DensePose 3D: Lifting Canonical Surface Maps of Articulated Objects to the Third Dimension

Roman Shapovalov David Novotny Benjamin Graham Patrick Labatut Andrea Vedaldi
Facebook AI Research

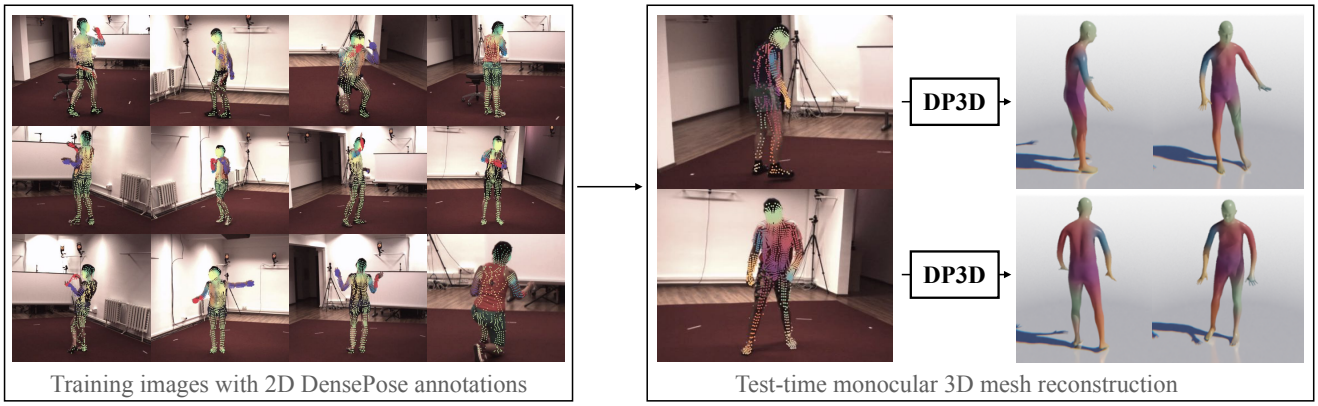

Figure 1: We propose DensePose 3D (DP3D), a method for monocular mesh recovery, which leverages a novel parametric mesh articulation model. Crucially, the model is trained in a weakly supervised manner on a dataset of single views of humans or animals in different poses and their DensePose labelling produced by an off-the-shelf pre-trained detector.

Abstract

We tackle the problem of monocular 3D reconstruction of articulated objects like humans and animals. We contribute DensePose 3D, a method that can learn such reconstructions in a weakly supervised fashion from 2D image annotations only. This is in stark contrast with previous deformable reconstruction methods that use parametric models such as SMPL pre-trained on a large dataset of 3D object scans. Because it does not require 3D scans, DensePose 3D can be used for learning a wide range of articulated categories such as different animal species. The method learns, in an end-to-end fashion, a soft partition of a given category-specific 3D template mesh into rigid parts together with a monocular reconstruction network that predicts the part motions such that they reproject correctly onto 2D DensePose-like surface annotations of the object. The decomposition of the object into parts is regularized by expressing part assignments as a combination of the smooth eigenfunctions of the Laplace-Beltrami operator. We show significant improvements compared to state-of-the-art non-rigid structure-from-motion baselines on both synthetic and real data on categories of humans and animals.

1. Introduction

Recent advances in deep learning have produced impressive results in monocular 3D reconstruction of articulated and deformable objects, at least for particular object categories such as humans. Unfortunately, while such techniques are general in principle, their success is rather difficult to replicate in other categories. Before learning to reconstruct 3D objects from images, one must first learn a model of the possible 3D shapes of the objects. For humans, examples of such models include SMPL [37] and GHUM [62]. Constructing these requires a large dataset of 3D scans of the objects deforming and articulating over time, which have to be acquired with specialised devices such as domes. Not only this hardware is uncommon, complex and expensive, but it is also difficult if not impossible to apply to many objects of interest, such as wild animals or even certain types of deformable inanimate objects. Then, after building a suitable 3D shape model, one still has to train a deep neural network regressor that can predict the shape parameters given a 2D image of the object as input [29, 63, 26]. Supervising such a network requires in turn a dataset of images paired with the corresponding ground-truth 3D shape parameters. Images with paired reconstructions are also very difficult to obtain in practice.

Some images may be available from the same scanners that have been used to construct the 3D model in the first place, but these are limited to 'laboratory condition' by definition. Thus, while there is abundance of 'in the wild' images of diverse object categories that can be obtained from the Internet, they are lacking 3D ground-truth and are thus difficult to use for learning 3D shape predictors.

In this paper, we are interested in bootstrapping 3D models and monocular 3D predictors *without* using images with corresponding 3D annotations or even unpaired 3D scans. Fortunately, other modalities can provide strong cues for reconstruction. For example, previous work [26, 43, 30, 15] leveraged 2D annotations for semantic keypoints to accurately reconstruct various object categories. While these keypoints provide a supervisory signal at sparse image locations, DensePose [21, 40, 50] provides *dense* correspondences between the images of humans or other animals and 3D templates of these categories. Example of these annotations are shown on the left of Figure 1, where the colours encode the indices of corresponding points on the template mesh. DensePose annotations can be seen as generalising sparse joint locations, with two important differences: the density is much higher, and the correspondences are defined on the *surface* of the object rather than in its skeleton joints. Such dense annotations can be obtained manually or with detectors pre-trained on those manual 2D annotations, with the same degree of flexibility and generality as sparse 2D landmarks, while providing much stronger cues for learning detailed 3D models of the objects. However, such annotations do not appear to have been used to bootstrap 3D object models before.

The main goal of this work is thus to leverage dense surface annotations, such as the ones provided by DensePose, in order to learn a parametric model of a 3D object category without using any 3D supervision. As done in [26, 43, 30, 15], we further aim to learn a deep neural network predictor that aligns the model to individual 2D input images containing the object of interest. Our method assumes only having an initial *rigid* canonical 3D template of the object category generated by a 3D artist. There is no loss of generality here since knowledge of the template is required to collect DensePose annotations in the first place.[1] Thus, pragmatically, we include this template in our model.

Our main contribution is a novel parametric mesh model for articulated object categories, which we call *DensePose 3D* (DP3D). In a purely data-driven manner, DP3D learns to softly assign the vertices of the initial rigid template to one of a number of latent parts, each of which moving in a rigid manner. The parametrization of the mesh articulation is then given by a set of per-part rigid transforms expressed in the space of the logarithms of $\mathsf{SE}(3)$. In order to pose the mesh, each vertex of the template shape is deformed with a vertex-specific transformation defined as a convex combination of the part-specific transforms, where the weights are supplied by the soft segmentation of the corresponding vertex. In order to prevent unrealistic shape deformations, we enforce smoothness of the part segmentation, and consequently of the vertex-specific offsets, by expressing the part assignment as a function of a truncated eigenbasis of the Laplace-Beltrami operator computed on the template mesh, which varies smoothly along the mesh surface. We further regularise the mesh deformations with the as-rigid-as-possible (ARAP) soft constraint.

DP3D is trained in a weakly supervised manner, in the sense that our pipeline (including DensePose training) does not require 3D annotations for the input images. In an end-to-end fashion, we train a deep pose regressor that, given a DensePose map extracted from an image, predicts the shape deformation parameters, poses the mesh accordingly, and minimises the distance between the projection of the posed mesh to the image plane and the input 2D DensePose annotations. We show that our method does not need manual DensePose annotations for the training images; it can learn even from the predictions of a DensePose model trained on a different dataset. This way, DP3D can learn to infer the shape of humans and animals from an unconstrained dataset containing diverse poses. Since DP3D does not use images directly but only the DensePose annotations or predictions, it is robust to changes in the object appearance statistics, which makes it suitable for transfer learning.

We conduct experiments on a synthetic dataset of human poses, and on the popular Human 3.6M benchmark, showing that the model trained on staged Human 3.6M generalises to a more natural 3DPW dataset. We also fit the models to animal categories in the LVIS dataset. Note that learning reconstruction of LVIS animals would be impossible with any method requiring 3D supervision since there are no scans or parametric models available for species like bears or zebras. DP3D produces more accurate reconstructions than a state-of-the-art Non-rigid Structure-from-Motion (NR-SfM) baseline and compares favourably with fully-supervised approaches.

2. Related work

In this section we review the relevant prior art: monocular human mesh reconstruction, canonical surface maps, and non-rigid SfM.

Image-based human body reconstruction. A popular method for reconstructing 3D humans from 2D images is test-time optimisation, where a parametric human model such as SMPL [37] or SCAPE [6] is fitted to a given test image by minimising various types of energies, including

[1] The 3D template is used by the human annotators as a reference to mark correspondences and defines the canonical surface mapping for the object category.

2D keypoint and mask reprojection losses [20, 52, 12, 36, 23, 64, 24, 44, 61]. Alternatively, one can learn a deep regressor which, given a single image as input, predicts the parameters of the 3D shape model directly. Most methods [7, 39, 48, 54, 29, 38] reconstruct only a sparse set of 3D points, usually corresponding to 2D body joint detections. HMR [25] and GraphCMR [29] regress instead full 3D meshes. Kolotouros *et al*. [28] combine the test-time optimization and deep regression paradigms. Biggs *et al*. [9] regress multiple mesh hypotheses to deal with the inherent ambiguity of monocular 3D reconstruction. While such methods achieve state-of-the-art monocular human mesh recovery, they require large dataset with 3D annotations to train the 3D shape model and the regressor. In contrast, our method is trained only with 2D image annotations.

Self-supervised 3D human pose estimation. Other methods aim at reconstructing 3D body skeletons without 3D annotations. Some works leverages multi-view constraints [27, 46, 47] while Pavlakos *et al*. [45] assume ordinal depth supervision. Alternatively, adversarial networks can also be used to learn 3D models from 2D annotations in a monocular setup [31, 17, 14]. The idea is to train a discriminator that tells if the 2D reprojection of the reconstructed 3D points from multiple random views is plausible or not. While these methods work well, their inability to deal with occluded keypoints makes them unsuitable for dense reconstruction.

Canonical surface maps. DensePose [21] was perhaps the first method to predict dense assignments from an image to a reference 3D template of the human body, also called a *Canonical Surface Map* (CSM). It introduced a dataset with manually labelled correspondences as well as a new deep network architecture to regress dense correspondences from images. Follow-up work introduced semi-supervised learning [41] and transferred human correspondences to quadrupeds [50]. Most recently, Neverova *et al*. [40] reformulated DensePose as a non-parametric problem by predicting canonical point embeddings for image pixels, which facilitates its application to a wider range of deformable object categories.

Other works aimed at learning CSMs with limited or no supervision: [56, 55, 51] do so by using principles such as transformation equivariance, whereas [33] enforces consistency with an initial 3D model of the object. Relevant to our work, the articulation-aware variant of it [32] produces canonical surface maps for categories such as quadruped animals. The method requires a segmented template mesh with a predefined skeleton structure; in contrast, we learn the articulated structure automatically without supervision.

Non-rigid structure-from-motion. NR-SfM is relevant to our work as its goal is to reconstruct a deformable 3D object from 2D keypoint annotations. The seminal work of Bregler [13], which proposed to express the possible deformations of the 3D shape as a linear combination of a small number of basis shapes, has since inspired many follow-up works [3, 18, 16, 67, 4, 5, 1, 19, 34, 35, 67, 2, 65, 66, 57]. Traditionally, such methods posed the problem as matrix factorization, but more recently some alternative that leverage deep learning have emerged. DeepNRSfM [30, 59] and, more relevantly to our work, C3DPO [43] train an MLP that maps the vectorised list of 2D keypoints to camera and shape parameters and minimise the distance between the input 2D keypoints and the 3D point reprojections. While C3DPO works well with sparse keypoints such as the human joints, as we show in the experiments, it fails to handle the dense collections of points required to reconstruct meshes. We address this issue by utilising the known category-level template mesh to learn deformations compatible with the articulation of a latent skeletal structure.

3. Method

We aim to learn reconstructing the 3D shape of a deformable object such as a human or an animal from 2D images, and to do so without 3D supervision. Instead, we only use dense 2D object points that can be annotated manually or predicted by means of a method such as DensePose, also known as a canonical surface map (CSMs).

We summarise the necessary CSM background in section 3.1 and then discuss our method.

3.1. Canonical surface maps

A CSM [56, 21, 41, 40, 33, 50, 32] is defined with respect to a reference 3D template, usually given as a triangular mesh with vertices $\mathbf{V} = (V_k)_{k=1}^K \in \mathbb{R}^{K\times 3}$. For humans, for example, a common reference mesh is the SMPL rest pose (which was created by a 3D artist).

A CSM such as DensePose takes as input an image $I : \Omega \to \mathbb{R}$ of the object and assigns to each pixel $y \in \Omega$ a point in the mesh $\mathbf{V}$, producing a map $\Omega \to \mathbf{V}$.[2] While this is useful information, it is not yet a 3D reconstruction of the object in the image because $\mathbf{V}$ is a fixed reference template. In order to obtain a 3D reconstruction, we need instead to *pose* the template by finding a suitable deformation $\mathbf{X} = (X_k)_{k=1}^K \in \mathbb{R}^{K\times 3}$ of its vertices.

As the first step in the posing process, we 'reverse' the CSM output and, for each vertex V_k of the template, find its corresponding pixel location y_k, resulting in a collection of 2D vertex locations $\mathbf{Y} = (y_k)_{k=1}^K \in \mathbb{R}^{K\times 2}$. Due to occlusions, a vertex may be invisible in the image, which prevents extracting its 2D location y_k from the CSM. Thus we also define visibility indicators $\mathbf{Z} = (z_k)_{k=1}^K \in \{0,1\}^K$.

Note that $\mathbf{Y}$ can also be obtained from the posed mesh $\mathbf{X}$ and the camera projection function π_I as $y_k = \pi_I(X_k)$.

[2] In practice, the map is valued in $\mathbf{V} \cup \{\text{bkg}\}$ to allow to mark pixels that do not belong to the object as background.

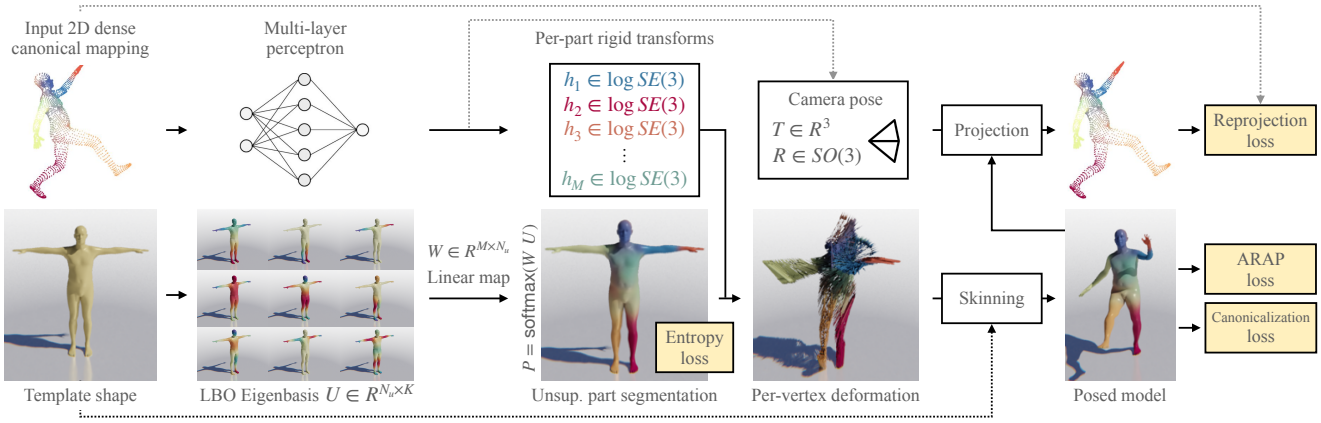

Figure 2: **Overview of our method.** The input 2D keypoints $\mathbf{Y}$ are passed to the network Φ that predicts global and per-part rigid transformations. LBO harmonics are used to regress the soft part segmentation $\mathbf{P}$. The transformations, part segmentation, along with the template mesh $\mathbf{V}$, are used for Linear Blend Skinning to obtain the shape $\mathbf{X}$. During training, this shape enters re-projection, canonicalisation, and ARAP losses, while the entropy loss is defined on part segmentation.

This calculation does not involve the CSM at all and, as we show later, can be used to constrain the reconstruction.

3.2. Shape model

In order to reconstruct the 3D shape of an object from 2D annotations, we must define a *shape model* that constrains the space of possible reconstructions $\mathbf{X}$. To this end, we assume that the underlying object, which could be a human or another animal, has a skeletal structure. Under this assumption, the pose of the object is expressed by the rigid transformations of M parts

$$g_m = (R_m, T_m) \in SE(3), \quad m = 1, \dots, M. \tag{1}$$

We assume that each vertex $\mathbf{V}_k$ in the template belongs to one of the M parts with membership strength $P_{km} \in [0, 1]$ such that $\sum_{m=1}^{M} P_{km} = 1$. The posed vertices $\mathbf{X}$ are given by the linear combination of part transformations, as in linear blend skinning (LBS):

$$X_k = \sum_{m=1}^{M} P_{km} \cdot g_m(g_{0m}^{-1}(V_k)). \tag{2}$$

Here $g_{0m} \in SE(3)$ stands for the rest pose of the m-th part.

While we do not force the parts to have a particular semantic, we expect learning to group together surface points that move rigidly together, e.g all points on a forearm. Next, we explain how we encourage such a solution to emerge.

Part segmentation. Having defined per-vertex deformations, we will now describe the part segmentation model $\mathbf{P} = [P_{km}] \in \mathbb{R}^{K \times M}$. As mentioned before, unlike other parametric models [37, 6], we do not require a pre-segmented template shape. Instead, we treat the part segmentation $\mathbf{P}$ as a latent variable and learn it together with the rest of the model parameters. Note that the part segmentation is independent of a particular input instance — this means that the part assignments stay constant once training finishes. Intuitively, limiting the number of parts and constraining deformations within parts to rigid ones should force the model to group the vertices that move according to the same rigid transform into the same part.

Smooth segmentation with LBO. While we have reduced the deformation of the template to the rigid motions of a small number of parts ($M = 10$), the assignment of the template vertices to the different parts can still be irregular, which may lead to unrealistic body deformations. We address this issue by enforcing the part assignments $\mathbf{P}$ to be smooth. Combined with eq. (2), this encourages the deformations of the template to be smooth as well.

We formalise this intuition by requiring the part assignment $\mathbf{P}$ to be a smooth function on the mesh surface. This can be enforced by making sure that $\mathbf{P}$ only contains 'low frequency' components. Formally, this is achieved by expressing $\mathbf{P}$ as a linear combination of selected eigenfunctions of the Laplace-Beltrami operator (LBO [49]), illustrated in Figure 3.

In more detail, consider the discrete approximation Δ of the LBO for the reference template mesh $\mathbf{V}$. Let $\mathbf{u}_i \in \mathbb{R}^K$ be the (orthonormal) eigenvectors of Δ sorted by increasing eigenvalue magnitude, and let $\mathbf{U} = (\mathbf{u}_i)_{i=1}^{N_u} \in \mathbb{R}^{K \times N_u}$ be the matrix containing the N_u first eigenvectors. We define the part segmentation as

$$\mathbf{P} = \text{softmax}(\mathbf{UW}), \tag{3}$$

where $\mathbf{W} \in \mathbb{R}^{N_u \times M}$ is a parameter matrix, and the softmax is taken with respect to the part index k.

Smoothness can be further increased by reducing N_u or by initialising $\mathbf{W} = [W_{im}]$ with decreasing magnitude.

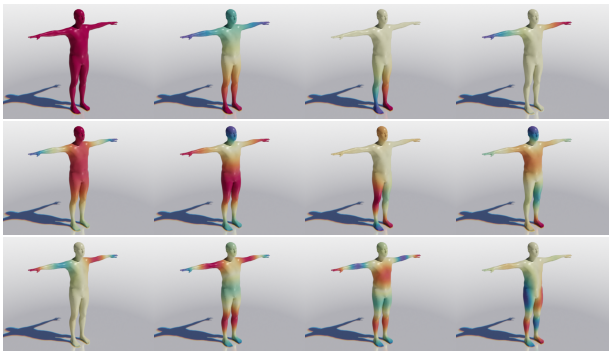

Figure 3: We express the per-vertex deformations as a linear map of the Laplace–Beltrami eigenbasis of the template shape. The figure shows 12 most significant eigenvectors of the LB operator of a human template mesh. These top eigenfunctions vary smoothly along the surface of the mesh, which enforces similarity of the neighboring per-vertex transforms, leading to natural mesh deformations.

Specifically, we use a variant of Xavier initialisation and set $W_{im} \sim \mathcal{N}(0, \frac{\exp(-i/\bar{\sigma})}{M^{1/2}})$. This focuses the model on low-frequency harmonics at the beginning of training.

Transformations predictor. Given input 2D keypoint locations $\mathbf{Y}$ and visibilities $\mathbf{Z}$, we train a multi-layer perceptron (MLP) to predict the $(M+1)$ rigid part transformations:

$$\{h_m\}_{m=0}^{M} = \Phi(\mathbf{Y}, \mathbf{Z}). \tag{4}$$

We express transformations in log-space, meaning that $(R_m, T_m) = g_m = \exp(h_m)$ where $\exp : \mathbb{R}^6 \rightarrow \mathbb{R}^{3\times3} \times \mathbb{R}^3$ is the exponential map of $SE(3)$; see [11] for details.

Note that we estimate the additional global transformation h_0; this is the camera pose used to re-project the posed shape, which is expressed in the object reference frame, back to the image (see eq. (5)). Note also that eq. (2) requires the inverse part transformation at rest g_{0m}^{-1}; these are learnt as logarithms of canonical pose angles $\mathbf{w}_0^r \in \mathbb{R}^{M\times6}$ so that $\forall m: \quad g_{0m}^{-1} = \exp(\mathbf{w}_{0m}^r)$.

3.3. Training

We train the MLP (4), mapping the 2D points to the pose parameters, and the part segmentation model (3) by combining a number of losses.

Re-projection loss. The first loss ensures that the posed mesh reprojects correctly onto the 2D points:

$$\mathcal{L}_{\text{rep}} = \frac{\sum_{k=1}^{K} z_k a_k \left\| y_k - \pi\left(X_k R_0 + T_0\right)\right\|}{\sum_{k=1}^{K} z_k a_k}, \tag{5}$$

where X_k and (R_0, T_0) are obtained by composing the pose regressor (4) with the skinning function (2). We weigh the mesh vertices V_k with the areas a_k of the corresponding barycells to make the loss resampling-invariant.

Canonicalisation loss. The authors of C3DPO [43] proposed the *canonicalisation* loss to remove the ambiguity in recovering the camera pose and a 3D reconstruction, which also helps with overfitting. The idea is to learn an auxiliary network $\hat{\mathbf{X}} \approx \Psi(\hat{\mathbf{X}}\tilde{R})$ tasked with undoing a random rotation $\tilde{R}$ applied to the point cloud $\hat{\mathbf{X}}$ (defined in the object coordinates). Novotny *et al*. [43] prove that this loss can be minimised only if the predicted shapes $\hat{\mathbf{X}}$ are indeed canonical w.r.t. orientation, meaning that the model cannot predict two different reconstructions $(\hat{\mathbf{X}}_1, \hat{\mathbf{X}}_2)$ that only differ by a rigid transformation. Specifically, the loss is formulated as

$$\mathcal{L}_{\text{canon}} = \sum_{k=1}^{K} \left\| \left[\hat{\mathbf{X}} - \Psi(\hat{\mathbf{X}}\tilde{R})\right]_k \right\|, \tag{6}$$

where $\tilde{R} \in \mathbb{R}^{3\times3}$ is a random rotation matrix and $[\cdot]_k$ extracts the k-th row of its argument.

ARAP loss. To further increase the robustness of the reconstruction, we encourage the deformation of the template shape to be as-rigid-as-possible (ARAP) [53]. This is particularly useful when, as it is often the case, the input DensePose annotations are noisy and biased. ARAP measures the cost of deforming the template mesh $\mathbf{V}$ into the posed mesh $\mathbf{X}$:

$$\mathcal{L}_{\text{arap}}(\mathbf{X}; \mathbf{V}) = \sum_{k=1}^{K} \min_{R\in SO(3)} \sum_{q\in\mathcal{N}(k)} w_{kq} \left\| V_{\vec{kq}} - X_{\vec{kq}} R \right\|, \tag{7}$$

where $\mathcal{N}(k)$ denotes indices of adjacent template vertices, $V_{\vec{kq}} = V_q - V_k$, $X_{\vec{kq}} = X_q - X_k$, and weights w_{kq} are defined proportionally to the area of the faces incident to the edge kq; see [53] for details. We back-propagate the error through estimated coordinates X_k and X_q but stop gradients after fitting the rotation R.

Entropy regularisation. Sometimes the model tends to assign several part indices to a single vertex, which makes deformations too rigid. We thus regularise the segmentation model by penalising the entropy of the part distribution for each vertex using the following loss:

$$\mathcal{L}_{\text{entropy}} = -\frac{1}{K} \sum_{k=1}^{K} \sum_{m=1}^{M} P_{km} \log P_{km}. \tag{8}$$

Learning formulation. To train the method, we optimise the parameters of the networks Φ, Ψ, and the matrix $\mathbf{W}$ (eq. (3)) minimising a weighed combination of the losses above:

$$\mathcal{L} = \mathcal{L}_{\text{rep}} + w_{\text{entropy}}\mathcal{L}_{\text{entropy}} + w_{\text{canon}}\mathcal{L}_{\text{canon}} + w_{\text{arap}}\mathcal{L}_{\text{arap}}. \tag{9}$$

Loss weights $\mathbf{w}$ are treated as hyper-parameters, see supplementary material for the values used for the experiments.

4. Experiments

We evaluate the quality of our reconstruction on human and animal data, both synthetic and real, and then ablate various components. We compare our results to C3DPO [43] because it is the best-performing Non-Rigid SFM approach that works under assumptions compatible with ours.

Implementation details of the networks and training are provided in the sup. mat. We will share the Pytorch code.

4.1. Datasets and metrics

UP-3D and Stanford Dogs. First, we evaluate the method on two clean, synthetic datasets: UP-3D (humans) and Stanford Dogs. UP-3D [36] contains SMPL fits for 8515 photos of people rendered under 30 random viewpoints. We orthographically project the mesh vertices and input their ground-truth visibility and vertex identity to DP3D directly (instead of using DensePose for UV extraction). For Stanford Dogs, we follow UP-3D and fit a dog model to a subset of ImageNet using SMALify [10] on the mask and 2D keypoint annotations provided in StanfordExtra dataset [8]. We obtain in this way 6511 training and 4673 test instances. Please refer to the sup. mat. for further details.

We report the mean per-joint position error (MPJPE) of the reconstructions. Since we have the "ground-truth" SMPL/SMAL model $\mathbf{X}$ for each test instance, we compute MPJPE of the estimated shape in camera coordinates $\bar{\mathbf{X}} = \hat{\mathbf{X}}R_0 + T_0$ using all keypoints (not only the visible ones): $\text{MPJPE}(\bar{\mathbf{X}}, \mathbf{X}) = \frac{1}{K}\sum_{k=1}^{K} \|\bar{X}_k - X_k\|$. Since via orthographic projection depth is only known up to a constant, we normalise depth by subtracting its mean from $\mathbf{X}$ and $\bar{\mathbf{X}}$ before computing the loss. We use the original train/test splits.

Human 3.6M consists of real images of 7 people equipped with motion-capture sensors performing various tasks in the lab environment. The dataset provides the locations of 3D joints rather than full body surface. Hence, for evaluation purposes, we compute the mean reconstruction error on $N_J = 14$ joints (RE_{14}). In order to obtain the joints' positions $\hat{\mathbf{J}}$ from the posed mesh $\hat{\mathbf{X}}$, we run the pre-trained linear joint regressor from the SMPL model [37]: if the resulting joints are correct, the mesh must have been posed correctly. We rigidly align the sets of points and find the optimal scale before computing the metric: $\text{RE}(\hat{\mathbf{J}}, \mathbf{J}) = \min_{s,R,T} \frac{1}{N_J}\sum_{i=1}^{N_J} \|(s\hat{J}_i R + T) - J_i\|$.

For training, we sampled the videos at 10 frames per second, resulting in 311,424 images. For evaluation, we use the scheme known as 'Protocol #1'. The test set videos are sampled at 25 FPS, resulting in 109,792 images. We ran the pre-trained DensePose detector from Detectron2 [60] on all images independently to obtain the input UV annotation, then converted them to 2D projections of SMPL vertices as described in Section 3.1. We use the standard train/test split, setting out all images of subjects 9 and 11 for testing.

Method	UP-3D	H3.6M	3DPW	Dogs
HMR [25]	—	56.8	81.3	—
GraphCMR [29]	—	50.1	70.2	—
SPIN [28]	—	41.8	59.3	—
Multi-bodies [9]	—	46.1	59.9	—
C3DPO [43]	107.0	216.6	199.9	345.1
no canon. loss (6)	183.6	135.4	120.3	**241.4**
no ARAP loss (7)	242.6	154.8	126.1	371.8
no entropy loss (8)	113.8	119.4	99.1	505.2
no parts model	205.9	125.0	102.3	684.3
DP3D (ours)	**91.2**	**113.6**	**95.2**	247.1

Table 1: **Evaluation of mesh reconstruction** reporting mean per-joint position error (MPJPE) on UP-3D and Dogs datasets, and reconstruction error (RE) on Human 3.6M and 3DPW. The first half of the table shows the results of methods that use 3D supervision. DP3D is then compared to C3DPO [43] applied to dense keypoints and ablated.

3DPW. We evaluate DensePose 3D in a transfer-learning setting, training it on Human 3.6M and evaluating it on 3DPW [58]. DP3D takes keypoints as input, so is invariant to appearance changes and generalises well, as can be seen in Table 1 and fig. 6. We follow the same evaluation protocol as for Human 3.6M, comparing RE on 14 joints.

LVIS. Finally, we fit our model to LVIS dataset [22] containing animal images taken "in the wild". This task is more challenging, since each category comprises only about 2000 training instances, many of which have occluded parts. To get input keypoints and visibilities $(\mathbf{Y}, \mathbf{Z})$, we pre-process the images with CSE [40] in a similar way to DensePose. The output of CSE is noisier than the one of DensePose, so we predict heteroscedastic variance for reprojection loss (5) and maximise the log-likelihood of the Laplace distribution as done by Novotny *et al*. [42]; see sup. mat. for details on pre-processing and the loss. Since there is no 3D ground truth, we provide only qualitative results in Figure 7.

4.2. Comparison to baselines

We compare our method to C3DPO [43], where we use 10-dimensional basis and find the optimal strength of canonicalisation loss in the interval $[0.1, 1]$. The results are in Table 1 and supplementary figures. Note that we train C3DPO on dense keypoints (i.e. 6890 input points for humans), while [43] trains on 17 sparse joints, which makes results from Table 1 incomparable with the ones in [43]. UP-3D and Dogs are less-challenging datasets with clean 2D keypoints and few extreme poses, so C3DPO's simple linear pose model is only slightly inferior to DP3D. In contrast, the gap is large on Human 3.6M and 3DPW: C3DPO outputs the mean pose failing to adapt to the data.

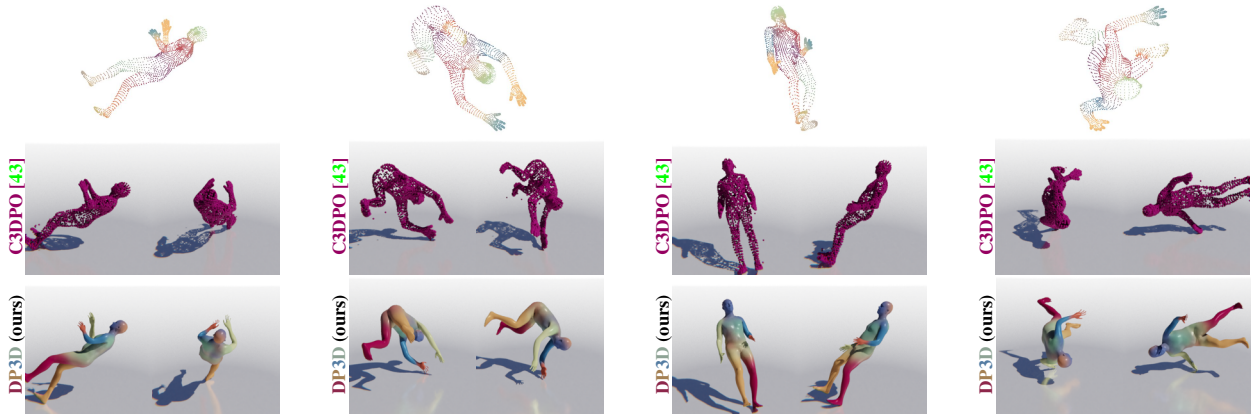

Figure 4: **Qualitative comparison on UP-3D**. The figure shows the input keypoints (colours encode keypoint indices), the reconstruction of C3DPO [43], and of our method (DP3D), where colours correspond to the learnt part segmentation.

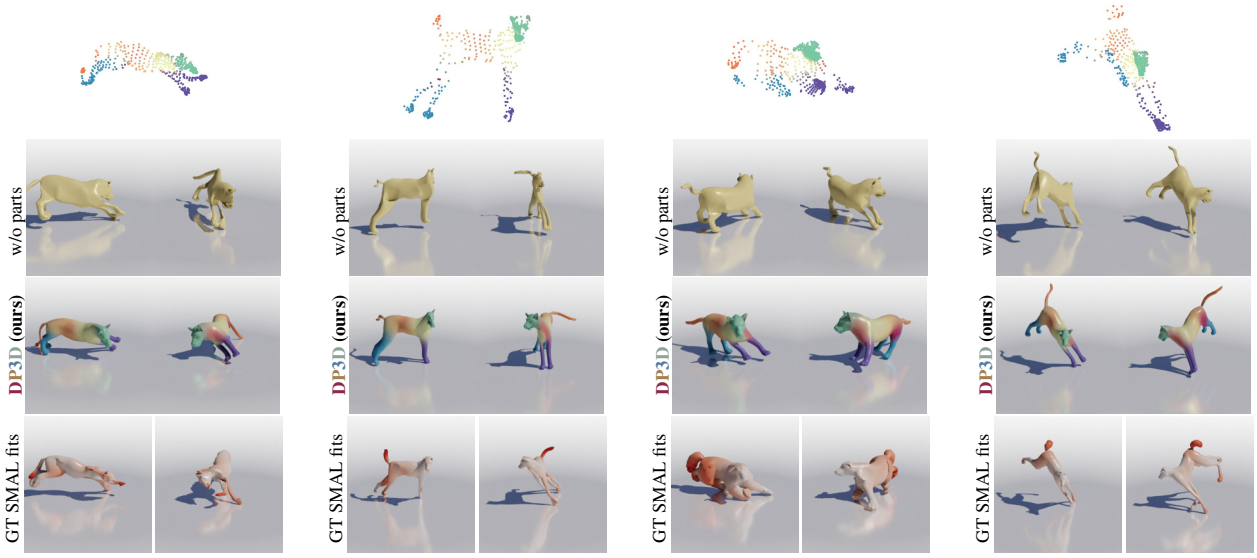

Figure 5: **Results on Stanford Dogs**. The first row shows input keypoints obtained by projecting SMAL fits from the last row, the middle rows show the result of the no-parts baseline and of our reconstruction from the camera's and from an alternative viewpoint (colours correspond to the learnt part segmentation), the last row color-codes errors on the "ground-truth" mesh.

4.3. Ablation study

Removing the loss functions. We report the effect of removing various regularisers in Table 1. Each of them proves important: the canonicalisation loss prevents predicting a degenerate, flat shape; the ARAP loss makes the prediction smooth and helps to learn smooth part segmentations by encouraging local rigidity; the entropy loss makes the part segmentation sharper, allowing the shape to flex more.

Removing the part-based model. Two reasons why C3DPO may work poorly on dense point clouds are: (1) learning a very large linear predictor for thousand of points may lead to overfitting, or (2) the linear model may be unable to capture surface deformations. We test these hypotheses by replacing the articulation model in our method with a C3DPO-like linear basis. To reduce the number of parameters in the basis, we express it as a function of the LBO basis $\mathbf{U}$ (section 3) and define the posed mesh as $\mathbf{X} = (\alpha \otimes I_3)\mathbf{W}^b\mathbf{U}$, where $\mathbf{W}^b \in \mathbb{R}^{D \times N_u}$ are trainable parameters, α is a D-dimensional vecror of shape coefficients, $\otimes$ is Kronecker product, and I_3 is a 3-dimensional identity matrix. We train using eq. (9) but remove the entropy loss (8) (as this model has no parts). We set the number of blendshapes $D = 10$ and find the optimal weight of canonicalisation loss in the $[0.1, 1]$ range.

The penultimate row in Table 1 reveals the correct hypothesis. The model without parts performs significantly better than C3DPO, proving that overfitting explains in large part C3DPO's poor performance. However, the no-parts model still cannot reach the performance of DP3D on

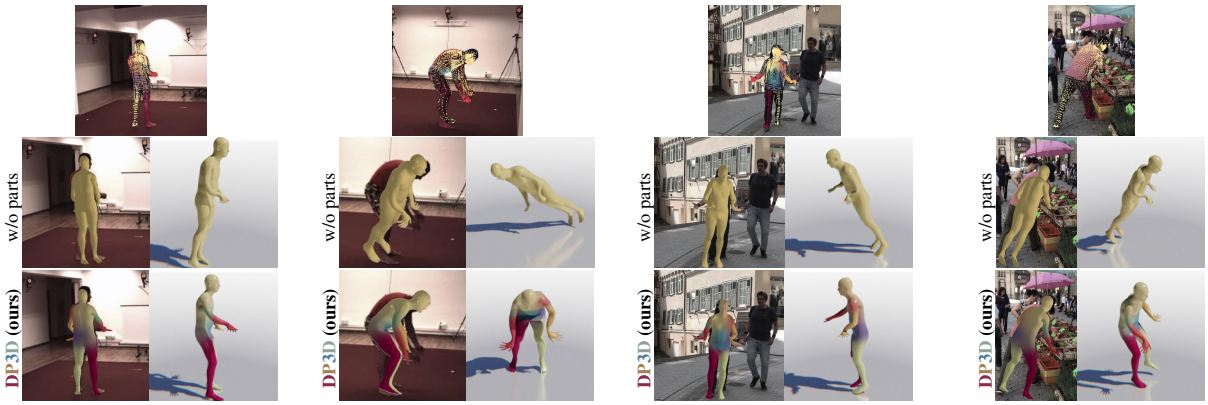

Figure 6: **Qualitative evaluation on Human 3.6M (left two images) and 3DPW (right two images)**. From top to bottom: input image and keypoints, reconstruction with the linear model instead of parts segmentation, and of the proposed method.

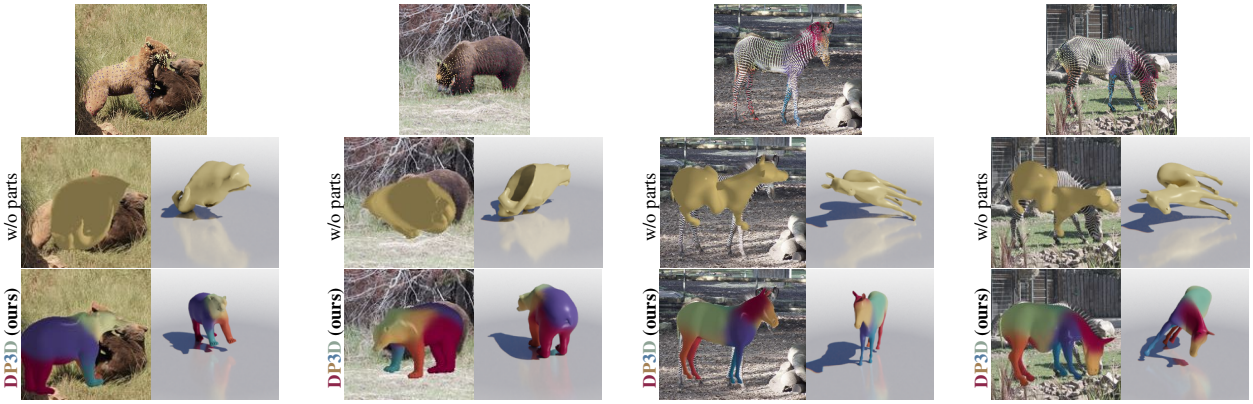

Figure 7: **Qualitative evaluation on LVIS**. From top to bottom: input image and keypoints, the reconstruction with the linear model instead of parts segmentation, and of the proposed method, where colours correspond to the learnt part segmentation.

real-world data (columns 2 and 3), which means that the latter is more efficient than using vanilla linear blendshapes. Remarkably, the no-parts model performs decently on the synthetic datasets, where DensePose annotations were simulated by projecting 3D locations obtained from a parametric model, probably because, despite of high dimensionality, the "rank" of the data is still small. The differences are more pronounced in the visual results in Figures 5 to 7. The linear model in most cases produces symmetrical shapes, which tend to be similar regardless of the input, while DP3D with parts reconstructs the movements of arms more accurately.

Number of latent parts. Figure 8 measures the reconstruction error as a function of the number of latent parts M on human datasets. As expected from human anatomy, the method needs at least 5 parts to model the articulation of the body. The metrics plateau after 10 parts.

Limitations and robustness. DensePose 3D can be only as good as training annotations provided by DensePose or CSE. In supp. mat., we investigate how sensitive the training is to annotation noise, random sparsity (typical for manual annotation), and missing body parts (caused by occlusions).

5. Conclusions

We presented a method that learns 3D deformable shape reconstruction given only a single artist-generated rigid template mesh and dense 2D keypoint annotations, without the need for 3D supervision with the deformable shape model or 3D pose regressor, which are difficult to obtain for most object categories. Because of this, we apply DP3D to the reconstruct animals that lack such 3D annotations.

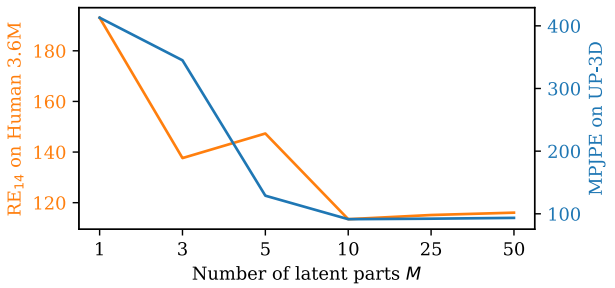

Figure 8: Reconstruction quality w.r.t. the number of parts.

References

[1] Antonio Agudo and Francesc Moreno-Noguer. Dust: Dual union of spatio-temporal subspaces for monocular multiple object 3d reconstruction. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6262–6270, 2017. 3

[2] Antonio Agudo and Francesc Moreno-Noguer. Deformable motion 3D reconstruction by union of regularized subspaces. In *IEEE Int. Conf. Image Process.*, pages 2930–2934. IEEE, 2018. 3

[3] Antonio Agudo, Melcior Pijoan, and Francesc Moreno-Noguer. Image collection pop-up: 3D reconstruction and clustering of rigid and non-rigid categories. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2607–2615, 2018. 3

[4] Ijaz Akhter, Yaser Sheikh, Sohaib Khan, and Takeo Kanade. Nonrigid structure from motion in trajectory space. In *Adv. Neural Inform. Process. Syst.*, 2009. 3

[5] Ijaz Akhter, Yaser Sheikh, Sohaib Khan, and Takeo Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(7):1442–1456, 2011. 3

[6] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. SCAPE: shape completion and animation of people. *ACM Trans. on Graphics (TOG)*, 2005. 2, 4

[7] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: shape completion and animation of people. In *ACM Trans. on Graphics*, 2005. 3

[8] 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. In *ECCV*, 2020. 6

[9] Benjamin Biggs, David Novotny, Sebastien Ehrhardt, Hanbyul Joo, Ben Graham, and Andrea Vedaldi. 3d multi-bodies: Fitting sets of plausible 3d human models to ambiguous image data. *Advances in Neural Information Processing Systems*, 33, 2020. 3, 6

[10] Benjamin Biggs, Thomas Roddick, Andrew W. Fitzgibbon, and Roberto Cipolla. Creatures great and SMAL: recovering the shape and motion of animals from video. In *Proc. ECCV*, 2018. 6

[11] Jose-Luis Blanco. A tutorial on SE(3) transformation parameterizations and on-manifold optimization. Technical report, 2010. 5

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

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

[14] Ching-Hang Chen, Ambrish Tyagi, Amit Agrawal, Dylan Drover, Stefan Stojanov, and James M Rehg. Unsupervised 3d pose estimation with geometric self-supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2019. 3

[15] Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. In *Advances in Neural Information Processing Systems*, pages 9609–9619, 2019. 2

[16] Yuchao Dai, Hongdong Li, and Mingyi He. A simple prior-free method for non-rigid structure-from-motion factorization. *International Journal of Computer Vision*, 107(2):101–122, 2014. 3

[17] Dylan Drover, Rohith MV, Ching-Hang Chen, Amit Agrawal, Ambrish Tyagi, and Cong Phuoc Huynh. Can 3D pose be learned from 2D projections alone? In *Eur. Conf. Comput. Vis.*, 2018. 3

[18] Katerina Fragkiadaki, Marta Salas, Pablo Arbelaez, and Jitendra Malik. Grouping-based low-rank trajectory completion and 3D reconstruction. In *Adv. Neural Inform. Process. Syst.*, pages 55–63, 2014. 3

[19] Paulo FU Gotardo and Aleix M Martinez. Non-rigid structure from motion with complementary rank-3 spaces. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3065–3072. IEEE, 2011. 3

[20] P. Guan, A. Weiss, A. O. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *Int. Conf. Comput. Vis.*, 2009. 3

[21] Riza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. DensePose: Dense human pose estimation in the wild. In *Proc. CVPR*, 2018. 2, 3

[22] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 6

[23] Yinghao Huang. Towards accurate marker-less human shape and pose estimation over time. In *Proc. 3DV*, 2017. 3

[24] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3D deformation model for tracking faces, hands, and bodies. In *Proc. CVPR*, 2018. 3

[25] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 3, 6

[26] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *European Conference on Computer Vision*, pages 386–402, 2018. 1, 2

[27] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Self-supervised learning of 3d human pose using multi-view geometry. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 3

[28] 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 *ICCV*, 2019. 3, 6

[29] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *Proc. CVPR*, 2019. 1, 3, 6

[30] Chen Kong and Simon Lucey. Deep non-rigid structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1558–1567, 2019. 2, 3

[31] Yasunori Kudo, Keisuke Ogaki, Yusuke Matsui, and Yuri Odagiri. Unsupervised adversarial learning of 3D human pose from 2D joint locations. *Eur. Conf. Comput. Vis.*, 2018. 3

[32] Nilesh Kulkarni, Abhinav Gupta, David F. Fouhey, and Shubham Tulsiani. Articulation-aware Canonical Surface Mapping. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

3

[33] Nilesh Kulkarni, Abhinav Gupta, and Shubham Tulsiani. Canonical surface mapping via geometric cycle consistency. In *Int. Conf. Comput. Vis.*, 2019. 3

[34] Suryansh Kumar, Anoop Cherian, Yuchao Dai, and Hongdong Li. Scalable dense non-rigid structure from motion: A grassmannian perspective. In *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2018. 3

[35] Suryansh Kumar, Yuchao Dai, and Hongdong Li. Spatial-temporal union of subspaces for multi-body non-rigid structure-from-motion. *Pattern Recognition Journal*, 2017. 3

[36] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J. Black, and Peter V. Gehler. Unite the people: Closing the loop between 3D and 2D human representations. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4704–4713, 2017. 3, 6

[37] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: a skinned multi-person linear model. *ACM Trans. on Graphics (TOG)*, 2015. 1, 2, 4, 6

[38] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. A simple yet effective baseline for 3d human pose estimation. In *Proc. ICCV*, 2017. 3

[39] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. VNect: Real-time 3d human pose estimation with a single RGB camera. In *Proc. SIGGRAPH*, 2017. 3

[40] Natalia Neverova, David Novotny, Marc Szafraniec, Vasil Khalidov, Patrick Labatut, and Andrea Vedaldi. Continuous surface embeddings. *Advances in Neural Information Processing Systems*, 33, 2020. 2, 3, 6

[41] Natalia Neverova, James Thewlis, Rıza Alp Güler, Iasonas Kokkinos, and Andrea Vedaldi. Slim DensePose: Thrifty Learning from Sparse Annotations and Motion Cues. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 3

[42] David Novotný, Diane Larlus, and Andrea Vedaldi. Capturing the geometry of object categories from video supervision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 6

[43] David Novotný, Nikhila Ravi, Benjamin Graham, Natalia Neverova, and Andrea Vedaldi. C3DPO: Canonical 3d pose networks for non-rigid structure from motion. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2019. 2, 3, 5, 6, 7

[44] 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 *Proc. CVPR*, 2019. 3

[45] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3d human pose estimation. In *Int. Conf. Comput. Vis.*, 2018. 3

[46] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Harvesting multiple views for marker-less 3d human pose annotations. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 3

[47] Helge Rhodin, Jörg Spörri, Isinsu Katircioglu, Victor Constantin, Frédéric Meyer, Erich Müller, Mathieu Salzmann, and Pascal Fua. Learning monocular 3d human pose estimation from multi-view images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 3

[48] Grégory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. LCR-Net++: Multi-person 2D and 3D pose detection in natural images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018. 3

[49] Raif M Rustamov. Laplace-beltrami eigenfunctions for deformation invariant shape representation. In *Proceedings of the fifth Eurographics symposium on Geometry processing*, pages 225–233, 2007. 4

[50] Artsiom Sanakoyeu, Vasil Khalidov, Maureen S McCarthy, Andrea Vedaldi, and Natalia Neverova. Transferring dense pose to proximal animal classes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5233–5242, 2020. 2, 3

[51] Tanner Schmidt, Richard A. Newcombe, and Dieter Fox. Self-supervised visual descriptor learning for dense correspondence. *IEEE Robotics and Automation Letters*, 2(2), 2017. 3

[52] Leonid Sigal, Alexandru O. Balan, and Michael J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Proc. NeurIPS*, 2008. 3

[53] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface morphing. *Eurographics Symposium on Geometry Processing (2007)*, 26(3):548–557, 2007. 5

[54] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proc. ECCV*, 2018. 3

[55] J. Thewlis, S. Albanie, H. Bilen, and A. Vedaldi. Unsupervised learning of landmarks via vector exchange. In *Int. Conf. Comput. Vis.*, 2019. 3

[56] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object frames by dense equivariant image labelling. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 3

[57] Lorenzo Torresani, Aaron Hertzmann, and Chris Bregler. Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(5):878–892, 2008. 3

[58] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3D human pose in the wild using IMUs and a moving camera. In *Proc. ECCV*, 2018. 6

[59] Chaoyang Wang, Chen-Hsuan Lin, and Simon Lucey. Deep nrsfm++: Towards 3d reconstruction in the wild. *arXiv preprint arXiv:2001.10090*, 2020. 3

[60] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 6

[61] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proc. CVPR*, 2019. 3

[62] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020. 1

[63] Andrei Zanfir, Eduard Gabriel Bazavan, Mihai Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Neural descent for visual 3d human pose and shape. *arXiv preprint arXiv:2008.06910*, 2020. 1

[64] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. Monocular 3D pose and shape estimation of multiple people in natural scenes - the importance of multiple scene constraints. In *Proc. CVPR*, 2018. 3

[65] Xiaowei Zhou, Menglong Zhu, Kosta Derpanis, and Kostas Daniilidis. Sparseness meets deepness: 3d human pose estimation from monocular video. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 3

[66] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, and Kostas Daniilidis. Sparse representation for 3D shape estimation: A convex relaxation approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016. 3

[67] Yingying Zhu, Dong Huang, Fernando De La Torre, and Simon Lucey. Complex non-rigid motion 3D reconstruction by union of subspaces. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1542–1549, 2014. 3