Learning monocular 3D reconstruction of articulated categories from motion

Filippos Kokkinos
University College London
filippos.kokkinos@ucl.ac.uk

Iasonas Kokkinos
University College London, Snap Inc.
i.kokkinos@cs.ucl.ac.uk

Abstract

Monocular 3D reconstruction of articulated object categories is challenging due to the lack of training data and the inherent ill-posedness of the problem. In this work we use video self-supervision, forcing the consistency of consecutive 3D reconstructions by a motion-based cycle loss. This largely improves both optimization-based and learning-based 3D mesh reconstruction. We further introduce an interpretable model of 3D template deformations that controls a 3D surface through the displacement of a small number of local, learnable handles. We formulate this operation as a structured layer relying on mesh-laplacian regularization and show that it can be trained in an end-to-end manner. We finally introduce a per-sample numerical optimisation approach that jointly optimises over mesh displacements and cameras within a video, boosting accuracy both for training and also as test time post-processing.

While relying exclusively on a small set of videos collected per category for supervision, we obtain state-of-the-art reconstructions with diverse shapes, viewpoints and textures for multiple articulated object categories. Supplementary materials, code, and videos are provided on the project page: https://fkokkinos.github.io/video_3d_reconstruction/.

1. Introduction

Monocular 3D reconstruction of general articulated categories is a task that humans perform routinely, but remains challenging for current computer vision systems. The breakthroughs achieved for humans [3, 17, 10, 47, 29, 21, 30, 16, 4] have relied on expressive articulated shape priors [26] and mocap recordings to provide strong supervision in the form of 3D joint locations. Still, for general articulated categories, such as horses or cows, the problem remains in its infancy due to both the lack of strong supervision [55] and the inherent challenge of representing and learning articulated deformations for general categories.

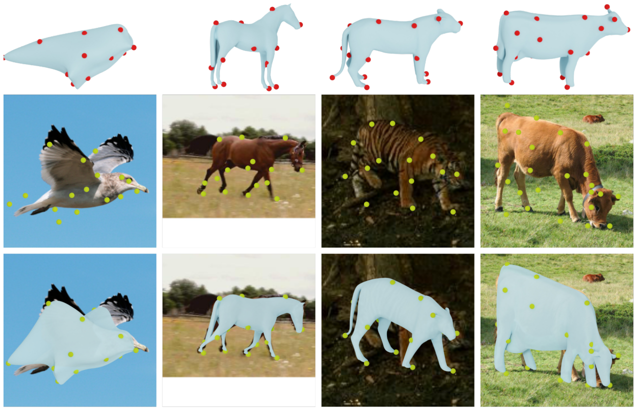

Figure 1: We tackle the problem of monocular 3D reconstruction for articulated object categories by guiding the deformation of a mesh template (top) through a sparse set of 3D control points regressed by a network (middle). Despite using only weak supervision in the form of keypoints, masks and video-based correspondence our approach is able to capture broad articulations, such as opening wings, as well as motion of the lower limbs and neck (bottom).

Recent works have started tackling this problem by relying on minimal, 2D-based supervision such as manual keypoint annotations or masks [43] and learning morphable model priors [43, 19, 18, 9] or hand-crafted mesh segmentations [23]. In this work we leverage the rich information available in videos, and use networks trained for the 2D tasks of object detection, semantic segmentation, and optical flow to complement (optional) 2D keypoint-level supervision.

We make three contributions towards pushing the envelope of monocular 3D object category reconstruction, by injecting ideas from structure-from-motion (SFM), geometry processing and bundle adjustment in the task of monocular 3D articulated reconstruction.

Firstly, we draw inspiration from 3D vision which has traditionally relied on motion information for SFM [38, 11], SLAM [20, 28] or Non-Rigid SFM [39, 8, 7]. These category-agnostic techniques interpret 2D point trajectories in terms of an underlying 3D scene and a moving camera. In this work we use the same principle to supervise monocular

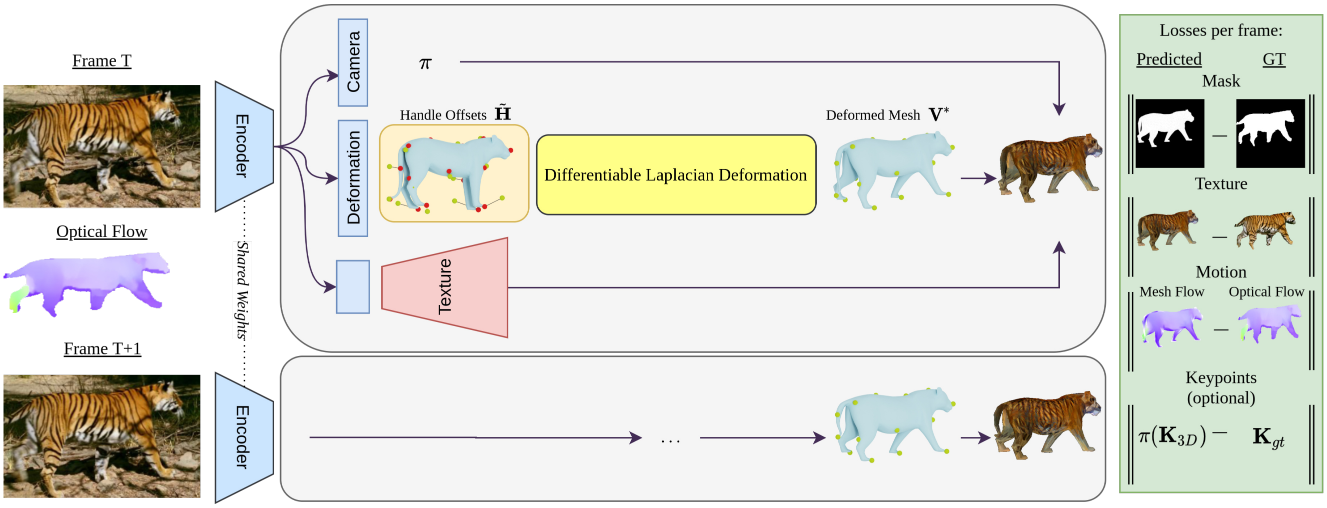

Figure 2: **Training overview:** Two consecutive frames are separately processed by a network that estimates the camera pose, deformation and UV texture parameters. The network regresses per frame a mesh $\mathbf{V}^*$ by estimating offsets to the handles $\mathbf{H}$ of the template shape and consequently solving the respective Laplacian optimization problem. The predictions are supervised by per-frame losses on masks, appearance, and optionally keypoints as well as a novel, intra-frame, motion-based loss that compares the predictions of an optical flow network to the mesh-based prediction of pixel displacements ('mesh flow').

3D category reconstruction, effectively allowing us to leverage video as a source of self-supervision. In particular we establish dense correspondences between consecutive video frames through optical flow and force the back projections of the respective 3D reconstructions to be consistent with the optical flow results. This loss can be back-propagated through the 3D lifting pipeline, allowing us to supervise both the camera pose estimation and mesh reconstruction modules through video. Beyond coming for free, this supervision also ensures that the resulting models will exhibit a smaller amount of jitter and be more flexible when processing videos, since the motion-based loss can penalize inconsistencies across consecutive frames and failure to covary with moving object parts.

Secondly, we introduce a model for regularised mesh deformations that allows for learnable, part-level mesh control and is back-propagateable, providing us with a drop-in replacement to the common morphable model paradigm adopted in [18]. For this we rely on the Laplacian surface deformation algorithm [34], commonly used in geometry processing to deform a template mesh through a set of control points ('handles') while preserving the surface structure and details. We observe that the result of this optimization-based algorithm is differentiable in its inputs, i.e. can be used as a structured layer, while incurring no additional cost at test time since the expression for the optimum can be folded within a linear layer. We incorporate this operation as the top layer of a deep network tasked with regressing the position of the control points given an RGB image. Our results show that we can learn meaningful control points that allow us to capture limb articulations while also providing a human-interpretable interface that enables manual post-processing and refinement using any available 3D software.

Thirdly, we adopt an optimization-based approach to 3D reconstruction that is inspired from bundle adjustment [40]: given a video, we use the 'bottom-up' reconstructions of consecutive frames delivered by our CNN in terms of cameras and handle positions as the initialisation for a numerical optimisation algorithm. We then jointly optimise the per-frame mask and/or keypoint reprojection losses, and video-level motion consistency losses with respect to the cameras and handle variables, giving a 'top-down' refinement of our solution that better matches the image evidence. We show that this improves the results at test-time based on whatever image evidence can be obtained without manual annotation.

We evaluate our approach on 3D shape, pose and texture reconstruction on a range of object categories that exhibit challenging articulations. Our ablation highlights the importance of the employed self-supervised losses and the tolerance of our method to the number of learnable handles, while both our qualitative and quantitative results indicate that our method largely outperforms recent approaches.

2. Related Work

Pose, Texture and Articulation Prediction Our work addresses the task of inferring the camera pose, articulation and texture corresponding to an input image. Recent works have addressed several aspects of this problem [18, 23, 22] with varying forms of supervision. Earlier approaches like CMR [18] treat the problem of 3D reconstruction from single images using known masks and manually labelled keypoints from single viewpoint image collections. Closer to our work is Canonical Surface Mapping (CSM) *et al.* [23, 22] which produces a 3D representation in the form of a rigid or articulated template using a 2D-to-3D cycle-consistency loss. The articulated variant of CSM [22] achieves non-rigid deformation by explicitly segmenting 3D parts of the template shape manually set prior to training the method. Finally, a line of recent research works [32, 35, 46] focus on the disentanglement of images into 3D surfaces with simultaneous camera, lighting and texture prediction without any ground-truth supervision, but are limited to categories of moderate shape variability, such as faces, cats or symmetric objects in general. By contrast to the works above, our method successfully learns models of highly articulated deformable objects without requiring any special preprocessing, such as manual part segmentation, or strong assumptions, such as symmetry.

Surface Deformation Recent works on monocular 3D reconstruction [18, 9] treat deformation as offsets added to mesh vertices, regressed by image-driven CNNs. However regressing vertices can result in surface distortions or corrupt features, while being opaque to a human modeller. By contrast we rely on geometry processing methods [34, 33, 15, 14], and in particular focus on the Laplacian Deformation method [34, 33] which uses a sparse set of control points to achieve a detail-preserving mesh deformation. We realise that the associated optimization problem can be used as a differentiable, structured layer and use it to both learn the control points and efficiently regress their 3D position.

Video-based supervision Video has been commonly used as a source of weak supervision in the context of dense labelling tasks such as semantic segmentation [37] or Densepose estimation [27]. Drawing on the classical use of motion for 3D reconstruction, e.g. [38, 11, 20, 28, 39, 8, 7] many recent works [41, 1, 44] have also incorporated optical flow information to supervise 3D reconstruction networks. Both in the category-specific [41, 1] and agnostic [52, 42, 44] setting, optical flow provides detailed point correspondences inside the object silhouette which can aid the prediction of object articulations and the reconstruction of the underlying 3D geometry. More recent works have leveraged videos for monocular 3D human reconstruction [30] or sparsely-supervised hand-object interactions [12] based on photometric losses. In this work we show that motion is a particularly effective source of supervision for our case, where we jointly learn the category-specific shape prior and the 3D reconstructions. We also rely on robust, occlusion-sensitive optical flow networks [51] which provide a stronger source of supervision than photometric consistency, since they are trained to both handle the aperture effect in the interior of objects to recover large displacement vectors when appropriate.

Cycle Consistency Our approach is reminiscent of the principle of cycle consistency [48, 53, 54], where the composition of two maps is meant to result in the identity mapping.

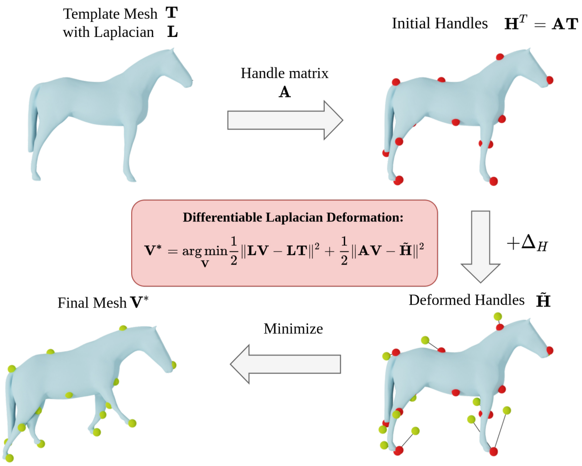
Template Mesh $\mathbf{T}$
with Laplacian $\mathbf{L}$
Initial Handles $\mathbf{H}^T = \mathbf{AT}$
Handle matrix $\mathbf{A}$
Differentiable Laplacian Deformation:
$\mathbf{V}^* = \arg\min_{\mathbf{V}} \frac{1}{2}\|\mathbf{LV} - \mathbf{LT}\|^2 + \frac{1}{2}\|\mathbf{AV} - \tilde{\mathbf{H}}\|^2$
$+\Delta_H$
Final Mesh $\mathbf{V}^*$
Deformed Handles $\tilde{\mathbf{H}}$
Minimize

Figure 3: **Learnable Deformation Layer:** The deformed mesh $\mathbf{V}^*$ is the result of an optimization scheme forcing $\mathbf{V}^*$ to retain the surface details of the template mesh while also minimizing constraints imposed by learnable handles. The optimization solution comes in a closed form, and can be backpropagated through, providing us with a new layer.

Our motion-based approach is in a sense the dual of [53], where 3D synthetic data were used to learn dense correspondences between categories; here we use correspondences from a pre-trained optical flow network to learn about 3D object categories.

3. Method Description

Given an image our target is to infer the 3D shape, camera pose, and texture of the depicted object. During training we only have at our disposal a single representative mesh for the category ('template'), a set of videos, and 2D-level supervision from pre-trained models for semantic segmentation; we can optionally also use ground truth for 2D joints and/or segmentation.

In our approach we use single-frame networks and exploit temporal information only for supervision. At test time we can deploy the learned networks on a per-frame level, but can also exploit temporal information to improve the accuracy of our results through a bundle adjustment-type joint optimization.

In this section we detail our method. We start by introducing our novel representation of an articulated object's 3D shape in terms of a differentiable deformation model in Section 3.1. We then turn to the use of motion as a source of supervision, introducing our motion-consistency loss in Section 3.2. In Section 3.3 we introduce a fine-tuning approach to refine our bottom-up network predictions with a more careful, sample-based optimization and also present other forms of weak supervision used in training.

3.1. Articulated Mesh Prediction

Our aim is to synthesise the shape of an articulated object category by a neural network. While in broad terms we adopt the deformable template paradigm adopted by most recent works [18, 9, 25], we deviate from the morphable model-based [2] modeling of shape adopted in [18, 9, 25]. In those works a shape estimate $\mathbf{V}$ is obtained in terms of offsets $\mathbf{\Delta}_V$ to a template shape $\mathbf{T}$, yielding $\mathbf{V} = \mathbf{\Delta}_V + \mathbf{T}$, where $\mathbf{\Delta}_V$ is delivered by the last, linear, layer of a shape decoder branch, effectively modeling shape variability as an expansion on a linear basis. Such global basis models are well-suited to categories such as faces or cars, but for objects with part-based articulation such as quadrupeds we argue that a part-level model of deformation is more appropriate - which is also the approach routinely taken in rigged modeling in graphics. Furthermore, the linear synthesis model is non-interpretable or controllable by humans and requires careful regularization during training to recover plausible meshes.

We propose instead a deformation model where a set of learnable, network-driven control points (or 'handles') deform a given template while preserving its shape, as captured by its curvature. For this we build on Laplacian surface editing [34]. This model is controllable, interpretable, and regularized by design, while our experiments show that it yields systematically more accurate mesh reconstructions.

In particular we represent the 3D shape of a category as a triangular mesh $M = (V, F)$ with vertices $\mathbf{V} \in \mathbb{R}^{N \times 3}$ and fixed edges $F \in \mathbb{Z}^{N_f \times 3}$. Our deformation approach relies on the cotangent-based discretization $\mathbf{L} \in \mathbb{R}^{N \times N}$ of the continuous Laplace-Beltrami operator used to calculate the curvature at each vertex of a mesh [36].

Instead of manually determining a set of handles, we propose to obtain K 3D handles through a learnable dependency matrix $\mathbf{A} \in \mathbb{R}_{+}^{K \times N}$ that is right-stochastic:

$$\mathbf{H} = \mathbf{A}\mathbf{V}, \quad \text{where} \sum_v \mathbf{A}_{k,v} = 1, \tag{1}$$

forcing each handle to lie in the convex hull of the mesh vertices. For a given image we obtain the target handle positions $\tilde{\mathbf{H}}$ by adding a network-driven update $\mathbf{\Delta}_H$ to the template handles $\mathbf{AT}$: $\tilde{\mathbf{H}} = \mathbf{AT} + \mathbf{\Delta}_H$. Based on $\tilde{\mathbf{H}}$, we obtain the deformed mesh $\mathbf{V}^*$ as the minimum of the following quadratic loss:

$$\mathbf{V}^* = \arg\min_{\mathbf{V}} \frac{1}{2} \|\mathbf{LV} - \mathbf{LT}\|^2 + \frac{1}{2} \left\|\mathbf{AV} - \tilde{\mathbf{H}}\right\|^2, \tag{2}$$

where as in [34] the first term forces the solution to respect the curvature of the template mesh, $\mathbf{LT}$, ensuring that salient, high-curvature details of the template shape are preserved, while the second term forces the location of the handles according to $\mathbf{V}$ to be close to the target location, $\tilde{\mathbf{H}}$.

The stationary point of (2) can be found by solving the following linear system:

$$(\mathbf{L}^\intercal \mathbf{L} + \mathbf{A}^\intercal \mathbf{A})\mathbf{V}^* = \mathbf{L}^\intercal \mathbf{L}\mathbf{T} + \mathbf{A}^\intercal \tilde{\mathbf{H}} \tag{3}$$

Given that $(\mathbf{L}^\intercal \mathbf{L} + \mathbf{A}^\intercal \mathbf{A})$ is symmetric, positive semi-definite and sparse, the solution $\mathbf{V}^*$ can be efficiently computed with conjugate gradients or sparse Cholesky factorization. We rely on efficient solvers that cannot be currently handled by automatic differentiation for backpropagating through the linear system solution, and therefore provide the explicit gradient expression in the supplemental material.

Backpropagating gradients through the Laplacian solver allows us to both learn the association of the vertices to the handles via the matrix $\mathbf{A}$ and also provide gradients back to the handle position $\tilde{\mathbf{H}}$ regressor. As such our method is end-to-end differentiable and no manual annotation, segmentation or rigging of the template mesh is required to achieve part-based articulations.

In practice we initialize the dependency matrix $\mathbf{A}$ based on Farthest Point Sampling (FPS) [6] of the mesh, shortlisting a set of vertices $\{v_k\}, k = 1 \ldots K$ that are approximately equidistant. For each vertex v_k we initialize the $k-$ th row of $\mathbf{A}$ based on the geodesic distance of the vertices to v_k:

$$\mathbf{A}[i, k] = \frac{\exp(1/d_{i,v_k})}{\sum_j \exp(1/d_{j,v_k})} \tag{4}$$

We note that at test time we have a constant affinity matrix, $\mathbf{A}$. Combined with the fixed values of $\mathbf{L}$ and $\mathbf{T}$, we can fold the solution of the linear system in Eq. 3 into a linear layer:

$$\mathbf{V}^* = \mathbf{C} + \mathbf{D}\tilde{\mathbf{H}}, \tag{5}$$

with $\mathbf{C}$ and $\mathbf{D}$ being constant matrices obtained by multiplying both sides of Eq. 3 by the inverse of $\mathbf{L}^\intercal \mathbf{L} + \mathbf{A}^\intercal \mathbf{A}$.

We can thus interpret our method as using at training time template-driven regularization to solve the ill-posed problem of monocular 3D reconstruction, but being as simple and fast as a linear layer at test-time.

3.2. Motion-based 3D supervision

Having described our deformation model, we turn to the use of video information for network training. We rely on optical flow [51] to deliver pixel-level correspondences between consecutive object-centered crops. Unlike traditional 3D vision which relies on category-agnostic point trajectories for 3D lifting, e.g. through factorization [38], we use the flow-based correspondences to constrain the mesh-level predictions of our network in consecutive frames.

In particular, our network takes as input a frame at time t and estimates a mesh $\mathbf{V}_t$ and a weak perspective camera $\mathbf{C}_t$. A mesh vertex i that is visible in both frames t and $t+1$ will project to two image points $\mathbf{p}_{i,t} = \pi(\mathbf{V}_{i,t}, \mathbf{C}_t)$ and $\mathbf{p}_{i,t+1} = \pi(\mathbf{V}_{i,t+1}, \mathbf{C}_{t+1})$ where π amounts to weak perspective projection. As such the displacement of point $\mathbf{p}_{i,t}$ according to our network will be $\tilde{\mathbf{u}}_i = \mathbf{p}_{i,t+1} - \mathbf{p}_{i,t}$.

This prediction is compared to the optical flow value u_i delivered at $\mathbf{p}_{i,t}$ by a pretrained network [51] that we treat as ground-truth. We limit our supervision to image positions in the interior to the object masks and vertices visible in both frames; vertex visibility is recovered by z-buffering, available in any differentiable renderer. We denote the vertices that are eligible for supervision in terms of a binary visibility mask $\boldsymbol{\gamma} : \{1, \ldots, \Gamma\} \rightarrow \{0, 1\}$.

We combine these terms in a 'motion re-projection' loss expressed as follows:

$$L_{\text{motion}} = \frac{1}{\sum_{i=1}^{\Gamma} \gamma_i} \sum_{i=1}^{\Gamma} \gamma_i \left\| \mathbf{u}_i - \tilde{\mathbf{u}}_i \right\|_1 \qquad (6)$$

where we use the ℓ_1 distance between the flow vectors for robustness and average over the number of visible vertices to avoid pose-specific value fluctuations. Since $\tilde{\mathbf{u}}_i = \pi(\mathbf{V}_{i,t+1}, \mathbf{C}_{t+1}) - \pi(\mathbf{V}_{i,t}, \mathbf{C}_t)$ continuously depends on the camera and mesh predictions of our network, we see that this loss can be used to supervise both the camera and mesh regression tasks.

This loss penalizes the cases where limb articulation observed in the image domain is not reflected in the 3D reconstructions, effectively forcing the 3D reconstructions to become more 'agile' by deforming the mesh more actively. Interestingly, we observed that beyond this expected behaviour this loss has an equally important effect on the camera prediction, by forcing the backprojected mesh to 'stand still' in the object interior: even though different camera poses could potentially backproject to the same object in a single image, a change in the camera across frames will cause large 2D displacements for the corresponding 3D vertices. These are picked up when compared to the predictions of an optical flow system that regresses small displacements in the object interior.

3.3. Optimization-based learning and refinement

The objective function for our 3D reconstruction task combines motion supervision with other common losses in a joint objective function:

$$L_{\text{total}} = L_{\text{motion}} + L_{\text{kp}} + L_{\text{pixel}} + L_{\text{rigid}} + L_{\text{mask}} + L_{\text{boundary}}\,, \qquad (7)$$

capturing keypoint, pixel-level appearance, rigidity priors, as well as mask- and boundary- level supervision for the shape; the forms of the losses are provided in Sec 3.3.1, while we omit the empirically-determined loss scaling for simplicity.

In principle a neural network could successfully minimize the sum of these losses and learn the correct 3D reconstruction of the scene. In practice we are asking the network to both recover and learn the solution to an ill-posed problem for multiple training samples, which has many local minima. This has been observed even in strongly-supervised human pose estimation, where careful per-sample numerical optimization [3] was shown to yield substantial performance improvements in [10, 21, 16, 24]. In our weakly-supervised case the local minima problem is even more pronounced.

We use focused, per-sample numerical optimization to refine the network's 'bottom-up' predictions so as to better match the image evidence by minimizing L_{total} with respect to the per-frame handles and camera poses; if the object were rigid this would amount to bundle adjustment, but in our case we also allow the handles to deform per frame. Our approach also applies to both videos and individual frames, where in the latter case we omit the motion-based loss. At test-time, as in the 'synergistic refinement' approach of [10], once the network has delivered its prediction for a test sample (frame/video), we start a numerical 'top-down' refinement of its estimate by minimizing L_{total} using only masks delivered by an instance segmentation network and flow computed from the video if applicable. The approach comes with a computational overhead due to the need for forward-backward passes over the differentiable renderer for every gradient computation.

Further attesting to the importance of per-sample optimization, we note that we have also found a careful initialization of the camera predictions to be critical to the success of our system: as detailed in the supplemental material we train our system by building on the camera multiplex technique [9], that we extend further with the handle deformations.

3.3.1 Loss terms

Keypoint reprojection loss, as in [23], penalizes the ℓ_1 distance between surface-based predictions and ground truth keypoints, when available:

$$L_{\text{kp}} = \sum_i \left\| \mathbf{k}_i - \pi\left(\mathbf{K}_i \mathbf{V}, \mathbf{C}\right) \right\|_1,$$

where $\mathbf{K}_i$ is a fixed vector that regresses the $i-$th semantic keypoint in 3D from the 3D mesh.

Texture Loss compares the mesh-based texture and the image appearance in terms of the perceptual similarity metric of [50] after masking by the silhoutette S:

$$L_{\text{pixel}} = \text{dist}\left(\tilde{I} \odot S, I \odot S\right).$$

As in [18] we enforce symmetric texture predictions by using a bilateral symmetric viewpoint.

Local Rigidity Loss, as in [19] aims at preserving the Euclidean distances between vertices in the extended neighborhood $\mathcal{N}(\mathbf{u})$ of a point $\mathbf{u}$:

$$L_{\text{rigid}} = \underset{\mathbf{u} \in V}{\mathbb{E}} \underset{\mathbf{u}' \in \mathcal{N}(\mathbf{u})}{\mathbb{E}} \left| \left\| V\left(\mathbf{u}\right) - V\left(\mathbf{u}'\right) \right\| - \left\| \bar{V}(\mathbf{u}) - \bar{V}\left(\mathbf{u}'\right) \right\| \right|$$

Method	mIoU	PCK
CMR [18]	0.703	81.2
CSM [23]	0.622	68.5
A-CSM [22]	0.705	72.4
Ours		
8	0.64	84.6
16	0.676	89.8
32	0.688	89.7
64	**0.711**	**91.5**

Table 1: **Ablation of deformation layer on CUB:** Even with only 8 points, our handle-based approach outperforms all competing methods in terms of PCK, while with more handles both the mIoU and PCK scores improve further.

Region similarity loss compares the object support computed from the mesh by a differentiable renderer [31] to instance segmentations S provided either by manual annotations or pretrained CNNs using their absolute distance:

$$L_{\text{mask}} = \sum_i \|S_i - f_{\text{render}}(V_i, \pi_i)\|$$

Chamfer-based loss penalizes smaller areas that are hard to align, like hooves or tails:

$$L_{\text{boundary}} = \mathop{\mathbb{E}}_{\mathbf{u}\in V} \mathcal{C}_{fg}(\pi(\mathbf{u})) + \mathop{\mathbb{E}}_{\mathbf{b}\in\mathcal{B}_{fg}} \min_{\mathbf{u}\in V} \|\pi(\mathbf{u}) - \mathbf{b}\|_2^2,$$

where as in [7, 19] the first term penalizes points of the predicted shape that project outside of the foreground mask using the Chamfer distance to it while the second term penalizes mask under-coverage by ensuring every point on the silhouette boundary has a mesh vertex projecting close to it.

4. Experiments

4.1. Model architecture

We use a similar architecture to CMR [18], using a Resnet18 encoder and three decoders -one each for predicting articulations, camera pose and texture. The articulation prediction module is a set of 2 fully connected layers with $\mathbb{R}^{K\times 3}$ outputs. In particular for texture prediction, we directly predict the RGB pixel values of the UV image through a residual decoder [9]. The texture head is a set of residual upsampling convolution layers that take as input the encoded features of ResNet18 and provide the color-valued UV image; we use Pytorch3D [31] as differentiable renderer. A more thorough description of the individual blocks can be found in the supplemental material.

4.2. Data

We report quantitative reconstruction results for objects with keypoint-annotated datasets, i.e birds, horses, tigers and cows. We have collected a dataset for a wide set of objects, mainly building on available video datasets [5, 49]. All of the videos in our datasets have been filtered manually for occluded or heavily truncated clips that are removed from the dataset. Indicative video samples are provided in the supplemental material; we will make our datasets publicly available to further foster research in this direction.

Birds We use the CUB [45] dataset for training and testing on birds which contains 6000 images. The train/val/test split we use for training and report is that of [18]. While this dataset is single-frame, we use it to compare our deformation module with prior works on similar grounds.

Quadrupeds (Horses, Tigers) We use the TigDog Dataset [5] which contains keypoint-annotated videos of horses and tigers. The segmentation masks are approximate since they are extracted using MaskRCNN [13]. We also drop the neck keypoint for both categories since there is a left-right ambiguity in all annotations. For every class we keep 14 videos purely for evaluation purposes and train with the rest, i.e 53 videos for horses and 44 for tigers. For these classes, the number of handles is set to $K = 16$.

Quadrupeds (cows, giraffes, zebras and 3 others) We use Youtube Video Instance Segmentation dataset (YVIS) [49] to reconstruct more animal classes in 3D. The cow category is used for evaluation since it is the only one for which keypoing ground-truth is available; for the remaining 5 classes we only provide qualitative results in the supplementary material.

For all categories we downloaded template shapes from the internet and downsampled to a fixed number of $N = 642$ vertices. For evaluation we use identical template shape and keypoint annotations to those of [22] for all classes.

4.3. Results

4.3.1 Handle-based deformation evaluation

We start with the CUB [45] dataset where we use the exact supervision of A-CSM [23]. We outperform the state-of-the-art system on reconstruction [18] by a significant margin in both mean Intersection over Union (mIoU) and keypoint reprojection accuracy (PCK), while following their evaluation conventions. We ablate in particular the effect of the number of handles on the achieved 3D reconstruction in Table 1. We observe that our results are outperforming previous methods even with a very small number of handles, however increasing the number of handles allows for improved performance. We also provide qualitative results in Figure 6 where we show that our method is capable of correctly deforming the template mesh to produce highly flexible wings, while the alternative methods barely capture open wing variation. These results clearly indicate the merit of our handle-based deformation layer.

Method	Supervision			Training Dataset	Horse		Tiger
	KP	Mask	Motion		TigDog	Pascal	TigDog
CSM	✓	✓		$P+I$	59.0	46.4	-
ACSM	✓	✓		$P+I$	57.8	57.3	-
ACSM	✓	✓		TD	68.7	44.4	36.2
Ours, inference	✓	✓	✓	TD	74.7	57.2	51.9
Ours, refinement	✓	✓	✓	TD	**83.1**	**69.5**	**55.7**
CSM		✓		$P+I$	44.7	49.7	-
ACSM		✓		$P+I$	58.1	54.2	-
ACSM		✓		$TD+YV$	26.7	33.3	15.1
Ours, inference		✓	✓	$TD+YV$	42.5	31.6	28.4
Ours, refinement		✓	✓	$TD+YV$	**61.3**	**54.9**	**32.5**

Datasets: Pascal (P), ImageNet (I), TigDog (TD), YVIS (YV)

Method	Supervision		Training Dataset	Cow
	Mask	Motion	Pascal	
CSM	✓		$P+I$	37.4
ACSM	✓		$P+I$	43.8
Ours, inference	✓	✓	$TD+YV$	44.6
Ours, refinement	✓	✓	$TD+YV$	**53.9**

Table 2: **Keypoint Reprojection Accuracy** We report PCK accuracy (higher is better) achieved by recent methods [23, 22] for articulate object categories. We indicate datasets used to train each method alongside with supervision method; the CSM/ACSM models trained on P+I do not contain tiger models, while for cows we cannot provide keypoint-supervised results due to the lack of keypoints on videos. Both when training with keypoints and without keypoints we observe substantial improvements over models that were trained without exploiting motion.

Horses	w/ L_{Motion}		w/o L_{Motion}	
	mIoU	PCK	mIoU	PCK
Inference	0.536	74.7	0.519	71.5
Mask refinement	0.691	79.5	0.691	79.5
Mask and motion refinement	0.631	83.1	0.675	72.5

Table 3: **Ablation** of horse reconstruction using motion-based supervision (left vs. right) and optimization-based reconstruction based on masks and motion (rows 1-3).

4.3.2 Motion- and Optimization- based evaluation

In Table 3 we ablate the impact of our motion-based supervision and optimization-based reconstruction for the category of horses. We consider firstly the impact that motion-based supervision has as a source of training (left versus right columns). We observe that motion supervision systematically improves accuracy across all configurations and evaluation measures.

When optimizing at test time as post-processing we observe how the terms that drive the optimization influence the final results: when using only masks we have a marked increase in mIoU, and a smaller increase in PCK, while when taking motion-based terms into account as well the increase in mIoU is not as big but we attain the highest improvement in PCK. We visualize in Figure 4 the mean shape of the horse along with the first 3 common deformation modes.

4.3.3 Comparisons on more categories

In Table 2 we report results on more categories where we have been able to compare to the currently leading approaches to monocular 3D reconstruction [18, 23, 22]. We use a small number of videos (53 for horses, 44 for tigers, 24 for cows) compared to the thousands of images available in Imagenet and Pascal used by the existing approaches.

Starting with the comparison on horses for the case where keypoints are available, we observe that our inference-only method has a clear lead when testing on the TigDog dataset (the other methods have not been trained on TigDog), while optimization results in a further boost. When tested on Pascal, our inference-only results are comparable to the best, while optimization gives us a clear edge. For cows we did not have videos with cow keypoints, as such we did not train our approach on it. Furthermore, we trained ACSM for horses and tigers on the TigDog dataset in order to have fair comparison to our method. The TigDog-trained ACSM got a significant boost on TigDog test set but the performance was still not on par with our result.

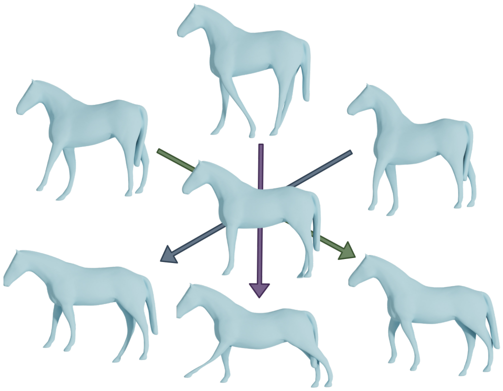

Figure 4: **Learned Deformations** Visualization of the predicted deformations by depicting the mean shape in the center and the first 3 modes obtained by PCA on the handle estimates obtained across the dataset.

Turning to results where we do not use keypoints, we observe that when used in tandem with post-processing optimization our method outperforms both CSM and ACSM, while when compared to ACSM trained on the same data we have a substantial boost on TigDog-Horse and TigDog-Tiger. Overall we observe a larger drop in accuracy compared to the results obtained when keypoint supervision is available. As we show in the supplemental material, this may be due to the

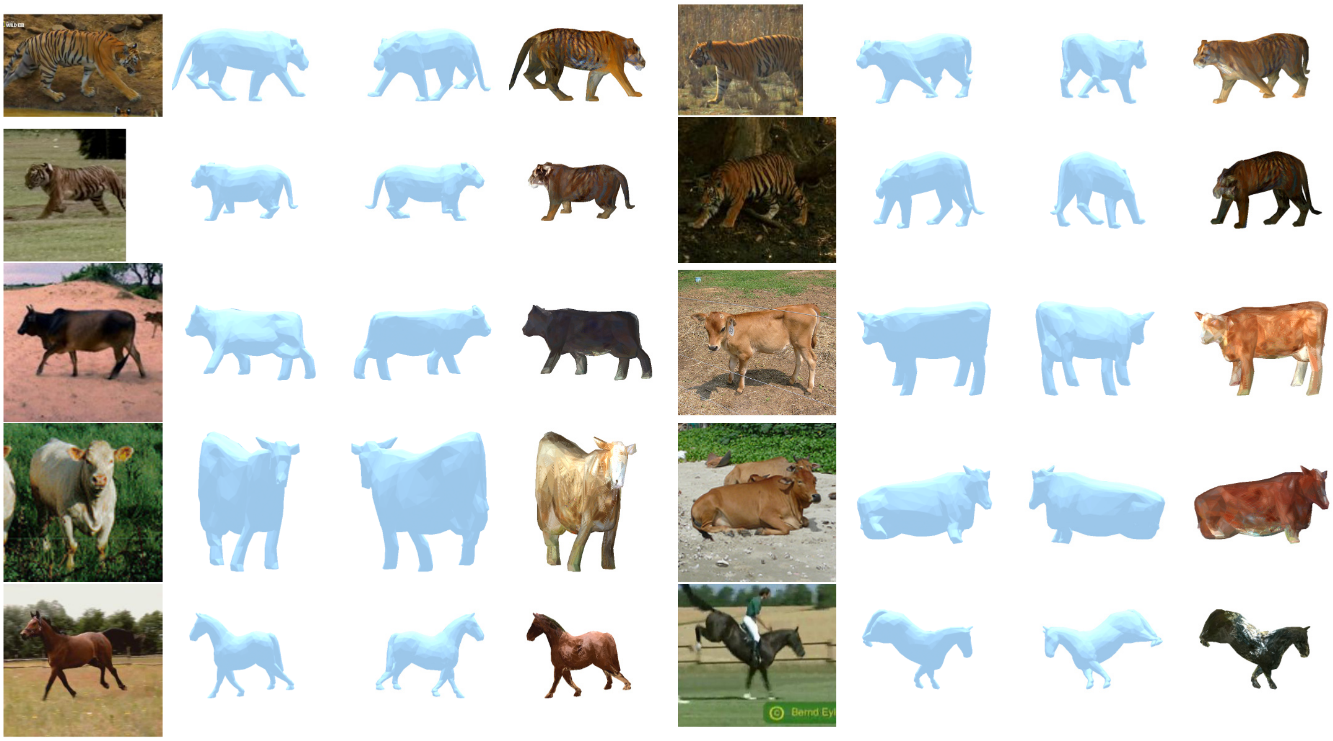

Figure 5: **Quadruped reconstructions** of our method. We provide renderings of the 3D reconstruction using the estimated camera pose, a different viewpoint and the texture reconstruction. We observe that our method successfully captures large articulated deformations as well as viewpoint variability. For videos of side-by-side comparisons to [22] please visit `https://fkokkinos.github.io/video_3d_reconstruction/`.

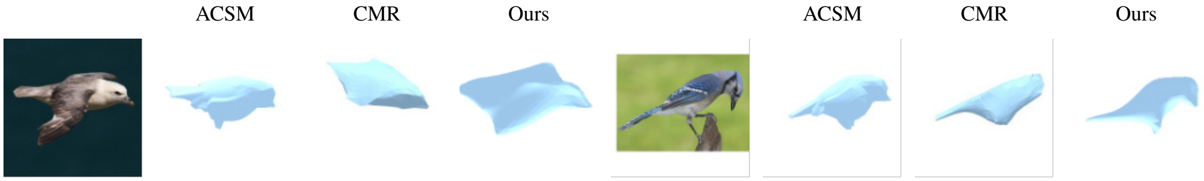

Figure 6: **Bird reconstructions** For each input image we provide the results of CMR [18] and ACSM [22] alongside with our method. We observe that we better capture wing and beak deformation.

large flexibility of our deformable model, which manages to "overfit" to the mask rather than performing the appropriate global, rigid transforms. For the case of cows we observe that even though our model was never trained on Pascal data, it outperforms the mask-supervised variants of ACSM.

A pattern that is common for both sets of results is that post-processing optimization yields a substantial improvement in accuracy. As our qualitative results indicate in Figure 5 and the Supplemental, this is reflected also in the large amount of limb articulation achievable by our model. Failure cases, provided in the supplementary material are predominantly due to wrong global camera parameters such as scale, which we attribute to the small diversity of appearance in our limited set of videos. We anticipate further improvements in the future by combining diverse images from static and strong, motion-based supervision from dynamic datasets. Finally, in some cases our model fails to predict good textures commonly for moving parts of quadrupeds like the legs.

5. Conclusion

We have presented a motion- and geometry-based deep learning framework for monocular reconstruction that combines ideas from deep learning and geometry for the unsupervised reconstruction of highly articulated objects; we anticipate that the interpretable and controllable nature of our approach will help handle multiple animate object classes in augmented reality and graphics.

References

[1] Thiemo Alldieck, Marc Kassubeck, Bastian Wandt, Bodo Rosenhahn, and Marcus Magnor. Optical flow-based 3d human motion estimation from monocular video. In Volker Roth and Thomas Vetter, editors, *Pattern Recognition*, pages 347–360, Cham, 2017. Springer International Publishing. 3

[2] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999. 4

[3] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science. Springer International Publishing, Oct. 2016. 1, 5

[4] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. Monocular expressive body regression through body-driven attention. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part X*, 2020. 1

[5] L. Del Pero, S. Ricco, R. Sukthankar, and V. Ferrari. Articulated motion discovery using pairs of trajectories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 6

[6] Y Eldar. *Irregular image sampling using the Voronoi diagram*. PhD thesis, M. Sc. thesis, Technion-IIT, Israel, 1992. 4

[7] Katerina Fragkiadaki, Han Hu, and Jianbo Shi. Pose from flow and flow from pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2059–2066, 2013. 1, 3, 6

[8] Ravi Garg, Anastasios Roussos, and Lourdes Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 1, 3

[9] Shubham Goel, Angjoo Kanazawa, , and Jitendra Malik. Shape and viewpoints without keypoints. In *ECCV*, 2020. 1, 3, 4, 5, 6

[10] Riza Alp Guler and Iasonas Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 5

[11] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 1, 3

[12] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 6

[14] Alec Jacobson, Ilya Baran, Ladislav Kavan, Jovan Popović, and Olga Sorkine. Fast automatic skinning transformations. *ACM Transactions on Graphics (TOG)*, 2012. 3

[15] Alec Jacobson, Ilya Baran, Jovan Popovic, and Olga Sorkine. Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph.*, 2011. 3

[16] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human model fitting towards in-the-wild 3d human pose estimation, 2020. 1, 5

[17] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of Human Shape and Pose. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[18] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 1, 2, 3, 4, 5, 6, 7, 8

[19] Abhishek Kar, Shubham Tulsiani, Joao Carreira, and Jitendra Malik. Category-specific object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1966–1974, 2015. 1, 5, 6

[20] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *2007 6th IEEE and ACM international symposium on mixed and augmented reality*, pages 225–234. IEEE, 2007. 1, 3

[21] Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 1, 5

[22] Nilesh Kulkarni, Abhinav Gupta, David F Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 452–461, 2020. 2, 3, 6, 7, 8

[23] Nilesh Kulkarni, Abhinav Gupta, and Shubham Tulsiani. Canonical Surface Mapping via Geometric Cycle Consistency. *International Conference on Computer Vision (ICCV)*, 2019. 1, 2, 3, 5, 6, 7

[24] Dominik Kulon, Riza Alp Guler, Iasonas Kokkinos, Michael M. Bronstein, and Stefanos Zafeiriou. Weakly-supervised mesh-convolutional hand reconstruction in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 5

[25] Xueting Li, Sifei Liu, Kihwan Kim, Shalini De Mello, Varun Jampani, Ming-Hsuan Yang, and Jan Kautz. Self-supervised Single-view 3D Reconstruction via Semantic Consistency. In *ECCV*, 2020. 4

[26] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: a skinned multi-person linear model. *ACM Trans. Graph.*, 34(6):248:1–248:16, 2015. 1

[27] Natalia Neverova, James Thewlis, Riza Alp Güler, Iasonas Kokkinos, and Andrea Vedaldi. Slim densepose: Thrifty learning from sparse annotations and motion cues. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 2019. 3

[28] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. Dtam: Dense tracking and mapping in real-time. In *2011 international conference on computer vision*. IEEE, 2011. 1, 3

[29] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[30] Georgios Pavlakos, Nikos Kolotouros, and Kostas Daniilidis. Texturepose: Supervising human mesh estimation with texture consistency. In *International Conference on Computer Vision, ICCV*, 2019. 1, 3

[31] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020. 6

[32] Mihir Sahasrabudhe, Zhixin Shu, Edward Bartrum, Riza Alp Güler, Dimitris Samaras, and Iasonas Kokkinos. Lifting autoencoders: Unsupervised learning of a fully-disentangled 3d morphable model using deep non-rigid structure from motion. 2019. 3

[33] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, pages 109–116, 2007. 3

[34] Olga Sorkine, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Rössl, and H-P Seidel. Laplacian surface editing. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 175–184, 2004. 2, 3, 4

[35] Attila Szabó, Givi Meishvili, and Paolo Favaro. Unsupervised generative 3d shape learning from natural images. *CoRR*, abs/1910.00287, 2019. 3

[36] Gabriel Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 351–358, 1995. 4

[37] P. Tokmakov, K. Alahari, and C. Schmid. Weakly-supervised semantic segmentation using motion cues. In *ECCV*, 2016. 3

[38] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: a factorization method. *International journal of computer vision*, 9(2):137–154, 1992. 1, 3, 4

[39] Lorenzo Torresani, Aaron Hertzmann, and Christoph Bregler. Learning non-rigid 3D shape from 2D motion. *Advances in neural information processing systems*, 2003. 1, 3

[40] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle Adjustment - A Modern Synthesis. In *Vision Algorithms: Theory and Practice*, 1999. 2

[41] Hsiao-Yu Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5236–5246. Curran Associates, Inc., 2017. 3

[42] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. Demon: Depth and motion network for learning monocular stereo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[43] Sara Vicente, João Carreira, de Lourdes Agapito, and Jorge Batista. Reconstructing pascal voc. *Computer Vision and Pattern Recognition*, pages 41–48, 2014. 1

[44] Sudheendra Vijayanarasimhan, Susanna Ricco, Cordelia Schmid, Rahul Sukthankar, and Katerina Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*, 2017. 3

[45] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 6

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

[47] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 10965–10974. Computer Vision Foundation / IEEE, 2019. 1

[48] Qi xing Huang and Leonidas Guibas. Consistent shape maps via semidefinite programming. In *In Computer Graphics Forum*, pages 177–186. Wiley Online Library, 2013. 3

[49] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5188–5197, 2019. 6

[50] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 5

[51] Shengyu Zhao, Yilun Sheng, Yue Dong, Eric I-Chao Chang, and Yan Xu. MaskFlownet: Asymmetric Feature Matching with Learnable Occlusion Mask. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3, 4, 5

[52] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 3

[53] Tinghui Zhou, Philipp Krähenbühl, Mathieu Aubry, Qixing Huang, and Alexei A. Efros. Learning dense correspondence via 3d-guided cycle consistency. 2016. 3

[54] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017. 3

[55] Silvia Zuffi, Angjoo Kanazawa, Tanya Y. Berger-Wolf, and Michael J. Black. Three-d safari: Learning to estimate zebra pose, shape, and texture from images "in the wild". In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 2019. 1