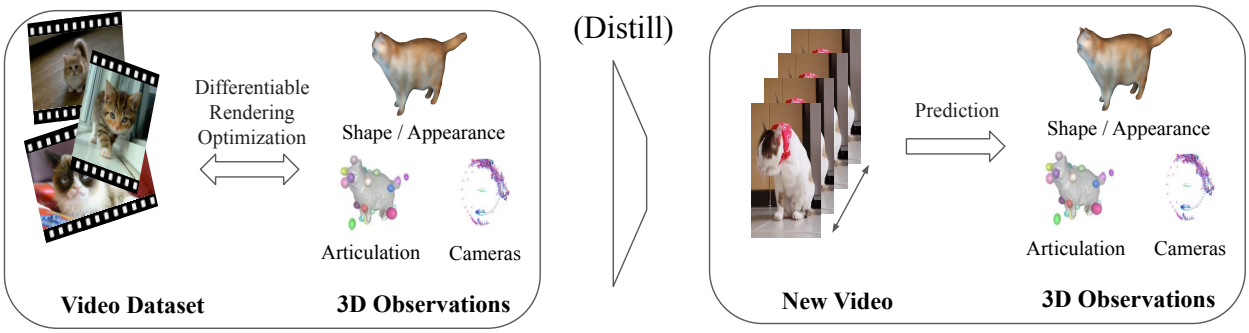
Distilling Neural Fields for Real-Time Articulated Shape Reconstruction

Jeff Tan
Carnegie Mellon University
jefftan@andrew.cmu.edu

Gengshan Yang
Carnegie Mellon University
gengshay@andrew.cmu.edu

Deva Ramanan
Carnegie Mellon University
deva@cs.cmu.edu



Shape-and-Motion Optimizer
(*0.016 fps = 63 sec / frame*)

Feed-Forward Network
(*100 fps = 10ms / frame*)

Figure 1. By distilling knowledge from shape-and-motion reconstructors fitted to offline video data at scale, such as dynamic NeRFs [42], we present a method to train category-specific real-time video shape predictors, which output *temporally-consistent* object pose, articulation, and appearance given casual input videos. Compared to existing model-based methods for reconstructing humans and animals in motion [12, 16, 28], our method does not require pre-defined 3D templates or ground-truth 3D data to train.

Abstract

Prior work for articulated 3D shape reconstruction often relies on pre-built deformable models (e.g. SMAL or SMPL) or slow per-scene optimization through differentiable rendering (e.g. dynamic NeRFs). Such methods fail to support arbitrary object categories, or are unsuitable for real-time applications. We present a method that builds articulated 3D models from videos in real time without per-scene optimization or expensive real-world annotations. While it is challenging to collect accurate and 3D training data for arbitrary deformable object categories in a scalable way, our key insight is to leverage off-the-shelf video-based dynamic NeRFs as 3D supervision to train a feed-forward temporal network, turning 3D pose and shape prediction into a supervised distillation task. Our temporal network uses articulated bones and blend skinning to represent arbitrary deformations, and is self-supervised on large-scale video datasets without requiring 3D shapes or poses as input. Through distillation, our network learns to 3D-reconstruct unseen articulated objects at interactive frame rates. On human and pet reconstruction datasets, our method shows higher-fidelity 3D reconstructions than prior real-time methods for animals, with the ability to render realistic images at novel viewpoints and poses.

1. Introduction

We are interested in building high-quality animatable models of articulated 3D objects from videos in real time. One promising application is virtual and augmented reality, where the goal is to create high-fidelity 3D experiences from images and videos captured live by users. For rigid scenes, structure from motion (SfM) and neural rendering techniques can be used to build accurate 3D models of cities and landmarks from large image collections [1, 18, 31], such as photos from the Internet. For articulated objects such as family members or pets, many works leverage targeted category templates such as SMPL for humans [16] and SMAL [4] for quadruped animals to parameterize the range of possible motions. These methods can be trained on large-scale shape and motion datasets; however they rely on parametric body template models built from extensive real-world 3D scans, which cannot easily be generated for diverse object

categories in the wild such as clothed humans or pets with distinctive morphologies, which are commonly the focus of user content.

Inspired by the breakthrough success of neural radiance fields [19], many works reconstruct arbitrary articulated objects in an analysis-by-synthesis framework [24, 25, 27, 34, 42] by defining 3D warping fields and establishing long-range correspondences on top of canonical shape and appearance models. These methods produce high-quality reconstructions of arbitrary object categories without 3D data or pre-defined template models, but the output representations are scene-specific and often require *hours* to compute from scratch on unseen videos - an unacceptable cost for real-time VR/AR applications. We are thus interested in dynamic 3D reconstruction algorithms that achieve the best of both worlds: the speed and data scalability of template-based models, combined with the quality and generalization ability of dynamic NeRFs.

In this paper we present a real-time method for inferring animatable 3D models from RGB videos. Our key insight is remarkably simple: *we train category-specific feed-forward 3D predictors at scale by self-supervising them with dynamic NeRF "teachers" fit to offline video data*. By leveraging scene-fitted dynamic NeRFs for 3D supervision at scale, our method learns a feed-forward predictor for appearance, 3D shape, and articulations of non-rigid objects from videos. Our learned 3D models use linear blend skinning to express articulations, allowing it to be animated by manipulating bone transformations. As recovering freely moving non-rigid objects from monocular video is a highly under-constrained problem where epipolar constraints are not directly applicable [5], we address three key challenges in our work: (1) how to supervise our feed-forward models on the internal representations of dynamic NeRFs; (2) how to represent 3D appearance and deformation with respect to a canonical space; and (3) how to produce temporally consistent predictions of pose, articulation, and appearance.

2. Related Work

Template-Based Dynamic Reconstruction A large body of work uses parametric body models [16, 44, 45] to recover 3D shape and motions at test-time for human and animal reconstruction, given a single image as input [2, 3, 12, 30]. These models are built from registered 3D scans of real humans or toy animals, and while these methods achieve great success in reconstructing categories for which large volumes of ground-truth 3D data are available (especially true for human reconstruction), it is challenging to apply these methods to arbitrary categories with distinct morphologies and limited 3D data, such as humans in diverse clothing, animals, and plants. Our work aims to generalize these approaches to arbitrary articulated object categories without requiring ground-truth 3D data or pre-registered 3D scans during training.

Template-Free Dynamic Reconstruction A number of methods attempt to build deformable 3D models without templates by recovering shapes and poses from internet-scale 2D image collections, using weak supervision such as keypoints and object silhouettes from off-the-shelf models or human annotators [6, 9, 13, 32]. As it is inherently ambiguous to reconstruct 3D outputs from the sparse and limited 2D observations available in images, these methods must leverage strong data priors and apply heavily regularization to ensure reasonable outputs, often resulting in blurry or oversmoothed shapes and textures. Leveraging the temporal context available in videos can help these methods learn temporally consistent results [36], however the output quality is still low perhaps due to limited supervision or over-regularized shape and motion.

Dynamic Neural Radiance Fields Neural implicits have emerged as a powerful paradigm for performing 3D reconstruction using 2D image supervision, achieving state-of-the-art fidelity on both static and dynamic scenes. Although historically limited to rigid scenes with known cameras [18, 19, 34], recent works extend NeRF to dynamic scenes by introducing 3D warping fields or blend-skinning parameterizations to deform view-space points to a canonical space over time [15, 24, 27, 29]. By leveraging the temporal context available in video collections to disentangle shape from motion; LASR, ViSER, and BANMo are able to learn high-fidelity animatable 3D models from one or many casually collected videos capturing the same object instance [40–42]. However, dynamic NeRFs are slow and often require optimization from scratch on unseen videos. Our aim is to leverage their high-quality outputs to supervise a faster and more lightweight architecture for articulated 3D reconstruction.

3. Method

In order to train a category-specific feed-forward 3D predictor from a dynamic NeRF teacher model, we combine a single-frame image encoder that regresses pose, shape, and appearance from images, with a temporal encoder that reasons about these predictions over time. In this section, we describe the dynamic NeRF model that our method is built upon, our representation of articulated objects, as well as our network architecture, training procedure, and losses.

3.1. Problem Setup

Given an input video $V = \{I_t\}_{t=1}^{T}$ of length T, where each frame is cropped to the bounding box surrounding an articulated object, we train a feed-forward network to predict the pose, articulations, and appearance of the articulated object, which are used to compute a posed and textured object model. Our network is supervised on the pseudo-ground truth results of BANMo [42], a learning

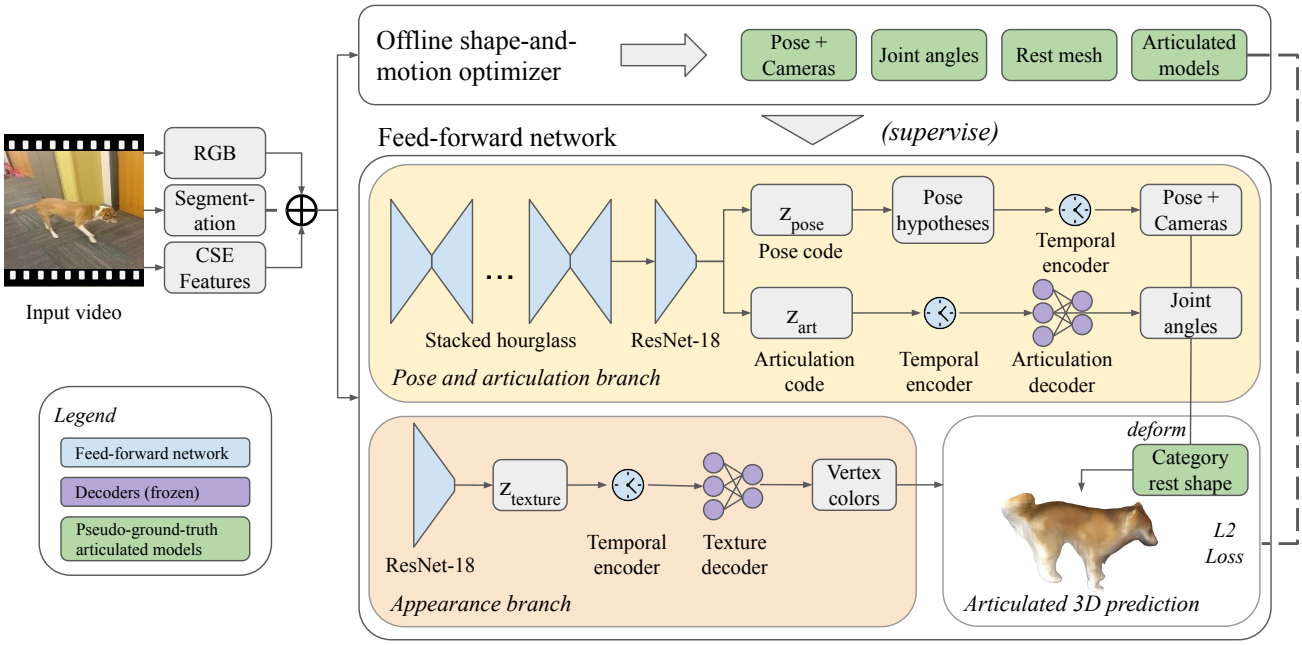


Figure 2. **Architecture**. After using an offline shape-and-motion predictor (in our case, BANMo [42]) to compute pseudo-ground-truth 3D shapes, articulations, and textures from videos, we train a supervised feed-forward network (shown in blue) to predict the *same* articulations and textures in a single efficient forward pass. To simplify the learning task, the outputs of our feed-forward network are in high-dimensional latent space: we use the offline shape-and-motion predictor's frozen decoder networks (shown in purple) to convert articulation codes into joint angles and texture codes into 3D textures. Spatial $\mathcal{L}_2$ losses are used to supervise the articulated 3D output models against the pseudo-ground truth articulated models (shown in green).

framework that reconstructs animatable 3D models from casually collected videos, including shape, appearance, and time-varying articulations. BANMo models articulated objects with a canonical neural field that is deformed at each time step. Similar to BANMo, our method requires no predefined shape templates, registered cameras, or 3D ground truths. Fig. 2 summarizes our overall architecture, consisting of a single-frame regressor and temporal encoder.

3.2. Object Representation

Category-level shape. We model articulated objects as a canonical rest shape that is transformed by time-dependent poses and articulations. The rest shape is an instance-independent triangular mesh $M = (V, F)$ defined for each category that represents the mean instance shape in the rest space. The mesh has vertices $V \in \mathbb{R}^{|V|\times 3}$ and faces F. The faces F define the connectivity of vertices in the mesh and we assume it remains fixed. To initialize our rest mesh, we run marching cubes on a 64^3 grid to find the zero level set of the neural field $\mathbf{MLP_{SDF}}$ that serves as BANMo's implicit rest shape. Although explicit representations such as meshes can be less expressive than the implicit neural fields that model rest shape in BANMo, we find that a feed-forward predictor benefits from the speed of rasterization and the simplicity of regressing mesh vertex colors.

Per-instance pose and articulation. At each frame t, we model the object's pose and articulation similar to BANMo. The object's 3D pose is a root body transformation $\mathbf{G}^t \in SE(3)$, and we employ neural blend skinning [8, 37] to express object articulations. See Sec. 3.3 for more details. For each frame, the bone configurations for neural blend skinning are parameterized by a joint angle vector $\mathbf{B}^t \in SO(3)^b$, which is predicted by our feed-forward network from input video frames.

Appearance representation. As our mesh topology is fixed, we can simply model the object's appearance as an array of per-vertex colors $\mathbf{C}^t \in \mathbb{R}^{|V|\times 3}$ at each frame t. Following the standard rasterization pipeline, barycentric coordinates are used to interpolate the per-vertex colors across the surface of the triangle mesh during rendering.

Rendering. We assume a perspective camera projection defined by a fixed camera at the origin pointed along the negative-z axis. To render the object in video frame t, we apply the predicted root body transformation $\mathbf{G}^t \in SE(3)$ from the rest pose to the time-t pose. We then apply the blend skinning deformation specified by the predicted bone configuration $\mathbf{B}^t \in SO(3)^b$ using forward kinematics and dual-quaternion blend skinning.

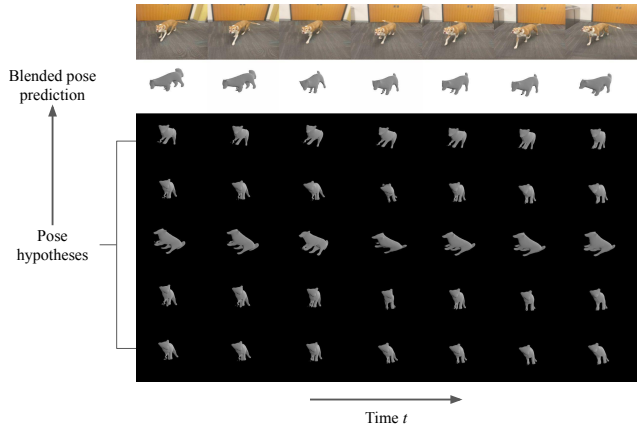


Figure 3. **Pose multiplexing during training**. Rather than training a feed-forward predictor to output a single root body pose, we predict a set of M pose hypotheses during training to overcome the discontinuous and multi-modal nature of pose optimization (here $M = 5$). Blending multiple pose hypotheses yields a more accurate pose prediction. **Top row**: Input video frames. **Second row**: Blended pose prediction, after temporal encoding. **Bottom row**: Multiplexed pose hypotheses outputted by image encoder.

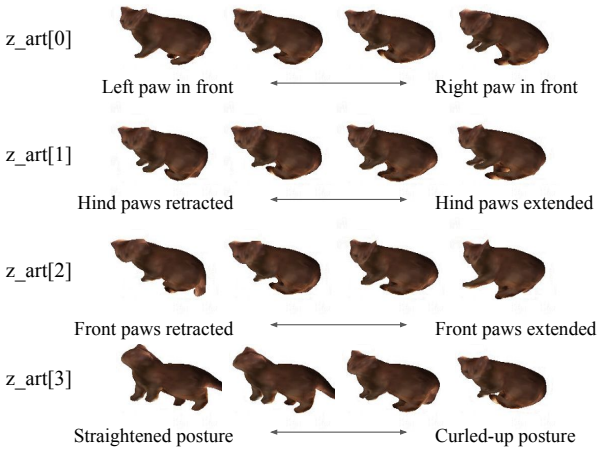


Figure 4. **Visualizing articulated pose codes**. Rather than training a feed-forward predictor to output high-dimensional (75) skeletal articulations, we predict a low-dimensional (16) pose code. We visualize the articulations captured by pose codes, by applying the teacher dynamic NeRF pose decoder network. Specifically, we visualize the first four principal components of the pose code space, which correspond to interpretable motions: (1) whether the left or right paw is in front, (2) whether the hind paws are retracted or extended, (3) whether the front paws are retracted or extended, and (4) whether the cat's posture is straight or curled-up.

3.3. Time-Varying Articulation via Blend Skinning

To represent articulated body motion, we use a neural blend skinning model as in BANMo to define a 3D warping field $\mathcal{W}^t$ on top of a category-level kinematic skeleton. After computing the skeleton's forward kinematics, each point is deformed by a weighted combination of per-bone transformations.

Category-level skeleton. Unlike color and 3D shape which are directly observable from imagery, an object's bone structure is much harder to infer. Automatic skeletal rigging methods [14, 23] rely heavily on shape priors, or are sensitive to input data. In contrast to 3D shape, which can be subject to morphological variations within each category, an object category's bone structure is largely fixed up to slight variations in bone lengths and body part scale. Thus, we can use readily available generic skeleton models of humans, quadrupeds, and other categories of interest to pre-specify a 3D skeleton for each category. The skeleton is defined by a tree structure with $B + 1$ fixed-length bones and B ball joints, where $B = 19$ for humans and $B = 26$ for cats and dogs.

Forward kinematics. Each bone b has a link transformation $\mathbf{L}_b \in SE(3)$ and joint transformation $\mathbf{J}_b^t \in SE(3)$. The link transformation $\mathbf{L}_b \in SE(3)$ is from the base to the end of the link. Here, the rotation magnitude is 0 and the translation is the fixed bone length. The joint transformation $\mathbf{J}_b^t \in SE(3)$ rotates link b by the current joint angle. Here, the translation magnitude is 0 and the joint angle $\mathbf{B}_b^t \in SO(3)$ is predicted per frame. For a given link b, the result is a sequence of alternating joint and link transformations from the base of the skeleton to the link's end, where $b_1, b_2, \ldots, b$ is the sequence of parent links up to b:

$$\mathbf{T}_b^t = \mathbf{J}_b^t \mathbf{L}_b \ldots \mathbf{J}_{b_2}^t \mathbf{L}_{b_2} \ldots \mathbf{J}_{b_1}^t \mathbf{L}_{b_1}$$

Blend skinning. From kinematic transformations $\mathbf{T}_b^t$ and root body pose $\mathbf{G}^t$, we apply dual-quaternion blend skinning to compute a 3D warping field:

$$\mathcal{W}(\mathbf{X})^t = (\textstyle\sum_b \mathbf{W}(\mathbf{X})_b^t \cdot \mathbf{T}_b^t) \cdot \mathbf{G}^t$$

As in BANMo, the skinning weights $\mathbf{W}(X)_b^t$ are specified by the softmax'ed distances between $\mathbf{X}$ and bone centers. Here, $\mathbf{X}$ are the coordinates of the rest mesh vertices.

3.4. Network Architecture

Single-frame image encoder. Given a video $V = \{I_t\}_{t=1}^T$ of length T, we encode the input frames with a convolutional stacked hourglass network [22], which uses repeated pooling and upsampling to process features across multiple scales and spatial locations. We use off-the-shelf PointRend [11] to compute object segmentations, and CSE [20, 21] to compute per-pixel features. Each RGB image is concatenated with features, then masked and cropped to 1.2x the tight bounding box of the object. We find that leveraging pretrained CSE features improves convergence speed over RGB inputs alone. We pass the stacked intermediate outputs of each hourglass module into a ResNet18 [7] network

Figure 5. **Qualitative results on humans (left), cats (middle), and dogs (right) in the test set**. From left to right for each category, we show the input images, articulated shape and texture predictions, as well as three different viewpoints of the predicted geometry. Our method operates on a video and predicts plausible shape, articulations, and textures in real time. Our predictions are aligned well with image evidence for challenging inputs, including uncommon poses and heavy occlusions.

that predicts latent vectors z_{pose} and z_{art}. z_{pose} is interpreted as the object's root body pose $\mathbf{G}^t \in SE(3)$, and z_{art} is interpreted as articulated joint angles $\mathbf{B}^t \in SO(3)^B$, where angles are represented as 6D rotations [43].

Pose branch. Due to the discontinuous and multi-modal nature of the space of possible root body poses, it can be difficult to approach the optimal pose through iterative gradient descent. Following [6], we use a pose decoder network $\mathbf{MLP}_{\text{pose}}$ to predict a set of M pose hypotheses $\mathbf{G}^t_{\{1,\ldots,M\}}$ and corresponding weights $w^t_{\mathbf{G}} \in \mathbb{R}^M$ from z_{pose}. Fig. 3 shows the variation over pose hypotheses at an early stage of training. To train our pose multiplex, we compute M articulated shapes and losses corresponding to each pose hypothesis in the multiplex, and weight each of these losses by their respective weights.

Articulation branch. Estimating 3D articulations from monocular images can be difficult due to depth ambiguities and occlusions. To resolve this, recent work on human pose estimation [35] leverages a normalizing flow articulation prior represented as an invertible neural network, trained on large-scale human motion capture datasets. To achieve a similar effect for arbitrary object categories without access to such data, we leverage BANMo's articulation priors by decoding z_{art} using BANMo's frozen articulation decoder $\mathbf{MLP}_{\text{art}}$. Fig. 4 visualizes the principal components of z_{art}'s latent articulation space. We find that perturbing z_{art} along these principal components causes the resulting articulated shape model to perform natural motions, such as curling up in a ball or putting one leg in front of the other.

Appearance branch. Separate from pose and articulation, we define an additional appearance branch to predict the per-vertex colors $\mathbf{C}^t \in \mathbb{R}^{|V|\times 3}$ of the articulated mesh at each frame. As the objects in the video are only partially observable at any given time, we must leverage data priors or information from nearby frames to output complete appearance predictions. BANMo represents global object appearances as a category-level neural field modulated by an environment code $z_{\text{color}} \in \mathbb{R}^{64}$. After predicting an environment code for each frame, we leverage these appearance priors by querying BANMo's neural field at the rest mesh vertex locations.

Temporal encoder. Predicting pose and shape from single images can be highly ambiguous due to motion blur, occlusions, and other uncertainties. To output temporally consistent results over long videos, we define a temporal encoder that shares information between z_{pose}, z_{art}, and z_{color} across multiple frames. We treat the pose multiplex z_{pose} as a single vector by concatenating along the M dimension. For simplicity and following prior work [10], our temporal encoder contains several layers of a 1D fully convolutional network that acts on a temporal window centered at time t.

3.5. Losses and Supervision

Optimization objective. Treating BANMo's outputs as pseudo-ground truth, our model can be trained in a standard supervised manner with geometry and color losses in 3D.

Table 1. **Datasets**. Summary of datasets used during training and evaluation. All videos are treated as casually collected monocular RGB videos, except when additional ground-truth meshes or masks are needed for evaluation.

Category	Total Videos	Total Frames	Length (mm:ss)	Train Frames	Test Frames
Humans	48	6.4k	10:38	4.8k	1.6k
Cats	77	11.7k	19:26	9.0k	2.7k
Dogs	88	9.7k	16:13	8.0k	1.7k

As all articulated meshes have the same topology, geometry loss is defined as the $\mathcal{L}_2$ error between predicted and actual vertex locations on the output articulated meshes. Color loss L_{color} is defined similarly on per-vertex RGB colors.

$$L_{\text{geom}} = \left\| \mathbf{X}^t - \widehat{\mathbf{X}}^t \right\|_2$$

To account for multiple possible solutions of inverse kinematics while solving for joint articulations, and to help learn meaningful deformations, we employ a joint loss defined as the geodesic distance between predicted and actual angles at each joint. We find that adding joint angle loss improves the deformation quality.

In summary, we train the image encoder and temporal encoder in stages with the following objective:

$$L = L_{\text{geom}} + L_{\text{joint}} + L_{\text{color}}$$

4. Experiments

4.1. Implementation Details

Our method is implemented in PyTorch, using a version of BANMomodified to support a skeleton deformation model. We use the AdamW optimizer and train the model for 120k iterations, taking around 24 hours on eight RTX-3090 GPUs. We use 224×224 images with batch size 56. Our image encoder contains 8 stacked hourglass blocks, and our temporal encoder uses a window size of 13 frames. All losses are weighted to have similar initial magnitudes.

Staged training. We adopt a two-stage training strategy at every epoch, to reduce the computational costs of evaluating an image encoder at every frame in the current time window when only a single frame will receive a gradient update. In the first stage, we treat our feed-forward network as a single-frame predictor. Without temporal encoding, we compute separate losses per frame and update the image encoder, caching values of z_{pose}, z_{art}, and z_{color} along the way. In the second stage, we run the temporal encoder on time windows of cached feature vectors and compute losses to update the temporal encoder. This two-stage training strategy reduces redundant evaluations of per-frame image encoder, and ensures that our temporal encoder acts as a regularizer on the per-frame feature vector outputs.

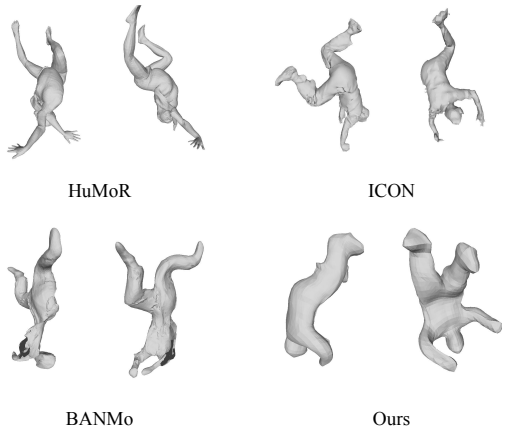


Figure 6. **Qualitative comparison on human sequences**. From top to bottom, we show the output of **top**: HuMoR, **top-middle**: ICON, **bottom-middle**: BANMo, and **bottom**: our method. Due to the unusual and highly challenging handstand pose, HuMoR [28] completely fails to make a plausible prediction. Both ICON [38] and BANMo output reasonable predictions, although BANMo's mesh looks physically implausible. While our mesh is well-articulated, it lacks realistic shape.

4.2. Datasets

We collect datasets for three categories: humans, cats, and dogs. For humans, we combine existing datasets from AMA [33], MonoPerfCap [39], DAVIS [26], and BANMo [42] to obtain 48 human videos with 6,382 images. AMA and MonoPerfCap are used for evaluation as they contain ground-truth meshes. For cats and dogs, we collect 77 cat videos and 88 dog videos from BANMo released data as well as the Pexels stock video website. The cat videos have 11,657 total frames and the dog videos have 9,734 total frames. In addition, we use an iPad Pro to capture an RGBD video for cats and an RGBD video for dogs to evaluate depth accuracy. Video frames are extracted at 10fps. Our datasets are summarized in Tab. 1.

4.3. Reconstructing Humans

Dataset. Following BANMo, we evaluate human reconstruction on the AMA dataset. AMA is a real-world dataset [33] containing 10 mesh sequences depicting 3 different humans performing various actions. The subjects wear loose-fitting clothes and perform challenging actions such as dancing, turning in circles, and performing a handstand. Although the AMA videos were captured in an 8-camera studio to enable ground-truth mesh extraction, we treat them as casually collected monocular videos in our model and do not use the camera intrinsics, camera extrinsics, or time synchronization.

Comparisons. We compare against template-free BANMo

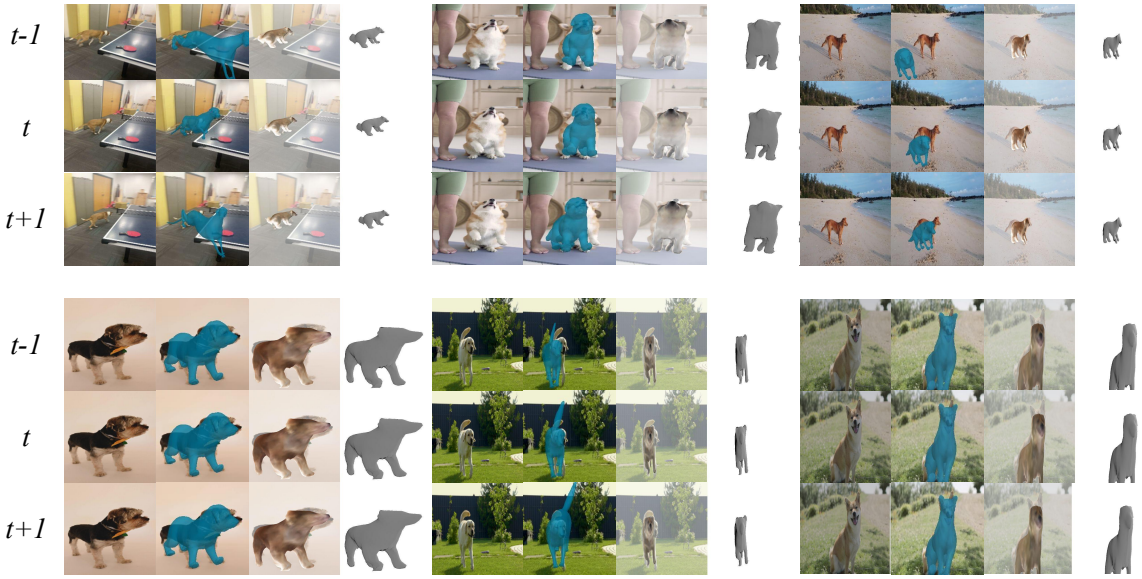


Figure 7. **Qualitative comparison against BARC**. From left to right in each column, we show **left**: the input image, **middle-left**: BARC's prediction, **middle-right**: our articulated shape and texture predictions, and **right**: our geometry predictions. Our method operates on videos and predicts plausible shape, articulations, and texture in real time. BARC operates on each video frame independently, resulting in jittery predictions when the dog is small or not clearly visible in frame (top row). While BARC's geometry and motion predictions are largely accurate, they can be occasionally biased for certain breeds (bottom left).

[42], as well as model-based methods HuMoR [28] and ICON [38]. BANMo fits an animatable 3D model to multiple monocular videos of an object instance by performing optimization based on differentiable rendering. We train BANMo on the same dataset as our model. As our model is supervised on BANMo pseudo-ground truths, BANMo's performance is an upper bound for our model. HuMoR is a human-specific temporal pose and shape predictor that performs test-time optimization on video sequences, leveraging OpenPose keypoint detection as well as large-scale human motion capture datasets for motion priors. ICON is the current SOTA for single-view human reconstruction, and it combines implicit functions with the SMPL human body model. To improve pose accuracy and reconstruction quality, ICON performs test-time optimization to fit surface normal predictions. All our baselines require far more processing time than our model, which runs in real time in a feed-forward manner.

Metrics. We report 3D chamfer distance and F-scores in Tab. 2, averaged across all frames. Chamfer distance computes the average distance between the ground-truth and predicted surface points by finding nearest neighbor matches. As chamfer distance can be sensitive to outliers, we also evaluate F-score at distance thresholds $d \in \{1\%, 2\%, 5\%\}$ to better quantify reconstruction error at different granularities. We scale predicted meshes by their view-space bounding box height to account for their unknown scale with respect to the registered ground-truths. Our model approaches the performance of BANMo and baselines, while requiring three orders of magnitude less compute at test time.

Qualitative Results. We show qualitative comparisons in Fig. 6. HuMoR completely fails to make a plausible prediction, while both ICON and BANMo have reasonable outputs. While our model outputs a reasonably articulated mesh, the shape is not physically realistic and blurs out the fine-grained geometry details of the person. We hypothesize that this failure case occurs due to the entanglement between articulation and per-instance morphology.

4.4. Reconstructing Cats and Dogs

Dataset. We evaluate cat and dog reconstruction on two RGBD pet videos, as well as a held-out test set of casual pet videos sourced from BANMo. The dataset contains challenging motions such as jumping off of a chair onto the ground, rapid turns, and repetitive scratching motions.

Comparisons. We compare against BANMo [42] and BARC [30], a model-based approach that is the current SOTA for dog shape and pose estimation from images. BARC trains a feed-forward network using synthetic SMAL dog models [4] and images with keypoint labels, leveraging breed losses as additional supervision. SMAL uses manual rigging and registration to fit a body model to 3D animal toys. As BARC is image-based, we run it separately on each video frame.

Metrics We report the root mean square depth error for all foreground pixels in Tab. 3, averaged across all frames. We render a synthetic depth map per frame, and following [17],

Table 2. **Quantitative results on `AMA` sequences.** 3D chamfer distance (cm, ↓) and F-score (%, ↑) for articulated meshes, averaged over all frames. We resize the 3D ground-truth such that the longest edge of the axis-aligned bounding box is 2m. Both our model and BANMo are trained on 48 videos spanning existing human datasets. Our model approaches the performance of BANMo and baselines while requiring three orders of magnitude less compute at test time. Other baselines are trained on 3D human data, rely on the SMPL body model, or leverage expensive test-time optimization to improve results. For each method, we also report the total processing time (in seconds) per frame. Since BANMo performs test-time optimization on multiple videos, we take the average over all input video frames. The best results are in bold.

Method	Time	samba				bouncing				handstand			
		CD	F@1%	F@2%	F@5%	CD	F@1%	F@2%	F@5%	CD	F@1%	F@2%	F@5%
HuMoR	42s	**9.8**	23.3	47.4	83.6	11.4	21.1	46.1	83.2	21.9	13.5	28.8	58.5
ICON	63s	10.1	18.9	39.9	**85.2**	**9.7**	**27.8**	**53.5**	**86.4**	**14.1**	**23.5**	**45.2**	**75.6**
BANMo	43s	10.3	25.3	48.8	83.6	11.6	26.7	50.8	81.5	14.2	22.5	44.1	74.7
Ours	**10ms**	10.3	**26.1**	**50.4**	84.1	12.0	24.4	48.1	79.9	17.3	17.9	35.9	67.2

Table 3. **Quantitative results on `RGBD-pet` sequences.** Root mean square depth error (↓) and depth accuracy (%, ↑) for all foreground pixels in the depth map, averaged over all frames. We also report the total processing time per frame in seconds on a RTX-3090 GPU. To account for the unknown global scale factor of the depth sensor, we align the median of the predicted depth map and the ground-truth. Our model is trained on 88 casually collected dog videos, and approaches the quality of BANMo while being three orders of magnitude faster at test time. The best results are in bold.

Method	Time	dog				cat			
		RMSE	Acc-1%	Acc-2%	Acc-5%	RMSE	Acc-1%	Acc-2%	Acc-5%
BANMo	54s	**0.0411**	**28.6**	**45.3**	**72.7**	**0.0757**	**27.7**	**47.4**	**76.5**
Ours	**10ms**	0.0569	17.7	31.6	56.6	0.1298	19.1	33.8	61.4

we account for the unknown global scale factor between depth maps by aligning the median rendered and ground-truth depths at each frame:

$$s_i = \underset{x}{\text{median}} \left\{ D_i^{pred}(x) / D^{gt}(x) \right\}$$

Qualitative Results We show qualitative results comparing to BARC in Fig. 7. BARC performs well at predicting coarse shape and deformations, and more faithfully captures the fine motion and geometry details of the dog when it is positioned well in frame. However, as BARC entangles shape and breed, we find that BARC may predict biased shapes for certain breeds. For example, in the bottom left of Fig. 7, BARC predicts a rounded back but the back is flat in reality. For the inputs on the top row, BARC's single-frame architecture makes jittery predictions from frame to frame, while our temporal architecture enforces consistency and prevents large discontinuities in pose and deformation. In the middle top image, which is particularly challenging because the dog is standing on its hind legs, both our model and BARC fail to register that the dog is lifting its front paws. As our model does not currently disentangle articulation and morphology variation between breeds, incorporating breed and/or instance information would likely improve our capacity to represent fine motion and geometry details that differ between dogs.

5. Discussion

We have presented a method to train category-specific feed-forward video shape predictors by distilling knowledge from dynamic NeRF "teachers", which are fitted to offline video data at scale. Our temporal architecture predicts consistent object pose, articulation, and appearance in real time, qualitatively outperforming existing feed-forward predictors for dog shape and pose. As a result, we approach the fidelity of test-time fitting methods while using three orders of magnitude less computation. Our method is general and we demonstrate video reconstruction results on humans, cats, and dogs, with the capability to reconstruct categories beyond these.

Limitations: As our model is trained on BANMo pseudo ground-truths, we are upper-bounded by BANMo's performance and reconstruction fidelity. Entangling articulations with per-instance object morphologies can result in blurry and over-smoothed shapes, as shown in Fig. 6 and 7. Incorporating breed or instance-specific losses as in [30] may help improve reconstruction fidelity.

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, and Richard Szeliski. Building rome in a day. *ICCV*, 2009. 1

[2] Marc Bager, Yufu Wang, Adarsh Modh, Ammon Perkes, Nikos Kolotouros, Bernd Pfrommer, Marc Schmidt, and Kostas Daniilidis. 3d bird reconstruction: A dataset, model, and shape recovery from a single view. *ECCV*, 2020. 2

[3] Benjamin Biggs, Ollie Boyne, James Charles, Andrew Fitzgibbon, and Roberto Cipolla. Who let the dogs out: 3d animal reconstruction with expectation maximization in the loop. *ECCV*, 2020. 2

[4] Benjamin Biggs, Thomas Roddick, Andrew Fitzgibbon, and Roberto Cipolla. Creatures great and smal: Recovering the shape and motion of animals from video. *ACCV*, 2018. 1, 7

[5] Christoph Bregler, Aaron Hertzmann, and Henning Biermann. Recovering non-rigid 3d shape from image streams. *CVPR*, 2000. 2

[6] Shubham Goel, Angjoo Kanazawa, and Jitendra Malik. Shape and viewpoints without keypoints. *ECCV*, 2020. 2, 5

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv*, 2015. 4

[8] Alec Jacobson and Olga Sorkine. Stretchable and twistable bones for skeletal shape deformation. *SIGGRAPH Asia*, 2011. 3

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

[10] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. *CVPR*, 2019. 5

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

[12] Nikos Kocabas, Muhammed an Athanasiou and Michael J. Black. Video inference for human body pose and shape estimation. *CVPR*, 2020. 1, 2

[13] Filippos Kokkinos and Iasonas Kokkinos. To the point: Correspondence-driven monocular 3d category reconstruction. *NeurIPS*, 2021. 2

[14] Binh Huy Le and Zhigang Deng. Robust and accurate skeletal rigging from mesh sequences. *ACM TOG*, 2014. 4

[15] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. *CVPR*, 2021. 2

[16] Matthew Loper, Naureen Mahmood, Javier Romero, gerard Pons-Moll, and Michael J. Black. Smpl: A skinned multi-person linear model. *SIGGRAPH Asia*, 2015. 1, 2

[17] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *SIGGRAPH*, 2020. 7

[18] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild. *CVPR*, 2021. 1, 2

[19] 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. *ECCV*, 2020. 2

[20] Natalia Neverova, David Novotny, Vasil Khalidov, Marc Szafraniec, Patrick Labatut, and Andrea Vedaldi. Continuous surface embeddings. *NeurIPS*, 2020. 4

[21] Natalia Neverova, Artsiom Sanakoyeu, Patrick Labatut, David Novotny, and Andrea Vedaldi. Discovering relationships between object categories via universal canonical maps. *CVPR*, 2021. 4

[22] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. *ECCV*, 2016. 4

[23] 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. 4

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

[25] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B. Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *SIGGRAPH Asia*, 2021. 2

[26] 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. *CVPR*, 2016. 6

[27] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. *CVPR*, 2020. 2

[28] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. Humor: 3d human motion model for robust pose estimation. *ICCV*, 2021. 1, 6, 7

[29] Non rigid Neural Radiance Fields: Reconstruction and Novel View Synthesis of a Dynamic Scene from Monocular Video. Edgar, tretschk and teawri, ayush and golyanik, vladislav and zollhofer, michael and lassner, christoph and theobalt, christian. *ICCV*, 2021. 2

[30] Nadine Rueegg, Silvia Zuffi, Konrad Schindler, and Michael J. Black. Barc: Learning to regress 3d dog shape from images by exploiting breed information. *CVPR*, 2022. 2, 7, 8

[31] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: Exploring photo collections in 3d. *SIGGRAPH*, 2006. 1

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

[33] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popovic. Articulated mesh animation from multi-view silhouettes. *ACM TOG*, 2008. 6

[34] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv*, 2021. 2

[35] Tom Wehrbein, Marco Rudolph, Bodo Rosenhahn, and Bastian Wandt. Probabilistic monocular 3d human pose estimation with normalizing flows. *ICCV*, 2021. 5

[36] Shangzhe Wu, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. Dove: Learning deformable 3d objects by watching videos. *arXiv*, 2022. 2

[37] Yuefan Wu, Zeyuan Chen, Shaowei Liu, Zhongzheng Ren, and Shenlong Wang. Casa: Category-agnostic skeletal animal reconstruction. *CVPR*, 2019. 3

[38] Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J. Black. Icon: Implicit clothed humans obtained from normals. *CVPR*, 2022. 6, 7

[39] Weipeng Xu, Avishek Chatterjee, Michael Zollhoefer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM TOG*, 2018. 6

[40] 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. *CVPR*, 2021. 2

[41] 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. *NeurIPS*, 2021. 2

[42] Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. Banmo: Building animatable 3d neural models from many casual videos. *CVPR*, 2022. 1, 2, 3, 6, 7

[43] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. *CVPR*, 2019. 5

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

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