# Kirin: Animal Motion Generation from In-the-Wild Video

Brian Nlong Zhao[1⋆], Zhuoyang Pan[2⋆],
James M. Rehg[1], Jiajun Wu[3], and Shangzhe Wu[4]

[1] University of Illinois Urbana-Champaign
[2] University of Pennsylvania
[3] Stanford University
[4] University of Cambridge


**Fig. 1:** Our framework takes an animal image and a text description of the desired motion as input, and generates a 3D animated mesh sequence of the animal. This is achieved by a animal motion generation model trained using 3D motion sequences extracted from large-scale in-the-wild video, in conjunction with an automatic rigging process to produce a fully animated mesh.

**Abstract.** Understanding animal motion is fundamental to modeling animal behavior and biomechanics, yet progress in this area lags far behind human motion research due to the scarcity of high-quality motion data. While human motion can be captured in controlled environments, it is impractical for most animal species, resulting in small, domain-limited datasets that restrict downstream applications such as animation. To address this challenge, we introduce Kirin, a framework that reconstructs motion from video, learns motion priors at scale, and generates realistic motion that can be directly applied to animated assets. Using large collections of in-the-wild animal videos, we reconstruct 3D motion sequences and pair them with captions to create AiM3D, the first large-scale dataset offering aligned video-text-motion tuples for quadruped animals. Building on this dataset, we develop a visual-guided motion generation model

⋆ Equal contribution.

that conditions on both text and image to guide the generation of realistic motion across diverse animal species. Finally, by leveraging an off-the-shelf image-to-3D model, we automatically rig and animate 3D meshes using generated motion, producing ready-to-render animated animals. Together, our dataset and framework establish a new foundation for large-scale, text and image conditioned animal motion generation and animation. Project page: https://kirin-ani.github.io/.

## 1 Introduction

> *"Elsewhere we have investigated in detail the movement of animals... there remains an investigation of the common ground of any sort of animal movement whatsoever."*
>
> Aristotle, On the Motion of Animals

Understanding how animals move in their natural habitats is a fundamental scientific challenge, with implications for studying behavior in biology and ecology, and for building more generalizable motion models in computer vision. Yet, compared to human motion, animal motion modeling remains vastly underexplored, primarily due to the lack of large-scale, high-quality motion data. For humans, motion capture technologies have led to large-scale precise 3D motion datasets, enabling rapid progress in 3D human modeling, animation, and generation. However, it is challenging to capture animal motion in controlled laboratory settings at scale, and impractical for wild or endangered species. This data scarcity has become the main bottleneck preventing progress in animal motion research.

One common solution is to rely on human crafted motion data. Previous efforts such as DeformingThings4D [19] and Truebones Zoo [39] include animal motions manually created by computer graphics artists, while AniMo [41] extracts motion sequences from video games that are similarly authored by human designers and animators. Although these datasets provide high quality and physically consistent motion, they are inherently constrained to a limited set of predefined actions such as walking, eating, or sleeping, and therefore fail to capture the full diversity and natural variability of animal behaviors observed in the wild.

On the other hand, the Internet offers an abundance of in-the-wild animal footage capturing diverse species, behaviors, and environments, far beyond what controlled datasets can provide. Meanwhile, recent advances in 3D reconstruction and shape estimation from monocular images and videos have made it possible to recover 3D structures from unstructured video data. Together, these developments open a new opportunity: *Can we learn realistic, generalizable models of animal motion directly from in-the-wild videos?*

Several prior works have explored this direction and made notable progress. Ponymation [37] extends MagicPony [44] to learn an *unconditional* generative model of 3D motion from videos of a single horse category. AiM [54] presents a

large scale dataset of animal videos from the Internet spanning 23 quadruped categories, but does not attempt to learn a generative model of their underlying motions.

In this paper, we present Kirin, a comprehensive framework for learning and generating 3D quadruped motion directly from large-scale video data. Our pipeline begins by enhancing existing 3D reconstruction methods to recover accurate and temporally smooth motion sequences from videos. Leveraging the AiM dataset, we develop a SMAL-based [58] 3D motion reconstruction system that produces consistent and realistic motion trajectories. Each reconstructed sequence is further paired with descriptive textual annotations generated by VLMs, resulting in the first large-scale dataset containing aligned text–video–motion tuples for quadruped animals.

To fully exploit the visual and semantic cues in this dataset, we introduce a novel adaptation of MDM [38], which yields an animal motion generation model capable of conditioning on both text and visual input. Unlike prior approaches that rely on manually designed or synthetic motion data, our method learns directly from in-the-wild videos, capturing a broader and more diverse spectrum of natural animal motion. Experimental results show that our dataset and generation model achieve state-of-the-art performance on both our test set and external out-of-distribution test sets, establishing a scalable foundation for data-driven animal motion modeling.

Furthermore, by leveraging off-the-shelf image-to-3D generation tools [52], Kirin can automatically rig the generated 3D mesh and apply the generated motion to produce realistic animated 3D mesh sequences. Comparisons show that our animation method produces more plausible 3D motion sequences compared to baseline approaches, while being more efficient. In summary, our work's contributions are as follows:

- We enhance state-of-the-art 3D animal pose reconstruction methods for video-based reconstruction and create AiM3D dataset, a large-scale animal motion dataset with aligned text, video, and motion data.
- We propose the first animal motion generation model, conditioning on both text and image input.
- Experiments demonstrate that training on our dataset with visual conditioning achieves state-of-the-art results on both in-distribution and external out-of-distribution test sets, highlighting the effectiveness of our dataset and model.
- In conjunction with a text-to-3D model, we present a fully automatic system, Kirin, that turns a 2D image into a realistic 3D animated mesh sequence, outperforming existing 4D animation methods.

## 2 Related Works

### 2.1 Animal Motion Datasets

A major obstacle in animal motion generation is the lack of high quality motion data. Unlike humans, whose movements can be captured in controlled labora-

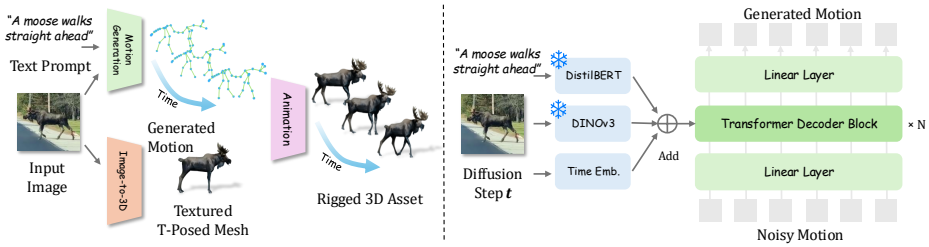


**Fig. 2: Left: Overview of Kirin generation pipeline.** A text description and an image are provided as inputs. The text and image are used for motion generation, while the image is also used to generate a T-posed mesh. The animation module then rigs the generated motion onto the mesh to produce the final animated 3D model. **Right: Overview of the motion generation architecture.** Text features are extracted using a frozen DistilBERT encoder, and image features are extracted using a frozen DINOv3 encoder. The text, image, and denoising step embeddings are combined and fed into a transformer decoder with cross-attention to generate motion sequences.

tory environments, most animal species cannot be brought into motion capture facilities and rarely behave naturally under such conditions. Their wide variety of anatomies and behaviors also makes standardized capture difficult.

Existing motion capture datasets collected in controlled settings [10, 33, 59] cover only a few species and offer limited ecological validity. Other efforts rely on artist created or human adapted motion data [19, 39, 41, 50], which are expensive to produce and introduce a significant domain gap.

Learning animal motion directly from videos is promising, but current data remains insufficient. BADJA [5] has only 11 sequences with 3D pose. AnimalKingdom [22] provides single image pose without motion. COP3D [35] contains mostly orbit shots of stationary pets and is limited to common categories such as cats and dogs. APT-36K [49] offers only fifteen frame clips, providing limited motion. Even the largest dataset, AiM [54], contains nearly 30k videos but still lacks 3D motion.

To address these limitations, we build upon AiM by reconstructing SMAL based 3D pose from video to obtain high quality motion sequences. We also use a vision language model to generate multiple behavior aware text descriptions for each clip, resulting in about 30k motion sequences paired with about 180k captions. To the best of our knowledge, this is the first large scale animal motion dataset with aligned video, motion, and descriptions for diverse animal species.

### 2.2 3D Animal Reconstruction

Many works estimate 3D or 4D animal shape and pose from single images or videos, using either model-based or model-free approaches, and either feed-forward or optimization-based pipelines.

Model-based methods such as ABM [2], SMAL [58], and their extensions [3–5, 17, 29, 30, 42, 55–57] optimize a parameterized mesh to fit 2D labels such as masks or keypoints. These optimization-based methods often overfit to 2D

projections and may yield unnatural 3D geometry. More recent systems [21,31] leverage SMAL to create image–3D paired data and enable feed-forward inference, reducing 3D inconsistencies with only minor loss in per-frame projection accuracy.

Model-free methods [1,9,14,15,18,23,43,44,51] learn shape directly from large datasets and provide flexible feed-forward predictions, but typically lack explicit skeletal structures, which limits their suitability for reconstructing articulated motion. Similarly, generic 3D and 4D reconstruction frameworks [26,28,47,48] can recover animal shape but do not supply consistent skeleton definitions.

Given these limitations, and since our goal is accurate skeletal motion rather than perfect shape, we adopt AniMer [21] as initialization, which uses SMAL model trained on 3D dataset, providing universal skeleton and at the same time avoiding unnatural 3D poses.

### 2.3 Animal Motion Generation

Although large scale data for animal motion generation remains limited, several recent works have begun exploring this direction. OmniMotionGPT [50] compensates for data scarcity by combining human motion prior with small human crafted animal motion datasets to transfer motion knowledge. AniMo [41] trains a two stage RVQ [16] model on artist made motion sequences extracted from video games to produce plausible synthetic motion.

Other efforts target different settings. SinMDM [24] learns motion motifs from a single motion example using a diffusion model with a restricted receptive field, but it does not generalize beyond the given exemplar. Puppeteer [36] and MotionAvatar [53] animate auto rigged meshes by matching or applying generated motion, yet both rely on synthetic motion sources rather than real world video.

The closest work to ours is Ponymation [37], which collects online horse videos and reconstructs 3D motion using the MagicPony [44] pipeline. However, their VAE based [12] model is unconditional and limited to a single species. In contrast, our work builds on Animal in Motion [54] to construct a large scale dataset spanning twenty three quadruped categories with paired videos and descriptive text. While Ponymation uses images only for textured mesh inference, our model conditions motion generation directly on both text and images. We adopt MDM [38] as our backbone and extend it with an image conditioning branch to learn motion synthesis grounded in visual context.

## 3 Method

Our method, Kirin, consists of three components: (1) given an animal video dataset, we first reconstruct motion from each clip to build a large-scale animal motion dataset; (2) using this dataset, we train a motion generation model; and (3) leveraging an image-to-3D module and an auto-rigging pipeline, we generate an animated 3D mesh based on user-provided text and image inputs. See Fig. 2 for our generation framework overview.

### 3.1 Extracting Animal Motions from Web Videos

Our goal is to recover accurate, temporally coherent 4D animal motions from *in-the-wild* web videos. To improve optimization stability and reconstruction quality, we decouple global translation estimation from local pose reconstruction, which allows the model to focus on optimizing fine-grained articulated motion without being affected by global displacement or noisy motion cues present in uncontrolled video data. Starting from a large collection of animal clips in the AiM dataset [54], we re-estimate per-video 3D body pose using a new SMAL-templated pipeline designed for robust projection alignment and efficient refinement. SMAL is a general parametric template for quadrupeds with *shape* coefficients $\boldsymbol{\beta} \in \mathbb{R}^{41}$, *pose* parameters $\boldsymbol{\theta} \in \mathbb{R}^{35\times 3}$, and a skinned mesh articulated by $J$=35 joints. We leverage AniMer [21] for per-frame initialization and perform sequence-level refinement to enforce tighter keypoint alignment and temporal smoothness. Global translation is estimated separately using SpatialTrackerV2 [45], an off-the-shelf 3D point tracking model, and is later combined with the optimized pose sequence to recover the complete global 4D motion.

**Initialization**. Each video in the dataset contains a single animal per frame by construction. For each frame $\mathbf{I}_t$, we use precomputed instance masks $\mathbf{M}_t$ from Grounded-SAM2 [25, 27] and 2D keypoints $\mathbf{P}_t$ from ViTPose++ [46]. We obtain per-frame initial estimates using Animer [21]: $\{(\boldsymbol{\beta}_t^{(0)}, \boldsymbol{\theta}_t^{(0)}, \boldsymbol{\pi}_t^{(0)})\}_{t=1}^{T}$, with $\boldsymbol{\pi}_t^{(0)} = (\mathbf{K}, \mathbf{E}_t)$, where: (i) $\boldsymbol{\beta}_t^{(0)} \in \mathbb{R}^{41}$ are shape coefficients of SMAL, (ii) $\boldsymbol{\theta}_t^{(0)} \in \mathbb{R}^{35\times 3}$ are joint poses in axis–angle (35 joints), and (iii) $\boldsymbol{\pi}_t^{(0)} = (\mathbf{K}, \mathbf{E}_t)$ is a weak-perspective camera with fixed intrinsics $\mathbf{K} = \mathrm{diag}(f, f, 1)$ ($f$=1000) and translation $\mathbf{E}_t \in \mathbb{R}^3$. We initialize a sequence-shared shape $\boldsymbol{\beta}^{(0)} = \frac{1}{T}\sum_t \boldsymbol{\beta}_t^{(0)}$, convert poses to 6D $\mathbf{r}_t^{(0)} \in \mathbb{R}^{35\times 6}$, and keep cameras fixed.

**Objective**. We optimize the sequence-shared shape $\boldsymbol{\beta}$ and the per-frame 6D joint rotations $\{\mathbf{r}_t\}_{t=1}^{T}$, while keeping the cameras fixed. We convert the 6D rotations to rotation matrices $\mathbf{R}_t \in \mathrm{SO}(3)^{35}$ using the standard 6D-to-SO(3) mapping $\rho(\cdot)$, where $\mathbf{R}_t = \rho(\mathbf{r}_t)$. The optimization objective is:

$$
\begin{aligned}
\min_{\boldsymbol{\beta},\{\mathbf{r}_t\}} \quad & \mathcal{L} = \lambda_{\text{proj}}\mathcal{L}_{\text{proj}} + \lambda_{\text{smooth}}\mathcal{L}_{\text{smooth}} \\
\mathcal{L}_{\text{proj}} &= \sum_{t=1}^{T} \frac{\left\|(\hat{\mathbf{k}}_t - \mathbf{k}_t) \odot \mathbf{w}_t\right\|_1}{\sum \mathbf{w}_t + \varepsilon}, \\
\mathcal{L}_{\text{smooth}} &= \sum_{t=2}^{T} d^2_{\mathrm{SO}(3)}(\mathbf{R}_{t-1}, \mathbf{R}_t) \\
&\quad + \alpha \sum_{t=3}^{T} d^2_{\mathrm{SO}(3)}(\mathbf{R}_{t-1}^{\top}\mathbf{R}_t, \mathbf{R}_{t-2}^{\top}\mathbf{R}_{t-1}).
\end{aligned}
\tag{1}
$$

Here $\hat{\mathbf{k}}_t$ are SMAL keypoint projections aligned to 2D keypoint indices, $\mathbf{w}_t$ are confidence×visibility weights from $\mathbf{P}_t$ and $\mathbf{M}_t$, and $d_{\mathrm{SO}(3)}$ denotes the geodesic distance summed over all joints.

**Optimization details**. We optimize with Adam [11] on the sequence-shared $\boldsymbol{\beta}$ and all per-frame 6D rotations with a learning rate $\eta = 0.001$ for a total epochs $K = 20$. We evaluate projections with the known intrinsics $\mathbf{K}$ and translations $\mathbf{E}_t$, and compute visibility by sampling $\mathbf{M}_t$ at ground-truth keypoint locations to attenuate occluded or off-canvas points. We set $\lambda_{\text{data}} = 1.0, \lambda_{\text{smooth}} = 100.0, \alpha = 0.2, \epsilon = 10^{-6}$ throughout our experiments, which yields an average runtime of **<1s/frame** on an NVIDIA A40 GPU.

**Why this matters?** Sequence-level refinement converts strong but noisy frame-wise predictions into temporally stable, pixel-aligned 3D motions, which we find essential for high-quality generation (see Sec. 4.2 for comparisons with Animer [21] and 4D-Fauna [54]).

**Global Translation**. The aforementioned methods operate on center-cropped videos in which the animal remains centered, allowing the model to focus solely on skeletal pose estimation. To recover global motion, we apply an off-the-shelf 3D point tracker [45] to the original uncropped video and estimate the animal's global translation by averaging the 3D trajectories of tracked points on the body. We then determine the scaling factor of global translation by aligning the animal size in 3D tracker coordinates and SMAL joints coordinates. Specifically, we first estimate the global translation of the animal from the tracker predictions, denoted as $\mathbf{T}_{\text{tracker}}$. At each time step $t$, the global translation is computed as the mean of the tracked 3D points:

$$\mathbf{T}_{\text{tracker}}^{(t)} = \frac{1}{|\mathcal{P}|} \sum_{i=1}^{|\mathcal{P}|} \mathbf{p}_i^{(t)}, \tag{2}$$

where $\mathcal{P} = \{\mathbf{p}_i\}$ denotes the set of tracked 3D point trajectories in world coordinate. The points are initialized by uniformly sampling a $10 \times 10$ grid on the first video frame and retaining only those located within the animal mask. The 3D trajectories $\mathbf{p}i^{(t)}$ are then obtained from the tracker outputs. To improve the stability of the tracker outputs, we further post-process the estimated 3D trajectories by smoothing the camera motion, aligning the reconstructed scene with a consistent ground plane, correcting residual translation drift using tracked ground points, and applying temporal smoothing to the recovered motion before estimating the animal's global translation. To align the scale of the global translation predicted by the tracker, we further rescale $\mathbf{T}_{\text{tracker}}$ to match the physical scale of the SMAL skeleton:

$$\mathbf{T}_{\text{smal}} = s \cdot \mathbf{T}_{\text{tracker}}, \tag{3}$$

where the scale factor $s$ is defined by matching the size of the square crop in tracker coordinates ($s_{\text{tracker}}$) with the head-to-tail distance of the SMAL skeleton ($s_{\text{smal}}$). The detailed formula to determine $s$ is left in the supplementary.

**Dataset**. We extract 3D motion for all video clips available in [54], yielding 29,979 motion sequences. Following their benchmark subset, we use the 230 sequence as test split, and the rest of 29,749 sequences as training split. In addition, we leverage Gemini 2.5 Flash [6] to infer 6 textual descriptions per

video input, resulting in 179,874 textual descriptions. We show example data in the supplementary material.

**Data Validation**. Following AiM [53], we report reconstruction metrics in main paper Table 3. Here we present human validation of motions and captions for 100 samples randomly selected from the test split. We consider a motion to be satisfactory if, when viewed from all angles, the motion is physically plausible with no unnatural poses, movements, or noticeable side bending of body parts, and the motion is as smooth as observed in the original video, with no perceptible jitter. Among 100 samples, 86 reconstructions are considered satisfactory. The most common failure cases arise from pure top, front, or back-view videos of moving animals, where legs are either invisible or occluded, resulting in missing leg motion in the reconstruction. Other failure cases include extremely fast animal movements or unstable camera motion in videos, which cause jitter in the reconstruction. We evaluate captions using a scoring protocol: (0) none of the captions correctly describes the motion; (1–2) only some captions are correct; (3) all captions correctly identify the action type but contain minor errors (e.g., direction or speed); (4) all captions are correct but lack diversity; and (5) all captions are correct and cover multiple levels of detail. Among 100 samples, the average caption score is 4.62, indicating that the captions are generally accurate and diverse. None of the samples received a score of 0 or 1. 3 samples contain some captions with incorrect action (e.g., a walking video described as standing still). 7 exhibit minor errors, such as incorrect turning direction (e.g., left vs. right), often due to ambiguity in camera vs. animal perspective. The remaining 90 samples have all correct captions, among which 75 include highly diverse descriptions for the same video. We also show R-precision scores using InternVideo2 and compare with AnimalKingdom video grounding annotations. Ours shows higher scores meaning that our caption is more accurate compared to AnimalKingdom annotations. Overall, we expect our dataset to contain **86% accurate motion reconstructions and 90% correct caption samples.**

### 3.2 Image-Conditioned Motion Diffusion

For animals, text alone is often underspecified: species, breed, size, shape, coat, and viewpoint all affect feasible kinematics, yet are rarely captured in a short prompt (e.g., a "dog trotting" could be a tall greyhound or a stocky corgi). We therefore add an *image* pathway to ground motion in visible morphology and scene context, yielding shape-aware, species-aware motion. We use images rather than videos as input, since videos would inject clip-specific motion and limit generalization, whereas images are easier to obtain and let the model learn motion priors pooled across all videos.

**Architecture**. Inspired by [8], we introduce a separate branch for image conditioning on a backbone model. Building on a variation of MDM [38] with DistilBERT [32] text encoder and transformer decoder [40] architecture, we introduce an additional conditioning stream for image and fuse it with text and timestep conditions by simply adding them after linear projection to align feature dimension. An off-the-shelf frozen DINOv3 [34] image encoder produces a global image

feature $\mathbf{z}_{\text{img}} \in \mathbb{R}^{1\times d}$; DistilBERT yields text tokens $\mathbf{z}_{\text{text}} \in \mathbb{R}^{l\times d}$, and timestep embedding $\mathbf{z}_t \in \mathbb{R}^{1\times d}$. The addition is performed by broadcasting and summing $\mathbf{z}_{\text{img}}$, $\mathbf{z}_{\text{text}}$, and $\mathbf{z}_t$, resulting in a combined conditioning feature $\mathbf{z}_c \in \mathbb{R}^{l\times d}$, where $l$ denotes the number of text tokens and $d$ is the feature dimension of the transformer decoder. This is then injected at every block of transformer decoder for cross-attention with the motion sequence. All other details we follow the implementation of MDM.

**Training and sampling**. We use classifier-free guidance with *per-modality* dropout: independently drop text or image with probabilities $q_{\text{text}} = 0.2$ and $q_{\text{img}} = 0.2$ during training. Following [38], we optimize the standard noise-prediction objective for diffusion. Let $\mathbf{x}_0 \in \mathbb{R}^{T\times D}$ be the motion where $D$ is the dimension of the joint representation, $\mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$, and $\mathcal{G}_\theta$ the denoiser conditioned on $\mathbf{z}_c$:

$$\mathcal{L}_{\text{DDPM}} = \mathbb{E}_{\mathbf{x}_0, c, t, \boldsymbol{\epsilon}} \left\| \boldsymbol{\epsilon} - \mathcal{G}_\theta(\mathbf{x}_t, t, \mathbf{z}_c) \right\|_2^2. \tag{4}$$

At inference we support text-only, image-only, and text+image inputs. Guided sampling uses CFG:

$$\mathcal{G}_{\text{cfg}}(\mathbf{x}_t | t, c) = \mathcal{G}(\mathbf{x}_t | t, c) + g[\mathcal{G}(\mathbf{x}_t | t, c) - \mathcal{G}(\mathbf{x}_t | t, \varnothing)], \tag{5}$$

where $c$ is the fused text–image code and $g$ is the guidance scale. This MDM-compatible fusion yields motions that respect the textual action while conforming to the animal morphology provided by the image.

### 3.3 Applying Motions to Generated Assets

To animate the generated motions on a 3D asset, we reconstruct a textured mesh from a single reference image using an image-to-3D model (Rodin [52]), yielding a T-pose mesh $\mathcal{M} = \{\mathbf{V}, \mathbf{F}\}$. We then retarget the generated motion to this asset in three steps: (i) SMAL template fitting to recover the asset's mesh-specific shape and bind-pose offsets; (ii) skinning-weight transfer from the fitted SMAL to $\mathcal{M}$; and (iii) linear blend skinning (LBS) to deform $\mathcal{M}$ per frame with the generated joint transforms.

**SMAL template fitting**. The generated T-pose asset is not guaranteed to match SMAL's canonical rest pose. We therefore estimate joint rotations that align SMAL to the asset and recover mesh-specific shape. Let $\mathbf{V}_{\text{smal}}(\boldsymbol{\beta}, \boldsymbol{\theta}_{\text{bind}})$ be SMAL vertices posed by per-joint rotations $\boldsymbol{\theta}_{\text{bind}}$. We solve

$$\min_{\boldsymbol{\beta},\, \boldsymbol{\theta}_{\text{bind}}} \; \lambda_{\text{ch}}\, D_{\text{ch}}\big(\mathbf{V}_{\text{smal}}(\boldsymbol{\beta}, \boldsymbol{\theta}_{\text{bind}}),\, \mathbf{V}\big) \;+\; \lambda_{\text{e}}\, E_{\text{edge}}, \tag{6}$$

where $D_{\text{ch}}$ is bidirectional Chamfer and $E_{\text{edge}}$ preserves edge lengths on the SMAL topology. This yields mesh-specific parameters $\hat{\boldsymbol{\beta}}$ and per-joint "bind" rotations $\hat{\boldsymbol{\theta}}_{\text{bind}}$. The latter define the inverse-bind corrections $\mathbf{B}_j := \text{FK}_j(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}}_{\text{bind}})$ used for animation, where $\text{FK}_j(\boldsymbol{\beta}, \boldsymbol{\theta}) \in SE(3)$ denotes the global forward-kinematics transform of joint $j$ under SMAL.

**Skinning transfer**. SMAL provides LBS weights $\mathbf{W}_{\text{smal}} \in \mathbb{R}^{N_{\text{smal}} \times J}$. For each asset vertex $\mathbf{v} \in \mathbf{V}$ we compute weights by $k$-NN interpolation from fitted SMAL vertices $\tilde{\mathbf{V}}$:

$$\begin{aligned} \mathbf{w}(\mathbf{v}) &= \sum_{i \in \mathcal{N}_k} \hat{\alpha}_i \, \mathbf{W}_{\text{smal}}[i, :], \\ \text{where } \hat{\alpha}_i &= \frac{\alpha_i}{\sum_{j \in \mathcal{N}_k} \alpha_j}, \quad \alpha_i = \frac{1}{\|\mathbf{v} - \tilde{\mathbf{v}}_i\|_2^2 + \varepsilon}. \end{aligned} \tag{7}$$

We choose $k = 10, \epsilon = 1e-8$ in all of our experiments.

**Animation**. For each frame $t$, MDM codes $\mathbf{x}_t$ are converted to target joints $\hat{\mathbf{J}}_t$ and we fit SMAL parameters $(\boldsymbol{\beta}', \boldsymbol{\theta}_t)$ by joint matching. With transferred skinning weights $\mathbf{w}_j(\mathbf{v})$, we use standard LBS in homogeneous form. Let $\tilde{\mathbf{v}} = [\mathbf{v}; 1]$, $\mathbf{G}_{j,t} = \text{FK}_j(\boldsymbol{\beta}', \boldsymbol{\theta}_t) \in SE(3)$ be the per-frame global joint transform, and $\mathbf{B}_j = \text{FK}_j(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}}_{\text{bind}})$ the bind transform. Then

$$\tilde{\mathbf{v}}_t = \sum_{j=1}^{J} \mathbf{w}_j(\mathbf{v}) \left( \mathbf{G}_{j,t} \, \mathbf{B}_j^{-1} \right) \tilde{\mathbf{v}}. \tag{8}$$

This yields textured, rigged assets driven by our synthesized animal motions.

## 4 Experiments

**Table 1: Comparison with baselines on AiM3D test set.** Methods are evaluated using metrics from [7], with top results in **red** (best) and **blue** (second-best). We report each metric's average and 95% confidence interval, based on 10 evaluations.

| Methods | R-Precision ↑ | | | FID ↓ | MM-Dist ↓ | Diversity ↑ | MModality ↑ |
|---|---|---|---|---|---|---|---|
| | Top-1 | Top-2 | Top-3 | | | | |
| Ground-Truth | $0.113_{\pm.002}$ | $0.206_{\pm.003}$ | $0.281_{\pm.003}$ | $0.004_{\pm.001}$ | $5.517_{\pm.008}$ | $3.934_{\pm.076}$ | - |
| AniMo (*AniMo4D [41] data*) | $0.029_{\pm.001}$ | $0.059_{\pm.001}$ | $0.089_{\pm.001}$ | $30.043_{\pm.086}$ | $8.564_{\pm.007}$ | $3.713_{\pm.055}$ | $3.723_{\pm.009}$ |
| AniMo (*AiM3D data*) | $0.029_{\pm.001}$ | $0.058_{\pm.001}$ | $0.088_{\pm.002}$ | $30.516_{\pm.072}$ | $8.659_{\pm.007}$ | $3.800_{\pm.060}$ | $3.720_{\pm.008}$ |
| Kirin (ours, *text*) | **0.032** $_{\pm.001}$ | **0.065** $_{\pm.001}$ | **0.097** $_{\pm.001}$ | **11.889** $_{\pm.024}$ | **7.007** $_{\pm.003}$ | **4.693** $_{\pm.071}$ | **4.609** $_{\pm.011}$ |
| Kirin (ours, *image*) | - | - | - | $25.089_{\pm.055}$ | - | $1.761_{\pm.034}$ | $5.804_{\pm.029}$ |
| Kirin (ours, *text + image*) | **0.043** $_{\pm.001}$ | **0.087** $_{\pm.001}$ | **0.130** $_{\pm.002}$ | **6.248** $_{\pm.014}$ | **6.218** $_{\pm.004}$ | **4.319** $_{\pm.093}$ | **4.272** $_{\pm.017}$ |

**Table 2: Comparison with baselines on AnimalML3D [50] data.** Best results are shown in **red**.

| Methods | R-Precision ↑ | | | FID ↓ | MM-Dist ↓ | Diversity ↑ | MModality ↑ |
|---|---|---|---|---|---|---|---|
| | Top-1 | Top-2 | Top-3 | | | | |
| Ground-Truth | $0.964_{\pm.028}$ | $1.000_{\pm.000}$ | $1.000_{\pm.000}$ | $0.235_{\pm.080}$ | $1.709_{\pm.084}$ | $11.778_{\pm.874}$ | - |
| AniMo (*AniMo4D [41] data*) | $0.031_{\pm.001}$ | $0.062_{\pm.001}$ | $0.093_{\pm.001}$ | $150.671_{\pm.0.913}$ | $17.171_{\pm.052}$ | $5.152_{\pm.126}$ | **5.159** $_{\pm.023}$ |
| AniMo (*AiM3D data*) | $0.031_{\pm.001}$ | $0.061_{\pm.001}$ | $0.093_{\pm.001}$ | $155.687_{\pm0.739}$ | $12.334_{\pm.069}$ | $4.571_{\pm.077}$ | $4.579_{\pm.015}$ |
| Kirin (ours) | **0.039** $_{\pm.001}$ | **0.076** $_{\pm.001}$ | **0.111** $_{\pm.001}$ | **138.145** $_{\pm.725}$ | **12.128** $_{\pm.003}$ | **5.152** $_{\pm.057}$ | $5.126_{\pm.020}$ |

*Baselines.* **Motion generation:** We compare against against AniMo [41], which is, to our knowledge, the only publicly available method tailored for *text-driven* animal motion generation. We compare with AniMo trained on ours dataset and

on their original AniMo4D dataset. We report our text-only and *text+image* setting to show that visual conditioning further stabilizes pose and style. **Mesh animation:** We qualitative compare with Puppeteer [36], a recent pipeline that also accepts text and image inputs to produce animated animal meshes. For a fair comparison, we use the same text, image, and generated T-posed mesh inputs for both methods, and follow Puppeteer's setup by employing Kling AI [13] as the video generation module.

*Datasets.* For motion evaluation, we evaluate on two animal motion datasets. We first show results on AiM3D, which is our reconstructed motion dataset using videos from AiM [54]. We use their cleaned benchmark data as test split, which has 230 motions, and we exclude them from training set. The same benchmark is also used for quantitatively evaluate motion reconstruction. Following [41], we also show results on AnimalML3D [50], which is an external out-of-distribution test set that none of the methods has trained on. AnimalML3D contains 1,260 hand-crafted motions curated from DeformingThings4D [19].

*Metrics.* For motion generation, we follow standard text-to-motion evaluation [7]: (1) **R-Precision**: text-motion retrieval accuracy in top-k accuracies; (2) **FID**: Fréchet distance between generated and real motion distributions; (3) **MM-Dist**: distance in a shared text-motion latent space; (4) **Diversity**: average pairwise distance between independently sampled motions; (5) **Multimodality**: variance among multiple motions generated from the same text.

For motion reconstruction, we follow the metrics described in [54]: For evaluating 4D animal reconstruction, we use the following metrics: (1) **Silhouette IoU**: IoU between the rendered silhouette and the ground-truth mask; (2) **PCK**: percentage of projected keypoints that fall within a normalized distance threshold; (3) **KT**: PCK error after a 2D-to-3D-to-2D re-projection to a novel view, used as a proxy for 3D shape consistency; (4) **MPJVE**: average error between predicted and ground-truth joint velocities in projected pixel space to evaluate temporal motion.

### 4.1 Quantitative Results on Motion Generation

For each text in the evaluation dataset, we generate 10 motion samples and repeat the evaluation over 10 trials. We report the mean and 95% confidence interval across trials, as shown in Tab. 1. We evaluate four configurations: the baseline model trained on its original data AniMo4D [41], the baseline retrained on our dataset, and our model in two variants—(1) trained with both image and text conditioning, and (2) trained and evaluated with the image branch disabled (text-only).

As shown in Tab. 1, both of our variants outperform the baseline across all metrics. Retraining the baseline on our dataset already leads to a improvement, demonstrating the quality of text and motion quality of our newly constructed dataset. The text-only version of our model further surpasses the baseline trained

on the same data, indicating that our architecture and method models the motion more effectively. Incorporating image conditioning yields additional gains on most metrics, achieving the best overall fidelity and text–motion alignment, producing motions that are both semantically coherent and visually plausible. Together, these results validate the effectiveness of both our dataset and our model design, highlighting the benefit of integrating textual and visual signals for animal motion generation.

To further demonstrate the generality and quality of our dataset, we evaluate on an external test-only dataset, AnimalML3D [50], where neither our models nor the baselines have been trained. We compare our methods trained on our dataset against AniMo trained on AniMo4D [41]. As shown in Tab. 2, our models outperform the baseline across all metrics, confirming that training on our dataset leads to better generalization and higher-quality motion modeling, even on unseen data.

### 4.2 Quantitative Results on Reconstruction

**Table 3: Benchmark comparison of 4D animal reconstruction methods.** Our method maintains nearly real-time processing speeds comparable to feed-forward approaches [20, 21], while achieving improved performance across all evaluation metrics.

| Method | IoU ↑ | PCK@0.1 ↑ | PCK@0.05 ↑ | KT-PCK@0.1 ↑ | KT-PCK@0.05 ↑ | MPJVE ↓ | Time ↓ |
|---|---|---|---|---|---|---|---|
| SMALify [4] | 0.867 | 0.954 | 0.787 | 0.623 | 0.372 | 0.023 | ∼ 30s/frame |
| 4D-Fauna [54] | 0.814 | 0.664 | 0.317 | 0.418 | 0.193 | 0.044 | ∼ 30s/frame |
| AniMer [21] | 0.677 | 0.537 | 0.199 | 0.566 | 0.256 | 0.038 | < 1s/frame |
| 3D-Fauna [20] | 0.670 | 0.470 | 0.177 | 0.329 | 0.130 | 0.058 | < 1s/frame |
| Kirin (ours) | 0.698 | 0.751 | 0.485 | 0.614 | 0.332 | 0.037 | < 1s/frame |

Following [54], we evaluate our motion reconstruction method on their benchmark, with results shown in Tab. 3. Among the baselines, AniMer [21] and 3D-Fauna [20] are feed-forward methods that operate in near real time, while SMALify [4] and 4D-Fauna [54] are optimization-based approaches that require significantly longer computation time to fit each sequence. Although our method includes a post-optimization step after feed-forward inference, it still achieves processing speeds comparable to feed-forward baselines, while simultaneously improving both projection-based and 3D-aware metrics. These results demonstrate that our reconstruction method is both accurate and efficient, making it suitable for large-scale data processing.

### 4.3 Qualitative Results

We qualitatively compare our motion generation results with AniMo [41] as the baseline and observe several notable failure cases in the baseline outputs. As shown in Fig. 3, in the first example, the baseline fails to follow the text input and produces an unrealistic walking motion, while our method successfully generates a coherent walking sequence that naturally turns left. In the second example, the baseline misinterprets the text prompt “rearing up,” generating a lying-down

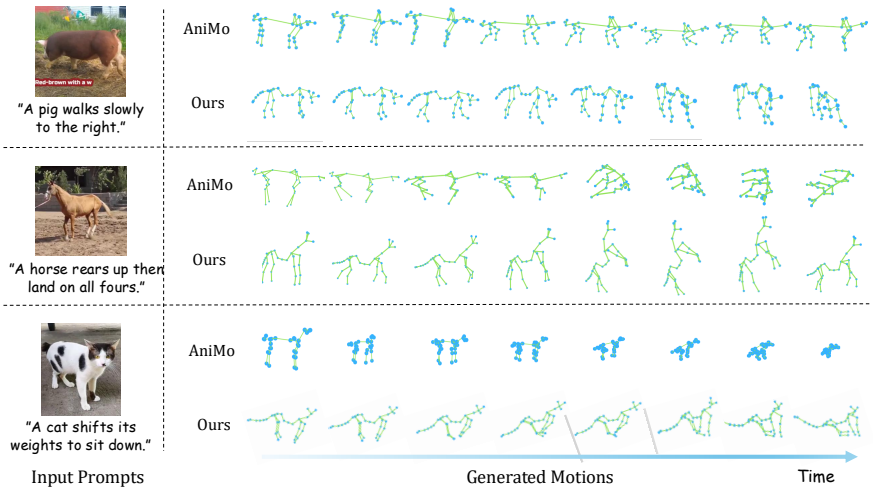


**Fig. 3: Visual comparison of generated skeletal motion with the baseline.** The left columns show the input text and image. AniMo uses only text, while our method conditions on both text and image. AniMo exhibits failures such as motions that do not follow the prompt and inconsistent skeleton shapes, whereas our method produces more realistic motion sequences.

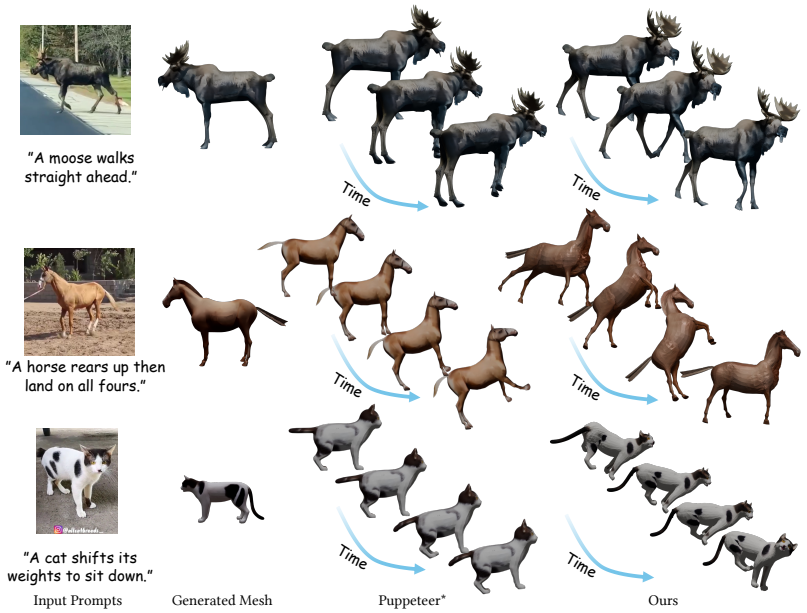


**Fig. 4: Visual comparison of generated mesh animation with the baseline.** The left columns show the input text and image, which are used for both pipelines. Puppeteer often produces little or no motion on the input mesh, whereas our method generates realistic movements that follow the text prompt.

motion instead, whereas our model correctly produces a smooth rearing-up and landing motion for the horse. A particularly critical failure of the baseline is its inconsistent skeleton structure across frames: as illustrated in the third example, although the cat roughly follows the "sitting" prompt, its skeleton collapses and shrinks to an unrealistic size, making the motion unusable for downstream tasks such as mesh rigging. In contrast, our method maintains consistent bone lengths and body proportions, resulting in stable and anatomically plausible motions that better align with the input text descriptions.

Qualitative results comparing with Puppeteer [36] show that our method produces more accurate and natural motions. While Puppeteer relies on optimizing a rigged mesh to match frames generated by a video generation model, this indirect approach often fails when the generated videos are inconsistent or physically implausible. In particular, we observe frequent failure cases where the generated video does not follow the input text, deviates from the initial frame, or exhibits abrupt shot changes and inconsistent animal appearance and shapes. These issues make motion extraction unreliable.

In contrast, our method learns motion directly from real-world animal videos, reconstructing temporally smooth 3D joint trajectories that generalize robustly to new inputs. As a result, rigging and animating a mesh using our generated motion is consistently more stable and faithful to the input text and image. This demonstrates that learning motion from real video data provides a more reliable and physically grounded foundation than relying on motion cues extracted from generated videos, validating the effectiveness of our pipeline design.

### 4.4 Ablation Studies

We conduct an ablation study to evaluate the effect of image conditioning in our motion generation model. As shown in Tab. 1, we compare models trained with and without the image-conditioning branch while keeping all other training configurations identical.

Adding image conditioning leads to consistent improvements across key perceptual and alignment metrics: R-Precision, FID, and MM-Distance all show notable gains, with FID exhibiting the largest improvement. This indicates that integrating visual cues helps the model generate more realistic and visually coherent motions that align better with both text and ground-truth motion distributions. The improvement in R-Precision and MM-Distance also confirms that image conditioning enhances semantic alignment between text and motion, suggesting that visual features provide an effective bridge between the two modalities. On the other hand, Diversity and Multimodality scores remain comparable but showing a marginal decrease when image features are used. We attribute this to the nature of visual conditioning, which introduces stronger spatial and appearance constraints that guide the motion synthesis process more tightly toward visually plausible outcomes, potentially reducing motion variation.

## 5 Conclusion

We introduce Kirin, a framework for learning and generating realistic 3D animal motion from in-the-wild videos. To address the challenge of data scarcity, we construct AiM3D, the first large-scale animal dataset containing aligned video, text, and 3D motion tuples. Building on this dataset, we develop an animal motion generation model that conditions on both text and image. Experiments show that our model achieves state-of-the-art performance on both in-distribution and external test sets. Finally, we present an animation pipeline that rigs the generated motions onto 3D meshes. Together, our reconstruction method and dataset, generative model, and rigging pipeline form a unified solution for data-driven animal motion modeling and animation. We believe this work opens new opportunities for biomechanics, behavioral analysis, and realistic character animation for media. All code and datasets will be released.

## Acknowledgments

This work is in part supported by NSF RI #2211258 and #2338203, ONR MURI N00014-22-1-2740, and ONR MURI N00014-24-1-2748.

## References


1. Agudo, A.: Safari from visual signals: Recovering volumetric 3d shapes. In: ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2495–2499 (2022). https://doi.org/10.1109/ICASSP43922.2022.9746343 5
2. Badger, M., Wang, Y., Modh, A., Perkes, A., Kolotouros, N., Pfrommer, B.G., Schmidt, M.F., Daniilidis, K.: 3d bird reconstruction: a dataset, model, and shape recovery from a single view. In: European conference on computer vision. pp. 1–17. Springer (2020) 4
3. Baieri, D., Cicciarella, R., Krützen, M., Rodolà, E., Zuffi, S.: Model-based metric 3d shape and motion reconstruction of wild bottlenose dolphins in drone-shot videos. arXiv preprint arXiv:2504.15782 (2025) 4
4. Biggs, B., Boyne, O., Charles, J., Fitzgibbon, A., Cipolla, R.: Who left the dogs out? 3d animal reconstruction with expectation maximization in the loop. In: European Conference on Computer Vision. pp. 195–211. Springer (2020) 4, 12
5. Biggs, B., Roddick, T., Fitzgibbon, A., Cipolla, R.: Creatures great and SMAL: Recovering the shape and motion of animals from video. In: ACCV (2018) 4
6. Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., et al.: Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261 (2025) 7
7. Guo, C., Zou, S., Zuo, X., Wang, S., Ji, W., Li, X., Cheng, L.: Generating diverse and natural 3d human motions from text. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5152–5161 (June 2022) 10, 11
8. Huang, H.P., Zhou, Y., Wang, J.H., Liu, D., Liu, F., Yang, M.H., Xu, Z.: Move-in-2d: 2d-conditioned human motion generation. arXiv preprint arXiv:2412.13185 (2024) 8
9. Kanazawa, A., Tulsiani, S., Efros, A.A., Malik, J.: Learning category-specific mesh reconstruction from image collections. In: Proceedings of the European conference on computer vision (ECCV). pp. 371–386 (2018) 5
10. Kearney, S., Li, W., Parsons, M., Kim, K.I., Cosker, D.: Rgbd-dog: Predicting canine pose from rgbd sensors. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8336–8345 (2020) 4
11. Kingma, D.P.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 7
12. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013) 5
13. Kling AI: Kling ai: Next-generation ai creative studio. https://klingai.com/global/ (2024), accessed: November 11, 2025 11
14. Kulkarni, N., Gupta, A., Fouhey, D.F., Tulsiani, S.: Articulation-aware canonical surface mapping. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 452–461 (2020) 5
15. Kulkarni, N., Gupta, A., Tulsiani, S.: Canonical surface mapping via geometric cycle consistency. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2202–2211 (2019) 5
16. Lee, D., Kim, C., Kim, S., Cho, M., Han, W.S.: Autoregressive image generation using residual quantization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11523–11532 (2022) 5

17. Li, C., Lee, G.H.: Coarse-to-fine animal pose and shape estimation. Advances in Neural Information Processing Systems **34**, 11757–11768 (2021) 4
18. Li, X., Liu, S., Kim, K., De Mello, S., Jampani, V., Yang, M.H., Kautz, J.: Self-supervised single-view 3d reconstruction via semantic consistency. In: European Conference on Computer Vision. pp. 677–693. Springer (2020) 5
19. Li, Y., Takehara, H., Taketomi, T., Zheng, B., Nießner, M.: 4dcomplete: Non-rigid motion estimation beyond the observable surface. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12706–12716 (2021) 2, 4, 11
20. Li, Z., Litvak, D., Li, R., Zhang, Y., Jakab, T., Rupprecht, C., Wu, S., Vedaldi, A., Wu, J.: Learning the 3d fauna of the web. arXiv preprint arXiv:2401.02400 (2024) 12
21. Lyu, J., Zhu, T., Gu, Y., Lin, L., Cheng, P., Liu, Y., Tang, X., An, L.: Animer: Animal pose and shape estimation using family aware transformer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 17486–17496 (2025) 5, 6, 7, 12
22. Ng, X.L., Ong, K.E., Zheng, Q., Ni, Y., Yeo, S.Y., Liu, J.: Animal kingdom: A large and diverse dataset for animal behavior understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19023–19034 (June 2022) 4
23. de Paco, S.M., Agudo, A.: 4dpv: 4d pet from videos by coarse-to-fine non-rigid radiance fields. In: Proceedings of the Asian Conference on Computer Vision. pp. 2596–2612 (2024) 5
24. Raab, S., Leibovitch, I., Tevet, G., Arar, M., Bermano, A.H., Cohen-Or, D.: Single motion diffusion. In: The Twelfth International Conference on Learning Representations (ICLR) (2024), https://openreview.net/pdf?id=DrhZneqz4n 5
25. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: Sam 2: Segment anything in images and videos (2024), https://arxiv.org/abs/2408.00714 6
26. Ren, J., Xie, C., Mirzaei, A., Kreis, K., Liu, Z., Torralba, A., Fidler, S., Kim, S.W., Ling, H., et al.: L4gm: Large 4d gaussian reconstruction model. Advances in Neural Information Processing Systems **37**, 56828–56858 (2024) 5
27. Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., Zeng, Z., Zhang, H., Li, F., Yang, J., Li, H., Jiang, Q., Zhang, L.: Grounded sam: Assembling open-world models for diverse visual tasks (2024) 6
28. Ren*, Z., Zhao*, X., Schwing, A.G.: Class-agnostic reconstruction of dynamic objects from videos. In: Neural Information Processing Systems (NeurIPS) (2021), * equal contribution 5
29. Rueegg, N., Zuffi, S., Schindler, K., Black, M.J.: Barc: Learning to regress 3d dog shape from images by exploiting breed information. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3876–3884 (2022) 4
30. Rüegg, N., Tripathi, S., Schindler, K., Black, M.J., Zuffi, S.: BITE: Beyond priors for improved three-D dog pose estimation. In: IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 8867–8876 (Jun 2023) 4
31. Sabathier, R., Mitra, N.J., Novotny, D.: Animal avatars: Reconstructing animatable 3d animals from casual videos. In: European Conference on Computer Vision. pp. 270–287. Springer (2024) 5
32. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108 (2019) 8

33. Shooter, M., Malleson, C., Hilton, A.: Benchmarking monocular 3d dog pose estimation using in-the-wild motion capture data. arXiv preprint arXiv:2406.14412 (2024) 4
34. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., et al.: Dinov3. arXiv preprint arXiv:2508.10104 (2025) 8
35. Sinha, S., Shapovalov, R., Reizenstein, J., Rocco, I., Neverova, N., Vedaldi, A., Novotny, D.: Common pets in 3d: Dynamic new-view synthesis of real-life deformable categories. CVPR (2023) 4
36. Song, C., Li, X., Yang, F., Xu, Z., Wei, J., Liu, F., Feng, J., Lin, G., Zhang, J.: Puppeteer: Rig and animate your 3d models. Advances in Neural Information Processing Systems (2025) 5, 11, 14
37. Sun, K., Litvak, D., Zhang, Y., Li, H., Wu, J., Wu, S.: Ponymation: Learning articulated 3d animal motions from unlabeled online videos. In: European Conference on Computer Vision. pp. 100–119. Springer (2024) 2, 5
38. Tevet, G., Raab, S., Gordon, B., Shafir, Y., Cohen-or, D., Bermano, A.H.: Human motion diffusion model. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=SJ1kSyO2jwu 3, 5, 8, 9
39. Truebones: Free fbx/.bvh zoo. over 75 animals with animations and textures. https://truebones.gumroad.com/l/skZMC/ (2022), accessed: 2024-07-01 2, 4
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 8
41. Wang, X., Ruan, K., Zhang, X., Wang, G.: Animo: Species-aware model for text-driven animal motion generation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 1929–1939 (2025) 2, 4, 5, 10, 11, 12
42. Wang, Y., Kolotouros, N., Daniilidis, K., Badger, M.: Birds of a feather: Capturing avian shape models from images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14739–14749 (2021) 4
43. Wu, S., Jakab, T., Rupprecht, C., Vedaldi, A.: DOVE: Learning deformable 3d objects by watching videos. IJCV (2023) 5
44. Wu, S., Li, R., Jakab, T., Rupprecht, C., Vedaldi, A.: Magicpony: Learning articulated 3d animals in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8792–8802 (2023) 2, 5
45. Xiao, Y., Wang, J., Xue, N., Karaev, N., Makarov, I., Kang, B., Zhu, X., Bao, H., Shen, Y., Zhou, X.: Spatialtrackerv2: 3d point tracking made easy. In: ICCV (2025) 6, 7
46. Xu, Y., Zhang, J., Zhang, Q., Tao, D.: Vitpose++: Vision transformer for generic body pose estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(2), 1212–1230 (2023) 6
47. Yang, G., Sun, D., Jampani, V., Vlasic, D., Cole, F., Liu, C., Ramanan, D.: Viser: Video-specific surface embeddings for articulated 3d shape reconstruction. In: NeurIPS (2021) 5
48. Yang, G., Vo, M., Neverova, N., Ramanan, D., Vedaldi, A., Joo, H.: Banmo: Building animatable 3d neural models from many casual videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2863–2873 (2022) 5
49. Yang, Y., Yang, J., Xu, Y., Zhang, J., Lan, L., Tao, D.: Apt-36k: A large-scale benchmark for animal pose estimation and tracking. Advances in Neural Information Processing Systems **35**, 17301–17313 (2022) 4

50. Yang, Z., Zhou, M., Shan, M., Wen, B., Xuan, Z., Hill, M., Bai, J., Qi, G.J., Wang, Y.: Omnimotiongpt: Animal motion generation with limited data (2023) 4, 5, 10, 11, 12
51. Yao, C.H., Hung, W.C., Li, Y., Rubinstein, M., Yang, M.H., Jampani, V.: Lassie: Learning articulated shapes from sparse image ensemble via 3d part discovery. Advances in Neural Information Processing Systems **35**, 15296–15308 (2022) 5
52. Zhang, L., Wang, Z., Zhang, Q., Qiu, Q., Pang, A., Jiang, H., Yang, W., Xu, L., Yu, J.: Clay: A controllable large-scale generative model for creating high-quality 3d assets. ACM Transactions on Graphics (TOG) **43**(4), 1–20 (2024) 3, 9
53. Zhang, Z., Wang, Y., Wu, B., Chen, S., Zhang, Z., Huang, S., Zhang, W., Fang, M., Chen, L., Zhao, Y.: Motion avatar: Generate human and animal avatars with arbitrary motion. arXiv preprint arXiv:2405.11286 (2024) 5
54. Zhao, B.N., Wu, J., Wu, S.: Web-scale collection of video data for 4d animal reconstruction. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track (2025) 2, 4, 5, 6, 7, 11, 12
55. Zhong, S., Peng, J., Zheng, Z., Huang, Z., Ma, W., Zhang, G., Liu, Q., Yuille, A., Chen, J.: 4d-animal: Freely reconstructing animatable 3d animals from videos. arXiv preprint arXiv:2507.10437 (2025) 4
56. Zuffi, S., Kanazawa, A., Berger-Wolf, T., Black, M.J.: Three-d safari: Learning to estimate zebra pose, shape, and texture from images" in the wild". In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5359–5368 (2019) 4
57. Zuffi, S., Kanazawa, A., Black, M.J.: Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3955–3963 (2018). https://doi.org/10.1109/CVPR.2018.00416 4
58. Zuffi, S., Kanazawa, A., Jacobs, D.W., Black, M.J.: 3d menagerie: Modeling the 3d shape and pose of animals. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6365–6373 (2017) 3, 4
59. Zuffi, S., Mellbin, Y., Li, C., Hoeschle, M., Kjellström, H., Polikovsky, S., Hernlund, E., Black, M.J.: VAREN: Very accurate and realistic equine network. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2024) 4