MoDA: Modeling Deformable 3D Objects from Casual Videos

Chaoyue Song[1], Jiacheng Wei[1], Tianyi Chen[1,2], Yiwen Chen[1], Chuan-Sheng Foo[3], Fayao Liu[3], Guosheng Lin[1*]
[1]Nanyang Technological University, Singapore
[2]South China University of Technology, China
[3]Institute for Inforcomm Research, A*STAR, Singapore
{chaoyue002@e., gslin@}ntu.edu.sg

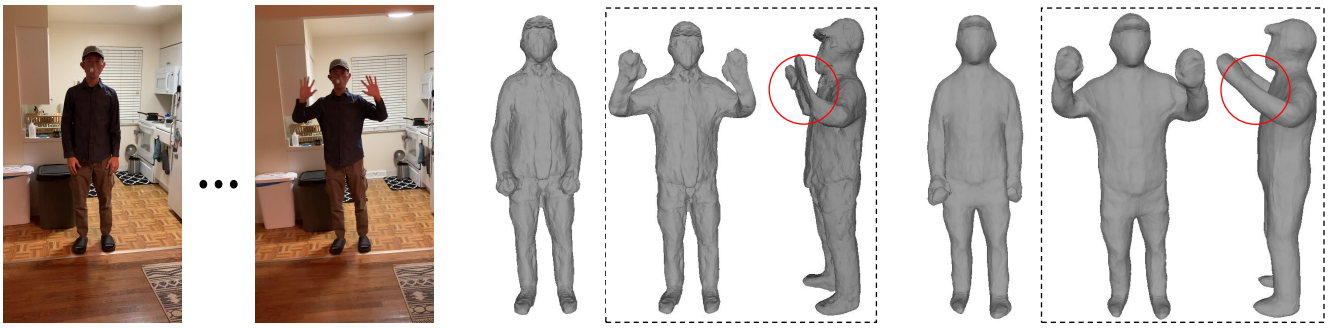

Figure 1: In this work, we introduce MoDA that can reconstruct deformable 3D objects from the input casual videos with neural deformation models. Deformation models are used to transform 3D points between the canonical space (rest pose) and the observation space (deformed pose). Previous work BANMo [66] uses linear blend skinning as their deformation model, resulting in visible skin-collapsing artifacts on the arms. MoDA can solve this problem with the proposed neural dual quaternion blend skinning.

Abstract

In this paper, we focus on the challenges of modeling deformable 3D objects from casual videos. With the popularity of neural radiance fields (NeRF), many works extend it to dynamic scenes with a canonical NeRF and a deformation model that achieves 3D point transformation between the observation space and the canonical space. Recent works rely on linear blend skinning (LBS) to achieve the canonical-observation transformation. However, the linearly weighted combination of rigid transformation matrices is not guaranteed to be rigid. As a matter of fact, unexpected scale and shear factors often appear. In practice, using LBS as the deformation model can always lead to skin-collapsing artifacts for bending or twisting motions. To solve this problem, we propose neural dual quaternion blend skinning (NeuDBS) to achieve 3D point deformation, which can perform rigid transformation without skin-collapsing artifacts. In the endeavor to register 2D pixels across different frames, we establish a correspondence between canonical feature embeddings that encodes 3D points within the canonical space, and 2D image features by solving an optimal transport problem. Besides, we introduce a texture filtering approach for texture rendering that effectively minimizes the impact of noisy colors outside target deformable objects. Extensive experiments on real and synthetic datasets show that our approach can reconstruct 3D models for humans and animals with better qualitative and quantitative performance than state-of-the-art methods. Project page: https://chaoyuesong.github.io/MoDA.

1. Introduction

Modeling deformable 3D objects from casual videos has many potential applications in virtual reality, 3D animated movies, and video games. With the popularity of 2D content creation based on advanced techniques, the demand for 3D content creation [56, 4] is becoming more and more urgent for users. Recent works for rigid objects [33, 9] cannot gen-

eralize to deformable object categories such as humans and animals, which tend to be the focus of content creation today. Others requiring synchronized multi-view video inputs [39, 38] are also not available to general users. Therefore, we focus on the challenges of learning deformable 3D objects from casually collected videos in this work. To achieve this goal, we need to learn how to represent deformable objects and model their articulated motions from videos.

Neural radiance field (NeRF) [29] is proposed as a representation of static 3D scenes into volume density and view-dependent radiance. It has shown impressive performance with volume rendering techniques. To extend NeRF to dynamic scenes, recent methods [40, 35, 36, 66, 38] introduce a canonical neural radiance field that models the shape and appearance, and a deformation model that achieves 3D point transformation between the observation space and the canonical space. NSFF[24] and D-NeRF[40] propose a displacement field to perform the deformation. Nerfies [35] and HyperNeRF [36] represent their deformation model as a dense $\mathrm{SE}(3)$ field. These methods fail when the motion between deformable objects and the background is large. Recently, BANMo [66] achieves their deformation model via linear blend skinning (LBS) to solve this problem. However, the linearly weighted combination of rigid transformation matrices is not necessarily a rigid transformation. As a matter of fact, unexpected scale and shear factors always appear. In practice, using LBS as the deformation model can always lead to skin-collapsing artifacts for bending or twisting motions as shown in Figure 1.

In this work, we present **MoDA** to **Mo**del **D**eform**A**ble 3D objects from multiple casual videos. To handle the large motions between deformable objects and the background without introducing skin-collapsing artifacts, we propose neural dual quaternion blend skinning (NeuDBS) as our deformation model to achieve the observation-to-canonical and canonical-to-observation transformation, which can guarantee the transformations are rigid by blending unit dual quaternions [13]. With a canonical NeRF as our shape and appearance model, we achieve rigid articulated motions with the proposed deformation model. In the endeavor to register 2D pixels across different frames, we establish a correspondence between canonical feature embeddings that encodes 3D points within the canonical space, and 2D image features. To further promote a one-to-one matching process, we have structured the learning of 2D-3D correspondence learning as an optimal transport problem. Besides, we introduce a texture filtering approach for texture rendering that effectively minimizes the impact of noisy colors (e.g., background colors) outside target deformable objects. Extensive experiments on real and synthetic datasets show that MoDA reconstructs 3D deformable objects like humans and animals with better qualitative and quantitative performance than state-of-the-art methods.

We summarize our main contributions as follows:

- We introduce MoDA to model deformable 3D objects from multiple casual videos. Through extensive experiments, we demonstrate that MoDA has a better performance than state-of-the-art methods quantitatively and qualitatively on several different datasets.
- To handle the large motions between deformable objects and the background without introducing skin-collapsing artifacts, we propose neural dual quaternion blend skinning (NeuDBS) as our deformation model to transform the 3D points between observation space and canonical space.
- To register 2D pixels across different frames, we establish the correspondence between canonical feature embeddings of 3D points in the canonical space and 2D image features by solving an optimal transport problem.
- We design a texture filtering approach for texture rendering that effectively minimizes the impact of noisy colors outside target deformable objects.

2. Related work

2.1. 3D human and animal models

Many methods rely on parametric shape models [27, 37, 54, 60, 76, 75, 28] to reconstruct 3D human and animal. These parametric models are constructed from registered 3D scans of humans or animals. They are popular in building 3D shapes from images or videos [1, 2, 16, 74] and 3D human or animal generation tasks [48, 49, 61]. Recently, the human model (SMPL) is also used to learn skinning weights for linear blend skinning in [38]. Nonetheless, constructing parametric models for certain categories, such as various types of animals, proves to be challenging due to the difficulty in obtaining a sufficient amount of data.

2.2. 3D reconstruction from images or videos

There are many prior methods [12, 6, 22, 23, 59, 70] learn 3D reconstruction from images or videos with the supervision of 2D annotations (key points, optical flow, etc). Their performance will usually be limited due to their reliance on rough shape templates. Neural implicit surface representations [34, 42, 43, 45, 46, 55, 69] also have many applications in image or video reconstruction. [33, 9] learn to reconstruct rigid objects from videos. In this work, we focus on the deformable categories, e.g., humans and animals. Recent work, such as LASR [64] and ViSER [65], optimizes a single 3D deformable model on a monocular video using the mask and optical flow supervision. However, they always introduce unrealistic articulated motions. With the popularity of neural radiance fields [29], there are many works [25, 39, 38, 32, 51, 57, 11, 36, 35, 71, 3, 50, 19, 21] learning to reconstruct the shape and appearance from images or videos with a NeRF-based template. Instead of

learning the density in NeRF directly, we use the Signed Distance Function (SDF) that has a well-defined surface at the zero level-set.

2.3. Neural radiance fields for dynamic scenes

Recently, many works represent dynamic scenes by learning a deformation model to map the observed points to a canonical space. NSFF[24] introduces scaled scene flow to displace the 3D points, D-NeRF [40] learns a displacement to transform the given point to the canonical space, NR-NeRF [52] learns a rigidity network to model the deformation of non-rigid objects. Nerfies and HyperNeRF [35, 36] learn a dense $\mathrm{SE}(3)$ field to formulate the deformation. These methods always fail when the motion between deformable objects and the background is large. [25, 39, 38, 32, 51, 57, 11] were proposed to solve this problem. However, they either rely on a 3D human model (e.g., SMPL [27]) or synchronized multi-view videos. BANMo [66] can build 3D shapes from casual videos without human or animal models, it learns the deformation model using linear blend skinning (LBS), which always produces skin-collapsing artifacts. To mitigate this issue, we model the deformation with neural dual quaternion blend skinning (NeuDBS) in this work.

2.4. Correspondence Learning

Several prior works[65, 66] have utilized soft-argmax regression to establish a correlation between a canonical feature embedding, which encodes semantic information of three-dimensional points in the canonical space, and two-dimensional pixel features. However, soft-argmax matching, which computes cosine similarities [62], can result in many-to-one matching problems. Optimal transport has emerged as an influential tool in addressing this issue, particularly due to its propensity to promote one-to-one matching. This approach has been used in scene flow prediction between point clouds [41, 20], 3D semantic segmentation [44] and few-shot segmentation [26]. Other research, such as those conducted by Song et al. [48, 49], has applied optimal transport to learn the correspondence between different meshes for 3D pose transfer. In this work, we extend this line of investigation by solving an optimal transport problem to build the correspondence between canonical feature embeddings and 2D pixel features, enabling the registration of pixel observations across varying frames.

3. Revisit linear blend skinning

The aim of linear blend skinning (LBS) [18, 10] is to blend transformation matrices linearly and then transform vertices in the rest pose to the expected position in the deformed pose. Each vertex in the mesh can be influenced by multiple joints. The influence of joints on each vertex is controlled by skinning weights. We assume that vertex

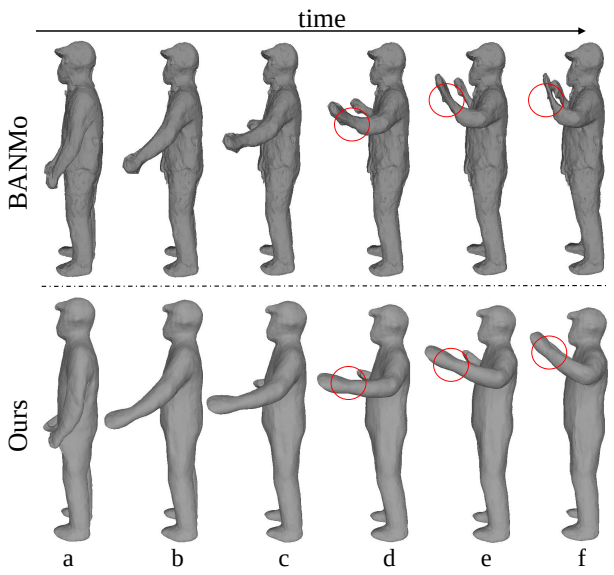

Figure 2: From state *a* to *c*, BANMo and our method can both perform well for motion with small joint rotations. From state *d* to *f*, BANMo has more and more obvious skin-collapsing artifacts for motion with large rotations, our method resolves the artifacts with the proposed NeuDBS.

$\mathbf{v}$ is influenced by joints $\{j_1, ..., j_n\}$ with skinning weights $\{w_1, ..., w_n\}$. Then the transformed vertex position can be formulated as

$$\mathbf{v}' = (\sum_{i=1}^{n} w_i T_i)\mathbf{v}, \tag{1}$$

where $T_i \in \mathrm{SE}(3)$ is the transformation matrix. Although each T_i represents a rigid transformation, the linearly weighted combination of them is not necessarily a rigid transformation since the addition of orthonormal matrices is not closed. Scale and shear factors always appear. Therefore, the blended transformation matrix applied to vertices tends to cause the limb to shrink and lose volume for bending and twisting motions, which is known as skin-collapsing artifacts. We refer readers to [13] for details.

LBS has shown impressive performance as the deformation model to represent dynamic scenes in BANMo [66], but it still has obvious limitations as discussed above. To better understand the performance of LBS as the deformation model, we compare BANMo and our method on a relatively complete human motion sequence from state *a* to *f* in Figure 2. For BANMo (the first line in Figure 2), it performs well and has no obvious skin-collapsing artifacts for motion with small joint rotations (state *a* to *c*), but the artifacts are more and more clear for larger rotations (state *d* to *f*). Our method (the second line in Figure 2) can solve this problem using neural dual quaternion blend skinning (NeuDBS) which will be described in Section 4.2.

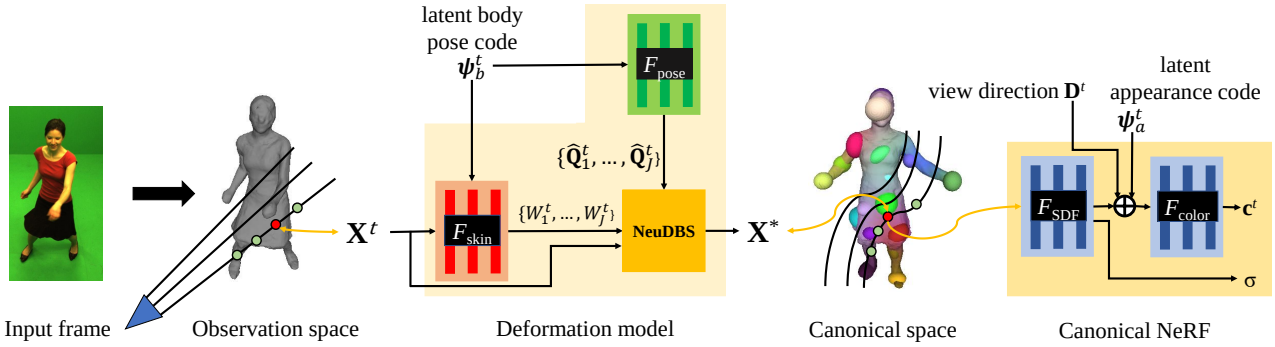

Figure 3: **The overview of MoDA.** We represent the deformable 3D objects from multiple casual videos with a shape and appearance model based on a canonical neural radiance field and a deformation model that achieves 3D point transformation between the observation space and the canonical space. Instead of linear blend skinning used in previous works, we propose NeuDBS as our deformation model. With the learned unit dual quaternions and the skinning weights, we can transform $\mathbf{X}^t$ from the observation space to $\mathbf{X}^*$ in the canonical space. We visualize the joints and the skinning weights (as surface colors) in the canonical space.

4. Method

The overview of our approach is shown in Figure 3. In this work, we represent the deformable 3D objects from multiple casual videos with a shape and appearance model (Section 4.1) based on a canonical neural radiance field and a deformation model (Section 4.2) that transforms 3D points between the observation space and the canonical space. To resolve the skin-collapsing artifacts seen in previous methods, we propose neural dual quaternion blend skinning (NeuDBS) to achieve the expected rigid transformations by blending unit dual quaternions. In order to register 2D pixels across different frames, we formulate the correspondence learning between canonical feature embeddings of 3D points in the canonical space and 2D image features as an optimal transport problem (Section 4.3). Furthermore, we design a texture filtering approach (Section 4.4) for texture rendering that effectively minimizes the impact of noisy colors outside target deformable objects.

4.1. Shape and appearance model

We first introduce how to model the shape and appearance of a deformable object in canonical space. As in Neural Radiance Fields (NeRF) [29], we learn the color and density of a 3D point $\mathbf{X}^* \in \mathbb{R}^3$ in the canonical space,

$$\mathbf{c}^t = F_{\text{color}}(\mathbf{X}^*, \mathbf{D}^t, \boldsymbol{\psi}_a^t), \tag{2}$$

$$\sigma = \Phi_\beta(F_{\text{SDF}}(\mathbf{X}^*)), \tag{3}$$

where F_{color} and F_{SDF} are MLP networks, $\mathbf{D}^t = (\phi^t, \theta^t)$ is the time-varying view direction and $\boldsymbol{\psi}_a^t$ is a 64-dimensional latent appearance code to encode the appearance variations.

Our canonical shape is modeled by F_{SDF}, which predicts signed distances for 3D points in the canonical space. To perform volume rendering as [29], we need to convert signed distances into density. In this work, we use the Cumulative Distribution Function of the Laplace distribution with zero mean and β scale, denoted as $\Phi_\beta(\cdot)$. β is a learnable parameter. As discussed in [55, 68], the Signed Distance Function (SDF) has a well-defined surface at the zero level-set compared with the density function used in NeRF.

4.2. Deformation model

In this section, we introduce how to achieve the 3D point transformation between the observation space and the canonical space via the deformation model. We denote the canonical-to-observation and observation-to-canonical deformation as $\mathcal{D}_{c\to o}^t$ and $\mathcal{D}_{o\to c}^t$ respectively. The body pose in the canonical space is also known as *rest pose*.

Canonical-Observation deformation. For any 3D point $\mathbf{X}^t$ in the observation space, we can map it to the corresponding point $\mathbf{X}^*$ in the canonical space via the deformation model. Here, we denote $\mathbf{C}^t \in \text{SE}(3)$ as the transformation of the camera pose from the canonical space to the observation space at time t, and $\hat{\mathbf{Q}}_j^t \in \mathbb{R}^8$ as the rigid transformation represented by a unit dual quaternion that transforms the j-th joint from the rest pose to the deformed position at time t, then

$$\mathbf{X}^t = \mathcal{D}_{c\to o}^t(\mathbf{X}^*) = \mathbf{C}^t \hat{\mathbf{Q}}_{c\to o}^t \mathbf{X}^*, \tag{4}$$

$$\mathbf{X}^* = \mathcal{D}_{o\to c}^t(\mathbf{X}^t) = \hat{\mathbf{Q}}_{o\to c}^t (\mathbf{C}^t)^{-1} \mathbf{X}^t, \tag{5}$$

where $\hat{\mathbf{Q}}_{o\to c}^t$ and $\hat{\mathbf{Q}}_{c\to o}^t$ are blended by J rigid transformations that transform the joints between the rest pose and the deformed positions at time t. Multiplication between dual quaternions and 3D coordinates can be done by simply converting the 3D coordinates to dual quaternion format.

Transformations for body pose and camera pose are respectively parametrized by MLP networks F_{pose} and F_{cam},

$$\hat{\mathbf{Q}}^t = F_{\text{pose}}(\boldsymbol{\psi}_b^t), \quad \mathbf{C}^t = F_{\text{cam}}(\boldsymbol{\psi}_c^t)\mathbf{C}_0^t, \tag{6}$$

where $\hat{\mathbf{Q}}^t = \{\hat{\mathbf{Q}}_1^t, ..., \hat{\mathbf{Q}}_J^t\}$ are the learned unit dual quaternions for rigid body pose transformations. $\mathbf{C}_0^t$ is the initial camera pose learned from *PoseNet* [66, 72]. $\boldsymbol{\psi}_b^t$ and $\boldsymbol{\psi}_c^t$ are 128-dimensional latent body pose code and camera pose code respectively.

Neural dual quaternion blend skinning. Dual quaternion blend skinning (DBS) was first proposed by Kavan *et al.* [13] and can effectively resolve the skin-collapsing artifacts. All DBS-related parameters (body pose transformation, skinning weights, and joints) are predefined in [13], but are unknown and difficult to obtain in our task. Therefore, our main challenge is to define and predict these parameters, the basic idea is that we learn them with MLP networks.

To prevent the skin-collapsing artifacts as discussed in Section 3 and better model the deformation of 3D objects, we propose neural dual quaternion blend skinning (NeuDBS) as our deformation model. Instead of blending the transformation matrices, our method blends the unit dual quaternions linearly and then normalizes the results to get the final dual quaternion. Given learned unit dual quaternions $\{\hat{\mathbf{Q}}_1^t, ..., \hat{\mathbf{Q}}_J^t\}$ that represent the rigid transformations, the blended unit dual quaternion can be computed as follows,

$$\hat{\mathbf{Q}}_{c\to o}^t = \frac{\sum_{j=1}^J W_{j,c\to o}^t \hat{\mathbf{Q}}_j^t}{\left\|\sum_{j=1}^J W_{j,c\to o}^t \hat{\mathbf{Q}}_j^t\right\|}, \tag{7}$$

$$\hat{\mathbf{Q}}_{o\to c}^t = \frac{\sum_{j=1}^J W_{j,o\to c}^t (\hat{\mathbf{Q}}_j^t)^{-1}}{\left\|\sum_{j=1}^J W_{j,o\to c}^t (\hat{\mathbf{Q}}_j^t)^{-1}\right\|}, \tag{8}$$

where $W_{j,c\to o}^t$ and $W_{j,o\to c}^t$ are skinning weights that control the influence of j-th joint on $\mathbf{X}^*$ and $\mathbf{X}^t$ respectively. By computing a unit dual quaternion, NeuDBS always returns a valid rigid transformation which can prevent the skin-collapsing artifacts.

In addition to the difference in parameter definition, Kavan *et al.* must convert predefined skinning matrices (3×3 rotation and 1×3 translation) to dual quaternions first to apply DBS. MoDA learns 7 scalars per joint via MLP and then directly derives dual quaternions (see next paragraph for details), which does not require the quaternion-matrix conversion and is more efficient. We will also introduce how to define and optimize the skinning weights in the supplement.

Learning of body pose transformation. In BANMo [66], they learn 6 scalars per joint for body pose transformation. 3 of them are for translation, the other 3 scalars are the logarithmic representation of the rotation matrix. They convert logarithmic representations of rotation matrices into 3×3 rotation matrices. Therefore, BANMo ends up requiring a 3×4 matrix per joint for body pose transformation.

In this work, we learn 7 scalars per joint via F_{pose} for the rigid transformation. 3 of them (t_1, t_2, t_3) are for translation, then we can obtain the quaternion $\mathbf{q}_t = [0, t_1, t_2, t_3]$ representing the translation, where 0 is the scalar part, $t_1\mathbf{i} + t_2\mathbf{j} + t_3\mathbf{k}$ is the vector part of a quaternion. The other 4 scalars are the quaternion $\mathbf{q}_r$ that represents the rotation. Then we convert them to the dual quaternion,

$$\mathbf{Q}_r = \hat{\mathbf{q}}_r, \quad \mathbf{Q}_d = \frac{1}{2}\mathbf{q}_t \bigotimes \hat{\mathbf{q}}_r, \tag{9}$$

where $\mathbf{Q}_r$ is the real part of the dual quaternion, $\mathbf{Q}_d$ is the dual part of the dual quaternion. $\hat{\mathbf{q}}_r$ is the normalized quaternion to make sure the calculated dual quaternion is the unit dual quaternion. $\bigotimes$ means the multiplication between quaternions.

In sum, BANMo learns 6 scalars and requires a 3×4 matrix per joint to represent the body pose transformation. Our MoDA learns 7 scalars but only requires 8 scalars per joint for the body pose transformation, which introduces a more efficient representation. Linear Blend Skinning (LBS) [10, 18] is widely recognized for its efficiency, and the training time required for MoDA with NeuDBS is comparable to that of BANMo, please refer to the appendix for details.

4.3. 2D-3D matching via optimal transport

To match pixels at different frames, we establish the correspondence between canonical feature embeddings of 3D points in the canonical space and 2D image features. In [65, 66], they employ soft-argmax regression to learn the 2D-3D correspondence by calculating the cosine similarity. To perform better matching, we formulate the 2D-3D matching as an optimal transport problem that encourages one-to-one matching. Given pixels $\{\mathbf{x}^t\}_{N_{pixel}}$ and 3D points $\{\mathbf{X}^*\}_{N_{point}}$ in the canonical space, we learn the pixel features $f_{pixel}(\mathbf{x}^t) \in \mathbb{R}^{16\times N_{pixel}}$ with CSE [30, 31] and the canonical feature embeddings $f_{point}(\mathbf{X}^*) = F_{emb}(\mathbf{X}^*) \in \mathbb{R}^{16\times N_{point}}$. We obtain the correlation matrix $\mathbf{M} \in \mathbb{R}^{N_{pixel}\times N_{point}}$ by calculating their cosine similarity,

$$\mathbf{M}(j,k) = \frac{f_{pixel}(j)^\top f_{point}(k)}{\|f_{pixel}(j)\| \, \|f_{point}(k)\|} \tag{10}$$

where $\mathbf{M}(j,k)$ is the matching score between j-th pixel and k-th point. To formulate the optimal transport problem, we define a matching matrix $\mathbf{T} \in \mathbb{R}^{N_{pixel}\times N_{point}}$ and the cost matrix $\mathbf{Z} = 1 - \mathbf{M}$. Our goal is to minimize the total cost to get the optimal matching matrix:

$$\begin{aligned} \mathbf{T}^* &= \arg\min_{\mathbf{T}} \sum_{jk} \mathbf{Z}(j,k)\mathbf{T}(j,k) \\ s.t. \quad \mathbf{T}\mathbf{1}_{N_{point}} &= \mathbf{1}_{N_{pixel}} N_{pixel}^{-1}, \\ \mathbf{T}^\top \mathbf{1}_{N_{pixel}} &= \mathbf{1}_{N_{point}} N_{point}^{-1}. \end{aligned} \tag{11}$$

This optimal transport problem can be solved by the Sinkhorn-Knopp algorithm [47]. Then we can find the 3D surface point in the canonical space matching to $\mathbf{x}^t$ by warping the sampled points $\mathbf{V}^*$ in a canonical 3D grid,

$$\widetilde{\mathbf{X}}^*(\mathbf{x}^t) = \sum_{\mathbf{X} \in \mathbf{V}^*} \mathbf{T}^* \mathbf{X}, \tag{12}$$

Based on the 3D surface points, we have the 2D-3D matching losses. We first define a point matching loss [66] as

$$\mathcal{L}_{match} = \sum_{\mathbf{x}^t} \left\| \widetilde{\mathbf{X}}^*(\mathbf{x}^t) - \mathbf{X}^*(\mathbf{x}^t) \right\|_2^2, \tag{13}$$

where $\mathbf{X}^*(\mathbf{x}^t)$ is calculated from Eq. 17. We also define a projection loss [17, 65, 66] that encourages the image projection after canonical-to-observation deformation of $\widetilde{\mathbf{X}}^*(\mathbf{x}^t)$ to be close to its original 2D coordinates.

$$\mathcal{L}_{proj} = \sum_{\mathbf{x}^t} \left\| \mathbf{P}^t(\mathcal{D}^t_{c \to o}(\widetilde{\mathbf{X}}^*(\mathbf{x}^t)) - \mathbf{x}^t \right\|_2^2, \tag{14}$$

where $\mathbf{P}^t$ is the projection matrix of a pinhole camera.

4.4. Volume rendering and optimization

Texture filtering for volume rendering. When predicting the object color as Eq. 2, the training process will inevitably introduce some noisy textures (e.g., background texture in the first row of Figure 8) that do not belong to the target deformable objects.

Inspired by [58, 73], we develop a texture filtering approach to mitigate this issue. Rather than relying on semantic fields to calculate opacity in [58, 73], we utilize a texture filtering function s for texture rendering, which can exclude the noisy colors outside objects (i.e., remove the estimated color $\mathbf{c}^t$ when SDF $d > 0$).

Here, we define $\mathbf{x}^t \in \mathbb{R}^2$ as the pixel location at time t, and $\mathbf{X}_k^t$ as the k-th sampled point along the ray that originates from $\mathbf{x}_t$. Then color $\mathbf{c}$ and opacity $\mathbf{o} \in [0, 1]$ are given by:

$$\mathbf{c}(\mathbf{x}^t) = \sum_{k=1}^{N} \tau_k (s_k \mathbf{c}_k^t), \quad \mathbf{o}(\mathbf{x}^t) = \sum_{k=1}^{N} \tau_k, \tag{15}$$

where N is the number of sampled points, $\tau_k = \alpha_k \prod_{i=1}^{k-1}(1-\alpha_i)$, $\alpha_k = 1 - \exp(-\sigma_k \delta_k)$, δ_k is the distance between the k-th sample and the next, and σ_k is the density in Eq. 3. Texture filtering function s is defined as

$$s = \frac{\gamma}{1 + e^{\lambda d}}, \tag{16}$$

which is a scaled sigmoid function based on SDF $d = F_{\text{SDF}}(\mathbf{X}^*)$. s gives 0 weights to $\mathbf{c}^t$ of the sampled points that are far away from the object (have large positive SDF values) to exclude them in the rendering process. γ and λ are scale and temperature parameters.

We can also calculate the surface point,

$$\mathbf{X}^*(\mathbf{x}^t) = \sum_{k=1}^{N} \tau_k \mathbf{X}_k^*, \tag{17}$$

where $\mathbf{X}_k^*$ is obtained by applying the deformation $\mathcal{D}^t_{o \to c}$ to the k-th 3D point $\mathbf{X}_k^t$.

Optimization. In addition to the 2D-3D matching losses, we incorporate the following reconstruction losses into our model optimization, which are commonly employed in existing methods such as [29, 69, 66]:

$$\mathcal{L}_{rgb} = \sum_{\mathbf{x}^t} \left\| \mathbf{c}(\mathbf{x}^t) - \widetilde{\mathbf{c}}(\mathbf{x}^t) \right\|^2, \tag{18}$$

$$\mathcal{L}_{sil} = \sum_{\mathbf{x}^t} \left\| \mathbf{o}(\mathbf{x}^t) - \widetilde{\mathbf{s}}(\mathbf{x}^t) \right\|^2, \tag{19}$$

where $\mathcal{L}_{rgb}$ and $\mathcal{L}_{sil}$ are the pixel color loss and silhouette loss respectively. $\widetilde{\mathbf{c}}$ and $\widetilde{\mathbf{s}}$ are observed pixel color and silhouette. Here, $\widetilde{\mathbf{s}}$ is extracted from off-the-shelf method [15]. Additional losses will be introduced in the supplement, such as flow loss (the estimated flow is obtained from [63]).

5. Experiments

5.1. Dataset, metrics, and implementation details

Casual videos. To demonstrate the effectiveness of MoDA, we test it on casual videos of humans and animals. The *casual-cat* dataset includes 11 videos (900 frames in total) of a British shorthair cat, which are collected by [66]. The *casual-human* dataset [66] includes 10 videos (584 frames in total). The *casual-adult* dataset includes 10 videos (1000 frames in total). The capture of the videos has no control for camera and object movements. We use the object silhouette and optical flow predicted by [15, 63] respectively.

AMA dataset. To evaluate our method quantitatively, we use the Articulated Mesh Animation (AMA) dataset [53] that provides ground truth meshes. AMA is collected with the setup consisting of a ring of 8 cameras. We train our models on 2 sets of videos of the same person (*swing* and *samba*, including 2600 frames in total) with ground truth object silhouettes and the optical flow predicted by [63].

Animated objects dataset. Besides cats and humans, we quantitatively evaluate our method on other deformable categories. We use the animated objects dataset from TurboSquid (known as *eagle* and *hands*). They both include 5 videos with 150 frames per video. We train these two

datasets with the ground truth camera poses, ground truth silhouettes and the optical flow predicted by [63].

Metrics. To compare different methods quantitatively, we use Chamfer distance (CD) [5] and F-scores as our evaluation metrics. CD is calculated between the point sets of the reconstructed mesh and the ground truth mesh. $(\mathbf{p}, \widetilde{\mathbf{p}})$

$$CD(\mathbf{p}, \widetilde{\mathbf{p}}) = \sum_{x \in \mathbf{p}} \min_{y \in \widetilde{\mathbf{p}}} \|x - y\|_2^2 + \sum_{y \in \widetilde{\mathbf{p}}} \min_{x \in \mathbf{p}} \|x - y\|_2^2. \quad (20)$$

For CD, the lower is better. For F-scores, we compare different methods at distance thresholds $d = 2\%$, and the bigger the better when using F-scores.

Implementation details. In this work, we set the number of joints to 25. The initialization of them is similar to BANMo [66], with unit scale, identity orientation, and uniformly distributed centers. The meshes are extracted by running marching cubes on a 256^3 grid. For more implementation details, please refer to our appendix.

5.2. Comparison results on multiple videos

In this section, we compare MoDA with ViSER [65] and BANMo [66] over multiple videos. For a fair comparison, we provide them with the same initial camera poses. And we train BANMo and ViSER using the implementations provided by the authors.

As shown in Figure 4, ViSER cannot learn detailed shapes and accurate poses from the given videos. To better demonstrate the influence of the deformation model, we also show the corresponding rest pose for each reference image. For motion with large joint rotations, the results of BANMo have obvious skin-collapsing artifacts (as shown in the red circles). The reconstructed shapes tend to shrink and lose volume, e.g., the arms of humans in *casual-adult*, *casual-human* and AMA-*samba*, and the body of the cat in *casual-cat*. Our method can solve these problems and achieve rigid articulated motions. For *eagle*, BANMo and our method have close performance since the motion of the eagle from the rest pose to the deformed pose is relatively slight. We will show more results on multiple-video setups (including the video demonstrations) in the supplement.

For the quantitative results as shown in Table 1, our method also has a better performance than BANMo and ViSER on multiple-video setups.

5.3. Comparison results on single-video setups

In this section, we compare MoDA with Nerfies [35], HyperNeRF [36] and BANMo [66] over single-video setups. To make a fair comparison, we also provide them with the same initial camera poses. We reproduce Nerfies, HyperNeRF, and BANMo using the implementations provided by the authors. Besides, we provide Nerfies and HyperNeRF with the ground truth object silhouettes of AMA and Animated object datasets to calculate the silhouette losses which can help to improve the performance.

Nerfies and HyperNerf have very close performance so we only show the results of HyperNeRF in Figure 5. HyperNeRF fails to learn reasonable shapes and deformations for deformable 3D objects when the motion between the object and the background is large. The reconstructed results of BANMo still have clear skin-collapsing artifacts as shown in the red circles while MoDA has a better performance and resolves the skin-collapsing artifacts. Obviously, the performance of BANMo and MoDA both degrade with some unexpected artifacts compared to the multiple-video setups.

For the quantitative results as shown in Table 1, our method has a better performance than Nerfies, HyperNeRF, and BANMo on single-video setups. The quantitative performance of BANMo and our method on single-video setups also degrades compared to the multiple-video setups.

5.4. Ablation study

Optimal transport. We also evaluate the importance of optimal transport for 2D-3D matching. To disable optimal transport, we use the soft-argmax regression by calculating the cosine similarity that is similar to [65, 66]. According to the results presented in Table 2, our method with optimal transport achieves better performance than using soft-argmax regression when testing on AMA dataset [53]. In the case of the AMA dataset, which has a clean background and accurate ground truth masks, the addition of optimal transport does not significantly improve performance.

When evaluating our method on the *casual-cat* dataset, the inclusion of optimal transport provides more obvious advantages. It facilitates improved 2D-3D matching, which, in turn, aids in refining the object silhouette. Figure 6 illustrates this improvement. The predicted mask obtained from [15] is inaccurate due to the close pixel colors between the cat and the snack. In such a scenario, both BANMo [66] and our method without optimal transport struggle to refine the mask and accurately capture the 3D geometry. In most frames, the two objects (cat and snack) are not closely located to the extent of being mistaken for a single object. As a result, by utilizing optimal transport to achieve better matching, we can refine the initial segmentation and predict the consistent 3D shape of the cat.

Deformation model. To further validate the effectiveness of our proposed NeuDBS, we compare it with other deformation models. Specifically, we replace NeuDBS with alternative deformation models, including the displacement field from [24, 40], SE(3) field from [35, 36] and neural linear blend skinning (LBS) from [66]. We also implement direct quaternion blending (DQB) [8] which decomposes transformation matrices into quaternions and translations and blends them linearly. To compare with DQB, we design two approaches for learning (translation, quaternion) pairs.

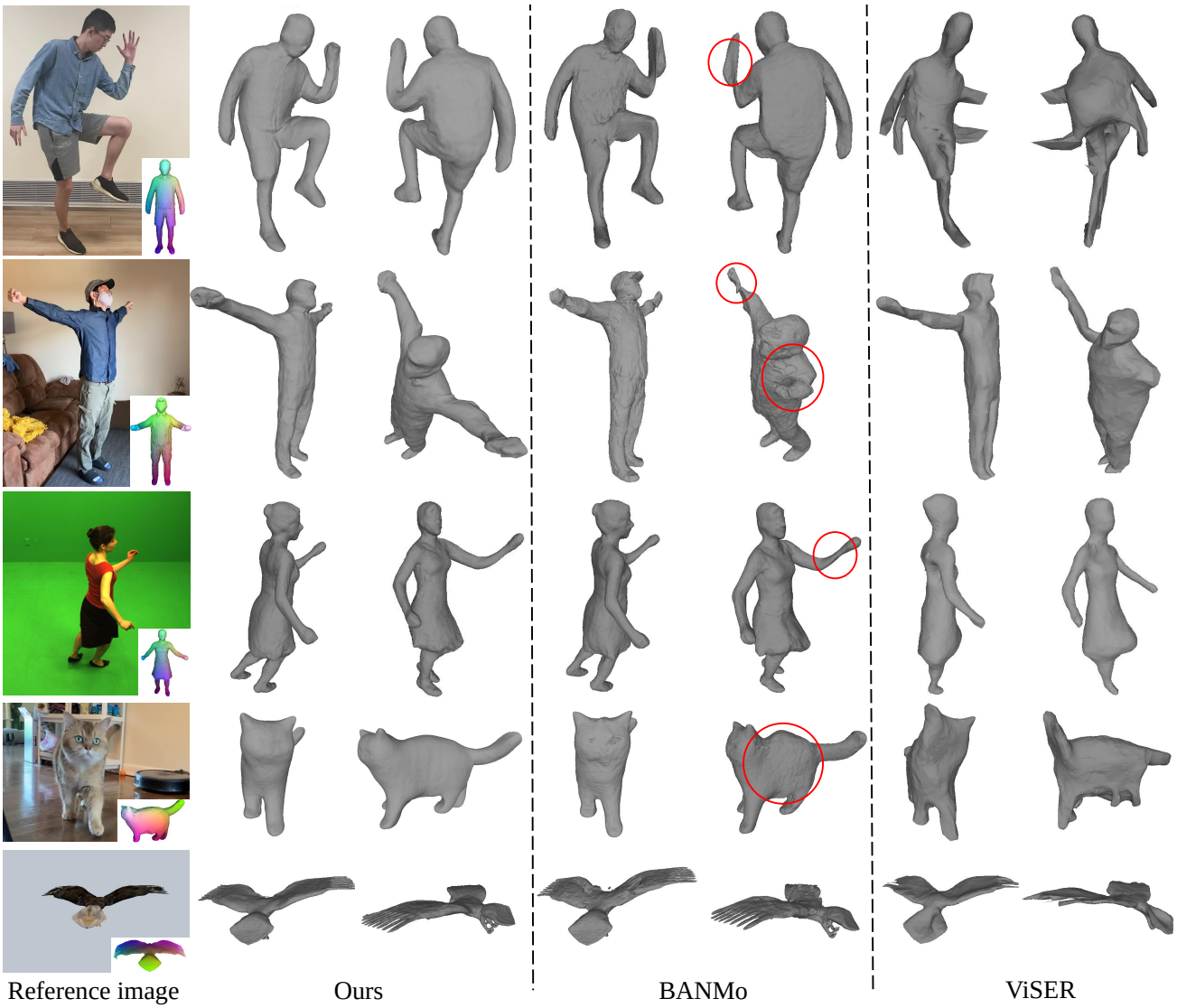

Figure 4: **Qualitative comparison on multiple videos.** The data is from *casual-adult*, *casual-human*, AMA-*samba*, *casual-cat*, *eagle* from top to bottom. The lower right corner of each reference image is the corresponding rest pose. We show 2 views of the reconstructed results based on the reference images. ViSER [65] fails to learn detailed 3D shapes and accurate poses from the videos. BANMo [66] has obvious skin-collapsing artifacts (in the red circles) for motions with large joint rotations while our method performs well. For *eagle* with slight motion, BANMo and our method have close performance.

DQB-BANMo: This method follows BANMo's approach, where we initially learn (translation, logarithmic representation of rotation matrix) pairs and then convert them to (translation, quaternion) pairs. *DQB-Ours*: In alignment with our NeuDBS method, we directly learn (translation, quaternion) pairs. Additionally, we also conduct an ablation study on DBS in RAC [67]. They first learn (translation, logarithmic representation of rotation matrix) pairs, following BANMo's approach, and subsequently converting them to dual quaternions.

The qualitative results on *casual-adult* and *casual-human* are shown in Figure 7, the displacement field and $\mathrm{SE}(3)$ field cannot accurately preserve shapes and learn the pose of humans, particularly in the case of large motions. Furthermore, LBS exhibits noticeable skin-collapsing artifacts on the arm during bending motions. Neither DQB-BANMo nor DQB-Ours alleviates these artifacts. Although RAC-DBS demonstrates some improvements over LBS and DQB, artifacts on the arms are still present. In contrast, our proposed NeuDBS demonstrates superior performance

Table 1: **Quantitative comparison between different methods.** We compare our method with state-of-the-art methods on multiple-video and single-video setups. To quantitatively evaluate different methods, we use Chamfer distance (cm, ↓) and F-score(%, ↑) as the metrics. Our method has the best performance for both multiple-video and single-video setups.

Type	Method	AMA-Swing		AMA-Samba		Eagle		Hands	
		CD(↓)	F(2%, ↑)	CD(↓)	F(2%, ↑)	CD(↓)	F(2%, ↑)	CD(↓)	F(2%, ↑)
Multiple	ViSER	35.8	9.9	33.8	10.0	36.9	2.5	13.5	32.6
	BANMo	7.9	60.8	7.9	60.4	5.2	68.7	5.1	69.5
	Ours	**7.0**	**66.3**	**5.6**	**75.5**	**4.8**	**75.0**	**4.4**	**73.9**
Single	Nerfies	39.1	5.5	42.7	4.8	27.9	10.4	33.8	6.8
	HyperNeRF	42.9	4.9	41.7	5.2	28.6	10.2	30.9	8.3
	BANMo	9.0	55.8	9.6	51.6	14.4	45.1	12.7	27.2
	Ours	**8.5**	**60.5**	**8.2**	**59.7**	**13.7**	**45.7**	**11.3**	**32.8**

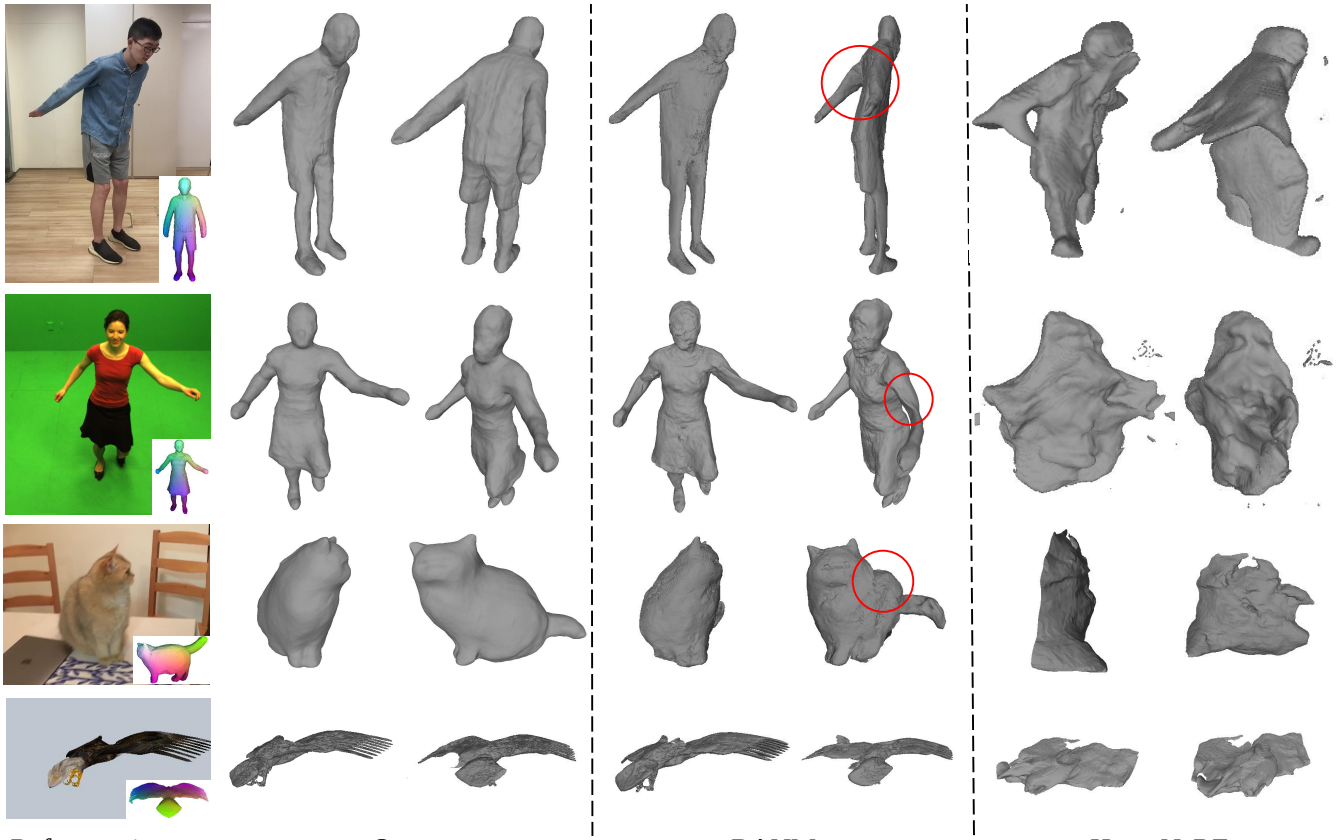

Figure 5: **Qualitative comparison on a single video.** The data is *casual-adult*, AMA-*swing*, *casual-cat*, *eagle* from top to bottom. The lower right corner of each reference image is the corresponding rest pose. We show 2 views of the reconstructed results based on the reference images. HyperNeRF [36] fails to learn reasonable shapes and deformations. For single-video setups, BANMo [66] still has obvious skin-collapsing artifacts (in the red circles) for motions with large joint rotations while our method performs better.

compared to these six deformation models. It successfully preserves shapes and avoids skin-collapsing artifacts, resulting in visually improved outputs. To provide quantitative evidence, we present the results on the AMA dataset [53] in Table 2. Our NeuDBS consistently outperforms the alternative deformation models. Note that while the AMA dataset contains multi-view data, offering richer information, the quantitative distinctions between various deforma-

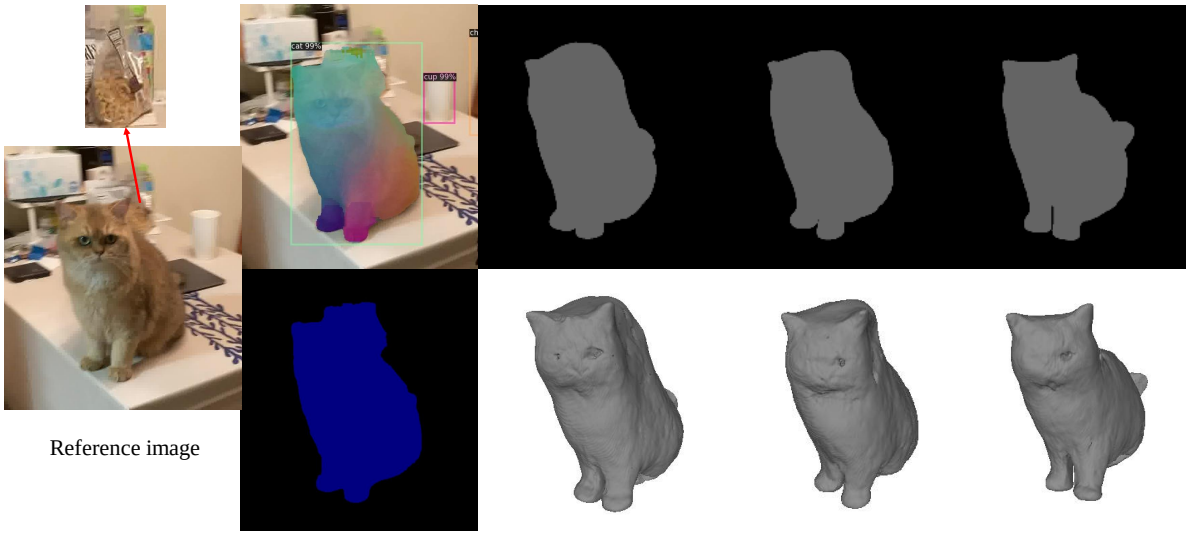

Figure 6: **Ablation study of optimal transport.** By registering 2D pixels across different frames with optimal transport, we can refine the bad segmentation and predict the consistent 3D shape of the cat.

tion models may not be readily apparent. The reconstruction results from casual videos can offer a more straightforward validation of the benefits provided by our NeuDBS.

Texture filtering. Here we also test the importance of the proposed texture filtering on AMA [53], *casual-adult*, and *casual-human*. The texture rendering results of BANMo [66] and MoDA without texture filtering exhibit noticeable noisy textures on arms (the first row in Figure 8) and human bodies (the second and the third rows in Figure 8). Adding texture filtering to both BANMo and MoDA can effectively alleviate this problem.

5.5. Motion re-targeting

We compare BANMo [66] and MoDA's ability of motion re-targeting. Given the pre-trained model on AMA [53] and driving videos of *casual-adult* and *casual-human*, we only optimize the frame-specific camera and body pose codes ψ_c^t and ψ_b^t while keeping other model parameters unchanged. As shown in Figure 9, BANMo consistently struggles to accurately learn human poses from the driving videos. For instance, BANMo fails to recover the standing pose in the bottom-right of Figure 9, which may stem from its failure to adequately disentangle the shape and pose of the human during optimization. In contrast, MoDA exhibits better performance compared to BANMo.

6. Conclusion and limitations

In this paper, we present MoDA, an effective approach for modeling deformable 3D objects from casual videos. We represent 3D objects with a canonical neural radiance field (NeRF) and a deformation model that achieves the 3D point transformation between the observation space and the canonical space. To handle large motions between deformable objects and the background without introducing skin-collapsing artifacts, we propose neural dual quaternion blend skinning (NeuDBS) as our deformation model that can return valid rigid transformations by blending unit dual quaternions. To register 2D pixels across different frames, we model the correspondence learning between canonical feature embeddings of 3D points in the canonical space and 2D image features as an optimal transport problem. Besides, we develop a texture filtering technology for texture rendering that effectively minimizes the impact of noisy colors outside target deformable objects. Extensive experiments on real and synthetic datasets show that the proposed approach can reconstruct 3D shapes for humans and animals with better qualitative and quantitative performance than state-of-the-art methods.

Although MoDA achieves impressive performance in most cases, there are still some limitations that need to be solved in the future. For example, MoDA does not reconstruct the detailed shape of the human hand and the performance on a single video can be further improved.

Acknowledgments

This research is supported by the MoE AcRF Tier 2 grant (MOE-T2EP20220-0007) and the MoE AcRF Tier 1 grant (RG14/22).

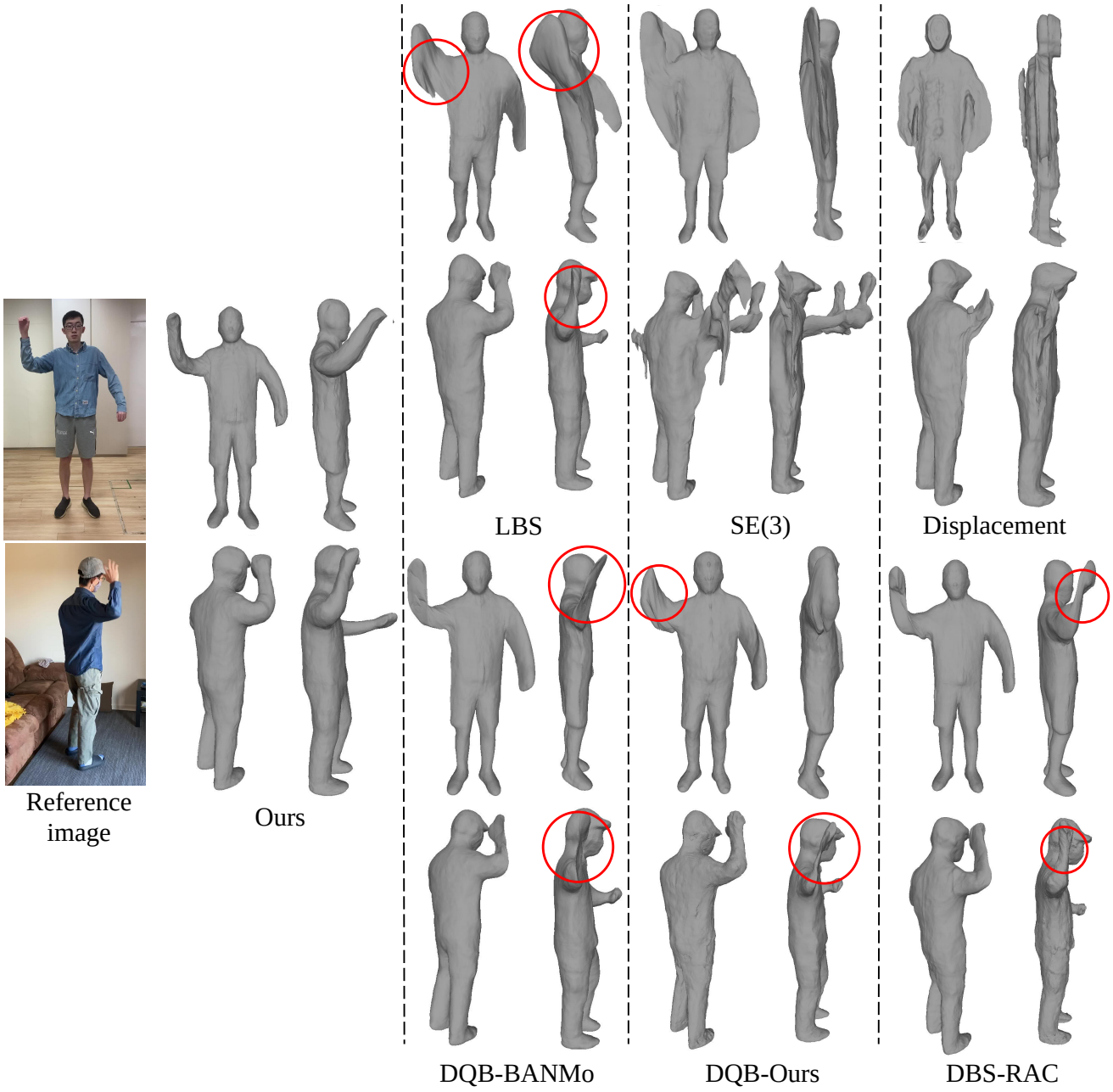

Figure 7: **Ablation study of deformation models.** The displacement field and $\mathrm{SE}(3)$ field cannot accurately preserve shapes and learn the pose of humans when dealing with large motions between humans and the background. LBS-based has obvious skin-collapsing artifacts (in the red circles) on the arm during bending motions. Neither DQB-BANMo nor DQB-Ours alleviates these artifacts. Although RAC-DBS demonstrates some improvements over LBS and DQB, artifacts on the arms are still present. Our method achieves the best performance.

Appendix A. More details of MoDA

A.1. Skinning weights for NeuDBS

We define the skinning weights for NeuDBS as $\mathbf{W} = \{W_1, ..., W_J\} \in \mathbb{R}^J$, where J is the number of joint. Learning the skinning weights only from neural networks is difficult to optimize. To obtain the skinning weights for the proposed NeuDBS, we first calculate the Gaussian skinning weights and then learn the residual skinning weights with an MLP network following [66].

Table 2: **Quantitative ablation studies.** We evaluate different deformation models, the optimal transport module, and texture filtering on AMA dataset. We use Chamfer distance (cm, ↓) and F-score(%, ↑) as the metrics.

Type	Method	AMA-Swing		AMA-Samba	
		CD(↓)	F(2%, ↑)	CD(↓)	F(2%, ↑)
Multiple	Displacement	11.3	44.5	10.7	54.2
	SE(3)	10.4	48.2	9.4	56.4
	LBS	7.6	63.7	6.7	66.9
	DQB-BANMo	7.8	62.6	6.1	69.7
	DQB-Ours	7.5	63.9	6.3	68.4
	DBS-RAC	7.1	65.8	6.0	73.3
	w/o OT	7.2	64.5	5.9	73.1
	Ours	**7.0**	**66.3**	**5.6**	**75.5**

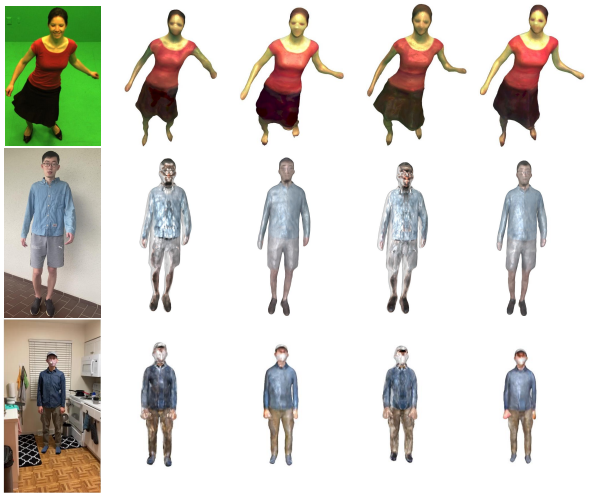

Figure 8: **Ablation study of texture filtering.** Texture rendering results of BANMo [66] and MoDA without texture filtering have obvious noisy textures. Adding texture filtering to them can effectively alleviate this issue.

Firstly, we compute the Gaussian skinning weights based on the Mahalanobis distance between 3D points and the Gaussian ellipsoids,

$$\mathbf{W}_G = (\mathbf{X} - \mathbf{O})^T \mathbf{V}^T \mathbf{\Lambda}^0 \mathbf{V} (\mathbf{X} - \mathbf{O}), \tag{21}$$

where $\mathbf{O} \in \mathbb{R}^{J\times 3}$ are the joint center locations, $\mathbf{V} \in \mathbb{R}^{J\times 3\times 3}$ are joint orientations and $\mathbf{\Lambda}^0 \in \mathbb{R}^{J\times 3\times 3}$ are diagonal scale matrices. The joints represented by explicit 3D Gaussian ellipsoids are composed of these 3 elements: center, orientation, and scale. To learn better skinning weights for 3D deformation, we predict the residual skinning weights from an MLP network,

$$\mathbf{W}_r = F_{skin}(\mathbf{X}, \boldsymbol{\psi}_b), \tag{22}$$

then we have the final skinning weights,

$$\mathbf{W} = \sigma_{softmax}(\mathbf{W}_G + \mathbf{W}_r). \tag{23}$$

To be specific, the skinning weights $\mathbf{W}^t_{o\to c}$ are learned from 3D points in the observation space and the body pose code $\boldsymbol{\psi}^t_b$ at time t, and $\mathbf{W}^t_{c\to o}$ are learned from 3D points in the canonical space and the rest pose code $\boldsymbol{\psi}^*_b$.

A.2. Loss functions

Optical flow loss. We render 2D flow to compute the optical flow loss. Specifically, we deform the canonical points to another time t' and get its 2D re-projection,

$$\mathbf{x}^{t'} = \sum_{k=1}^{N} \tau_k \mathbf{P}^{t'} (\mathcal{D}^{t'}_{c\to o}(\mathbf{X}^*_k)), \tag{24}$$

where $\mathbf{P}^{t'}$ is the projection matrix of a pinhole camera. Then we can compute the 2D flow,

$$\mathbf{f}(\mathbf{x}^t, t \to t') = \mathbf{x}^{t'} - \mathbf{x}^t, \tag{25}$$

and the optical flow loss $\mathcal{L}_{of}$ is defined as

$$\mathcal{L}_{of} = \sum_{\mathbf{x}^t,(t,t')} \left\| \mathbf{f}(\mathbf{x}^t, t \to t') - \widetilde{\mathbf{f}}(\mathbf{x}^t, t \to t') \right\|^2, \tag{26}$$

where $\widetilde{\mathbf{f}}$ is the observed optical flow that are extracted from off-the-shelf method [63].

3D cycle consistency loss. Similar to [24, 66], we introduce a 3D cycle consistency loss to learn better deformations. We deform the sampled points in the observation space to the canonical space and then deform them back to their original coordinates,

$$\mathcal{L}_{cyc} = \sum_k \tau_k \left\| \mathcal{D}^t_{c\to o}(\mathcal{D}^t_{o\to c}(\mathbf{X}^t_k)) - \mathbf{X}^t_k \right\|^2_2, \tag{27}$$

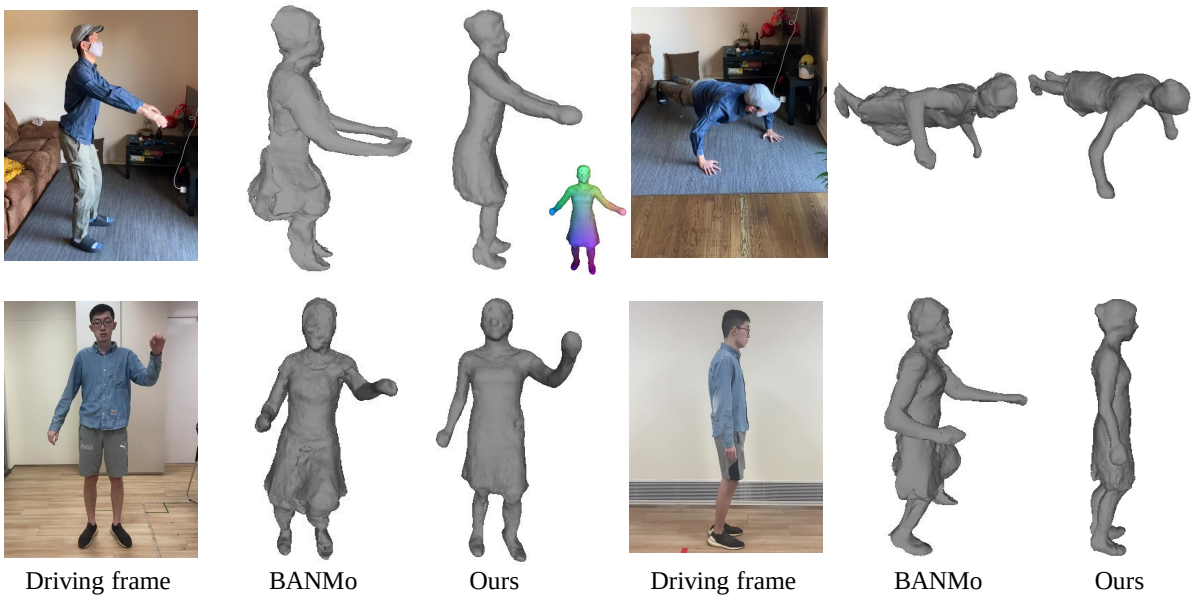

Figure 9: **Motion re-targeting.** We compare the motion re-targeting results from the pre-trained AMA model to *casual-adult* and *casual-human* videos, our method performs better.

where τ_k weighs the sampled points to guarantee the points closer to the surface have stronger regularization.

Eikonal loss. Following [66, 67], we also adopt the implicit geometric regularization term [7] as :

$$\mathcal{L}_{eikonal} = \sum_{\mathbf{X}\in\mathbf{V}^*} (\|\nabla F_{\text{SDF}}(\mathbf{X})\|_2 - 1)^2. \quad (28)$$

A.3. Implementation details

Training strategy. The optimization strategies of MoDA include three stages. Firstly, we optimize all losses and parameters. In this stage, MoDA already reconstructs good shape and deformation. Then we improve the articulated motions, where we only update the parameters related to the deformation model while keeping the shape parameters fixed. Finally, we improve the details of the reconstructions through importance sampling while freezing the camera poses. The design of MLP networks in MoDA is similar to BANMo [66]. The hyperparameters γ and λ in the texture filtering function are set to 1.5 and 10 respectively (See Figure 10 for the function). Our code will be available on GitHub once the paper is accepted.

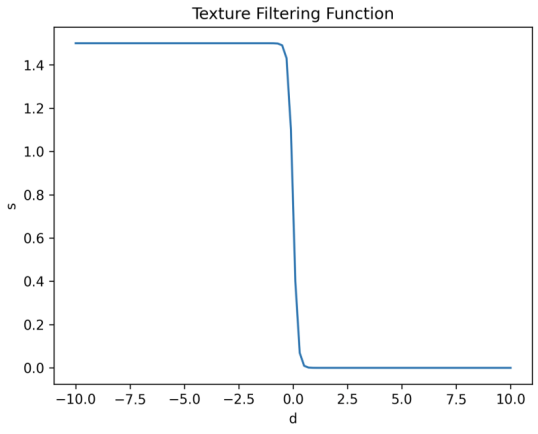

Figure 10: **The texture filtering function.**

Sampling details for 2D-3D matching. In optimal transport, the sampling of 3D points is similar to BANMo [66]. We establish a canonical 3D grid $\mathbf{V}^* \in \mathbb{R}^{20\times20\times20}$ ($N_{point} = 8000$) to build correspondence between pixels and canonical points. This grid is centered at the origin and axis-aligned with bounds $[x_{min}, x_{max}]$, $[y_{min}, y_{max}]$, and $[z_{min}, z_{max}]$, undergoes iterative refinement during optimization. Every 200 iterations, we update the bounds of the grid by approximating the object's bounds. This approximation is obtained by applying marching cubes on a 64^3 grid to extract a surface mesh.

A.4. Dataset

We use 6 datasets in this work, where *casual-cat*, *casual-human*, *eagle* and *hands* are collected by BANMo [66]. We have obtained permission to use these datasets. AMA dataset is the published dataset collected by [53]. The usage of *casual-adult* has also obtained consent.

Figure 11: **Correspondence between different videos on *casual-human* and *casual-cat*.** Distinct colors represent the correspondence.

Figure 12: **Texture comparison of BANMo and our method on casual-cat.** The fourth column is a screenshot of the example provided on BANMo's website.

Appendix B. More experimental results

In this section, we show more experimental results.

Correspondence. In Figure 11, we illustrate the correspondence between different videos in both the *casual-human* and *casual-cat* datasets. Distinct colors represent the correspondence.

Texture rendering results. Here, we compare texture rendering results of BANMo [66] and our method on *casual-cat*. As shown in 12, we also provide a screenshot of the example on BANMo's website[1]. The result of BANMo is aligned with the example on their website, appears unrealistic and potentially influenced by noise. In contrast, our method produces results more closely resembling the reference image.

More results on multiple-video setups. We also show

[1]https://banmo-www.github.io

Table 3: **Training time comparison.** We compare the training times of our method and BANMo [66] on different datasets. We also show the number of videos and frames of different datasets for reference. The unit of time is hour.

Dataset	Video	Frame	Time	
			BANMo	MoDA
AMA	16	2600	10.00	11.00
casual-cat	11	900	8.75	9.50
casual-human	10	584	8.00	9.00
casual-adult	10	1000	9.00	10.25
eagle	5	750	8.25	9.25
hands	5	750	8.25	9.25

more results comparing MoDA with BANMo [66] and ViSER [65] on *casual-human* and *casual-adult* in Figure 13.

Appendix C. Training time

We compare the training times of our method and BANMo [66] on different datasets. We train the models on two RTX 3090 GPUs. As shown in Table 3, MoDA and BANMo both have fast training on different datasets. BANMo takes around 8-10 hours. MoDA takes about one hour more than BANMo. MoDA requires more computational time compared to BANMo due to two primary factors. Firstly, DBS employed in MoDA is inherently more time-consuming than Linear Blend Skinning (LBS), as reported in Figure 18 of [14]. Secondly, the optimization process for solving the optimal transport problem adds additional computational overhead. However, the increased training time is acceptable considering MoDA's superior performance.

References

[1] Marc Badger, Yufu Wang, Adarsh Modh, Ammon Perkes, Nikos Kolotouros, Bernd G Pfrommer, Marc F Schmidt, and Kostas Daniilidis. 3d bird reconstruction: a dataset, model, and shape recovery from a single view. *European Conference on Computer Vision*, pages 1–17, 2020. 2

[2] Benjamin Biggs, Oliver Boyne, James Charles, Andrew Fitzgibbon, and Roberto Cipolla. Who left the dogs out? 3d animal reconstruction with expectation maximization in the loop. *European Conference on Computer Vision*, pages 195–211, 2020. 2

[3] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14124–14133, 2021. 2

[4] Cheng Chen, Xiaofeng Yang, Fan Yang, Chengzeng Feng, Zhoujie Fu, Chuan-Sheng Foo, Guosheng Lin, and Fayao Liu. Sculpt3d: Multi-view consistent text-to-3d generation with sparse 3d prior. *arXiv preprint arXiv:2403.09140*, 2024. 1

[5] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017. 7

[6] Shubham Goel, Angjoo Kanazawa, and Jitendra Malik. Shape and viewpoint without keypoints. *European Conference on Computer Vision*, pages 88–104, 2020. 2

[7] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020. 13

[8] Jim Hejl. Hardware skinning with quaternions. pages 487–495. Charles River Media, 2004. 7

[9] Philipp Henzler, Jeremy Reizenstein, Patrick Labatut, Roman Shapovalov, Tobias Ritschel, Andrea Vedaldi, and David Novotny. Unsupervised learning of 3d object categories from videos in the wild. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4700–4709, 2021. 1, 2

[10] Alec Jacobson, Zhigang Deng, Ladislav Kavan, and John P Lewis. Skinning: Real-time shape deformation (full text not available). pages 1–1. 2014. 3, 5

[11] Wei Jiang, Kwang Moo Yi, Golnoosh Samei, Oncel Tuzel, and Anurag Ranjan. Neuman: Neural human radiance field from a single video, 2022. 2, 3

[12] Angjoo Kanazawa, Shubham Tulsiani, Alexei A Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 371–386, 2018. 2

[13] Ladislav Kavan, Steven Collins, Jiří Žára, and Carol O'Sullivan. Skinning with dual quaternions. *Proceedings of the 2007 symposium on Interactive 3D graphics and games*, pages 39–46, 2007. 2, 3, 5

[14] Ladislav Kavan, Steven Collins, Jiří Žára, and Carol O'Sullivan. Geometric skinning with approximate dual quaternion blending. *ACM Transactions on Graphics (TOG)*, 27(4):1–23, 2008. 15

[15] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9799–9808, 2020. 6, 7

[16] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5253–5263, 2020. 2

[17] Nilesh Kulkarni, Abhinav Gupta, David F Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 452–461, 2020. 6

[18] John P Lewis, Matt Cordner, and Nickson Fong. Pose space deformation: a unified approach to shape interpolation and

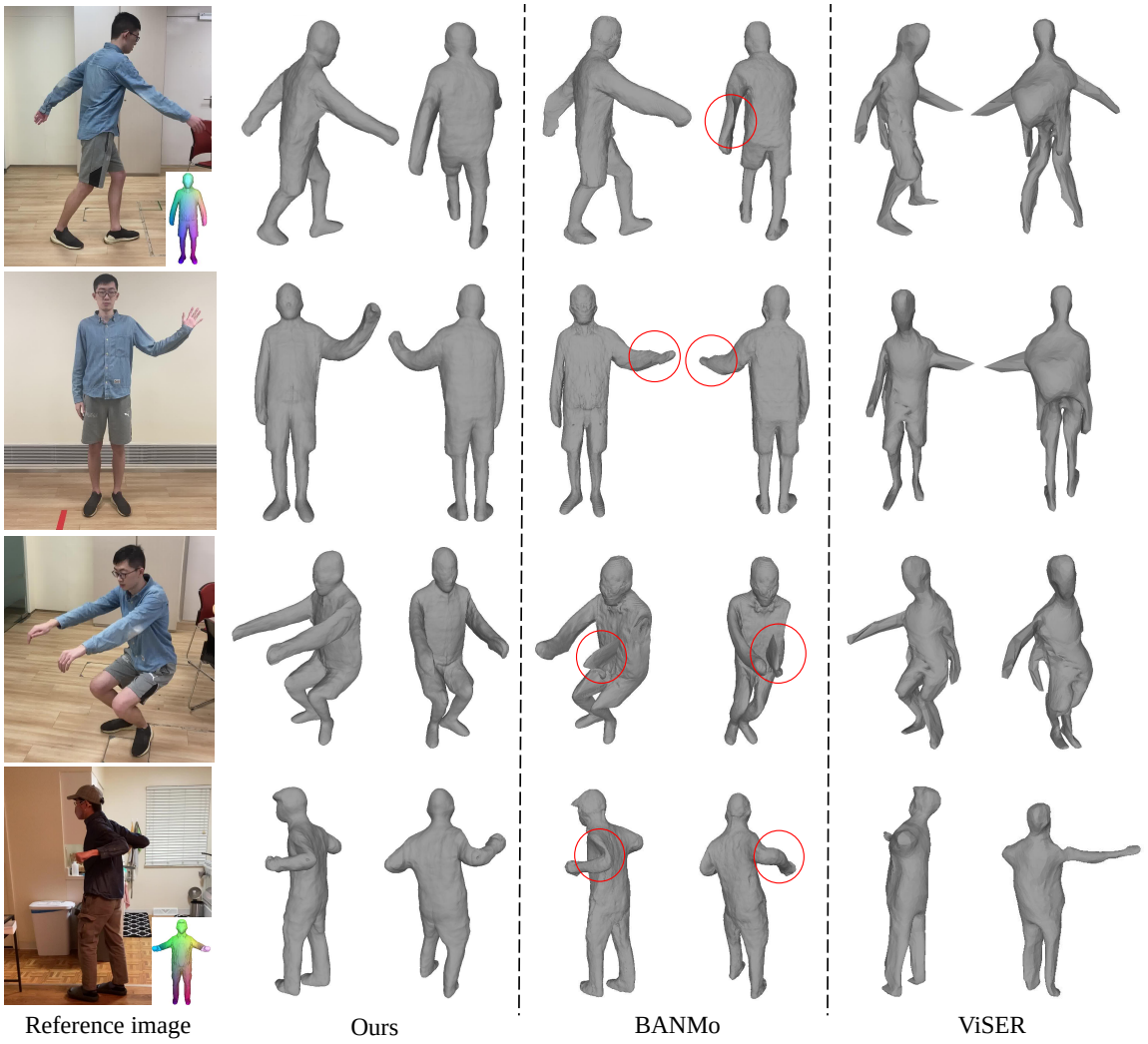

Figure 13: **Qualitative comparison of different methods on *casual-human* and *casual-adult* (multiple-video setups).** We show 2 views of the reconstruction results based on the reference images. ViSER [65] fails to learn detailed 3D shapes and accurate poses from the videos. BANMo [66] has obvious skin-collapsing artifacts (in the red circles) for motions with large joint rotations while our method performs well. The rest poses of *casual-human* and *casual-adult* are shown in the lower right corner.

skeleton-driven deformation. *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 165–172, 2000. 3, 5

[19] Lingzhi Li, Zhen Shen, Li Shen, Ping Tan, et al. Streaming radiance fields for 3d video synthesis. *Advances in Neural Information Processing Systems*. 2

[20] Ruibo Li, Guosheng Lin, and Lihua Xie. Self-point-flow: Self-supervised scene flow estimation from point clouds with optimal transport and random walk. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15577–15586, June 2021. 3

[21] Ruilong Li, Julian Tanke, Minh Vo, Michael Zollhöfer, Jürgen Gall, Angjoo Kanazawa, and Christoph Lassner. Tava: Template-free animatable volumetric actors. *European Conference on Computer Vision*, pages 419–436, 2022. 2

[22] 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. *Advances in Neural Information Processing Systems*, 33:15009–15019, 2020. 2

[23] Xueting Li, Sifei Liu, Kihwan Kim, Shalini De Mello, Varun Jampani, Ming-Hsuan Yang, and Jan Kautz. Self-supervised single-view 3d reconstruction via semantic consistency. *European Conference on Computer Vision*, pages 677–693, 2020. 2

[24] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 6498–6508, 2021. 2, 3, 7, 12

[25] Lingjie Liu, Marc Habermann, Viktor Rudnev, Kripasindhu Sarkar, Jiatao Gu, and Christian Theobalt. Neural actor: Neural free-view synthesis of human actors with pose control. *ACM Transactions on Graphics (TOG)*, 40(6):1–16, 2021. 2, 3

[26] Weide Liu, Chi Zhang, Henghui Ding, Tzu-Yi Hung, and Guosheng Lin. Few-shot segmentation with optimal transport matching and message flow. *IEEE Transactions on Multimedia*, 2022. 3

[27] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 2, 3

[28] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5442–5451, 2019. 2

[29] 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. *Communications of the ACM*, 65(1):99–106, 2021. 2, 4, 6

[30] Natalia Neverova, David Novotny, Marc Szafraniec, Vasil Khalidov, Patrick Labatut, and Andrea Vedaldi. Continuous surface embeddings. *Advances in Neural Information Processing Systems*, 33:17258–17270, 2020. 5

[31] Natalia Neverova, Artsiom Sanakoyeu, Patrick Labatut, David Novotny, and Andrea Vedaldi. Discovering relationships between object categories via universal canonical maps. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 404–413, 2021. 5

[32] Atsuhiro Noguchi, Xiao Sun, Stephen Lin, and Tatsuya Harada. Neural articulated radiance field. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5762–5772, 2021. 2, 3

[33] David Novotny, Diane Larlus, and Andrea Vedaldi. Learning 3d object categories by looking around them. *Proceedings of the IEEE international conference on computer vision*, pages 5218–5227, 2017. 1, 2

[34] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 2

[35] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 2, 3, 7

[36] 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. *ACM Trans. Graph.*, 40(6), dec 2021. 2, 3, 7, 9

[37] 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. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019. 2

[38] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Animatable neural radiance fields for modeling dynamic human bodies. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14314–14323, 2021. 2, 3

[39] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9054–9063, 2021. 2, 3

[40] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 2, 3, 7

[41] Gilles Puy, Alexandre Boulch, and Renaud Marlet. Flot: Scene flow on point clouds guided by optimal transport. *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII*, pages 527–544, 2020. 3

[42] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2304–2314, 2019. 2

[43] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 84–93, 2020. 2

[44] Hanyu Shi, Jiacheng Wei, Ruibo Li, Fayao Liu, and Guosheng Lin. Weakly supervised segmentation on outdoor 4d point clouds with temporal matching and spatial graph propagation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11840–11849, June 2022. 3

[45] Yue Shi, Dingyi Rong, Bingbing Ni, Chang Chen, and Wenjun Zhang. Garf: Geometry-aware generalized neural radiance field. *arXiv preprint arXiv:2212.02280*, 2022. 2

[46] Yue Shi, Yuxuan Xiong, Bingbing Ni, and Wenjun Zhang. Usr: Unsupervised separated 3d garment and human reconstruction via geometry and semantic consistency. *arXiv preprint arXiv:2302.10518*, 2023. 2

[47] Richard Sinkhorn. Diagonal equivalence to matrices with prescribed row and column sums. *The American Mathematical Monthly*, 74(4):402–405, 1967. 6

[48] Chaoyue Song, Jiacheng Wei, Ruibo Li, Fayao Liu, and Guosheng Lin. 3d pose transfer with correspondence learning

and mesh refinement. *Advances in Neural Information Processing Systems*, 34:3108–3120, 2021. 2, 3

[49] Chaoyue Song, Jiacheng Wei, Ruibo Li, Fayao Liu, and Guosheng Lin. Unsupervised 3d pose transfer with cross consistency and dual reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–13, 2023. 2, 3

[50] Liangchen Song, Anpei Chen, Zhong Li, Zhang Chen, Lele Chen, Junsong Yuan, Yi Xu, and Andreas Geiger. Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. *arXiv preprint arXiv:2210.15947*, 2022. 2

[51] Shih-Yang Su, Frank Yu, Michael Zollhöfer, and Helge Rhodin. A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. *Advances in Neural Information Processing Systems*, 34:12278–12291, 2021. 2, 3

[52] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Nonrigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12959–12970, 2021. 3

[53] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. Articulated mesh animation from multi-view silhouettes. pages 1–9. 2008. 6, 7, 9, 10, 13

[54] Minh Phuoc Vo, Yaser A Sheikh, and Srinivasa G Narasimhan. Spatiotemporal bundle adjustment for dynamic 3d human reconstruction in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[55] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 2, 4

[56] Jiacheng Wei, Hao Wang, Jiashi Feng, Guosheng Lin, and Kim-Hui Yap. Taps3d: Text-guided 3d textured shape generation from pseudo supervision. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16805–16815, June 2023. 1

[57] Chung-Yi Weng, Brian Curless, Pratul P Srinivasan, Jonathan T Barron, and Ira Kemelmacher-Shlizerman. Humannerf: Free-viewpoint rendering of moving people from monocular video. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16210–16220, 2022. 2, 3

[58] Qianyi Wu, Xian Liu, Yuedong Chen, Kejie Li, Chuanxia Zheng, Jianfei Cai, and Jianmin Zheng. Object-compositional neural implicit surfaces. *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVII*, pages 197–213, 2022. 6

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

[60] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10974, 2019. 2

[61] Fan Yang, Tianyi Chen, Xiaosheng He, Zhongang Cai, Lei Yang, Si Wu, and Guosheng Lin. Attrihuman-3d: Editable 3d human avatar generation with attribute decomposition and indexing. *arXiv preprint arXiv:2312.02209*, 2023. 2

[62] Fan Yang and Guosheng Lin. Ct-net: Complementary transfering network for garment transfer with arbitrary geometric changes. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9899–9908, June 2021. 3

[63] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. *Advances in neural information processing systems*, 32, 2019. 6, 7, 12

[64] 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. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15980–15989, 2021. 2

[65] 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. *Advances in Neural Information Processing Systems*, 34:19326–19338, 2021. 2, 3, 5, 6, 7, 8, 15, 16

[66] Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. Banmo: Building animatable 3d neural models from many casual videos. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2863–2873, 2022. 1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16

[67] Gengshan Yang, Chaoyang Wang, N Dinesh Reddy, and Deva Ramanan. Reconstructing animatable categories from videos. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16995–17005, 2023. 8, 13

[68] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 4

[69] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33:2492–2502, 2020. 2, 6

[70] Yufei Ye, Shubham Tulsiani, and Abhinav Gupta. Shelf-supervised mesh prediction in the wild. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8843–8852, 2021. 2

[71] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 2

[72] Jason Zhang, Gengshan Yang, Shubham Tulsiani, and Deva Ramanan. Ners: Neural reflectance surfaces for sparse-view 3d reconstruction in the wild. *Advances in Neural Information Processing Systems*, 34:29835–29847, 2021. 5

[73] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. *Proceedings of the*

IEEE/CVF International Conference on Computer Vision, pages 15838–15847, 2021. 6

[74] 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". *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5359–5368, 2019. 2

[75] Silvia Zuffi, Angjoo Kanazawa, and Michael J Black. Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3955–3963, 2018. 2

[76] Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6365–6373, 2017. 2