MorphGS: Morphology-Adaptive Articulated 3D Motion Transfer from Videos

Appendix

In this appendix, we provide implementation details (Sec. A), extended ablation studies (Sec. B), an analysis of morphology adaptation (Sec. C), and an identifiability analysis of the proposed morphology parameterization (Sec. D). We further present additional real-world results that extend those in the main paper, along with full per-scene tables and 3D visualizations (Sec. E).

Video Results: Supplementary videos are available via `./supplementary_videos/index.html` or can be directly accessed in `./supplementary_videos/static/videos`.

A Implementation Details

A.1 Training objective and regularization

Following Sec. 3.4 of the main paper, we optimize the following objective at each frame:

$$\mathcal{L}_t = \lambda_{\text{render}}\big(\mathcal{L}_{\text{render}} + \lambda_{\text{mv}}\mathcal{L}_{\text{mv}}\big) + \lambda_{\text{keyp}}\mathcal{L}_{\text{keyp}} + \mathcal{L}_{\text{reg}}, \tag{1}$$

where $\mathcal{L}_{\text{reg}} = \mathcal{L}_{\text{smooth}} + \mathcal{L}_{\text{chamf}} + \mathcal{L}_{\text{trans}} + \mathcal{L}_{\text{arap}}$. We define each term below.

Temporal smoothness To discourage abrupt changes over time, we further impose an $l1$ penalty on frame-to-frame differences in the joint motion parameters $\boldsymbol{\theta}_j^t$ and the root translation $\boldsymbol{\delta}_{\text{root}}^t$:

$$\mathcal{L}_{\text{smooth}} = \lambda_{\text{smooth}}\left(\sum_{j=1}^{J}\left\|\boldsymbol{\theta}_j^t - \boldsymbol{\theta}_j^{t-1}\right\|_1 + \left\|\boldsymbol{\delta}_{\text{root}}^t - \boldsymbol{\delta}_{\text{root}}^{t-1}\right\|_1\right), \tag{2}$$

where λ_{smooth} is the loss weight.

2D Chamfer distance Following Yao et al. [18], we use a 2D Chamfer distance to align the deformed 3D skeleton with the source silhouette structure in the image plane. Specifically, let $\mathcal{S}_t$ denote the 2D skeleton point set extracted from the source foreground mask Ω_t by morphological thinning [21]. We sample 3D points c^t on the deformed skeleton and project them to the image plane with π_t, yielding the projected point set $\hat{\mathcal{S}}_t = \pi_t(c^t)$:

$$\mathcal{L}_{\text{chamf}} = \lambda_{\text{chamf}}\,\text{CD}_{l_1}\big(\mathcal{S}_t, \hat{\mathcal{S}}_t\big), \tag{3}$$

where CD_{l_1} denotes the Chamfer distance [4] under the l_1 norm, and λ_{chamf} is the corresponding loss weight.

Transformation regularization To discourage excessive joint rotation angles and root translations, we impose an ℓ_1 penalty on the per-frame root translation and joint rotation angles. Let $\boldsymbol{\theta}_j^t = \alpha_j^t \hat{\mathbf{u}}_j^t$ denote the axis-angle pose parameter of joint j at frame t, where α_j^t is the rotation angle and $\hat{\mathbf{u}}_j^t$ is the unit rotation axis.

$$\mathcal{L}_{\text{trans}} = \lambda_{\text{trans}} \frac{\|\boldsymbol{\delta}_{\text{root}}^t\|_1 + \sum_{j=1}^{J} |\alpha_j^t|}{J}. \tag{4}$$

Here, λ_{trans} is the loss weight, $\boldsymbol{\delta}_{\text{root}}^t$ is the root translation at frame t, and J is the number of joints.

As-Rigid-As-Possible We use an ARAP loss defined on canonical nearest-neighbor pairs. For each Gaussian k, the neighborhood $\mathcal{N}(k)$ is defined in the canonical pose using the canonical Gaussian centers $\bar{\mathbf{v}}_k$. We then penalize deviations between the canonical distances $\|\bar{\mathbf{v}}_k - \bar{\mathbf{v}}_n\|$ and the posed distances $\|\mathbf{v}_k^t - \mathbf{v}_n^t\|$ for $n \in \mathcal{N}(k)$, where $\mathbf{v}_k^t$ denotes the center of Gaussian k at frame t:

$$\mathcal{L}_{\text{arap}} = \lambda_{\text{arap}} \frac{1}{K} \sum_{k=1}^{K} \frac{1}{|\mathcal{N}(k)|} \sum_{n \in \mathcal{N}(k)} \log\Big(1 + \big|\|\bar{\mathbf{v}}_k - \bar{\mathbf{v}}_n\| - \|\mathbf{v}_k^t - \mathbf{v}_n^t\|\big|\Big), \tag{5}$$

We use the $\log(1 + \cdot)$ form to reduce the contribution of larger discrepancies.

Loss weights For Eq. 1, we use $\lambda_{\text{render}} = 1.0$, $\lambda_{\text{mv}} = 0.25$, and $\lambda_{\text{keyp}} = 0.006$. For motion regularization, we use $\lambda_{\text{trans}} = 0.005$, $\lambda_{\text{smooth}} = 0.01$, $\lambda_{\text{chamf}} = 0.001$, and $\lambda_{\text{arap}} = 1.0$.

A.2 Model and optimization details

Pose network The pose network is an MLP-based motion field following Uzolas et al. [14]. A scalar time input is encoded with a sinusoidal positional embedding and mapped to a 30D latent code by a small time encoder. This code is processed by an 8-layer MLP with 256 hidden units and a skip connection. Two linear heads predict the global 3D translation and per-joint axis-angle rotations. These are converted to joint $SE(3)$ transforms via Rodrigues' formula and skeletal kinematics, and applied using linear blend skinning.

Optimization details We use learning rates of 0.005, 0.01, and 0.001 for the motion network, global scale, and bone lengths, respectively. Bone lengths are optimized from the start of training, with a step decay of 0.1 applied every 1K iterations, starting at iteration 1K. For Gaussian attributes including offsets, we use the default learning-rate settings of 3D Gaussian Splatting [5].

Table A1: Effect of semantic feature extractors. PMD ($\times 10^3$, $\downarrow$) and FVMD ($\times 10^{-3}$, $\downarrow$) for different frozen feature extractors show that Geo-Aware yields the best overall results.

Method	PMD	FVMD
DINOv2 [7]	2.90	9.45
DINOv3 [10]	2.19	9.42
Geo-Aware [20]	**1.91**	**8.82**

Table A2: Effect of correspondence density. PMD ($\times 10^3$, $\downarrow$) and FVMD ($\times 10^{-3}$, $\downarrow$) remain stable as the retained ratio of sampled correspondences decreases.

Ratio (%)	PMD	FVMD
100	**1.91**	**8.82**
50	1.94	9.05
30	1.95	9.14
10	1.96	9.36

Table A3: Analysis on synthesized-view supervision. PMD ($\times 10^3$, $\downarrow$) and FVMD ($\times 10^{-3}$, $\downarrow$) improve consistently as the number of synthesized views increases.

#Synth. views	PMD	FVMD
0 (w/o $\mathcal{L}_{mv}$)	2.96	11.93
1	2.27	11.21
2	2.18	9.41
4	1.91	8.82
8	**1.85**	**7.97**

A.3 Camera and synthesized-view setup

Camera setup In all experiments, we render the posed target from a fixed reference camera chosen from the predefined canonical views of the target mesh, with intrinsics kept fixed throughout optimization. Instead, frame-wise source–target alignment is modeled by a per-frame root transform in $SE(3)$. The same setup is used for both synthetic and real videos regardlessly. For synthesized-view guidance, auxiliary views are rendered by rotating the fixed reference camera around the object center by the prescribed azimuth offset [17], while keeping the intrinsics unchanged.

B Extended Ablation Studies

We extend the ablation studies in Sec. 4.4 of the main paper with additional analysis on design choices. All experiments are conducted on the Mixamo dataset unless otherwise noted.

Geometry-aware semantic features Distinguishing semantically similar but geometrically different parts, such as left and right limbs, is important for accurate correspondence matching. We therefore use a pretrained geometry-aware semantic feature extractor (Geo-Aware) [20]. We compare Geo-Aware, DINOv2 [7], and DINOv3 [10] while keeping the rest of the optimization pipeline fixed. All feature extractors are used frozen. Tab. A1 shows that Geo-Aware achieves the lowest PMD and FVMD among the compared variants, indicating that features trained to resolve left-right ambiguity are better suited to our motion learning setting.

Keypoint density We further vary the density of keypoint supervision by randomly retaining a fraction of the 1K sampled correspondences per frame. As shown in Tab. A2, reducing the retained ratio from 100% to 10% causes only a modest degradation, with PMD increasing from 1.91 to 1.96 (+2.6%) and FVMD

Table A4: Extended monocular ablation analysis. Best baseline indicates SPT$^+$ for Mixamo, NPR$^+$ for DT4D-Quadrupeds, and Pinocchio$^+$ for DT4D-Others.

	Mixamo		DT4D-Quadrupeds		DT4D-Others	
Method	PMD	FVMD	PMD	FVMD	PMD	FVMD
Best baseline*	2.96	12.56	3.28	16.32	7.16	17.52
Ours w/o $\mathcal{L}_{mv}$	2.96	11.93	2.29	11.75	3.16	13.15
Ours	**1.91**	**8.82**	**1.33**	**8.81**	**1.95**	**12.83**

Table A5: Effect of target rig initialization. Results on DT4D-Quadrupeds with different auto-rigging methods used to initialize the target rig. PMD ($\times 10^3$, $\downarrow$) and FVMD ($\times 10^{-3}$, $\downarrow$) vary with the choice of rig initializer.

Method	PMD	FVMD
MorphGS + Pinocchio [2]	1.49	9.79
MorphGS + RigNet [15]	1.33	8.81
MorphGS + MagicArticulate [12]	1.21	8.94
MorphGS + Puppeteer [11]	**1.14**	**8.62**

Table A6: Effect of pose parameterization. Comparison between direct per-frame pose optimization and a time-conditioned pose network. The pose network yields lower PMD ($\times 10^3$), FVMD ($\times 10^{-3}$), and Jitter (km/s^3) than per-frame pose optimization.

Method	PMD	FVMD	Jitter
Per-frame Pose	2.60	12.98	0.15
Pose Network	**1.91**	**8.82**	**0.09**

from 8.82 to 9.36 (+6.1%). This indicates that the sampled correspondences mainly constrain coarse part-level alignment rather than fine-grained geometry. Once this alignment is provided, further increasing the correspondence density has a limited effect on the final result.

Number of synthesized views Tab. A3 reports the effect of varying the number of synthesized views N used in $\mathcal{L}_{\text{mv}}$ (Eq. 13). PMD and FVMD decrease monotonically as N increases from 0 to 8, confirming that additional viewpoints provide complementary supervision, with diminishing returns beyond $N = 4$. We use $N = 4$ as the default, as increasing to $N = 8$ yields only a modest further improvement (PMD: 1.91 $\rightarrow$ 1.85, FVMD: 8.82 $\rightarrow$ 7.97) while increasing the cost of novel-view synthesis. Notably, even without multi-view supervision ($N = 0$), MorphGS achieves PMD of 2.96 and FVMD of 11.93, already matching SPT$^+$ in PMD and outperforming it in FVMD (12.56), indicating that the rendering and keypoint losses alone provide a competitive baseline. We further extend the monocular ($N = 0$) versus multi-view ($N = 4$) comparison to every evaluated synthetic dataset split in Tab. A4, where MorphGS remains competitive with or surpasses the best baseline.

Auto-rigging methods We ablate the target rig initialization used in MorphGS by replacing the off-the-shelf auto-rigging tool [2, 11, 12, 15] while keeping the rest of the optimization unchanged. We report this comparison on DT4D-Quadrupeds [6], focusing on the setting where target rig initialization is required

Table A7: Quantitative evaluation of adapted skeletons on two character pairs. MPJPE is measured in the rest pose. PCK@0.1 and PCK@0.2 use thresholds of 10% and 20% of the characteristic length ($L_{\text{char}} = \sqrt[3]{V_{\text{GT}}}$), respectively.

Method	Mannequin → Ortiz			Mannequin → Pumpkinhulk		
	MPJPE ↓	PCK@0.1 ↑	PCK@0.2 ↑	MPJPE ↓	PCK@0.1 ↑	PCK@0.2 ↑
Pinocchio [2]	0.1013	0.6316	0.8947	0.1280	0.3684	0.8947
Ours	**0.0736**	**0.7368**	**0.9868**	**0.0790**	**0.6316**	**0.9605**

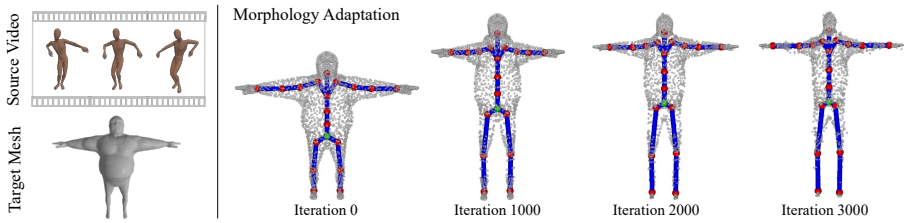

Fig. A1: Visualization of morphology adaptation. During optimization, the global scale, bone lengths, and offset vectors are updated from the target initialization to better match the source subject.

and all compared rigging methods are applicable. As shown in Tab. A5, MorphGS is applicable with multiple auto-rigging methods, while stronger rig initialization leads to better PMD and FVMD.

Pose network vs. per-frame pose optimization Tab. A6 compares direct per-frame pose optimization against our time-conditioned pose network (Sec. 3.3). We additionally report *Jitter* [19], the mean jerk of all body joints in global space (km/s^3), which reflects the temporal smoothness of the recovered motion. The pose network outperforms per-frame optimization across all three metrics (PMD: 1.91 vs. 2.60, FVMD: 8.82 vs. 12.98, Jitter: 0.09 vs. 0.15). The gain is consistent with the MLP's implicit temporal continuity, which encourages motion consistency across adjacent frames and suppresses high-frequency artifacts, as reflected in the lower Jitter score.

C Analysis of Morphology Adaptation

We analyze whether the optimized morphology parameters yield geometrically meaningful skeletal adaptation. Fig. A1 shows morphology adaptation during optimization. The global scale, bone lengths, and offset vectors are updated from the target initialization.

For evaluation, we measure how closely the adapted target skeleton matches the ground-truth skeleton in the rest pose. We compare against Pinocchio [2], a geometry-driven auto-rigging baseline that fits the target skeleton directly

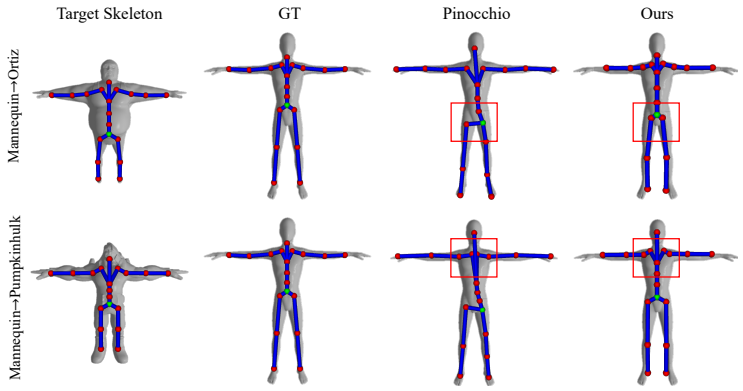

Fig. A2: Qualitative comparison of adapted skeletons in the rest pose. Joint positions and bones are shown as colored spheres overlaid on the target mesh for two character pairs: Mannequin → Ortiz and Mannequin → Pumpkinhulk. Red boxes highlight the region where our adapted skeleton is more closely aligned with the ground-truth structure.

to the source mesh. In contrast, our method adapts the target skeleton solely from the source video, without access to the source 3D geometry. We report normalized Mean Per-Joint Position Error (MPJPE, ↓) and Percentage of Correct Keypoints (PCK, ↑) [9] between the adapted skeleton and the ground-truth skeleton. PCK@0.1 and PCK@0.2 use thresholds of 10% and 20% of the characteristic length $L_{\text{char}} = \sqrt[3]{V_{\text{GT}}}$, where V_{GT} denotes the volume of the ground-truth 3D character mesh. For each character pair, the results are averaged over five motion sequences.

As shown in Tab. A7 and Fig. A2, our method yields lower MPJPE and higher PCK than Pinocchio on both character pairs. This indicates that the adapted skeleton is more closely aligned with the ground-truth skeleton, while not requiring the source mesh.

D Theoretical Analysis of Morphology Parameterization

In this section, we provide an additional theoretical motivation for our morphology parameterization and the staged optimization strategy in Sec. 4.1. By coupling surface geometry to the kinematic chain through a small set of structured parameters, the monocular optimization problem becomes well-constrained: during the early optimization stage where offsets $\{\mathbf{o}_k\}$ remain fixed (optimization stage 1), the skeletal proportions are identifiable up to a global scale under assumptions on the observed motion. We adopt the notation from the main paper (Eq. 3–6).

D.1 Setup

Consider a monocular video $\{I_t\}_{t=1}^{T}$ of an articulated object observed under weak-perspective projection. The object consists of B rigid bones forming a kinematic tree $\mathcal{T}$. We assume:

($\mathcal{A}_1$) **Piecewise rigidity.** Each bone is a rigid body, and its attached surface vertices maintain fixed local-frame positions across all frames. This serves as an idealized approximation of the rig-coupled surface model used in the main method.

($\mathcal{A}_2$) **Non-degenerate motion.** Each bone undergoes sufficiently rich motion across the sequence, including rotations about at least two linearly independent axes, and adjacent bones exhibit non-degenerate relative motion. The motion is temporally continuous, so depth ordering does not change discontinuously across consecutive frames.

($\mathcal{A}_3$) **Weak-perspective projection with fixed camera.** The camera coordinate frame is fixed. The 2D projection of $\mathbf{x} = [x, y, z]^\top$ at frame t is $\boldsymbol{\pi}_t(\mathbf{x}) = s_f^t[x, y]^\top + \mathbf{t}^t$, where $s_f^t > 0$ is a per-frame scale factor and $\mathbf{t}^t \in \mathbb{R}^2$ is a 2D translation.

With the local offsets $\{\mathbf{o}_k\}$ fixed from the target rig, the canonical vertex position is derived from morphology parameters:

$$\bar{\mathbf{v}}_k = s_{\text{global}}\big(\mathbf{p}_k(\{\ell_b\}) + \mathbf{o}_k\big),$$

as defined in Eq. 5–6 of the main paper. The unknowns thus reduce to the global scale s_{global}, bone lengths $\{\ell_b\}$, and per-frame joint rotations $\{\boldsymbol{\theta}_j^t\}$, which in turn determine the per-bone orientations $\{\mathbf{R}_b^t\}$ through forward kinematics (Eq. 7–8).

D.2 Identifiability Statement

Our optimization begins from the target character's bone proportions and adjusts them to best explain the source video observations. Under the idealized assumptions above, the structured morphology parameterization substantially reduces ambiguity compared with unconstrained vertex-wise deformation. The following proposition states that, in this early stage with fixed offsets, the skeletal morphology is generically identifiable up to a single global scale.

Proposition 1 (Identifiability of Skeletal Morphology). *Under $\mathcal{A}_1$–$\mathcal{A}_3$, suppose that each bone has at least four non-coplanar associated surface points, and that these points are observed across sufficiently many frames with non-degenerate motion. Then the morphology parameters $(s_{\text{global}}, \{\ell_b\})$ and the per-frame bone orientations $\{\mathbf{R}_b^t\}$ are determined by the 2D observations up to a single global scale factor. In particular, all bone-length ratios $\ell_b/\ell_{b'}$ are uniquely determined.*

D.3 Proof Sketch of Proposition 1

We now provide a proof sketch of Proposition 1, explaining how the structured morphology parameterization reduces the ambiguity of the monocular reconstruction problem.

Step 1: Per-bone structure recovery up to a similarity ambiguity Under $\mathcal{A}_1$, each bone b is associated with a set of surface points whose relative configuration is fixed over time. For such points, the weak-perspective observation at frame t can be written as

$$\mathbf{u}_k^t = s_f^t \, \mathbf{R}_{b,2\times 3}^t \mathbf{v}_k + \mathbf{t}_b^t, \tag{6}$$

where $s_f^t > 0$ is the per-frame projection scale shared across all bones, $\mathbf{R}_{b,2\times 3}^t$ denotes the first two rows of the bone pose in the camera frame, and $\mathbf{t}_b^t \in \mathbb{R}^2$ collects the projected translation term. Subtracting the per-frame centroid removes translation, yielding:

$$\tilde{\mathbf{u}}_k^t = s_f^t \, \mathbf{R}_{b,2\times 3}^t \tilde{\mathbf{v}}_k. \tag{7}$$

Let

$$\mathbf{S}_b = \begin{bmatrix} \tilde{\mathbf{v}}_1 \cdots \tilde{\mathbf{v}}_{P_b} \end{bmatrix} \in \mathbb{R}^{3\times P_b} \tag{8}$$

collect the centered 3D point coordinates of bone b, and let:

$$\mathbf{M}_b = \begin{bmatrix} s_f^1 \, \mathbf{R}_{b,2\times 3}^1 \\ s_f^2 \, \mathbf{R}_{b,2\times 3}^2 \\ \vdots \\ s_f^T \, \mathbf{R}_{b,2\times 3}^T \end{bmatrix} \in \mathbb{R}^{2T\times 3} \tag{9}$$

stack the scaled projected rotations across frames. Then stacking the centered 2D observations over all frames gives the measurement matrix:

$$\tilde{\mathbf{W}}_b = \mathbf{M}_b \mathbf{S}_b, \tag{10}$$

where $\tilde{\mathbf{W}}_b \in \mathbb{R}^{2T\times P_b}$. By construction, $\mathrm{rank}(\tilde{\mathbf{W}}_b) \leq 3$. Under the stated assumptions, this rank is generically equal to 3. This is because the associated surface points span a three-dimensional local configuration, and the motion is sufficiently rich across frames according to $\mathcal{A}_2$. It then follows from standard scaled-orthographic rigid structure-from-motion factorization arguments [8, 13] that $\tilde{\mathbf{W}}_b$ determines the 3D point configuration of bone b and its per-frame pose sequence up to a bone-specific similarity ambiguity. Concretely, for each bone one may recover a shape basis and motion matrices that are equivalent to the true ones up to an invertible 3×3 transform. After enforcing the metric constraints (i.e., each row pair of $\mathbf{M}_b$ consists of two orthogonal vectors of equal norm s_f^t), the remaining ambiguity reduces to a similarity transform consisting of a scale, a 3D rigid change of coordinates, and possibly a reflection. Temporal continuity helps exclude inconsistent frame-to-frame sign flips, but a per-bone gauge ambiguity still remains at this stage. Therefore, Step 1 recovers each bone only up to an independent similarity class, not yet in a common global frame.

Step 2: Scale and orientation unification via joint constraints Adjacent bones are not independent because neighboring bones share joints. If bones b and b' share a joint j, then their reconstructed motions must be consistent with the same joint trajectory across frames. These shared-joint constraints couple the per-bone similarity ambiguities left unresolved in Step 1. Under non-degenerate relative motion ($\mathcal{A}_2$), they fix both the relative scale and relative orientation between adjacent bones, since an incorrect relative scale or an inconsistent reflection would cause the joint trajectories inferred from the two bones to disagree over time. Applying this argument recursively along the kinematic tree unifies all per-bone scales into a single global scale α and all per-bone reflections into at most a single global reflection. The per-frame projection scale s_f^t is shared across all bones and cancels in these inter-bone constraints. Finally, temporal continuity ($\mathcal{A}_2$) resolves the remaining global reflection, as only one of the two choices produces a depth trajectory that varies continuously over time.

Step 3: Bone length recovery Once the joint trajectories are expressed in a common 3D frame up to the single global scale α, the length of bone b is given by the distance between its parent and child joints. Let $\mathbf{j}^t(j)$ denote the recovered 3D position of joint j at frame t. Then

$$\hat{\ell}_b = \left\| \mathbf{j}^t(j_{\text{child}}) - \mathbf{j}^t(j_{\text{parent}}) \right\|. \tag{11}$$

Because this parent–child distance is invariant for a rigid bone, $\hat{\ell}_b$ is independent of t and satisfies

$$\hat{\ell}_b = \alpha\, \ell_b. \tag{12}$$

Therefore, all bone-length ratios $\ell_b/\ell_{b'}$ are determined, while the absolute lengths remain ambiguous only up to the single global scale absorbed into s_{global}.

Step 4: Bone orientation and twist disambiguation The preceding steps determine the joint trajectories and bone lengths in a common 3D frame up to the single global scale. This is sufficient to fix the bone axis of each segment, but not, in general, the rotation about that axis. In other words, recovering orientation from joint positions alone leaves a residual twist ambiguity. This remaining ambiguity is resolved by the surface observations. Each bone is associated with surface points whose local offsets remain fixed, and these offsets generically include components perpendicular to the bone axis. A twist rotation about the bone axis therefore moves such off-axis points, which in turn changes their projected 2D locations. Consequently, different twist angles produce different image observations. Therefore, under generic non-axisymmetric local surface geometry, the surface observations determine the residual twist angle of each bone. Together with Steps 1–3, this yields the per-frame bone orientations $\{\mathbf{R}_b^t\}$ uniquely, while the morphology parameters $(s_{\text{global}}, \{\ell_b\})$ are determined up to the single global scale already accounted for above. □

D.4 Practical Implications

Effect of structured parameterization. In a naïve formulation, each vertex $\bar{\mathbf{v}}_k \in \mathbb{R}^3$ is optimized independently, allowing individual vertices to drift freely regardless of the underlying skeletal structure. Our parameterization instead decomposes each vertex into a skeleton-dependent anchor $\mathbf{p}_k(\{\ell_b\})$ and a local offset $\mathbf{o}_k$ (Eq. 5–6), coupling vertex positions to the kinematic chain through bone lengths and skinning weights. As a result, adjusting a single bone length coherently displaces all vertices attached to the affected limb, enforcing that shape changes remain structurally consistent across the body. This kinematic coupling is what enables the identifiability result in Proposition 1: in the early stage where offsets are fixed, the skeletal proportions are determined up to a single global scale from monocular observations alone.

Fixing offsets in the early stage. Although our parameterization couples vertex positions to the kinematic chain, a potential ambiguity remains when offsets $\{\mathbf{o}_k\}$ and bone lengths are optimized simultaneously: a change $\ell_b \to \ell_b + \delta\ell_b$ shifts the anchor $\mathbf{p}_k$ by $\Delta\mathbf{p}_k$, which can be compensated by $\mathbf{o}_k \to \mathbf{o}_k - \Delta\mathbf{p}_k$, leaving the canonical vertex position $\mathbf{v}_k = \mathbf{p}_k + \mathbf{o}_k$ largely unchanged. Fixing the offsets during the early optimization stage, while estimating only the skeletal parameters, removes this potential ambiguity and allows bone lengths to be determined primarily from the observed motion. The offsets are released in the later stage once the skeleton is sufficiently constrained, to capture residual shape details beyond what the skeletal proportions alone can express.

Role of offsets under morphology mismatch. The identifiability analysis above relies on idealized assumptions such as piecewise rigidity and weak-perspective projection. In practice, the source and target often differ substantially in local geometry (e.g., limb thickness, torso proportions, or surface details) that skeletal parameters alone cannot account for. The offset parameters $\{\mathbf{o}_k\}$, released in the later optimization stage, provide additional representational capacity to accommodate such local shape discrepancies, reducing the model mismatch between the rig-based representation and the actual source appearance observed in the video.

E Extended Results

Extended real-world results. To further validate real-world applicability, we evaluate MorphGS on 8 additional human clips from SoloDance [16] and 8 animal clips from AiM [22]. For AiM scenes, we use the provided masks, whereas for SoloDance we extract foreground masks using SAM3 [3] with the text query "dancing person". As reported in Tab. A8 and Tab. A9, MorphGS consistently outperforms the baselines across both categories. Fig. A3 further shows qualitatively that MorphGS transfers motion across source–target pairs with substantially different morphologies.

Table A8: Quantitative evaluation on the SoloDance dataset. We use FVMD ($\times 10^{-3}$, $\downarrow$) as the quantitative metric for motion fidelity.

Method	jazz_1	jazz_2	latin_1	latin_2	lyrical_1	lyrical_2	tap_1	tap_2	avg
SPT$^+$	10.58	21.91	12.19	9.29	16.73	7.54	13.66	7.12	12.38
NPR$^+$	11.84	24.41	11.37	11.91	16.57	11.06	13.27	6.20	13.33
Pinocchio$^+$	9.80	24.27	12.73	9.95	17.15	7.70	14.31	6.57	12.81
Ours	**9.12**	**18.15**	**9.26**	**5.05**	**14.63**	**7.53**	**12.38**	**5.90**	**10.25**

Table A9: Quantitative evaluation on the AiM dataset. We use FVMD ($\times 10^{-3}$, $\downarrow$) as the quantitative metric for motion fidelity.

Method	boar	cow	hippo	horse	pig	elephant	tiger	rhino	avg
NPR$^+$	10.72	27.00	13.71	18.01	9.06	15.76	26.80	7.77	16.10
Pinocchio$^+$	11.83	27.50	21.27	14.62	7.20	11.63	20.49	7.63	15.27
Ours	**7.57**	**21.27**	**9.88**	**11.96**	**4.56**	**6.49**	**14.59**	**3.86**	**10.02**

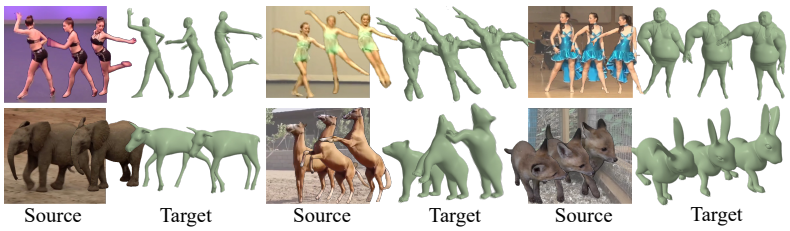

Fig. A3: Qualitative evaluation on extended real-world applicability.

3D coherence and per-scene quantitative results. As MorphGS performs 2D-to-3D motion transfer, its output must hold up as a coherent 3D reconstruction beyond the input view. Fig. A4 shows side-view renderings, where the recovered structure remains coherent under off-axis viewpoints for both synthetic and real-video examples. We provide detailed quantitative results for all evaluation scenes from the Mixamo [1] and DT4D [6] datasets in Tab. A10, Tab. A11, and Tab. A12.

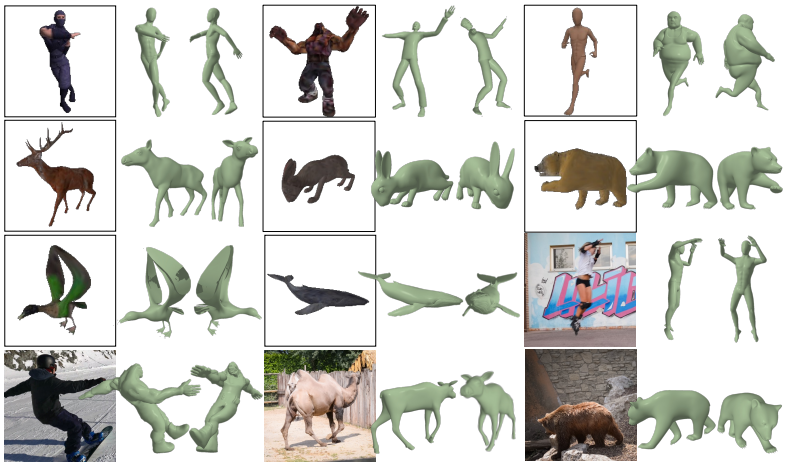

Fig. A4: 3D consistency from off-axis views.

Table A10: Quantitative evaluation across all scenes from the Mixamo dataset. Lower is better for both PMD and FVMD. Best and second-best results are highlighted in red and orange, respectively.

Method	JumpingJacks		Running		SideStep		SkinningTest		StandingJump		SwingDance	
	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD
SPT$^+$	0.79	8.55	2.08	11.27	1.68	6.68	2.48	20.14	4.28	24.15	1.69	7.25
NPR$^+$	0.85	8.63	2.80	16.11	1.62	21.01	6.93	24.00	5.51	32.04	1.71	8.99
Transfer4D	6.14	16.27	5.66	17.92	1.61	25.50	5.56	18.77	9.62	24.12	2.56	11.09
Pinocchio$^+$	15.58	18.46	2.76	10.64	1.16	7.37	2.50	17.85	4.55	26.96	1.61	11.33
Ours	0.39	4.19	1.54	6.54	1.03	6.84	1.71	17.66	5.08	22.55	1.04	5.74

Method	Walking		Floating		HipHopDance		Header		Dying		Snatch	
	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD
SPT$^+$	1.83	6.07	5.08	10.20	1.51	8.25	4.86	7.80	5.50	7.88	3.71	32.46
NPR$^+$	2.97	9.55	4.20	11.35	1.64	10.22	5.44	8.87	8.95	9.24	5.17	39.98
Transfer4D	7.16	12.44	13.89	7.40	2.75	11.57	19.08	6.90	6.38	8.19	11.40	33.70
Pinocchio$^+$	2.32	6.83	2.79	11.10	1.36	7.80	5.20	11.32	3.15	5.05	11.61	33.70
Ours	0.93	4.11	1.97	7.63	0.60	6.82	3.11	6.81	3.08	5.38	2.44	11.52

Table A11: Quantitative evaluation across all scenes from DT4D-Quadrupeds. PMD and FVMD values are reported in 10^3 and 10^{-3} scales, respectively. Lower is better. Best and second-best results are highlighted in red and orange, respectively.

Method	Punch		Walk1		Death		Walk2		KickBack	
	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD
NPR$^+$	2.92	8.48	2.32	23.23	5.31	27.51	1.27	8.64	2.95	20.61
Transfer4D	4.54	10.40	1.96	21.45	5.31	23.21	8.74	12.06	3.56	20.64
Pinocchio$^+$	5.50	9.11	3.20	19.75	5.20	20.40	1.60	9.31	3.70	24.38
Ours	0.77	6.81	0.56	8.62	3.96	14.12	0.60	4.92	1.47	13.70

Method	Swim		Jump		Walk3		Aggression		Howl	
	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD
NPR$^+$	3.56	12.03	2.05	5.65	3.92	18.24	3.51	25.66	3.38	16.15
Transfer4D	5.21	12.31	3.11	8.19	3.39	7.60	4.87	31.32	4.26	7.00
Pinocchio$^+$	4.80	7.83	16.30	6.71	10.30	17.73	6.60	25.63	7.90	14.00
Ours	2.23	7.61	1.66	2.71	0.75	4.94	1.26	18.72	1.05	6.11

Method	Hit Back		Run Stop		Run Forward		Drink		Hop Forward	
	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD
NPR$^+$	2.25	10.02	5.61	12.39	4.74	5.02	2.80	27.67	2.55	23.48
Transfer4D	5.29	13.72	4.22	23.86	8.83	5.24	5.28	17.13	1.44	15.91
Pinocchio$^+$	3.40	17.07	2.90	10.89	4.20	4.28	4.30	20.51	3.30	21.07
Ours	0.88	9.32	1.22	8.36	2.39	2.92	0.65	16.45	0.51	6.80

Table A12: Quantitative evaluation across all scenes from the DT4D-others dataset. Lower is better for both PMD (↓) and FVMD (↓). Best and second-best results are highlighted in red and orange, respectively.

Method	Fly		Attack		Running		Walk		Swimming	
	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD	PMD	FVMD
Transfer4D	13.21	45.51	2.67	21.40	14.08	11.15	3.59	3.80	2.25	5.74
Pinocchio$^+$	2.91	30.63	2.14	11.01	29.99	17.37	0.43	8.97	2.45	3.18
Ours	1.56	36.07	1.81	8.81	5.19	11.26	0.57	3.36	0.62	4.63

References

1. Adobe: Mixamo. https://www.mixamo.com, accessed: 2026-03-04
2. Baran, I., Popović, J.: Automatic rigging and animation of 3d characters. ACM Transactions on graphics (TOG) **26**(3), 72–es (2007)
3. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. arXiv preprint arXiv:2511.16719 (2025)
4. Fan, H., Su, H., Guibas, L.J.: 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)
5. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
6. Li, Y., Takehara, H., Taketomi, T., Zheng, B., Nießner, M.: 4dcomplete: Non-rigid motion estimation beyond the observable surface. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12706–12716 (2021)
7. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
8. Poelman, C.J., Kanade, T.: A paraperspective factorization method for shape and motion recovery. IEEE transactions on pattern analysis and machine intelligence **19**(3), 206–218 (1997)
9. Rhodin, H., Spörri, J., Katircioglu, I., Constantin, V., Meyer, F., Müller, E., Salzmann, M., Fua, P.: Learning monocular 3d human pose estimation from multi-view images. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8437–8446 (2018)
10. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., Massa, F., Haziza, D., Wehrstedt, L., Wang, J., Darcet, T., Moutakanni, T., Sentana, L., Roberts, C., Vedaldi, A., Tolan, J., Brandt, J., Couprie, C., Mairal, J., Jégou, H., Labatut, P., Bojanowski, P.: DINOv3 (2025), https://arxiv.org/abs/2508.10104
11. Song, C., Li, X., Yang, F., Xu, Z., Wei, J., Liu, F., Feng, J., Lin, G., Zhang, J.: Puppeteer: Rig and animate your 3d models. arXiv preprint arXiv:2508.10898 (2025)
12. Song, C., Zhang, J., Li, X., Yang, F., Chen, Y., Xu, Z., Liew, J.H., Guo, X., Liu, F., Feng, J., et al.: Magicarticulate: Make your 3d models articulation-ready. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 15998–16007 (2025)
13. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. International journal of computer vision **9**(2), 137–154 (1992)
14. Uzolas, L., Eisemann, E., Kellnhofer, P.: Template-free articulated neural point clouds for reposable view synthesis. Advances in Neural Information Processing Systems **36**, 31621–31637 (2023)
15. Xu, Z., Zhou, Y., Kalogerakis, E., Landreth, C., Singh, K.: Rignet: Neural rigging for articulated characters. arXiv preprint arXiv:2005.00559 (2020)
16. Yang, Z., Zhu, W., Wu, W., Qian, C., Zhou, Q., Zhou, B., Loy, C.C.: Transmomo: Invariance-driven unsupervised video motion retargeting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5306–5315 (2020)

17. Yao, C.H., Xie, Y., Voleti, V., Jiang, H., Jampani, V.: Sv4d 2.0: Enhancing spatio-temporal consistency in multi-view video diffusion for high-quality 4d generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13248–13258 (2025)
18. Yao, Y., Deng, Z., Hou, J.: Riggs: Rigging of 3d gaussians for modeling articulated objects in videos. In: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)
19. Yi, X., Zhou, Y., Habermann, M., Shimada, S., Golyanik, V., Theobalt, C., Xu, F.: Physical inertial poser (pip): Physics-aware real-time human motion tracking from sparse inertial sensors. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2022)
20. Zhang, J., Herrmann, C., Hur, J., Chen, E., Jampani, V., Sun, D., Yang, M.H.: Telling left from right: Identifying geometry-aware semantic correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3076–3085 (2024)
21. Zhang, T.Y., Suen, C.Y.: A fast parallel algorithm for thinning digital patterns. Communications of the ACM **27**(3), 236–239 (1984)
22. Zhao, B.N., Wu, J., Wu, S.: Web-scale collection of video data for 4d animal reconstruction. Advances in Neural Information Processing Systems **38** (2026)