Learning the 3D Fauna of the Web

Zizhang Li[1*] Dor Litvak[1,2*] Ruining Li[3] Yunzhi Zhang[1] Tomas Jakab[3] Christian Rupprecht[3]
Shangzhe Wu[1†] Andrea Vedaldi[3†] Jiajun Wu[1†]

[1]Stanford University [2]UT Austin [3]University of Oxford

kyleleey.github.io/3DFauna/

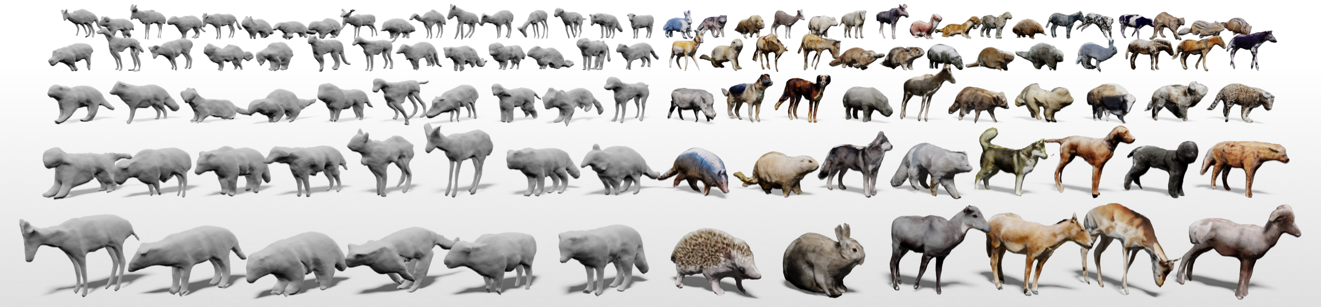

Figure 1. **Learning Diverse 3D Animals from the Internet.** Our method, *3D-Fauna*, learns a pan-category deformable 3D model of more than 100 different animal species using only 2D Internet images as training data. At test time, the model can turn a single image of an quadruped instance into an articulated, textured 3D mesh in a feed-forward manner, ready for animation and rendering.

Abstract

Learning 3D models of all animals in nature requires massively scaling up existing solutions. With this ultimate goal in mind, we develop 3D-Fauna, an approach that learns a pan-category deformable 3D animal model for more than 100 animal species jointly. One crucial bottleneck of modeling animals is the limited availability of training data, which we overcome by learning our model from 2D Internet images. We show that prior approaches, which are category-specific, fail to generalize to rare species with limited training images. We address this challenge by introducing the Semantic Bank of Skinned Models (SBSM), which automatically discovers a small set of base animal shapes by combining geometric inductive priors with semantic knowledge implicitly captured by an off-the-shelf self-supervised feature extractor. To train such a model, we also contribute a new large-scale dataset of diverse animal species. At inference time, given a single image of any quadruped animal, our model reconstructs an articulated 3D mesh in a feed-forward manner in seconds.

1. Introduction

Computer vision models can nowadays reconstruct humans in monocular images and videos robustly and accurately, recovering their 3D shape, articulated pose, and even appearance [3, 11, 12, 14, 21, 35]. However, humans are but a tiny fraction of the animals that exist in nature, and 3D models remain essentially blind to the vast majority of biodiversity.

While in principle the same approaches that work for humans could work for many other animal species, in practice scaling it to each of the 2.1 million different animal species on Earth is nearly hopeless. In fact, building a human model such as SMPL [35] and a corresponding pose predictor [3, 14] requires collecting 3D scans of many people in laboratory [21], crafting a corresponding articulated deformable model semi-automatically, and collecting extensive manual labels to train corresponding pose regressors. Of all animals, only humans are currently of sufficient importance in applications to justify the costs.

A technically harder but much more practical approach is to learn animal models automatically from images and videos readily available on the Internet. Several authors have demonstrated that at least rough models can be learned from such uncontrolled image collections [22, 63, 74]. Even so, many limitations remain, starting from the fact that these methods can only reconstruct one or a few specific animal exemplars [74], or at most a single class of animals at a given time [22, 63]. The latter restriction is particularly glaring, as it defeats the purpose of using the Internet as a vast data source for modeling biodiversity.

We introduce *3D-Fauna*, a method that learns a pan-

[*] Equal contribution
[†] Equal advising

category deformable model for a large number (> 100) of different quadruped animal species, such as dogs, antelopes, and hedgehogs, as shown in Fig. 1. For the approach to be as automated and thus as scalable as possible, we assume that *only* Internet images of the animals are provided as training data and only consider as prerequisites a pre-trained 2D object segmentation model and off-the-shelf unsupervised visual features. 3D-Fauna is designed as a feed-forward network that deforms and poses the deformable model to reconstruct any animal given a single image as input. The ability to perform monocular reconstruction is necessary for training on (single-view) Internet images, and is also useful in many real-world applications.

Crucial to 3D-Fauna is to learn a *single joint model* of *all animals* in one go. Despite posing a challenge, modeling many animals jointly is essential for reconstructing rarer species, for which we often have only a small number of images to train on. This allows us to exploit the structural similarity of different animals that results from evolution, and maximize statistical efficiency. Here, we focus our attention on animals that share a given body plan, in particular, quadrupeds, and share the structure of the underlying skeletal model, which would otherwise be difficult to pin down.

Learning such a model from only unlabeled single-view images requires several technical innovations. The most important is to develop a 3D representation that is sufficiently *expressive* to model the diverse shape variations of the animals, and at the same time *tight* enough to be learned from single-view images without overfitting individual views. Prior work partly achieved this goal by using skinned models, which consider small shape variations around a base template followed by articulation [63]. We found that this approach does not provide sufficient inductive biases to learn *diverse* animal species from Internet images alone. Hence, we introduce the *Semantic Bank of Skinned Models* (SBSM), which uses off-the-shelf unsupervised features, such as DINO [5, 41], to hypothesize how different animals may relate semantically, and automatically learns a low-dimensional base shape bank.

Lastly, Internet images, which are not captured with the purpose of 3D reconstruction in mind, are characterized by a strong photographer bias, skewing the viewpoint distribution to mostly frontal, which significantly hinders the stability of 3D shape learning. To mitigate this issue, 3D-Fauna further encourages the predicted shapes to look realistic from all viewpoints, by introducing an efficient mask discriminator that enforces the silhouettes rendered from a *random* viewpoint to stay within the distribution of the silhouettes of the real images.

Combining these ideas, 3D-Fauna is an end-to-end framework that learns a pan-category model of 3D quadruped animals from online image collections. To train 3D-Fauna, we collected a large-scale animal dataset of over 100 quadruped species, dubbed the *Fauna Dataset*, as part of the contribution. After training, the model can turn a single test image of any quadruped instance into a fully articulated 3D mesh in a feed-forward fashion, ready for animation and rendering. Extensive quantitative and qualitative comparisons demonstrate significant improvements over existing methods. Code and data will be released.

2. Related Work

Optimization-Based 3D Reconstruction of Animals. Due to the lack of explicit 3D data for the vast majority of animals, reconstruction has mostly relied on pre-defined shape models or multi-view images. Initially, efforts focus on fitting a parametric 3D shape model obtained form 3D scans, e.g., SMAL [80], to animal images using annotated 2D keypoints and segmentation masks, which is further extended to multi-view images [81]. Other works aim to optimize the 3D shape [6, 58, 69–71, 74–76] directly from image or video collections of a smaller scale using various forms of supervision in addition to masks, such as keypoints [6, 58], self-supervised semantic correspondences [74–76], optical flow [68–71], surface normals [71], category-specific template shapes [6, 58].

Learning 3D from Internet Images and Videos. Recently, authors have attempted to learn 3D priors from Internet images and videos at a larger scale [1, 13, 20, 22, 29, 30, 55, 60–63, 77], mostly focusing on a single category at a time. Reconstructing animals presents additional challenges due to their highly deformable nature, which often necessitates stronger supervisory signals for training, similar to the ones used in optimization-based methods. Some methods have, in particular, learned to model articulated animals, such as horses, from single-view image collections without any 3D supervision, adopting a hierarchical shape model that factorizes a category-specific prior shape from instance-specific shape deformation and articulation [20, 62, 63]. However, these models are trained in a category-specific manner and fail to generalize to less common animal species as shown in Sec. 5.3.

Attempts to model diverse animal species again resort to pre-defined shape models, e.g., SMAL. Ruegg et al. [44, 45] model multiple dog breeds and regularize the learning process by encouraging intra-breed similarities using a triplet loss, which requires breed labels for training, in addition to keypoint annotations and template shape models. In contrast, our approach reconstructs a significantly broader set of animals and is trained in a category-agnostic fashion, without relying on existing 3D shape models or keypoints. Another related work [19] aims to learn a category-agnostic 3D shape regressor by exploiting pre-trained CLIP features and an off-the-shelf normal estimator, but does not model deformation and produces coarse shapes. Concurrent work

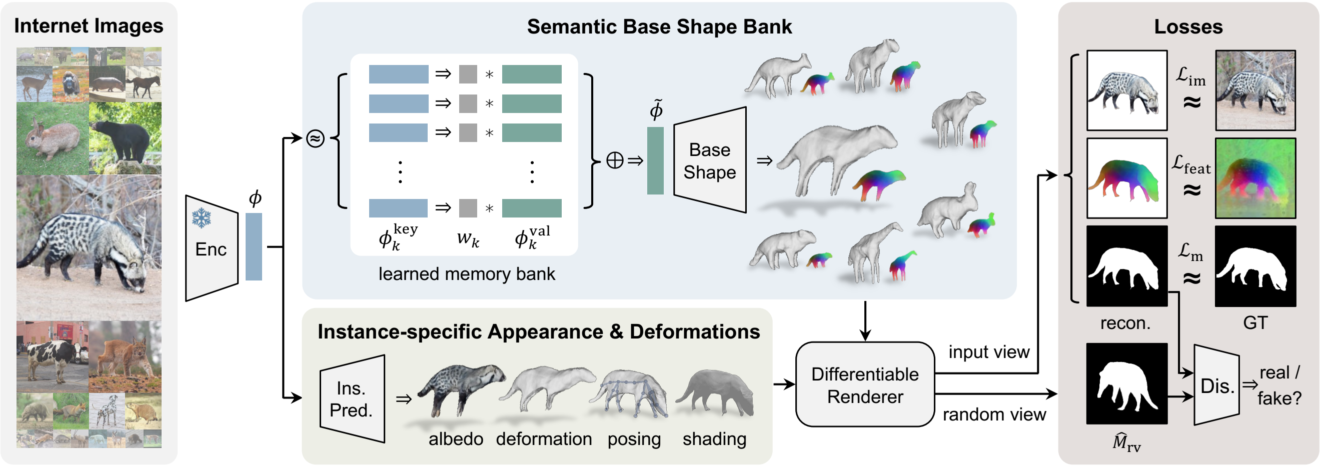

Figure 2. **Training Pipeline.** 3D-Fauna is trained using only single-view images from the Internet. Given each input image, it first extracts a feature vector ϕ using a pre-trained unsupervised image encoder [5]. This is then used to query a learned memory bank to produce a base shape and a DINO feature field in the canonical pose. The model also predicts the albedo, instance-specific deformation, articulated pose and lighting, and is trained via image reconstruction losses on RGB, DINO feature map and mask, as well as a mask discriminator loss.

SAOR [2] also trains one model to reconstruct diverse animal categories, but obtains less realistic results and tends to suffer from strong photographer bias.

Another line of research attempts to distill 3D reconstructions from 2D generative models trained on large-scale datasets of Internet images, which can be GAN-based [7, 8, 15, 39] or more recently, diffusion-based models [9, 18, 36, 50] using Score Distillation Sampling [42] and its variants. This idea has been extended to learn image-conditional multi-view generator networks [26, 31–34, 43, 47, 51, 52, 59, 67, 72]. However, most of these methods optimize one single shape at a time, whereas our model learns a pan-category deformable model that can reconstruct any animal instance in a feed-forward fashion.

Animal Datasets. Learning 3D models often requires high-quality images without blur or occlusion. Existing high-quality datasets were only collected for a small number of categories [49, 57, 62, 70], and more diverse datasets [38, 65, 66, 73] often contain many noisy images unsuitable for training off the shelf. To train our pan-category model for a wide range of quadruped animal species, we aggregate these existing datasets after substantial filtering, and additionally source more images from the Internet to create a large-scale object-centric image dataset spanning over 100 quadruped species, as detailed in Sec. 4.

3. Method

Our goal is to learn a deformable model of a large variety of different animals using only Internet images for supervision. Formally, we learn a function $f : I \mapsto O$ that maps any image $I \in \mathbb{R}^{3\times H\times W}$ of an animal to a corresponding 3D reconstruction O, capturing the animal's shape, deformation and appearance.

3D reconstruction is greatly facilitated by using multi-view data [17], but this is not available at scale, or at all, for most animals. Instead, we wish to reconstruct animals from weak single-view supervision obtained from the Internet. Compared to prior works [63, 74–76], which focused on reconstructing a single animal type at a time, here we target a large number of animal species at once, which is significantly more difficult. We show in the next section how solving this problem requires carefully exploiting the semantic similarities and geometric correspondences between different animals to regularize their 3D geometry.

3.1. Semantic Bank of Skinned Models

Given an image I, consider the problem of estimating the 3D shape (V, F) of the animal contained in it, where $V \in \mathbb{R}^{K\times 3}$ is a list of vertices of a 3D mesh with face connectivity given by triplets $F \subset \{1, \dots, K\}^3$. While recovering a 3D shape from a single image is ill-posed, as we train the model f on a large dataset, we can ultimately observe animals from a variety of viewpoints. However, different images show different animals with different 3D shapes. Non-Rigid Structure-from-Motion [4, 53, 54] shows that reconstruction is still possible, but only if one makes the space of possible 3D shapes sufficiently *tight* to remove the reconstruction ambiguity. At the same time, the space must be sufficiently *expressive* to capture all animals.

Skinned Models (SM). Following SMPL [35], many works [20, 62, 63, 71] have adopted a Skinned Model (SM) to model the shape of deformable objects when learning from single-view image collections or videos. An SM starts from a base shape V_{base} of the object (e.g., human or animal) at 'rest', applies as a *small* deformation $V_{\text{ins}} = f_{\text{ins}}(V_{\text{base}}, \phi)$ to capture instance-specific details, and then applies a larger

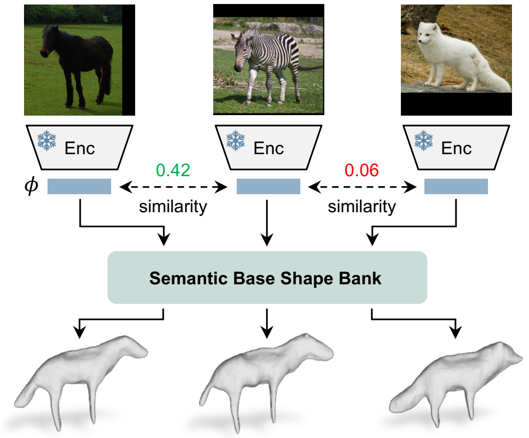

Figure 3. **Queries from the Semantic Base Shape Bank.** Without requiring any category labels, the Semantic Bank (Sec 3.1) automatically learns diverse base shapes for various animals and preserves the semantic similarities across different instances.

deformation via a skinning function $V = f_{\text{pose}}(V_{\text{ins}}, \phi)$, controlled by the articulation of the underlying skeleton. We assume that deformations are predicted by neural networks that receive as input image features $\phi = f_\phi(I)$ extracted from a powerful self-supervised image encoder.

In our case, a single SM is insufficient to capture the very large shape variations between different animals, which include horses, dogs, antelopes, hedgehogs, etc. Naïvely attempting to capture this diversity using the network f_{ins} means that the resulting deformations *cannot be small* any longer, which throws off the tightness of the model.

Semantic Bank of Skinned Models. In order to increase the expressiveness of the model while still avoiding overfitting individual images, we propose to exploit the fact that different animals often have similar 3D shapes as a result of evolution. We can thus reduce the shape variation to a small number of shape bases V_{base}, and interpolate between them.

To do so, we introduce a *Semantic Bank of Skinned Models* that automatically discovers a set of latent shape bases and learns to project each image into a linear combination of these bases. Key to this method is to use pre-trained unsupervised image features [5, 41] to automatically and implicitly identify similar animals. This is realized by means of a small memory bank with K learned key-value pairs $\{(\phi_k^{\text{key}}, \phi_k^{\text{val}})\}_{k=1}^K$. Specifically, given an image embedding ϕ, we query the memory bank to obtain a latent shape embedding $\tilde{\phi}$ as a linear combination of the value tokens $\{\phi_k^{\text{val}}\}$ via a mechanism similar to attention [56]:

$$\tilde{\phi} = \sum_{k=1}^{K} w_k \, \phi_k^{\text{val}}, \text{ where } w_k = \frac{\text{cossim}(\phi, \phi_k^{\text{key}})}{\sum_{j=1}^{K} \text{cossim}(\phi, \phi_j^{\text{key}})}, \tag{1}$$

and cossim denotes cosine similarity between two feature vectors. This embedding $\tilde{\phi}$ is then used as a condition to the base shape predictor $(V_{\text{base}}, F) = f_{\text{s}}(\tilde{\phi})$, which produces semantically-adaptive base shapes without relying on any category labels or being bound to a hard categorization.

In practice, the image features ϕ are obtained from a well-trained feature extractor like DINO-ViT [5, 41]. Defining the weights based on the cosine similarities between the image features ϕ and a small number of bases $\{\phi_k^{\text{key}}\}$ captures the semantic similarities across different animal instances. For instance, as illustrated in Fig. 3, the cosine similarity between the image features of a zebra and a horse is 0.42, whereas the similarity between a zebra and an arctic fox is only 0.06. Ablations in Fig. 6 further verify the importance of this Semantic Bank, without which the model easily overfits each training image and fails to reconstruct plausible 3D shapes.

Implementation Details. The base shape is predicted using a hybrid SDF-mesh representation [46, 63] parameterized by a coordinate MLP, with a conditioning vector $\tilde{\phi}$ injected via layer weight modulation [24, 25]. Since extracting meshes from SDFs using DMTet [46] is memory and compute intensive, in practice, we only compute it once for each iteration, by assuming the batched images contain the same animal species, and simply averaging out the embeddings $\tilde{\phi}$. The instance-specific deformation is predicted using another coordinate MLP that outputs the displacement $\Delta V_{\text{ins},i} = f_{\Delta V}(V_{\text{base},i}, \phi)$ for each vertex $V_{\text{base},i}$ of the base mesh conditioned on the image feature ϕ, resulting in the deformed shape $V_{\text{ins}} = \Delta V_{\text{ins}} + V_{\text{base}}$. We enforce a bilateral symmetry on both the base shape and the instance deformation by mirroring the query locations for the MLPs. Given the instance mesh V_{ins}, we initialize a quadrupedal skeleton using a simple heuristic [63], and predict the rigid pose $\xi_1 \in SE(3)$ and bone rotations $\xi_b \in SO(3), b = 2, \ldots, B$ using a pose network. These posing parameters are then applied to the instance mesh via a linear blend skinning equation [35]. Refer to the sup. mat. for more details.

Appearance. Assuming a Lambertian illumination model, we model the appearance of the object using an albedo field $a(\boldsymbol{x}) = f_{\text{a}}(\boldsymbol{x}, \phi) \in [0, 1]^3$ and a dominant directional light. The final shaded color of each pixel is computed as $\hat{I}(\boldsymbol{u}) = (k_a + k_d \cdot \max\{0, \langle \boldsymbol{l}, \boldsymbol{n} \rangle\}) \cdot a(\boldsymbol{x})$, where $\boldsymbol{n}$ is the normal direction of the *posed* mesh at pixel $\boldsymbol{u}$, and $k_a, k_d \in [0, 1]$ and $\boldsymbol{l} \in \mathbb{S}^2$ are respectively the ambient intensity, diffuse intensity and dominant light direction predicted by the lighting network $(k_a, k_d, \boldsymbol{l}) = f_{\text{l}}(\phi)$.

3.2. Learning Formulation

The entire pipeline is trained in an unsupervised fashion, using only self-supervised image features [5, 41] and object masks obtained from off-the-shelf segmenters [27, 28].

Reconstruction Losses. Given the final predicted posed shape V and appearance of the object, we use a differentiable renderer $\mathcal{R}$ to obtain an RGB image $\hat{I}$ as well as a mask image $\hat{M}$, which are compared to the input image I and the pseudo-ground-truth object mask M:

$$\mathcal{L}_{\text{m}} = \|\hat{M} - M\|_2^2 + \lambda_{\text{dt}}\|\hat{M} \odot \texttt{dt}(M)\|_1, \tag{2}$$

$$\mathcal{L}_{\text{im}} = \|\tilde{M} \odot (\hat{I} - I)\|_1, \tag{3}$$

where $\texttt{dt}(\cdot)$ is distance transform for more effective gradients [22, 61], $\odot$ denotes the Hadamard product, λ_{dt} specifies the balancing weight, and $\tilde{M} = \hat{M} \odot M$ is the intersection of the predicted and ground-truth masks.

Correspondences from Self-Supervised Features. Self-supervised feature extractors are notoriously good at establishing semantic correspondences between objects, which can be distilled to facilitate 3D reconstruction [63]. To do so, we extract a patch-based feature map $\Phi \in \mathbb{R}^{D\times H\times W}$ from each training image. These raw feature maps can be noisy and may preserve image-specific information irrelevant to other images. To distill more effective semantic correspondences across different images, we perform a Principal Component Analysis (PCA) across all feature maps [63], reducing the dimension to $D' = 16$. We then task the model to also learn a feature field in the canonical frame $\psi(\boldsymbol{x}, \tilde{\phi}) \in \mathbb{R}^{D'}$ that is rendered into a feature image $\hat{\Phi}$ given predicted posed shape using the same renderer $\mathcal{R}$. Training then encourages the rendered feature images $\hat{\Phi}$ to match the pre-extracted PCA features Φ': $\mathcal{L}_{\text{feat}} = \|\tilde{M} \odot (\hat{\Phi} - \Phi')\|_2^2$. Note that although the space of the PCA features Φ' is shared across different animal instances, the feature field ψ still receives the latent embedding $\tilde{\phi}$ as a condition. This is because different animals vary in shape, resulting in different feature fields.

Mask Discriminator. In practice, despite exploiting these semantic correspondences, we still find that the viewpoint prediction may easily collapse to only frontal viewpoints, due to the heavy photographer bias in Internet photos. This can lead to overly elongated shapes as shown in Fig. 6, and further deteriorates the viewpoint predictions. To mitigate this, we further encourage the shape to look realistic from arbitrary viewpoints. Specifically, we introduce a mask discriminator D that encourages the mask images $\hat{M}_{\text{rv}}$ rendered from a random viewpoint to stay within the distribution of the ground-truth masks $\mathcal{M}$. The discriminator also receives the base embedding $\tilde{\phi}$ (with gradients detached) as a condition to make this adversarial guidance tailored to specific types of animals and thus more effective. Formally, this is achieved via an adversarial loss [15]:

$$\mathcal{L}_{\text{adv}} = \mathbb{E}_{M\sim\mathcal{M}}[\log D(M;\tilde{\phi})] + \mathbb{E}_{\hat{M}_{\text{rv}}\sim\mathcal{M}_{\text{rv}}}[\log(1 - D(\hat{M}_{\text{rv}};\tilde{\phi}))]. \tag{4}$$

Note that we do not use a discriminator on the rendered RGB images, as the predicted texture is often much less realistic when compared to real images, which gives the discriminator a trivial task. Moreover, the distribution of mask images is less susceptible to viewpoint bias than RGB images, and hence we can simply sample random viewpoints uniformly, without requiring a precise viewpoint distribution of the training images.

Overall Loss. We further enforce the Eikonal constraint $\mathcal{R}_{\text{Eik}}$ on the SDF network as well as the viewpoint hypothesis loss $\mathcal{L}_{\text{hyp}}$ and the magnitude regularizers $\mathcal{R}_{\text{def}}$ on vertex deformations and $\mathcal{R}_{\text{art}}$ on articulation parameters ξ. See the supplementary materials for details.

The final training objective $\mathcal{L}$ is thus

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_{\text{hyp}}\mathcal{L}_{\text{hyp}} + \lambda_{\text{adv}}\mathcal{L}_{\text{adv}} + \mathcal{R}, \tag{5}$$

where $\mathcal{L}_{\text{rec}} = \lambda_{\text{m}}\mathcal{L}_{\text{m}} + \lambda_{\text{im}}\mathcal{L}_{\text{im}} + \lambda_{\text{feat}}\mathcal{L}_{\text{feat}}$ summarizes the three reconstruction losses, $\mathcal{R} = \lambda_{\text{Eik}}\mathcal{R}_{\text{Eik}} + \lambda_{\text{art}}\mathcal{R}_{\text{art}} + \lambda_{\text{def}}\mathcal{R}_{\text{def}}$ summarizes the regularizers, and λ's balance the contribution of each term.

Training Schedule. We design a robust training schedule that comprises three stages. First, we train the base shapes and the viewpoint network without articulation or deformation. This significantly improves the stability of the training and allows the model to roughly register the rigid pose of all instances and learn the coarse base shapes.

As the viewpoint prediction stabilizes after 20k iterations, in the second stage, we instantiate the bones and enable the articulation, allowing the shapes to gradually grow legs and fit the articulated pose in each image. Meanwhile, we also turn on the mask discriminator to prevent viewpoint collapse and shape elongation. In the final stage, we optimize the instance shape deformation field to allow the model to capture the fine-grained geometric details of individual instances, with the discriminator disabled, as it may corrupt the shape if overused.

4. Dataset Collection

In order to train this pan-category model for all types of quadruped animals, we create a new animal image dataset, dubbed the **Fauna Dataset**, spanning 128 quadruped species from dogs, antelopes to minks and platypuses, with a total of 78,168 images. We first aggregate the training sets of existing animal image datasets, including Animals-with-Attributes [65], APT-36K [73], Animal3D [66] and DOVE [62]. Many of these images are blurry or contain heavy occlusions, which will impact the stability of the training. We thus filter the images using automatic scripts first, followed by manual inspection. This results in 8,378 images covering approximately 70 animal species. To further increase the size as well as the diversity of the dataset, we additionally collect 69,790 images from the Internet, including 63,115 video frames and 2,358 images for 7 common animals (bear, cow, elephant, giraffe, horse, sheep, zebra) as well as 4,317 images for another 51 less common species. We use off-the-shelf segmentation models [27, 28] to detect and segment the instances in the images. Out of the 121 few-shot categories, we hold out 5 as novel categories unused at training. For validation, we randomly select 5 images in each of the rest 116 few-shot categories, and 2,462 images for the 7 common species. To reduce the viewpoint bias in the few-shot categories, we manually identify a few (1–10) backward-facing instances in the training set and duplicate them to match the size of the rest.

5. Experiments

5.1. Technical Details

We base our architecture on MagicPony [63], adding the new SBSM and mask discriminator. For the Semantic Bank, we use $K = 60$ key-value pairs. The dimension of keys is 384 (same as DINO-ViT) and the dimension of values is 128. As the texture network tends to struggle to predict detailed appearance in one go, partially due to limited capacity, for all the visualizations, we follow [63] and fine-tune (only) the texture network for 50 iterations, which takes < 10 seconds. Refer to the sup. mat. for further details.

5.2. Qualitative Results

After training, 3D-Fauna takes in a single test image of any quadruped animal and produces an articulated and textured 3D mesh in a feed-forward manner, as visualized in Fig. 4. The model can reconstruct very different animals, such as antelopes, armadillos, and fishers, without requiring any category labels. All the input images in Fig. 4 have not been seen during training. In particular, the model also performs well on held-out categories, e.g. the wolf in the third row.

5.3. Comparisons with Prior Work

Baselines. To the best of our knowledge, ours is the first deformable model designed to handle 100+ quadruped species, learned purely from 2D Internet data. We carry out quantitative and qualitative comparisons to methods that are at least in principle applicable to this setting. The baseline is MagicPony [63], which however is *category-specific* (they first train on horses, and fine-tune on giraffes, cows and zebras). We also compare with two popular deformable models that can work in the wild, namely UMR [30] and A-CSM [29]. However, they require weakly-supervised part segmentations and shape templates, respectively. Other works, such as LASSIE [74] and its follow-ups [75, 76], optimize a deformable model on a small set of about 20 images covering a single animal category at a time. More recently, image-to-3D methods based on distilling 2D diffusion models and/or large 3D datasets [32] have also demonstrated plausible 3D reconstructions of animals from a single image. In contrast, our model predicts an *articulated* mesh from a single image within seconds. Although it is difficult to establish a fair numerical comparison given these different settings, in Sec. 5.3, we provide a side-by-side qualitative comparison against baselines [32, 74, 75]. We use the publicly released code [32, 63, 74, 75] and report numbers [29, 30] included in MagicPony [63].

	PASCAL		APT-36K	Animal3D
	KT-PCK@0.1	PCK@0.1	PCK@0.1	PCK@0.1
UMR [30]	0.284	-	-	-
A-CSM [29]	0.329	0.687	0.649	0.822
MagicPony [63]	0.429	-	0.756	0.867
Ours	**0.539**	**0.782**	**0.841**	**0.901**

Table 1. **Quantitative Comparisons** on PASCAL VOC [10], APT-36K [73] and Animal3D [66]. When compared to baselines including the competitive MagicPony [63], our method demonstrates significantly improved performance on all datasets.

Quantitative Comparisons. We conduct quantitative evaluation across three different datasets, APT-36K [73], Animal3D [66], and PASCAL VOC [10], which contain images of various animals with 2D keypoint annotations. Following MagicPony [63], we first evaluate on horses in PASCAL VOC [10] using the widely used Keypoint Transfer metric [22, 29, 30]. We use the same protocol as in A-CSM [29] and randomly sample 20k source-target image pairs. For each source image, we project the visible vertices of the predicted mesh onto the image and map each annotated 2D keypoint to its nearest vertex. We then project that vertex to the target image and check if it lies within a small distance (10% of image size) to the corresponding keypoint in the target image. We summarize the results using the Percentage of Correct Keypoints (KT-PCK@0.1) in Tab. 1.

In Tab. 1, we follow CMR [22] to evaluate the three datasets on more species, optimizing a linear mapping from mesh vertices to desired keypoints for each category, and reporting PCK@0.1 between the predicted and annotated 2D keypoints. Our model demonstrates significant improvement over existing methods on all datasets. A performance breakdown for each category is provided in the sup. mat.

Qualitative Comparisons. Figure 5 compares 3D-Fauna qualitatively to several recent works [32, 63, 74, 75]. To establish a fair comparison with MagicPony [63], for categories demonstrated in their paper (e.g. horse), we simply run inference using the released model. For each of the other categories, we use their public code to train a per-category model on our dataset from scratch (which contains less than 100 images for some rare categories). For LASSIE [74] and Hi-LASSIE [75], which optimize over a small set of images, we train their models on the *test* image together with additional 29 images randomly selected from the training set of that category. Hi-LASSIE [75] is further

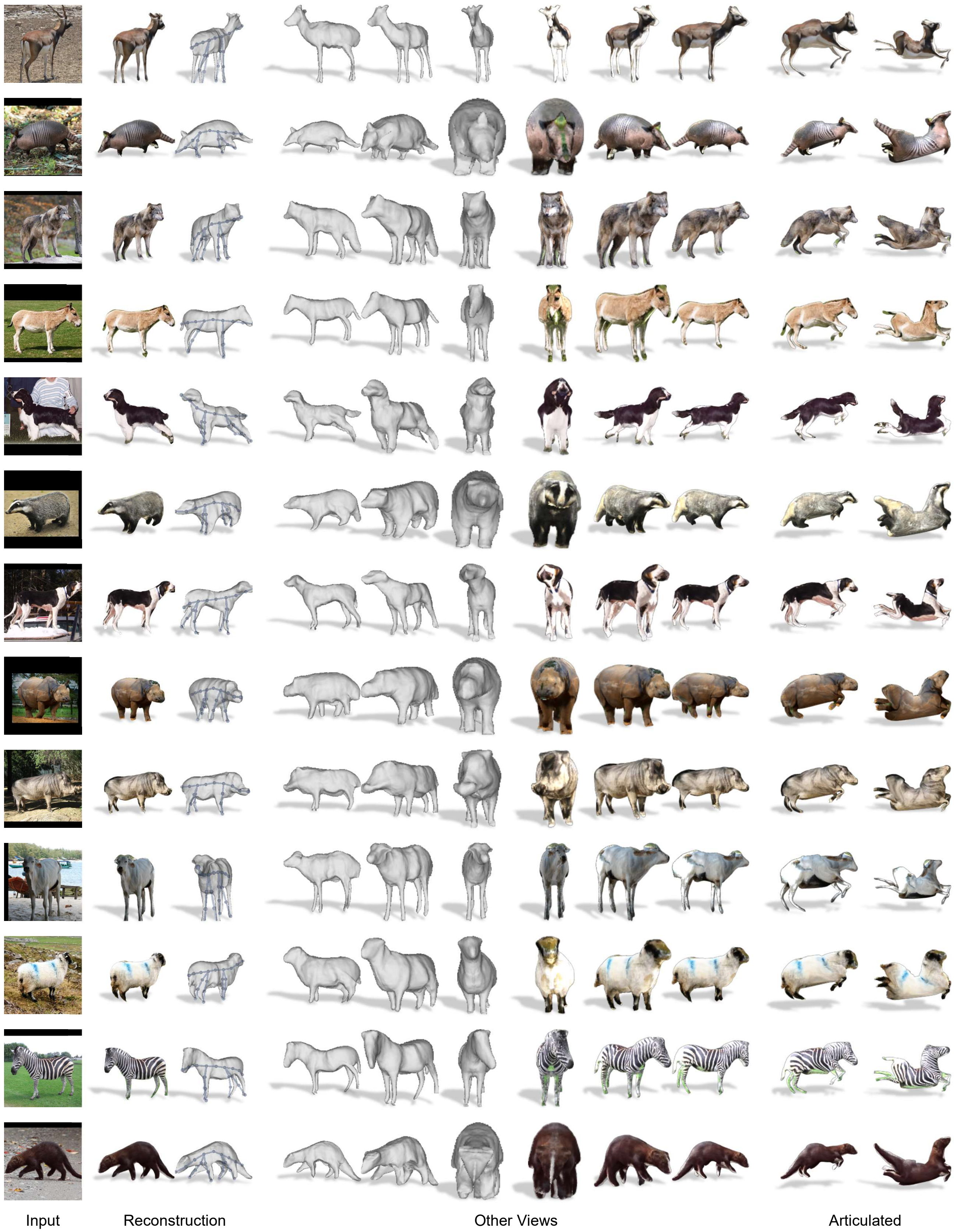

Figure 4. **Single Image 3D Reconstruction.** Given a single image of any quadruped animal at test time, our model reconstructs an articulated and textured 3D mesh in a feed-forward manner without requiring category labels, which can be readily animated.

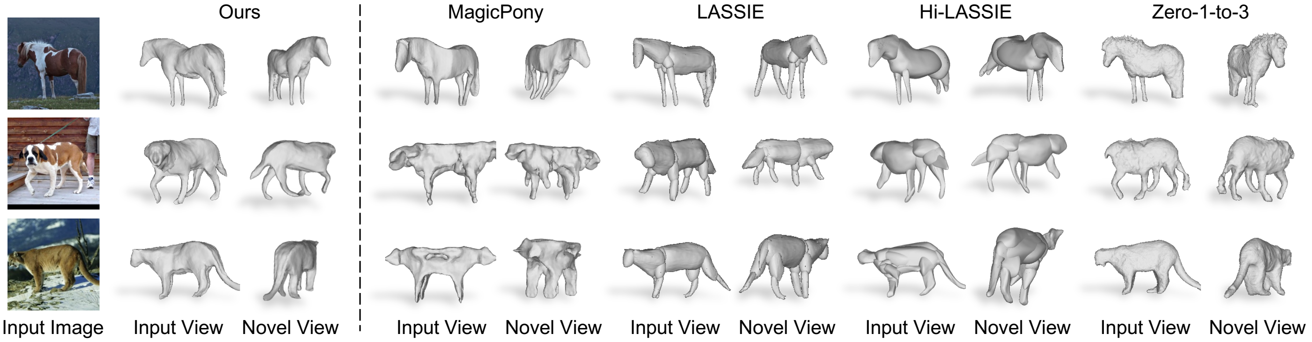

Figure 5. **Qualitative Comparisons** against MagicPony [63], LASSIE [74], Hi-LASSIE [75] and Zero-1-to-3 [32]. Compared to all baselines, our method predicts more stable poses and higher-fidelity reconstructions. Note that our method is learning-based and predicts 3D meshes in a feed-forward fashion (as opposed to [74, 75] that optimize on test images), which is orders of magnitude faster.

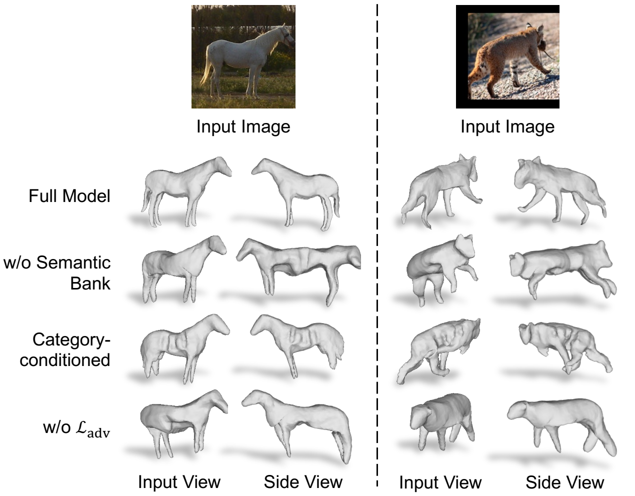

Figure 6. **Ablation Studies.** Both the Semantic Bank and the mask discriminator improve the results as discussed in Sec. 5.4.

fine-tuned on the test image after training. To compare with Zero-1-to-3 [32], we use the implementation in threestudio [16] to first distill a NeRF [37] using Score Distillation Sampling [42] given the masked test image, and then extract a 3D mesh for fair comparison. Note that our model predicts 3D meshes within seconds, whereas the optimization takes at least 10–20 mins for the other methods [32, 74, 75].

As shown in Fig. 5, MagicPony is sensitive to the size of the training set. When trained on rare categories with fewer (<100) images, such as the puma in Fig. 5, it fails to learn meaningful shapes and produces severe artifacts. Despite optimizing on the test images, LASSIE and Hi-LASSIE produce coarser reconstructions, partially due to the part-based representation that struggles in capturing the detailed geometry and articulation, as well as unstable viewpoint prediction. Zero-1-to-3, on the other hand, often fails to correctly reconstruct the legs, and does not explicitly model the articulated pose. On the contrary, our method predicts accurate viewpoint and reconstructs fine-grained articulated shapes for all different animals, with only one *single* model.

5.4. Ablation Study

In Fig. 6, we present ablation results on three key design choices in our pipeline: SBSM, category-agnostic training, and mask discriminator. If we remove the SBSM and directly condition the base shape network on each individual image embedding ϕ, the model tends to overfit each training views without learning meaningful canonical 3D shapes and pose. Alternatively, we can simply condition the base shape on an explicit (learned) category-specific embedding and train the model in a category-conditioned manner. This also leads to sub-optimal reconstructions, in particular on rare categories with few training images. Lastly, training without the mask discriminator results in biased viewpoint prediction (towards frontal) and produces elongated shapes.

6. Conclusions

We have presented 3D-Fauna, a deformable model for 100 animal categories learned using only Internet images. 3D-Fauna can reconstruct any quadruped image by instantiating in seconds a posed version of the deformable model to match the input image. Despite capable of modeling diverse animals, the current model is still limited to quadruped species that share a same skeletal structure. Furthermore, the training images still need to be lightly curated. Nevertheless, 3D-Fauna still presents a significant leap compared to prior works and moves us closer to models that will be able to understand and reconstruct all animals in nature.

Acknowledgments. We thank Cristobal Eyzaguirre, Kyle Sargent, and Yunhao Ge for their insightful discussions and Chen Geng for proofreading. The work is in part supported by the Stanford Institute for Human-Centered AI (HAI), NSF RI #2211258, #2338203, ONR MURI N00014-22-1-2740, ONR YIP N00014-24-1-2117, the Samsung Global Research Outreach (GRO) program, Amazon, Google, and EPSRC VisualAI EP/T028572/1.

References

[1] Kalyan Vasudev Alwala, Abhinav Gupta, and Shubham Tulsiani. Pre-train, self-train, distill: A simple recipe for supersizing 3d reconstruction. In *CVPR*, 2022. 2

[2] Mehmet Aygün and Oisin Mac Aodha. Saor: Single-view articulated object reconstruction. In *CVPR*, 2024. 3

[3] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *ECCV*, 2016. 1

[4] Christoph Bregler, Aaron Hertzmann, and Henning Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, 2000. 3

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 2, 3, 4

[6] Thomas J. Cashman and Andrew W. Fitzgibbon. What shape are dolphins? building 3d morphable models from 2d images. *IEEE TPAMI*, 2012. 2

[7] Eric Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-GAN: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *CVPR*, 2021. 3

[8] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *CVPR*, 2022. 3

[9] Congyue Deng, Chiyu ”Max” Jiang, Charles R. Qi, Xinchen Yan, Yin Zhou, Leonidas Guibas, and Dragomir Anguelov. Nerdi: Single-view nerf synthesis with language-guided diffusion as general image priors. In *CVPR*, 2023. 3

[10] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 2015. 6, 13

[11] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient matching of pictorial structures. In *CVPR*, 2000. 1

[12] Martin A. Fischler and Robert A. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. on Computers*, 1973. 1

[13] Shubham Goel, Angjoo Kanazawa, and Jitendra Malik. Shape and viewpoints without keypoints. In *ECCV*, 2020. 2

[14] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *ICCV*, 2023. 1

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 2014. 3, 5

[16] Yuan-Chen Guo, Ying-Tian Liu, Ruizhi Shao, Christian Laforte, Vikram Voleti, Guan Luo, Chia-Hao Chen, Zi-Xin Zou, Chen Wang, Yan-Pei Cao, and Song-Hai Zhang. threestudio: A unified framework for 3d content generation. https://github.com/threestudio-project/threestudio, 2023. 8

[17] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 3

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 3

[19] Zixuan Huang, Varun Jampani, Anh Thai, Yuanzhen Li, Stefan Stojanov, and James M Rehg. Shapeclipper: Scalable 3d shape learning from single-view images via geometric and clip-based consistency. In *CVPR*, 2023. 2

[20] Tomas Jakab, Ruining Li, Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi. Farm3d: Learning articulated 3d animals by distilling 2d diffusion. In *3DV*, 2024. 2, 3

[21] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Godisart, Bart Nabbe, Iain Matthews, et al. Panoptic studio: A massively multiview system for social interaction capture. *IEEE TPAMI*, 2019. 1

[22] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 1, 2, 5, 6

[23] Angjoo Kanazawa, Shubham Tulsiani, Alexei A Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 13

[24] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, 2020. 4, 15

[25] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *NeurIPS*, 2021. 4, 15

[26] Subin Kim, Kyungmin Lee, June Suk Choi, Jongheon Jeong, Kihyuk Sohn, and Jinwoo Shin. Collaborative score distillation for consistent visual synthesis. *arXiv preprint arXiv:2307.04787*, 2023. 3

[27] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *CVPR*, 2020. 4, 6, 16

[28] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023. 4, 6, 16

[29] Nilesh Kulkarni, Abhinav Gupta, David F Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. In *CVPR*, 2020. 2, 6

[30] Xueting Li, Sifei Liu, Kihwan Kim, Shalini De Mello, Varun Jampani, Ming-Hsuan Yang, and Jan Kautz. Self-supervised single-view 3d reconstruction via semantic consistency. In *ECCV*, 2020. 2, 6

[31] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Zexiang Xu, Hao Su, et al. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *NeurIPS*, 2023. 3

[32] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *ICCV*, 2023. 6, 8

[33] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. SyncDreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023.

[34] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023. 3

[35] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. SMPL: A skinned multi-person linear model. *ACM TOG*, 2015. 1, 3, 4

[36] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Realfusion: 360deg reconstruction of any object from a single image. In *CVPR*, 2023. 3

[37] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 8

[38] Xun Long Ng, Kian Eng Ong, Qichen Zheng, Yun Ni, Si Yong Yeo, and Jun Liu. Animal kingdom: A large and diverse dataset for animal behavior understanding. In *CVPR*, 2022. 3

[39] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. HoloGAN: Unsupervised learning of 3d representations from natural images. In *ICCV*, 2019. 3

[40] Michael Niemeyer and Andreas Geiger. GIRAFFE: Representing scenes as compositional generative neural feature fields. In *CVPR*, 2021. 15

[41] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 4, 14, 16

[42] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *ICLR*, 2023. 3, 8

[43] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023. 3

[44] Nadine Rüegg, Silvia Zuffi, Konrad Schindler, and Michael J Black. Barc: Learning to regress 3d dog shape from images by exploiting breed information. In *CVPR*, 2022. 2

[45] Nadine Rüegg, Shashank Tripathi, Konrad Schindler, Michael J Black, and Silvia Zuffi. Bite: Beyond priors for improved three-d dog pose estimation. In *CVPR*, 2023. 2

[46] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. *NeurIPS*, 2021. 4

[47] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 3

[48] Yawar Siddiqui, Justus Thies, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Texturify: Generating textures on 3d shape surfaces. In *ECCV*, 2022. 14

[49] Samarth Sinha, Roman Shapovalov, Jeremy Reizenstein, Ignacio Rocco, Natalia Neverova, Andrea Vedaldi, and David Novotny. Common pets in 3d: Dynamic new-view synthesis of real-life deformable categories. In *CVPR*, 2023. 3

[50] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 3

[51] Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, and Yebin Liu. Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. *arXiv preprint arXiv:2310.16818*, 2023. 3

[52] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023. 3

[53] Lorenzo Torresani, Aaron Hertzmann, and Christoph Bregler. Learning non-rigid 3d shape from 2d motion. *NeurIPS*, 2004. 3

[54] Edith Tretschk, Navami Kairanda, Mallikarjun BR, Rishabh Dabral, Adam Kortylewski, Bernhard Egger, Marc Habermann, Pascal Fua, Christian Theobalt, and Vladislav Golyanik. State of the art in dense monocular non-rigid 3d reconstruction. In *Comput. Graph. Forum*, pages 485–520, 2023. 3

[55] Shubham Tulsiani, Nilesh Kulkarni, and Abhinav Gupta. Implicit mesh reconstruction from unannotated image collections. *arXiv preprint arXiv:2007.08504*, 2020. 2

[56] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 4, 14

[57] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 3

[58] Yufu Wang, Nikos Kolotouros, Kostas Daniilidis, and Marc Badger. Birds of a feather: Capturing avian shape models from images. In *CVPR*, 2021. 2

[59] Haohan Weng, Tianyu Yang, Jianan Wang, Yu Li, Tong Zhang, CL Chen, and Lei Zhang. Consistent123: Improve consistency for one image to 3d object synthesis. *arXiv preprint arXiv:2310.08092*, 2023. 3

[60] Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi. Unsupervised learning of probably symmetric deformable 3d objects from images in the wild. In *CVPR*, 2020. 2

[61] Shangzhe Wu, Ameesh Makadia, Jiajun Wu, Noah Snavely, Richard Tucker, and Angjoo Kanazawa. De-rendering the world's revolutionary artefacts. In *CVPR*, 2021. 5

[62] Shangzhe Wu, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. DOVE: Learning deformable 3d objects by watching videos. *IJCV*, 2023. 2, 3, 5

[63] Shangzhe Wu, Ruining Li, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. Magicpony: Learning articulated 3d animals in the wild. In *CVPR*, 2023. 1, 2, 3, 4, 5, 6, 8, 12, 13, 14, 15

[64] Shangzhe Wu, Ruining Li, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. MagicPony: Learning articulated 3D animals in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 12

[65] Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE TPAMI*, 2019. 3, 5

[66] Jiacong Xu, Yi Zhang, Jiawei Peng, Wufei Ma, Artur Jesslen, Pengliang Ji, Qixin Hu, Jiehua Zhang, Qihao Liu, Jiahao Wang, et al. Animal3d: A comprehensive dataset of 3d animal pose and shape. In *ICCV*, 2023. 3, 5, 6, 12, 13

[67] Yinghao Xu, Hao Tan, Fujun Luan, Sai Bi, Wang Peng, Jihao Li, Zifan Shi, Kaylan Sunkavalli, Wetzstein Gordon, Zexiang Xu, and Zhang Kai. DMV3D: Denoising multi-view diffusion using 3d large reconstruction model. *arXiv preprint arXiv:2311.09217*, 2023. 3

[68] Gengshan Yang, Deqing Sun, Varun Jampani, Daniel Vlasic, Forrester Cole, Huiwen Chang, Deva Ramanan, William T. Freeman, and Ce Liu. LASR: Learning articulated shape reconstruction from a monocular video. In *CVPR*, 2021. 2

[69] Gengshan Yang, Deqing Sun, Varun Jampani, Daniel Vlasic, Forrester Cole, Ce Liu, and Deva Ramanan. ViSER: Video-specific surface embeddings for articulated 3d shape reconstruction. In *NeurIPS*, 2021. 2

[70] Gengshan Yang, Minh Vo, Neverova Natalia, Deva Ramanan, Vedaldi Andrea, and Joo Hanbyul. BANMo: Building animatable 3d neural models from many casual videos. In *CVPR*, 2022. 3

[71] Gengshan Yang, Chaoyang Wang, N. Dinesh Reddy, and Deva Ramanan. Reconstructing animatable categories from videos. In *CVPR*, 2023. 2, 3

[72] Jiayu Yang, Ziang Cheng, Yunfei Duan, Pan Ji, and Hongdong Li. Consistnet: Enforcing 3d consistency for multi-view images diffusion. *arXiv preprint arXiv:2310.10343*, 2023. 3

[73] Yuxiang Yang, Junjie Yang, Yufei Xu, Jing Zhang, Long Lan, and Dacheng Tao. Apt-36k: A large-scale benchmark for animal pose estimation and tracking. *NeurIPS*, 2022. 3, 5, 6, 12, 13

[74] Chun-Han Yao, Wei-Chih Hung, Yuanzhen Li, Michael Rubinstein, Ming-Hsuan Yang, and Varun Jampani. Lassie: Learning articulated shapes from sparse image ensemble via 3d part discovery. *NeurIPS*, 2022. 1, 2, 3, 6, 8, 12

[75] Chun-Han Yao, Wei-Chih Hung, Yuanzhen Li, Michael Rubinstein, Ming-Hsuan Yang, and Varun Jampani. Hi-lassie: High-fidelity articulated shape and skeleton discovery from sparse image ensemble. In *CVPR*, 2023. 6, 8, 12

[76] Chun-Han Yao, Amit Raj, Wei-Chih Hung, Yuanzhen Li, Michael Rubinstein, Ming-Hsuan Yang, and Varun Jampani. Artic3d: Learning robust articulated 3d shapes from noisy web image collections. *NeurIPS*, 2023. 2, 3, 6

[77] Yufei Ye, Shubham Tulsiani, and Abhinav Gupta. Shelf-supervised mesh prediction in the wild. In *CVPR*, 2021. 2

[78] Yunzhi Zhang, Shangzhe Wu, Noah Snavely, and Jiajun Wu. Seeing a rose in five thousand ways. In *CVPR*, 2023. 15

[79] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 15

[80] Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *CVPR*, 2017. 2

[81] Silvia Zuffi, Angjoo Kanazawa, and Michael J Black. Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. In *CVPR*, 2018. 2

A. Additional Results

We provide additional visualizations, including shape interpolation and generation, as well as additional comparisons in this supplementary material. Please see https://kyleleey.github.io/3DFauna/ for 3D animations.

A.1. Shape Interpolation between Instances

With the predictions of our model, we can easily interpolate between two reconstructions by interpolating the base embeddings $\tilde{\phi}$, instance deformations and the articulated poses ξ, as illustrated in Fig. 8. Here, we first obtain the predicted base shape embeddings $\tilde{\phi}$ for each of the three input images from the learned Semantic Bank. We then linearly interpolate between these embeddings to produce smooth a transition from one base shape to another, as shown in the last row of Fig. 8. Furthermore, we can also linearly interpolate the predicted articulated the image features ϕ (which is used as a condition to the instance deformation field $f_{\Delta V}$) as well as the predicted articulation parameters ξ, to generate smooth interpolations of between posed shapes, shown in the middle row. These results confirm that our learned shape space is continuous and smooth, and covers a wide range of animal shapes.

A.2. Shape Generation from the Semantic Bank

Moreover, we can also *generate* new animal shapes by sampling from the learned Semantic Bank, as shown in Fig. 9. First, we visualize the base shapes captured by each of the learned value tokens ϕ_k^{val} in the Semantic Bank. In the top two rows of Fig. 9, we show 20 visualizations of these base shapes randomly selected out of the 60 value tokens in total. We can also fuse these base shapes by linearly fusing the value tokens ϕ_k^{val} with a set of random weights (with a sum of 1), and generate the a wide variety of animal shapes, as shown in the bottom two rows.

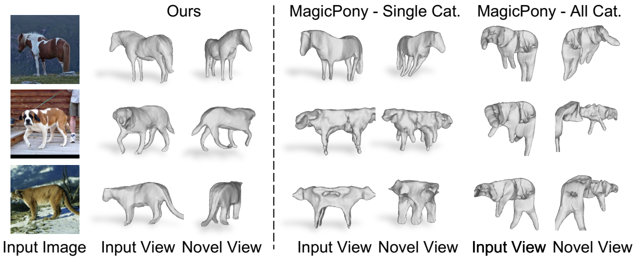

Figure 7. **Qualitative Comparisons** against two variants of MagicPony [63]. In the middle are reconstruction results of the category-specific MagicPony model trained on individual categories. On the right are results of MagicPony trained on all categories jointly, i.e. assuming all quadrupeds belong to one single category.

	APT-36K			
	Horse	Giraffe	Cow	Zebra
MagicPony [64]	0.775	0.699	0.769	0.778
Ours	**0.853**	**0.796**	**0.876**	**0.840**

	Animal3D		
	Horse	Cow	Zebra
LASSIE [74]	0.850	0.887	0.878
Hi-LASSIE [75]	0.410	0.720	0.704
MagicPony [64]	0.835	0.895	0.919
Ours	**0.884**	**0.903**	**0.942**

Table 2. **Quantitative Comparisons** on APT-36K [73] and Animal3D [66] for each category. Our method consistently performs better than MagicPony [63], LASSIE [74] and Hi-LASSIE [75] on all the categories.

	APT-36K			
	Horse	Giraffe	Cow	Zebra
Final Model	**0.853**	**0.796**	**0.876**	**0.840**
w/o Semantic Bank	0.402	0.398	0.371	0.373
Category-conditioned	0.822	0.776	0.832	0.798
w/o $\mathcal{L}_{\text{adv}}$	0.831	0.782	0.823	0.828

	Animal3D		
	Horse	Cow	Zebra
Final Model	**0.884**	**0.903**	**0.942**
w/o Semantic Bank	0.402	0.701	0.630
Category-conditioned	0.842	0.886	0.910
w/o $\mathcal{L}_{\text{adv}}$	0.813	0.871	0.873

Table 3. **Quantitative Ablation Studies** on APT-36K [73] and Animal3D [66] for each category.

A.3. Comparisons with Prior Work

Quantitative Results for Each Category. Here, we provide the per-category performance break for the quantitative comparisons in Tab. 2, which correspond to the aggregated results in Tab. 1. On APT36K [73], we evaluate on four categories including horse, giraffe, cow and zebra. On Animal3D [66], we use the available three categories: horse, cow and zebra. Our pan-category model consistently outperforms the MagicPony [63] baseline across all the categories, which highlights the benefits of the joint training of all categories. We also compare to LASSIE [74] and Hi-LASSIE [75] quantitatively by optimizing on three Animal3D categories individually, as each category contains a small size (< 100) of images similar to the default setup proposed in their papers.

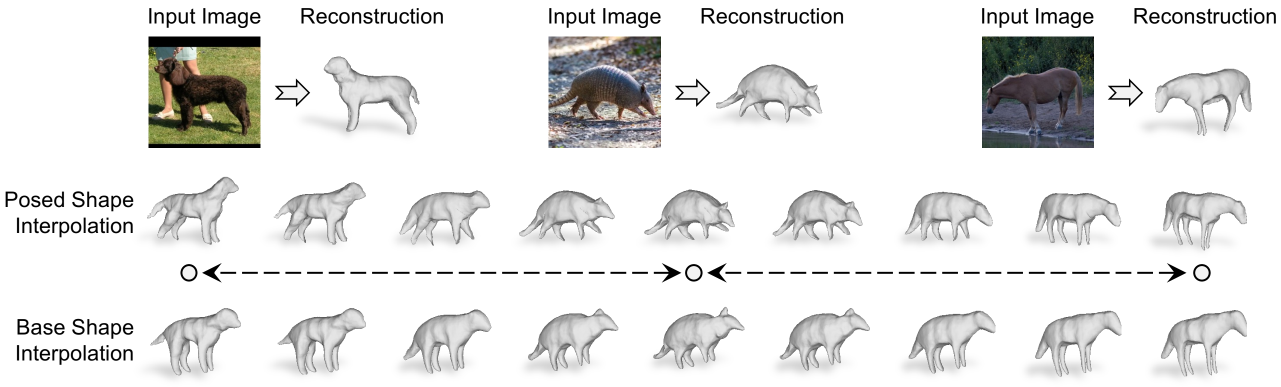

Figure 8. **Shape Interpolation between Instances.** On the top row, we show the 3D reconstructions from three input images. On the second and the third rows, we show the interpolation between the posed shapes and the base shapes.

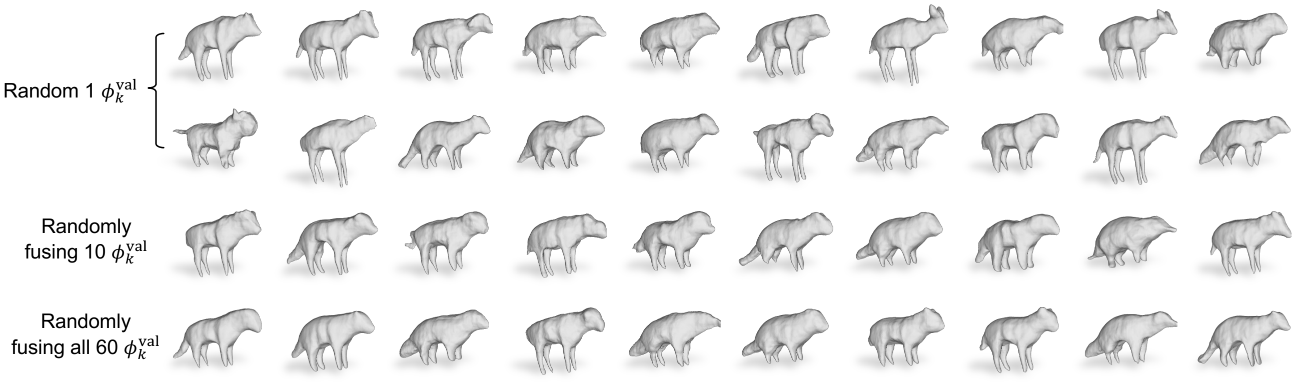
Random 1 ϕ_k^{val}
Randomly fusing 10 ϕ_k^{val}
Randomly fusing all 60 ϕ_k^{val}

Figure 9. **Shape Generation from the Learned Semantic Bank.** On the top two rows, we visualize 20 base shapes generated from the individual value tokens ϕ_k^{val} in the learned Semantic Bank. On the bottom two rows, we show the base shapes obtained by randomly fusing 10 and 60 value tokens ϕ_k^{val}.

K	2	10	60	100	500
PCK0.1	0.724	0.766	0.782	0.788	0.789

Table 4. **Bank Size Ablation Studies** on PASCAL [10].

MagicPony on All Categories. In Fig. 5, we show that MagicPony [63] fail to produce plausible 3D shapes when trained in a *category-specific* fashion on species with limited ($<$ 100) number of images. Alternatively, we can also train the MagicPony on our entire image dataset of all the animal species, i.e. treating all the images as in one single category. The results are shown in Fig. 7. As MagicPony maintains only one single base shape for all animal instances, which is not able to capture the wide variation of shapes of different animal species. On the contrary, our proposed Semantic Base Shape Bank learns various base shapes automatically adapted to different species, based on self-supervised image features.

A.4. Quantitative Ablation Studies

In addition to the qualitative comparisons in Fig. 6, Tab. 3 shows the quantitative ablation studies on APT-36K [73] and Animal3D [66]. As explained in Sec. 5.3 of the paper, we follow CMR [23] and optimize a linear mapping from our predicted vertices to the annotated keypoints in the *input view*. These numerical results are consistent with the visual comparisons in Fig. 6.

We also conducted additional experiments with different bank sizes, including $K = 2$, 10, 60, 100, 500, and report the PCK scores on PASCAL [10] in Tab. 4. The quality grows with K; we pick $K = 60$ as a good trade-off with the computational cost.

A.5. More Visualizations from 3D-Fauna

We show more visualization results of 3D-Fauna on a wide variety of animals in Figure 13, Figure 14 and Figure 15, including horse, weasel, pika, koala and so on. Note that

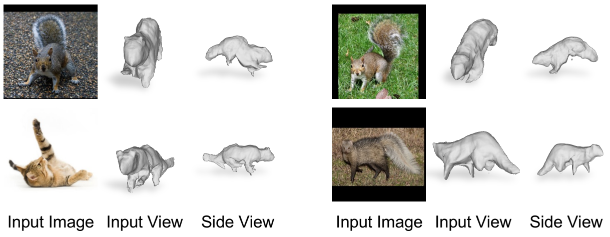

Figure 10. **Failure Cases.** For fluffy and highly deformable animals in challenging poses, our model still struggles in predicting the accurate poses and shapes.

our model produces these articulated 3D reconstructions from just a single test image in feed-forward manner, without even knowing the category labels of the animal species. With the articulated pose prediction, we can also easily animate the reconstructions in 3D. More visualizations are presented at `https://kyleleey.github.io/3DFauna/`.

A.6. Failure Cases and Limitations

Despite promising results on a wide variety of quadruped animals, we still recognize a few limitations of the current method. First, we only focus on quadrupeds which share a similar skeletal structure. Although this covers a large number animals, including most mammals as well as many reptiles, amphibians and insects, the same assumption will not hold for many other animals in nature. Jointly estimating the skeletal structure and 3D shapes directly from raw images remains a fundamental challenge for modeling the entire biodiversity. Furthermore, for some fluffy animals that are highly deformable, like cats and squirrels, our model still struggles to reconstruct accurate poses and 3D shapes, as shown in Fig. 10.

Another failure case is the confusion of left and right legs, when reconstructing images taken from the side view, for instance, in the second row of Fig. 13. Since neither the object mask nor the self-supervised features [41] can provide sufficient signals to disambiguate the legs, the model would ultimately have to resort to the subtle appearance cues, which still remains as a major challenge. Finally, the current model still struggles at inferring high-fidelity appearance in a feed-forward manner, similar to [63], and hence, we still employ a fast test-time optimization for better appearance reconstruction (within seconds). This is partially due to the limited size of the dataset and the design of the texture field. Leveraging powerful diffusion-based image generation models [48] could provide additional signals to train a more effective 3D appearance predictor, which we plan to look into for future work.

B. Additional Technical Details

B.1. Modeling Articulations

In this work, we focus on quadruped animals which share a similar quadrupedal skeleton. Here, we provide the details for the bone instantiation on the rest-pose shape based on a simple heuristic, the skinning model, and the additional bone rotation constraints.

Adaptive Bone Topology. We adopt a similar quadruped heuristic for rest-pose bone estimation as in [63]. However, unlike [63] which focuses primarily on horses, our method needs to model a much more diverse set of animal species. Hence, we make several modifications in order for the model to adapt to different animals automatically. For the ‘spine’, we still use a chain of 8 bones with equal lengths, connecting the center of the rest-pose mesh to the two most extreme vertices along z-axis. To locate the four feet joints, we do not rely on the four xz-quadrants as the feet may not always land separately in those four quadrants, for instance, for animals with a longer body. Instead, we locate the feet based on the distribution of the vertex locations. Specifically, we first identify the vertices within the lower 40% of the total height (y-axis). We then use the center of these vertices as the origin of the xz-plane and locate the lowest vertex within each of the new quadrants as the feet joints. For each leg, we create a chain of three bones of equally length connecting the foot joint to the nearest joint in the spine.

Bone Rotation Prediction. Similar to [63], the viewpoint and bone rotations are predicted separately using different networks. The viewpoint ξ_1 is predicted via a multi-hypothesis mechanism, as discussed in Appendix B.2. For the bone rotations $\xi_{2:B}$, we first project the middle point of each *rest-pose* bone onto the image using the predicted viewpoint, and sample its corresponding local feature from the feature map using bilinear interpolation. A Transformer-based [56] network then fuses the global image feature, local image feature, 2D and 3D joint locations as well as the bone index, and produces the Euler angle for the rotation of each bone. Unlike [63], we empirically find it beneficial to add the bone index on top of other features instead of concatenation, which tends to encourage the model to separate the legs with different rotation predictions.

Skinning Weights. With the estimated bone structure, each bone b except for the root has the parent bone $\pi(b)$. Each vertex $V_{\text{ins},i}$ on the shape V_{ins} is then associated to all the bones by skinning weights w_{ib} defined as:

$$\begin{aligned} w_{ib} &= \frac{e^{-d_{ib}/\tau_s}}{\sum_{k=1}^{B} e^{-d_{ik}/\tau_s}}, \quad \text{where} \\ d_{ib} &= \min_{r\in[0,1]} ||V_{\text{ins},i} - r\tilde{\mathbf{J}}_b - (1-r)\tilde{\mathbf{J}}_{\pi(b)}||_2^2 \end{aligned} \tag{6}$$

is the minimal distance from the vertex $V_{\text{ins},i}$ to each bone b, defined by the rest-pose joint location $\tilde{\mathbf{J}}_b$ in world coordinates. The τ_s is a temperature parameter set to 0.5. We then use the *linear blend skinning equation* to pose the vertices:

$$\begin{gathered} V_i(\xi) = \left(\sum_{b=1}^{B} w_{ib} G_b(\xi) G_b(\xi^*)^{-1} \right) V_{\text{ins},i}, \\ G_1 = g_1, \quad G_b = G_{\pi(b)} \circ g_b, \quad g_b(\xi) = \begin{bmatrix} R_{\xi_b} & \mathbf{J}_b \\ 0 & 1 \end{bmatrix}, \end{gathered} \tag{7}$$

where the ξ^* denotes the bone rotations at rest pose.

Bone Rotation Constraints. Following [63], we regularize the magnitude of bone rotation predictions by $\mathcal{R}_{\text{art}} = \frac{1}{B-1} \sum_{b=2}^{B} ||\xi_b||_2^2$. In experiments, we find a common failure mode where instead of learning a reasonable shape with appropriate leg lengths, the model tends to predict excessively long legs for animals with shorter legs and bend them away from the camera. To avoid this, we further constrain the range of the angle predictions. Specifically, we forbid the rotation along y-axis (side-way) and z-axis (twist) of the lower two segments for each leg. We also set a limit to the rotation along y-axis and z-axis of the upper segment for each leg as $(-10°, 10°)$. For the body bones, we further limit the rotation along the z-axis within $(-6°, 6°)$.

B.2. Viewpoint Learning Details

Recovering the viewpoint of an object from only one input image is an ill-posed problem with numerous local optima in the reconstruction objective. Here, we adopt the multi-hypothesis viewpoint prediction scheme introduced in [63]. In detail, our viewpoint prediction network outputs four viewpoint rotation hypotheses $R_k \in SO(3), k \in \{1, 2, 3, 4\}$ within each of the four xz-quadrants together with their corresponding scores σ_k. For computational efficiency, we randomly sample one hypothesis at each training iteration, and minimize the loss:

$$\mathcal{L}_{\text{hyp}}(\sigma_k, \mathcal{L}_{\text{rec},k}) = (\sigma_k - \texttt{detach}(\mathcal{L}_{\text{rec},k}))^2, \tag{8}$$

where `detach` indicates that the gradient on reconstruction loss is detached. In this way, σ_k essentially serves as an estimate of the expected reconstruction error for each hypothesis k, without actually evaluating it which would otherwise require the expensive rendering step. During inference time, we can then take the `softmax` of its inverse to obtain the probability p_k of each hypothesis k: $p_k \propto \exp(-\sigma_k/\tau)$, where the temperature parameter τ controls the sharpness of the distribution.

B.3. Mask Discriminator Details

To sample another viewpoint and render the mask for the mask discriminator, we randomly sample an azimuth angle and rotate the predicted viewpoint by that angle. For conditioning, the detached input base embedding $\tilde{\phi}$ is concatenated to each pixel in the mask along the channel dimension, similar to CycleGAN [79]. In practice, we also add a gradient penalty term in the discriminator loss following [40, 78].

Parameter	Value/Range
Optimiser	Adam
Learning rate on prior and bank	1×10^{-3}
Learning rate on others	1×10^{-4}
Number of iterations	800k
Enable articulation iteration	20k
Enable deformation iteration	500k
Mask Discriminator iterations	$(80\text{k}, 300\text{k})$
Batch size	6
Loss weight λ_{m}	10
Loss weight λ_{im}	1
Loss weight λ_{feat}	$\{10, 1\}$
Loss weight λ_{Eik}	0.01
Loss weight λ_{def}	10
Loss weight λ_{art}	0.2
Loss weight λ_{hyp}	$\{50, 500\}$
Loss weight λ_{adv}	0.1
Image size	256×256
Field of view (FOV)	$25°$
Camera location	$(0, 0, 10)$
Tetrahedral grid size	256
Initial mesh centre	$(0, 0, 0)$
Translation in x- and y-axes	$(-0.4, 0.4)$
Translation in z-axis	$(-1.0, 1.0)$
Number of spine bones	8
Number of bones for each leg	3
Viewpoint hypothesis temperature τ	$(0.01, 1.0)$
Skinning weight temperature τ_{s}	0.5
Ambient light intensity k_a	$(0.0, 1.0)$
Diffuse light intensity k_d	$(0.5, 1.0)$

Table 5. **Training details and hyper-parameter settings.**

B.4. Network Architectures

We adopt the architectures in [63] except the newly introduced Semantic Base Shape Bank and mask discriminator. For the SBSM, we add a modulation layer [24, 25] to each of the MLP layers to condition the SDF field on the base embeddings $\tilde{\phi}$. To condition the DINO field, we simply concatenate the embedding to the input coordinates to the network. The mask discriminator architecture is identical to that of GIRAFFE [40], except that we set input dimension as $129 = 1 + 128$, accommodating the 1-channel mask and the 128-channel shape embedding. We set the size of the memory bank $K = 60$. In practice, to allow bank to represent categories with diverse kinds of shapes, we only

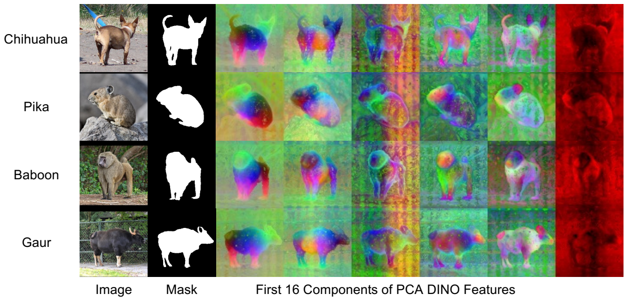

Figure 11. **Data Samples.** We show some samples of our training data. Each sample consists of the RGB image, automatically-obtained segmentation mask, and the corresponding 16-channel PCA feature map.

fuse the value tokens with top 10 cosine similarities.

B.5. Hyper-Parameters and Training Schedule

The hyper-parameters and training details are listed in Tab. 5. We train the model for 800k iterations on a single NVIDIA A40 GPU, which takes roughly 5 days. In particular, we set λ_{feat}=10, and λ_{hyp}=50 at the start of training. After 300k iterations we change the values to λ_{feat}=1, λ_{hyp}=500. During the first 6k iterations, we allow the model to explore all four viewpoint hypotheses by randomly sampling the four hypotheses uniformly, and gradually decrease the chance of random sampling to 20% while sampling the best hypothesis for the rest 80% of the time. To save memory and computation, at each training iteration, we only feed images of the same species in a batch, and extract one base shape by averaging out the base embeddings. At test time, we just directly use the shape embedding for each individual input image.

B.6. Data Pre-Processing

We use off-the-shelf segmentation models [27, 28] to obtain the masks, crop around the objects and resize the crops to a size of 256 × 256. For the self-supervised features [41], we randomly choose 5k images from our dataset to compute the Principal Component Analysis (PCA) matrix. Then we use that matrix to run inference across all the images in our dataset. We show some samples of different animal species in Fig. 11. It is evident that these self-supervised image features can provide efficient semantic correspondences across different categories. Note that masks are only for supervision, our model takes the raw image shown on the left as input for inference.

B.7. Species Size Distribution

We show a plot of the distribution of different species in our dataset below, including 7 well-represented categories (red) and 121 few-shot categories (orange). To balance the training, we duplicate the samples of few-shot categories to match the size of the rest. Many examples in Fig. 4 and Fig. 13 in fact belong to the few-shot categories, such as koala, fisher and prairie dog.

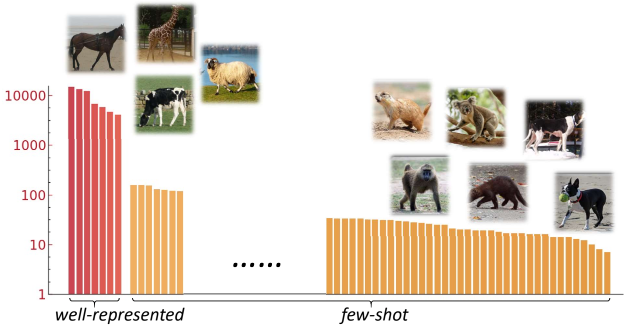

Figure 12. **Species Distribution.** We show the distribution of different animal species in our training dataset, including well-represented species with thousands of images and rare species with less than 100 images.

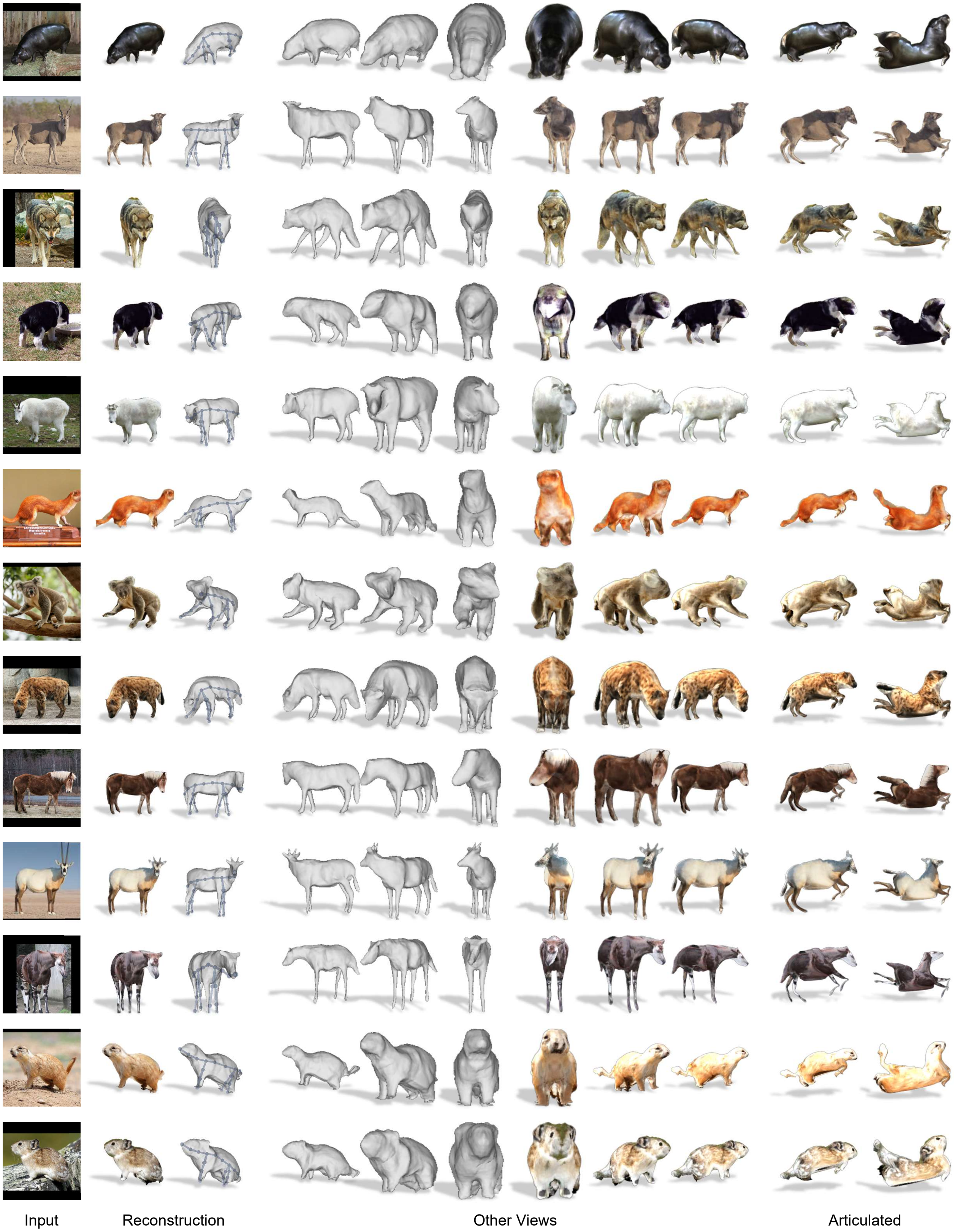

Figure 13. **Single Image 3D Reconstruction.** Given a single image of any quadruped animal at test time, our model reconstructs an articulated and textured 3D mesh in a feed-forward manner without requiring category labels, which can be readily animated.

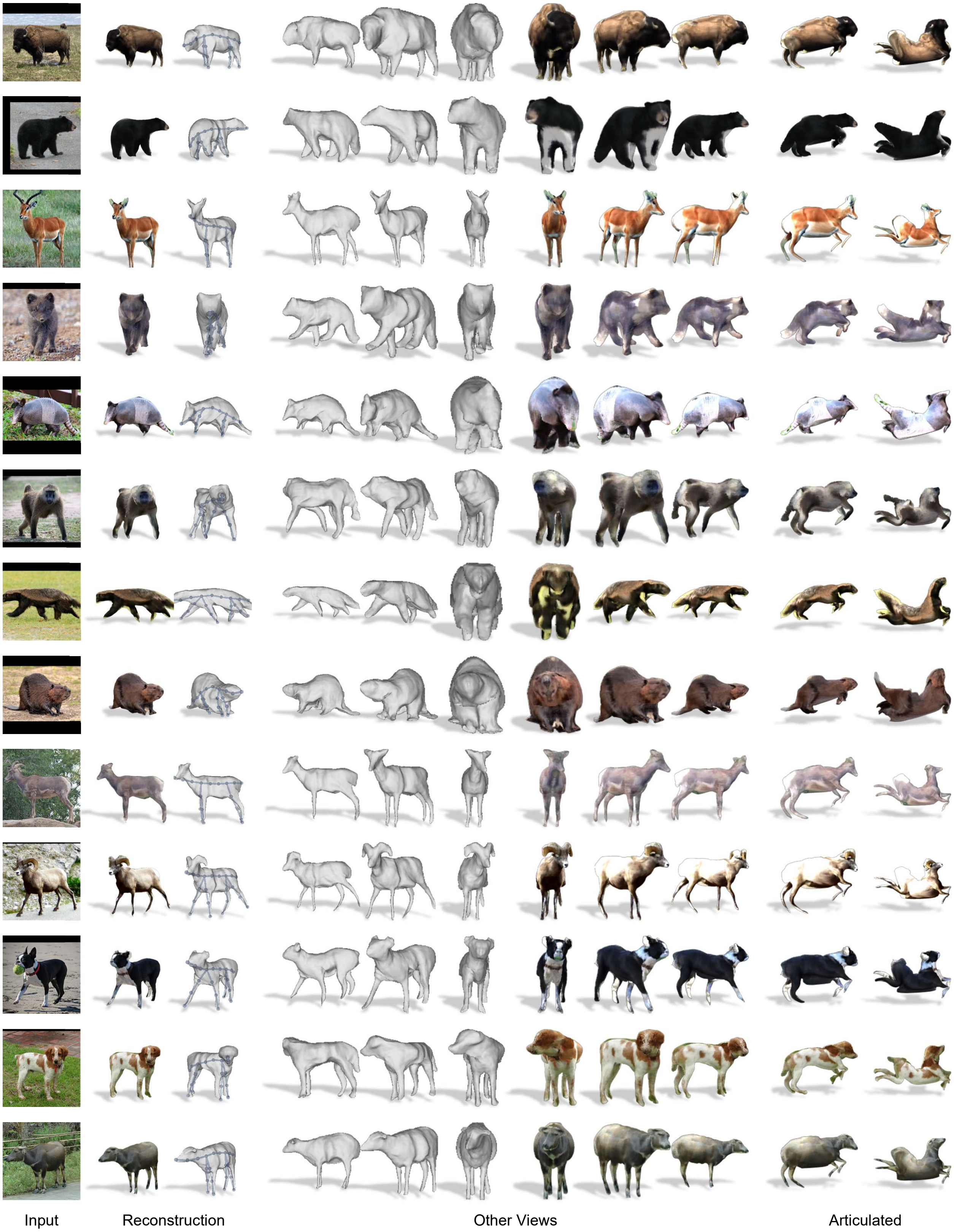

Figure 14. **Single Image 3D Reconstruction.** Given a single image of any quadruped animal at test time, our model reconstructs an articulated and textured 3D mesh in a feed-forward manner without requiring category labels, which can be readily animated.

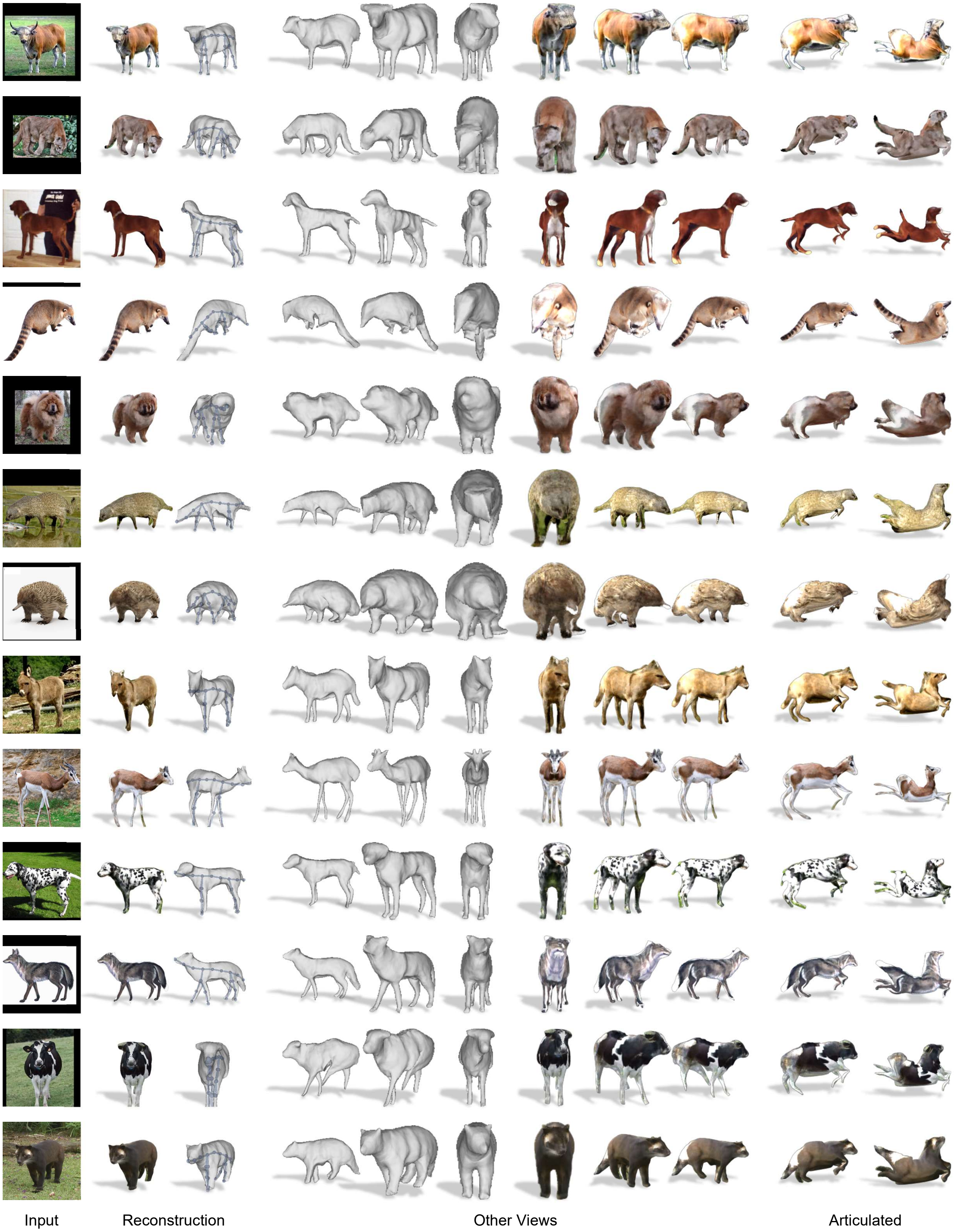

Figure 15. **Single Image 3D Reconstruction.** Given a single image of any quadruped animal at test time, our model reconstructs an articulated and textured 3D mesh in a feed-forward manner without requiring category labels, which can be readily animated.