Gaussian See, Gaussian Do: Semantic 3D Motion Transfer from Multiview Video

YARIN BEKOR*, Technion - Israel Institute of Technology
GAL MICHAEL HARARI*, Technion - Israel Institute of Technology
OR PEREL, NVIDIA, University of Toronto, and Vector Institute
OR LITANY, Technion - Israel Institute of Technology and NVIDIA

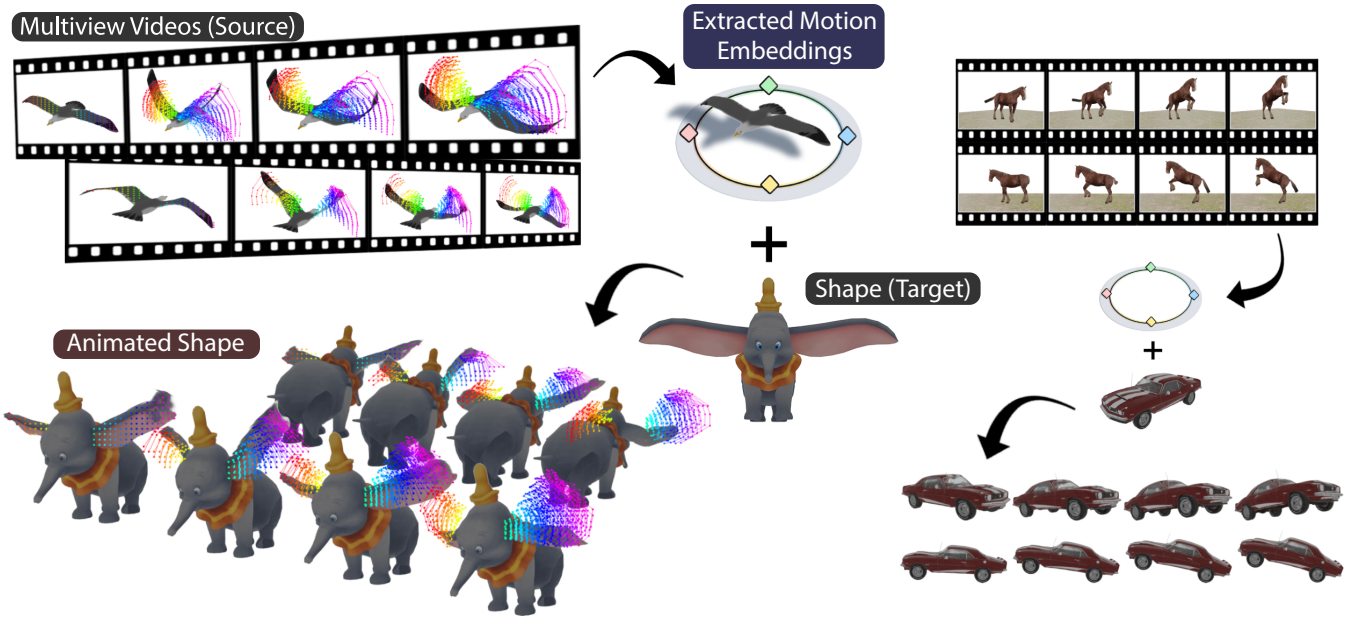

Fig. 1. **Semantic 3D Motion Transfer in Action**. Our method extracts motion embeddings from a multiview video and applies them to a static 3D Gaussian Splatting (3DGS) asset, bringing it to life with motion that matches the semantics of the source. **Left:** A bird's wing flapping motion is transferred to an elephant cartoon's ears. **Right:** A horse's rearing motion animates a vehicle lifting its front wheels. *We encourage watching the supplementary video for a clearer depiction of motion, which is best appreciated in dynamic form.*

We present Gaussian See, Gaussian Do, a novel approach for semantic 3D motion transfer from multiview video. Our method enables rig-free, cross-category motion transfer between objects with semantically meaningful correspondence. Building on implicit motion transfer techniques, we extract motion embeddings from source videos via condition inversion, apply them to rendered frames of static target shapes, and use the resulting videos to supervise dynamic 3D Gaussian Splatting reconstruction. Our approach introduces an anchor-based view-aware motion embedding mechanism, ensuring cross-view consistency and accelerating convergence, along with a robust 4D reconstruction pipeline that consolidates noisy supervision videos. We establish the first benchmark for semantic 3D motion transfer and demonstrate superior motion fidelity and structural consistency compared to adapted baselines. Code and data for this paper available at gsgd-motiontransfer.github.io.

*Equal contribution.

Authors' Contact Information: Yarin Bekor, Technion - Israel Institute of Technology, yarin.bekor@campus.technion.ac.il; Gal Michael Harari, Technion - Israel Institute of Technology, Gal.harari11@gmail.com; Or Perel, NVIDIA and University of Toronto and Vector Institute, orr.perel@gmail.com; Or Litany, Technion - Israel Institute of Technology and NVIDIA, or.litany@gmail.com.

1 Introduction

Animation is the art of bringing still objects and characters to life. The demand for realistic 3D animation is rapidly growing across industries, from gaming and virtual/augmented reality (VR/AR) to robotics and autonomous system simulations. As 3D content reconstruction and generation scales, the need for a controllable, data-driven approach to animating 3D objects across diverse categories is more important than ever. However, generating high-quality 3D motion remains a significant challenge. A major bottleneck is the lack of methods that enable 3D objects—especially those without predefined kinematic structures—to acquire motion in a semantically meaningful way. Traditional motion generation relies heavily on rigging, where a predefined skeletal structure governs movement. While well-established techniques exist for animating humans and, to some extent, animals [Baran and Popović 2007; Li et al. 2021; Ma and Zhang 2023; Xu et al. 2020], applying these principles to arbitrary, non-rigged objects remains an open challenge [Chu et al. 2024; Liu et al. 2025]. Recent advances in 3D Gaussian Splatting (3DGS) [Guo et al. 2024a] have streamlined the reconstruction of

3D assets from images, yet these representations lack an inherent motion structure, making animation even more difficult.

Even if we could control such objects, instructing them to move meaningfully is another challenge. Some methods rely on textual descriptions of motion [Bahmani et al. 2024b; Ling et al. 2024b; Petrovich et al. 2022; Ren et al. 2023; Singer et al. 2023; Tevet et al. 2022; Wang et al. 2024b; Zhang et al. 2024a; Zhao et al. 2023b; Zheng et al. 2024], but accurately describing complex motion in words is inherently limited [Petrovich et al. 2024]. Motion is often too nuanced for text-based instructions to capture effectively, making such methods imprecise and difficult to generalize. A more natural approach is motion transfer—enabling an object to mimic motion demonstrated by another. While much research has focused on motion retargeting within the same category [Aberman et al. 2020; Chen et al. 2023; Li et al. 2021; Raab et al. 2024; Sun et al. 2022; Villegas et al. 2018; Zhang et al. 2023b], transferring motion between objects that do not share an explicit kinematic mapping is significantly more challenging [Muralikrishnan et al. 2024].

Recently, semantic motion transfer techniques have emerged in 2D video generation [Kansy et al. 2024; Ling et al. 2024a; Materzyńska et al. 2023; Wang et al. 2024a; Wei et al. 2024; Xiao et al. 2024; Zhao et al. 2023a], disentangling motion intent from appearance and transferring it to different targets. However, these capabilities remain absent in 3D, despite their significant potential and necessity.

We propose a novel method for semantic 3D motion transfer from multiview video, allowing motion from a dynamic source object to be adapted to a static 3D target object while ensuring realistic and coherent movement.

We assume access to multiview video of the source motion to recover its full dynamics, since a single-view input can miss occluded regions—for example, an octopus articulating its legs differently. Such multiview videos may come from either computer animation, or multiview captures obtainable with simple off-the-shelf equipment, as demonstrated by the AIST Dance dataset [Tsuchida et al. 2019].

Uniquely, our method does not rely on structural similarity between the source and target, enabling motion transfer across vastly different object categories—such as a horse rearing to a sports car lifting its front wheels. Building on condition inversion in generative video models [Gal et al. 2022; Kansy et al. 2024], we extend these principles to 3D through a view-aware motion embedding mechanism. Specifically, we introduce an anchor-based interpolator that optimizes embeddings collaboratively across views, significantly accelerating convergence while enhancing motion fidelity. We use these embeddings to generate target supervision videos, which, while capturing the intended motion, often contain artifacts and inconsistencies.

To address these imperfections, we propose a robust 4D reconstruction pipeline that effectively transforms noisy supervision signals into high-quality, temporally stable motion reconstructions. Specifically, we employ a 3D Gaussian splatting representation of the target object and infer its animation by optimizing a motion field applied to a set of control points governing deformation. Through carefully designed regularization strategies, we demonstrate that even when supervision videos are inconsistent, our method produces realistic and stable 4D reconstructions.

To facilitate rigorous evaluation and drive progress on this underexplored and challenging problem setup, we introduce the first benchmark for semantic 3D motion transfer from multiview video. It spans diverse, cross-category motion scenarios and includes carefully curated source-target pairs of structurally different 3D objects, along with tailored evaluation metrics for both motion fidelity and structural consistency.

Our work makes the following key contributions:

- We introduce a **new problem setup for semantic 3D motion transfer**, where motion from a multiview video of a dynamic source object is adapted to a static 3D Gaussian Splatting (3DGS) target, even across vastly different categories, without requiring predefined skeletal correspondences or structural similarity.
- We propose a novel **view-aware motion embedding** strategy, optimized collaboratively through an anchor-based interpolator. This approach balances global coherence with view-dependent details, ensuring motion fidelity while significantly accelerating convergence.
- We develop a **robust 4D reconstruction pipeline** that refines noisy, artifact-prone supervision videos into high-quality, temporally stable dynamic 3D reconstructions.
- We introduce a **benchmark for semantic 3D motion transfer**, evaluating diverse cross-category motion transfers between structurally different 3D objects.

Our method outperforms adapted baselines in both motion fidelity and structural consistency, delivering high-quality motion transfer across diverse, cross-category scenarios. A user study further confirms that our results are consistently preferred by human evaluators. Beyond synthetic settings, we also demonstrate compelling results on real-world scenes by animating 3D assets reconstructed from in-the-wild imagery (see Fig. 4), highlighting the practicality and robustness of our approach.

2 Related Work

4D Generation. Over the past years, there has been a surge of research in 4D content generation. [Bahmani et al. 2024a,b; Jiang et al. 2024; Ling et al. 2024b; Miao et al. 2024; Ren et al. 2023; Uzolas et al. 2024; Wu et al. 2024b; Zeng et al. 2024] primarily focus on full generation, where both the 3D target object and its motion are synthesized from textual input. Despite their capabilities, these generation methods cannot incorporate unique motion patterns from inputs videos, or to apply movement while preserving the target's distinctive identity. [Ling et al. 2024b; Ren et al. 2023] split their pipeline, by first generating a static 3D figure, then animating it, potentially allowing for target identity preservation.

3D Motion Transfer. Transferring motion from source input onto a target 3D figure, often referred to as "Motion Retargeting", is a well-studied problem [Aberman et al. 2020; Kim et al. 2024; Raab et al. 2024; Villegas et al. 2018; Wang et al. 2025; Zhang et al. 2024b, 2023b]. However, these methods typically rely on rigged figures and are predominantly constrained to human characters. In contrast, our approach allows for motion transfer onto arbitrary 3D objects without requiring manual rigging, also enabling cross-category motion transfer. One exception is the generative work of SC4D [Wu

et al. 2024b] which introduces 3D motion transfer without requiring a rigged figure.

Generative Models Based Motion Transfer. Recent years have witnessed significant advancements in video generation models [Blattmann et al. 2023; Guo et al. 2024b; Wang et al. 2023; Xing et al. 2023; Zhang et al. 2023a]. These models provide a rich motion prior, encouraging researchers to distill, represent, and apply the motion learned by these models [Kansy et al. 2024; Ling et al. 2024a; Materzyńska et al. 2023; Wang et al. 2024a; Wei et al. 2024; Xiao et al. 2024; Yatim et al. 2024; Zhao et al. 2023a]. Typically, these approaches extract motion from a reference video and apply it as a motion embedding to a new image or prompt, generating a new video that replicates the reference motion. However, they remain limited to 2D motion transfer. Our approach extends this paradigm to the 3D domain, leveraging the 2D learned motion priors to enable rigless, cross-category motion transfer, effectively bridging the gap between 2D motion learning and 3D animation.

3 Preliminaries

Semantic 2D Motion Transfer. Our setting assumes a conditional latent Image-to-Video diffusion model [Blattmann et al. 2023], which also uses an additional embedding input, e.g. a CLIP [Radford et al. 2021] embedding of a text prompt or image. [Kansy et al. 2024] demonstrated that such video diffusion models [Blattmann et al. 2023; Zhang et al. 2023a] exhibit an implicit disentanglement in the way motion and appearance are encoded, where the image input controls the *appearance* and the embedding input manipulates the *motion* of the generated video. Based on this finding, Kansy et al. [Kansy et al. 2024] use condition embedding inversion [Gal et al. 2022] to learn a multiframe representation $\mathbf{m}^* \in \mathbb{R}^{M\times(F+1)\times d}$, where M denotes the number of d-dimensional tokens, capturing the motion of a source video $\mathbf{x}_0 \in \mathbb{R}^{F\times H\times W\times 3}$ with F frames of size $H \times W$. This motion representation can then be transferred to animate images of different subjects while largely preserving the semantic intent of the original motion. Formally, the process is optimized through a frozen video diffusion denoiser D_θ by minimizing the denoising score matching loss:

$$\mathbf{m}^* = \arg\min_{\mathbf{m}} \mathbb{E}\left[\lambda_\sigma \|D_\theta(\mathbf{x}_0 + \boldsymbol{n}; \sigma, \boldsymbol{m}, \boldsymbol{c}) - \mathbf{x}_0\|_2^2\right], \tag{1}$$

where the expectation is w.r.t $\sigma, \boldsymbol{n} \sim p(\sigma, \boldsymbol{n})$, $\boldsymbol{c}$ encompasses condition signals (e.g. first frame), and
$p(\sigma, \boldsymbol{n}) = p(\sigma)\mathcal{N}(\mathbf{n}; \mathbf{0}, \sigma^2 \boldsymbol{I})$ with $p(\sigma)$ being the probability distribution over noise level σ. Then $\boldsymbol{n}$ denotes the noise, and λ_σ is a weighting function. For the rest of the manuscript we will drop $\boldsymbol{c}$ and replace $\mathbf{x}_0 + \boldsymbol{n}$ with $\widetilde{\mathbf{x}}_0$.

3D Gaussian Splatting and Low-Dimension Controllability. Our setting assumes an input static target object, represented with 3D Gaussian Splats (3DGS) [Kerbl et al. 2023]. 3DGS optimizes differentiable primitives to model volumetric radiance fields. Specifically, Gaussian particles are projected and alpha-composited through rasterization to obtain high-quality 3D reconstructions of objects and scenes. An object can be represented as a set of 3D Gaussians $\mathcal{G}_i$, each parameterized with $\{\boldsymbol{\mu}_i, \boldsymbol{q}_i, \boldsymbol{s}_i, \sigma_i, \boldsymbol{c}_i\}$, with mean $\boldsymbol{\mu}_i$, rotation $\boldsymbol{q}_i$, scale $\boldsymbol{s}_i$, opacity σ_i and view dependent color $\boldsymbol{c}_i$ modeled with Spherical Harmonic coefficients. The contribution of a single Gaussian $\mathcal{G}_i$ to the radiance field at point $\boldsymbol{x} \in \mathbb{R}^3$ is expressed by the probability density function: $\mathcal{G}(\boldsymbol{x}) = e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\Sigma_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)}$, where the covariance can be decomposed to $\Sigma_i = R_i S_i S_i^T R_i^T$, with R_i, S_i being the rotation matrix derived from the rotation quaternion $\boldsymbol{q}_i$ and scale vector $\boldsymbol{s}_i$ respectively. The final color of a pixel is determined by all Gaussians that overlap it, by alpha-blending in depth order:

$$C = \sum_i \boldsymbol{c}_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) \tag{2}$$

Here, α_i is a function of each Gaussian's opacity and PDF post-projection to image coordinates.

Building on this foundation, SC-GS [Huang et al. 2024] extends 3DGS for dynamic scenes by decomposing canonical appearance and motion control using 3D Gaussians $\mathcal{G}_i$ and control points $\{\boldsymbol{p}_k \in \mathbb{R}^3\}$ with a learnable coordinate and radius. A time-conditioned MLP predicts the rotation and translation of control points at time t as: $\Psi : (\boldsymbol{p}_k, t) \rightarrow (R_k^t | T_k^t)$. Gaussians are then deformed using Linear Blend Skinning (LBS) [Sumner et al. 2007], where the warped position $\boldsymbol{\mu}_i^t$ and rotation $\boldsymbol{q}_i^t$ for Gaussian $\mathcal{G}_i$ are computed as:

$$\boldsymbol{\mu}_i^t = \sum_{k\in\mathcal{N}_i} w_{ik}\left(R_k^t(\boldsymbol{\mu}_i - \boldsymbol{p}_k) + \boldsymbol{p}_k + T_k^t\right); \boldsymbol{q}_i^t = \left(\sum_{k\in\mathcal{N}_i} w_{ik}\boldsymbol{r}_k^t\right)\otimes \boldsymbol{q}_i \tag{3}$$

such that the weights w_{ik} are based on the distances d_{ik} between the center of Gaussian $\mathcal{G}_i$ and its neighboring control points $\mathcal{N}_i$ and their learned radii. $\boldsymbol{r}_k$ is the quaternion form of R_k and $\otimes$ symbolizes a product of quaternions.

4 Semantic 3D Motion Transfer

We aim to transfer the dynamics of a source object $\mathcal{V}_S$, represented through a sparse set of N videos captured by static cameras positioned at different views, to a static target object of a different structure and posture, $\mathcal{G}_T$, represented as 3DGS [Kerbl et al. 2023] (Section 3). Our goal is to reenact $\mathcal{G}_T$ with the semantic motion patterns of $\mathcal{V}_S$ while preserving its identity, appearance and geometry as faithfully as possible. Formally, we take as input tensor $\mathcal{V}_S \in \mathbb{R}^{N\times F\times H\times W\times 3}$, of N videos of F frames, where each video is associated with known camera extrinsics $[R_{C_i}|T_{C_i}] \in SE(3)$. Our pipeline is made of two stages. First, we extract structured motion embeddings $\boldsymbol{m}_1^*..\boldsymbol{m}_K^*$ from $\mathcal{V}_S$ using a pre-trained video diffusion model D_θ [Blattmann et al. 2023], where $K \leq N$. In the second stage, we fix a set of control points $\mathcal{P}_T$, and use them to drive the target shape $\mathcal{G}_T$. This process is governed by optimizing a decoder Ψ, with self-supervised labels generated by frozen video diffuser D_θ conditioned on the previously extracted motion embeddings $\boldsymbol{m}_1^*..\boldsymbol{m}_K^*$. Our setting allows the same motion code to be reused over multiple target shapes.

4.1 Structured Multiview Motion Inversion

Inspired by the 2D motion inversion introduced in [Kansy et al. 2024], we devise a motion inversion process in 3D by extracting motion embeddings from each input view-video. Lifting the inversion technique of [Kansy et al. 2024] to 3D, however, is not straightforward. Through experimentation, we observe that trained motion

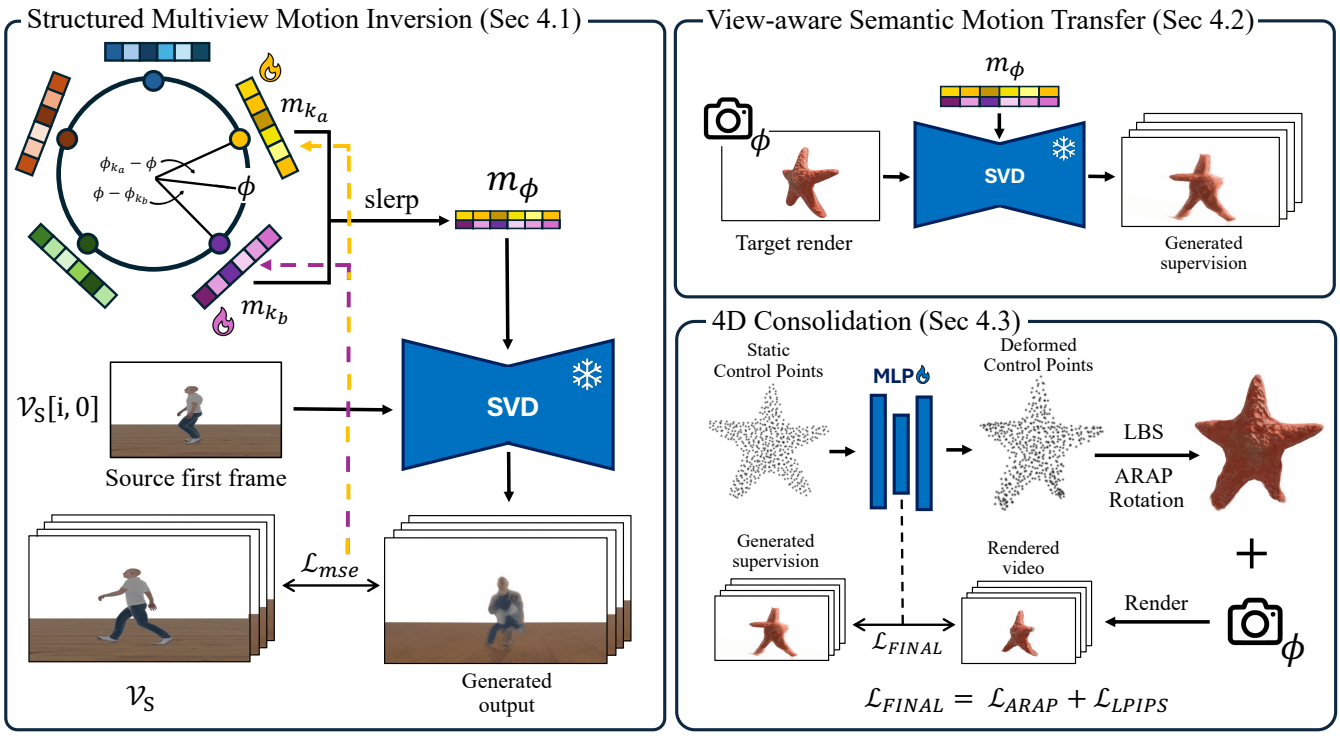

Fig. 2. Pipeline Overview. (1) **Structured Multiview Motion Inversion.** We extract motion embeddings from the source using the slerp interpolation from the two nearest achor points. (2) **View-aware Semantic Motion Transfer.** We use the motion embeddings to generate supervision for the motion transfer process, and then (3) **4D Consolidation.** We apply the supervision onto the target shape to introduce actual motion

embeddings that have been optimized for a specific angle are not necessarily suitable for generating motion from a different angle, as shown in Figure 6. Intuitively, one can imagine that the pixels depicting a "kicking" animation appearing somewhat different in orthogonal directions. One simple solution is to optimize motion embeddings $\boldsymbol{m}_1^*, \ldots, \boldsymbol{m}_N^*$ for each source view $\{V_S[i]\}_{i=1}^N$:

$$\boldsymbol{m}_i^* = \arg\min_{\boldsymbol{m}_i} \mathbb{E}\left[\lambda_\sigma \| D_\theta(\widetilde{\mathcal{V}}_S[i]; \sigma, \boldsymbol{m}_i) - \mathcal{V}_S[i] \|_2^2\right], \tag{4}$$

and use them along with corresponding rendered views of the static 3DGS target object to generate N videos of the target object at matching angles, which then serve as supervision for 4D reconstruction. However, this approach is suboptimal for several reasons. First, it is inefficient in terms of runtime and storage costs - as it requires training and bookkeeping a separate motion embedding for each possible source angle. That leads to increased computational and storage costs as the number of embeddings grows linearly with the number of videos in $\mathcal{V}_S$. Additionally, this method lacks generalization, as embeddings trained on specific angles cannot be reused or adapted to unseen viewpoints, further limiting its flexibility and applicability.

To overcome these limitations, we hypothesize that since motions in different views share a common underlying 3D motion, their embeddings should share information. Since that does not seem to naturally occur in the standard per view inversion procedure, we propose to instill this relation by inducing view structure into the optimization procedure. By doing so we aim to not only make the resulting embeddings more generalizable but also speed up the optimization procedure. Specifically, we propose to optimize a fixed number of K anchor motion embeddings $\{\boldsymbol{m}_i\}_{i=1}^K$, distributed evenly across a range of source view angles $\{\phi_i\}_{i=1}^K$. When a motion embedding for a specific angle $\phi \in \mathbb{R}$ is required, we perform a spherical linear interpolation between the two closest anchor embeddings with:

$$\begin{aligned} &\boldsymbol{m}_\phi = \text{slerp}(\boldsymbol{m}_i, \boldsymbol{m}_j, \frac{\phi - \phi_i}{\phi_j - \phi_i}) \text{ s.t.} \\ &i = \arg\max_k \{\phi_k | \phi_k \le \phi\} \; ; \; j = (i \bmod K) + 1 \end{aligned} \tag{5}$$

Here, $\text{slerp}(\cdot, \cdot; \phi)$ denotes Spherical Linear Interpolation, ensuring smooth transitions between the closest anchor embeddings $\boldsymbol{m}_i^*$ and $\boldsymbol{m}_j^*$. The parameter ϕ controls the interpolation, allowing for a continuous blend between the two anchors. This formulation enables the model to dynamically adjust motion embeddings while maintaining efficiency and scalability. To sum up, each iteration of our training loop samples a single video $\mathcal{V}_S[i]$ of angle ϕ, and uses Equation 5 to obtain an interpolated code m_ϕ, which we then feed into the score matching function:

$$\{\boldsymbol{m}_k^*\}_1^K = \underset{\{\boldsymbol{m}_k\}}{\arg\min} \mathbb{E}\left[\lambda_\sigma \| D_\theta(\widetilde{\mathcal{V}_S}[i]; \sigma, \boldsymbol{m}_\phi) - \mathcal{V}_S[i] \|_2^2\right]. \quad (6)$$

It then follows that our backwards pass updates multiple anchor motion embeddings at once.

4.2 View-aware Semantic Motion Transfer

After extracting motion embeddings $\{\boldsymbol{m}_i^*\}_{i=1}^K$ from the multi-view video $\mathcal{V}_S$, our pipeline can readily apply them to a 3D shape represented with 3DGS [Kerbl et al. 2023] for flexible motion transfer. First, we render an image of the target, $\mathcal{R}[\mathcal{G}_T; C]$, with random camera $C \in SE(3)$, sampled uniformly. Then, we use it to generate 2D supervision video $\mathcal{V}_T$ by conditioning the denoising process at the right side of Equation 6 on the rendered image $\mathcal{R}[\mathcal{G}_T; C]$, and its corresponding motion embedding $\boldsymbol{m}_\phi$. This process can be applied to any number of views, as motion embeddings can be interpolated between anchors, allowing flexible generation of supervision videos from different perspectives.

4.3 4D Consolidation

Our final step is a consolidation process, transforming the generated supervisions into a 4D representation. Since the static target shape is already provided, we model the 4D scene as a dynamic-3DGS representation leveraging the SC-GS foundation (Section 3), e.g. our goal is to drive the 3DGS static shape with control points. We begin by initializing a fixed set of control points $\mathcal{P}_T$ over target shape $\mathcal{G}_T$, using furthest point sampling. Then, we use the skinning scheme depicted in Equation 3 with a relaxed weighting scheme: $w_{ik} = exp\left(-{d_{ik}}^2/2\beta^2\right)$, where d_{ik} is the distance between gaussian i and control point k, and β is a non-learned scalar which modulates the area of effect per control point.

Given multi-view supervision videos $\mathcal{V}_T$, we optimize a deformation field network Ψ to map each control point $\boldsymbol{p}_k \in \mathcal{P}_T$ to a transformation $(R_k^t | T_k^t)$ that when applied, aligns $\mathcal{G}_T$ with the animation depicted within the supervision $\mathcal{V}_T$. That is, our loss function minimizes the discrepancy between the rendered dynamic 3DGS and the supervision videos. The supervision videos, being the output of a video generative model, are inherently noisy and spatially inconsistent, see Appendix B.3. Hence, using them directly as supervision leads to a poorly reconstructed object (Figure 7, Vanilla).

ARAP Loss and ARAP Rotation. The As-Rigid-As-Possible (ARAP) loss [Sorkine and Alexa 2007] is a widely used technique [Huang et al. 2024; Li et al. 2024; Luiten et al. 2024] for preserving local rigidity during deformation by minimizing distortions in the transformed shape. In its standard form, ARAP loss is computed for each control point $p_k \in P_T$ and its nearest neighbors $\mathcal{N}_k$, ensuring that local relative positions remain consistent across transformations:

$$\mathcal{L}_{ARAP} = \sum_{t \in 1..F} \sum_{i \in \mathcal{N}k} w_{ik} || (p_i^t - p_k^t) - \hat{R}_k^t (p_i - p_k) ||^2 \quad (7)$$

where the optimal rotation $\hat{R}_k^t$ is estimated as:

$$\hat{R}_k^t = \underset{R}{\text{argmin}} \sum_{i \in \mathcal{N}_k} w_{ik} || (p_i^t - p_k^t) - R(p_i - p_k) ||^2 \quad (8)$$

However, a key limitation of the standard ARAP loss is that it only regularize the positions of control points while leaving their associated rotations R_k^t unconstrained. This lack of rotational supervision leads to visible artifacts in the deformation process, as seen in Figure 7, 'Vanilla'. These artifacts arise from inaccurate rotation predictions by the mlp Ψ which result from noisy and inconsistent supervision. To address this issue, we propose an ARAP Rotation mechanism, which explicitly enforces rigid rotational constraints during training. Specifically, instead of relying on the MLP-predicted rotation R_k^t, we replace it with the optimized rotation $\hat{R}_k^t$ which is derived by minimizing the same rigid motion assumption as in Equation 8. In contrast to previous approaches, where such explicit constraints are typically applied only in a post-processing step, we integrate this constraint directly into the training process.

Perceptual Loss. When optimizing 4D reconstruction against consistent and clean videos, MSE and mask loss are a popular and pragmatic choice [Li et al. 2024; Wu et al. 2024b]. However, in scenarios with spatial inconsistencies and significant noise, as is in our supervision videos, pixel-wise comparisons become insufficient, and the learned deformations tend to inherit the spatial inconsistency from the supervision itself, resulting in distorted reconstructed structure. To address this, we replace pixel-wise losses with the LPIPS [Zhang et al. 2018] loss, which operates in a perceptual space rather than enforcing explicit pixel-wise correspondences. This approach has previously been shown to be effective previously [Wu et al. 2024a] in mitigating reconstruction artifacts. By leveraging perceptual similarity rather than direct pixel alignment, our model becomes more robust to view inconsistencies, reducing unwanted deformations introduced by artifacts and leading to more stable and visually coherent reconstructions.

SDS Loss. A widely used strategy for reconstruction with generative models under view inconsistencies is the Score Distillation Sampling (SDS) loss [Poole et al. 2022], commonly adopted in 4D generation (e.g. [Wu et al. 2024b]). We experimented with adding either plain SDS Loss or Iterative Dataset Update [Haque et al. 2023], but found both to underperform compared to our method (see E.2).

5 Semantic Motion Transfer Benchmark

We introduce the first benchmark for general, cross-category 3D motion transfer from multiview video. Since no existing benchmark supports this task, we construct one to enable systematic evaluation. Our benchmark combines curated source videos and static 3D target objects drawn from a subset of the Mixamo dataset [Inc. 2024] and from publicly available assets collected from the web. Notably, the Mixamo subset includes paired motions—the same motion performed by both a source and a target character—allowing for reference-based evaluation, as described in the next section. The web-crawled, cross-category subset features a diverse range of motions (human, animal, and object) and targets (from skeletons to robotic arms), supporting broad evaluation of motion transfer across semantically and structurally different entities. All target assets were reconstructed using 3DGS. A detailed explanation about the benchmark can be found in Appendix A.

6 Experimental Evaluation

Evaluation Metrics. As no established metric exists for semantic motion transfer in 3D, we adopt the *Motion Fidelity* measure from [Yatim et al. 2024], originally proposed for 2D video motion. We also assess appearance preservation using *CLIP-I* [Rahamim et al. 2024; Zhao et al. 2023b], and similarity to ground truth via the *CLIP similarity score*. See Appendix C for details.

Methods in Comparison. To the best of our knowledge, no prior work tackles semantic motion transfer from multiview video to a 3DGS-represented target object. We therefore adapt the most relevant methods to our setting to enable meaningful comparisons.

Our first baseline is SC4D [Wu et al. 2024b], a 2D-to-4D generation method that takes a 2D video and a textual prompt describing the target's appearance to synthesize a dynamic 3DGS output. Unlike our approach, SC4D does not perform direct motion transfer but instead *modifies the appearance* of the generated object based on the prompt. To align it with our setup, we adapt SC4D in two ways: (1) we run its 2D-to-4D optimization using a primary view of the source video to capture relevant motion patterns; (2) we employ GPT-4o to generate a detailed textual description of the target, which SC4D uses to condition its appearance generation and match the intended object.

Our second baseline is DreamGaussians4D (DG4D)[Ren et al. 2023], an image-to-4D model that generates a dynamic 3DGS output from a single 2D image. DG4D first produces a driving video and a static 3D reconstruction, then optimizes a deformation field using Score Distillation Sampling (SDS) with a generative view synthesis model. To adapt it to our task, we replace the single input image with the full 3D target object and substitute the driving video with a reenacted source motion sequence, as described in Section4.1.

For our method, we use 5 anchor embeddings, 16 input views, 3K inversion steps, and 5K optimization iterations during 4D reconstruction. Further implementation details are provided in Appendix B.

6.1 Comparing with Baselines

Table 1. **Quantitative Comparison with Baselines.** Our method achieves significantly higher Motion Fidelity, CLIP-I, and CLIP similarity scores across both benchmarks compared to the baselines. We encourage the reader to refer to Figure 3 for a visual illustration of these improvements.

	Motion Fidelity ↑		**CLIP-I ↑**		**CLIP Score ↑**
	Mini-Mixamo	Cross-Category	Mini-Mixamo	Cross-Category	Mini-Mixamo
SC4D Motion Transfer	0.65	0.56	0.888	0.872	0.905
DreamGaussians4D	0.61	0.54	0.931	0.908	0.945
Ours	**0.74**	**0.66**	**0.950**	**0.948**	**0.963**
Ground Truth	0.87	-	-	-	1.0

A quantitative comparison with the adapted baselines is presented in Table 1. Our method outperforms both SC4D and DG4D in Motion Fidelity and CLIP score, demonstrating superior alignment between source and target motions while preserving the target object's structure and identity. To further illustrate these improvements, Figure 3 provides a qualitative evaluation, showing (Left) rendered views of the target object and source videos, and (Right) views of the reconstructed dynamic 3DGS outputs from our method and the baselines. As shown, while SC4D captures the correct motion to some extent, its text-based appearance transfer introduces a significant visual discrepancy from the original object. Similarly, DG4D, despite leveraging the target's original 3DGS, struggles to maintain motion coherence over time, as reflected in both quantitative scores and observable distortions. In contrast, our approach produces visually coherent reconstructions while faithfully transferring the semantic essence of the motion. For example, our method accurately captures the stalking crouch of the skeleton transferred from the panther and the sliding motion of the robotic arm, transferred from a simple cube—demonstrating its ability to generalize across diverse object categories and motion types.

6.2 In-the-wild Motion Transfer

Leveraging a video diffusion prior trained on real-world footage, our model generalizes beyond synthetic settings by enabling the application of motion embeddings to arbitrary 3D targets in real scenes (see Fig.4). This capability is increasingly important given the rapid progress in 3DGS, which supports high-quality reconstructions from in-the-wild imagery. As 3DGS pipelines become more accessible and robust for real-world data, the ability to animate such assets with semantically meaningful motion becomes a key step toward practical deployment.

6.3 Human Preference Study

We conducted a human preference study to evaluate the perceived quality of semantic motion transfer. Participants compared side-by-side outputs from different methods and rated them based on visual fidelity (i.e., adherence to the 3D asset) and motion plausibility. Results are shown in Fig. 5. On the left, our method outperforms adapted baselines in appearance quality, with an average rating of 4.66/5, and is the only one to consistently preserve object identity. For motion quality, our approach matches SC4D—though SC4D fails to maintain appearance—and surpasses DG4D.

On the right, we show an ablation study on regularization. Both ARAP rotation and LPIPS loss improve perceived motion realism and appearance fidelity. Survey examples and interface screenshots are provided in Sec. E.3 of the supplementary.

6.4 Novel View Motion Synthesis

Our anchor-based mechanism is primarily designed to accelerate motion transfer and collaborate motion cues across input views. However, due to its interpolatory nature, it is natural to examine whether it exhibits the emergent property of synthesizing motion embeddings at unseen views. To evaluate this, we interpolate the anchors to create novel-view embeddings and pair them with a single frame from a previously unseen viewpoint to synthesize a novel-view video of the source object. Table 2 presents the results. Remarkably, the output exhibits excellent motion consistency and structural integrity. Furthermore, we show that simple interpolation of simple motion embeddings or a single global embedding both fail at this task, underscoring the effectiveness of our approach (See visual comparison in Fig. 6). This capability extends to the target object as well (see Appendix D). We hypothesize that this ability could be particularly valuable for viewpoint densification

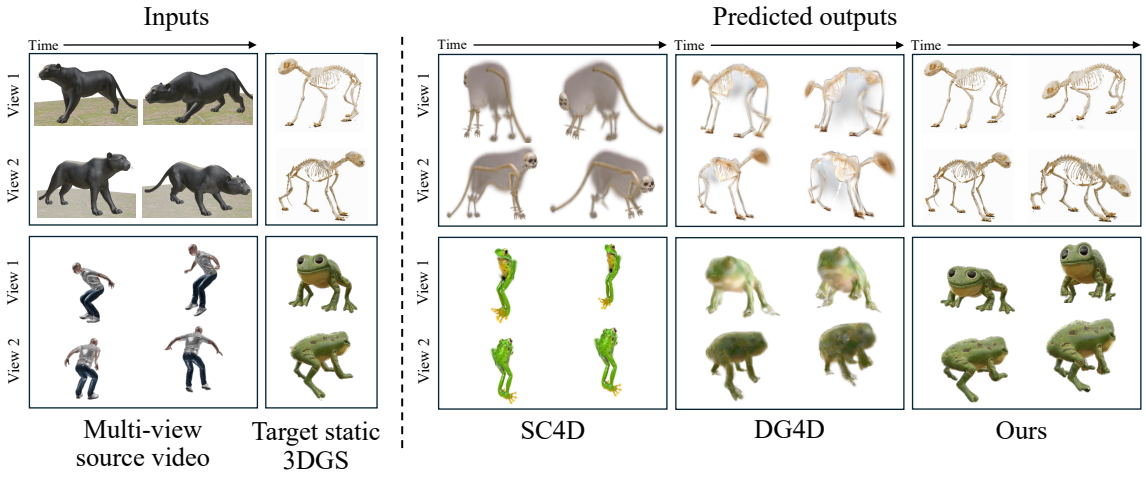

Fig. 3. **Qualitative comparison of semantic 3D motion transfer.** We compare our method to adapted baselines, showing two views of the source motion, the target 3DGS object, and the generated output for each method. We demonstrate superior identity preservation while also accurately transferring the source motion to the target.

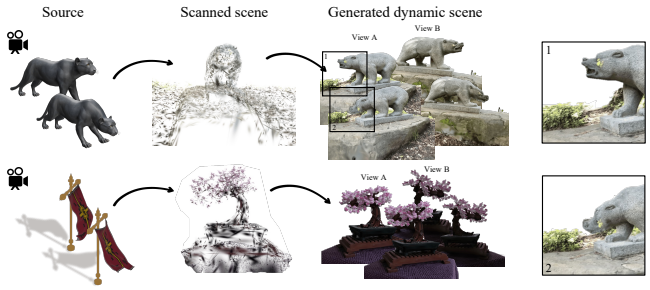

Fig. 4. **In-the-wild Motion Transfer.** Our method animates 3D assets reconstructed from real-world imagery, demonstrating robust semantic motion transfer in real scenes using 3DGS. The 'Scanned scene' column visualizes sparse sample of the 3DGS scene produced by our reconstruction stage, highlighting that the motion is applied directly to the scene.

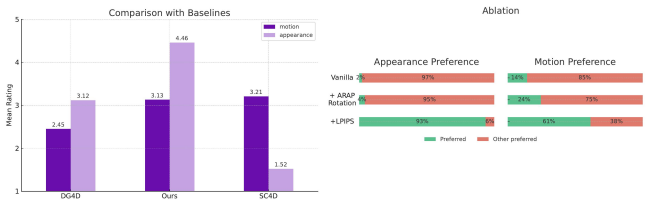

Fig. 5. **Human preference study.** Left: Mean subjective ratings for motion plausibility and appearance fidelity. Our method is the only one to preserve target identity while delivering high-quality motion transfer. Right: Preference results from an ablation study, showing that both LPIPS and ARAP rotation substantially improve perceptual quality.

applications. Incorporating this capability into a motion transfer pipeline is an exciting avenue for future work.

Table 2. Quantitative results for Novel-view Motion Synthesis using MSE and LPIPS. Metrics are computed only on viewing angles unseen during training.

Method	MSE ↓	LPIPS ↓
Simple	0.0111 ± 0.0034	0.1058 ± 0.0179
Global	0.0069 ± 0.0049	0.0649 ± 0.0306
Ours	**0.0028 ± 0.0016**	**0.0403 ± 0.0170**

6.5 Ablation Studies

Number of anchors. A key parameter in our method is the number of anchors, K, which trades off convergence speed and reconstruction quality. As shown in Figure 8, fewer anchors speed up convergence by encouraging broader view consistency, but too few—e.g., a single global embedding—degrade performance and cause motion hallucinations. We choose $K = 5$ as a balanced setting. Visual examples and further analysis appear in Appendix E.1.

Regularization terms. We refine the reconstruction pipeline using ARAP Rotation and LPIPS regularization, which significantly improve structural integrity and fine-detail preservation (Figure 7). These terms enhance texture stability, maintaining legibility of details like text and intricate patterns. See Appendix E.2 for details.

7 Conclusion & Limitations

We introduced Gaussian See, Gaussian Do, pioneering the first approach to lift implicit video diffusion motion transfer to 3D, achieving semantic 3D motion transfer from multiview video to 3DGS. We also established the first benchmark for this task, demonstrating superior performance over adapted baselines. However, we identify several limitations as exciting directions for future work. Runtime remains a challenge, as condition inversion is computationally expensive, and while our anchor-based mechanism accelerates convergence, further speed improvements are necessary. Additionally,

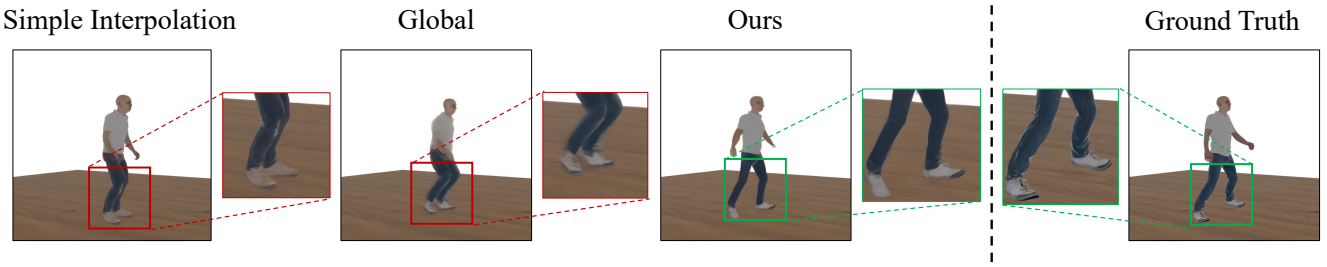

Fig. 6. **Qualitative comparison of novel-view motion synthesis**. Interpolating simple motion embeddings (Eq. 4) and a single global embedding both fail to generalize motion embeddings to views unseen during optimization. In contrast, our anchor-based mechanism successfully recovers faithful motion. A similar behavior is observed for the target object before and after reconstruction (Appendix D).

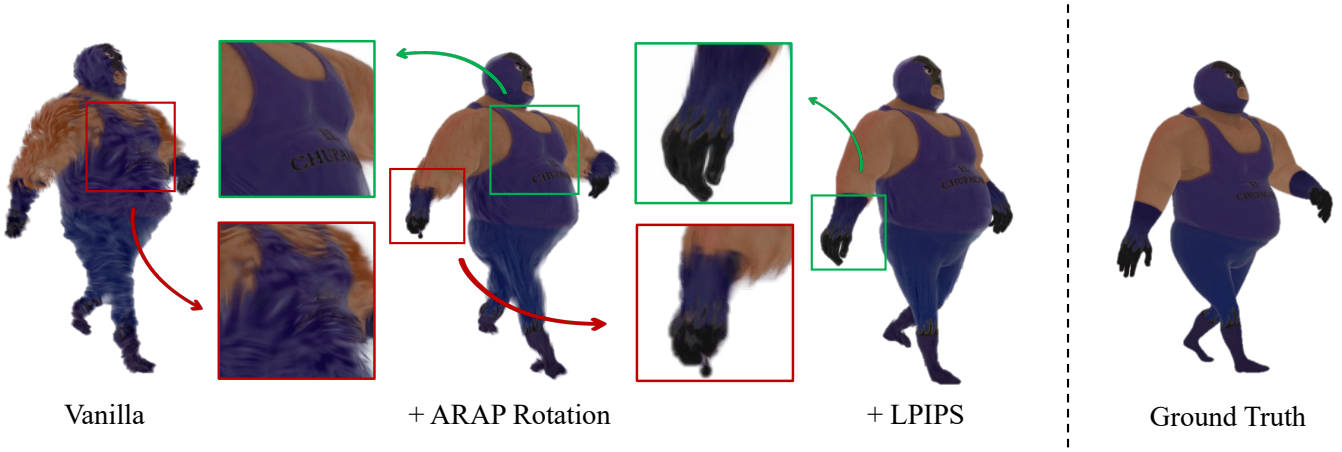

Fig. 7. **Qualitative comparison of reconstruction improvements.** The vanilla reconstruction introduces geometric and texture artifacts, while ARAP rotation and LPIPS regularization significantly enhance reconstruction quality, preserving fine details.

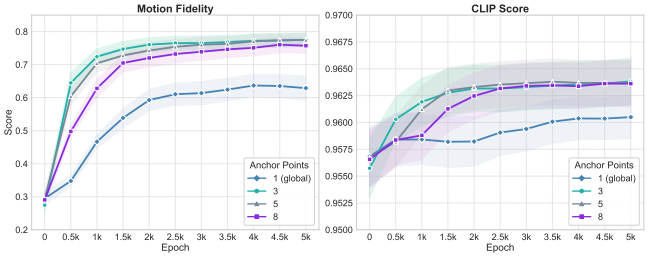

Fig. 8. **Effect of number of anchors.** Reducing anchors speeds up convergence, but too few results in both poor quality and slower optimization. See Section 6.5 for details.

the lack of a robust 3D semantic motion metric highlights a need for better evaluation tools. Finally, our method shows promise in synthesizing motion at novel views, and we aim to incorporate this capability into the pipeline to further boost performance.

Release & Licensing. We release the code and benchmark under the BigCode OpenRAIL-M v0.1 license, which explicitly prohibits impersonation and deepfakes without consent, as specified in clause h.

Acknowledgments. We thank Manuel Kansy for helpful discussions. Or Litany acknowledges the support of the Israel Science Foundation (grant No. 624/25) and the Azrieli Foundation Early Career Faculty Fellowship.

References

Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. 2020. Skeleton-Aware Networks for Deep Motion Retargeting. ACM Transactions on Graphics (TOG) 39, 4 (2020), 62.

Sherwin Bahmani, Xian Liu, Wang Yifan, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, Andrea Tagliasacchi, and David B. Lindell. 2024a. TC4D: Trajectory-Conditioned Text-to-4D Generation. arXiv (2024).

Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B. Lindell. 2024b. 4D-fy: Text-to-4D Generation Using Hybrid Score Distillation Sampling. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024).

Ilya Baran and Jovan Popović. 2007. Automatic rigging and animation of 3d characters. ACM Transactions on graphics (TOG) 26, 3 (2007), 72–es.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. 2023. Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv preprint arXiv:2311.15127 (2023).

Jinnan Chen, Chen Li, and Gim Hee Lee. 2023. Weakly-supervised 3d pose transfer with keypoints. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 15156–15165.

Zedong Chu, Feng Xiong, Meiduo Liu, Jinzhi Zhang, Mingqi Shao, Zhaoxu Sun, Di Wang, and Mu Xu. 2024. HumanRig: Learning Automatic Rigging for Humanoid

Character in a Large Scale Dataset. arXiv preprint arXiv:2412.02317 (2024).
Victor G Turrisi Da Costa, Giacomo Zara, Paolo Rota, Thiago Oliveira-Santos, Nicu Sebe, Vittorio Murino, and Elisa Ricci. 2022. Dual-head contrastive domain adaptation for video action recognition. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 1181–1190.
Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion. arXiv preprint arXiv:2208.01618 (2022).
Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. 2021. Multimodal neurons in artificial neural networks. Distill 6, 3 (2021), e30.
Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. 2024b. AnimateDiff: Animate Your Personalized Text-to-Image Diffusion Models without Specific Tuning. arXiv:2307.04725 [cs.CV] https://arxiv.org/abs/2307.04725
Zhiyang Guo, Jinxu Xiang, Kai Ma, Wengang Zhou, Houqiang Li, and Ran Zhang. 2024a. Make-It-Animatable: An Efficient Framework for Authoring Animation-Ready 3D Characters. arXiv preprint arXiv:2411.18197 (2024).
Ayaan Haque, Matthew Tancik, Alexei Efros, Aleksander Holynski, and Angjoo Kanazawa. 2023. Instruct-NeRF2NeRF: Editing 3D Scenes with Instructions. In Proceedings of the IEEE/CVF International Conference on Computer Vision.
Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. 2024. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 4220–4230.
Adobe Systems Inc. 2024. Mixamo. https://doi.org/10.57702/xbooguyb DOI retrieved: December 2, 2024.
Ajay Jain, Matthew Tancik, and Pieter Abbeel. 2021. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 5885–5894.
Yanqin Jiang, Li Zhang, Jin Gao, Weiming Hu, and Yao Yao. 2024. Consistent4D: Consistent 360° Dynamic Object Generation from Monocular Video. In The Twelfth International Conference on Learning Representations. https://openreview.net/forum?id=sPUrdFGepF
Manuel Kansy, Jacek Naruniec, Christopher Schroers, Markus Gross, and Romann M Weber. 2024. Reenact Anything: Semantic Video Motion Transfer Using Motion-Textual Inversion. arXiv preprint arXiv:2408.00458 (2024).
Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. 42, 4 (2023), 139–1.
Boeun Kim, Jungho Kim, Hyung Jin Chang, and Jin Young Choi. 2024. MoST: Motion Style Transformer between Diverse Action Contents. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 1705–1714.
Peizhuo Li, Kfir Aberman, Rana Hanocka, Libin Liu, Olga Sorkine-Hornung, and Baoquan Chen. 2021. Learning skeletal articulations with neural blend shapes. ACM Transactions on Graphics (TOG) 40, 4 (2021), 1–15.
Zhiqi Li, Yiming Chen, and Peidong Liu. 2024. Dreammesh4d: Video-to-4d generation with sparse-controlled gaussian-mesh hybrid representation. Advances in Neural Information Processing Systems 37 (2024), 21377–21400.
Huan Ling, Seung Wook Kim, Antonio Torralba, Sanja Fidler, and Karsten Kreis. 2024b. Align Your Gaussians: Text-to-4D with Dynamic 3D Gaussians and Composed Diffusion Models. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
Pengyang Ling, Jiazi Bu, Pan Zhang, Xiaoyi Dong, Yuhang Zang, Tong Wu, Huaian Chen, Jiaqi Wang, and Yi Jin. 2024a. MotionClone: Training-Free Motion Cloning for Controllable Video Generation. arXiv preprint arXiv:2406.05338 (2024).
Isabella Liu, Zhan Xu, Wang Yifan, Hao Tan, Zexiang Xu, Xiaolong Wang, Hao Su, and Zifan Shi. 2025. RigAnything: Template-Free Autoregressive Rigging for Diverse 3D Assets. arXiv preprint arXiv:2502.09615 (2025).
Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. 2024. Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis. In 3DV.
Jing Ma and Dongliang Zhang. 2023. TARig: Adaptive template-aware neural rigging for humanoid characters. Computers & Graphics 114 (2023), 158–167.
Joanna Materzyńska, Josef Sivic, Eli Shechtman, Antonio Torralba, Richard Zhang, and Bryan Russell. 2023. Customizing Motion in Text-to-Video Diffusion Models. arXiv preprint arXiv:2312.04966 (2023).
Qiaowei Miao, Yawei Luo, and Yi Yang. 2024. PLA4D: Pixel-Level Alignments for Text-to-4D Gaussian Splatting. arXiv:2405.19957 [cs.CV]
Sanjeev Muralikrishnan, Niladri Dutt, Siddhartha Chaudhuri, Noam Aigerman, Vladimir Kim, Matthew Fisher, and Niloy J. Mitra. 2024. Temporal Residual Jacobians for Rig-Free Motion Transfer. In Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LVIII (Milan, Italy). Springer-Verlag, Berlin, Heidelberg, 93–109. doi:10.1007/978-3-031-73636-0_6
Mathis Petrovich, Michael J Black, and Gül Varol. 2022. Temos: Generating diverse human motions from textual descriptions. In European Conference on Computer Vision. Springer, 480–497.
Mathis Petrovich, Or Litany, Umar Iqbal, Michael J Black, Gul Varol, Xue Bin Peng, and Davis Rempe. 2024. Multi-track timeline control for text-driven 3D human motion generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 1911–1921.
Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. 2022. DreamFusion: Text-to-3D using 2D Diffusion. arXiv (2022).
Sigal Raab, Inbar Gat, Nathan Sala, Guy Tevet, Rotem Shalev-Arkushin, Ohad Fried, Amit Haim Bermano, and Daniel Cohen-Or. 2024. Monkey see, monkey do: Harnessing self-attention in motion diffusion for zero-shot motion transfer. In SIGGRAPH Asia 2024 Conference Papers. 1–13.
Sigal Raab, Inbal Leibovitch, Guy Tevet, Moab Arar, Amit H Bermano, and Daniel Cohen-Or. 2023. Single motion diffusion. arXiv preprint arXiv:2302.05905 (2023).
Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In International Conference on Machine Learning. https://api.semanticscholar.org/CorpusID:231591445
Ohad Rahamim, Ori Malca, Dvir Samuel, and Gal Chechik. 2024. Bringing Objects to Life: 4D generation from 3D objects. arXiv preprint arXiv:2412.20422 (2024).
Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. 2023. DreamGaussian4D: Generative 4D Gaussian Splatting. arXiv preprint arXiv:2312.17142 (2023).
Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, and Yaniv Taigman. 2023. Text-To-4D Dynamic Scene Generation. arXiv:2301.11280 (2023).
Olga Sorkine and Marc Alexa. 2007. As-rigid-as-possible surface modeling. In Symposium on Geometry processing, Vol. 4. Citeseer, 109–116.
Robert W. Sumner, Johannes Schmid, and Mark Pauly. 2007. Embedded deformation for shape manipulation. ACM Trans. Graph. 26, 3 (July 2007), 80–es. doi:10.1145/1276377.1276478
Yang-Tian Sun, Qian-Cheng Fu, Yue-Ren Jiang, Zitao Liu, Yu-Kun Lai, Hongbo Fu, and Lin Gao. 2022. Human motion transfer with 3d constraints and detail enhancement. IEEE Transactions on Pattern Analysis and Machine Intelligence 45, 4 (2022), 4682–4693.
Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. 2022. Human motion diffusion model. arXiv preprint arXiv:2209.14916 (2022).
Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and Masataka Goto. 2019. AIST Dance Video Database: Multi-genre, Multi-dancer, and Multi-camera Database for Dance Information Processing. In Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019. Delft, Netherlands.
Lukas Uzolas, Elmar Eisemann, and Petr Kellnhofer. 2024. MotionDreamer: Zero-Shot 3D Mesh Animation from Video Diffusion Models. arXiv preprint arXiv:2405.20155 (2024).
Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. 2018. Neural kinematic networks for unsupervised motion retargetting. In Proceedings of the IEEE conference on computer vision and pattern recognition. 8639–8648.
Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. 2023. ModelScope Text-to-Video Technical Report. arXiv:2308.06571 [cs.CV] https://arxiv.org/abs/2308.06571
Luozhou Wang, Ziyang Mai, Guibao Shen, Yixun Liang, Xin Tao, Pengfei Wan, Di Zhang, Yijun Li, and Yingcong Chen. 2024a. Motion Inversion for Video Customization. arXiv:2403.20193 [cs.CV] https://arxiv.org/abs/2403.20193
Rong Wang, Wei Mao, Changsheng Lu, and Hongdong Li. 2025. Towards High-Quality 3D Motion Transfer with Realistic Apparel Animation. In European Conference on Computer Vision. Springer, 35–51.
Xinzhou Wang, Yikai Wang, Junliang Ye, Fuchun Sun, Zhengyi Wang, Ling Wang, Pengkun Liu, Kai Sun, Xintong Wang, Wende Xie, et al. 2024b. Animatabledreamer: Text-guided non-rigid 3d model generation and reconstruction with canonical score distillation. In European Conference on Computer Vision. Springer, 321–339.
Yujie Wei, Shiwei Zhang, Zhiwu Qing, Hangjie Yuan, Zhiheng Liu, Yu Liu, Yingya Zhang, Jingren Zhou, and Hongming Shan. 2024. DreamVideo: Composing Your Dream Videos with Customized Subject and Motion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 6537–6549.
Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor Verbin, Jonathan T Barron, Ben Poole, et al. 2024a. Reconfusion: 3d reconstruction with diffusion priors. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 21551–21561.
Zijie Wu, Chaohui Yu, Yanqin Jiang, Chenjie Cao, Wang Fan, and Xiang. Bai. 2024b. SC4D: Sparse-Controlled Video-to-4D Generation and Motion Transfer. arxiv:2404.03736 (2024).
Zeqi Xiao, Yifan Zhou, Shuai Yang, and Xingang Pan. 2024. Video Diffusion Models are Training-free Motion Interpreter and Controller. arXiv preprint arXiv:2405.14864 (2024).

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Xintao Wang, Tien-Tsin Wong, and Ying Shan. 2023. DynamiCrafter: Animating Open-domain Images with Video Diffusion Priors. arXiv preprint arXiv:2310.12190 (2023).

Zhan Xu, Yang Zhou, Evangelos Kalogerakis, Chris Landreth, and Karan Singh. 2020. Rignet: Neural rigging for articulated characters. arXiv preprint arXiv:2005.00559 (2020).

Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. 2024. Space-time diffusion features for zero-shot text-driven motion transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 8466–8476.

Yifei Zeng, Yanqin Jiang, Siyu Zhu, Yuanxun Lu, Youtian Lin, Hao Zhu, Weiming Hu, Xun Cao, and Yao Yao. 2024. STAG4D: Spatial-Temporal Anchored Generative 4D Gaussians. arXiv:2403.14939 [cs.CV] https://arxiv.org/abs/2403.14939

Jiaxu Zhang, Xin Chen, Gang Yu, and Zhigang Tu. 2024b. Generative motion stylization of cross-structure characters within canonical motion space. In Proceedings of the 32nd ACM International Conference on Multimedia. 7018–7026.

Jiaxu Zhang, Junwu Weng, Di Kang, Fang Zhao, Shaoli Huang, Xuefei Zhe, Linchao Bao, Ying Shan, Jue Wang, and Zhigang Tu. 2023b. Skinned Motion Retargeting with Residual Perception of Motion Semantics & Geometry. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 13864–13872.

Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. 2024a. Motiondiffuse: Text-driven human motion generation with diffusion model. IEEE transactions on pattern analysis and machine intelligence 46, 6 (2024), 4115–4128.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In CVPR.

Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. 2023a. I2VGen-XL: High-Quality Image-to-Video Synthesis via Cascaded Diffusion Models. arXiv:2311.04145 [cs.CV] https://arxiv.org/abs/2311.04145

Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jiawei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. 2023a. MotionDirector: Motion Customization of Text-to-Video Diffusion Models. arXiv preprint arXiv:2310.08465 (2023).

Yuyang Zhao, Zhiwen Yan, Enze Xie, Lanqing Hong, Zhenguo Li, and Gim Hee Lee. 2023b. Animate124: Animating one image to 4d dynamic scene. arXiv preprint arXiv:2311.14603 (2023).

Yufeng Zheng, Xueting Li, Koki Nagano, Sifei Liu, Otmar Hilliges, and Shalini De Mello. 2024. A unified approach for text-and image-guided 4d scene generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 7300–7309.

A Additional details on Benchmark

A.1 Mini-Mixamo Rig-Less benchmark

To facilitate the evaluation of 3D rigless motion transfer, we introduce a benchmark derived from the Mixamo dataset [Inc. 2024]. While motion transfer has been extensively studied in rigged settings, no existing benchmark enables controlled and quantitative assessment of motion transfer in a rig-less setup. Our benchmarks aims to bridge this gap by providing a standardized evaluation protocol for future research in this under-explored domain. The core challenge in evaluating motion transfer lies in the need for paired data: a source object performing a given motion and a ground-truth target object executing the same motion. This pairing allows computation of transfer accuracy by measuring discrepancies between the predicted and actual target motion. Such pairing naturally exists in the Mixamo dataset, which is already a widely used benchmark in the rigged motion transfer domain [Da Costa et al. 2022; Raab et al. 2023; Zhang et al. 2023b]. Leveraging this property, we construct our benchmark by selecting two source figures performing ten distinct motions and ten target figures, for which we provide both their static canonical poses and their ground-truth motion executions.

A.2 Web-Crawled Inter- and Cross-Category Dataset

A key aspect of 3D cross-domain motion transfer is enabling motion adaptation across diverse, rigless objects without requiring predefined skeletal structures. Existing datasets rely heavily on Mixamo, which is limited in diversity and requires rigging, restricting the scope of motion transfer benchmarks. In contrast, our dataset removes this constraint, introducing motion transfer for a wider variety of human and non-human targets. This necessitates an unsupervised evaluation approach, for which we use the Motion Fidelity metric to assess motion plausibility as well as target identity preserving. In doing so, we establish a benchmark for this setting.
Our dataset serves two main purposes:

Inter-Category Diverse Motion Tranfer. By eliminating the reliance on rigged characters, we provide a more diverse and challenging evaluation set for human-to-human and animal-to-animal motion transfer. Unlike prior datasets constrained to Mixamo's limited rigged models, ours enables motion adaptation to a broader range of rigless human representations.

Cross-Category Motion Transfer. We extend motion transfer beyond human-to-human interactions, covering human-to-"human-like" and animal-to-"animal-like" motion transfer. While these categories contain fewer target objects, they introduce meaningful diversity beyond conventional datasets.
To construct the dataset, we sourced free, publicly available static 3D objects from the web and preprocessed them into a 3D Gaussian Splatting (3DGS) format. This preprocessing step was essential for our setup. Additionally, we built our multi-view motion sequences using free, publicly available animated 3D objects, rendering them into multi-view videos. Our dataset includes 31 diverse motion sequences applied to 33 rigless 3DGS target objects, covering humans, animals, and objects with semantic correspondence to humans or animals. All assets were independently curated to ensure a diverse and challenging dataset.

A.3 Data Production - Mini-Mixamo

Our dataset is derived from a portion of the Mixamo dataset and follows a structured pipeline to generate both source and ground truth videos, as well as target object reconstructions. Below, we outline the steps taken to construct the dataset. Two characters, Brian and Megan, were chosen as source figures, while ten target characters were used: Aj, Amy, Castle Guard 01, Crypto, Eve, Knight, Megan, Michelle, Mousey, Ortiz, and Sporty Granny.

To ensure consistency, all source and target objects were aligned in a canonical pose with their right foot toe set at . Additionally, a 10x10 plane with a wooden texture (laminate_floor_03 from PolyHaven) was added at position (0,0, z canon) beneath the target figures. For rendering, fourteen frames of the source figures were captured along with the wooden plane to create the source videos for each motion. Similarly, fourteen frames of both the source and target figures were rendered without the wooden plane to generate the ground truth videos.

For the reconstruction of target objects, we rendered 100 images per object using the BlenderNeRF plugin in Blender. The camera was set at a radius of 4, and a global exposure of 3.5 was applied to maintain visual consistency. These rendered images were then used to train each object with the original 3DGS repository for 30,000 iterations, with the maximum spherical harmonic order (max_sh) set to 0.

This structured approach ensured consistency across our dataset, allowing for accurate comparisons and evaluations in our experiments.

A.4 Data Production - Web Crawled

Our dataset includes a diverse range of motions and target objects, encompassing human, animal, and inanimate subjects. In total, we incorporated thirteen human motions, with the majority (eleven) sourced from the Mixamo benchmark and the remaining two from Sketchfab. Additionally, we included eleven animal-specific motions, all sourced from Sketchfab, and seven motions corresponding to objects.

The dataset also features a variety of targets across different categories. For animal-related targets, we included twelve instances, ranging from skeleton models to animal toys, all acquired from Sketchfab. Additionally, we incorporated six animal-like object targets, primarily consisting of robotic figures and uniquely shaped tables. For human-related targets, we gathered ten different models from Sketchfab, including humanoid robots and figures demonstrating secondary motion characteristics. Furthermore, we introduced five human-like object targets, such as clothing items, robotic arms, and wooden cross structures.

This broad range of motions and targets ensures diversity in our dataset, allowing for extensive evaluation of semantic motion transfer and adaptation across different entities.

As in A.3, we rendered 100 images per object using the BlenderNeRF plugin in Blender. These rendered images were then used to train each object with the original 3DGS repository for 30,000 iterations, with the maximum spherical harmonic order (max sh) set to 0. All of these reconstructed objects were then manually scaled and rotated to ensure pose and scale consistency across the dataset.

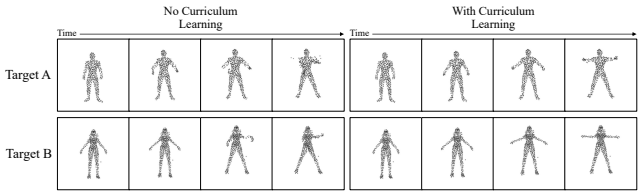

Fig. 9. **Qualitative comparison of motion learning with and without curriculum learning.** The left side demonstrates the control points of a dynamic object after 4D consolidation without curriculum learning, where the model struggles to generalize and produces broken transitions. The right side also shows the control points of a dynamic object, but with curriculum learning durin consolidation, leading to smoother and more natural motion transitions. This effect is particularly significant for cases with large motion variations.

B Method

B.1 Implementation details

Our model uses 5 anchors, 3000 inversion iterations, and motion embeddings of size 15×5×1024. As a backbone, the generative video model is the SVD model (stabilityai/stable-video-diffusion-img2vid). With control points radius of $\beta = 0.0745$ and 512 control points initialized via FPS. The MLP structure encodes input time into a 12D vector and control points into a 60D vector using positional encoding. These are concatenated, projected to 256D via a linear layer with ReLU, followed by a 4-layer 256→256 ReLU loop. A residual connection concatenates the input, followed by a 256D linear layer with ReLU, a 2-layer 256→256 ReLU loop, and a final linear layer reducing output to 3D for deformation prediction.

B.2 Curriculum Learning

One optional component we experimented with, to attempt and fix "hard" examples is Curriculum Learning. During training, we noticed that the model struggles to directly learn poses that deviate significantly from the canonical target shape. Furthermore, it tends to overfit to these challenging poses, leading to discrete, step-like motion rather than a smooth, continuous transition. To address this, we introduce curriculum learning, where motion is gradually incorporated over time. Initially, the model learns only small deformations, focusing on local consistency. As training progresses, the allowed deformation range expands, enabling the model to generalize to larger motion variations while maintaining smooth, coherent movement. We note that while curriculum learning plays a crucial role in certain cases, its impact is primarily significant when the magnitude of motion change is large, requiring the model to handle substantial deformations. In cases with smaller motion variations, it is mostly unnecessary. Figure 9 showcase qualitative analysis of one of the fail cases curriculum fix.

B.3 Noisy Supervisions

To better illustrate the noisy 2D supervisions generated by the Stable Diffusion model, we provide two examples of the resulting videos. Each figure presents a matrix in which each column corresponds to a video frame, progressing from left to right. The first row displays a single-view recording of the source motion, followed by multiple rows showing 2D supervision videos from different viewpoints, generated by applying the motion to the target 10,11.

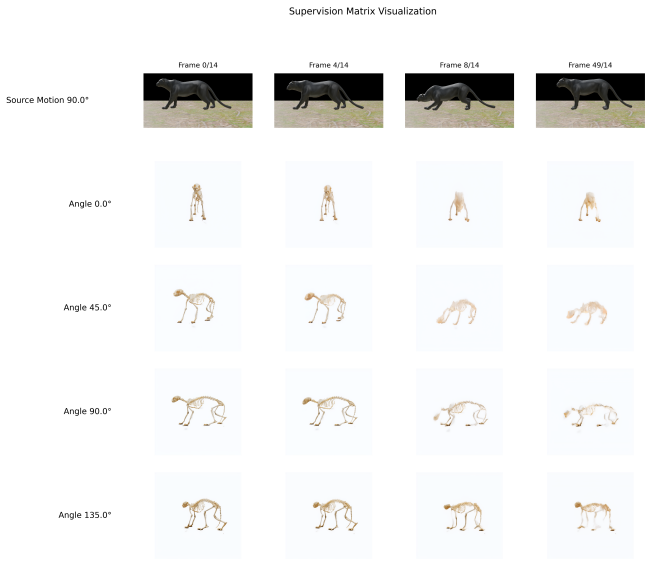

Fig. 10. **Supervision videos example 1.** These are the videos generated by the video model, conditioned on the interpolated motion embedding and on the rendered first frame. They are being used as the supervision videos during the 4D consolidation stage of our method.

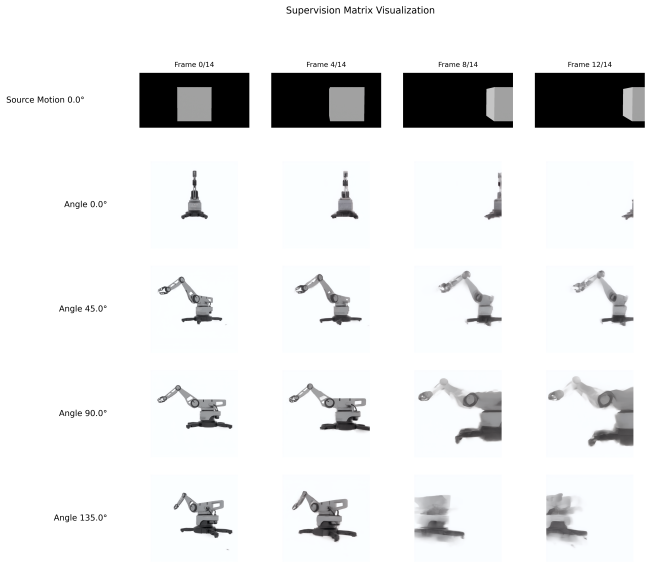

Fig. 11. **Supervision videos example 2.** These are the videos generated by the video model, conditioned on the interpolated motion embedding and on the rendered first frame. They are being used as the supervision videos during the 4D consolidation stage of our method.

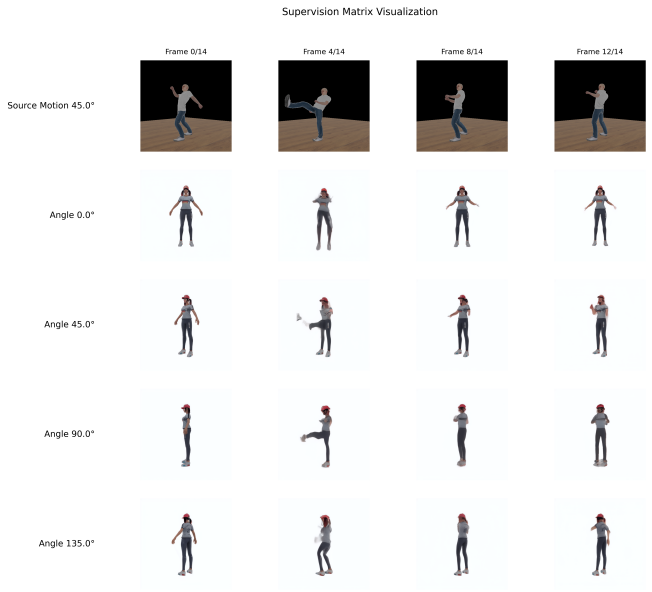

Fig. 12. **Supervision videos example - Failure case.** The generated videos in this example are highly inconsistent and introduce noise into the consolidation process, resulting in visible artifacts in the final dynamic scene.

Fig. 13. **Final result example - Failure case.** Final result rendered at a 45° angle, showing the kicking motion on the woman figure. The video supervisions leading to this failure are shown in Fig. 12.

B.4 Failure Cases

Our method struggles with highly articulated motions, such as kicking or jumping jacks. In these cases, per-view motion inversion often fails, causing the synthesized reference videos to exhibit severe artifacts and lose synchronization. Once this happens, our consolidation phase cannot recover, and the artifacts propagate into the final generated motion. We illustrate this in Figure 12, which shows errors in the generated supervision for such a complex motion. Figure 13 illustrates the final outcome—failure to move the legs and breakage of the thin arms. Since the main bottleneck of our method stems from the noisy video supervisions generated by the 2D video model, it is important to emphasize that our method will improve as inversion techniques in video models improve.

C Evaluation Metrics

C.1 3D Motion Fidelity Metric

We observe that baselines often introduce degeneracies, such as a hand remaining static and moving along with the torso instead of articulating independently. The original Motion Fidelity metric computes a two-way trajectory distance, which fails to penalize such degenerate cases due to the lack of a bijectivity requirement. To address this, we modify the metric to measure motion fidelity in only one direction—matching target trajectories to the closest source trajectories. Formally, we define our modified Motion Fidelity ($\mathrm{MF}_{3\mathrm{D}}$) as:

$$MF_{3\mathrm{D}} = \frac{1}{V}\sum_{v=1}^{V}\frac{1}{m}\sum_{\tilde{\tau}\in\text{Target}}\max_{\tau\in\text{Source}}\mathbf{corr}(\tau,\tilde{\tau}), \tag{9}$$

where V is the number of rendered views, m is the number of tracklets in the source view, and $\mathrm{corr}(\tau,\tilde{\tau})$ remains unchanged:

$$\mathbf{corr}(\tau,\tilde{\tau}) = \frac{1}{F}\sum_{k=1}^{F}\frac{v_t^x\cdot\tilde{v}_t^x + v_t^y\cdot\tilde{v}_t^y}{\sqrt{(v_t^x)^2+(v_t^y)^2}\cdot\sqrt{(\tilde{v}_t^x)^2+(\tilde{v}_t^y)^2}} \tag{10}$$

An additional advantage of this measure is that it does not require ground truth, allowing us to evaluate on both of our benchmarks.

C.2 CLIP-I and CLIP score

In addition to motion quality, we assess appearance preservation using *CLIP-I* [Rahamim et al. 2024; Zhao et al. 2023b], which measures the cosine similarity between the CLIP visual features of the initial image rendered from the static object and each frame in the predicted video. This provides a reference-free, view-invariant signal for how well the object's visual identity is preserved throughout the animation. We also compute *CLIP similarity* between individual frames of the predicted dynamic 3DGS and reference target videos. This frame-level comparison requires both source and target to perform approximately the same motion and is therefore only applied to the Mini-Mixamo benchmark. As with Motion Fidelity, all CLIP-based metrics are averaged across rendered views. While CLIP is robust to viewpoint variation and sensitive to visual attributes[Goh et al. 2021; Jain et al. 2021], it may overestimate similarity in synthetic scenes, often producing high scores despite structural inconsistencies.

D Novel-view Motion Synthesis

As explained in Section 6.4, we leverage the continuity of our anchor-based embedding representation to interpolate motion embeddings for any viewpoint between two anchors. Given a novel-view image of the source motion, we can then generate a complete video of the source object from this new perspective. In addition to Figure 6 where we demonstrated the source object animated from a novel angle for which we did not obtain ground truth data, in Figure 14 we showcase the ability to generalize to new angles, and new objects at the same time.

D.1 Appearance preservation

In addition to motion quality, we assess appearance preservation using *CLIP-I*[Rahamim et al. 2024; Zhao et al. 2023b], which measures the cosine similarity between the CLIP visual features of the initial image rendered from the static object and each frame in the predicted video. This provides a reference-free, view-invariant signal for how well the object's visual identity is preserved throughout the animation. We also compute *CLIP similarity* between individual frames

Fig. 14. **Qualitative comparison of novel-view motion synthesis.** The top row shows raw supervision, while the bottom row presents results after reconstruction. Simple interpolation and global embeddings struggle to generalize to unseen views, whereas our anchor-based mechanism successfully recovers faithful motion, even for a different target.

of the predicted dynamic 3DGS and reference target videos. This frame-level comparison requires both source and target to perform approximately the same motion and is therefore only applied to the Mini-Mixamo benchmark. As with Motion Fidelity, all CLIP-based metrics are averaged across rendered views. While CLIP is robust to viewpoint variation and sensitive to visual attributes[Goh et al. 2021; Jain et al. 2021], it may overestimate similarity in synthetic scenes, often producing high scores despite structural inconsistencies.

E Additional details on Ablation

E.1 Inversion Ablation

Number of anchors. A key parameter of our method is the number of anchors, K. Here, we analyze its impact on performance metrics during optimization. In Figure 8 of the main paper, each point represents the average accuracy over five random seeds, with shaded areas indicating the 95% confidence intervals. Different colors and markers denote the number of learnable anchors used in training and reconstruction. Figure 8 also reveals a notable behavior: First, for both metrics, increasing the number of anchors generally decreases convergence speed. This phenomenon can be attributed to the information-sharing mechanism among anchors. When fewer anchors are used, each anchor benefits from a broader aggregation of information across multiple source views. However, this trend breaks when employing a single global embedding, which effectively acts as a single anchor. In this case, we observe the slowest convergence rate and lower final performance. This can be explained by the fact that distant views require distinct embeddings, and forcing them into a single representation results in suboptimal optimization. To support this hypothesis, we visualize this effect in Figure 16, where motion observed from opposite viewpoints—where in one view the motion appears to the right and in another to the left—confuses a single global embedding, leading to hallucinations in the reconstructed motion. Based on this analysis, we select $K = 5$ anchors as a practical balance between convergence speed and reconstruction quality.

E.2 4D Consolidation Ablation

The Effect of Regularization on 4D Consolidation Quality. As detailed in Section 4, we introduce several enhancements to the reconstruction stage of our pipeline, including rotational ARAP and LPIPS regularization. Figure 7 in the main paper visually demonstrates the impact of these refinements, showing significant improvements in structural integrity and fine-detail preservation. For example, text originally visible on the target remains legible even during motion (Figure 7, + rotational ARAP column), highlighting the improved stability of our reconstructions—we encourage the reader to zoom in for details. Similarly, our refinements ensure that fine details, such as the flames on the glove, remain clearly visible throughout motion (Figure 7, + LPIPS column), demonstrating the effectiveness of our approach in preserving high-frequency information. Importantly, applying both regularizations together further improves CLIP similarity scores, as shown in Table 3. Despite these clear visual enhancements, motion fidelity **scores** remain largely unchanged. This suggests that our baseline method already preserves motion effectively, but also highlights a limitation of the metric: motion fidelity is primarily sensitive to trajectory alignment rather than reconstruction quality. Even when structural inconsistencies or artifacts are present, high scores can still be achieved as long as sufficient trajectory matches exist. This gap is reflected in the *human preference study* which demonsrtaes clear preference to our regularized variant in both appearance and motion preference (see Fig. 5). Thus, while our refinements significantly enhance visual quality and structural consistency, these improvements are not fully reflected in motion fidelity scores.

SDS Loss. As discussed in Section 4, SDS [Poole et al. 2022] is a widely used strategy for reconstruction with generative models under view inconsistencies, commonly adopted in 4D generation [Bahmani et al. 2024a,b; Jiang et al. 2024; Ling et al. 2024b; Miao et al. 2024; Ren et al. 2023; Uzolas et al. 2024; Wu et al. 2024b; Zeng et al. 2024]. we explored SDS loss as an alternative to LPIPS loss and ARAP regularization for handling multiview inconsistencies. However, applying plain SDS in a single-step UNet inference led to blur in moving object regions (see Fig. 15). This blur caused a mismatch with the sharp 3DGS primitives, prompting the optimization to overfit blurry areas with detailed splats—ultimately collapsing the shape after a few iterations.

In an attempt to address this, we experimented with the Iterative Dataset Update strategy [Haque et al. 2023], which replaces single-step inference with multiple UNet steps per iteration, simulating the full generative process. It generates new supervision videos by injecting noise into renderings and denoising iteratively. However, this still produced similar artifacts, requiring ARAP regularization and LPIPS loss for stability. While we did observe slight improvement in motion quality when combined with our regularization strategy, it substantially increased optimization time (1 hour vs. 5 minutes), so we opted not to include it in our final method. Quantitative comparisons are reported in Table 3.

Method	Motion Fidelity ↑	CLIP Score ↑
Naive	0.7663 ± 0.0098	0.9423 ± 0.0019
+ ARAP Rotation	0.7567 ± 0.0108	0.9608 ± 0.0010
+ LPIPS Loss	0.7602 ± 0.0106	0.9636 ± 0.0010
Naive + Iterative DU	0.7822 ± 0.0384	0.9412 ± 0.0075
ARAP Rotation + LPIPS + Iterative DU	0.7686 ± 0.0272	0.9606 ± 0.0027

Table 3. **Ablation study results.**

Fig. 15. **SDS failure.** In our setup, when trying to use SDS-Loss for the 4D consolidation, the loss causes the Gaussians to collapse. We identify the reason as the blurry single-step denoising prediction generated by SVD. See Sec. E.2 for details.

E.3 Human Preference Study

We include here screenshots from our survey, both from the qualitative comparison, and the ablation study.

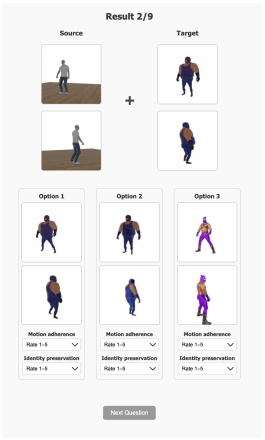

Fig. 17. **Screenshot 1.** A screenshot showing the qualitative comparison survey.

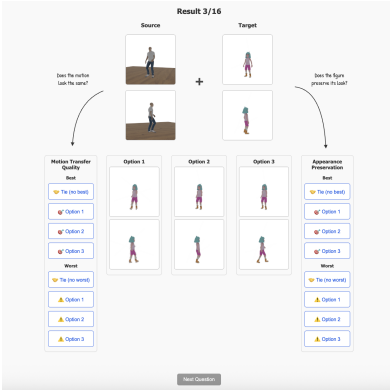

Fig. 18. **Screenshot 2.** A screenshot showing the ablation user study.

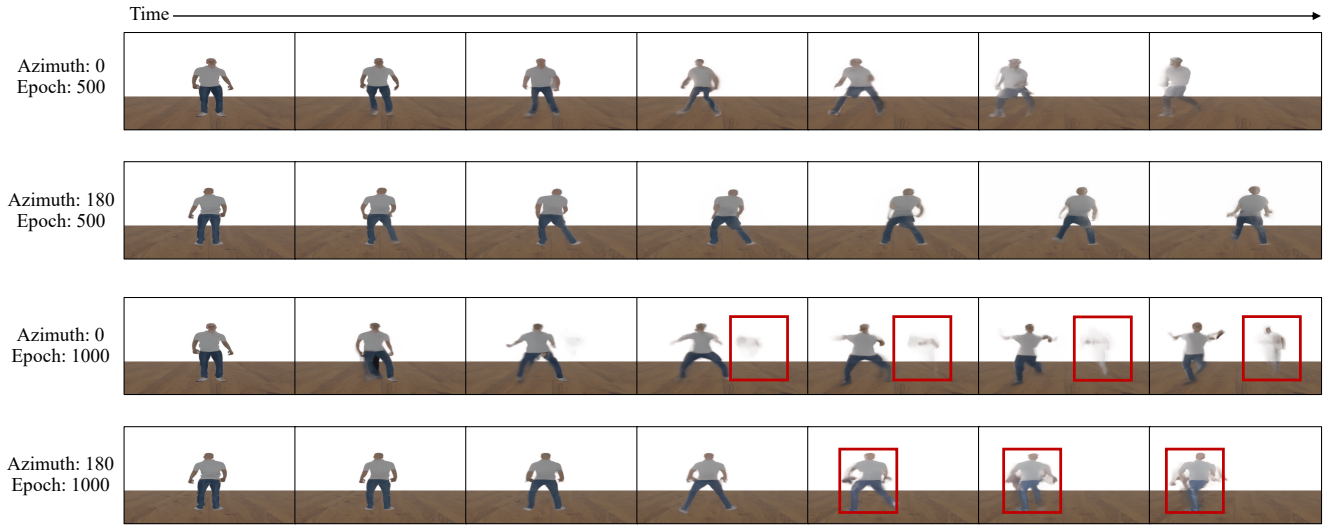

Fig. 16. **Intermediate steps of the optimization process for a single motion embedding.** From an azimuth angle of 0°, the person in the video appears to be moving toward the left side of the screen, whereas from an azimuth of 180°, the 2D motion looks entirely different, with the person appearing to move toward the right. At the beginning of the training process (epoch 500, first two rows), the model starts learning the motion from the 0° viewpoint but struggles with the motion from the 180° viewpoint. By epoch 1000, the model has overfitted to the motion observed from azimuth 0°, fully learning to move the object in the first frame toward the left side of the screen. However, as a consequence, it fails to generalize to the azimuth 180° viewpoint, where the figure should move to the right but instead also moves to the left. Additionally, there is some appearance leakage from the back view, resulting in artifacts in the front view.