CAMO: CATEGORY-AGNOSTIC 3D MOTION TRANSFER FROM MONOCULAR 2D VIDEOS

Anonymous authors
Paper under double-blind review

ABSTRACT

Motion transfer from 2D videos to 3D assets is a challenging problem, due to inherent pose ambiguities and diverse object shapes, often requiring category-specific parametric templates. We propose CAMO, a category-agnostic framework that transfers motion to diverse target meshes directly from monocular 2D videos without relying on predefined templates or explicit 3D supervision. The core of CAMO is a morphology-parameterized articulated 3D Gaussian splatting model combined with dense semantic correspondences to jointly adapt shape and pose through optimization. This approach effectively alleviates shape-pose ambiguities, enabling visually faithful motion transfer for diverse categories. Experimental results demonstrate superior motion accuracy, efficiency, and visual coherence compared to existing methods, significantly advancing motion transfer in varied object categories and casual video scenarios.

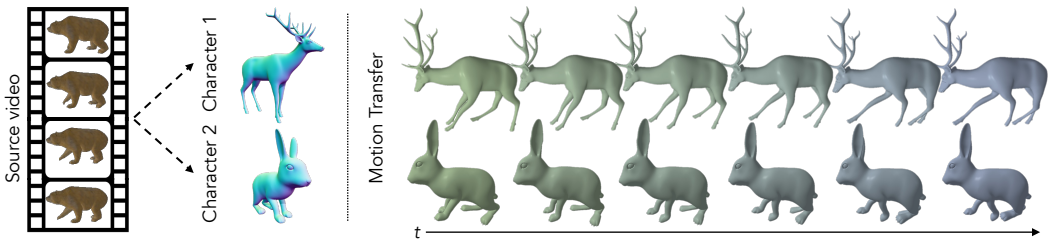

Figure 1: **Conceptual overview of CAMO.** Our method directly transfers articulated motion from 2D video to diverse target objects, without requiring 3D reconstruction of the source or parametric templates.

1 INTRODUCTION

Efficient 3D character animation remains an important goal in both computer graphics research and content industries such as film (Bregler, 2007), interactive media (Rachmavita, 2020), and robotics (Arduengo et al., 2021). Motion transfer techniques (Aberman et al., 2020; Liao et al., 2022) provide an efficient alternative to manual keyframing or marker-based motion capture by enabling the reuse of existing animations across different characters.

However, many existing techniques rely on precomputed 3D sequences, such as articulated skeletons (Aberman et al., 2020) or sparse 3D keypoints (Chen et al., 2023), limiting their applicability in real-world scenarios where such data is scarce or expensive to obtain. Wang et al. (2023) and Muralikrishnan et al. (2024) explored extracting and transferring motion cues from readily accessible 2D monocular videos which reduces this reliance. A common strategy within this domain involves a two-stage, *reconstruct-then-retarget* approach: first, a 3D proxy representation of the source subject (e.g., its pose or shape) is reconstructed from the 2D video, and this intermediate 3D representation is then fed into established 3D-to-3D motion transfer techniques.

Despite demonstrating effective retargeting performance under controlled conditions, these sequential pipelines inherently possess several limitations. A primary limitation stems from their dependence on category-specific priors, such as parametric template models (Loper et al., 2015; Zuffi et al., 2017), which require large-scale, high-fidelity training data. Although models built on such priors (Kanazawa et al., 2018; Zhang et al., 2021; Rueegg et al., 2022) achieve robust and transferable pose estimation within the structural biases of their target domains, their ability to generalize

to diverse shapes and semantic categories remains limited. Furthermore, the cascaded structure of these pipelines can lead to error propagation, where inaccuracies from the reconstruction stage detrimentally impact the fidelity of the final transferred motion.

Our category-agnostic motion transfer framework, **CAMO**, adopts an alternative strategy to conventional reconstruct-then-retarget pipelines. Rather than relying on intermediate 3D reconstructions of the source, we directly project the target character into the 2D observation space, enabling pose optimization purely through image-space supervision. Specifically, we repurpose articulated 3D Gaussian splatting (Yao et al., 2025) (articulated-GS), originally developed for reconstructing articulated animatable objects from 2D videos, to facilitate motion transfer.

CAMO extends this by explicitly modeling morphological differences between source and target characters. Structural variations are decomposed from the target's original shape and adapted to transfer the source motion while preserving topology. To complement this morphology-adaptive optimization and further mitigate shape-pose ambiguity, dense semantic correspondences are established between the 2D source frames and the 3D target mesh, providing semantic guidance for coherent pose recovery. This integration of structural modeling and semantic correspondence guides both visually plausible and semantically coherent pose optimization processes, enabling robust generalization across diverse categories and complex motions. Fig. 1 illustrates the overview of CAMO.

We comprehensively validate CAMO on synthetic benchmarks spanning diverse categories such as humanoids, quadrupeds, and other non-standard animals, as well as on real-world monocular videos. Across all these settings, CAMO consistently preserves motion fidelity and generalizes across diverse morphologies, achieving substantial improvements in both PMD (↓) and FID (↓), with reductions reaching up to 85% on the challenging categories compared to state-of-the-art methods.

2 Related Work

Motion transfer between 3D assets. Traditional techniques in motion transfer have leveraged 3D skeletal structures to enable efficient retargeting across various characters (Gleicher, 1998; Villegas et al., 2018; Aberman et al., 2020; Villegas et al., 2021; Chen et al., 2023). These approaches commonly build upon category-specific skeletal priors, which enable effective performance within their target domains but constrain their generalization to categories outside those domains.

Beyond skeleton-based approaches, skeleton-free deformation methods (Gao et al., 2018; Wang et al., 2020; Liao et al., 2022; Wang et al., 2023; Muralikrishnan et al., 2024; Yoo et al., 2024) are independent from explicit skeletal models, relaxing categorical constraints. Nevertheless, these approaches typically rely on high-quality 3D motion data, which is generally not available for objects across diverse categories. As a result, generalizing these methods to a wider variety of object categories remains a notable challenge, primarily due to the substantial cost and scarcity of such 3D data.

Shape and pose estimation from 2D videos. Another line of research focuses on capturing 3D pose from monocular video. These methods achieve impressive reconstructions within specific domains, often leveraging parametric templates. Representative works include human pose estimation (Zhang et al., 2021; Goel et al., 2023) with SMPL (Loper et al., 2015), and quadruped pose estimaiton (Rüegg et al., 2023; Lyu et al., 2024) with SMAL (Zuffi et al., 2017). Although effective in domains with abundant 3D scan data, these methods are constrained by their reliance on parametric templates, which limits generalization to categories without extensive 3D pose annotations.

Recent approaches (Yao et al., 2022; Wu et al., 2023a;b; Aygun & Mac Aodha, 2024; Li et al., 2024) explore parametric template-free construction of articulated models from image collections. While promising for intra-class generalization without strong parametric template priors, these methods often struggle to generalize across categories. Uzolas et al. (2023) and Yao et al. (2025) inherently avoid this limitation by employing per-scene optimization to directly decompose shape and skeletal pose from individual dynamic scene observations. However, as their focus lies in reconstruction, their ability to retarget motion to novel characters remains underexplored.

2D to 3D motion transfer. Recent 3D-to-3D motion transfer methods (Wang et al., 2023; Muralikrishnan et al., 2024) extend to 2D-to-3D by combining parametric template-based pose and shape estimators (Zhang et al., 2021; Rueegg et al., 2022) with 3D pose transfer techniques. These shape

estimators are typically demonstrated on humanoid or quadruped characters respectively, where the reliance on categorical templates (Loper et al., 2015; Zuffi et al., 2017) fundamentally limits their ability to generalize to novel categories. Moreover, we observe that sequentially combining independently trained components often leads to cumulative errors, ultimately degrading the fidelity of transferred motion.

Maheshwari et al. (2023) propose a category-agnostic approach that removes template priors, transferring motion from RGB-D videos to 3D meshes by estimating skeletal motion from reconstructed meshes; its performance, however, hinges on accurate depth input, limiting robustness in casual or monocular RGB settings. In contrast, Fu et al. (2024) and Zhang et al. (2024a) achieve 2D-to-3D motion transfer without depth by reconstructing motion with neural bones (Yang et al., 2022) or by leveraging image-to-3D generative models. Despite improved generalizability, these approaches remain tied to intermediate reconstruction stages (e.g., pseudo-3D supervision, skeletonization, or physics simulation), which makes them sensitive to reconstruction errors and less robust under large morphological variations.

In contrast, we directly leverage 2D RGB videos as motion sources through morphology-adaptive shape and pose parameter optimization, bypassing intermediate 3D reconstruction. This approach mitigates reconstruction errors and enables robust motion transfer across diverse object categories without category-specific templates.

3 METHODS

Our goal is to transfer articulated motion from a monocular video to arbitrary 3D characters. We take as input a static 3D target mesh $\mathcal{M}^{tgt}$ and a source monocular RGB video with paired foreground masks $\{I_t, M_t\}_{t=0}^{T}$, where I_t is a frame from time t, and M_t is obtained via off-the-shelf segmentation model (Kirillov et al., 2023). We aim to produce a temporally coherent sequence of deformed meshes $\{\mathcal{M}_t^{tgt}\}_{t=0}^{T}$ that faithfully reproduces the source motion.

We first encapsulate the target mesh with an articulated-GS (Yao et al., 2025) representation with pose parameters (Sec. 3.1). We then parameterize morphology using learnable bone lengths, a global scale, and local Gaussian offsets (Sec. 3.2). This representation disentangles shape variation from pose dynamics. Finally, all shape and pose parameters are optimized jointly via differentiable rendering and dense semantic correspondences (Sec. 3.3–3.4), yielding semantically coherent motion aligned to the source. Fig. 2 illustrates the full pipeline.

3.1 ARTICULATED 3D GAUSSIAN SPLATTING

Retargeting pose from monocular 2D observations to 3D assets involves challenges such as occlusion and geometric ambiguity, since full 3D dynamics must be inferred from partial 2D inputs. To address these inherent complexities without resorting to explicit 3D source reconstruction, we pioneers a direct optimization strategy centered on a target-specific, deformable 3D representation.

To this end, we build upon articulated-GS (Yao et al., 2025) to represent the target character, leveraging its framework that supports differentiable rendering and skeletal articulation. This representation models a dynamic object with a skeletal structure and 3D Gaussians, enabling kinematic pose-driven deformation of Gaussians through Linear Blend Skinning (LBS).

The skeleton is represented as a tree $\mathcal{T} = (\mathcal{J}, \mathcal{A})$, where $\mathcal{J}$ is the set of joints and $\mathcal{A} = \{A_j\}_{j \in \mathcal{J} \setminus \{j_r\}}$ assigns each non-root joint j to its parent joint A_j, with j_r denoting the root. A 3D Gaussian G_i is parameterized by its mean $\boldsymbol{\mu}_i \in \mathbb{R}^3$, rotation $\boldsymbol{q}_i \in \mathbb{R}^4$, scale $\boldsymbol{s}_i \in \mathbb{R}^3$, opacity $\sigma_i \in [0, 1]$, and color $\mathcal{SH}_i \in \mathbb{R}^K$ encoded with spherical harmonics, where K is the number of basis functions.

Unlike the previous articulated-GS (Yao et al., 2025), which initializes Gaussian centers from SfM point clouds or random sampling, we leverage target mesh geometry for initialization of Gaussian positions $\boldsymbol{\mu}_i$ (Sec. 3.2). We assume a rigged skeleton is available for skeletal structure; for unrigged meshes, we employ existing automatic rigging methods (Xu et al., 2020; Zhang et al., 2025).

As shown in Fig. 2, we predict the skeletal motion using a temporally conditioned multilayer perceptron (MLP) for each timestamp t. Specifically, the MLP takes a sinusoidal time embedding $\mathrm{emb}(\cdot)$

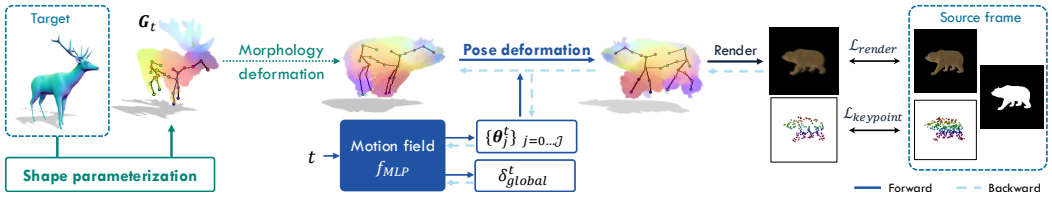

Figure 2: **Morphology-adaptive articulated Gaussian splatting pipeline.** Our framework optimizes the pose and shape of the target mesh directly in 2D observation space using a differentiable Gaussian representation, enabling category-agnostic motion transfer without explicit source reconstruction.

and outputs joint-wise transformations comprising relative rotations and the global translation:

$$\left\{\{\boldsymbol{\theta}_j^t\}_{j\in\mathcal{J}},\ \boldsymbol{\delta}_{\text{global}}^t\right\} = f_{\text{MLP}}(\text{emb}(t))\,, \tag{1}$$

where $\boldsymbol{\theta}_j^t \in \mathbb{R}^4$ is the unit quaternion representing the relative 3D rotation of joint j with respect to its parent in the kinematic hierarchy, and $\boldsymbol{\delta}_{\text{global}}^t \in \mathbb{R}^3$ denotes the global translation of the root joint at time t.

The predicted joint transformations are applied to deform the canonical Gaussian centers using LBS, ensuring visually plausible motion. Specifically, the deformed position $\boldsymbol{\mu}_i^t \in \mathbb{R}^3$ of Gaussian i at time t is computed as a weighted combination of per-joint transformations:

$$\boldsymbol{\mu}_i^t = \boldsymbol{\delta}_{\text{global}}^t + \sum_{j\in\mathcal{J}} w_{ij}\mathbf{T}_j^t\bar{\boldsymbol{\mu}}_i, \quad \mathbf{T}_j^t = \prod_{k\in\text{P}(\text{root},j)} \bar{\mathbf{T}}_k^t\,, \quad \bar{\mathbf{T}}_k^t = \begin{pmatrix} \mathbf{R}_k^t & \mathbf{J}_{A_k} - \mathbf{R}_k^t\mathbf{J}_{A_k} \\ 0 & 1 \end{pmatrix}, \tag{2}$$

where $\bar{\boldsymbol{\mu}}_i$ denotes the canonical center of Gaussian i, and w_{ij} is the skinning weight of joint j on Gaussian i. The transformation $\mathbf{T}_j^t$ is obtained by recursively composing the joint transformations $\bar{\mathbf{T}}_k^t$ along the kinematic path $\text{P}(\text{root}, j)$. Here, $\mathbf{R}_k^t$ denotes the relative joint rotation derived from the predicted quaternion $\boldsymbol{\theta}_k^t$, and $\mathbf{J}_{A_k}$ denotes the transformation of the parent joint.

The posed 3D Gaussians are rasterized onto the image plane. Given a camera viewpoint $\boldsymbol{v}_i$ and pixel location $\boldsymbol{u} \in \mathbb{R}^2$, the pixel color $\mathcal{C}(\boldsymbol{u})$ is computed via alpha compositing with spherical harmonics:

$$\mathcal{C}(\boldsymbol{u}) = \sum_{i\in\mathcal{N}} T_i\,\alpha_i\,\mathcal{SH}(\boldsymbol{sh}_i, \boldsymbol{v}_i), \quad T_i = \prod_{j=1}^{i-1}(1-\alpha_j). \tag{3}$$

Here, $\boldsymbol{sh}_i \in \mathbb{R}^K$ denotes the spherical harmonics coefficients, and the opacity is computed as $\alpha_i = \sigma_i \exp\!\left(-\frac{1}{2}(\mathbf{u}-\hat{\boldsymbol{\mu}}_i)^\top\hat{\boldsymbol{\Sigma}}_i(\mathbf{u}-\hat{\boldsymbol{\mu}}_i)\right)$, where $\hat{\boldsymbol{\mu}}_i \in \mathbb{R}^2$ and $\hat{\boldsymbol{\Sigma}}_i \in \mathbb{R}^{2\times2}$ are the 2D-projected mean and covariance under the camera projection.

3.2 Morphology-Adaptive Shape Parameterization

While the articulated-GS detailed in Sec. 3.1 implements skeletal pose-driven deformation, it is formulated for single-object reconstruction, and thus faces limitations in extracting and transferring motion between source and target with morphological differences. In this paper, we use the term *morphology* to refer to the character's limb proportions, body scale, and local shape details. Parameterization of such variance is a key objective in our framework, enabling the articulated-GS framework to accommodate diverse body shapes while preserving kinematic coherence (Fig. 3 (b)).

Bone length parameterization. We first extend the standard skeleton definition by allowing each bone to have a learnable length parameter. Specifically, let $\mathcal{B}$ denote the set of bones, each defined by a parent-child joint pair. For each bone $b \in \mathcal{B}$, we define its direction vector $\boldsymbol{v}_b \in \mathbb{R}^3$ which denotes the unit vector from parent node to child node of the bone, and a learnable scalar length $\ell_b \in \mathbb{R}^+$. The canonical position of a joint $j \in \mathcal{J}$ in the rest pose is then computed by a linear combination of scaled bone vectors along the kinematic chain:

$$\mathbf{j}_{\text{rest}}(j) = \mathbf{j}_{\text{rest}}(j_{root}) + \sum_{b\in\text{P}(\text{root},j)} \ell_b\boldsymbol{v}_b, \tag{4}$$

where $\text{P}(\text{root}, j)$ denotes the ordered path of bones connecting the root joint to joint j.

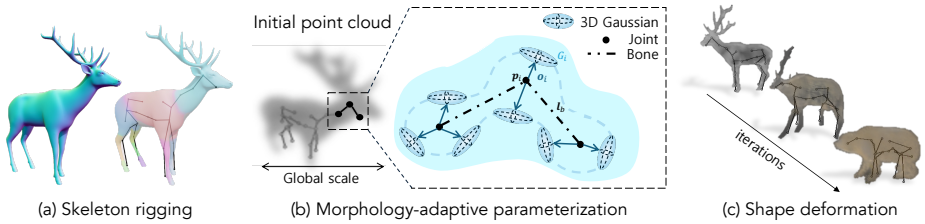

Figure 3: **Deformable morphology parameterization.** **(a)** We initialize the target character with skeleton rigging (Xu et al., 2020), acquiring the topological structure and skinning weights. **(b)** Morphology-adaptive parameterization of structural variations. **(c)** During optimization, shape parameters deform the target's morphological structure to align with the morphology of the source.

Gaussian mean initialization. The deformable surface of our shape parameterization is represented by Gaussian primitives. We parameterize Gaussian means $\boldsymbol{\mu}$ in a morphology-aware manner to determine local shape. Specifically, we parameterize $\boldsymbol{\mu}_i$ of Gaussian $\boldsymbol{G}_i$ relative to a skeleton-blended reference position $\boldsymbol{p}_i$, computed from LBS weights w_{ij} of $\boldsymbol{G}_i$ and rest-pose joint $\boldsymbol{j}_{\text{rest}}(j)$:

$$\boldsymbol{\mu}_i = \boldsymbol{p}_i + \boldsymbol{o}_i, \quad \text{where} \quad \boldsymbol{p}_i = \sum_{j \in \mathcal{J}} w_{ij}\, \boldsymbol{j}_{\text{rest}}(j), \tag{5}$$

where $\boldsymbol{o}_i \in \mathbb{R}^3$ is a learnable vector representing local shape deviation from the skeleton-driven reference. This formulation allows $\boldsymbol{\mu}_i$ to incorporate learned skeletal proportions via $\boldsymbol{p}_i$, while providing a dedicated parameter $\boldsymbol{o}_i$ for fine-grained local geometry.

Global scale. To further account for overall size differences across characters, we introduce a global scaling factor $s_{global} \in \mathbb{R}^+$ that uniformly scales the entire skeleton and associated Gaussian means. This scaling allows the representation to adapt to different character sizes during optimization.

The final canonical position $\bar{\boldsymbol{\mu}}_i$ for each morphology-parameterized Gaussian in the rest pose is obtained by applying this global scale s_{global} to the morphology-aware mean $\boldsymbol{\mu}_i$ (defined in Eq. 5):

$$\bar{\boldsymbol{\mu}}_i = s_{global} \cdot \boldsymbol{\mu}_i. \tag{6}$$

Resulting $\bar{\boldsymbol{\mu}}_i$ encapsulates all morphological parameters and serves as the canonical center in LBS transformation (Eq. 2). Our deformable morphology parameterization enables target character to smoothly conform its appearance to the source cues while preserving original topology (Fig. 3 (c)).

3.3 Target-source dense semantic correspondence

While our proposed shape parameterization accounts for morphological differences, a key challenge in transferring articulated motion from 2D to 3D remains: *shape–pose ambiguity*. This refers to the inherent uncertainty in disentangling an object's underlying pose from its observation. Photometric loss provides essential low-level supervision, but relying on it alone may produce motion artifacts, as it captures only visual cues and lacks explicit semantic correspondences between characters. These artifacts can be mitigated by incorporating additional semantic cues, which help disambiguate overlapping projections, particularly when source and target morphologies differ.

To address this, we establish robust 2D-3D semantic correspondences by leveraging pre-trained vision foundation models. Specifically, we utilize an orientation-sensitive feature extractor (Yang et al., 2020) that produces spatially consistent descriptors across varied poses and morphologies, then obtain dense pixel-to-vertex mappings through semantic feature matching between input images and target mesh renderings (Shtedritski et al., 2024). This provides automatic correspondence estimation without requiring manual registration or additional training.

The detailed pipeline of our dense correspondence extraction module is illustrated in Fig. 4. We first compute the similarity score of the dense semantic features extracted by the feature extractor $\phi(\cdot)$ from a source video frame I_t with those from multiple rendered views $\{I_v^{\text{tgt}}\}$ of the target mesh $\mathcal{M}^{\text{tgt}}$. Then, given a source pixel $\boldsymbol{p} \in I_t$ with the extracted feature $\phi(I_t)$, we compute a pooled similarity score $\Sigma_{I_t}(\boldsymbol{p}, \boldsymbol{x}_k)$ for each vertex $\boldsymbol{x}_k \in \mathcal{M}^{\text{tgt}}$ as:

$$\Sigma_{I_t}(\boldsymbol{p}, \boldsymbol{x}_k) = \underset{v, \boldsymbol{x}_k \in \text{vis}(I_v^{\text{tgt}})}{\text{pool}} S\left(\phi(I_t)[\boldsymbol{p}],\ \phi(I_v^{\text{tgt}})[\pi_v(\boldsymbol{x}_k)]\right), \tag{7}$$

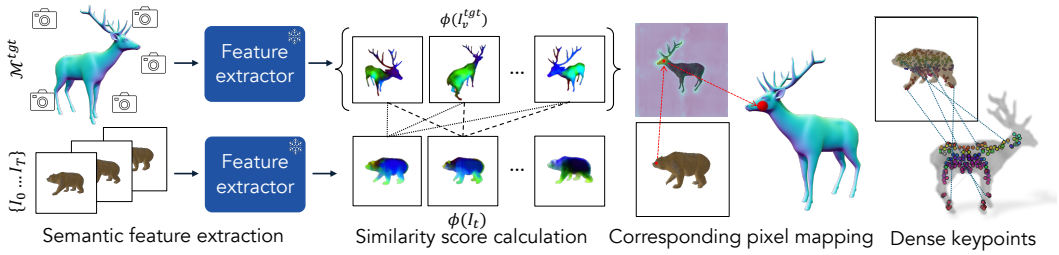

Figure 4: **Dense target-source correspondences matching.** We extract robust 2D-to-3D semantic correspondences by matching semantic features between source frames and rendered target views.

where $S(\cdot)$ denotes a cosine similarity, $\pi_v(\boldsymbol{x}_k)$ denotes the 2D projection of vertex $\boldsymbol{x}_k$ onto the rendered image I_v^{tgt}, and $\phi(I_v^{\text{tgt}})[\pi_v(\boldsymbol{x}_k)]$ is the corresponding feature vector at the 2D projected location. The operator pool aggregates similarity scores via max-pooling across all v target-rendered views where $\boldsymbol{x}_k$ is visible.

The best-matching 3D vertex $\tilde{\boldsymbol{x}}_{\boldsymbol{p},t}^{3D}$ for each pixel $\boldsymbol{p}$ in frame t is obtained by selecting the vertex with the highest pooled similarity score:

$$\tilde{\boldsymbol{x}}_{\boldsymbol{p},t}^{3D} = \arg \max_{\boldsymbol{x}_k \in \mathcal{V}(\mathcal{M}^{\text{tgt}})} \Sigma_{I_t}(\boldsymbol{p}, \boldsymbol{x}_k), \tag{8}$$

where $\mathcal{V}(\mathcal{M}^{\text{tgt}})$ denotes the set of vertices of the target mesh. These retrieved 3D points $\tilde{\boldsymbol{x}}_{\boldsymbol{p},t}^{3D}$ serve as semantic keypoints, providing supervision to guide semantic structure alignment of cross-modality during optimization, as the keypoint loss L_{keypoint} (Sec. 3.4).

3.4 Optimization

As formalized in Eq. 1 and visualized in Fig. 2, our primary objective is to recover the target mesh's time-varying skeletal pose parameters aligned with the source motion, relying solely on 2D observations without ground-truth 3D annotations or any form of pose template prior. The entire framework, composed of morphology-parameterized articulated Gaussians, is optimized end-to-end by minimizing a composite loss function. Our optimization objective combines photometric reconstruction, semantic correspondence, and multiple regularization terms: $\mathcal{L}_{\text{total}} = \lambda_{\text{render}}\mathcal{L}_{\text{render}} + \lambda_{\text{keypoint}}\mathcal{L}_{\text{keypoint}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}$, where the weights balance their respective contributions.

The render loss enforces photometric consistency between the rendered frame $\hat{I}_t$ (from Eq. 3) and the source frame I_t by combining an ℓ_1 term with a SSIM (Wang et al., 2004) term:

$$\mathcal{L}_{\text{render}} = \sum_{t=0}^{T} \Big[(1 - \lambda_{\text{dSSIM}}) \left\| \hat{I}_t - I_t \right\|_1 + \lambda_{\text{dSSIM}} \big(1 - \text{SSIM}(\hat{I}_t, I_t)\big) \Big]. \tag{9}$$

The keypoint loss supervises geometric alignment by minimizing projection error between source image pixels and their matched 3D vertices derived from dense semantic correspondences:

$$\mathcal{L}_{\text{keypoint}} = \sum_{t=0}^{T} \sum_{\boldsymbol{p} \in \mathcal{P}_t} \left\| \boldsymbol{p} - \pi_t \left(\tilde{\boldsymbol{x}}_{\boldsymbol{p},t}^{3D} \right) \right\|_2, \tag{10}$$

where $\tilde{\boldsymbol{x}}_{\boldsymbol{p},t}^{3D}$ is the best-matching 3D vertex obtained via Eq. 8, and $\mathcal{P}_t$ represents sampled foreground pixels. Finally, $\mathcal{L}_{\text{reg}}$ comprises multiple regularization terms that encourage temporal smoothness and geometric consistency (detailed formulations provided in the Appendix).

4 Experiments

4.1 Datasets and Implementations

Datasets. We evaluate our approach on mesh-animation pairs sampled from DeformingThings-4D (DT4D) (Li et al., 2021) and Mixamo (Adobe). From DT4D, we select 20 animation pairs spanning diverse animal categories of quadrupeds and non-quadrupeds exhibiting varied motions.

Table 1: **Quantitative evaluation on Mixamo and DT4D datasets.** Our method consistently outperforms all baselines across diverse categories. Results are averaged across scenes, with per-scene results in the Appendix.

	Mixamo		DT4D-Quadrupeds		DT4D-Others	
	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓
SPT$^+$	0.0029	0.0366	-	-	-	-
NPR$^+$	0.0099	0.0551	0.0032	0.0669	-	-
Transfer4D	0.0084	0.0855	0.0058	0.0505	0.0133	0.0805
Ours	**0.0028**	**0.0304**	**0.0018**	**0.0171**	**0.0023**	**0.0124**

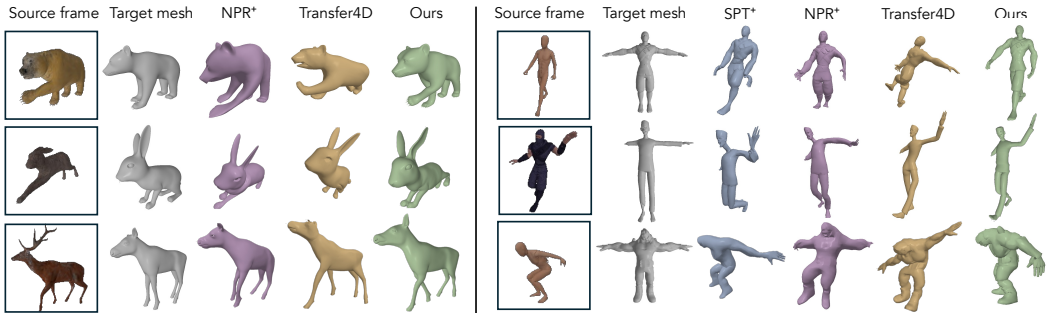

Figure 5: **Qualitative results on Mixamo and DT4D-Quadruped datasets.** Our method shows superior pose alignment compared to baselines across diverse objects. Refer to the supplementary video for full animation.

From Mixamo, we utilize 12 humanoid mesh-animation pairs across different character models and motion types. To simulate a *casually captured* monocular video scenario, we render each source animation using a single camera with constrained movement (±30° angular range), generating input frames with corresponding ground-truth 3D target mesh animations. We further conduct qualitative evaluation on real-world videos sourced from the DAVIS dataset (Perazzi et al., 2016) and two publicly available online videos (Daley, n.d.; Nicky Pe, n.d.), as well as 2D-to-2D motion transfer scenarios using additional synthetic sequences (Pumarola et al., 2021; Liu et al., 2024). Details on dataset preparation and configuration are provided in the Appendix.

Implementation details. We employ a two-stage optimization strategy that first performs global alignment of scale and translation, then jointly refines local pose and shape parameters (bone length, Gaussians) to adapt morphology while preserving essential motion characteristics. All experiments use the Adam optimizer (Kingma & Ba, 2014) with adaptive learning rates over 10k iterations. Our method achieves efficient optimization, completing training in under 10 minutes on a single RTX 4090 GPU. Detailed hyperparameter specifications are provided in the Appendix.

4.2 2D-TO-3D MOTION TRANSFER

Baselines and metrics. We compare our method against two baseline categories: *composite pipelines* combining 2D-to-3D reconstruction with 3D motion transfer, and a template-free optimization-based approach (Transfer4D (Maheshwari et al., 2023)). For composite baselines, we adopt a two-stage setup with mesh reconstruction followed by motion transfer using SPT (Liao et al., 2022) and NPR (Yoo et al., 2024), denoted as SPT$^+$ and NPR$^+$ (see Appendix for baseline implementation details). SPT$^+$ is evaluated only on humanoid motion, as the original method was designed and tested on stylized human characters. Transfer4D performs motion retargeting by extracting skeletal structure from RGB-D input. On datasets with non-quadruped animals, where parametric templates of reconstruction methods are not applicable, we compare only to Transfer4D.

We quantify motion transfer by comparing the retargeted and ground-truth mesh sequences. Consistent with prior work (Liao et al., 2022; Yoo et al., 2024), we adopt Point-wise Mesh Distance (PMD) to measure per-vertex accuracy and Fréchet Inception Distance (FID) (Heusel et al., 2017) to assess perceptual fidelity. To compute FID, both ground-truth and retargeted animations are rendered from 12 viewpoints and their image distributions are compared.

Comparison results. We evaluate our method and baselines on DT4D and Mixamo datasets. As shown in Tab. 1, our approach achieves superior performance on both PMD and FID metrics. These results show that our approach achieves strong performance in a data-efficient manner, relying only

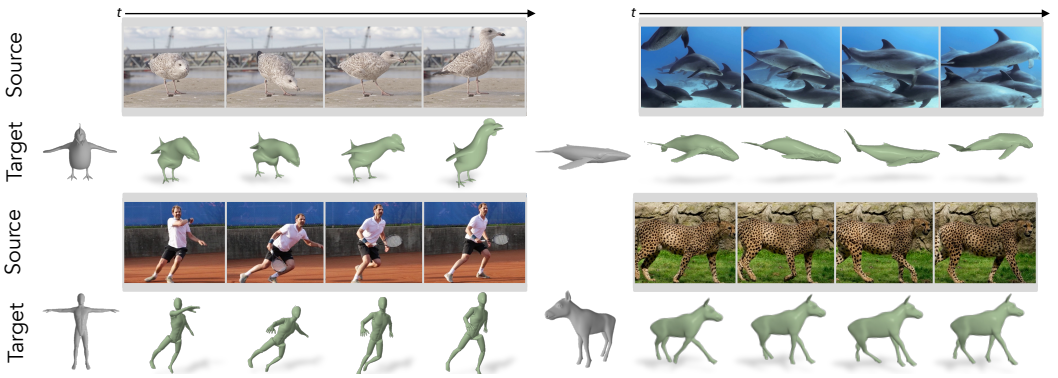

Figure 6: **Qualitative results on real-world datasets.** Our motion transfer pipeline effectively transfers motion from both synthetic and real-world videos in a category-agnostic manner.

on direct optimization without explicit 3D supervision. On non-quadrupeds (DT4D-Others), we significantly outperform Transfer4D even without depth input, demonstrating strong performance beyond parametric model categories.

Fig. 5 demonstrates that our method preserves the target shape and transfers motion faithfully, while baselines often produce distorted shapes by estimating incorrect transformation (Liao et al., 2022; Maheshwari et al., 2023) or relying on predicted surface Jacobians (Yoo et al., 2024). This shape fidelity is attributed to our morphology-parameterization, which we also analyze in Sec. 4.3.

Qualitative results on real-world videos. To evaluate real-world applicability, we apply our method to in-the-wild monocular videos featuring diverse animal categories with complex backgrounds and occlusions. These noisy or open-domain scenarios represent cases where obtaining corresponding 3D animations is challenging. As shown in Fig. 6, our approach successfully transfers motion across these varied scenarios while preserving target mesh structure and proportions. These results demonstrate effective motion transfer directly from monocular input without requiring 3D motion generation, highlighting the practical value of our 2D-grounded motion transfer approach.

4.3 Ablation Study

We ablate key components of our framework in Tab. 2 and Fig. 7. Removing the rendering loss severely degrades performance (PMD $\uparrow \sim 5\times$), indicating it as the primary driver of motion transfer, while the keypoint loss adds complementary semantic guidance. Fig. 7 shows that dropping the keypoint loss yields suboptimal transfers due to unresolved shape-pose ambiguities.

Excluding our shape parameterization (bone lengths l_b, Gaussian means μ, global scale s_{global}) causes distorted geometry and misaligned orientations, especially under large morphological differences. With shape parameters fixed, global translation lowers render loss by pulling the object toward the camera, partially recovering motion but distorting orientation and pose (Fig. 7; see supplementary videos). Overall, adding each component yields consistent gains (Tab. 2), confirming their complementary roles to enhance robustness. Extended ablation studies appear in the Appendix.

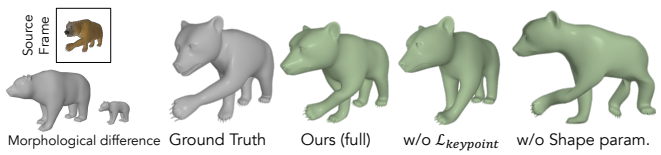

Figure 7: **Qualitative ablations.** Keypoint loss complements motion details and accuracy. Excluding shape parameters induces severe geometric artifacts for large morphological variation.

Table 2: **Quantitative evaluation of component contributions.**

Ablation	PMD (↓)	FID (↓)
Full Model	**0.0018**	**0.0171**
w/o $\mathcal{L}_{\text{render}}$	0.0090	0.0463
w/o Shape param.	0.0047	0.0747
w/o μ update	0.0039	0.0552
w/o l_b & s_{global} update	0.0040	0.0488
w/o $\mathcal{L}_{\text{keypoint}}$	0.0031	0.0252

4.4 Diverse Application Scenarios

Cross-category motion transfer. Our method demonstrates strong generalization across diverse categories, as shown in Fig. 8 (a). We successfully transfer motion between different animal species

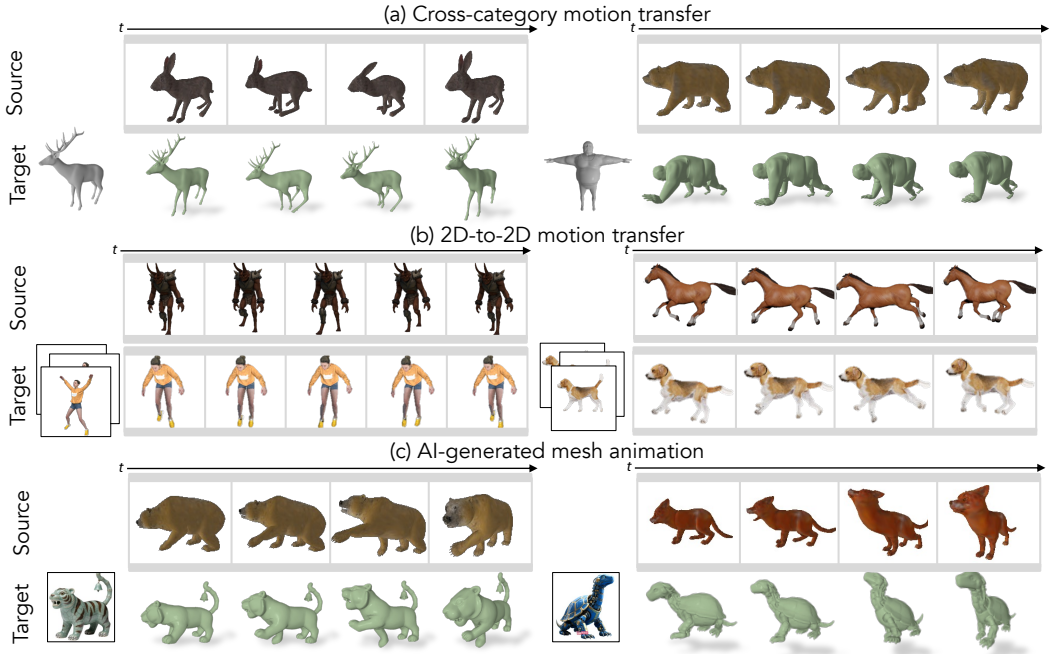

Figure 8: **Results on diverse applications.** Our method transfers motion for (a) cross-category source-target pairs, (b) 2D-to-2D videos, and (c) AI-generated mesh animations.

(rabbit-to-deer) and even across broader categories (animal-to-human). This flexibility stems from our universal optimization approach that does not rely on category-specific skeletal structures or explicit category matching between source and target.

2D-to-2D motion transfer. A key advantage of our method is its representation-agnostic applicability across articulated 3D assets. While primarily demonstrated on mesh targets, our framework seamlessly extends to Gaussian-based 3D representation without modification of core design. Fig. 8(b) shows motion transfer to 3DGS reconstructed from multi-view images (Yao et al., 2025), enabling video-to-video transfer when both source and target originate from RGB sequences. Together, these results yield a single, category-agnostic framework that operates consistently across varied 3D representations.

AI-generated mesh animation. Another interesting application is animating meshes synthesized by generative models. As shown in Fig. 8 (c), we achieve effective motion transfer using meshes generated from an off-the-shelf image-to-mesh model (Zhao et al., 2025). This demonstrates the versatility of our approach to meshes from diverse sources, supporting modern content creation workflows that increasingly incorporate AI-generated assets.

5 Discussion

We introduce **CAMO**, a framework that transfers motion from monocular videos to 3D assets without relying on category-specific templates. By reformulating motion retargeting as an efficient morphology-adaptive optimization on articulated Gaussian splats, our method avoids error accumulation in traditional reconstruct-then-retarget pipelines without any 3D supervision or large datasets. The integration of morphology-adaptive modeling and semantic correspondences provides complementary cues that reduce shape-pose ambiguities and enable broad applicability across different skeletal structures and 3D representations.

Limitations and future work. Our approach inherits common challenges of monocular input, such as occlusion and depth ambiguities with a single character. Future work includes addressing severe occlusion, modeling multi-character or object interactions, adding contact-aware or physics-based constraints, and extending to longer sequences with complex interactions.

REFERENCES

Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. Skeleton-aware networks for deep motion retargeting. *ACM Transactions on Graphics (TOG)*, 39(4):62–1, 2020.

Adobe. Mixamo. `https://www.mixamo.com`.

Miguel Arduengo, Ana Arduengo, Adrià Colomé, Joan Lobo-Prat, and Carme Torras. Human to robot whole-body motion transfer. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pp. 299–305. IEEE, 2021.

Mehmet Aygun and Oisin Mac Aodha. Saor: Single-view articulated object reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10382–10391, 2024.

Chris Bregler. Motion capture technology for entertainment [in the spotlight]. *IEEE Signal Processing Magazine*, 24(6):160–158, 2007.

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

Paul Daley. Close-up video of white seagull. `https://www.pexels.com/video/close-up-video-of-white-seagull-1536290/`, n.d. Pexels video; accessed 2025-09-23.

Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 605–613, 2017.

Zhoujie Fu, Jiacheng Wei, Wenhao Shen, Chaoyue Song, Xiaofeng Yang, Fayao Liu, Xulei Yang, and Guosheng Lin. Sync4d: Video guided controllable dynamics for physics-based 4d generation. *arXiv preprint arXiv:2405.16849*, 2024.

Lin Gao, Jie Yang, Yi-Ling Qiao, Yu-Kun Lai, Paul L Rosin, Weiwei Xu, and Shihong Xia. Automatic unpaired shape deformation transfer. *ACM Transactions on Graphics (ToG)*, 37(6):1–15, 2018.

Michael Gleicher. Retargetting motion to new characters. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pp. 33–42, 1998.

Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14783–14794, 2023.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Yang Li, Hikari Takehara, Takafumi Taketomi, Bo Zheng, and Matthias Nießner. 4dcomplete: Non-rigid motion estimation beyond the observable surface. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12706–12716, 2021.

Zizhang Li, Dor Litvak, Ruining Li, Yunzhi Zhang, Tomas Jakab, Christian Rupprecht, Shangzhe Wu, Andrea Vedaldi, and Jiajun Wu. Learning the 3d fauna of the web. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9752–9762, 2024.

Zhouyingcheng Liao, Jimei Yang, Jun Saito, Gerard Pons-Moll, and Yang Zhou. Skeleton-free pose transfer for stylized 3d characters. In *European Conference on Computer Vision*, pp. 640–656. Springer, 2022.

Isabella Liu, Hao Su, and Xiaolong Wang. Dynamic gaussians mesh: Consistent mesh reconstruction from dynamic scenes. *arXiv preprint arXiv:2404.12379*, 2024.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

Jin Lyu, Tianyi Zhu, Yi Gu, Li Lin, Pujin Cheng, Yebin Liu, Xiaoying Tang, and Liang An. Animer: Animal pose and shape estimation using family aware transformer. *arXiv preprint arXiv:2412.00837*, 2024.

Shubh Maheshwari, Rahul Narain, and Ramya Hebbalaguppe. Transfer4d: A framework for frugal motion capture and deformation transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12836–12846, 2023.

Sanjeev Muralikrishnan, Niladri Dutt, Siddhartha Chaudhuri, Noam Aigerman, Vladimir Kim, Matthew Fisher, and Niloy J Mitra. Temporal residual jacobians for rig-free motion transfer. In *European Conference on Computer Vision*, pp. 93–109. Springer, 2024.

Nicky Pe. A cheetah walking and looking around. `https://www.pexels.com/video/a-cheetah-walking-and-looking-around-8451567/`, n.d. Pexels video; accessed 2025-09-23.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 724–732, 2016.

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10318–10327, 2021.

FP Rachmavita. Interactive media-based video animation and student learning motivation in mathematics. In *Journal of Physics: Conference Series*, volume 1663, pp. 012040. IOP Publishing, 2020.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Nadine Rueegg, Silvia Zuffi, Konrad Schindler, and Michael J Black. Barc: Learning to regress 3d dog shape from images by exploiting breed information. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3876–3884, 2022.

Nadine Rüegg, Shashank Tripathi, Konrad Schindler, Michael J Black, and Silvia Zuffi. Bite: Beyond priors for improved three-d dog pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8867–8876, 2023.

Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. Shic: Shape-image correspondences with no keypoint supervision. In *ECCV*, 2024.

Chaoyue Song, Jianfeng Zhang, Xiu Li, Fan Yang, Yiwen Chen, Zhongcong Xu, Jun Hao Liew, Xiaoyang Guo, Fayao Liu, Jiashi Feng, et al. Magicarticulate: Make your 3d models articulation-ready. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15998–16007, 2025.

Lukas Uzolas, Elmar Eisemann, and Petr Kellnhofer. Template-free articulated neural point clouds for reposable view synthesis. *Advances in Neural Information Processing Systems*, 36:31621–31637, 2023.

Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. Neural kinematic networks for unsupervised motion retargetting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8639–8648, 2018.

Ruben Villegas, Duygu Ceylan, Aaron Hertzmann, Jimei Yang, and Jun Saito. Contact-aware retargeting of skinned motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9720–9729, 2021.

Jiashun Wang, Chao Wen, Yanwei Fu, Haitao Lin, Tianyun Zou, Xiangyang Xue, and Yinda Zhang. Neural pose transfer by spatially adaptive instance normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5831–5839, 2020.

Jiashun Wang, Xueting Li, Sifei Liu, Shalini De Mello, Orazio Gallo, Xiaolong Wang, and Jan Kautz. Zero-shot pose transfer for unrigged stylized 3d characters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8704–8714, 2023.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Shangzhe Wu, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. Dove: Learning deformable 3d objects by watching videos. *International Journal of Computer Vision*, 131(10):2623–2634, 2023a.

Shangzhe Wu, Ruining Li, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. Magicpony: Learning articulated 3d animals in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8792–8802, 2023b.

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

Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. Banmo: Building animatable 3d neural models from many casual videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2863–2873, 2022.

Karren Yang, Bryan Russell, and Justin Salamon. Telling left from right: Learning spatial correspondence of sight and sound. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9932–9941, 2020.

Chun-Han Yao, Wei-Chih Hung, Yuanzhen Li, Michael Rubinstein, Ming-Hsuan Yang, and Varun Jampani. Lassie: Learning articulated shapes from sparse image ensemble via 3d part discovery. *Advances in Neural Information Processing Systems*, 35:15296–15308, 2022.

Yuxin Yao, Zhi Deng, and Junhui Hou. Riggs: Rigging of 3d gaussians for modeling articulated objects in videos. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Seungwoo Yoo, Juil Koo, Kyeongmin Yeo, and Minhyuk Sung. Neural pose representation learning for generating and transferring non-rigid object poses. *arXiv preprint arXiv:2406.09728*, 2024.

Hao Zhang, Di Chang, Fang Li, Mohammad Soleymani, and Narendra Ahuja. Magicpose4d: Crafting articulated models with appearance and motion control. *arXiv preprint arXiv:2405.14017*, 2024a.

Hongkun Zhang, Zherong Pan, Congyi Zhang, Lifeng Zhu, and Xifeng Gao. Texpainter: Generative mesh texturing with multi-view consistency. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024b.

Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11446–11456, 2021.

Jia-Peng Zhang, Cheng-Feng Pu, Meng-Hao Guo, Yan-Pei Cao, and Shi-Min Hu. One model to rig them all: Diverse skeleton rigging with unirig. *arXiv preprint arXiv:2504.12451*, 2025.

Tongjie Y Zhang and Ching Y. Suen. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 27(3):236–239, 1984.

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6365–6373, 2017.

A Technical Appendices and Supplementary Material

This appendix provides additional implementation details, ablations, and extended results supporting the main paper. The overall structure for the Appendices is as follows:

1. **Datasets and baselines** (Sec. A.1)
    - Synthetic datasets
    - Real-world videos
    - Implementation of composite baselines
2. **Implementational details** (Sec. A.2)
    - Skeletal motion field
    - Motion regularizers
    - Training details
3. **Ablation on design choices** (Sec. A.3)
    - Shape parameterization
    - Dense keypoint loss
    - Rigging modules
    - Geometry-aware semantic features
4. **Performance analysis** (Sec. A.4)
    - Performance on diverse morphological differences
    - Performance on different motion scales
    - Performance on extreme cases
    - Failure cases
5. **Extended results** (Sec. A.5)
    - Extended quantitative evaluations
    - Extended qualitative evaluations
6. **Ethical considerations** (Sec. A.6)

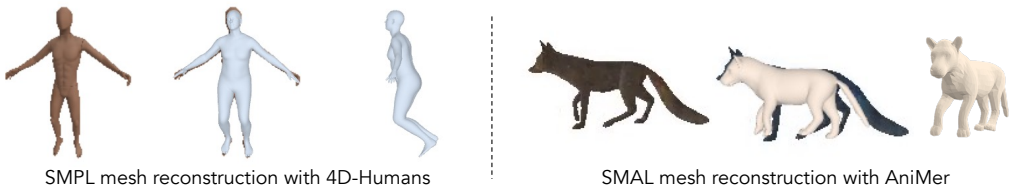

Figure 9: **Intermediate mesh reconstruction with template-based 3D pose and shape estimation models.**

A.1 Datasets and baselines

Synthetic datasets. DeformingThings4D (DT4D) (Li et al., 2021) is a large-scale synthetic dataset of non-rigidly deforming objects, featuring 1,972 animation sequences across 147 characters from 31 categories made by CG experts. We specifically select animal motion sequences (DT4D-animals) for evaluation. We collect 20 pairs of animations, each pair sharing identical pose parameters but differing in character shape. For humanoid characters, we utilize Mixamo (Adobe) to acquire 12 character-motion pairs of discrete motions. Example datasets URL for animal and humanoid datasets are provided in our `index.html`. Code and full datasets will be released for reproducibility.

To generate the monocular video, we render source animations from the DT4D and Mixamo datasets at a resolution of 256×256 using PyTorch3D's `PerspectiveCamera`, ensuring consistent viewpoint changes by varying camera azimuth within ±30°. Untextured DT4D sequences are textured with texture maps generated from TexPainter (Zhang et al., 2024b) to improve visual realism.

Real-world videos. For real-world videos collected from different sources (Daley, n.d.; Nicky Pe, n.d.; Perazzi et al., 2016), we clip and resize the videos at a resolution of 256x256. While synthetic datasets provide ground-truth camera configurations and global orientation alignment between source and target sequences, real-world videos lack such information; thus, we assume a fixed camera for real-world videos. To align the 3D target mesh with the source video's object orientation and scale for motion transfer, we adopt a render-and-compare strategy guided by semantic correspondences. Specifically, we pre-render target mesh with candidate camera poses and evaluate each pose by calculating patch-wise feature cosine similarity to the source frame. The camera pose yielding the maximum similarity serves as our initial alignment, providing a stable and semantically grounded initialization for subsequent optimization.

Implementation of Composite baselines. As described in Sec. 4 in main paper, we compare our method with *composite pipelines* that first reconstruct 3D source meshes from 2D videos, followed by 3D-to-3D motion retargeting. Intermediate reconstructions are obtained by fitting parametric templates to each video frame: SMPL (Loper et al., 2015) for humans and SMAL (Zuffi et al., 2017) for quadrupeds (see Fig.9 for reconstruction examples).

For humanoid motion transfer on the Mixamo dataset, we first extract SMPL meshes using 4D-Humans (Goel et al., 2023), then apply SPT (Liao et al., 2022) and NPR (Yoo et al., 2024) with pretrained checkpoints based on the SMPL model. For quadruped experiments on DT4D, we train NPR's pose extractor and shape applier modules using SMAL meshes reconstructed from monocular videos via AniMer (Lyu et al., 2024).

A.2 Implementational details

Skeletal motion field. The skeletal motion field is parameterized by MLPs. Temporal inputs are first processed using sinusoidal embeddings (13-dimensional), and subsequently passed through a two-layer embedding network producing a 30-dimensional temporal representation. This representation is then fed into an 8-layer MLP featuring 256 hidden units and a skip connection at the fourth layer. The MLP outputs are then directed to a 2-layer global translation head predicting 3D translation vectors, and a 2-layer joint rotation head predicting normalized quaternions. Together, these outputs define SE(3) transformation governing the skeletal motion.

Motion regularizers. Our method employs four distinct motion regularizers to ensure stable and plausible motion. To prevent excessive motions during early training, we apply an L_1 penalty jointly

to global translations and joint rotations:

$$\mathcal{L}_{\text{trans}} = \lambda_{\text{trans}} \frac{\|\boldsymbol{\delta}^t_{\text{global}}\|_1 + \sum_{j=1}^{J} \|r^t_j\|_1}{J}, \tag{11}$$

where λ_{trans} is the regularization weight, $\boldsymbol{\delta}^t_{\text{global}}$ the global translation vector at frame t, r^t_j the rotation angle for joint j at frame t, and J the number of joints.

To enforce temporal smoothness, we additionally penalize frame-to-frame motion:

$$\mathcal{L}_{\text{smooth}} = \lambda_{\text{smooth}} \left(\sum_{j=1}^{J} \left\| r^t_j - r^{t-1}_j \right\|_1 + \left\| \boldsymbol{\delta}^t_{\text{global}} - \boldsymbol{\delta}^{t-1}_{\text{global}} \right\|_1 \right), \tag{12}$$

where λ_{smooth} is the smoothness weight.

Following Yao et al. (Yao et al., 2025), we impose 2D projection constraints on 3D points sampled along the articulated skeleton. First, we extract 2D skeleton points p^t_{skeleton} from the source foreground mask M^t_{src} via a morphological thinning algorithm (Zhang & Suen, 1984). Then, at each frame t, we sample a set of 3D points c^t on the deformed skeleton, project them into the image plane using the camera projection π_t, and penalize their misalignment to the 2D skeleton:

$$\mathcal{L}_{\text{chamf}} = \lambda_{\text{chamf}} \, \text{CD}_{\ell_1} \left(p^t_{\text{skeleton}}, \; \pi_t(c^t) \right), \tag{13}$$

where CD_{ℓ_1} denotes the Chamfer distance (Fan et al., 2017) under the ℓ_1 norm, and λ_{chamf} is a hyperparameter that controls the regularization strength.

To ensure each joint remains within its assigned skinning region, we penalize the mean squared error between the deformed joint positions $\mathbf{j}$ and the centroids of their corresponding Gaussian groups—computed as weighted averages of the Gaussian means μ with normalized skinning weights:

$$\mathcal{L}_{\text{skin}} = \lambda_{\text{skin}} \sum_{j=1}^{J} \Big\| \sum_{i=1}^{N} \tilde{w}^{\top}_{ij} \, \mu_i \; - \; \mathcal{J}_j \Big\|^2, \quad \tilde{w}_{ij} = \frac{w_{ij}}{\sum_{i'=1}^{N} w_{i'j}}, \tag{14}$$

where $w \in \mathbb{R}^{N \times J}$ are the LBS skinning weights, $\mu \in \mathbb{R}^{N \times 3}$ the Gaussian mean positions, $\mathcal{J} \in \mathbb{R}^{J \times 3}$ the joint positions, and λ_{skin} the skinning regularization weight.

Training details. As described in Sec. 3.4 of the main paper, we balance the rendering and keypoint losses with $\lambda_{\text{render}} = 1.0$ and $\lambda_{\text{keypoint}} = 0.001$. The motion regularization weights are set to $\lambda_{\text{transform}} = 0.005$, $\lambda_{\text{smooth}} = 0.001$, $\lambda_{\text{chamf}} = 0.0001$, and $\lambda_{\text{skin}} = 0.1$.

Training follows a two-stage schedule over 10K iterations using Adam optimizer (Kingma & Ba, 2014). The first 500 iterations optimize only global scale, bone length, and global translation for stable initialization. Subsequently, all parameters including shape parameters are jointly optimized with exponential learning rate decay.

We employ differentiated update frequencies based on parameter characteristics. Frame-specific parameters including articulated 3D Gaussians and local motion heads are updated per frame to capture temporal details. Shape parameters such as bone length and global scale, and the global translation are updated every 10 frames to maintain cross-frame consistency and motion smoothness.

A.3 Ablation on design choices

Shape parameterization. Shape parameters are essential for accurately capturing both global and local motion dynamics and ensuring consistent spatial orientation (Sec. 4.3. Inadequate scale regularization causes temporal drift toward the camera, where optimization compensates for scale discrepancies through global translation (shown in the supplementary videos). This compensation disrupts orientation estimation and motion coherence. In contrast, our complete formulation with comprehensive shape parameters preserves geometric consistency and produces stable motion reconstructions.

Dense keypoint loss. As described in Sec.4.3, eliminating dense correspondence guidance leads to misaligned motion cues and misperception of semantic parts. Results illustrated in Fig.10 demonstrate that our dense semantic correspondence effectively encodes object-level semantics, enabling spatially consistent and semantically faithful motion generation.

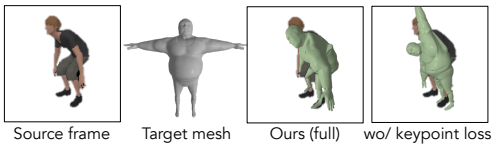

Figure 10: **Qualitative ablation on keypoint loss.** When arms and legs lie nearby in the 2D plane of the source video, the absence of semantic guidance causes overlaps between semantically different body parts, resulting in incorrect motion reconstruction.

Table 3: **Ablation on keypoint confidence thresholding.** PMD and FID decrease slightly with higher thresholds but remain stable across the tested range.

Thr.	PMD ↓	FID ↓
0.0	0.0020	0.0178
0.7	0.0019	0.0179
0.9	0.0018	0.0171

Table 4: **Ablation on number of keypoints.** While 1K points yield slight improvements, performance remains comparable even at sparse points (#50).

#Keyp.	PMD ↓	FID ↓
50	0.0020	0.0191
100	0.0020	0.0185
500	0.0019	0.0179
1000	0.0018	0.0171
1500	0.0020	0.0185

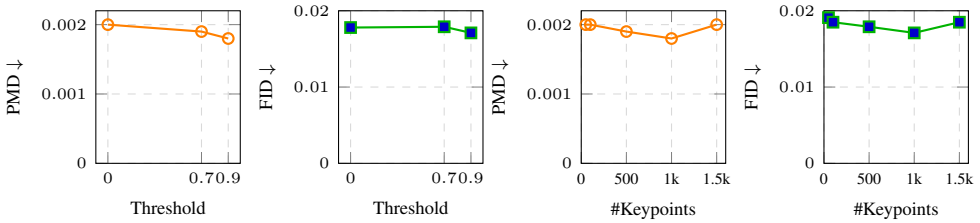

Figure 11: **Ablations on semantic keypoints.** Left: effects of confidence threshold level (PMD, FID). Right: effects of keypoint counts (PMD, FID).

We sample 1K keypoints with confidence above 90% to minimize the effect of outliers. Tab. 3 and Fig. 11 show that performance remains consistent across different confidence thresholds, demonstrating robustness to noisy correspondences. Tab. 4 and Fig. 11 suggest stable behavior of keypoint density effects across guidance densities, with slight improvement on 1K points.

Rigging module ablation. We evaluate the impact of rigging quality on our DT4D dataset, including quadrupeds and non-quadrupeds, by comparing three rigging modules: RigNet (Xu et al., 2020), MagicArticulate (Song et al., 2025), and UniRig (Zhang et al., 2025). Enhanced rigging priors generally improve performance, as shown in Tab. 5 (DT4D-sub). We observe that skinning weight quality significantly affects results. While MagicArticulate and UniRig perform well on the subset, their performance varies on the full dataset, particularly for large motions (Tab. 5, DT4D-all). These results demonstrate the importance of high-quality skinning weights and suggest potential benefits from incorporating adaptive skinning refinement mechanisms.

Geometry-aware semantic features. Distinguishing geometrical differences (e.g., left/right limbs) is crucial for accurate motion transfer. We utilize a pretrained geometry-aware semantic feature extraction module (Yang et al., 2020) for dense correspondence matching. Tab. 6 ablates this design choice, comparing motion transfer performance when using alternative pretrained semantic features from foundation models (Stable Diffusion (Rombach et al., 2022) and DINOv2 (Oquab et al., 2023)) for correspondence matching. Performance significantly drops with SD or DINOv2 features, confirming the effectiveness of geometry-aware features for motion transfer tasks.

A.4 Performance Analysis

Performance across morphological variations. Our dataset encompasses diverse morphological differences between source and target subjects. We quantify these variations using two metrics: (1) *global scale* measured by mesh volume ratio to capture overall size differences, and (2) *shape distance* measured by Chamfer Distance (CD) on normalized meshes to assess geometric variations independent of scale. The dataset spans volume ratios from 1.08× to 16.00× and shape distances from 0.0004 to 0.0023, enabling comprehensive evaluation across morphological diversity.

Regarding global size, we find no correlation between performance and global scale differences. Dividing our dataset into three groups by scale magnitude, low and high groups achieve similar mean

Table 5: **Ablation study on rigging modules.** Performance comparison across different rigging methods on DT4D dataset subsets. DT4D-sub contains scenes with relatively small motions. "CAMO + Best" represents oracle results using the optimal method for each individual scene.

	DT4D-sub		DT4D-all	
Method	PMD ↓	FID ↓	PMD ↓	FID ↓
CAMO + RigNet	0.0018	0.0153	0.0019	0.0159
CAMO + MagicArticulate	0.0016	0.0094	0.0021	0.0117
CAMO + UniRig	0.0015	0.0110	0.0026	0.0168
CAMO + Best	0.0012	0.0086	0.0015	0.0108

Table 6: **Ablation on semantic features.** Our framework achieves best performance with geometry-aware semantic features by distinguishing geometrical relationships between body parts.

Method	PMD ↓	FID ↓
CAMO + Stable Diffusion	0.0028	0.0561
CAMO + DINOv2	0.0025	0.0191
CAMO + Geo-Aware	0.0018	0.0171

Table 7: **Performance on DT4D dataset according to the morphological variations.** Shape differences increase substantially across groups (up to 341% from low to high), yet performance degrades gracefully.

	Shape Diff.	PMD ↓	FID ↓
Low	0.00084 ± 0.00036	0.0012	0.0091
Med	0.00211 ± 0.00016	0.0017	0.0205
High	0.00287 ± 0.00036	0.0026	0.0218

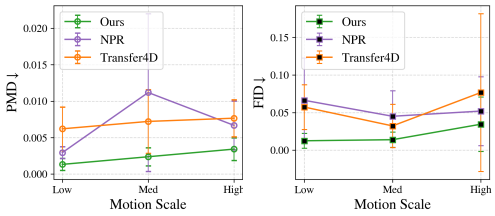

Figure 12: **Performance on DT4D dataset according to the motion scale.** Our method outperforms baselines at all motion levels with minimal degradation on large-motion sequences.

PMD (0.00150 vs 0.00136). Note that this requires the target mesh to initially lie within the camera frustum for valid optimization signals. Regarding shape differences, we categorize source-target pairs into three groups by shape distance. As shown in Tab. 7, our method achieves optimal results with minimal morphological differences while maintaining robust performance under considerable shape variations.

Performance on different motion scales. We define motion magnitude as the maximum average vertex displacement from the first frame across the sequence, computed in normalized coordinate space for cross-mesh comparability. Our dataset spans diverse motion scales (min: 0.03, max: 0.23, avg: 0.11), which we categorized into three distinct groups ranging from small to large motion. Fig. 12 demonstrates consistent performance across motion scales, confirming robustness to motion scale variations. This robustness stems from our time-conditioned MLP jointly optimizing across frames to capture global trajectories and temporal dependencies, while the joint-rotation head provides frame-specific refinements, maintaining global coherence with localized flexibility.

Performance on challenging cases. We evaluate our method on two challenging scenarios that can potentially compromise performance: long video sequences and thin geometric structures.

Our method maintains robust performance on sequences up to 300 frames without degradation. To evaluate longer sequences, we synthetically extended videos by repeating motions. While performance remains stable at 300 frames (PMD: 0.0016), it degradation gently starts from 600 frames (PMD: 0.0021). This limitation stems from our fixed-capacity MLPs. As sequence length increases, mapping sinusoidal time embeddings to motion parameters demands greater representational capacity. When this complexity exceeds the network's capacity, it struggles to capture fine-grained variations.

Challenging structures, such as thin bird wings, present difficulties for both visual guidance and mesh deformation. Their 2D projected regions cover only a few pixels, yielding limited visual cues, while their slender geometry is easily distorted during deformation. As shown in Fig.13, our method robustly addresses these geometrically challenging scenarios within reasonable performance. This robustness is enabled by 2D skeletal projection constraints and temporal smoothness regularization, which jointly enforce motion consistency across frames (Sec.A.1).

Failure case analysis. Despite the robust performance, CAMO exhibits limitations when faced with significant occlusion or ambiguous left-right limb distinction in the source video, leading to

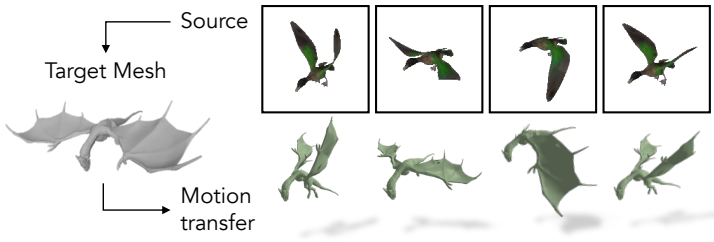

Figure 13: **Qualitative results on challenging case with thin structure.** Our method achieves robust performance on characters with thin structures, which may pose fundamental difficulties in motion transfer.

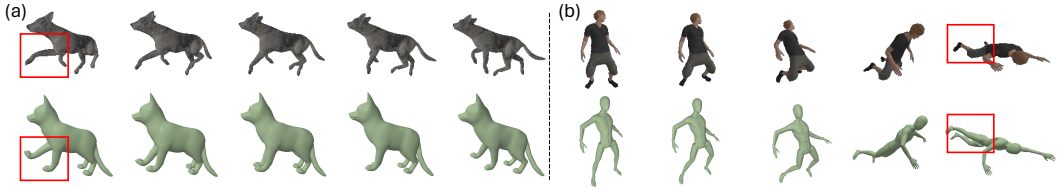

Figure 14: **Failure cases.** Representative failure cases include misperception of geometric semantics leading to left-right confusion (a) and pose estimation errors due to severe occlusion in the source video (b).

less faithful motion transfer. Specifically, Fig. 14 (a) illustrates a failure case where motion quality degrades due to an unclear differentiation of the left and right legs. Fig. 14 (b) highlights performance degradation attributed to extensive self-occlusion, where insufficient visual information hinders accurate motion reconstruction.

A.5 Extended tables and qualitative videos

Quantitative evaluations across all scenes We provide detailed quantitative results for all evaluation scenes from the DT4D (Li et al., 2021) and Mixamo (Adobe) datasets in Tab. 8, Tab. 9, and Tab. 10. These per-scene metrics supplement the averaged results presented in Tab. 1 of the main paper, consistently demonstrating our method's superior performance across diverse motion categories and scenarios.

Qualitative Video Results Qualitative comparisons between our approach and baseline methods are available via `index.html` file, or can be directly accessed in `./static/videos`.

A.6 Ethical considerations

Ethical considerations We acknowledge ethical concerns such as potential misuse for deceptive content and privacy issues when using videos of identifiable individuals. As with any motion synthesis technology, responsible development and deployment practices are important to mitigate risks of generating misleading media.

Applications using human motion data raise privacy and consent considerations, particularly when source videos contain identifiable individuals. While our method transfers motion rather than identity, ethical data collection and privacy protection remain important for deployment. This work is intended for beneficial applications in digital content creation, robotics, and virtual reality, and we advocate for its use in alignment with ethical guidelines and legal frameworks.

Table 8: **Quantitative evaluation across all scenes from DT4D-Quadrupeds.** Lower is better for both PMD and FID (↓). Best and second-best results are highlighted in red and orange, respectively. The complete set of source–motion–target pairs used in this evaluation will be released.

Method	Punch		Walk1		Death		Walk2		KickBack	
	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓
NPR$^+$	0.0027	0.0961	0.0027	0.1535	0.0039	0.0215	0.0010	0.0118	0.0024	0.0245
Transfer4D	0.0032	0.0145	0.0136	0.1395	0.0047	0.0399	0.0019	0.0099	0.0045	0.0383
Ours	0.0012	0.0074	0.0009	0.0029	0.0020	0.0343	0.0003	0.0043	0.0020	0.0211

Method	Swim		Jump		Walk3		Aggression		Howl	
	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓
NPR$^+$	0.0030	0.0676	0.0024	0.0526	0.0022	0.0369	0.0025	0.0489	0.0022	0.0285
Transfer4D	0.0062	0.0837	0.0026	0.0055	0.0078	0.0388	0.0042	0.0075	0.0026	0.0079
Ours	0.0040	0.0385	0.0013	0.0085	0.0005	0.0023	0.0019	0.0335	0.0015	0.0141

Method	Hit Back		Run Stop		Run Forward		Drink		Hop Forward	
	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓
NPR$^+$	0.0013	0.0234	0.0059	0.3740	0.0083	0.0269	0.0043	0.0152	0.0026	0.0227
Transfer4D	0.0014	0.0434	0.0123	0.2292	0.0123	0.0303	0.0065	0.0207	0.0029	0.0485
Ours	0.0008	0.0077	0.0024	0.0377	0.0057	0.0261	0.0016	0.0134	0.0013	0.0050

Table 9: **Quantitative evaluation across all scenes from the Mixamo dataset.** Lower is better for both PMD and FID (↓). Best and second-best results are highlighted in red and orange, respectively. The complete set of source–motion–target pairs used in this evaluation will be released.

Method	JumpingJacks		Running		SideStep		SkinningTest		StandingJump		SwingDance	
	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓
SPT$^+$	0.0016	0.0047	0.0030	0.0069	0.0022	0.0170	0.0036	0.0376	0.0025	0.1816	0.0019	0.0143
NPR$^+$	0.0017	0.0027	0.0287	0.0194	0.0042	0.0308	0.0092	0.0656	0.0084	0.2369	0.0029	0.0167
Transfer4D	0.0077	0.0107	0.0122	0.0450	0.0066	0.0294	0.0086	0.0664	0.0098	0.5631	0.0050	0.0108
Ours	0.0010	0.0035	0.0042	0.0229	0.0013	0.0089	0.0042	0.0195	0.0033	0.1728	0.0018	0.0087

Method	Walking		Floating		HipHopDance		Header		Dying		Snatch	
	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓
SPT$^+$	0.0018	0.0079	0.0074	0.0192	0.0027	0.0047	0.0036	0.0156	0.0025	0.0314	0.0024	0.0982
NPR$^+$	0.0036	0.0057	0.0084	0.0059	0.0032	0.0028	0.0089	0.0177	0.0111	0.0772	0.0282	0.1798
Transfer4D	0.0113	0.0274	0.0043	0.0055	0.0034	0.0037	0.0071	0.0170	0.0072	0.0489	0.0173	0.1981
Ours	0.0011	0.0027	0.0037	0.0029	0.0032	0.0025	0.0026	0.0042	0.0043	0.0416	0.0029	0.0746

Table 10: **Quantitative evaluation across all scenes from the DT4D-others dataset.** Lower is better for both PMD and FID (↓). Best results are highlighted in **bold**. The DT4D-Others dataset contains animals that cannot be reconstructed with parametric templates, including birds, whales, dinosaurs, dragons, and elephants. The complete set of source-motion-target pairs will be released.

Method	Fly		Attack		Running		Walk		Swimming	
	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓	PMD ↓	FID ↓
Transfer4D	0.0283	0.0971	0.0086	0.0409	0.0162	0.0165	0.0119	0.2455	0.0015	**0.0024**
Ours	**0.0045**	**0.0415**	**0.0022**	**0.0122**	**0.0033**	**0.0028**	**0.0006**	**0.0020**	**0.0007**	0.0034