CASA: Category-agnostic Skeletal Animal Reconstruction

Anonymous Author(s)
Affiliation
Address
email

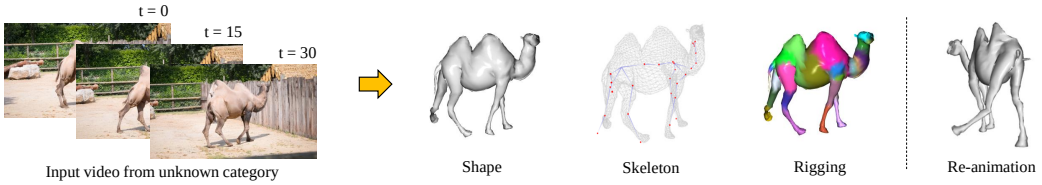

Figure 1: Given a monocular video with the object from unknown category, CASA jointly learns articulated shape with skeleton, and rigging, which can be animated with novel poses.

Abstract

Recovering a skeletal shape from a monocular video is a longstanding challenge. Prevailing nonrigid animal reconstruction methods often adopt a control-point driven animation model and optimize bone transforms individually without considering skeletal topology, yielding unsatisfactory shape and articulation. In contrast, humans can easily infer the articulation structure of an unknown character by associating it with a seen articulated object in their memory. Inspired by this fact, we present CASA, a novel category-agnostic articulated animal reconstruction method. Our method consists of two components, a video-to-shape retrieval process and a neural inverse graphics framework. During inference, CASA first finds a matched articulated shape from a 3D character assets bank so that the input video scores highly with the rendered image, according to a pretrained image-language model. It then integrates the retrieved character into an inverse graphics framework and jointly infers the shape deformation, skeleton structure, and skinning weights through optimization. Experiments validate the efficacy of our method in shape reconstruction and articulation. We further show that we can use the resulting skeletal-animated character for re-animation.

1 Introduction

Recovering the shape, articulation, and dynamics of animals from images and videos is a long-standing task in computer vision and graphics. Achieving this goal will enable new future applications for 3D modeling and reanimation of characters captured with mobile devices. Nevertheless, accurately reasoning about the geometry and kinematics of an animal in the wild remains an ambitious problem for several reasons: (1) *partial visibility* of the captured animal; (2) *variability* of shape across different animal categories; (3) *ambiguity* of unknown kinematics. Take the elephant in Fig. 1 as an example – to faithfully modeling it requires: 1) hallucinate the unobserved (*e.g.*, occluded, back-side) region, 2) recovering the unique shape and scale, and 3) predicting its kinematic structure.

Remarkable progress has been made to address the aforementioned challenges by either exploiting richer sensor configurations, such as multi-camera cage [10, 16] and depth sensors [33, 41, 34], or alternatively by making strong assumptions specific to a particular object category like humans [29,

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

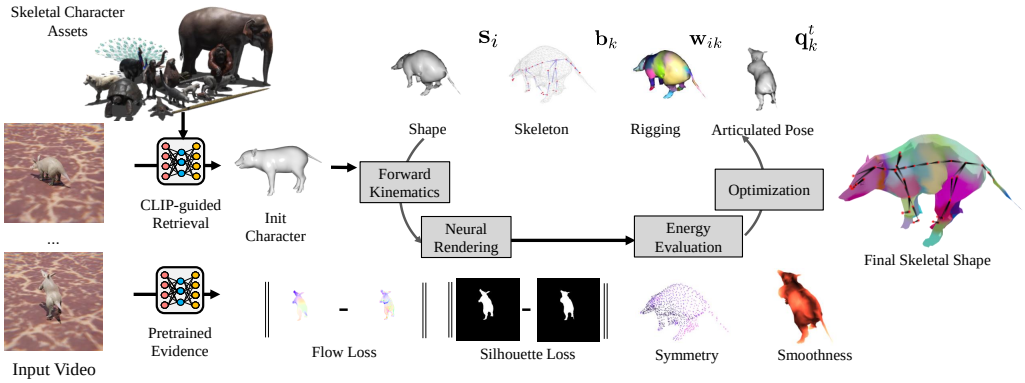

Figure 2: Overview. Given an input video, A video-to-shape retrieval process (§ 3.2) is first conducted with the guidance of pre-trained CLIP. Initialized by the retrieved shape, we jointly optimize shape, skeleton (including bone length and joint angle), and skinning through inverse rendering. (§ 3.3).

46, 45, 63, 47] and quadruped animals [66, 67, 68]. However, these assumptions greatly limit the applicability of reconstruction systems, which fail to generalize to animals in the wild.

We aim to reconstruct an arbitrary articulated animal using a monocular video casually captured in the wild. A few recent works so far demonstrate promising results toward tackling this challenging problem [60, 61, 56, 62, 37]. However, they often impose non-realistic assumptions on articulation, such as control point driven deformation [60, 61, 56, 62] or free-form deformation [37, 41]. As a result, they fall short of the goal of modeling skeletal characters that can be realistically re-animated in downstream applications. Furthermore, there remains significant improvement space for the quality of the inferred animal shape.

In this work, we propose CASA , a novel solution for category-agnostic animal reconstruction in the wild. CASA jointly estimates an arbitrary animal's 3D shape, kinematic structure, rigging weight, and articulated poses of each frame from a monocular video (see Fig. 1). Unlike previous nonrigid reconstruction works [60, 61, 56], we exploit a skeleton-based articulation model as well as forward kinematics, ensuring the realism of the resulting skeletal shape (§ 3.1). Specifically, we propose two novel components: 1) a video-to-shape retrieval process, as well as 2) a neural inverse graphics framework. Given an input video, CASA first finds a template shape from a 3D character assets bank so that the input video scores highly with the rendered image, according to a pretrained image-language model [40] (§ 3.2). Using the retrieved character as initialization, we jointly optimize bone length, joint angle, 3D shape, and blend skinning weight so that the final outputs are consistent with visual evidence, *i.e.*, input video (§ 3.3).

Another issue that hinders the study of articulated 3D reconstruction is that the existing datasets [12, 27, 3] lack realistic video footage and ground-truth labels across dynamic objects of different classes. To overcome this challenge, we introduce a photorealistic synthetic dataset PlanetZoo, which is generated using the physical-based rendering [7] and rich 3D assets from a zoo simulator [30]. PlanetZoo consists of a realistic rendering of 251 categories of articulated animals. Importantly, PlanetZoo also provides ground-truth shape, skeleton, joint angles, and rigging, allowing us to evaluate category-agnostic 4D reconstruction holistically (§ 4).

We validate our proposed approach on both PlanetZoo and the real-world DAVIS dataset. Our experiments demonstrate that CASA can recover fine shape details and realistic skeleton topology, handles a wide variety of shapes, and adapts well to unseen categories. In addition, CASA enjoys great generalizability since all the evidence neural networks are pre-trained from external datasets without fine-tuning. Finally, CASA recovered a skeletal-animatable character that is readily compatible for downstream tasks such as re-animation, rendering and simulation.

2 Related Work

Parametric shape reconstruction. Parameteric models have been proposed to efficiently model body [29], hand [44], face [25] and animal [68] 3D-shapes and body poses. Recent works use such models on human body capture [65, 32, 2, 6, 57, 38, 36] and animal reconstruction [68, 1, 67]. While model-based approaches show great results in reconstructing objects of a certain category with fixed

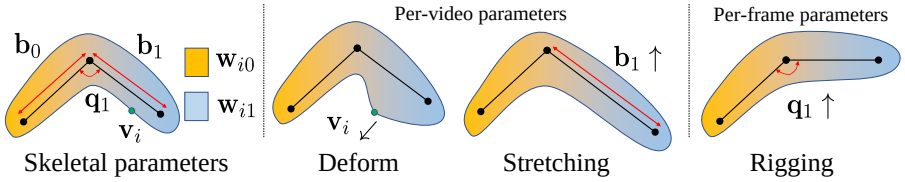

Figure 3: **Skeletal shape parametric model.**[Rebuttal: Our parametric model consists of joint angles, bone length, skinning weight, and vertex positions, where joint angles change per-frame and the other parameters are consistent for all frames. With this model, the shape would be tuned by vertex deformation as well as the stretching of bones. The target articulation would be fit by predicting per-frame joint angles.]

topology, it is challenging to apply those models to categories that are hard to distinguish or have insufficient 3D data, especially for animals.

Category-specific reconstruction. Many recent works [32, 17, 26, 22, 53, 47, 56, 26, 20] study the problem of predicting deformable objects from image collections or videos within a specific category. These methods heavily rely on strong category priors. CMR [17] uses keypoint annotations and masks to predict the shape deformation. U-CMR [56] replaces the keypoints in CMR with category-specific canonical shapes. VMR [26] extends the input of CMR from images to videos, as well as adopts an as-rigid-as-possible (ARAP) constraint for per-frame shape reconstructions based on video inputs. Besides the strict requirement on the category of input objects, most of these methods are trained on single-view images, limiting their ability to recover high-fidelity mesh details. [Rebuttal: ACSM [20] infers the articulation and pose of template models corresponding to input videos by predicting a canonical surface mapping.]

Category-agnostic nonrigid shape reconstruction. Classic approaches use Non-Rigid Structure from Motion (NRSfM) [4, 21, 50] for category-agnostic nonrigid shape reconstruction. They use a collection of images capturing the same object from multiple views with 2D correspondences. Given that the long-range correspondences are hard to obtain, they often require manual annotations. [Rebuttal: WIM [35] learns re-poseable shape representation for objects of different categories, but depending on multi-view observations.]Recent works adopts learning-based approaches by either dynamic reconstruction [24, 41] or inverse graphics optimization [60, 61], LASR [60] and ViSER [61] optimize articulated shapes from monocular videos and achieve promising reconstruction results. However, these methods suffer from incorrect shape prediction when certain parts are occluded or invisible in the input video sequence.

3D reconstruction dataset A plethora of synthetic datasets [51, 64, 15, 43] and interactive simulated environment [48, 49, 9, 55] aimed for 3D reconstruction have been proposed in recent years. The ShapeNet [8] dataset provides a benchmark for common static 3D object shape reconstruction. Among all of these datasets, only a few aim for dynamic object reconstruction [27]. Given that the synthetic dataset has a large domain gap to the real-world settings, photo-realistic rendered samples are strongly preferred. To this end, we propose a photorealistic synthetic dataset PlanetZoo to study the dynamic animal reconstruction problem. Our dataset contains high-fidelity rendered images and covers a wide range of animal categories.

3 Category-agnostic Skeletal Animal Reconstruction

We present CASA, a novel category-agnostic skeletal animal reconstruction framework. The input to our method are RGB frames from a monocular video $\{\mathbf{I}_t\}_{1...T}$, object segmentation masks $\{\mathbf{M}_t\}_{1...T}$ and optical flow maps $\{\mathbf{F}_t\}_{1...T}$ computed from each pair of consecutive frames. From these inputs, we aim to recover an animal shape $\mathbf{s}_0$ and its articulated deformed shape $\mathbf{s}_t$ at each time t.

Our method exploits the skeletal articulation model and forward kinematics to ensure the realism of the resulting skeletal shape (§ 3.1). Our method consists of two phases. In the retrieval phase (§ 3.2), CASA finds a template character from an existing asset bank based on the similarity to the input video in the deep embedding space using a pre-trained encoder. The retrieved template is fed into the neural inverse graphics phase as initialization(§ 3.3). Finally, we jointly reason the final shape, skeleton, rigging and the articulated pose of each frame through an energy minimization framework. Fig. 2 demonstrates an overview of our approach.

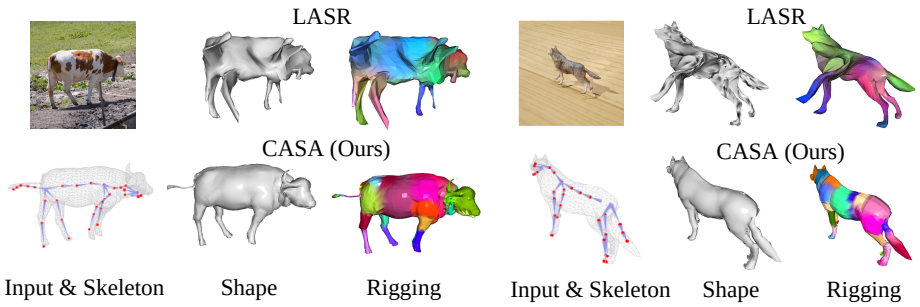

Figure 4: **Reconstructed articulated shape.** Top-left: input; top: LASR; bottom: ours.

3.1 Articulation model

Skeletal parameters. We exploit a stretchable bone-based skeletal model for our deformable shape parameterization. This articulated model consists of three components: 1) a triangular mesh consisting of a set of vertices and shapes in the canonical pose, describing the object's shape; 2) a set of bones connected by a kinematic tree. Each one has a bone length parameter and associated rigging weight over each vertex; 3) a joint angle describing the relative transformation between each adjacent bone. To summarize, a deformed shape $\mathbf{s}_t$ at time t can be represented as:

$$\mathbf{s}_t = \left\{ \left\{ \mathbf{v}_i \in \mathbb{R}^3 \right\}_N, \left\{ \mathbf{f}_j \in \mathbb{I}^3 \right\}_T, \left\{ \mathbf{w}_i \in \Delta^K \right\}_N, \left\{ b_k \in \mathbb{R}^1 \right\}_K, \left\{ \mathbf{q}_k^t \in \mathbb{SO}(3) \right\}_K \right\} \tag{1}$$

where $\mathbf{v}_i$ is the vertex position, $\mathbf{f}_j$ represents a triangle face parameterized as a tuple of three vertex indices, $\mathbf{w}_i$ is the rigging weight constrained in a K-dimensional simplex: $\Delta^K = \left\{ (w_0, \ldots, w_K) \in \mathbb{R}^K \mid \sum_{k=0}^K w_k = 1 \text{ and } w_k \geq 0 \right\}$; b_k is the scalar bone length for each bone and $\mathbf{q}_k^t$ is the joint angle for each bone at a time t, represented as an unit quaternion in the $\mathbb{SO}(3)$ space. Note that all the variables except joint angles are shared across time.

Fig. 3 gives an overview of our skeletal model. It is grounded by the nature of many articulated objects. Unlike the commonly used control-point based articulation [33, 61, 60], our model reflects the constraints imposed by bones and joints, leading to more natural articulated deformation. Compared to category-specific parametric models [29, 68], it is more flexible and generalizable.

Forward kinematics. We use the forward kinematic model [23] to compute the transformation of each bone along the kinematic tree. Specifically, the rigid transformation of a bone is uniquely defined by the joint angles and bone length scales of the bone itself and its ancestors along the kinematic tree. Given a bone k, the rigid transform between its own frame and the root can be computed through recursively applying the relative rigid transformation along the chain:

$$\mathbf{T}_k^t(\mathbf{q}_{\text{ans(k)}}^t, \mathbf{b}_{\text{ans(k)}}) = \mathbf{T}_{\text{pa}(k)}^t \mathbf{T}_{k,\text{pa}(k)}^t(\mathbf{q}_k^t, b_k) = \prod_{k' \in \text{ans}(k)} \mathbf{T}_{k',\text{pa}(k')}^t(\mathbf{q}_{k'}^t, b_{k'}) \tag{2}$$

where $\mathbf{T}_k^t, \mathbf{T}_{\text{pa}(k)}^t$ is the transformation of bone k and its parent node at frame t. $\mathbf{q}_{\text{ans(k)}}, \mathbf{b}_{\text{ans(k)}}$ are the joint angles and bone lengths from all the ancestors of target bone k. $\mathbf{T}_{k,\text{pa}(k)}^t(\mathbf{q}_k^t, b_k)$ is the relative rigid transform between the bone and its parent frame, consists of a joint center translation along z axis decided by the bone length $b_k \mathbf{e}_z = [0, 0, b_k]$ and a rotation around the joint center $\mathbf{R}(\mathbf{q}_k^t)$.

Linear blend skinning. Given the rigid transformation of each bone, we adopt linear blend skinning (LBS) to compute the transform of each point. Specifically, the deformation of each point $\mathbf{v}$ is decided by a linear composing the rigid transforms of bones through its skinning weight: $\mathbf{v}_i^t = \sum_{k=0}^K w_{i,k} \mathbf{T}_k^t \mathbf{v}_i$ where $w_{i,k}$ is the rigging weight. $\mathbf{T}_k^t$ is the transformation of bone k at frame t as defined in Eq. 2. For simplicity, we omit its dependent variables.

Stretchable bones. The aforementioned nonrigid deformation assumes the independent shape and bone lengths. Naturally, the shape should deform accordingly as bone length changes. To model such a relationship, we exploit a stretchable bone deformation model [14]. For each bone, we compute an additional scaling transform along the bone vector direction based on its bone length change and apply this scaling transform to any points associated with this bone. Fig. 3 provides an illustration of stretching. For mathematical details on stretching, we refer readers to our supplementary document. Stretchable bones bring a two-fold advantage: (a) it allows us to model shape variations within the same topological structures; (b) it makes the topological structure adjustable by "shrinking" the bones, e.g. a quadrupedal animal can be evolved into a seal-like skeletal model through optimization.

Figure 5: **Overview of PlanetZoo**. We collect a novel dataset PlanetZoo containing high-fidelity and animatable 3D animal models. On the right we show renderings for the elephant with different poses.

3.2 Video-to-shape retrieval

Directly optimizing all skeletal parameters defined in Eq. 1 is challenging due to the variability and highly structured skeletal parameterization. Many deformable objects share similar topology structures, albeit with significant shape differences. Motivated by this, we propose initializing skeletal shapes in a data-driven manner by choosing the best matching template from an asset bank.

Videos and skeletal shapes are in two different modalities; hence establishing a similarity measure is hard. Inspired by the recent success of image-language pretraining models for 3D [54, 31], we utilize realistic rendering and pretraining image embedding models [40] to bridge this gap. Specifically, we first pre-render video footages for each character in the asset bank through physical-based rendering [7]. The environment lighting, background, articulated poses, and camera pose for each video are randomized to gain diversity and robustness. We then extract the image embedding features for each video using CLIP [40], which is a language-image embedding model that is pretrained on a large-scale image-caption dataset. It captures the underlying semantic similarity between images well despite the large appearance and viewpoint difference. This is particularly suitable for retrieving shapes with similar kinematic structures, as the kinematics of animals is often related to semantics.

During inference, we extract embedding features from a given input video and measure the cosine similarity between the input video and the rendered video of each object in the embedding space. We then select the highest-scoring articulated shape as the retrieved character.

3.3 Neural inverse graphics via energy minimization

We expect our final output skeletal shape to 1) be consistent with the input video observation; 2) reflect prior knowledge about articulated shape, such as symmetry and smoothness. Inspired by this, we formulate our as an energy minimization problem.

Energy formulation. We exploit visual cues from each frame including segmentation mask $\{\mathbf{M}_t\}$ and optical flow $\{\mathbf{F}_t\}$ through off-the-shelf neural networks [52, 60]. We also incorporate two types of prior knowledge, including the motion smoothness for joint angles $\mathbf{q}_k^t$ of bones across frames, as well as the symmetry constraint for the per-vertex offset $\mathbf{v}_i$ at the reference frame. In particular, let $\mathcal{I} = (\{\mathbf{I}_t\}, \{\mathbf{M}_t\}, \{\mathbf{F}_t\})$ be the input video frames and corresponding visual cues, and $\{\mathbf{s}_t\}$ be the predicted shapes of all frames, we formulate the energy for minimization as:

$$\min_{\{\mathbf{s}_t\}_{1\dots T}} \lambda_1 E_{\text{cue}}(\{\mathbf{s}_t\}, \mathcal{I}) + \lambda_2 E_{\text{smooth}}(\{\mathbf{s}_t\}) + \lambda_3 E_{\text{symm}}(\{\mathbf{s}_t\}) \tag{3}$$

where E_{cue} measures the consistency between the articulated shape and the visual cues; E_{smooth} promotes smooth transitions overtime; E_{symm} encodes the fact that many articulated objects are symmetric. The three energy terms complement each other, helping our model capture motions of the object according to visual observations, as well as constraining the deformed object to be natural and temporally consistent. We describe details of each term below.

Visual cue consistency. The visual cue consistency energy measures the agreement between the rendered maps of the articulated shape and 2D evidence (flow and silhouettes) from videos. We use a differentiable renderer [28] to generate projected object silhouettes $\{\mathbf{M}_t\}$. Additionally, we project vertices of predicted shapes at two consecutive frames to the camera view and compute the projected 2D displacement to render optical flow, following the prior work [60]. We leverage PointRend [19]

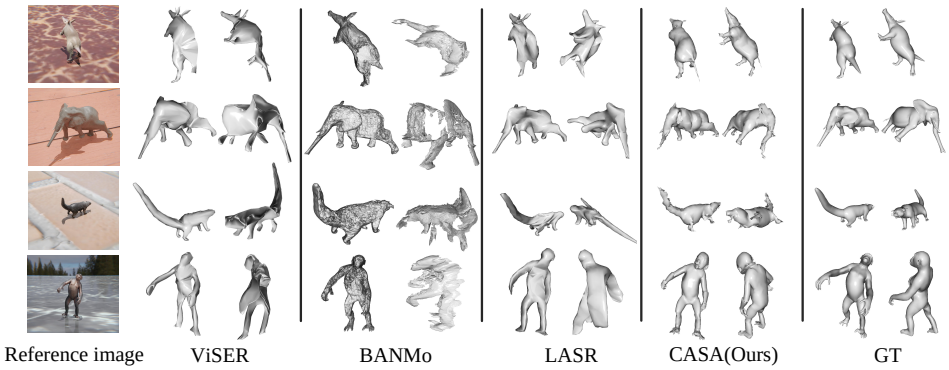

Figure 6: **Qualitative comparison of our method on PlanetZoo.** The reconstruction results of the reference image are shown in two viewpoints.

Table 1: Quantitative results on PlanetZoo.

Method	mIoU ↑	mCham ↓	Skinning ↓	Joint ↓	Re-animation ↓
ViSER [61]	0.190	0.133	6.398	0.416	0.028
LASR [60]	0.306	0.094	**2.576**	**0.158**	0.033
CASA (ours)	**0.435**	**0.051**	3.238	0.324	**0.010**

for object segmentation and volumetric correspondence net [59] for flow prediction. The energy measures the difference between the rendered and inferred cues in ℓ_2 distance:

$$E_{\text{cue}} = \sum_t \left\{ \|\mathbf{M}_t - \pi_{\text{seg}}(\mathbf{s}_t)\|^2 + \beta \|\mathbf{F}_t - \pi_{\text{flow}}(\mathbf{s}_t)\|^2 \right\} \tag{4}$$

where β is the trade-off weight; $\pi_{\text{flow}}(\mathbf{s}_t)$ is the rendered the 2D flow map for each pixel given the deformed shape $\mathbf{s}_t$; $\pi_{\text{seg}}(\mathbf{s}_t)$ is the rendered the object mask. Similar to previous inverse rendering work [61], the object-camera transform is encoded as the root node transform.

Motion smoothness. This energy term encodes that motions of animals should be smooth and continuous across frames. We impose a constraint to ensure that there is little difference in joint angles of one bone from two consecutive frames should be slight. We implement this by computing the multiplication of the joint quaternion at the current frame and the inverse of the joint quaternion at the next frame, which should be close to an identity quaternion $\mathbf{q} = (0, 0, 0, 1)$:

$$E_{\text{smooth}} = \sum_t \sum_k \|(\mathbf{q}_k^t)^{-1} \circ \mathbf{q}_k^{t+1} - \mathbf{q}\|^2 \tag{5}$$

Symmetry offset. This term encourages the resulting shape in canonical space to be symmetric at the reference frame. It is inspired by the fact that most animals in the real world are symmetric if putting them into a certain canonical pose (e.g. 'T-pose' for bipedal animal). In practice, we enforce this property at the canonical shape (where the joint angles are all zero). Specifically, we calculate the chamfer distance for measuring the similarity between the set of vertices $\{\mathbf{v}_i\}$ under the canonical shape and its reflection:

$$E_{\text{symm}} = \mathcal{L}_{\text{cham}}((\{\mathbf{v}_i\}), \mathbf{H}(\{\mathbf{v}_i\})) \tag{6}$$

where $\mathbf{H}$ is the Householder reflection matrix.

3.4 Inference

We reason the skeletal shape by minimizing the energy defined in Eq. 3. Our optimization variables include the vertex position $\mathbf{v}_i$ at the canonical pose, the rigging weight $\mathbf{w}_i$, bone length $\mathbf{b}_k$ as well as the joint angles $\mathbf{q}_k^t$. All the variables except joint angles are shared across time, and joint angles are optimized per frame.

Initialization. We initialize the vertex position, bone length, and the rigging weight $\mathbf{v}_i, \mathbf{b}_k, \mathbf{w}_i$ of canonical shape using the retrieved template character. The skeleton tree structure of this character is also taken as the basis of our skeletal parameterization. We initialize all the joint rotations as a unit quaternion.

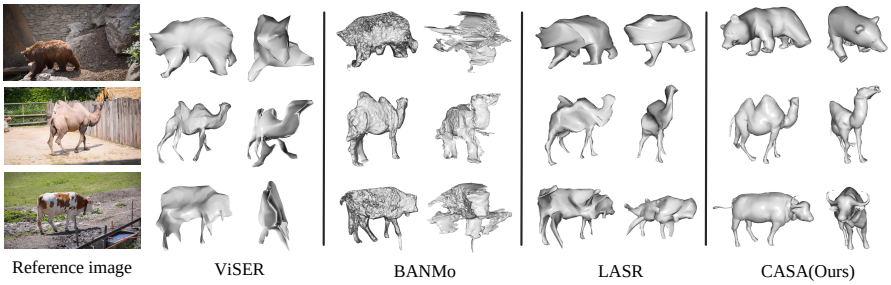

Figure 7: **Qualitative comparison of our method on real-world videos.** The reconstruction results of the reference image are shown in two viewpoints.

Reparameterization. Optimizing each vertex position offers flexibility yet might lead to undesirable mesh due to the lack of regularization. We use a neural displacement field to re-parameterize the vertex deformation to incorporate regularization, such as smooth and structured vertex deformation. We use a coordinate-based multi-layer perceptron (MLP) to define this displacement field $\mathcal{V}$. In addition, we incorporate a global scale scalar $\mathbf{u}$ in our framework to handle the shape misalignment between our initialization and the target. The position of each vertex is defined as:

$$\mathbf{v}_i' = u\mathbf{v}_i + \mathcal{V}_\theta(u\mathbf{v}_i) \tag{7}$$

where $u\mathbf{v}_i$ is the scaled position at the canonical pose, $\mathcal{V}_\theta(u\mathbf{p}_i)$ represents the vertex offset parameterized by θ, and $\mathbf{p}_i'$ is updated position for vertex i. During inference, instead of directly optimizing $\mathbf{v}_i$, we will only optimize the parameters of the displacement network. This displacement field reparameterization allow us to smoothly deform canonical shape during inference. In practice we find this implicit regularization compares favorably over explicit smoothness terms such as as-rigid-as-possible and laplacian smoothness.

Optimization. The energy function is fully differentiable and can be optimized end-to-end. We use Adam optimizer [18] to learn all the optimization variables. We adopt a scheduling strategy to avoid getting stuck at a local minimum. Specifically, we firstly optimize the mesh scaling factor based on silhouette correspondences. We then jointly update the bone length scale, joint angles, and the neural offset field by minimizing the energy function defined in Eq. 3.

4 PlanetZoo Dataset

Training and validating animatable 3D shape reconstruction require to have a large amount of non-rigid action sequences with ground truth 3D shapes and articulation structures. To achieve this goal, we construct a synthetic non-rigid dataset PlanetZoo, which consists of a large number of animated animals with textures and skeletons. We extracted those characters from the zoo simulator game Planetzoo [30] where hundreds of different animals are available.

Data Generation. In Planetzoo, the shapes are represented as a mesh. We use Cobra-tools [11] to extract those meshes along with their skeleton. With the extracted mesh and skeleton, we generate RGB maps, segmentation masks and optical flow for each frame. All data is obtained using Blender.

Assets. To set a diverse environment, we randomly set the background with public released HDRI pictures, providing a background light, and set floor textures with materials got from ambientCG[1]. To reduce the gap between synthetic data and real-world data, we randomly set the location of the light source along with its strength to simulate different environments in real-world situations.

Articulation. To generate realistic action sequences, we randomly change camera locations between every 12 frames, resulting in a constant view-point change following the animal. The camera is allowed to rotate at a certain angle, ranging from $15°$ to $22.5°$. In order to obtain animated animal sequences, for every 12 frames, 8 bones are selected from the skeleton tree with a transformation attached. The angle value of rotation for each bone is sampled from a uniform distribution. By doing this, we are able to cover the whole action space, providing diverse action sequences.

Rendering. We generate segmentation masks, optical flow, depth map, camera parameters, and photo-realistic RGB images using Vision Blender [7]. The physical-based renderer provided by Blender is capable of showing details such as fur, making the rendering results more realistic. For

[1] https://ambientcg.com/

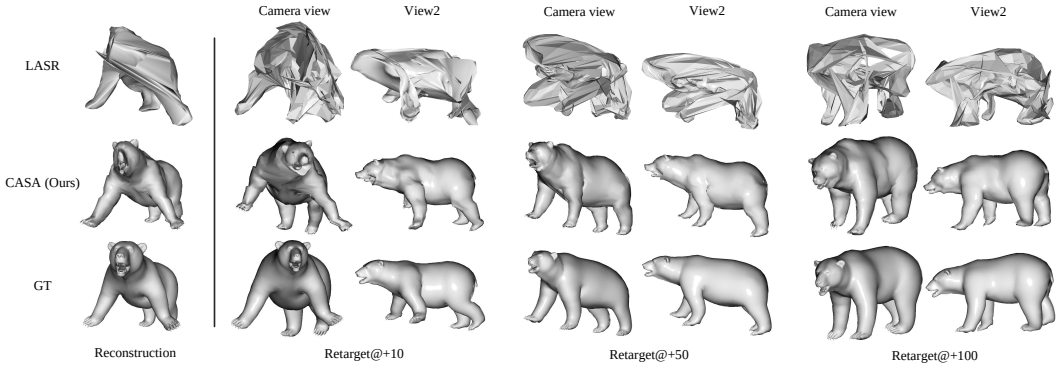

Figure 8: **Reanimation results.**

each animal in the dataset, 180 frames are rendered. Fig. 5 depicts a snapshot of the assets and rendering images from our dataset, demonstrating the diversity and quality.

5 Experiments

In this section, we first introduce our experimental setting (§ 5.1). We then compare CASA against a comprehensive set of articulated baselines in various reconstruction and articulation metrics on both simulation (§ 5.2) and real-world datasets (§ 5.3). Finally, we demonstrate our inferred shape can be used for downstream reanimation tasks (§ 5.4).

5.1 Experimental Setup

Benchmarks. We validate our proposed method on two datasets. Our proposed photorealistic rendering dataset PlanetZoo as well as a real-world animal video dataset DAVIS[39], containing multiple real animal videos with mask annotations. For dataset PlanetZoo, we choose 24 animals in 249 total animals as our testing and the rest as validation and training. The testing dataset includes diverse articulated animals from unseen categories, including multiple quadrupeds, bipeds, birds categories, as well as unseen articulated topology such as pinnipeds.

Metrics. We measure the reconstruction quality by Intersection over Union (IOU) and Chamfer distance, as well as skinning distance, joint distance and re-animation quality on PlanetZoo.

1) *IOU* measures the volumetric similarity between two shapes. We first voxelized reference shape and predicted shape into occupancy grids and then calculate the IOU ratio.

2) *Chamfer Distance* computes the bidirectional vertex-to-vertex distances between the reference shape and the generated shape.

3) *Joint* is the symmetric Chamfer Distance between joints. We evaluate CD-J2J following [58]. Given a test shape, we compute the Euclidean distance between each joint and its nearest joint in the reference shape, then divide it by the total joint number.

4) *Skinning* distance measures the similarity between the skinning weights. We first extract vertices associated with each joint using skinning weight. For each pair of joints from GT and prediction, we calculate the Chamfer distance between their associated vertices. Finally, we exploit the Jonker-Volgenant algorithm to find the minimum distance matching between prediction and reference.

5) *Re-animation* measures how well we can re-pose an articulated shape to a target shape. Specifically, we minimize the Chamfer Distance between the reference shape and the predicted shape by optimizing the joint angles of each bone for skeletal shape, or the rigid transform for control point-based shape. We consider this a holistic metric that jointly reflects the quality of shapes, skinning, and skeleton.

Baselines. We compare with state-of-the-art approaches for single monocular-video articulated shape reconstruction including LASR [60], ViSER [61], and BANMo [62], which are methods that reconstruct articulated shapes on monocular videos. Similar to our method, they utilize 2D supervision of videos for training including segmentation masks and optical flows. We download the open-source version of them from github. For the input data, we use our ground truth silhouette either in PlanetZoo dataset or DAVIS[39] dataset. We follow the optimization scripts in their code and get the baseline results.

3D reconstruction from a monocular video inevitably brings scale ambiguities. To alleviate the issue of scale differences, for every predicted shape among all the competing baselines, we conduct a line search to find the optimal scale that maximizes the IOU between the reference and predicted shape.

5.2 Results on PlanetZoo

We present quantitative results on PlanetZoo in Tab. 1. CASA achieves higher performance on metrics mIOU and mCham, demonstrating a shape with higher fidelity. Our approach also outperforms other competing methods in reanimation, demonstrating the superior performance of holistic articulated shape reasoning and potential for downstream reanimation tasks. CASA compares less favorably to LASR in the joint distance and skinning quality. We conjecture this is due to additional degrees of freedom benefits brought by joint control point articulation.

Fig. 4 and Fig. 6 depicts a few qualitative examples of all competing algorithms on PlanetZoo. We show the reconstructed mesh from both the camera view and the opposite view. The figure shows that CASA can recover accurate shape across various view angles under partial observation. In contrast, baselines fail to reconstruct reliable 3D mesh when the objects are partially observed. In addition, CASA also produces meshes with higher fidelity and local details in the visible region.

5.3 Real-world Reconstruction

We demonstrate qualitative results of the competing algorithms on the real-world animal video dataset DAVIS [39] in Fig 7 and Fig. 4. This figure shows that both our method and baselines can get good reconstruction results from the camera perspective. But both two baselines fail to reconstruct those unseen parts. In contrast, CASA can reliably reason the shape and articulation in the unseen regions, thanks to its symmetry constraints, skeletal parameterization, and 3D template retrieval.

5.4 Reanimation

We now show how the reconstructed articulated 3D shape can be retargeted to new poses. Given an inferred skeletal shape and the target GT shape at a different articulated pose, we apply an inverse kinematic to compute the articulated pose to reanimate the inferred shape. Specifically, we optimize the articulated transforms such that the reanimated mesh is as close to the target in Chamfer distance. Joint quaternions are optimized for skeletal mesh, and the articulation transforms are optimized for the control-point-based method [60]. Fig. 8 shows a comparison between the retargeted meshes of our methods and LASR on PlanetZoo, with a GT target mesh as a reference. We observe that our retargeted meshes look realistic and accurate and preserve geometric details. Despite having more degree-of-freedoms in reanimation, LASR fails to produce realistic retarget results. Tab. 1 reports a quantitative comparison in chamfer distance between the retargeted and the reference mesh, demonstrating our approach outperforms competing methods by a large margin.

5.5 Limitations

While demonstrating superior performance on animal reconstruction, our method has a few remaining drawbacks. Firstly, the source asset bank restricts the diversity of the retrieved articulation topology. Bone length optimization partially alleviates such limitations through "shrinking" bones. Still, it cannot add new bones to the kinematic tree, e.g., we cannot create a spider from a quadrupedal template. Secondly, our method so far does not impose the constraint that bones are inside the mesh. We plan to tackle such challenges in the future.

6 Conclusion

In this paper, we propose CASA, a novel category-agnostic animation reconstruction algorithm. Our method can take a monocular RGB video and predict a 4D skeletal mesh, including the surface, skeleton structure, skinning weight, and the joint angle at each frame. Our experimental results show that the proposed method achieves state-of-the-art performances in two challenging datasets. Importantly, we demonstrate that we could retarget our reconstructed 3D skeletal character and generate new animated sequences.

References

[1] Benjamin Biggs, Thomas Roddick, Andrew Fitzgibbon, and Roberto Cipolla. Creatures great and SMAL: Recovering the shape and motion of animals from video. In *ACCV*, 2018. 2

[2] 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. 2

[3] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J Black. Dynamic FAUST: Registering human bodies in motion. In *CVPR*, 2017. 2

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

[5] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qizhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *ECCV*. 2020. 14

[6] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 2

[7] João Cartucho, Samyakh Tukra, Yunpeng Li, Daniel S. Elson, and Stamatia Giannarou. Visionblender: a tool to efficiently generate computer vision datasets for robotic surgery. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 2020. 2, 5, 7

[8] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3

[9] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *CORL*, 2017. 3

[10] Kaiwen Guo, Peter Lincoln, Philip Davidson, Jay Busch, Xueming Yu, Matt Whalen, Geoff Harvey, Sergio Orts-Escolano, Rohit Pandey, Jason Dourgarian, et al. The relightables: Volumetric performance capture of humans with realistic relighting. *ACM TOG*, 2019. 1

[11] HENDRIX-ZT2. cobra-tools. https://github.com/OpenNaja/cobra-tools, 2020. 7

[12] Yuan-Ting Hu, Jiahong Wang, Raymond A Yeh, and Alexander G Schwing. SAIL-VOS 3D: A Synthetic Dataset and Baselines for Object Detection and 3D Mesh Reconstruction from Video Data. In *CVPR*, 2021. 2

[13] Yuan-Ting Hu, Jiahong Wang, Raymond A Yeh, and Alexander G Schwing. Sail-vos 3d: A synthetic dataset and baselines for object detection and 3d mesh reconstruction from video data. In *CVPR*, 2021. 14

[14] Alec Jacobson and Olga Sorkine. Stretchable and twistable bones for skeletal shape deformation. In *SIGGRAPH Asia*, 2011. 4

[15] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017. 3

[16] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social interaction capture. *PAMI*, 2017. 1

[17] Angjoo Kanazawa, Shubham Tulsiani, Alexei A Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 3

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[19] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *CVPR*, pages 9799–9808, 2020. 5

[20] Filippos Kokkinos and Iasonas Kokkinos. Learning monocular 3d reconstruction of articulated categories from motion. In *CVPR*, 2021. 3

[21] Chen Kong and Simon Lucey. Deep non-rigid structure from motion. In *ICCV*, 2019. 3

[22] Nilesh Kulkarni, Abhinav Gupta, David F Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. In *CVPR*, 2020. 3

[23] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006. 4

[24] Jiahui Lei and Kostas Daniilidis. Cadex: Learning canonical deformation coordinate space for dynamic surface representation via neural homeomorphism. In *CVPR*, 2022. 3

[25] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM TOG*, 2017. 2

[26] Xueting Li, Sifei Liu, Shalini De Mello, Kihwan Kim, Xiaolong Wang, Ming-Hsuan Yang, and Jan Kautz. Online adaptation for consistent mesh reconstruction in the wild. In *NeurIPS*, 2020. 3

[27] Yang Li, Hikari Takehara, Takafumi, Taketomi, Bo Zheng, and Matthias Nießner. 4dcomplete: Non-rigid motion estimation beyond the observable surface. In *ICCV*, 2021. 2, 3

[28] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *ICCV*, 2019. 5

[29] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM TOG*, 2015. 1, 2, 4

[30] Ingo Lütkebohle. PlanetZoo Simulator. https://www.planetzoogame.com/, 2021. 2, 7

[31] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *CVPR*, 2022. 5

[32] Marko Mihajlovic, Yan Zhang, Michael J Black, and Siyu Tang. LEAP: Learning articulated occupancy of people. In *CVPR*, 2021. 2, 3

[33] Richard A. Newcombe, Dieter Fox, and Steven M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *CVPR*, 2015. 1, 4

[34] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Occupancy flow: 4d reconstruction by learning particle dynamics. In *ICCV*, 2019. 1

[35] Atsuhiro Noguchi, Umar Iqbal, Jonathan Tremblay, Tatsuya Harada, and Orazio Gallo. Watch it move: Unsupervised discovery of 3D joints for re-posing of articulated objects. In *CVPR*, 2022. 3

[36] Ahmed AA Osman, Timo Bolkart, and Michael J Black. Star: Sparse trained articulated human body regressor. In *ECCV*, 2020. 2

[37] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *ICCV*, 2021. 2

[38] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, 2019. 2

[39] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Computer Vision and Pattern Recognition*, 2016. 8, 9

[40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 5

[41] Zhongzheng Ren*, Xiaoming Zhao*, and Alexander G. Schwing. Class-agnostic reconstruction of dynamic objects from videos. In *NeurIPS*, 2021. 1, 2, 3

[42] Stephan R Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *ICCV*, 2017. 14

[43] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *ICCV*, 2021. 3

[44] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *SIGGRAPH ASIA*, 2017. 2

[45] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *ICCV*, 2019. 2

[46] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *CVPR*, 2020. 2

[47] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J Black. Scanimate: Weakly supervised learning of skinned clothed avatar networks. In *CVPR*, 2021. 2, 3

[48] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *ICCV*, 2019. 3

[49] Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne Tchapmi, et al. igibson 1.0: A simulation environment for interactive tasks in large realistic scenes. In *IROS*, 2020. 3

[50] Vikramjit Sidhu, Edgar Tretschk, Vladislav Golyanik, Antonio Agudo, and Christian Theobalt. Neural dense non-rigid structure from motion with latent space constraints. In *ECCV*, 2020. 3

[51] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 3

[52] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 5

[53] Shubham Tulsiani, Nilesh Kulkarni, and Abhinav Gupta. Implicit mesh reconstruction from unannotated image collections. *arXiv preprint arXiv:2007.08504*, 2020. 3

[54] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *CVPR*, 2022. 5

[55] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *IROS*, 2020. 3

[56] Shangzhe Wu, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. Dove: Learning deformable 3d objects by watching videos. *arXiv preprint arXiv:2107.10844*, 2021. 2, 3

[57] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *CVPR*, 2019. 2

[58] Zhan Xu, Yang Zhou, Evangelos Kalogerakis, Chris Landreth, and Karan Singh. Rignet: Neural rigging for articulated characters. *ACM TOG*, 2020. 8

[59] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. *NeurIPS*, 32, 2019. 6

[60] 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, 3, 4, 5, 6, 8, 9

[61] 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, 3, 4, 6, 8

[62] Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. Banmo: Building animatable 3d neural models from many casual videos. In *CVPR*, 2022. 2, 8

[63] Ze Yang, Shenlong Wang, Sivabalan Manivasagam, Zeng Huang, Wei-Chiu Ma, Xinchen Yan, Ersin Yumer, and Raquel Urtasun. S3: Neural shape, skeleton, and skinning fields for 3d human modeling. In *CVPR*, 2021. 2

[64] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *ECCV*, 2020. 3

[65] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *ICCV*, 2019. 2

[66] Silvia Zuffi, Angjoo Kanazawa, Tanya Berger-Wolf, and Michael J Black. Three-D Safari: Learning to Estimate Zebra Pose, Shape, and Texture from Images In the Wild. In *ICCV*, 2019. 2

[67] 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

[68] 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, 4

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] we will release the code and data upon acceptance
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We use eight GTX 2080 Ti and RTX A6000 GPUs.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] upon acceptance
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] BADJA is open-sourced; PlanetZoo has commercial license, following prior practices in our community [42, 5, 13], we will ask users to purchase the game before accessing the dataset.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] Both datasets do not contain personal identifiable information.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]