlusion-sensitive rigidity, the coefficients become sparser and exhibit smoother spatial transitions. Occluded regions are encouraged to share similar motion coefficients with their neighbors, which helps to enforce local rigidity and preserve geometry through occlusion. This in turn improves the stability of 3D trajectories over time, leading to better tracking quality for both visible and invisible parts of the object.

4.3 Automatic Geometry Correction

Since our motion editing pipeline relies on videos generated by image-to-video models, it is common for these synthesized sequences to contain incorrect geometry or motion artifacts. This highlights an additional advantage of incorporating the original video as an auxiliary input during training: by using accurate observations as supervision, the model can automatically correct geometric and motion-related errors present in the generated video. As shown in Figure 6, our method successfully fixes several failure cases introduced by the video generation model. In the first example, the model mistakenly attaches a road sign to the bus, corrupting the appearance of the foreground. In the second case, a part of the background is erroneously fused to the camel's hump, and the motion of its legs is incorrectly changed. Through joint training with the original sequence, Restage4D is able to correct these inconsistencies, recovering plausible appearance and producing more realistic deformation,
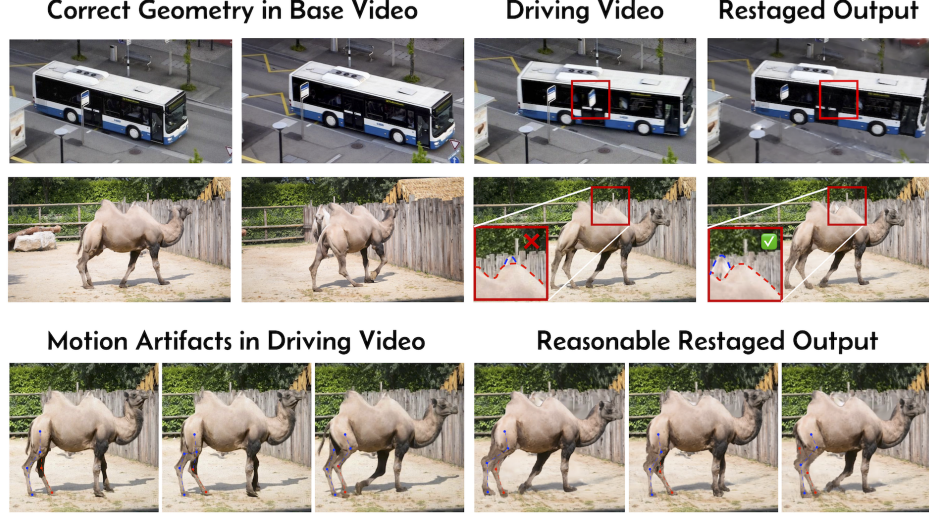
Figure 6: Samples of geometry and motion correction. Conditioning on the base video, Restage4D can use the learned articulation to correct some appearance and motion artifacts in the driving video.

even from flawed motion inputs. These results demonstrate that leveraging accurate prior observations from the original sequence not only improves geometric consistency, but also enables automatic correction of hallucinated artifacts and implausible motion, bridging the gap between generative and physically faithful reconstructions.

5 Conclusion and Limitation

In this work, we introduce **4D restaging**, a new video-conditioned 4D content creation task that aims to reanimate deformable 3D reconstructions from a single real-world video. By combining a pre-trained video generation model with our geometry-preserving video-rewinding training pipeline, Restage4D enables the transfer of motion dynamics from a synthetic video while preserving the physical plausibility and geometric fidelity of the original scene. Our approach leverages the input video as a source of shape and articulation priors, allowing us to generate high-quality 4D scenes that remain consistent even in occluded or disoccluded regions.

However, our method has limitations. If the video generation model produces sequences with severe geometric artifacts or unrealistic motions, the resulting supervision may not be sufficient to recover accurate 4D geometry. Additionally, while we leverage geometry supervision from real videos, we assume the input object remains temporally consistent across sequences, which may not hold for highly deformable or textureless objects. Our framework involves generative video models, which may pose risks when applied to human-centric content, potentially leading to the synthesis of misleading or harmful videos. One possible solution is incorporating prompt filtering mechanisms during the prompt refinement stage to detect and block inappropriate or sensitive prompts.

We hope that our work offers insights into how real-world video properties (*e.g.*, geometry priors, articulation, visibility) can be used to stabilize and guide 4D content creation, bridging generative modeling with reconstruction-driven pipelines.

6 Acknowledgement

References

[1] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695. 1, 3

[2] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014. 1

[3] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni *et al.*, "Make-a-video: Text-to-video generation without text-video data," *arXiv preprint arXiv:2209.14792*, 2022. 1

[4] S. Li, C. Li, W. Zhu, B. Yu, Y. Zhao, C. Wan, H. You, H. Shi, and Y. Lin, "Instant-3d: Instant neural radiance field training towards on-device ar/vr 3d reconstruction," in *Proceedings of the 50th Annual International Symposium on Computer Architecture*, 2023, pp. 1–13. 1

[5] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "Dreamfusion: Text-to-3d using 2d diffusion," *arXiv preprint arXiv:2209.14988*, 2022. 1

[6] U. Singer, S. Sheynin, A. Polyak, O. Ashual, I. Makarov, F. Kokkinos, N. Goyal, A. Vedaldi, D. Parikh, J. Johnson *et al.*, "Text-to-4d dynamic scene generation," *arXiv preprint arXiv:2301.11280*, 2023. 1

[7] Y. Zhao, Z. Yan, E. Xie, L. Hong, Z. Li, and G. H. Lee, "Animate124: Animating one image to 4d dynamic scene," *arXiv preprint arXiv:2311.14603*, 2023. 1

[8] S. Bahmani, I. Skorokhodov, V. Rong, G. Wetzstein, L. Guibas, P. Wonka, S. Tulyakov, J. J. Park, A. Tagliasacchi, and D. B. Lindell, "4d-fy: Text-to-4d generation using hybrid score distillation sampling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 7996–8006. 1, 4

[9] Q. Wang, V. Ye, H. Gao, J. Austin, Z. Li, and A. Kanazawa, "Shape of motion: 4d reconstruction from a single video," *arXiv preprint arXiv:2407.13764*, 2024. 2, 3, 4, 5

[10] J. Lei, Y. Weng, A. Harley, L. Guibas, and K. Daniilidis, "Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds," *arXiv preprint arXiv:2405.17421*, 2024. 2, 3

[11] D. Vlasic, I. Baran, W. Matusik, and J. Popović, "Articulated mesh animation from multi-view silhouettes," in *Acm Siggraph 2008 papers*, 2008, pp. 1–9. 2

[12] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 724–732. 3, 7, 8, 9

[13] Y. Zheng, A. W. Harley, B. Shen, G. Wetzstein, and L. J. Guibas, "Pointodyssey: A large-scale synthetic dataset for long-term point tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 855–19 865. 3, 7

[14] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021. 3

[15] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *European conference on computer vision*. Springer, 2022, pp. 333–350. 3

[16] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM transactions on graphics (TOG)*, vol. 41, no. 4, pp. 1–15, 2022. 3

[17] R. Martin-Brualla, N. Radwan, M. S. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, "Nerf in the wild: Neural radiance fields for unconstrained photo collections," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 7210–7219. 3

[18] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5501–5510. 3

[19] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 5855–5864. 3

[20] S. Fridovich-Keil, G. Meanti, F. R. Warburg, B. Recht, and A. Kanazawa, "K-planes: Explicit radiance fields in space, time, and appearance," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 479–12 488. 3

[21] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023. 3

[22] J. You, C. H. Lin, W. Lyu, Z. Zhang, and M.-H. Yang, "Instainpaint: Instant 3d-scene inpainting with masked large reconstruction model," in *Adv. Neural Inform. Process. Syst.*, 2025. 3

[23] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 10 318–10 327. 3

[24] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Nerfies: Deformable neural radiance fields," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 5865–5874. 3

[25] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz, "Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields," *arXiv preprint arXiv:2106.13228*, 2021. 3

[26] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, and X. Jin, "Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 20 331–20 341. 3

[27] Z. Lu, X. Guo, L. Hui, T. Chen, M. Yang, X. Tang, F. Zhu, and Y. Dai, "3d geometry-aware deformable gaussian splatting for dynamic view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8900–8910. 3

[28] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, "4d gaussian splatting for real-time dynamic scene rendering," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 20 310–20 320. 3

[29] H. Jung, N. Brasch, J. Song, E. Perez-Pellitero, Y. Zhou, Z. Li, N. Navab, and B. Busam, "Deformable 3d gaussian splatting for animatable human avatars," *arXiv preprint arXiv:2312.15059*, 2023. 3

[30] C. H. Lin, Z. Lv, S. Wu, Z. Xu, T. Nguyen-Phuoc, H.-Y. Tseng, J. Straub, N. Khan, L. Xiao, M.-H. Yang *et al.*, "Dgs-lrm: Real-time deformable 3d gaussian reconstruction from monocular videos," in *Adv. Neural Inform. Process. Syst.*, 2025. 3

[31] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 10 371–10 381. 3

[32] C. Doersch, Y. Yang, M. Vecerik, D. Gokay, A. Gupta, Y. Aytar, J. Carreira, and A. Zisserman, "Tapir: Tracking any point with per-frame initialization and temporal refinement," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 10 061–10 072. 3, 6

[33] Z. Teed and J. Deng, "Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras," *Advances in neural information processing systems*, vol. 34, pp. 16 558–16 569, 2021. 3

[34] J. Deutscher, A. Blake, and I. Reid, "Articulated body motion capture by annealed particle filtering," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, vol. 2. IEEE, 2000, pp. 126–133. 3

[35] Y. Horry, K.-I. Anjyo, and K. Arai, "Tour into the picture: using a spidery mesh interface to make animation from a single image," in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 1997, pp. 225–232. 3

[36] C. Farhat, M. Lesoinne, and P. Le Tallec, "Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity," *Computer methods in applied mechanics and engineering*, vol. 157, no. 1-2, pp. 95–114, 1998. 3

[37] M. Gleicher, "Retargetting motion to new characters," in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, 1998, pp. 33–42. 3

[38] J. Lee, J. Chai, P. S. Reitsma, J. K. Hodgins, and N. S. Pollard, "Interactive control of avatars animated with human motion data," in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, 2002, pp. 491–500. 3

[39] R. W. Sumner and J. Popović, "Deformation transfer for triangle meshes," *ACM Transactions on graphics (TOG)*, vol. 23, no. 3, pp. 399–405, 2004. 3

[40] K. Aberman, P. Li, D. Lischinski, O. Sorkine-Hornung, D. Cohen-Or, and B. Chen, "Skeleton-aware networks for deep motion retargeting," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 62–1, 2020. 3

[41] J. Zhang, Z. Tu, J. Weng, J. Yuan, and B. Du, "A modular neural motion retargeting system decoupling skeleton and shape perception," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 3

[42] Y.-H. Huang, Y.-T. Sun, Z. Yang, X. Lyu, Y.-P. Cao, and X. Qi, "Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 4220–4230. 3, 9

12

[43] J. Waczynska, P. Borycki, J. Kaleta, S. Tadeja, and P. Spurek, "D-miso: Editing dynamic 3d scenes using multi-gaussians soup," *Advances in Neural Information Processing Systems*, vol. 37, pp. 107 865–107 889, 2024. 3

[44] B. F. Labs, "Flux," https://github.com/black-forest-labs/flux, 2024. 3

[45] T. Brooks, B. Peebles, C. Holmes, W. DePue, and Y. Guo, "Video generation models as world simulators," https://openai.com/index/video-generation-models-as-world-simulators/, 2024, openAI Technical Report. 3, 5

[46] R. Liu, R. Wu, B. Van Hoorick, P. Tokmakov, S. Zakharov, and C. Vondrick, "Zero-1-to-3: Zero-shot one image to 3d object," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 9298–9309. 3

[47] H. Liang, Y. Yin, D. Xu, H. Liang, Z. Wang, K. N. Plataniotis, Y. Zhao, and Y. Wei, "Diffusion4d: Fast spatial-temporal consistent 4d generation via video diffusion models," *arXiv preprint arXiv:2405.16645*, 2024. 3

[48] J. Tang, J. Ren, H. Zhou, Z. Liu, and G. Zeng, "Dreamgaussian: Generative gaussian splatting for efficient 3d content creation," *arXiv preprint arXiv:2309.16653*, 2023. 3

[49] A. Haque, M. Tancik, A. A. Efros, A. Holynski, and A. Kanazawa, "Instruct-nerf2nerf: Editing 3d scenes with instructions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 740–19 750. 4

[50] A. Raj, S. Kaza, B. Poole, M. Niemeyer, N. Ruiz, B. Mildenhall, S. Zada, K. Aberman, M. Rubinstein, J. Barron *et al.*, "Dreambooth3d: Subject-driven text-to-3d generation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 2349–2359. 4

[51] T. Igarashi, T. Moscovich, and J. F. Hughes, "As-rigid-as-possible shape manipulation," *ACM transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 1134–1141, 2005. 5

[52] C. Doersch, P. Luc, Y. Yang, D. Gokay, S. Koppula, A. Gupta, J. Heyward, I. Rocco, R. Goroshin, J. Carreira *et al.*, "Bootstap: Bootstrapped training for tracking-any-point," in *Proceedings of the Asian Conference on Computer Vision*, 2024, pp. 3257–3274. 6

[53] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763. 7

[54] D. Lazzarotto and T. Ebrahimi, "Towards a multiscale point cloud structural similarity metric," in *2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP)*, 2023, pp. 1–6. 7

[55] Q. Yang, Z. Ma, Y. Xu, Z. Li, and J. Sun, "Inferring point cloud quality via graph similarity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 6, pp. 3015–3029, 2022. 7

[56] T. Li, M. Slavcheva, M. Zollhoefer, S. Green, C. Lassner, C. Kim, T. Schmidt, S. Lovegrove, M. Goesele, R. Newcombe *et al.*, "Neural 3d video synthesis from multi-view video," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5521–5531. 8, 9

[57] T. Yin, Q. Zhang, R. Zhang, W. T. Freeman, F. Durand, E. Shechtman, and X. Huang, "From slow bidirectional to fast autoregressive video diffusion models," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2025. 8

[58] Y. HaCohen, N. Chiprut, B. Brazowski, D. Shalem, D. Moshe, E. Richardson, E. Levin, G. Shiran, N. Zabari, O. Gordon, P. Panet, S. Weissbuch, V. Kulikov, Y. Bitterman, Z. Melumian, and O. Bibi, "Ltx-video: Realtime video latent diffusion," *arXiv preprint arXiv:2501.00103*, 2024. 8

[59] Y. Wang, X. Wang, Z. Chen, Z. Wang, F. Sun, and J. Zhu, "Vidu4d: Single generated video to high-fidelity 4d reconstruction with dynamic gaussian surfels," in *Adv. Neural Inform. Process. Syst.*, 2024. 9

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We proposed the 4D restaging task, aiming at reanimating the 3D reconstruction using a single video input. We conducted several experiments to prove our motivation of using video information to improve the restaging quality. We want to explore the possibility of video-conditioned 4D content creation, and provide insights of learning motion articulation from real-world video and use them for down-stream tasks.

   Guidelines:
   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 5.

   Guidelines:
   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We used a Lemma for the main motivation, see Lemma 2. The corresponding proof is in the supplementary.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We listed the training details in Section 4. For the training methods we use, we listed the algorithm in Section 3. For all the metrics we used, we provide their calculation formulation in Section 4.1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will release code and data we use after the double-blinding review.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See Section 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: See the experiment table.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section 4.

   Guidelines:
   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Our research focuses on reanimating 4D content from single video input using publicly available datasets (*e.g.*, DAVIS, PointOdyssey) and pre-trained generative models. The videos used do not contain personally identifiable information or sensitive content. We do not involve human subjects or deploy the model in a way that affects people or communities. The purpose of our work is academic, and we have taken care to avoid generating or reinforcing any harmful biases. Thus, we believe our research adheres to the NeurIPS Code of Ethics.

   Guidelines:
   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: We discuss the societal impact of our work in the supplementary materials.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our work does not involve the release of any high-risk models, such as large-scale language models, generative adversarial networks for face synthesis, or web-scraped datasets that may contain unsafe content. The data used in our research is either publicly available or synthetically generated using existing safe models. As such, we believe no special safeguards are necessary.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We have properly cited DAVIS and PointOdessy.

Guidelines:

- The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We plan to release the full implementation of our Restage4D pipeline, along with example input videos and the reconstructed 4D scenes for academic use. Documentation will be provided alongside the code and data to describe usage instructions, training configuration, data formats, and licensing. All released assets are either synthetic or derived from publicly available datasets that do not involve personal or sensitive information.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: In the pipeline we used a LVM for prompt refinement, and we described the usage and showed its input and output in the supplementary.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

A   Proof of Lemma 2

We introduce Lemma 2 to demonstrate the necessity of using base video as additional input, which is

Lemma 2. *Let $x_A(t), x_B(t) \in \mathbb{R}^3$ be the positions of two 3D points at time $t \in [0, T]$, defined by a shared time-independent motion coefficient vector $\beta \in \mathbb{R}^K$, and a set of time-dependent, spatially smooth motion bases $\{M_k(t)\}_{k=1}^K$.*

Assume the following:

(i) The scene is trained over two segments: an edited clip $[0, t_1]$ without supervision, and an original clip $[t_1, t_2]$ with ground-truth geometry supervision;

(ii) A temporal smoothness regularization is applied on the motion bases $M_k(t)$, i.e., $\sum_{k,t} \|M_k(t+1) - M_k(t)\|^2$;

(iii) Each point's position at time t is given by a transformation composed from the bases and coefficients:

$$x(t) = \sum_{k=1}^K \beta_k \cdot M_k(t)(x_0),$$

where x_0 is the canonical position.

Then, the variance of the pairwise distance $d(t) = \|x_A(t) - x_B(t)\|$ within the edited clip satisfies:

$$\sigma_2 := Var_{t \in [0, t_1]}(d(t)) < \sigma_0,$$

where σ_0 is the variance obtained by training only on the edited clip without supervision.

We can prove this lemma using

Proof. Let $x_A^0, x_B^0 \in \mathbb{R}^3$ be fixed canonical positions of two points. At time t, their positions in world coordinates are given by:

$$x_i(t) = \sum_{k=1}^{K} \beta_k \cdot M_k(t)(x_i^0), \quad \text{for } i \in \{A, B\},$$

where $\beta \in \mathbb{R}^K$ is shared and fixed, and $M_k(t) : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ are time-varying motion basis functions.

Define the pairwise squared distance:

$$d^2(t) = \|x_A(t) - x_B(t)\|^2 = \left\| \sum_{k=1}^{K} \beta_k \cdot \left(M_k(t)(x_A^0) - M_k(t)(x_B^0) \right) \right\|^2.$$

Let $\Delta_k(t) := M_k(t)(x_A^0) - M_k(t)(x_B^0) \in \mathbb{R}^3$, then:

$$d^2(t) = \left\| \sum_{k=1}^{K} \beta_k \cdot \Delta_k(t) \right\|^2.$$

This is a quadratic form in the temporal functions $\Delta_k(t)$, linearly combined by fixed weights β_k. Its time-variance depends on the temporal variability of $\Delta_k(t)$.

Now, consider the regularized training setup:

(i) The motion bases $\{M_k(t)\}$ are supervised only in $[t_1, t_2]$, enforcing accurate deformation there;

(ii) A temporal smoothness regularization is imposed:

$$\mathcal{L}_{\text{smooth}} = \sum_{k=1}^{K} \sum_{t} \|M_k(t+1) - M_k(t)\|^2.$$

Due to this regularization, each $M_k(t)$ evolves smoothly over time. Let us denote the discrete temporal second difference of $\Delta_k(t)$ as:

$$\delta_k(t) := \Delta_k(t+1) - \Delta_k(t).$$

Then, smoothness implies that $\|\delta_k(t)\|$ is small, especially near the boundary $t = t_1$, where $M_k(t)$ is influenced by supervision at $t > t_1$. Since the functions $\Delta_k(t)$ are more constrained in this case (compared to unregularized training), their fluctuations in $[0, t_1]$ are suppressed.

Let us define:

$$f(t) := \sum_{k=1}^{K} \beta_k \cdot \Delta_k(t), \quad \text{then} \quad d^2(t) = \|f(t)\|^2.$$

The temporal variance of $d(t)$ satisfies:

$$\text{Var}_{t \in [0, t_1]}(d(t)) = \text{Var}_{t \in [0, t_1]}(\|f(t)\|).$$

Since $f(t)$ is a fixed linear combination of smoother functions $\{\Delta_k(t)\}$, its temporal fluctuation is reduced by the regularization. That is:

$$\text{Var}_{t \in [0, t_1]}(\|f(t)\|) \downarrow \quad \text{as} \quad \mathcal{L}_{\text{smooth}} \downarrow.$$

In contrast, training on $[0, t_1]$ alone without regularization leads to unconstrained $M_k(t)$, and thus larger temporal variability in $f(t)$. Therefore:

$$\sigma_2 := \text{Var}_{t \in [0, t_1]}(d(t)) < \sigma_0,$$

where σ_0 is the variance from training without supervision or smoothness.
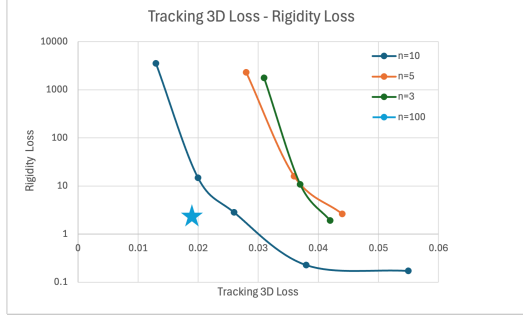
$\square$

Figure 7: Effect of changing the number of bases.

B Training Details

B.1 2D Priors

To enable 4D reconstruction from a monocular video, we leverage a set of 2D foundation models to provide reliable initialization. Following the pipeline of Shape-of-Motion, we use Segment Anything for foreground segmentation and Track Anything to propagate the masks across frames. For camera motion and point correspondence, we adopt Tapir for 2D point tracking and MegaSAM for estimating camera poses. Additionally, we incorporate dense depth maps from VideoDepthAnything to provide richer geometric details for the scene.

B.2 Hyperparameters

To achieve geometry-preserving reconstruction while maintaining high-frequency motion details, we found it essential to carefully tune the rigidity loss. Since the overall motion is primarily determined during the motion initialization stage, we evaluate the effect of different combinations of rigidity loss weights and the number of motion bases.

As shown in Figure 7, each curve represents the trade-off between object rigidity and tracking accuracy for a given number of motion bases, as we vary the rigidity loss weight from 0 to 10^{-1}. Our results show that increasing the number of motion bases can improve this trade-off curve, and that a loss weight of 10^{-3} offers a good balance. Based on this analysis, we use 100 motion bases and set the rigidity loss weight to 10^{-3} in all experiments.

C Samples from DAVIS Dataset

To qualitatively evaluate our 4D restaging pipeline, we present several representative samples from the DAVIS dataset in Figure 8. For each sequence, we show: (1) two keyframes from the base video (original monocular input); (2) the corresponding motion prompt describing the desired new motion; and (3) two keyframes from the generated driving video, synthesized using a pre-trained image-to-video model (*e.g.*, Sora) conditioned on the prompt and the first frame. These examples demonstrate the diversity of motion prompts supported by our pipeline, including changes in direction, speed, and posture. They also highlight the realism and coherence of the generated driving video, which serves as the motion source for our 4D reconstruction. For each example, we visualize the base and drive videos using two keyframes (top and bottom) to illustrate the overall temporal dynamics.

D Evaluation on Neural 3D Video Synthesis dataset

Evaluating the novel view by comparing with ground truth is important for validating view generalization in monocular reconstruction, but the groundtruth is hard to get. To address this, we conduct additional evaluation using the Neural 3D Video Synthesis dataset, which provides 20 calibrated camera views of dynamic scenes, from which we can derive ground truth for novel views We adopt the following protocols:

To simulate a monocular causal video with natural camera motion, we perform depth-first search (DFS) to find a smooth camera trajectory across the 20 views. We then sample 100 frames along

this path to construct a pseudo-monocular video. We treat the first 60 frames as the base video and the remaining 40 frames as the driving video. For evaluation, we select a camera that is not on the sampled path and render from this novel viewpoint, which has ground truth available for comparison. We compute PSNR scores for three training strategies: (i) training only on the generated video, (ii) joint training with the base video, and (iii) joint training with ARAP refinement.

E   More Results

To demonstrate generalization of the pipeline, we also provide visualization on some web-collected sequences shown in Figure 3.

F   Video Demo

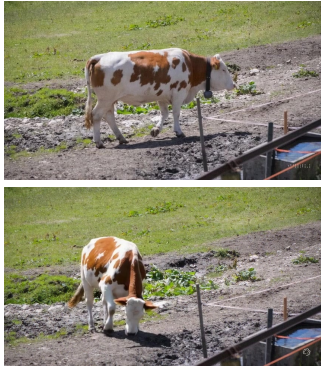Please refer to the video demo in the attachment.
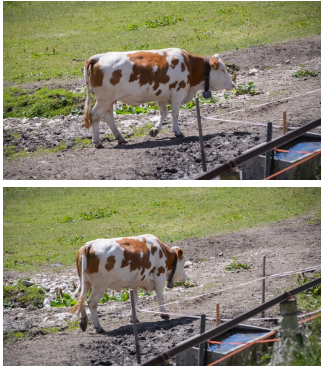
Base Video	Prompt	Drive Video

"A small silver car slowly drives up onto the sidewalk from the street in a quiet urban area. The camera stays fixed in position but smoothly pans to follow the vehicle's motion as it mounts the curb and continues onto the pedestrian pavement. Surrounding buildings are modern and light-colored, with crosswalk lines, bollards, and a few pedestrians in the background. The motion feels slightly unusual yet calm, with no abrupt changes in camera movement."

"A brown and white cow slowly turns its body and walks toward the grassy field to graze. The camera remains fixed in physical position but gently pans to follow the cow's movement as it turns and begins eating grass. The scene takes place in a quiet rural pasture under bright daylight, with patches of soil, stones, and fencing visible in the foreground. The motion is natural and unhurried, with the cow staying centered in the frame."

"A light-colored dog wearing a harness suddenly dashes into the nearby bushes on the right side of the frame. The camera remains completely still, capturing the dog's quick movement as it disappears into the foliage. The setting is a dry grassy area with sparse leaves, and a fence and road are visible in the background. The scene feels spontaneous and natural, with no camera tracking or zooming."

Figure 8: Examples of motion prompts and corresponding base/drive videos. Each column shows the input video, the motion prompt, and the generated video. Each video is visualized using two keyframes (top and bottom).
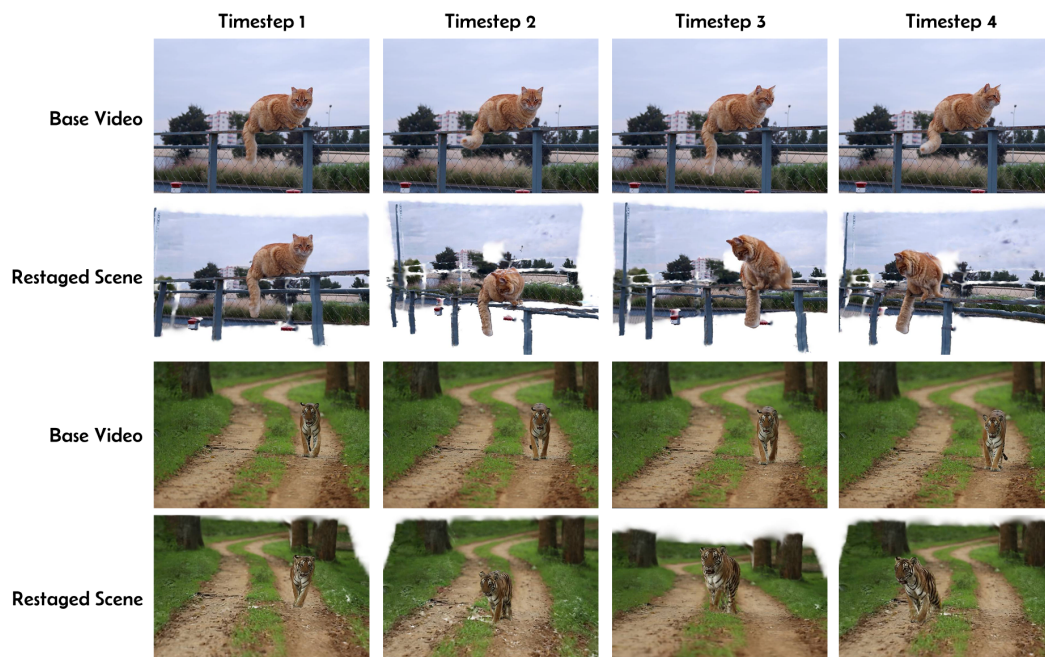
Figure 9: Some samples of 4D restaging on in-the-wild video, which are collected from Internet.