TOPX: SKELETON-AWARE IN-CONTEXT TEXT-TO-MOTION DIFFUSION ACROSS HETEROGENEOUS TOPOLOGIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-motion generation has mostly been built for one body: the joint count, joint order and pose parameterisation of a single skeleton are baked into the architecture, so every new character means a new model or a retargeting step. Serving hundreds of production rigs with one network is hard for several reasons: their animations share no common representation, the network must be told what each joint of an unfamiliar tree is, a loss summed over bodies of 34 to 102 joints lets the largest dominate, and a rig outside the training library must be served with no clips or a handful. We present TOPX, a text-conditioned motion diffusion transformer that serves 311 heterogeneous animal skeletons with one set of weights, built on three design choices that answer these difficulties. TOPX-17, a per-joint representation we design, converts any rigged animation from its kinematic tree, rest pose and motion alone and standardises every (rig, joint, channel) cell by its own statistics, so that rigs of any proportion share one network input without a template, retargeting or inverse kinematics. The skeleton enters as a prompt—its rest pose leads the sequence as a clean frame and its tree distances bias the attention—so nothing in the model is specific to a skeleton. A grouped flow-matching objective with calibrated channel weights keeps every channel group's share of the gradient independent of the number of cells it spans. Scored by a skeleton-aware text–motion evaluator, TOPX reaches R@1 0.867, where the real clips score 0.904, and an FID of 0.005, and returns per-joint rotations and a world-space root trajectory that the rig plays directly. For rigs outside the library, the same model has both zero-shot and few-shot capability.

1 INTRODUCTION

Text-to-motion generation has made striking progress, but almost all of it on one body. The standard pipeline—a diffusion model, a motion tokeniser or a latent generator trained on HumanML3D (Guo et al., 2022)—assumes a fixed kinematic tree, typically the 22-joint SMPL skeleton, and bakes it into every layer: the input projection is sized for that joint count and any codebook is learned for one morphology. The result is a family of models that animate a human well and cannot animate anything else. The applications that most need generative motion—games, film previsualisation, virtual production—are populated by characters that are not human, and a studio does not have one skeleton but hundreds, each authored by a rigger and each with its own joint count, naming scheme and helper bones.

Why one model for many skeletons is hard. Four things stand in the way. There is no common tensor: rigs differ in joint count, joint order, naming and scale, so cross-skeleton work has either *retargeted* onto a canonical skeleton (Villegas et al., 2018; Aberman et al., 2020; Lee et al., 2023), which degrades as morphology diverges, or fixed a template body plan, which excludes every other body. The network has to be told what each joint of an unfamiliar tree is and how the joints relate, without a parameter per skeleton. A loss summed over a batch of rigs with 34 to 102 joints lets the largest bodies dominate, and training such a model at 0.3B parameters is unstable without care. And a rig outside the training library has to be served with no clips, or a handful, because a model per character multiplies data and compute by the size of the cast. Generation across skeletons has

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

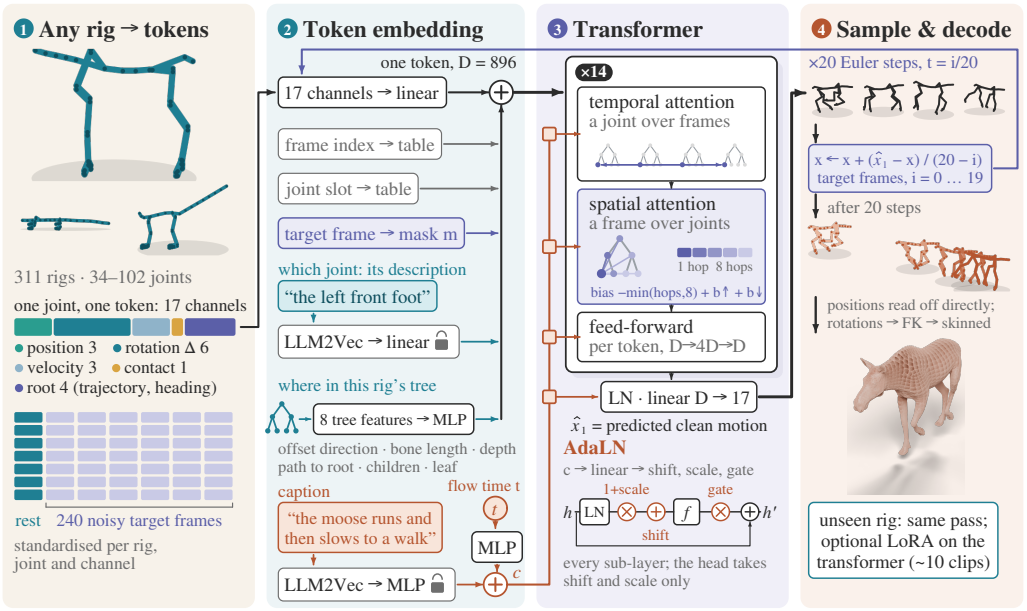

Figure 1: TOPX overview. A rig enters as data: one token per joint and frame with 17 standardised channels, and the sequence is the rig's clean rest pose followed by the noise-mixed target frames. The skeleton enters through the token sum (panel 2) and a tree-distance bias on spatial attention; the caption and the flow time modulate every block through AdaLN. The head predicts the clean motion, the Euler steps update the target frames only, and the rotations are played on the rig. An unseen rig takes the same pass. All skeleton drawings are the model's own output on a moose run-to-walk clip.

been shown without text by AnyTop (Gat et al., 2025) on the Truebones zoo, and with text by OmniZoo (Chen et al., 2025a) on 140 species and by G-MDM (Lee et al., 2025) on Truebones with joint rotations but no root translation.

Our position. A generative model should treat the skeleton as *input*, not as architecture: told the kinematic tree, the rest pose and what each joint is, the network needs no per-skeleton parameter, and adding a character becomes a data question. We study text-conditioned generation of joint rotations and root trajectories across 311 production rigs at 0.3B parameters, on a corpus that is a faithful sample of studio reality: 311 animal rigs with 34 to 102 deformation joints (median 64), 108 distinct kinematic trees and 77,894 captioned clips (6.2M frames).[1] The output is what a rig plays, per-joint rotations and a world-space root trajectory, with no inverse-kinematics fit.

Contributions.

1. **One text-to-motion system for hundreds of production skeletons, at 0.3B parameters** (§4.2). A single network trained on 311 rigs generates text-conditioned motion for every one of them and returns what the rig plays—rotations and a world-space root trajectory, no inverse-kinematics fit. The recipe trains a 303M model stably, and retrieval and FID improve monotonically from 36M to 303M parameters.
2. **TOPX-17, a motion representation designed so that any rigged animation becomes training data** (§3.1). A clip is converted from its kinematic tree, rest pose and motion alone—no template, retargeting, tokeniser or inverse kinematics—and every (rig, joint, channel) cell is standardised by its own statistics, which is what lets rigs of very different proportions share one network: 1.7–2.0 points of R@1 over a per-rig scale.

[1]The clips and their captions follow the AniMo4D annotation set of Wang et al. (2025); the rigs are re-exported with deformation bones only, the joint descriptions are ours, and the conversion to TOPX-17 is new (App. A). The Truebones zoo, a publicly distributed set of 66 animal rigs, is kept entirely outside training.

3. **Skeleton awareness inside the model: a rest-pose prompt and graph-biased attention** (§3.2). The rig's clean rest pose leads the token sequence, and a tree-distance bias with learned ancestor tables lets spatial attention tell parent from child and one limb from another. Nothing in the network is indexed by rig, so a new rig is a new input; the flat padded-joint construction by which fixed-skeleton generators are usually brought to variable skeletons trails this design by 0.19–0.20 of R@1.
4. **High fidelity in the library and zero- and few-shot generalisation outside it** (§4.3). TOPX reaches R@1 0.867 against 0.904 for the real clips and FID 0.005. A rig the model has never seen is served by the same forward pass, and a low-rank adapter on roughly ten clips brings its rotation and position channels into agreement on bodies far from the training distribution (a winged dragon: from 4.46 to 0.30 bone lengths).

2 RELATED WORK

Text-to-motion generation. Human text-to-motion has moved from VAEs on HumanML3D (Guo et al., 2022) to diffusion in pose space (Tevet et al., 2023; Zhang et al., 2024) or in a latent (Chen et al., 2023), later distilled for real time (Dai et al., 2024). Token models over a VQ-VAE followed (Zhang et al., 2023; Guo et al., 2024), scaled by larger corpora (Lin et al., 2023; Fan et al., 2025) and refined by skeleton-aware latent attention (Hong et al., 2025). All fix one kinematic tree; TOPX keeps the pose-space diffusion and text conditioning and removes the fixed skeleton.

Motion across skeleton topologies. Cross-skeleton motion has mostly meant *retargeting* between two known rigs (Villegas et al., 2018; Aberman et al., 2020) or skeleton-agnostic embeddings such as SAME (Lee et al., 2023) and SATA (Zhang et al., 2026); SATA's text-conditioned generation is a downstream demonstration on HumanML3D. For generation on arbitrary topologies, SinMDM (Raab et al., 2024) fits one motion at a time; AnyTop (Gat et al., 2025) generates, without text, across the skeletons of the Truebones zoo (Truebones Motions Animation Studios, 2022); UniMoGen (Khani et al., 2025) is skeleton-agnostic and controlled by style and trajectory. Text-conditioned generation for one body plan has been shown on a fixed 35-joint SMAL template (OmniMotionGPT (Yang et al., 2024)), a shared human–animal topology (X-MoGen (Wang et al., 2026), 115 species), a unified 30-joint skeleton (AniMo (Wang et al., 2025), whose AniMo4D captions our corpus carries) and user-defined human skeletons (OmniSkel (Liu et al., 2025)). Across topologies it has been shown by OmniZoo (Chen et al., 2025a) (140 species), by G-MDM (Lee et al., 2025) on re-captioned Truebones data with Euler joint rotations and no root translation, and, concurrently with this work, by UniMate (Mou et al., 2026). TOPX is text-conditioned; it denoises a per-joint representation to which any rig is converted without a template, retargeting or inverse kinematics; it embeds natural-language *descriptions* of the joints with LLM2Vec; and it normalises every (rig, joint, channel) cell with its own statistics, so that hundreds of rigs of very different proportions share one network. Among earlier systems, none reports the combination of text conditioning, an arbitrary kinematic tree at inference and an output the rig plays as generated; UniMate, concurrent with this work, does (App. F).

Building blocks. TOPX is a DiT-style denoiser (Peebles & Xie, 2023) trained with flow matching (Lipman et al., 2023; Liu et al., 2023), classifier-free guidance on the caption (Ho & Salimans, 2022) and query–key normalisation (Dehghani et al., 2023); its grouped per-channel-family objective follows Kimodo (Rempe et al., 2026). Captions and joint descriptions are embedded with LLM2Vec (BehnamGhader et al., 2024) rather than CLIP (Radford et al., 2021) or T5 (Raffel et al., 2020). New rigs are handled zero-shot through the skeleton prompt, in the spirit of in-context conditioning for image diffusion (Wang et al., 2023), and with a per-rig LoRA (Hu et al., 2022) when they are far from the library's distribution. Retrieval metrics follow the contrastive text–motion evaluators of Guo et al. (2022) and TMR (Petrovich et al., 2023), which are tied to one skeleton; ours is skeleton-aware (§4.1).

3 METHOD

A rig $\mathcal{R} = (\mathcal{T}, \mathbf{p}^{\text{rest}}, \mathcal{D})$ is a kinematic tree $\mathcal{T}$ over J joints, a rest pose $\mathbf{p}^{\text{rest}} \in \mathbb{R}^{J\times 3}$ and a natural-language description $\mathcal{D} = \{d_j\}_{j=1}^{J}$ of its joints. A motion is a tensor $\mathbf{x} \in \mathbb{R}^{L\times J\times 17}$ in the

TopX-17 representation of §3.1. Given $\mathcal{R}$ and a caption c, TopX samples $\mathbf{x} \sim p_\theta(\mathbf{x} \mid \mathcal{R}, c)$ with one transformer shared by every rig (§3.2), trained by a calibrated grouped flow-matching objective (§3.3) and sampled in 20 Euler steps (§3.4). A rig outside the training library is served by the same forward pass and, when a few clips are available, by a per-rig adapter (§4.3); retrieval metrics come from a skeleton-aware evaluator (§4.1).

3.1 TopX-17: a per-joint representation for any rig

Channels. Every joint carries the same 17 channels (Table 6 in App. A). Positions are expressed relative to the root's own ground-plane track, with height absolute, so that the body's pose and its travel are separated the way animators separate them. Rotations are continuous 6D (Zhou et al., 2019) *deltas* from the joint's rest orientation in the global frame: "no motion" is the identity rotation, $[1, 0, 0, 0, 1, 0]$ before normalisation, on every rig, so that the rest frame means the same thing on every rig when it serves as the skeleton prompt (§3.2). Velocities are supervision only and are never integrated at decode time, and the root comes from the trajectory channels. Every frame therefore decodes on its own, without accumulated drift, by either of two routes that do not read each other: the position channels, or forward kinematics from the rotation channels. Their disagreement is the FK–pose gap this paper reports — a property the representation exposes, not a metric imposed on it. Everything is expressed in one canonical frame ($+Y$ up, rest forward $+Z$) obtained by a rigid per-rig re-alignment to the foot support plane, after which forward kinematics is unchanged to numerical precision (App. A). Building TopX-17 for a new rig needs only its kinematic tree, rest pose and animation: there is no template skeleton, no retargeting and no inverse-kinematics fitting, and the same channel definition converts every Truebones rig used in §4.3.

Per-cell normalisation. Rigs differ in size, proportion and convention by orders of magnitude, so every (rig, joint, channel) cell is standardised with its own mean and standard deviation, the latter floored at ϵ_σ (Table 13, App. J), and cells that are constant across a rig's library, such as never-firing contact flags, are removed from supervision (App. A). The statistics are a property of the rig, computed from its motion library and supplied with it at deployment, just as motion-transfer tools require example motions of the target character (Chen et al., 2025b). The library, its split and the conversion are described in §4.1 and App. A.

3.2 In-context skeleton conditioning

Tokens and layout. One token per frame and joint (Figure 1, *any rig → tokens*): the 17 channels of a cell are projected to width D and receive a frame-index and a joint-slot embedding. The sequence is $[\,\mathbf{x}^{\mathrm{demo}} \mid \mathbf{x}_t^{\mathrm{tgt}}\,]$ along time: one clean frame holding the rig's analytic rest pose, never noised, followed by the L target frames (Table 13), which carry the flow interpolant and a learned marker and are the only frames denoised and scored. Under the rest-delta parameterisation the rest frame's rotations are the identity on every rig, so it conveys proportions, joint count and graph rather than motion; after per-cell normalisation it also carries the rig's statistics.

Three signals identify the skeleton. Nothing in the network is indexed by rig; what it knows about a body arrives as input (Figure 1, *token embedding*). *Geometry*: the rest frame above. *Identity*: each joint description d_j is embedded once with frozen LLM2Vec (BehnamGhader et al., 2024) (4096-d), projected and added to every token of joint j. We use descriptions rather than bone names because "front left upper leg" is the same concept on a horse and a lizard whereas `L_Femur_02` and `b_hindleg_l_1` are not, and a new rig's naming scheme need never have been seen. *Place in the tree*: an 8-d structural vector per joint (rest-offset direction, bone length, tree depth, root-path length, child count, leaf flag) passes through a zero-initialised MLP and is added to the same tokens.

The graph enters as an attention bias. Each of the N_b blocks factorises attention over the two token axes (Figure 1, *transformer*): temporal attention, one joint over its frames; spatial attention, one frame over its joints; then a $4\times$ MLP. Only the spatial attention sees the kinematic tree, as an additive per-head bias on the logits between joints i and j, $b_{ij}^{(h)} = -g_{ij} + u_{a_{ij}}^{(h)} + d_{s_{ij}}^{(h)}$, where g_{ij} is the hop distance on the tree, clipped at $g_{\max}$, a_{ij}, s_{ij} are the hops up and down to the lowest common ancestor, clipped at $a_{\max}$ (Table 13), and u, d are zero-initialised per-head tables. Nearer joints attend more by default, and the tables let a head tell parent from child, and a joint on the same

limb from one across the body (App. B). The tree is a soft prior on a fully connected joint attention, not a fixed layer, so one network serves any joint count and topology.

Text and time modulate. The conditioning vector $\mathbf{c} = \mathrm{MLP}(t) + \mathrm{MLP}(\mathbf{e}_c)$, with $\mathbf{e}_c$ the pooled LLM2Vec caption embedding, drives the AdaLN (Peebles & Xie, 2023) shift, scale and gate of every sub-layer (shift and scale only on the output head), so text reaches the model only through modulation and the token stream stays kinematic. The caption is dropped with probability p_{drop} (Table 13) for classifier-free guidance; the rest-pose prompt is never dropped.

Stability. Queries and keys are RMS-normalised per head before the dot product (Dehghani et al., 2023); without it, every attempt at 0.3B diverged (App. K).

3.3 Calibrated grouped flow-matching objective

Flow matching with x-prediction. We train on the straight path $\mathbf{x}_t = (1-t)\,\mathbf{x}_0 + t\,\mathbf{x}_1$ with $\mathbf{x}_0 \sim \mathcal{N}(0, I)$ and $t \sim \mathcal{U}(0,1)$ (Lipman et al., 2023; Liu et al., 2023); the network predicts the clean sample $\hat{\mathbf{x}}_1$, and the velocity-space weight of Li & He (2025) is applied,

$$\ell(\mathbf{x}_t, t) = \frac{w(t)}{\mathbb{E}[w]}\,\mathrm{Huber}_\delta(\hat{\mathbf{x}}_1 - \mathbf{x}_1)\,, \qquad w(t) = \frac{1}{\max(1-t,\,\sigma_{\min})^2}, \tag{1}$$

with $\mathrm{Huber}_\delta(r) = r^2$ for $|r| \le \delta$ and $\delta(2|r| - \delta)$ beyond, applied per cell; δ sits below the 99.99th percentile of the normalised targets (App. C), $\sigma_{\min}$ and δ are given in Table 13, and t is sampled uniformly (App. K).

Grouped loss. A rig with 102 joints has three times the body cells of one with 34 and the same single root row, so a plain mean over cells would drown the root trajectory, heading and contact channels—the ones that decide whether a walk is a walk—in body rotations on large rigs. We partition the channels into nine groups g: position, rotation and velocity, each for the root and for the body, plus contact, root track and heading, and use

$$\mathcal{L} = \sum_g \gamma_g \sqrt{\frac{N_g}{N}}\;\bar{\ell}_g, \tag{2}$$

where $\bar{\ell}_g$ is the masked mean of Eq. (1) over the N_g valid cells of group g in the batch and $N = \sum_g N_g$; the γ_g are set by the calibration below, following the target-share weighting of Kimodo (Rempe et al., 2026). Under this weighting the explicit N_g dependence of the output-gradient energy of group g cancels: it is proportional to $\gamma_g^2 Q_g$, where Q_g is the group's mean weighted squared residual, so a group's weight in $\partial\mathcal{L}/\partial\hat{\mathbf{x}}_1$ no longer grows with the number of cells the group spans in the batch (App. C).

Calibration. The γ_g are not hand-tuned. A target share profile over channel families is fixed in advance, the group energies E_g are measured over one pass of the training set under exactly the training crops and masks, and the weights are solved as $\gamma = \sqrt{\mathrm{share}/E}$, with the rotation family anchored at γ_{rot} (Table 13) and the others solved relative to it. A *mechanism check* then takes a few optimiser-free steps of a small model of the same recipe and verifies that every family's measured share of $\|\partial\mathcal{L}/\partial\hat{\mathbf{x}}_1\|^2$ matches the $\gamma_g^2 Q_g$ prediction (App. C). It certifies the wiring of the weights, not that the profile fixed in advance is realised during training: the realised share moves with the error energies, which evolve.

Auxiliary terms. Four terms are added with weights λ_{FK}, λ_{vel}, λ_{lock} and λ_{acc} (Table 13), none carrying the time weight $w(t)$: a forward-kinematics consistency term between predicted positions and positions forward-kinematised from predicted rotations, a velocity term on world joint displacements, a foot-lock term gated by the contact channel, and an acceleration-matching term (App. C).

3.4 Sampling

Samples follow the straight path with Euler steps $\mathbf{x} \leftarrow \mathbf{x} + (\hat{\mathbf{x}}_1 - \mathbf{x})/(S - i)$ over S steps; invalid cells are projected back to zero after every step, and guidance is applied to the caption only, $\hat{\mathbf{x}}_1 =$

$\hat{\mathbf{x}}_1^{\varnothing} + s\,(\hat{\mathbf{x}}_1^{c} - \hat{\mathbf{x}}_1^{\varnothing})$ with guidance scale s (Table 13). Twenty steps is the operating point used throughout; ten give the same retrieval at half the cost, and App. H gives the latency and quality of both.

4 EXPERIMENTS

4.1 SETUP

Library and splits. The training library is a 311-rig animal corpus in TOPX-17 (77,894 clips, 6.18M frames at 30 fps; deformation bones only, 34–102 joints per rig, median 64, 108 distinct kinematic trees); validation holds 3,899 clips, 5% of every rig's (App. O). Rigs outside the library come from the Truebones zoo (Truebones Motions Animation Studios, 2022): 986 clips over 66 rigs converted to TOPX-17 with the same pipeline (802 captioned), each rig split $80/20$ into train and validation clips. Per-cell statistics are those of all of a rig's clips (App. A); Table 9 uses the adaptation clips alone.

Models and optimisation. The main model has $N_b{=}14$ blocks of width $D{=}896$ with $H{=}14$ heads (302.9M parameters); the 36M pilot ($N_b{=}8$, $D{=}384$, $H{=}6$) shares every other setting except learning rate and length of training, and is used for the ablations. Both train with AdamW—warm-up, cosine decay, gradient clipping and rejection of any step whose pre-clipping gradient norm exceeds a threshold—in bf16; every symbol of §3 has its value in Table 13 (App. J).

Evaluator. Retrieval metrics require a text–motion embedding, and existing ones are tied to one skeleton. We train two towers into a shared embedding space: a frozen DistilBERT text tower with a trainable pooling head and a graph-aware motion tower over the TOPX-17 channels excluding the contact flag, with a symmetric InfoNCE objective (App. O). It sees only real data, never generated samples, shares no weights with the generator, and scores at chance when the captions are shuffled.

Protocol. Generation uses $S{=}20$ Euler steps with guidance scale $s{=}2$ everywhere, at the length of the real clip (at most $L{=}240$ frames); Table 1 is the mean of five sampling seeds and every other table uses seed 42 (the Truebones arms: App. D). Retrieval metrics are text-to-motion R@1/2/3 in pools of 64 formed by chunking the dataset order (a retrieved motion is correct if it is the caption's own clip), the FID on the evaluator's ℓ_2-normalised embeddings and the text–motion matching score, each next to the evaluator's score on the real clips of the same pools. The evaluator scores the sampled channels; re-scored on the played animation (positions and velocities from the forward kinematics of the predicted rotations), R@1 moves by $+0.001$ at 303M and -0.003 at 36M (App. I). Geometry is measured in world space on that deployable output. The *FK–pose gap* is the mean distance, in the rig's mean bone length, between the forward-kinematised and the directly predicted joint positions: a consistency of the two output channels, not a distance to the target. On Truebones rigs, for which no skeleton-aware evaluator exists, articulation, jitter and root speed are reported as ratios to the rig's real clips (App. I).

4.2 ONE MODEL FOR 311 SKELETONS

Table 1 reports text-to-motion retrieval on the validation set. All 311 rigs appear in training, so this measures the generation of held-out clips on known rigs; rigs outside the library are the subject of §4.3. The 303M model reaches R@1 0.867, matching score 0.764, FID 0.005 and Diversity 1.400, where the real clips score 0.904, 0.782 and 1.399 under the same evaluator in the same pools; the 36M pilot reaches R@1 0.840 and FID 0.008. R-precision uses the 3,840 clips of the 60 complete pools; the other columns use all 3,899. One sequence takes 4.1 s at the protocol's 20 steps on one H200 (batch 1, fp32, 240 padded frames) and 0.69 s for the pilot; halving the steps leaves R@1 unchanged (0.863 against 0.864) for 0.003 of FID (0.005 for the pilot), and five steps cost 0.008 of R@1 (Table 2).

4.3 EXTENDING THE MODEL TO RIGS IT HAS NEVER SEEN

Zero training and per-rig LoRA. We take four Truebones rigs that differ from the library in body plan and joint count (Table 3) and compare three arms on the same held-out clips of each: (a)

Table 1: Text-to-motion generation on the validation set (3,899 clips over 311 rigs; 60 pools of 64, 20 steps, s=2; Table 19 gives the other pool sizes). Each model row is the mean over five sampling seeds with the half-width of the 95% interval; the reference row scores the real clips with the same evaluator in the same pools. R-precision is text→motion; Match is the mean text–motion cosine; FID is on the evaluator's normalised embeddings; Diversity (→: nearer the real clips' value) and MModality follow Guo et al. (2022) (App. I).

	R@1↑	R@2↑	R@3↑	FID↓	Match↑	Diversity→	MModality↑
Real clips (reference)	0.904	0.991	0.998	0	0.782	1.399±0.002	–
TopX 36M (pilot)	0.840±0.003	0.957±0.003	0.981±0.003	0.0082±0.0001	0.746±0.001	1.400±0.004	0.247
TopX 303M	0.867±0.003	0.972±0.001	0.990±0.001	0.0050±0.0001	0.764±0.001	1.400±0.002	0.197

Table 2: Sampling cost and quality against the number of Euler steps (s=2, seed 42, validation set; runs at 10 and 5 steps are stamped as protocol variants). Latency: batch 1, one H200, fp32, 240 padded frames, 38 joints.

		303M			36M		
Euler steps	evaluations	latency (ms)	R@1↑	FID↓	latency (ms)	R@1↑	FID↓
20 (protocol)	40	4083	0.864	0.005	688	0.840	0.008
10	20	2041	0.863	0.008	344	0.840	0.013
5	10	1021	0.856	0.013	172	0.822	0.022

the library model with the new rig's per-cell statistics as its only rig-specific input (*zero training*); (b) rank-r adapters (Hu et al., 2022) with scale α on the attention projections, the MLPs and the conditioning paths, trained with the backbone recipe on the rig's 8–26 training clips and folded back into the weights at export (*LoRA*; App. D); (c) a 36M model of the same recipe trained on all 802 captioned Truebones clips, evaluation clips included (*TB-only*). Table 3 gives the FK–pose gap before and after adaptation next to the articulation and jitter of the same sampled output: eight to twenty-six clips bring the gap from 0.57–4.5 bone lengths to 0.23–0.35, with the largest drop on the two rigs farthest from the library, and the TB-only reference sits at 0.08–0.21; the adapted output's articulation and jitter ratios are 0.24–0.77 and 0.20–0.86 (App. E). The jitter excess is gone within 50–100 adapter epochs (50–300 optimiser steps on the 8–26 clips), and one rank-128 adapter pooled over eight flying rigs removes it on all seven scored rigs (median jitter ratio 5.5 to 0.99; Tables 10 and 8). With statistics from the adaptation clips alone, the gap is 0.25–0.36 and the jitter excess is gone (Table 9); Figure 3 shows the adapted model on the four rigs. The released AnyTop weights were trained on the whole Truebones zoo, including these four rigs; App. G reports its per-rig geometry beside ours as an in-distribution reference.

4.4 Ablations

Table 4 removes the ingredients of §3.2 and §3.3 one at a time from the 36M pilot and adds the flat construction by which fixed-skeleton generators are usually brought to variable skeletons (App. L; same recipe and budget, 5% more parameters). The calibrated group weights are worth 0.036–0.043 of R@1; the joint descriptions none on their own; removing the descriptions together with the geodesic and direction biases and the structural features costs a further 0.029–0.033; and flattening the joint axis costs 0.192–0.201 and multiplies the FID by 3.7. A 64-frame motion clip of the same rig in the demonstration slot, in place of the rest frame, is equal in retrieval (R@1 0.822 against 0.821) and worse where the two representations of a pose have to agree (FK–pose gap 0.202 against 0.171 bone lengths), so we keep the rest frame (App. N).

Table 5 measures, on the deployable output, what the representation buys together with the two world-coordinate loss terms written on its channels, which the 13-channel arm trains without (App. K). The 13-channel AnyTop layout derived from the same clips is within a point of R@1 of TopX-17 at every matched epoch, and within 0.008 once its objective and partition are matched (App. K, Table 17): retrieval does not resolve the layout. Geometry does. The two output channels of a TopX-17 sample agree to 0.38 bone lengths against 1.45 at 36M (0.24 at 303M); its leaf joints articulate, whereas the 13-channel layout has no rotation slot for them and the conversion fills it from the parent; its feet slide less while flagged in contact; and its root travels within 4% of the

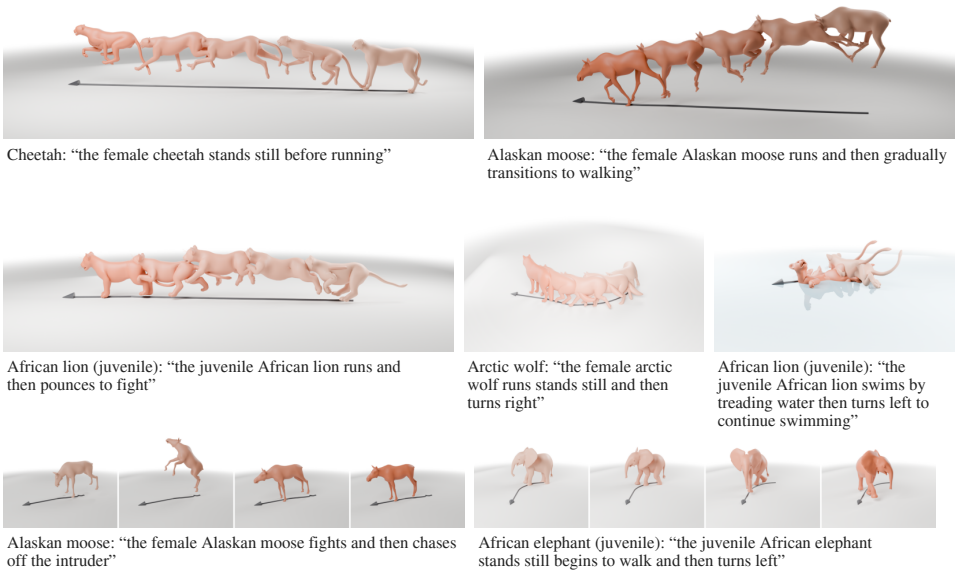

Figure 2: Seven generated clips of five library rigs on their production meshes. Travelling actions: four to five poses at equal distances along the root path, light (first) to dark (last), with the root trajectory on the ground. Actions on the spot: four frames at equal times from a fixed camera. Prompts are the clips’ own captions.

Table 3: Extending TopX (303M) to four Truebones rigs, each supplied with its own per-cell statistics and masks. *zero*: the library model served with those statistics and nothing else; *LoRA*: the per-rig adapter; *TB-only*: a 36M model trained on all Truebones clips including these. All columns come from the same generated sequences: the gap is the FK–pose gap of §4.1; articulation and jitter are of the sampled FK output relative to all real clips of the rig (1 = real). Settings and further arms: App. D, E.

		FK–pose gap, sampled (bl)↓			Articulation ratio			Jitter ratio		
Rig	Train / val	zero	LoRA	TB-only	zero	LoRA	TB-only	zero	LoRA	TB-only
Buffalo (30 j.)	15 / 5	0.797	0.349	0.144	1.45	0.30	0.67	3.32	0.33	1.02
Gazelle (29 j.)	8 / 2	0.574	0.227	0.081	5.39	0.77	1.22	13.2	0.86	1.92
Dragon (95 j.)	10 / 3	4.458	0.304	0.206	1.02	0.49	1.23	2.53	0.52	1.12
Spider (68 j.)	26 / 6	1.896	0.232	0.185	2.87	0.24	0.35	6.49	0.20	0.37

data’s displacement at 36M (10% at 303M), against 22% further for the layout that integrates a root velocity. Per-cell statistics are worth about two points of R@1 and 22–30% of the FID over a per-rig scale (Table 15); at matched optimiser steps the 36M, 116M and 303M models reach R@1 0.821, 0.843 and 0.857 (Table 18).

5 Conclusion

A text-to-motion model can treat the skeleton as an input rather than as part of the architecture: one 303M-parameter model trained on 311 animal rigs generates text-conditioned motion for all of them (R@1 0.867 against 0.904 for the real clips), returning rotations and a root trajectory the rig plays directly; a rig outside the library is served by the same forward pass, and a low-rank adapter on roughly ten clips brings its two output channels into agreement. The library is animal-only.

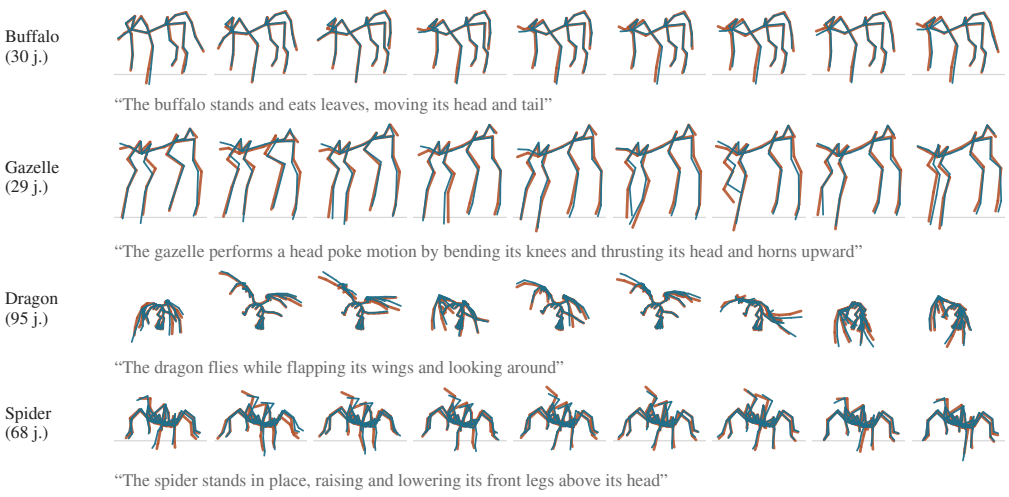

Figure 3: Four rigs outside the library with their per-rig adapters (Table 3): one held-out clip per rig, nine frames spread evenly over the clip, prompt quoted. Each panel draws the sample's two decodings, the directly predicted positions (clay) with the forward kinematics of the predicted rotations (teal) over them; a visible separation is a disagreement of the two. Frames are anchored on the root horizontally and keep their world height, the grey line marking the floor height beneath the root; the flying Dragon is anchored in every axis. Clip selection and the unadapted model: App. M.

Table 4: One ingredient at a time on the 36M pilot; every row shares clips, split, captions, batch, schedule and budget (frozen protocol, seed 42). Last row: the adapted baseline of App. L.

		R@1↑ at epoch			FID↓
Arm	Params	50	75	100	at 100
TopX-17 (full recipe)	36.3M	0.798	0.803	0.808	0.0138
without the joint descriptions	36.3M	0.800	0.806	0.810	0.0124
without the calibrated weights	36.3M	0.760	0.760	0.772	0.0222
without both, and without the graph	36.1M	0.731	0.731	0.739	0.0309
Flat padded joints (App. L)	38.0M	0.597	0.603	0.616	0.0515

Table 5: Geometry of the generated clips against their ground truth (36M arms at the last epoch of Table 4, 303M at the checkpoint of Table 1; bl = mean rest bone length; full table: Table 16). TopX-*17 closer*: share of clips on which the 36M TopX-17 sample is nearer the ground truth than the 13-channel one.

		TopX-17			
Quantity	Data	36M	303M	AnyTop-13	TopX-17 closer
FK–pose gap, sampled (bl)	0.000	0.381	0.236	1.447	100%
Bone-length error, direct pos. (bl)	0.000	0.077	0.052	0.119	91%
Leaf articulation, mean dev. from rest (deg)	11.6	10.6	11.3	0.0	92%
Foot slide while in contact (bl/s)	2.02	2.32	1.84	3.80	71%
Root displacement (bl)	8.87	9.24	9.75	10.83	61%

Reproducibility Statement

Section 3.1 and Appendix A specify the representation channel by channel, the conversion from a kinematic tree, rest pose and motion, and the per-cell normalisation; Appendix C gives the calibration protocol, the target share profile and the mechanism check; Appendix D lists every adapter hyper-parameter; Appendix L gives the adapted baseline's construction and everything it is given; Appendix G gives the AnyTop sampling protocol and the geometric protocol, including the frame-

rate handling. All evaluation runs are seeded, and the sharded generation used for retrieval metrics records the checkpoint and the sampler settings, so that a table can be re-scored from the stored samples.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods (drafting and reviewing code, including the data-conversion and evaluation scripts), to provide feedback on the research methodology and experimental protocols, to assist in interpreting results, and to generate the joint descriptions from the joint names. We have not used generative AI tools to formulate mathematical claims or to write proofs; synthetic dataset generation, translation and qualitative or thematic analysis are not applicable to this work. Additionally, we used generative AI tools for literature search, for drafting and editing the text and for reviewing drafts. We have reviewed all AI-assisted work: all AI-generated code was tested, every number in this paper was checked by the authors against the run logs, and the clip captions are the AniMo4D annotations (Wang et al., 2025) rather than AI output. The research questions, the design decisions and the final interpretation of the results are the authors'. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The work involves no human subjects and no personal data. The training library follows the clip inventory and captions of AniMo4D (Wang et al., 2025), and the evaluation rigs come from the publicly distributed Truebones zoo. The paper specifies the conversion channel by channel, so the representation can be rebuilt from the same sources. Generated animal motion for games and film carries no risk beyond the generic misuse of animation tools.

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 (SIGGRAPH)*, 39(4), 2020.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. Llm2vec: Large language models are secretly powerful text encoders. In *COLM*, 2024.

Keyi Chen, Mingze Sun, Zhenyu Liu, Zhangquan Chen, and Ruqi Huang. Topology-agnostic animal motion generation from text prompt. *arXiv preprint arXiv:2512.10352*, 2025a.

Ling-Hao Chen, Yuhong Zhang, Zixin Yin, Zhiyang Dou, Xin Chen, Jingbo Wang, Taku Komura, and Lei Zhang. Motion2motion: Cross-topology motion transfer with sparse correspondence. In *SIGGRAPH Asia*, 2025b. arXiv:2508.13139.

Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *CVPR*, 2023.

Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Motionlcm: Real-time controllable motion generation via latent consistency model. In *ECCV*, 2024.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *ICML*, 2023.

Ke Fan, Shunlin Lu, Minyue Dai, Runyi Yu, Lixing Xiao, Zhiyang Dou, Junting Dong, Lizhuang Ma, and Jingbo Wang. Go to zero: Towards zero-shot motion generation with million-scale data. In *ICCV*, 2025. MotionMillion; arXiv:2507.07095.

Inbar Gat, Sigal Raab, Guy Tevet, Yuval Reshef, Amit H. Bermano, and Daniel Cohen-Or. Anytop: Character animation diffusion with any topology. In *ACM SIGGRAPH Conference Papers*, 2025. arXiv:2502.17327; code and weights released.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *CVPR*, 2022.

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *CVPR*, 2024.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Seokhyeon Hong, Chaelin Kim, Serin Yoon, Junghyun Nam, Sihun Cha, and Junyong Noh. Salad: Skeleton-aware latent diffusion for text-driven motion generation and editing. In *CVPR*, 2025. arXiv:2503.13836.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

Aliasghar Khani, Arianna Rampini, Evan Atherton, and Bruno Roy. Unimogen: Universal motion generation. *arXiv preprint arXiv:2505.21837*, 2025.

Sunmin Lee, Taeho Kang, Jungnam Park, Jehee Lee, and Jungdam Won. Same: Skeleton-agnostic motion embedding for character animation. In *SIGGRAPH Asia Conference Papers*, 2023.

Wonkwang Lee, Jongwon Jeong, Taehong Moon, Hyeon-Jong Kim, Jaehyeon Kim, Gunhee Kim, and Byeong-Uk Lee. How to move your dragon: Text-to-motion synthesis for large-vocabulary objects. In *ICML*, 2025. arXiv:2503.04257; captions released, model code unreleased as of 2026-08.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025.

Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset. In *NeurIPS Datasets and Benchmarks*, 2023.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023.

Q. Liu, K. Lv, K. Dong, J. Xue, Z. Niu, and J. Wang. Text-to-any-skeleton motion generation without retargeting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. OmniSkel; SkeleMotion-3D, derived from HumanML3D.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023.

Linzhan Mou, Jiahui Lei, Zhiyang Dou, Chenyue Cai, Chaoyue Song, Adam Finkelstein, and Szymon Rusinkiewicz. Unimate: One unified model to animate diverse skeletons. *arXiv:2609.05415*, 2026. SIGGRAPH Asia 2026; concurrent with this submission.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.

Mathis Petrovich, Michael J. Black, and Gül Varol. Tmr: Text-to-motion retrieval using contrastive 3d human motion synthesis. In *ICCV*, 2023.

Sigal Raab, Inbal Leibovitch, Guy Tevet, Moab Arar, Amit H. Bermano, and Daniel Cohen-Or. Single motion diffusion. In *ICLR*, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21(140), 2020.

Davis Rempe, Mathis Petrovich, Ye Yuan, Haotian Zhang, Xue Bin Peng, Yifeng Jiang, Tingwu Wang, Umar Iqbal, David Minor, Michael de Ruyter, Jiefeng Li, Chen Tessler, Edy Lim, Eugene Jeong, Sam Wu, Ehsan Hassani, Michael Huang, Jin-Bey Yu, Chaeyeon Chung, Lina Song, Olivier Dionne, Jan Kautz, Simon Yuen, and Sanja Fidler. Kimodo: Scaling controllable human motion generation. *arXiv preprint arXiv:2603.15546*, 2026.

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model. In *ICLR*, 2023.

Truebones Motions Animation Studios. Truebones zoo: Motion capture animal animations. `https://truebones.gumroad.com`, 2022.

Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. Neural kinematic networks for unsupervised motion retargetting. In *CVPR*, 2018.

Xuan Wang, Kai Ruan, Xing Zhang, and Gaoang Wang. Animo: Species-aware model for text-driven animal motion generation. In *CVPR*, 2025.

Xuan Wang, Kai Ruan, Liyang Qian, Zhizhi Guo, Chang Su, and Gaoang Wang. X-mogen: Unified motion generation across humans and animals. 2026. arXiv:2508.05162.

Zhendong Wang, Yifan Jiang, Yadong Lu, Yelong Shen, Pengcheng He, Weizhu Chen, Zhangyang Wang, and Mingyuan Zhou. In-context learning unlocked for diffusion models. In *NeurIPS*, 2023.

Zhangsihao Yang, Mingyuan Zhou, Mengyi Shan, Bingbing Wen, Ziwei Xuan, Mitch Hill, Junjie Bai, Guo-Jun Qi, and Yalin Wang. Omnimotiongpt: Animal motion generation with limited animal data. In *CVPR*, 2024.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2m-gpt: Generating human motion from textual descriptions with discrete representations. In *CVPR*, 2023.

Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *IEEE TPAMI*, 2024.

Zongye Zhang, Yuzhuo Cui, Qingjie Liu, and Yunhong Wang. Semantic-aware motion encoding for topology-agnostic character animation. 2026. arXiv:2605.27055.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *CVPR*, 2019.

Table 6: TOPX-17 channel layout per joint and frame. Channels 13–16 exist on the root row only; every (rig, joint, channel) cell is normalised with its own statistics and exactly-constant cells are excluded from supervision.

slice	content	frame	supervised on
0:3	position (XZ relative to the root track, Y absolute)	world	all joints
3:9	rotation, continuous 6D delta from the rest orientation	global	free-DOF joints
9:12	world velocity (supervision only, never integrated)	world	all joints
12	contact flag	—	joints with contact
13:15	root trajectory XZ	world	root only
15:17	heading $(\cos\phi, \sin\phi)$	world	root only, valid frames

A TOPX-17 DETAILS

Where the global state lives. Four of the seventeen channels — the root track and the heading — describe the body as a whole rather than a joint, and TOPX-17 puts them on the physical root's row, where they are structurally invalid on every other row and excluded by the mask. The alternative is a virtual world node: an extra row that owns the global state and leaves the remaining thirteen channels uniform. We do not use one, because it redefines J from the number of physical joints to the number of rows, and everything the topology is described by — adjacency, geodesic distance, the parent tree the forward kinematics walks, the canonical form that decides when two rigs share a topology — would then be computed over a node the skeleton does not have. The root's own row costs one declared exception instead, and the first thirteen channels keep one meaning on every row of every rig.

Validity. Not every cell exists on every rig or frame. Frame and joint masks handle padding; rotation is supervised only on joints with free degrees of freedom, contact only on joints that can touch the ground, heading only on frames where it is defined. The per-rig cell mask multiplies the interpolant, the loss and the per-step projection at sampling time, so an invalid cell is exactly zero everywhere rather than "small". Per-frame heading validity is the one part of the mask that training has and sampling does not: it gates the target's heading channels during training, while at generation the target's heading is what is being generated, so the sampler carries the demonstration's flag only and the target heading is produced unconditionally.

Why standardise rather than scale. We found this to be decisive. Under the scale-only normalisation of the source protocol a large part of the objective is already explained by a constant pose, and the network settled on a frozen pose while the root travelled. On a 671-window probe of an earlier build, a constant prediction recovers 75% of the scale-only objective; centring each cell on its mean brings that to 41.6%, and centring and dividing by the standard deviation, the normalisation used here, brings it lower still. Table 15 measures the consequence directly on the corpus used here: at every matched epoch the standardised representation is about two points of R@1 ahead, with the FID 22–30% lower.

Conversion. A clip enters as a kinematic tree, a rest pose and per-frame local rotations and root translation, and those rotations are the source's own: a clip whose rotations cannot be traced to the rotation channels of its own file does not enter the corpus, a joint that carries a rotation channel keeps it whatever its depth, and a joint the source fixes is written from the source hierarchy and rebuilt from its parent at decode rather than left to the model. It is first resampled to 30 fps (linear interpolation of the root translation, spherical interpolation of the local rotations, then forward kinematics) and expressed in a right-handed canonical basis. Global joint rotations follow from forward kinematics; the *heading* is the ground-plane direction between a designated pair of joints of the rig, stored as the unit vector (f_z, f_x) and marked invalid (channels set to zero, excluded from supervision) on frames where that projected direction has length below 10^{-6}. The *root track* is the root joint's own XZ trajectory, stored unsmoothed on the root row; joint positions store XZ relative to it and Y as absolute height, so every window can be re-based to its own first frame without touching the pose. Rotations are the global rotation composed with the inverse rest rotation of the same joint, encoded as the first two matrix columns (continuous 6-D), so a joint at rest is the identity on every

rig. Velocities are one-step finite differences of world positions, per second, supervised but never integrated. A contact flag marks the frames on which a joint that can touch the ground is within 0.3 of the lowest point its rig's contact joints reach in the clip and whose squared displacement to the next frame is at most 0.002 (canonical units; both tests read the following frame, so the flag at frame t describes the step from t to $t{+}1$). The Truebones corpus was converted by an earlier build of the same codec, which smoothed the root track with a 1 Hz zero-phase filter, took the heading from the carrier joint's rotation with a 0.05 validity floor, and set contact from normalised height and speed (0.05 and 0.25); the main-body pruning of Appendix D precedes it. Both builds write the same 17 channels with the same meanings, which is what lets one model serve a rig from either corpus, and every Truebones number here is measured within that corpus.

Per-cell statistics. For every rig, joint and channel the mean and standard deviation are taken over all frames of all of the rig's clips (heading channels over valid frames only), and a cell is served as $(x - \mu)/(\sigma + 10^{-6})$ with σ raised to at least 0.05. A cell whose value never changes on a rig (contact of a joint that never touches the ground, a fixed degree of freedom) is exactly constant; it is normalised to zero, excluded from supervision and held at its constant when decoding. On the library this concerns $9{,}185$ of the $256{,}512$ cells that exist on the 311 rigs, 9,123 of them contact flags. A rig outside the library is served the same way from statistics computed on its own clips; how far a rig can be served without them is the subject of §4.3.

Corpus statistics. The library holds 311 rigs and 77,894 clips (73,995 training, 3,899 validation), 6.18M frames at 30 fps, or 57 hours. Clips are short: median 62 frames, mean 79, maximum 299, and 2.1% exceed the 240-frame training window. Rigs have between 64 and 946 clips (median 236). Joint counts run from 34 to 102 (median 64): 61 rigs in the thirties, 48 in the forties, 14 in the fifties, 63 in the sixties, 48 in the seventies, 27 in the eighties, 47 in the nineties and 3 above a hundred. The clips and their captions follow the official AniMo4D annotation set of Wang et al. (2025): 77,894 of its 78,149 entries have a processed clip here and the remaining 255 have none. 53,448 clips carry three captions and 24,443 carry one (three carry more), about ten words each at the median; the primary caption is the one used for training and for evaluation. One caption of one clip was dropped as corrupt (a 12,296-character repetition); its two siblings were kept. The rigs are our own re-export with deformation bones only, the natural-language joint descriptions are ours (written from the joint names, §3.2), and the conversion to TopX-17 is new.

B Model details

Figure 4 lays out the 303M model as implemented: tensor shapes along the forward pass, what each token receives, the attention bias, the sampler, and the training objective of Section 3.3 with its auxiliary terms.

Structure and graph bias. An 8-d structural vector per joint (unit rest-offset direction, log bone length, tree depth, root-path length, child count, leaf flag) enters through a zero-initialised MLP. Spatial attention receives a fixed bias $-\min(g_{ij}, 8)$ from the geodesic distance g_{ij} on the tree, plus two learned tables indexed by the number of hops up and down to the lowest common ancestor (clipped at 15), so that "parent", "sibling" and "tail of the same limb" are distinguishable; padded joints receive -10^4.

C Calibration of the grouped objective

Huber knee. The knee $\delta = 10$ lies below the measured 99.99th percentile of the normalised targets (11.45); it acts on residuals, and at initialisation it binds on 0.015% of the supervised cells, a figure recorded in the calibration artifact.

Auxiliary terms. Four terms are added, none carrying the time weight $w(t)$: a forward-kinematics consistency term between the predicted positions and the positions forward-kinematised from the predicted rotations, in de-normalised coordinates (weight 0.07, ramped over the first 5k steps); a velocity term matching world joint displacements between consecutive frames over every valid joint

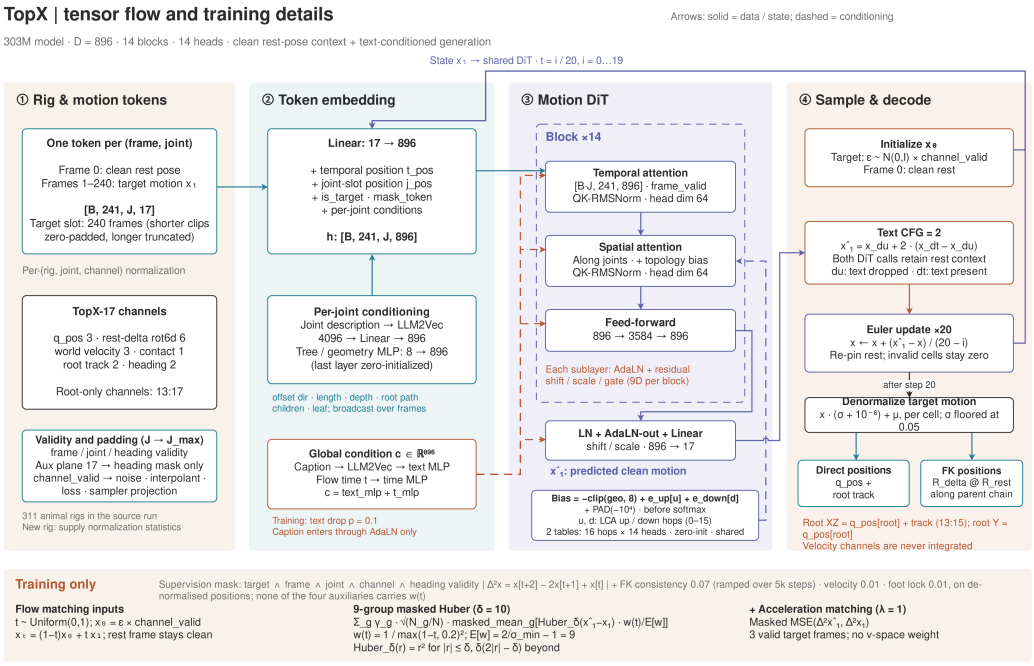

Figure 4: Tensor flow and training details of the 303M model. Solid arrows carry data or state, dashed arrows conditioning. Left to right: rig and motion tokens with the 17-channel layout and the validity masks; the token embedding and the two conditioning paths (per-joint, and the global vector $\mathbf{c}$ that drives AdaLN); the fourteen factorised blocks with the spatial-attention bias; sampling with caption guidance, the 20 Euler steps, de-normalisation and the two decodes. The bottom band lists the training inputs, the grouped Huber objective of Eqs. (1)–(2), the acceleration term and the three further auxiliary terms.

(0.01); a foot-lock term, gated by the de-normalised contact channel, that demands zero displacement of a joint flagged in contact (0.01); and an acceleration-matching term on the second temporal differences of the normalised channels (1.0), introduced against the residual jitter of early runs.

Groups, families and target shares. The 17 channels form nine groups (root position, root rotation, root velocity, body position, body rotation, body velocity, contact, root track, heading), collected into six families with a prescribed share of the objective: root trajectory (root position and root track) 0.304, heading 0.012, body position 0.304, rotation (body and root) 0.304, velocity (body and root) 0.027 and contact 0.049. Each group's energy E_g is the mean squared standardised target over its valid cells, measured in one pass over the training cohort at the run's batch and demonstration condition; the family weight is then the closed form $\gamma_f = \sqrt{s_f / \sum_{g \in f} E_g}$, applied to every group of the family, with the rotation family anchored at 10 and the others scaled relative to it. On the library this gives 12.2 for the root trajectory, 14.2 for body position, 10 for rotations, 3.0 for velocities, 5.6 for contact and 2.8 for heading. The cancellation is of a group's aggregate cell count over the batch: within a batch of mixed rigs, a rig's share of a group's gradient is its share of the group's cells.

Mechanism check and artifact. Because the weights act on gradients rather than on targets, the artifact is certified by running 30 optimiser-free steps of a small model under the run's exact objective (velocity-space loss, $\sigma_{\min}$=0.2, Huber knee 10, uniform t) and requiring the gradient share of every family, accumulated over those steps, to lie within 25% of $\gamma_f^2 \sum_{g \in f} Q_g$, where Q_g is the group's mean time-weighted, Huber-clipped squared residual. This is a check of the wiring: it says the weights act on the gradient the way the solve assumes, at initialisation. It says nothing about the share realised during training, which moves with the error energies – the artifact itself records that it

cannot verify the design target. The artifact records the corpus generation, the normalisation variant, the statistics file, the batch, the demonstration condition and the loss code; a derived training view records its manifest, clip set and cut as well. The trainer refuses to start on any mismatch of what the artifact records, so a calibration cannot be reused across a change of data, batch, normalisation or objective without being re-measured.

D ADAPTER SETTINGS

Data view of a new rig. The data view of a new rig is built by the corpus script with three additions: joints are pruned to the deformation-bone convention of the training library (end markers, cosmetic and helper joints dropped by name or description, never the root, heading carrier, trunk or wing joints, and only leaf-closed subtrees so no forward kinematics is re-run); captions are rewritten to the corpus style "The ⟨species⟩ ⟨action⟩"; and the clips are split 80/20 into train and validation. Training follows the backbone recipe with the learning rate scaled linearly to the batch, one pass over every clip per epoch, 300–900 epochs and a shortened forward-kinematics warm-up. A single adapter can also be shared by a group of rigs, which removes the unadapted model's jitter excess on all of them and, on the one rig where both were trained, comes close to its own adapter (App. E).

Hyper-parameters. Table 7 lists the per-rig settings. The Truebones arms of Table 3 are sampled with seed 7; the zero-training arm and the adapters start from the same backbone snapshot. The learning rate follows the linear-scaling rule from 10^{-4} at batch 8; the batch divides the rig's training-clip count so that every clip is seen exactly once per epoch, and is at most 13. Adapters act, with scale $\alpha{=}1$, on the query, key, value and output projections of both attention blocks, the feed-forward layers and the conditioning MLPs (62 linear layers of the 36M backbone, 104 of the 303M one); a 100-step warm-up precedes a half-cosine decay over five sixths of the epochs to 1% of the peak, and the forward-kinematics term ramps over 100 steps. At rank 64 the adapter holds 7.06M trainable parameters on the 36M backbone and 26.8M on the 303M one (16% and 8% of the respective totals); Dragon uses rank 128 (53.6M, 15%) and three times the epochs.

Table 7: Per-rig adapter settings. Clips are the usable Truebones clips after the source audit, split 80/20.

Rig	Joints	Train / val clips	Batch	Learning rate	Rank	Epochs
Buffalo	30	15 / 5	5	$6.25{\times}10^{-5}$	64	300
Gazelle	29	8 / 2	8	$1.0{\times}10^{-4}$	64	300
Dragon	95	10 / 3	5	$6.25{\times}10^{-5}$	128	900
Spider	68	26 / 6	13	$1.63{\times}10^{-4}$	64	300

E ADDITIONAL EXTENSION RESULTS

Smaller backbones. Repeating the study with the 36M pilot as the backbone gives the same picture: the zero-training arm is less jittery than the 303M zero-training arm on every rig (jitter ratio 1.92 / 6.24 / 0.74 / 4.71 vs. 3.32 / 13.2 / 2.53 / 6.49) with articulation 1.06 / 2.48 / 0.21 / 1.93, and the LoRA arm is close to the 303M LoRA (0.27 / 0.92 / 0.62 / 0.23). Rank 64 is 16% of the trainable parameters of a pilot and 8% of the main model; these are same-rank, not same-relative-capacity, controls. The Truebones-only 36M model, which has trained on the evaluation clips, reproduces the real clips' statistics on every rig (jitter ratio 1.02 / 1.92 / 1.12 / 0.37 and articulation 0.67 / 1.22 / 1.23 / 0.35 against all clips of the rig; 0.95–1.29 and 0.85–1.00 against the clips it was asked to reproduce), unchanged between 30k and 60k steps. Against those matched targets it is the one arm whose jitter sits at the real value; the LoRA arms come out *smoother* than real motion under the all-clip denominator used in Table 3 (jitter ratios 0.2–0.9), and the Truebones-only model does so on Spider alone (0.37).

An adapter shared across species helps, and does not replace a per-rig one. One rank-128 adapter trained on the pooled clips of eight flying Truebones rigs (Bat, Bird, Buzzard, Dragon, Eagle, Giant bee, Parrot, Pteranodon; 78 training clips) is scored on the validation clips of seven

of them with the geometric protocol of §4.1 (all real clips of the rig as the reference; at the rigs' native 30 Hz, as no external source is involved). Table 8 compares it with the zero-training arm on seven rigs. The shared adapter removes the zero-training arm's jitter excess (median 5.5 to 0.99 times real) and settles articulation at 0.46–0.83 of the real clips. On Dragon, the one rig that also has an adapter of its own, the two are close: at this table's 30 Hz, articulation 0.52 shared against 0.49 species-specific and jitter 0.95 against 0.70 (the species-specific adapter reads 0.52 in Table 3, which is scored at 20 Hz beside the AnyTop samples), both nearer the real motion's 1 for the shared adapter, while the species-specific one is nearer on root jitter. Dragon is the only rig here that has both, so what this table establishes is that pooling eight bodies removes the unadapted jitter excess on all seven and, on that one rig, reaches an adapter of its own; whether it does so on the other six is not measured.

Table 8: A shared adapter over eight flying rigs against zero training (303M backbone, FK output relative to all real clips of the rig, 30 Hz; 1 = real). n: validation clips.

		Articulation ratio		Jitter ratio	
Rig	n	zero	shared LoRA	zero	shared LoRA
Bat	2	1.79	0.51	5.52	0.57
Bird	3	3.89	0.81	12.2	1.34
Buzzard	2	1.40	0.46	4.18	0.38
Dragon	3	1.14	0.52	4.28	0.95
Eagle	2	3.88	0.83	13.2	0.99
Giant bee	2	1.72	0.69	4.28	1.02
Parrot	3	1.84	0.55	7.14	1.14

Statistics from the adaptation clips alone. The per-cell statistics a new rig arrives with are measured over all of its clips in Table 3, evaluation clips included. Table 9 re-runs both arms with statistics measured from the rig's 8–26 adaptation clips only: a sibling view that declares those statistics, a calibration measured on it, the adapter re-trained under the published recipe, and both arms re-sampled and scored by the protocol of §4.1. The adapter still brings the two channels together and still removes the jitter excess; the adapted gap moves by at most 0.13 bone lengths, and the articulation and jitter ratios by less than the difference between rigs.

Table 9: Statistics from the adaptation clips only (303M backbone, same recipe, same protocol as Table 3; released values in brackets). Gap in bone lengths; articulation and jitter of the sampled FK output relative to all real clips of the rig (1 = real). Spider's real root is static, so its root-speed ratio is undefined.

Rig	Arm	FK–pose gap, sampled (bl)↓	Articulation ratio	Jitter ratio
Buffalo (30 j.)	zero	0.722 (0.797)	1.34 (1.45)	3.00 (3.32)
	LoRA	0.311 (0.349)	0.34 (0.30)	0.39 (0.33)
Gazelle (29 j.)	zero	0.579 (0.574)	5.40 (5.39)	12.97 (13.17)
	LoRA	0.250 (0.227)	0.76 (0.77)	0.79 (0.86)
Dragon (95 j.)	zero	4.564 (4.458)	1.08 (1.02)	2.75 (2.53)
	LoRA	0.363 (0.304)	0.56 (0.49)	0.71 (0.52)
Spider (68 j.)	zero	2.135 (1.896)	2.99 (2.87)	6.67 (6.49)
	LoRA	0.364 (0.232)	0.28 (0.24)	0.27 (0.20)

How long to train the adapter. Table 10 re-trains the four adapters of Table 3 with a snapshot every 50 epochs (Dragon: every 150 of 900) and scores every snapshot under the protocol of §4.1. The jitter excess of the zero-training arm is gone within 50–100 epochs on every rig (50–300 optimiser steps on 8–26 clips), the FK–pose gap reaches its plateau by 100–150 epochs (Dragon: 450) and then moves by less than the five-or-fewer-clip sampling noise, and the articulation ratio settles at 0.2–0.8 of the real clips whichever length is used: longer adaptation removes the disagreement between the rotation and position channels but does not restore the amplitude of motion. Gazelle, with two validation clips, is the noisiest row.

Table 10: Adapter training length. Snapshots of the per-rig LoRA (303M backbone, same clips, learning rate and batch as Table 3); FK–pose gap of the sampled output in bone lengths, articulation and jitter of the sampled FK output relative to all real clips of the rig (1 = real). Epoch 0 is the library model with the rig's statistics and no adapter update; every column, the gap included, is measured on that snapshot's own generated sequences. Dragon's snapshots are at 150 / 300 / 450 / 600 / 900 epochs.

Rig		0	50	100	150	200	300
Buffalo	gap	0.797	0.537	0.298	0.334	0.276	0.294
	articulation	1.45	0.43	0.30	0.26	0.27	0.28
	jitter	3.32	0.66	0.34	0.30	0.30	0.30
Gazelle	gap	0.574	0.349	0.247	0.242	0.240	0.223
	articulation	5.39	0.73	0.36	0.56	0.79	0.78
	jitter	13.2	1.04	0.45	0.58	0.80	0.80
Spider	gap	1.896	0.617	0.472	0.241	0.262	0.217
	articulation	2.87	0.44	0.28	0.20	0.28	0.24
	jitter	6.49	0.57	0.28	0.19	0.25	0.22
		0	150	300	450	600	900
Dragon	gap	4.458	0.534	0.376	0.312	0.307	0.299
	articulation	1.02	0.47	0.42	0.50	0.49	0.51
	jitter	2.53	0.61	0.52	0.59	0.59	0.56

F CAPABILITIES OF RELATED SYSTEMS

Table 11 lists the capabilities of the related systems as described in their papers and released code. Two capabilities separate this task from its neighbours, and one concurrent system, UniMate (Mou et al., 2026), combines them as this paper does: text and skeleton conditioned, joint rotations with a global root, an optional foot lock after generation, and seven held-out skeleton types scored quantitatively. A caption is not what AnyTop (Gat et al., 2025) conditions on: generation selects a character rather than a prompt, and the released code has no classifier-free guidance. OmniZoo (Chen et al., 2025a) draws the same line when it chooses its own baselines, excluding AnyTop "because it performs unconditional generation without text control". Conversely, the text-conditioned systems with arbitrary-skeleton support generate for skeletons of one body plan — OmniSkel's (Liu et al., 2025) user-defined skeletons are human, and SATA's (Zhang et al., 2026) text-to-motion is a downstream demonstration on HumanML3D, its quantitative evidence being cross-topology reconstruction and retargeting.

Table 11: Capabilities of motion generators across skeletons, as described in the papers and released code. *Any topology*: one model serves skeletons of different joint counts and trees. *Text*: language-conditioned. *Rotations*: joint rotations are predicted. *IK-free*: the exported animation needs no inverse-kinematics fit. *Root*: a world-space root trajectory is generated. *Unseen rig*: a rig absent from training is served, and how. *Baselines*: how the paper compares to other methods. n.r. = not reported. G-MDM omits root translation in its current implementation.

	Any topology	Text	Rotations	IK-free	Root	Unseen rig	Baselines
Human T2M diffusion (Tevet et al., 2023; Zhang et al., 2024)	×	✓	✓	× (fit on HumanML3D)	✓	×	standard HumanML3D protocol
AniMo (Wang et al., 2025) (30-joint skeleton)	×	✓	✓	✓	✓	×	retrained human T2M models
OmniMotionGPT (Yang et al., 2024)	×	✓	✓	✓	✓	×	retrained on its data
X-MoGen (Wang et al., 2026)	×	✓	×	n.r.	n.r.	×	retrained on its data
AnyTop (Gat et al., 2025)	✓	×	✓	× (IK export)	✓	zero-shot, qualitative	MDM adapted, SinMDM per rig
OmniZoo (Chen et al., 2025a)	✓	✓	✓	n.r.	n.r.	qualitative demos	padded MoMask / MMM / AniMo
G-MDM (Lee et al., 2025)	✓	✓	✓	✓	×	qualitative	per-object MDMs, retargeting
UniMate (Mou et al., 2026) (concurrent)	✓	✓	✓	✓ (optional foot lock)	✓	7 held-out skeleton types, quantitative	AnyTop with text added, AnimateAnyMesh, V2M4
TOPX (ours)	✓	✓	✓	✓	✓	given its statistics, + adapter	released AnyTop as reference

G ANYTOP AS AN IN-DISTRIBUTION REFERENCE

AnyTop on the same rigs. AnyTop (Gat et al., 2025) is the released arbitrary-topology generator; it is unconditional and was trained on the whole Truebones zoo, including the four rigs above, so on them it is an in-distribution model. Of our own arms only the unadapted library model has never

Table 12: Per-rig kinematics on the Truebones zoo, on the deployable output, relative to all real clips of the rig (1 = real). AnyTop: released weights, 8 samples per rig, unconditional, *trained on these rigs* (an in-distribution reference, not a competitor). Ours: 303M model, held-out clips, caption-conditioned; *zero* = statistics only, *LoRA* = per-rig adapter (FK–pose gap as in Table 3). Spider's real root is static, so its root-speed ratio is undefined. n = number of sequences.

Rig	Source (n)	Jitter	Articulation	Root speed	FK–pose gap, sampled (bl)
Buffalo	AnyTop (8)	0.83	0.55	0.30	–
	Ours, zero (5)	3.32	1.45	0.43	0.797
	Ours, LoRA (5)	0.33	0.30	0.44	0.349
Gazelle	AnyTop (8)	0.86	0.85	0.16	–
	Ours, zero (2)	13.2	5.39	2.13	0.574
	Ours, LoRA (2)	0.86	0.77	0.21	0.227
Dragon	AnyTop (8)	0.70	0.50	0.26	–
	Ours, zero (3)	2.53	1.02	1.44	4.458
	Ours, LoRA (3)	0.52	0.49	0.47	0.304
Spider	AnyTop (8)	0.41	0.48	n/a	–
	Ours, zero (6)	6.49	2.87	n/a	1.896
	Ours, LoRA (6)	0.20	0.24	n/a	0.232

seen these rigs: the adapters train on the rig's own support clips, and the Truebones-only model trains on all of its captioned clips, evaluation clips included. AnyTop generates from a rig alone, so Table 12 answers one question: what does a model that trained on these rigs produce, and where do our zero-training and adapted arms sit beside it on the same rigs and the same statistics? We sample the released weights eight times per rig (6 s each) and report, for its direct output positions and for the BVH it exports after an inverse-kinematics fit, the same per-rig statistics as for our arms. These say how much and how smoothly the outputs move relative to the rig's real clips, so a ratio of 1 is the target from either side. AnyTop's IK export raises its jitter by 1.0–$1.7\times$ over its direct positions, whereas the pose and FK rows of our adapters have similar jitter (0.33 vs. 0.32 on Buffalo); the un-adapted library model is 1.3–$1.8\times$ jerkier through its rotations than through its positions, which the adapter removes.

Protocol details. AnyTop samples are produced with the authors' released model and sampling script; their BVH files declare 24 fps but the generator runs at an effective 20 fps, and we use the effective rate. Our samples use the deployment sampler (20 Euler steps, s=2, seed 7, rest-frame demonstration) and are read in both families: the directly predicted positions and the forward kinematics of the predicted rotations on the rig's own bone lengths; the table reports the kinematic family, which is the deployable output, and the gap between the two is the FK–pose gap of §4.1. Joint order is validated against the rig's conditioning table before any metric is taken. All sequences, real and generated, are decimated to a common 20 Hz with an anti-aliased polyphase filter when an AnyTop sample is present, and kept at the native 30 Hz otherwise.

H SAMPLING COST

Table 2 (§4.2) gives the wall-clock time of one sample at batch size 1 on one H200 GPU in fp32, for the protocol's padded 240-frame sequence of a 38-joint rig (109 of the frames are real), with caption guidance (two network evaluations per Euler step), median of five timed runs after two warm-up runs, next to the retrieval and FID scores of the same checkpoints at 10 and 5 steps. The time is linear in the number of function evaluations and the 36M pilot is six times faster than the 303M model. Halving the steps leaves R@1 unchanged (0.8643 against 0.8633 for the 303M model, 0.8401 against 0.8396 for the pilot) and costs 0.003–0.005 in FID; five steps cost 0.008 in R@1 for the 303M model and 0.018 for the pilot, so the smaller model needs its steps more. Departures from the 20-step protocol are stamped as such in the evaluation reports (`protocol.variant`) and are never mixed with frozen-protocol numbers.

I METRIC DEFINITIONS

The Truebones experiments (§4.3) and the geometry tables are measured in world space on the deployable output, the forward kinematics of the predicted rotations. The *articulation ratio* is the mean joint speed relative to the root divided by the real clips' value, the *jitter ratio* the mean second temporal difference likewise normalised, and the *root speed ratio* the same for the root. Ratios are taken against *all* real clips of the rig at a common 20 Hz, so that AnyTop samples and ours are measured identically. The FK–pose gap of the library ablations is read from the training-time validation pass at the stated epoch (App. C); on the Truebones arms and on the frozen samples of Tables 5 and 16 it is measured on the generated sequences themselves, so that it comes from the same output as the articulation and jitter reported beside it.

Diversity and MModality. Both follow Guo et al. (2022) on the evaluator's normalised motion embeddings. Diversity is the mean Euclidean distance between two lists of 300 clips drawn without replacement from the 3,899 generated (or real) clips, computed once per sampling seed on the samples of Table 1 and averaged over the five seeds. MModality generates 30 samples for each of 403 captions (one caption per rig-shard of the validation order, seeds 101–130, the sampler of §3.4) and averages the distance over ten pairs per caption; the pairing is drawn 500 times and averaged, which removes the variance of a single draw. Real clips have one sample per caption, so the reference row has no MModality.

Retrieval on the played animation. The evaluator of §4.1 scores TOPX-17 channels; the rig plays rotations and a root trajectory. To score what is played, the seed-42 samples of Table 1 were rescored with their position channels replaced by the forward kinematics of the predicted rotations and their velocity channels recomputed from those positions with the codec's own convention, rotations, contact and root channels as predicted, under the same evaluator, pools and acceptance rule. At 303M this gives R@1 0.865, FID 0.0049 and matching 0.762 against 0.864, 0.0051 and 0.764 on the sampled channels; at 36M, 0.837, 0.0087 and 0.742 against 0.840, 0.0083 and 0.746. The real clips' reference is unchanged, and the two decodings of a real clip agree to 1e-6 bone lengths under the same code, so the two scores differ by the FK–pose gap of the samples and by nothing else.

J TRAINING DETAILS

Table 13 gives every symbol of §3 its value. Both models train in bf16 on 8 H200 GPUs, 578 optimiser steps per epoch at the global batch of that table; the optimiser and schedule settings are those of that table.

Table 13: Symbols of §3 and their values. Two values are the 303M main model / the 36M pilot; one value is shared.

Architecture, representation, conditioning			Objective, optimisation, sampling		
D	width	896 / 384	$\sigma_{\min}$	time-weight floor, Eq. (1)	0.2
N_b	blocks	14 / 8	δ	Huber knee, Eq. (1)	10
H	heads	14 / 6	γ_{rot}	rotation-family anchor	10
–	parameters	302.9M / 36.3M	η	peak learning rate	$1.5 / 2 \times 10^{-4}$
L	target frames	240	B	global batch	128
ϵ_σ	std floor of a cell	0.05	–	warm-up steps; epochs	4k; 290 / 400
$g_{\max}$	hop-distance clip	8	–	cosine floor	10^{-2} over 40 epochs
$a_{\max}$	LCA hop clip	15	–	gradient clip; spike reject	1.0; 200
p_{drop}	caption dropout	0.1	S, s	Euler steps, guidance	20, 2
$\lambda_{\text{FK}}, \lambda_{\text{vel}}, \lambda_{\text{lock}}, \lambda_{\text{acc}}$	auxiliary weights: 0.07 (ramped over the first 5k steps), 0.01, 0.01, 1.0				

K ABLATION NOTES

What each ingredient is worth. Table 4 prices them one at a time. Every arm is a 36M pilot on the same clips with the same cut, split, captions, rest-pose demonstration, batch, learning rate, schedule and budget, read at the same three epochs under the frozen protocol.

The arm *without the joint descriptions* zeroes and freezes the description projection and keeps everything else. It is level with the full recipe at all three epochs, within 0.003 of R@1, and its FID is 10% lower at epoch 100.

The arm *without the calibrated weights* keeps every conditioning pathway and runs the grouped objective at uniform weights (γ_g=1 for every group, where the calibrated set anchors the rotation family at 10, so the grouped term's weight against the auxiliary terms differs as well). It loses 0.036–0.043 of R@1 and its FID rises by 61% at epoch 100, and it gains less between epochs 50 and 100 than the full recipe does.

The arm that drops *both, and the graph conditioning with them* — descriptions zeroed, geodesic and direction biases off, structural features dropped, uniform weights — loses 0.067–0.072. Against the arm without the calibrated weights, the remaining 0.029–0.033 is the cost of removing the descriptions together with the geodesic and direction biases and the structural features; the description-only arm above was run with calibrated weights, so the two removals are not separated under uniform weights.

The last arm is the adapted baseline of App. L: the same corpus, split, captions, rest pose, objective, weights, schedule and budget, with the joint axis flattened into one token per frame. It loses 0.192–0.201 of R@1 and its FID is 3.7 times the full recipe's, and it is 0.123–0.134 below the arm that keeps per-joint tokens with every skeleton pathway switched off.

Table 14: Ablations on the 36M pilot at matched optimiser steps (epoch 200; validation loss, FK–pose gap from the validation pass, R@1 in pools of 64 and FID on the validation set). The variant without the acceleration term optimises a different objective, so its validation loss is not listed.

Variant	Val. loss↓	FK–pose gap (bl)↓	R@1↑	FID↓
Full recipe (rest demo, calibrated, $\mathbf{x}$-pred, qk-norm)	0.247	0.171	0.821	0.011
Motion demonstration in place of the rest frame (App. N)	0.270	0.202	0.822	0.010
Without the acceleration-matching term	–	0.171	0.822	0.011

Prediction target and sampler. Predicting $\mathbf{x}_1$ rather than the velocity lowered the high-frequency energy of the samples in our measurements, and we deliberately keep the uniform t sampler: the logit-normal sampler and $\sigma_{\min}$=0.05 of Li & He (2025) converged faster in loss but concentrated nearly all supervision at $t \geq 0.8$ and visibly degraded global form, so they were rolled back on visual grounds.

Normalisation of the representation. TOPX-17 is normalised per rig and per cell (joint, channel) by the cell's mean and standard deviation over the rig's clips (App. A). The alternative that keeps every cell's own mean and divides only by a per-rig scale (a scale-only convention) was trained with the 36M recipe unchanged and scored at matched epochs under the frozen protocol (its samples are mapped into the per-cell space before scoring). Table 15 shows that the scale-only variant is behind in retrieval and FID at every matched epoch, by about two points of R@1 and 22–30% of the FID, although its two output channels agree more closely with each other: the FK–pose gap is computed after each arm's own normalisation is reversed, in the same bone-length unit, so the scale-only model's smaller gap says that its two channels are more consistent with each other, not that its samples are closer to the target – retrieval and FID say the opposite. The per-cell normalisation is the one used everywhere else in the paper.

Table 15: Per-cell against scale-only normalisation at matched epochs (36M recipe, global batch 128, frozen protocol: pool 64, 20 Euler steps, guidance 2, seed 42). FK–pose gap in bone lengths from the training-time validation pass at the same epoch.

	Per-cell mean/std (ours)			Scale-only		
Epoch	R@1↑	FID↓	gap, val. pass↓	R@1↑	FID↓	gap, val. pass↓
50	0.798	0.0157	0.198	0.778	0.0201	0.117
75	0.803	0.0146	0.193	0.786	0.0194	0.111
100	0.808	0.0138	0.187	0.789	0.0198	0.107

Representation content. To separate what the channel layout of TopX-17 contributes from what its normalisation contributes, the older 13-channel AnyTop layout (root-relative, heading-canonical joint positions, the root's height and planar velocity, per-parent local 6D rotations, local velocities and contacts; no root-track or heading channels) was derived analytically from the same clips, splits and exclusion cut, and the 36M recipe was trained on it with the same per-cell normalisation and the same schedule. The two world-coordinate terms of the training loss (forward-kinematic position and foot lock) read TopX-17's channels directly, and the 13-channel layout carries neither the root track nor the heading they are written against, so that run had them off; the velocity and acceleration terms were kept. The geometry of Table 16 is therefore the representation and those two terms together, which is the combination this paper trains. Its samples are converted back to TopX-17 before scoring, so the frozen protocol and the reference numbers are unchanged. Table 17 shows the two layouts within a point of R@1 of each other at every matched epoch (0.804, 0.808 and 0.814 against 0.798, 0.803 and 0.808), with the FID differing by at most 0.002: the retrieval gains in Table 15 come from the normalisation, not from the channel layout.

That comparison changes the objective as well as the layout, so we ran TopX-17 with those two terms off, at the 13-channel arm's own batch partition. At epoch 50 it reaches R@1 0.812 and FID 0.0140, against 0.804 and 0.0154 for the 13-channel layout under the same objective and partition: with the objective held fixed the two layouts are 0.008 of R@1 apart, and the layout is not what the retrieval numbers are measuring.

The same arm separates the two kinds of measurement. It is *ahead* of the full recipe on both evaluator metrics at every matched epoch — R@1 0.812 / 0.814 / 0.814 against 0.798 / 0.803 / 0.808, and FID 0.0140 / 0.0136 / 0.0133 against 0.0157 / 0.0146 / 0.0138 at epochs 50, 75 and 100 — and behind it on essentially every geometric quantity, clip for clip over the same 3,899 samples: the FK–pose gap doubles, from 0.41 to 0.82 bone lengths, with the full recipe closer to the ground truth on 98.8% of clips; bone-length error 0.081 against 0.113 (95.6%); foot slide while in contact 2.40 against 3.62 where the real clips give 2.04 (72.8%); root jitter 0.048 against 0.073 against a real 0.038 (82.6%); and overall jitter 0.092 against 0.118 against a real 0.094 (69.5%; the geometry is measured at epoch 50, where the retrieval gap is widest). Articulation is unchanged and the turn magnitude is marginally better without the terms. The two terms cost between half a point and a point and a half of R@1 — the gap narrows as both arms train — and buy the property that makes the output playable, which the retrieval score does not measure. The 13-channel layout has no rotation slot for leaf joints, whose orientation the conversion fills from the parent (about ten degrees of mean error on ground truth); the layout used in the paper keeps it. What the layout changes, and what it changes only together with those two terms, is measured on the same frozen samples (Table 16): every generated clip and its ground-truth clip are decoded with the official codec and compared in the rig's mean bone length. The two decodes of a TopX-17 sample, its positions and the forward kinematics of its rotations, agree to 0.38 bone lengths against 1.45 for the 13-channel layout; its direct positions keep the rest bone lengths better; its leaf joints articulate (10.6 degrees of mean deviation from rest against 11.6 degrees in the data, where the 13-channel layout has no leaf rotation to articulate); its feet slide less while flagged in contact (2.3 against 3.8 bone lengths per second, the data 2.0); and its root travels and stands where the data does, within a few percent in displacement and height, against 22% further and 4% lower for the layout that integrates a root velocity. The 13-channel layout is a little smoother at the root and articulates internal joints a little more. These are the quantities the layout was designed around; retrieval does not resolve them.

Model size. Table 18 scores three sizes of the same recipe – width, depth and head count grow together, so it measures model size and not the contribution of any one of them – at the same number of optimiser steps and data passes (epoch 200, global batch 128, frozen protocol). Retrieval and FID improve monotonically with size; at the same sampling seed the 303M model at epoch 200 is 0.007 of R@1 below the checkpoint of Table 1 (0.857 against 0.864, Table 19), so most of its gain over the pilot comes from size, not from the longer schedule.

Stability. Without query–key normalisation the 303M model diverged in every attempt, between epochs 11 and 60, whichever of learning rate, gradient clipping, spike rejection or batch size was changed; probing the attention logits of a healthy and of a collapsing checkpoint of one such run showed the first block's temporal attention already saturated on the healthy checkpoints (maximum logit 1372–1483, where a sane transformer runs at tens) and exploding to 12392–23391 across the

Table 16: Geometry of the generated clips against their ground truth (epoch 100, the 3,899 validation clips and the same frozen samples as Table 17; official codec; bl = the rig's mean rest bone length). Last column: share of clips on which the TOPX-17 sample is closer to the ground truth than the 13-channel sample (ties count for neither). Every row uses all 3,899 clips except foot slide, which each stream computes under its own contact flags and which is therefore compared only on the 3610 clips where all three are defined; over all 3,899 clips the streams flag contact on a similar share of frames — 0.58 for TOPX-17, 0.56 for the 13-channel layout, 0.57 for the 303M model and 0.59 in the real clips — so the lower sliding is not a matter of flagging fewer frames. Epochs 50 and 75 give the same picture. The 303M column is the main model's own frozen samples (the checkpoint of Table 1), decoded through the same stream and scored on the same 3,899 clips; it is read against the ground truth clip for clip exactly as the 36M column is, and it is closer to the ground truth than the 36M pilot on the FK–pose gap (95.4% of clips), bone-length error (96.3%), foot slide (65.8%) and internal articulation (72.9%), and farther from it on root displacement, root path length and the two jitter rows. The TOPX-17 closer column compares the two 36M layouts, which is what that comparison is about. Foot slide is the one reference that depends on which pair of arms supplies the contact frames: 2.02 against the 13-channel arm, 2.03 against the 303M model.

		TOPX-17			
Quantity	Data	36M	303M	AnyTop-13	TOPX-17 closer
FK–pose gap, sampled (bl)	0.000	0.381	0.236	1.447	100%
Bone-length error, direct pos. (bl)	0.000	0.077	0.052	0.119	91%
Leaf articulation, mean dev. from rest (deg)	11.6	10.6	11.3	0.0	92%
Leaf articulation, temporal std (deg)	5.86	5.24	5.69	0.00	89%
Foot slide while in contact (bl/s)	2.02	2.32	1.84	3.80	71%
Root displacement (bl)	8.87	9.24	9.75	10.83	61%
Root path length (bl)	11.26	11.04	11.81	12.87	55%
Root height (bl)	6.97	7.05	7.01	6.71	54%
Heading turn (rad)	1.07	1.13	1.15	1.04	43%
Root jitter (bl)	0.038	0.047	0.050	0.044	30%
Joint jitter (bl)	0.094	0.090	0.099	0.112	49%
Internal-joint artic., mean dev. from rest (deg)	19.3	17.9	19.0	18.8	42%
Internal-joint artic., temporal std (deg)	8.99	7.11	8.26	8.16	32%

Table 17: Channel layout at matched epochs (36M recipe, global batch 128, per-cell normalisation in both; frozen protocol: pool 64, 20 Euler steps, guidance 2, seed 42).

	TOPX-17 (ours)		AnyTop-13 layout, derived	
Epoch	R@1↑	FID↓	R@1↑	FID↓
50	0.798	0.0157	0.804	0.0154
75	0.803	0.0146	0.808	0.0161
100	0.808	0.0138	0.814	0.0155

damaging step. With query–key normalisation the same recipe trained for 290 epochs without a single rejected step.

L A FLAT, PADDED-JOINT BASELINE

Fixed-skeleton text-to-motion generators are brought to variable skeletons by making the tensor shape uniform. Gat et al. (2025) concatenate all joint features of a character and pad them to a length of $J \times D$, put the vectorised rest pose on the temporal axis as frame 0, and use the skeleton's name as the text. OmniZoo (Chen et al., 2025a) extends MoMask and MMM with joint padding and binary masking and appends species tags to the text prompts, on the same 95/5 split.

We build that construction on our corpus. One token per *frame* carries every joint's channels, zero-padded to 102 (the library's largest rig) and masked by the batch's joint-validity flags; frames attend to frames; the caption modulates every block through AdaLN. It reads the same per-cell normalised clips, the same cut and split, the same captions — which name the species, so no tag is appended

Table 18: Model size at matched optimiser steps (epoch 200, global batch 128; frozen protocol, pool 64, 20 Euler steps, guidance 2, seed 42).

Model	Width / depth / heads	R@1↑	R@3↑	FID↓
36M	384 / 8 / 6	0.821	0.973	0.0113
116M	640 / 10 / 10	0.843	0.982	0.0071
303M	896 / 14 / 14	0.857	0.987	0.0052

— the rig's rest pose as frame 0, and the same channel-validity masks, the fixed-DOF rotation rows and the statistically constant cells among them. It optimises the same flow-matching objective at the same group weights, on the same batch, learning rate, schedule and budget, and is read under the same frozen protocol. Width and heads are the pilot's and the depth is 12 rather than 8, which gives it 38,020,934 parameters against the pilot's 36,276,177.

What it does not have is the joint axis and what lives on it: per-joint tokens, the natural-language joint descriptions, the learned joint-slot embedding, the geodesic attention bias, the directional bias and the structural joint features. Table 4 reads it at the same three epochs. It reaches R@1 0.597, 0.603 and 0.616 against the pilot's 0.798, 0.803 and 0.808, with FID 0.0515 against 0.0138; the three epochs rise monotonically. The arm above it in that table keeps per-joint tokens and switches every skeleton pathway off; it also runs the objective at uniform weights, so the two differ in the weighting as well as in the joint axis.

M Additional qualitative results

Figure 5 shows the zero-training and LoRA arms of Table 3 on one held-out clip of three of the four rigs: the clip of each rig whose adapted sample is nearest its held-out motion (mean joint distance after anchoring both on the root, in mean rest bone lengths); Figure 3 applies the same rule to all four rigs. Every panel overlays the two decodings of the same sample: the directly predicted joint positions and the forward kinematics of the predicted rotations. The zero-training arm produces two skeletons that disagree on every rig, most on the Spider, whose gap approaches two bone lengths; the adapter brings them into agreement, which is what the drop of the FK–pose gap in Table 3 measures. Figure 6 shows the Dragon.

Two prompts on one unseen rig. Figure 6 draws adapted samples without anchoring the root: every frame keeps the height the model produced, and the root's height profile runs through the strip. Each pair shares the rig, the adapter, the per-cell statistics, the rest-pose demonstration, the sampler and its seed, and differs only in the prompt. On the Dragon the dive spans 9.22 bone lengths of root height against 2.08 for level flight; on the Spider, whose root stays where the prompt leaves it, the joints move 0.88 bone lengths per second relative to the root while it scrambles and 0.16 while it lies injured.

N A motion demonstration in place of the rest frame

The demonstration slot of §3.2 can hold motion as well as a pose. We trained a 36M pilot identical to the mainline except that the demonstration is a 64-frame window of another real clip of the same rig, drawn afresh for every training sample (the grouped objective was re-calibrated for this condition and returned the same weights, so the two validation losses are on the same scale). At the same optimiser step (115,600, epoch 200, scored on the same GPUs) the motion demonstration is worse where the two representations of a pose have to agree — validation loss 0.270 against 0.247 and FK–pose gap 0.202 against 0.171 bone lengths — equal in retrieval (R@1 0.822 against 0.821) and slightly better in FID (0.010 against 0.011). The attribution of the demonstration pathway is an order of magnitude smaller than with the rest frame (3×10^{-4} against 5×10^{-3}), and the sequence is 26% longer. On the four Truebones rigs of §4.3 neither condition wins: zero-training jitter ratios 0.95 / 6.89 / 1.54 / 2.94 against the rest frame's 1.92 / 6.24 / 0.74 / 4.71, and after adaptation 0.24 / 0.74 / 0.70 / 0.17 against 0.27 / 0.92 / 0.62 / 0.23. Because the demonstration clip is an unrelated motion of the same rig, everything it carries beyond the body plan is already stated by the caption, and the

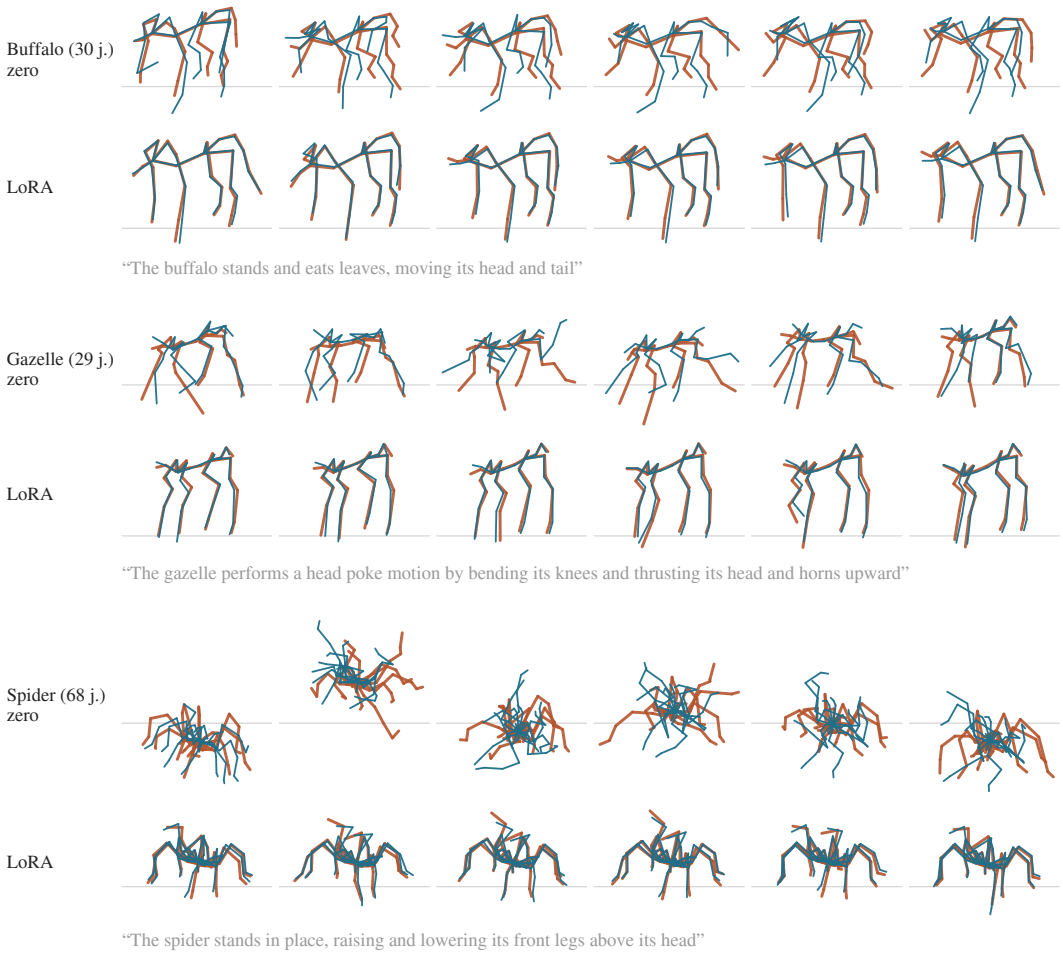

Figure 5: Truebones arms of Table 3: one held-out clip per rig, six frames spread evenly over the clip, three-quarter view; the quoted sentence is the conditioning prompt. In every panel the directly predicted joint positions (clay) are overlaid by the forward kinematics of the predicted rotations (teal); a visible separation is the projected FK–pose discrepancy (a single view can hide a disagreement in depth). Rows: the library model with the rig's statistics only (*zero*, the unadapted snapshot, from which every column of that table's *zero* block is read) and the per-rig adapter (*LoRA*). Frames are anchored on the root horizontally and keep their height above the floor; the grey line marks the floor height beneath the root. Both rows of a rig are drawn at the same scale. The clip shown is the one of that rig whose adapted sample is closest to the held-out motion; Table 3 reports all four rigs over all of their held-out clips.

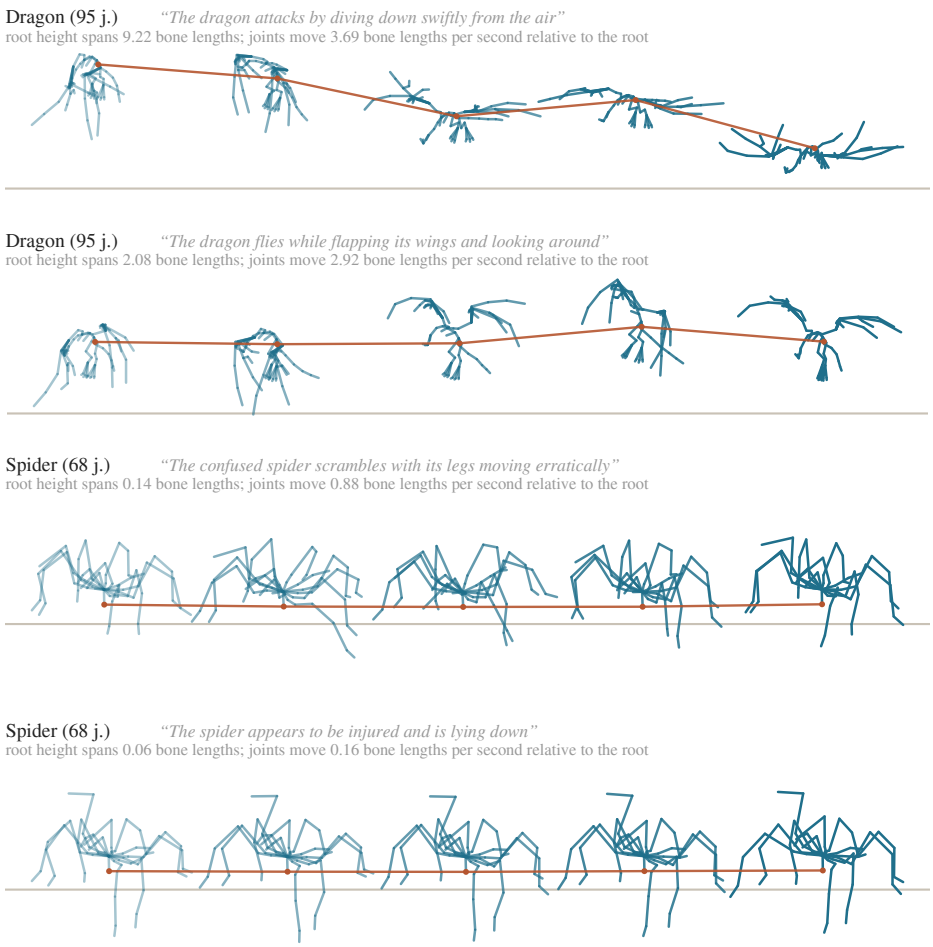

Figure 6: Two prompts on one rig, in world coordinates. Five frames at equal times of one held-out clip (light first, dark last) with the per-rig adapter of Table 3. Every frame keeps the height the model produced; the horizontal placement is the panel's own time grid, the clay line is the root's height profile through the five frames and the grey line is the floor beneath the root. Skeletons are the deployable output: the forward kinematics of the predicted rotations. The quoted sentence is the conditioning prompt; the two rows of a rig differ in nothing else and are drawn at one scale.

network learns to ignore it; a moving demonstration also has to exist at test time, which a rest pose always does. We therefore keep the rest frame.

O VALIDATION SPLIT AND EVALUATOR

Evaluator. The two towers of §4.1 share a 512-d embedding space. The text tower is a frozen DistilBERT with a trainable pooling head; the motion tower is graph-aware, with 6 graph layers and 4 temporal layers over the 16 TOPX-17 channels other than the contact flag; the objective is a symmetric InfoNCE over real clips and their captions. As a control, scoring the validation clips against shuffled captions gives R@1 0.035 in pools of 32, against a chance level of 0.031.

Split. The library is split per rig: 5% of each rig's clips (at least one, none for a rig with a single clip) are held out for validation, so that every rig with enough clips is represented in both parts. The split is fixed before any model is trained and is shared by the evaluator and by every generator and adapter in this paper; checkpoints are selected on the validation loss. Validation holds 3,899 clips over 311 rigs and training 73,995 clips. Retrieval pools are formed by chunking the validation set in dataset order (rigs sorted by name), so a pool of 64 holds clips of one rig or of neighbouring rigs; the last 59 clips do not fill a pool and are excluded from the R-precision but not from the FID or the matching score. The five sampling seeds averaged in Table 1 are 7, 13, 23, 31 and 42.

Pool size. Retrieval among more candidates is harder for the real clips and for the generated ones alike, and the pool size is a free parameter of the protocol. Table 19 re-scores the same generated samples at four pool sizes; the ranking of the two models does not depend on the choice, while the gap to the real clips widens with the pool (from 0.028 at 16 to 0.044 at 128 for the 303M model); the main text uses 64.

Table 19: R@1 of the same samples (20 steps, s=2, seed 42) re-scored in pools of 16, 32, 64 and 128 formed by chunking the dataset order; the incomplete last pool is dropped. The real clips are scored by the same evaluator in the same pools.

Pool size	16	32	64	128
Pools	243	121	60	30
Real clips (reference)	0.939	0.920	0.904	0.899
TOPX (pilot, 36M)	0.893	0.864	0.840	0.830
TOPX (303M)	0.911	0.884	0.864	0.855