Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Garvita Tiwari[1,2], Dimitrije Antić[1], Jan Eric Lenssen[2], Nikolaos Sarafianos[3], Tony Tung[3], and Gerard Pons-Moll[1,2]

[1] University of Tübingen, Germany
{garvita.tiwari, dimirije.antic, gerard.pons-moll}@uni-tuebingen.de
[2] Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
jlenssen@mpi-inf.mpg.de
[3] Meta Reality Labs Research, Sausalito, USA
{nsarafianos, tony.tung}@fb.com

Abstract. We present Pose-NDF, a continuous model for plausible human poses based on neural distance fields (NDFs). Pose or motion priors are important for generating realistic new poses and for reconstructing accurate poses from noisy or partial observations. Pose-NDF learns a manifold of plausible poses as the zero level set of a neural implicit function, extending the idea of modeling implicit surfaces in 3D to the high-dimensional domain $SO(3)^K$, where a human pose is defined by a single data point, represented by K quaternions. The resulting high-dimensional implicit function can be differentiated with respect to the input poses and thus can be used to project arbitrary poses onto the manifold by using gradient descent on the set of 3-dimensional hyperspheres. In contrast to previous VAE-based human pose priors, which transform the pose space into a Gaussian distribution, we model the actual pose manifold, preserving the distances between poses. We demonstrate that Pose-NDF outperforms existing state-of-the-art methods as a prior in various downstream tasks, ranging from denoising real-world human mocap data, pose recovery from occluded data to 3D pose reconstruction from images. Furthermore, we show that it can be used to generate more diverse poses by random sampling and projection than VAE-based methods. We will release our code and pre-trained model for further research at https://virtualhumans.mpi-inf.mpg.de/posendf/.

1 Introduction

Realistic and accurate human motion capture and generation is essential for understanding human behavior and human interaction in the scene [23,41,68,67,9]. Human motion capturing systems, like marker-based systems [37,34], IMU-based methods [23,41], or reconstruction from RGB/RGB-D data [55,22,29,70], often suffer from artifacts like skating, self-intersections and jitters and produce non-realistic human poses, especially in the presence of noisy data and occlusion. To make the results applicable in fields like 3D scene understanding, human motion generation, or AR/VR applications, it is often required to apply exhaustive manual or automatic cleaning procedures.

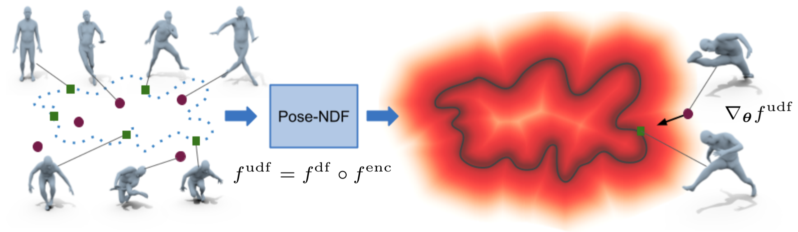

Pose-NDF
$f^{\text{udf}} = f^{\text{df}} \circ f^{\text{enc}}$
$\nabla_{\boldsymbol{\theta}} f^{\text{udf}}$

Fig. 1. We present Pose-NDF, a neural unsigned distance field in $SO(3)^K$, which learns the manifold of plausible poses as zero level set. We learn the distance field representation from samples of plausible (🟩) and unrealistic (●) poses (**left**). We encode the input pose (given as a set of quaternions) using a structural MLP f^{enc} and predict the distance from the joint representation using an MLP f^{df}. The gradient $\nabla_{\boldsymbol{\theta}} f^{\text{udf}}$ and distance value $f^{\text{udf}}(\boldsymbol{\theta})$ are used to project implausible poses onto the manifold (**right**).

In recent years, learned data priors to post-process such non-realistic human poses has become increasingly popular. Prior human pose models mainly focus on learning a joint distribution of individual joints in pose space [10] or recently in a latent space, using VAEs [49,52,66]. They have demonstrated to greatly improve the plausibility of poses after model fitting. However, VAE-based methods, such as VPoser [49] or HuMoR [52] make a Gaussian assumption on the space of possible poses, which leads to several limitations: **1)** They have the *tendency of producing more likely poses* that lie near the mean of the computed Gaussian. Those poses however, might not be the correct ones. **2)** Distances between individual human poses are *not preserved* in the VAE latent space. Hence, taking small steps towards the Gaussian mean might result in large steps in pose space. **3)** VAEs have been shown to *fold* a manifold into a Gaussian distribution [39], exposing *dead regions* without any data points in the outer parts of the distribution. Thus, they produce non-plausible samples that are far from the input when traversed in outer regions, as we demonstrate in our experiments.

To alleviate these issues, we present Pose-NDF, a human pose prior that models the full manifold of plausible poses in high-dimensional pose space. We represent the manifold as a surface, where plausible poses lie on the manifold, hence having a zero distance, and non-plausible poses lie outside of it, having a non-zero distance from the surface. We propose to learn this manifold using a high-dimensional neural field, analogously to representing 3D shapes using neural distance fields [48,13]. This formulation preserves distances between poses and allows to traverse the pose space along the negative gradient of the distance function, which points to the direction of maximum distance decrease. Using gradient descent in pose space from an initial potentially non-plausible pose, we always find the closest point on the manifold of plausible poses.

An overview of our method is given in Fig. 1. We formulate the problem of learning the pose manifold as a surface completion task in n-dimensional space. In order to learn a pose manifold, there are two key challenges: **a)** the input space is high-dimensional, and **b)** the input space is not Euclidean, as it is for 3-dimensional implicit surfaces [48,13]. Instead, the pose space is given as $SO(3)^K$, in which a single pose can be represented by K elements of the rotation group $SO(3)$, describing the orientations of joints in a human body model. To represent group elements, we opted for a quaternion representation, as they are continuous, have an easy-to-compute distance, and are subject to an efficient gradient descent algorithm. We map a given pose to a distance by applying a hierarchical implicit neural function, which encodes the pose based on the kinematic structure of the human body. We train our model using the AMASS dataset [38], where each sample from the dataset is treated as a point on the manifold. The learned neural field representation can be used to project any pose onto the manifold, similar to [13]. We leverage this property and use Pose-NDF for diverse pose generation, pose interpolation, as a pose prior for 3D pose estimation from images [49,10], and motion denoising [23,52], improving on state-of-the-art methods in all areas. In summary our contributions are:

- A novel high-dimensional neural field representation in $SO(3)^K$, Pose-NDF, which represents the manifold of plausible human poses.
- Pose-NDF improves the state of the art in human body fitting from images by acting as a pose prior. It outperforms other human pose priors, such as VPoser [49] and the human motion prior HuMoR [52] on motion denoising.
- Our method is as fast or faster than current state-of-the-art methods, is fully differentiable and the distance from the manifold can be leveraged for finding the optimal step size during optimization.
- Pose-NDF generates more diverse samples than previous methods with Gaussian assumptions, which are biased towards generating more likely poses.

2 Related Work

Our method is a *human pose prior* build as *neural field* in high-dimensional space. Thus, we review related work in both of these areas.

Pose and Motion Priors. Human pose and motion priors are crucial for preserving the realism of models estimated from captured data [40,23,38] and to estimate human pose from images [10,49,65,14,33] and videos [32,59]. Further, they can be powerful tools for data generation. Initial work along this direction mainly focused on learning constraints for joint limits in Euler angles [17] or swing and twist representations [6,56,1], to avoid twists and bends beyond certain limits. A next iteration of methods fits a Gaussian Mixture Model (GMM) to a pose dataset and uses the GMM-based prior for downstream tasks like image-based 3D pose estimation [10,54] or registration of 3D scans [4,8,60]. Additionally, simple statistical models, such as PCA, have been proposed [47,62,57]. With the rise of deep learning and GANs [20], adversarial training has been used to bring predicted poses close to real poses [31,19] and for motion prediction [7]. However these are task specific models, HMR [31] models $p(\boldsymbol{\theta}|I)$ and requires an image I. HP-GAN [7] models $p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1})$ and requires pose parameters $\boldsymbol{\theta}_{t-1}$ for previous frame/time. Therefore they cannot be used as a prior for other tasks.

More recent work uses VAEs to learn pose priors [49], which can be used for generating pose samples, as prior in pose estimation, or 3d human reconstruction from images or sparse/occluded data. Some works [52,66,50] propose VAE-based human motion models. HuMoR [52] proposes to learn a distribution of possible pose transitions in motion sequences using a conditional VAE. ACTOR [50] learns an action conditioned VAE-Transformer prior. Further work designs pose representations along the hierarchy of human skeletons [2] and uses it for character animation [5]. Concurrent work [16] learns a human pose prior using GANs and highlights the shortcomings of Gaussian assumption based models like VPoser [49]. A VPoser decoder is used as generator (mapping $z \rightarrow \theta$) and an HMR [31]-like discriminator is used to train the model. As described in Sec. 1, our approach follows a different paradigm than the VAE and GAN-based methods, as we directly model the manifold of plausible poses in high-dimensional space, which leads to a distance-preserving representation.

Before the rise of deep learning, modeling partial pose spaces as implicit functions was common, e.g. as fields on a single shoulder joint quaternion [26] or an elbow joint quaternion, conditioned on the shoulder joint [25]. However, those ignore the real part of the quaternion, leading to ambiguities in representation, are not differentiable, and are limited to 2 joints in the human body model. In contrast, our method uses a fully differentiable neural network, which learns an implicit surface in higher dimension, taking all human joints and all four components of each quaternion into account.

Neural Fields. Neural fields [13,48,42] for surface modeling have received increasing interest over the recent years. They have been used to model fields in 2D or 3D, representing images or partial differentiable equations [58,21], signed or unsigned distances from static 3D shapes [13,48,27,24], pose-conditioned distance field [53,61,43], radiance fields [44,51] and more recently for human-object [63,9] and hand-object [69] interactions. For a more detailed overview of neural fields please refer to [64]. Neural fields have recently been brought to higher dimensions to model surfaces in Euclidean spaces [45]. In this work, we apply the concept to the high-dimensional, non-Euclidean space of $SO(3)^K$, modeling the unsigned distance to manifolds of plausible human body poses in pose space.

3 Method

In this section, we describe our method Pose-NDF, a model for manifolds of plausible human poses based on high-dimensional neural fields. We assume that the realistic and plausible human poses lie on a manifold embedded in pose space $SO(3)^K$, with K being the number of joints in the human body. Given a neural network $f : SO(3)^K \mapsto \mathbb{R}^+$, which maps a pose, $\boldsymbol{\theta} \in SO(3)^K$ to a non-negative scalar, we represent the manifold of plausible poses as the zero level set:

$$\mathcal{S} = \{\boldsymbol{\theta} \in SO(3)^K \mid f(\boldsymbol{\theta}) = 0\}, \tag{1}$$

such that the value of f represents the unsigned distance to the manifold, similar to neural fields-based 3D shape learning [12,13,21,48]. Without loss of generality, we use the SMPL body model [36,49], resulting in poses $\boldsymbol{\theta}$ with $K = 21$ joints.

3.1 Quaternions as Representation of SO(3)

A human pose is represented by 3D rotations of individual joints in the human skeleton. The 3-dimensional rotation group $SO(3)$ has several common vector space representations that are used to describe group elements in practice. Frequently used examples are rotation matrices, axis-angle representations or unit quaternions [28]. Pose-NDF requires the representation to have specific properties: a) we aim to model a *manifold*, continuously embedded in pose space. Thus, the chosen representation should be continuous in parameter space, which prohibits axis-angle representations; b) the representation should enable efficient computation of the geodesic distance between two elements; c) our algorithm requires *efficient gradient descent* in pose space. As described in Sec. 3.4, quaternions are subject to such a gradient descent algorithm that makes use of the efficient reprojection to $SO(3)$ by vector normalization. In contrast, rotation matrices would require more expensive orthogonalization. Therefore, we chose unit quaternions as the best-suited $SO(3)$ representation of joints, as they fulfill all three properties. We will use $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K\}$ to denote the quaternions for all K joints of a pose. Each quaternion represents the rotation of a joint with respect to its parent node. Since quaternions lie on S^3 (embedded in 4-dimensional space) the full pose $\boldsymbol{\theta}$ can be easily used as input for a neural network $f : \mathbb{R}^{4K} \to \mathbb{R}^+$. We define the distance $d : (S^3)^K \times (S^3)^K \to \mathbb{R}^+$ between two poses $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K\}$ and $\hat{\boldsymbol{\theta}} = \{\hat{\boldsymbol{\theta}}_1, ..., \hat{\boldsymbol{\theta}}_K\}$ as:

$$d(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = \sqrt{\sum_{i=1}^{K} \frac{w_i}{2} (\arccos |\boldsymbol{\theta}_i^\top \cdot \hat{\boldsymbol{\theta}}_i|)^2}, \tag{2}$$

where the individual elements of summation are a metric on $SO(3)$ [28] and w_i is the weight associated with each joint based on their position in the kinematic structure of the SMPL body model (i.e. early joints in the chain have higher weights). It should be noted that the double cover property of unit quaternions, that is, the quaternions $\mathbf{q}$ and $-\mathbf{q}$ represent the same $SO(3)$ element, does not lead to additional challenges. We simply train the network to be point symmetric by applying sign flip augmentation on input quaternions.

3.2 Hierarchical Implicit Neural Function

We represent the human pose with quaternions in local coordinate frames of the parent joint, using the kinematic structure of the SMPL body model. We treat the joints in local coordinate frame, so that continuous manipulation of a single joint corresponds to realistic motion. However, this might result in unrealistic combination of rotation of joints. The plausibility of individual joints depends on the ancestor rotations and thus needs to be conditioned on them. In order to incorporate this dependency, we use a hierarchical network f^{enc}, which encodes the human pose based on the model structure [2,19,43], before predicting the distance based on the joint representation.

Formally, for a given pose $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K\}$, where $\boldsymbol{\theta}_k$ is the pose for joint k, and a function $\tau(k)$, mapping the index of each joint to its parent joints index, we encode each pose using an MLP as:

$$f_1^{\text{enc}} : (\boldsymbol{\theta}_1) \mapsto \mathbf{v}_1 \qquad f_k^{\text{enc}} : (\boldsymbol{\theta}_k, \mathbf{v}_{\tau(k)}) \mapsto \mathbf{v}_k, \quad k \in \{2 \dots K\} \tag{3}$$

which takes the quaternion pose and encoded feature $\mathbf{v}_{\tau(k)} \in \mathbb{R}^l$ of its parent joint as input and generates $\mathbf{v}_k \in \mathbb{R}^l$, where l is the dimension of feature. We then concatenate the encoded feature for every joint to get a combined pose embedding $\mathbf{p} = [\mathbf{v}_1 || \dots || \mathbf{v}_K]$. This embedding is processed by an MLP $f^{\text{df}} : \mathbb{R}^{l \cdot K} \to \mathbb{R}^+$, which predicts the unsigned distance for the given pose representation $\mathbf{p}$. Collectively the complete model $f^{\text{udf}}(\boldsymbol{\theta}) = (f^{\text{df}} \circ f^{\text{enc}})(\boldsymbol{\theta})$, is termed as Pose-NDF, where $f^{\text{udf}} : SO(3)^K \mapsto \mathbb{R}^+$.

3.3 Loss functions

We train the hierarchically structured neural field f^{udf} to predict the geodesic distance to the plausible pose manifold for a given pose. The training data is given as a set $\mathcal{D} = \{(\boldsymbol{\theta}_i, d_i)\}_{1 \le i \le N}$, containing pairs of poses $\boldsymbol{\theta}_i$ and distances d_i (Eq. 2). We train the network with the standard distance loss $\mathcal{L}_{\text{UDF}}$, and an Eikonal regularizer $\mathcal{L}_{\text{eikonal}}$, which encourages a unit-norm gradient for the distance field outside of the manifold [15,21]:

$$\mathcal{L}_{\text{UDF}} = \sum_{(\boldsymbol{\theta}, d) \in \mathcal{D}} ||f^{\text{udf}}(\boldsymbol{\theta}) - d_{\boldsymbol{\theta}}||_2 \quad \mathcal{L}_{\text{eikonal}} = \sum_{(\boldsymbol{\theta}, d) \in \mathcal{D},\ d \neq 0} (||\nabla_{\boldsymbol{\theta}} f^{\text{udf}}(\boldsymbol{\theta})|| - 1)^2, \tag{4}$$

More details about training data, network architecture is provided in the supplementary material.

3.4 Projection Algorithm

Given a trained model f^{udf}, it can be applied to project an arbitrary pose $\boldsymbol{\theta}$ to the manifold of plausible poses. We use the predicted distance $f^{\text{udf}}(\boldsymbol{\theta})$ and gradient information $\nabla_{\boldsymbol{\theta}} f^{\text{udf}}(\boldsymbol{\theta})$ to project a query pose to the manifold surface $\mathcal{S}$, as was previously done in unsigned distances functions for 3D shapes [13]. In our case, given $SO(3)$ poses, this amounts to finding:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in SO(3)^K}{\arg\min}\ d(\boldsymbol{\theta}, \mathcal{S}), \tag{5}$$

where $d(\boldsymbol{\theta}, \mathcal{S})$ is the distance (Eq. 2) of $\boldsymbol{\theta}$ to the closest point in $\mathcal{S}$. We find $\hat{\boldsymbol{\theta}}$ by applying gradient descent on the 3-sphere, using gradient information $\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta})$ and distances $f(\boldsymbol{\theta})$, obtained from the implicit neural function f. One step is given as:

$$\boldsymbol{\theta}^i = \boldsymbol{\theta}^{i-1} - \alpha f(\boldsymbol{\theta}^{i-1}) \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}^{i-1}), \tag{6}$$

followed by a re-projection to the sphere (i.e. vector normalization) after several iterations. This algorithm is guaranteed to converge to local minima on the sphere, which in our case, assuming a correctly learned distance function, is the nearest point on the pose manifold.

4 Experiments and Results

In this section we evaluate Pose-NDF and show the different use cases of our pose model, which include the ability to serve as a *prior in denoising motion sequences or recovery* from partial observations (Sec. 4.2), *prior for recovering plausible poses* from images (Sec. 4.3) using an optimization-based method, *pose generation* (Sec. 4.4) and *pose interpolation* (Sec. 4.5). We demonstrate that the Pose-NDF method *outperforms the state-of-the-art VAE-based human pose prior methods.* We also show the *advantages* of our distance field formulation over VAEs or Gaussian assumption models (Sec. 4.6). Before turning to the results, we explain training and implementation details of Pose-NDF in Sec. 4.1.

4.1 Experimental Setup

We use the AMASS dataset [38] for training. As mentioned in Sec. 3.3, we train the network with supervision on predicted distance values, and hence we create a dataset of pose and distance pairs $(\boldsymbol{\theta}, d_{\boldsymbol{\theta}})$. Since the training samples from AMASS lie on the desired manifold, $d_{\boldsymbol{\theta}} = 0$ is assigned to all poses in the dataset. We then randomly generate negative samples with distance $d_{\boldsymbol{\theta}} > 0$ by adding noise to AMASS poses. Please find details of data preparation in supplementary. We train our model in a **multi-stage** regime by varying the type of training samples used. We start our training with manifold poses $\boldsymbol{\theta}_m$ and non-manifold poses $\boldsymbol{\theta}_{nm}$ with a large distance to the desired manifold. Then we increase the number of non-manifold poses $\boldsymbol{\theta}_{nm}$ with a small distance in each training batch. This training scheme helps to initially learn a smooth surface and to iteratively introduce the fine details over the course of training. Our **network architecture** consists of one 2-layer MLP f^{enc} with an output feature size of $l = 6$ for each joint, similar to [43]. Thus, the pose encoding network generates a feature vector $\mathbf{p} \in \mathbb{R}^{126}$. We implement the distance field network f^{df} as a 5-layer MLP. For training, we use the softplus activation in the hidden layer and train the network end-to-end using the loss functions described in Eq. (4).

4.2 Denoising Mocap Data

Human motion capture has been done using diverse setups ranging from RGB, RGB-D to IMU based capture systems. The data captured from these sources often produce artifacts like jitters, unnaturally rigid joints or weird bends at some joints, or positions with only partial observations. Prior work [52] improves the quality of captured motion sequences by using an optimization-based method, with the goal of recovering the captured data and preserving the realism of human poses. A robust and expressive human pose prior is key to preserve the realism of optimized poses, along with preserving the original data. Following HuMoR [52], we demonstrate the effectiveness of our pose manifold for: 1) motion denoising and 2) fitting to partial data.

Data	HPS [23]			AMASS [38]			Noisy AMASS		
# frames	60	120	240	60	120	240	60	120	240
Method									
VPoser [49]	4.91	4.16	3.81	1.52	1.55	1.47	8.96	9.13	9.15
HuMoR [52]	9.69	8.73	10.86	3.21	3.62	3.67	11.04	17.14	30.31
Pose-NDF	**2.32**	**2.14**	**2.11**	**0.59**	**0.55**	**0.54**	**7.96**	**8.31**	**8.46**

Table 1. Motion denoising: We compare the per-vertex error (in *cm*) on mocap data from HPS (**left**) and AMASS (**middle**) and on artificially created noisy AMASS data (**right**). In all cases, Pose-NDF based motion denoising results in the least error. We also observe that in case of mocap data (HPS, AMASS), motion denoising using Pose-NDF results in very small error (small change from input), which is the desired behavior as these mocap poses are already realistic and hence close to our learned manifold. On the other hand, HuMoR changes the input pose significantly.

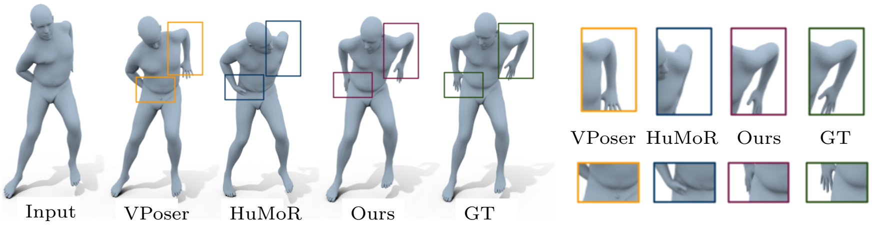


Fig. 2. Motion denoising: We observe that Pose-NDF based motion denoising makes the pose realistic and solves small intersection issues, while VPoser and HuMoR still result in unrealistic poses.

We follow the same experimental setup as [52], but only deal with human poses and thus, remove the terms corresponding to human-scene contact and translation of root joint. In total, we find the pose parameters $\hat{\theta}^t$ at frame t as:

$$\hat{\boldsymbol{\theta}}^t = \arg\min_{\boldsymbol{\theta}} \lambda_{\mathrm{v}} \mathcal{L}_{\mathrm{v}} + \lambda_{\boldsymbol{\theta}} \mathcal{L}_{\boldsymbol{\theta}} + \lambda_t \mathcal{L}_t, \tag{7}$$

where $\mathcal{L}_{\mathrm{v}}$ makes sure that the optimized pose is close to the observation and the temporal smoothness term $\mathcal{L}_t$ enforces temporal consistency:

$$\mathcal{L}_{\mathrm{v}} = ||\mathcal{J}(\boldsymbol{\beta}_0, \hat{\boldsymbol{\theta}}^t) - \mathcal{J}_{\mathrm{obs}}||_2^2 \qquad \mathcal{L}_t = ||M(\boldsymbol{\beta}_0, \hat{\boldsymbol{\theta}}^{\mathrm{t}}) - M(\boldsymbol{\beta}_0, \boldsymbol{\theta}^{\mathrm{t}-1})||_2^2. \tag{8}$$

Here, $\mathcal{J}(\boldsymbol{\beta}, \boldsymbol{\theta})$ represent vertices (mocap markers) and $M(\boldsymbol{\beta}, \boldsymbol{\theta})$ represents SMPL mesh vertices for a given pose ($\boldsymbol{\theta}$) and shape ($\boldsymbol{\beta}$) parameters of SMPL [36]. Finally, we use Pose-NDF as a pose prior term in the optimization by minimizing the distance of the current pose from our learned manifold, $\mathcal{L}_{\boldsymbol{\theta}} = f^{\mathrm{udf}}(\boldsymbol{\theta})$. We leverage the distance $f^{\mathrm{udf}}(\boldsymbol{\theta})$ to get the optimal step size during optimization.

We evaluate on two different settings: 1) clean mocap datasets and 2) a noisy mocap dataset. For clean mocap datasets, we use HPS [23] and the test split of AMASS [38,49]. For the noisy mocap dataset, we create random noisy sequences by adding Gaussian noise to AMASS test sequences and call it "Noisy AMASS".

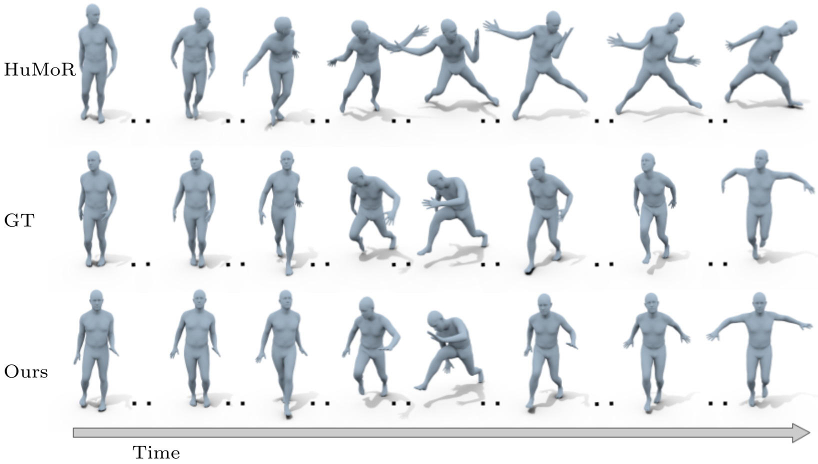


Fig. 3. Motion denoising: We compare the results on motion denoising using Pose-NDF and HuMoR [52] as priors with GT data and visualize every 10^{th} frame of a sequence. We observe that for HuMoR (**top**) the correction in input pose accumulates over time and makes the output pose significantly different from the GT (**middle**). Pose-NDF remains close to observations while correcting unrealistic poses (**bottom**).

The average noise introduced in "Noisy AMASS" is 9.3 *cm*. We use a list of SMPL mesh vertices $\mathcal{J}$ as observation during optimization. We created the data with a fixed shape and do not optimize for shape parameters $\boldsymbol{\beta}$. Instead of adding noise to joint locations, we add noise directly to the rotation of each joint. This is done for all methods to ensure a fair comparison. For HuMoR [52], we use the *TestOpt* optimization from the original work. VPoser does not have motion experiments, which is why we combine the latent space optimization from the original work with our optimization given in Eq. (7) to ensure that we compare against the best possible result. Specifically, we first encode the rotation matrix representation of noisy input pose $\boldsymbol{\theta}^t$ using the VPoser encoder as $\boldsymbol{z}^t = f_{\text{v\_enc}}(\boldsymbol{\theta}^t)$, then add random noise $(\hat{\epsilon})$ in the latent space and reconstruct the pose by $\tilde{\boldsymbol{\theta}}^{\text{t}} = f_{\text{v\_dec}}(\boldsymbol{z}^t + \hat{\epsilon})$. Following [66], we observe that the temporal term in latent space yields better results than the temporal term in input pose/vertices, which we used in the VPoser experiment. The prior and temporal term for VPoser-based denoising are given as:

$$\mathcal{L}_{\boldsymbol{\theta}}^{\text{VPoser}} = ||\hat{\epsilon}||_2 \qquad \mathcal{L}_t^{\text{VPoser}} = ||\boldsymbol{z}^{t-1} - \hat{\boldsymbol{z}}^t||_2. \tag{9}$$

Results. We compare motion denoising between HuMoR [52] (*TestOpt*), Eq. (7) with VPoser prior [49], and Eq. (7) with Pose-NDF prior in Table 1. Pose-NDF achieves the lowest error in all settings. For mocap datasets like AMASS and HPS the motion is realistic, but can have small artifacts and jitter. Thus, an ideal motion/pose prior should not change the overall pose of these examples, but only fix these local artifacts. We observe that, numerically, VPoser and Pose-NDF-based optimization do not change the input pose significantly. However HuMoR

Data	Occ. Leg			Occ. Arm+hand			Occ. Shoulder +Upper Arm		
# frames	60	120	240	60	120	240	60	120	240
Method									
VPoser [49]	2.53	2.57	2.54	8.51	8.52	8.59	9.98	9.49	9.48
HuMoR [52]	5.60	6.19	9.09	7.83	8.44	10.25	**4.75**	**5.11**	**4.95**
Pose-NDF	**2.49**	**2.51**	**2.47**	**7.81**	**8.13**	**7.98**	7.63	7.89	6.76

Table 2. Motion estimation from partial 3D observations: We compare per-vertex error (in *cm*). It can be seen that for leg and arm/hand occlusions, Pose-NDF reconstructs the pose better than VPoser and HuMoR. For occluded shoulders, HuMoR takes the lead. We observe that results of Pose-NDF depend on the initialization of the occluded joint, as it is expected from manifold projection.

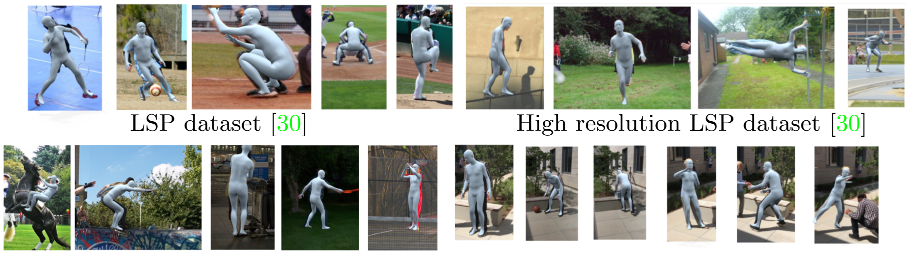
changes the pose and this change increases with an increasing number of frames. This is because HuMoR is a motion-based prior (conditioned on the previous pose) and, hence, over time the correction in pose accumulates and makes the output pose significantly different from the input.

For the "Noisy AMASS" data, Pose-NDF-based optimization outperforms prior work. We visualise the denoising results in Fig. 2, and observe that the Pose-NDF-based method produces realistic and close to GT results. We further compare results of a sequence with HuMoR in Fig. 3. HuMoR results in large deviations from the input/GT, due to accumulation of correction over time.

Fitting to partial data. We use the test set of AMASS and randomly create occluded poses (e.g. missing arm or legs or shoulder joint) and quantitatively compare with HuMoR [52] and VPoser [49] in Table 2. We use Eq. (7) for VPoser and Pose-NDF-based optimization. We only optimize for the occluded joints and for our model, we initialize the occluded joint pose randomly (close to 0). For HuMoR, we use the *TestOpt* provided in their paper. We evaluate on three different type of occlusions: 1) occluded left leg, 2) occluded left arm and 3) occluded right shoulder and upper arm. For the occluded leg case, VPoser and our prior-based method perform better. We believe this is because the majority of the poses in both AMASS training and test are upright with nearly straight legs and hence VPoser is biased towards these poses. For our method, it highly depends on initialization. Since we have used an initialization close to rest position, our optimization method generates smaller error for occluded legs but higher errors for occluded arms and shoulders, as they usually are more far away from the rest pose. For HuMoR, the motion generated is realistic and plausible, but in some cases results in large deviation from ground truth, because the correction in input pose accumulates over the time.

4.3 3D pose Estimation from Images

We now show that Pose-NDF can also be used as a prior in optimization-based 3D pose estimation from images [49]. We use the objective function proposed in SMPLify-X [49], see Eq. (10). Since we are working with a SMPL body only (without hands or faces), we remove the respective loss and prior terms. Thus,



LSP dataset [30]

High resolution LSP dataset [30]

COCO dataset [35]

3DPW dataset [40]

We

further show

Fig. 4. 3D pose and shape estimation from in-the-wild images using Pose-NDF-based optimization method.

Method	Optimization			ExPose	ExPose + Optimization			
	VPoser [49]	GAN-S [16]	Pose-NDF	-	+No prior	+ VPoser [49]	+ GAN-S [16]	+**Pose-NDF**
Per-vertex error (*mm*)	60.34	59.18	57.39	54.76	99.78	67.23	54.09	53.81

Table 3. 3D pose and shape estimation from images using Pose-NDF, GAN-S [16] and VPoser [49] as pose prior terms in optimization-based method (**left**). We also use proposed prior and optimization pipeline to further improve the results of the SoTA 3D pose and shape estimation network, ExPose [14] (**right**).

we find the desired pose $\hat{\boldsymbol{\theta}}$ and shape $\hat{\boldsymbol{\beta}}$ as:

$$\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\beta}, \boldsymbol{\theta}} \mathcal{L}_J + \lambda_{\boldsymbol{\theta}} \mathcal{L}_{\boldsymbol{\theta}} + \lambda_{\boldsymbol{\beta}} \mathcal{L}_{\boldsymbol{\beta}} + \lambda_\alpha \mathcal{L}_\alpha, \tag{10}$$

with data term $\mathcal{L}_J$, bending term $\mathcal{L}_\alpha$, shape regularizer $\mathcal{L}_{\boldsymbol{\beta}}$, and prior term $\mathcal{L}_{\boldsymbol{\theta}}$. The data term and the bending term are given as:

$$\mathcal{L}_J = \sum_{i \in \text{joints}} \gamma_i w_i \rho(\Pi_K(R_\theta(J(\boldsymbol{\beta}))) - J_{\text{est,i}}) \quad \mathcal{L}_\alpha = \sum_{i \in (\text{elbow,knees})} \exp(\boldsymbol{\theta}_i), \tag{11}$$

where $J_{\text{est,i}}$ are 2D pose keypoints estimated by a SoTA 2D-pose estimation method [11], R_θ transforms the joints along the kinematic tree according to the pose $\boldsymbol{\theta}$, Π_K represents a 3D to 2D projection with intrinsic camera parameters and ρ represents a robust Geman-McClure error [18]. Further, the bending term $\mathcal{L}_\alpha$ penalizes large bending near the elbow and knee joints, and the shape regularizer is given as $\mathcal{L}_{\boldsymbol{\beta}} = ||\boldsymbol{\beta}||^2$ [49]. For VPoser, the prior term is given as $\mathcal{L}_\theta = ||z||_2^2$, where z is the 32-dimensional latent vector of the VAE. In our model, we use $\mathcal{L}_\theta = f^{\text{udf}}(\boldsymbol{\theta})$ and minimize the distance of the pose from our learned manifold using our projection algorithm. We leverage the distance information provided by our model in optimization by setting $\lambda_{\boldsymbol{\theta}} = w f^{\text{udf}}(\boldsymbol{\theta})$, where w has a fixed value. This ensures that if the pose is getting close to the manifold (i.e. $f^{\text{udf}}(\boldsymbol{\theta})$ is very small), the prior term is down-weighted, which results in faster convergence.

Results. We use the EHF dataset [49] for quantitative evaluation and compare our work with the state-of-the-art priors VPoser [49] and GAN-S [16]. A Pose-NDF prior term slightly improves on the VPoser and GAN-S based optimization

(Tab. 3). We observe that the neural network based model ExPose [14] outperforms all optimization-based results. However, we show that such methods can benefit from an optimization-based refinement step. We refine the ExPose output using Eq (10) with Pose-NDF as prior and compare this refinement with no-prior and other priors (Tab. 3). With no prior, the optimization objective only minimizes the joint projection loss, resulting in unrealistic poses. In contrast, GAN and Pose-NDF improve the result (qualitatively and quantitatively), generating realistic poses, while Pose-NDF outperforms the GAN prior. Finally, in Fig. 4 we show qualitative results of optimization-based 3D pose estimation on in-the-wild images from 3DPW [40], LSP [30] and MS-COCO [35] datasets.

4.4 Pose Generation

We evaluate our model on the task of pose generation. Due to our distance field formulation, we can generate diverse poses by sampling a random point from $SO(3)^K$ and projecting it onto the manifold (Sec. 3.4). We compare the results of our model with sampling from the state-of-the-art pose prior VPoser [49], GMM [10,46] and GAN-S [16] in Fig. 5. We use Average Pairwise Distance (APD) [3], to quantify the diversity of generated poses. APD is defined as mean joint distance between all pairs of samples. We randomly sample 500 poses for each GMM, VPoser, GAN-S and Pose-NDF, which results in APD values of **48.24**, **23.13**, **27.52**, **32.31** (in cm), respectively. We see that numerically, the GMM produces very large variance, but also results in unrealistic poses, as seen in Fig 5 (top-left). Pose-NDF generates more diverse poses than VPoser while producing only plausible poses. We also calculate the percentage of self-intersecting faces in generated poses, to evaluate one aspect of realism in poses. Pose-NDF generates poses with less self-intersecting faces (**0.89**%), as compared to the GAN-S (**1.43**%) and VPoser (**2.10**%).

4.5 Pose Interpolation

Pose-NDF learns a manifold of plausible human poses, so it can be used to interpolate between two distinct poses by traversing the manifold. Specifically, for any given pose, we first project start ($\boldsymbol{\theta}_0$) and end pose ($\boldsymbol{\theta}_\mathrm{T}$) on our manifold using Eq (6), to get $\boldsymbol{\theta}'_0$ and $\boldsymbol{\theta}'_\mathrm{T}$. We then move along the direction of $\boldsymbol{\theta}'_\mathrm{T}$ from $\boldsymbol{\theta}'_0$ with step size τ using Eq (12). The interpolated pose ($\boldsymbol{\theta}_\mathrm{t}$) is again projected on the manifold to get a realistic pose ($\boldsymbol{\theta}'_\mathrm{t}$). In the subsequent interpolation steps, we move from $\boldsymbol{\theta}'_\mathrm{t}$ to $\boldsymbol{\theta}'_\mathrm{T}$, where $\boldsymbol{\theta}'_\mathrm{t}$ is updated after each step.

$$\boldsymbol{\theta}_\mathrm{t} = \boldsymbol{\theta}'_{\mathrm{t}-1} + \tau(\boldsymbol{\theta}'_\mathrm{T} - \boldsymbol{\theta}'_{\mathrm{t}-1}) \tag{12}$$

Results: We compare the results of Pose-NDF with those from VPoser [49] and GAN-S [16] interpolation. For VPoser [49], we project the start and end pose into the latent space and perform linear interpolation using the latent vectors. For GAN-S [16], we use the spherical interpolation in latent space, as suggested in the work. We qualitatively evaluate the interpolation quality by calculating mean

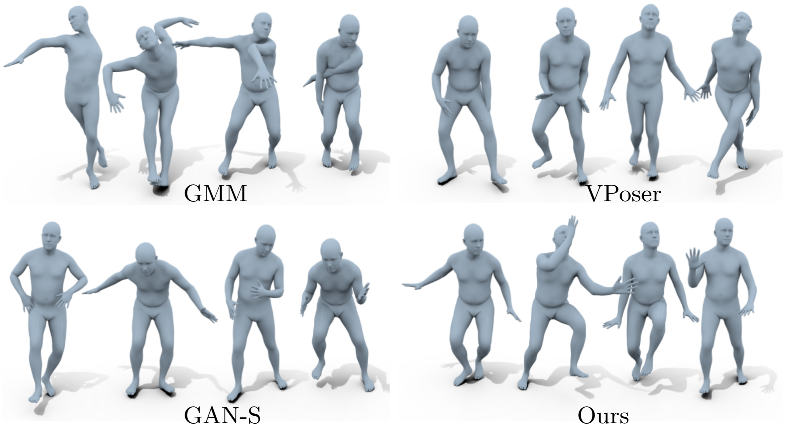


Fig. 5. Pose generation: GMM generates wrong and unrealistic poses, whereas VPoser, GAN-S and Pose-NDF generate much more realistic poses. We notice from APD, that variance of poses generated by Pose-NDF (32.31 cm) is larger than VPoser (23.13 cm) and GAN-S (27.52 cm).

per-vertex distance between consecutive frames. Smaller value means smooth interpolation. We observe that Pose-NDF-based interpolation has a mean per-vertex distance of 2.72 ± 2.16, GAN-S has 2.71 ± 2.45 and VPoser has 2.53 ± 4.62, which shows that Pose-NDF and GAN-S based interpolation is smooth and the distance in input space is not entirely preserved in case of VAEs. We compare VPoser based interpolation with Pose-NDF in Fig. 6) and observe large jumps in VPoser interpolation. This behaviour is not observed in GAN-S and Pose-NDF based interpolation. Since the VAE learns a compact latent representation of poses, the distance between two input poses is not preserved in the latent space.

4.6 Pose-NDF vs. Gaussian Assumption models

Prior work [52,49] uses VAE-based models as pose/motion prior, which follow a Gaussian assumption in the latent space. This has three major limitations, as mentioned in Sec. 1. Conversely, Pose-NDF learns the manifold directly in the pose-space without such assumptions and, hence, overcomes these limitations.

We report the cumulative error based on deviation from the mean pose. We evaluate on AMASS Noisy (60 and 120 frames) and report cumulative error for samples with $\sigma, 2\sigma, 3\sigma$ for both Pose-NDF and VPoser motion denoising. We obtain per-vertex error of $\mathbf{8.18}, \mathbf{8.20}, \mathbf{8.21}$ *cm* for Pose-NDF and $\mathbf{8.35}, \mathbf{9.11}, \mathbf{9.13}$ *cm* for VPoser, and $\mathbf{10.08}, \mathbf{11.38}, \mathbf{16.86}$ *cm* for HuMoR which reflects that VPoser and HuMoR perform well for poses close to the mean but the error increases for samples deviating from mean pose. Since the Gaussian distribution is unbounded, it produces *dead regions*, without any data points in these parts of distribution. Hence sampling in these regions might result in completely unrealistic poses for GMM and VPoser (Fig. 5). Lastly, since we learn the manifold

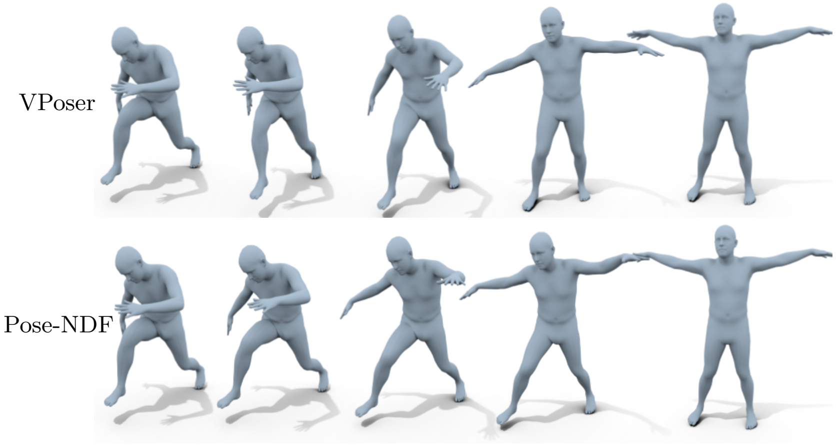


Fig. 6. Pose interpolation: We observe that VPoser-based interpolation (**top**) is less smooth than Pose-NDF-based pose interpolation (**bottom**).

in pose space, the distance between individual poses is preserved and leads to smoother interpolation compared to VPoser (see Sec. 4.5).

5 Conclusion

We introduced a novel human pose prior model represented by a scalar neural distance field that describes a manifold of plausible poses as zero level set in $SO(3)^K$. The method extends the idea of classic 3D shape representation using neural fields to higher the dimensions of human poses and maps quaternion-based poses to an unsigned distance value, representing the distance to the pose manifold. The resulting network can be used to project arbitrary poses to the pose manifold, opening applications in several areas. We comprehensively evaluate the performance of our model in diverse pose sampling, pose estimation from images, and motion denoising. We show that our model is able to generate poses with much more diversity than prior VAE-based works and improves state-of-the-art results in reconstruction from images and motion estimation.

Acknowledgements. Special thanks to the RVH team and reviewers, their feedback helped improve the manuscript and Andrey Davydov, for providing the code for GAN-based pose prior. This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 409792180 (Emmy Noether Programme, project: Real Virtual Humans), German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A and a Facebook research award. Gerard Pons-Moll is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645. The project was made possible by funding from the Carl Zeiss Foundation.

References

1. Akhter, I., Black, M.J.: Pose-conditioned joint angle limits for 3D human pose reconstruction. In: CVPR (2015) 3
2. Aksan, E., Kaufmann, M., Hilliges, O.: Structured prediction helps 3D human motion modelling. In: ICCV (2019) 4, 5
3. Aliakbarian, S., Sadat Saleh, F., Salzmann, M., Petersson, L., Gould, S.: A stochastic conditioning scheme for diverse human motion prediction. In: CVPR (2020) 12
4. Alldieck, T., Magnor, M., Bhatnagar, B.L., Theobalt, C., Pons-Moll, G.: Learning to reconstruct people in clothing from a single RGB camera. In: CVPR (2019) 3
5. Andreou, N., Lazarou, A., Aristidou, A., Chrysanthou, Y.: A hierarchy-aware pose representation for deep character animation (2021) 4
6. Baerlocher, P., Boulic, R.: Parametrization and range of motion of the ball-and-socket joint. In: Proceedings of the IFIP TC5/WG5.10 DEFORM'2000 Workshop and AVATARS'2000 Workshop on Deformable Avatars (2000) 3
7. Barsoum, E., Kender, J., Liu, Z.: HP-GAN: probabilistic 3D human motion prediction via gan. In: CVPR Workshops (2018) 3
8. Bhatnagar, B.L., Tiwari, G., Theobalt, C., Pons-Moll, G.: Multi-garment net: Learning to dress 3D people from images. In: ICCV (2019) 3
9. Bhatnagar, B.L., Xie, X., Petrov, I., Sminchisescu, C., Theobalt, C., Pons-Moll, G.: BEHAVE: Dataset and method for tracking human object interactions. In: CVPR (2022) 1, 4
10. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In: ECCV (2016) 2, 3, 12
11. Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., Sheikh, Y.A.: OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019) 11
12. Chabra, R., Lenssen, J.E., Ilg, E., Schmidt, T., Straub, J., Lovegrove, S., Newcombe, R.A.: Deep local shapes: Learning local SDF priors for detailed 3D reconstruction. In: ECCV (2020) 4
13. Chibane, J., Mir, A., Pons-Moll, G.: Neural unsigned distance fields for implicit function learning. In: NeurIPS (2020) 2, 3, 4, 6
14. Choutas, V., Pavlakos, G., Bolkart, T., Tzionas, D., Black, M.J.: Monocular expressive body regression through body-driven attention. In: ECCV (2020) 3, 11, 12
15. Crandall, M.G., Lions, P.L.: Viscosity solutions of Hamilton-Jacobi equations. Transactions of the American mathematical society **277**(1), 1–42 (1983) 6
16. Davydov, A., Remizova, A., Constantin, V., Honari, S., Salzmann, M., Fua, P.: Adversarial parametric pose prior. In: CVPR (2022) 4, 11, 12
17. Engell-Nørregård, M., Niebe, S., Erleben, K.: A joint-constraint model for human joints using signed distance-fields. Multibody System Dynamics **28** (2012) 3
18. Geman, S., McClure, D.E.: In: Statistical methods for tomographic image reconstruction (1987) 11
19. Georgakis, G., Li, R., Karanam, S., Chen, T., Košecká, J., Wu, Z.: Hierarchical kinematic human mesh recovery. In: ECCV (2020) 3, 5
20. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NeurIPS (2014) 3
21. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. In: Proceedings of Machine Learning and Systems (2020) 4, 6

22. Guo, C., Zuo, X., Wang, S., Zou, S., Sun, Q., Deng, A., Gong, M., Cheng, L.: Action2Motion: Conditioned generation of 3d human motions. In: ACMMM (2020) 1
23. Guzov, V., Mir, A., Sattler, T., Pons-Moll, G.: Human POSEitioning System (HPS): 3D human pose estimation and self-localization in large scenes from body-mounted sensors. In: CVPR (2021) 1, 3, 8
24. He, T., Xu, Y., Saito, S., Soatto, S., Tung, T.: Arch++: Animation-ready clothed human reconstruction revisited. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021) 4
25. Herda, L., Urtasun, R., Fua, P.: Hierarchical implicit surface joint limits for human body tracking. In: ECCV (2004) 4
26. Herda, L., Urtasun, R., Hanson, A.: Automatic determination of shoulder joint limits using quaternion field boundaries. In: FG (2002) 4
27. Huang, Z., Xu, Y., Lassner, C., Li, H., Tung, T.: Arch: Animatable reconstruction of clothed humans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3093–3102 (2020) 4
28. Huynh, D.Q.: Metrics for 3D rotations: Comparison and analysis. J. Math. Imaging Vis. **35**(2), 155–164 (oct 2009) 5
29. Ji, Y., Xu, F., Yang, Y., Shen, F., Shen, H.T., Zheng, W.S.: A large-scale RGB-D database for arbitrary-view human action recognition. In: ACM International Conference on Multimedia (ACMMM) (2018) 1
30. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: BMVC (2010) 11, 12
31. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: CVPR (2018) 3, 4
32. Kocabas, M., Athanasiou, N., Black, M.J.: Vibe: Video inference for human body pose and shape estimation. In: CVPR (2020) 3
33. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In: ICCV (2019) 3
34. Krebs, F., Meixner, A., Patzer, I., Asfour, T.: The kit bimanual manipulation dataset. In: IEEE/RAS International Conference on Humanoid Robots (Humanoids) (2021) 1
35. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft COCO: Common Objects in Context (2014) 11, 12
36. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM Trans. Graphics (Proc. SIGGRAPH Asia) **34**(6), 248:1–248:16 (2015) 4, 8
37. Loper, M.M., Mahmood, N., Black, M.J.: MoSh: Motion and shape capture from sparse markers. ACM Transactions on Graphics, (Proc. SIGGRAPH Asia) (2014) 1
38. Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: AMASS: Archive of motion capture as surface shapes. In: ICCV (2019) 3, 7, 8
39. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I.: Adversarial autoencoders. In: ICLR (2016) 2
40. von Marcard, T., Henschel, R., Black, M., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3d human pose in the wild using imus and a moving camera. In: ECCV (2018) 3, 11, 12
41. von Marcard, T., Henschel, R., Black, M.J., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3D human pose in the wild using IMUs and a moving camera. In: ECCV (2018) 1

42. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3D reconstruction in function space. In: CVPR (2019) 4
43. Mihajlovic, M., Zhang, Y., Black, M.J., Tang, S.: LEAP: Learning articulated occupancy of people. In: CVPR (2021) 4, 5, 7
44. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020) 4
45. Novello, T., da Silva, V., Lopes, H., Schardong, G., Schirmer, L., Velho, L.: Neural implicit surfaces in higher dimension (2022) 4
46. Omran, M., Lassner, C., Pons-Moll, G., Gehler, P., Schiele, B.: Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In: 3DV (2018) 12
47. Ormoneit, D., Sidenbladh, H., Black, M., Hastie, T.: Learning and tracking cyclic human motion. Advances in Neural Information Processing Systems **13** (2000) 3
48. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: DeepSDF: Learning continuous signed distance functions for shape representation. In: CVPR (2019) 2, 3, 4
49. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3D hands, face, and body from a single image. In: CVPR (2019) 2, 3, 4, 8, 9, 10, 11, 12, 13
50. Petrovich, M., Black, M.J., Varol, G.: Action-conditioned 3D human motion synthesis with transformer VAE. In: ICCV (2021) 4
51. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-NeRF: Neural radiance fields for dynamic scenes. In: CVPR (2020) 4
52. Rempe, D., Birdal, T., Hertzmann, A., Yang, J., Sridhar, S., Guibas, L.J.: HuMoR: 3D human motion model for robust pose estimation. In: ICCV (2021) 2, 3, 4, 7, 8, 9, 10, 13
53. Saito, S., Yang, J., Ma, Q., Black, M.J.: SCANimate: Weakly supervised learning of skinned clothed avatar networks. In: CVPR (2021) 4
54. Sarafianos, N., Boteanu, B., Ionescu, B., Kakadiaris, I.A.: 3D human pose estimation: A review of the literature and analysis of covariates. Computer Vision and Image Understanding **152**, 1–20 (2016) 3
55. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: NTU RGB+D: A large scale dataset for 3d human activity analysis. In: CVPR (2016) 1
56. Shao, W., Ng-Thow-Hing, V.: A general joint component framework for realistic articulation in human characters. In: Proceedings of the 2003 symposium on Interactive 3D graphics. pp. 11–18 (2003) 3
57. Sidenbladh, H., Black, M.J., , Fleet, D.: Stochastic tracking of 3D human figures using 2D image motion. In: ECCV (2000) 3
58. Sitzmann, V., Martel, J.N., Bergman, A.W., Lindell, D.B., Wetzstein, G.: Implicit neural representations with periodic activation functions. In: NeurIPS (2020) 4
59. Stoll, C., Gall, J., de Aguiar, E., Thrun, S., Theobalt, C.: Video-based reconstruction of animatable human characters. In: ACM SIGGRAPH Asia (2010) 3
60. Tiwari, G., Bhatnagar, B.L., Tung, T., Pons-Moll, G.: SIZER: A dataset and model for parsing 3D clothing and learning size sensitive 3D clothing. In: ECCV (2020) 3
61. Tiwari, G., Sarafianos, N., Tung, T., Pons-Moll, G.: Neural-GIF: Neural generalized implicit functions for animating people in clothing. In: ICCV (2021) 4
62. Urtasun, R., Fleet, D., Fua, P.: 3D people tracking with Gaussian process dynamical models. In: CVPR (2006) 3

63. Xie, X., Bhatnagar, B.L., Pons-Moll, G.: Chore: Contact, human and object reconstruction from a single rgb image. In: European Conference on Computer Vision (ECCV). Springer (October 2022) 4
64. Xie, Y., Takikawa, T., Saito, S., Litany, O., Yan, S., Khan, N., Tombari, F., Tompkin, J., Sitzmann, V., Sridhar, S.: Neural fields in visual computing and beyond. Computer Graphics Forum (2022) 4
65. Xu, Y., Zhu, S.C., Tung, T.: Denserac: Joint 3D pose and shape estimation by dense render and compare. In: International Conference on Computer Vision (2019) 3
66. Zhang, S., Zhang, H., Bogo, F., Pollefeys, M., Tang, S.: Learning motion priors for 4D human body capture in 3D scenes. In: ICCV (2021) 2, 4, 9
67. Zhang, S., Zhang, Y., Ma, Q., Black, M.J., Tang, S.: PLACE: Proximity learning of articulation and contact in 3D environments. In: 3DV (2020) 1
68. Zhang, Y., Hassan, M., Neumann, H., Black, M.J., Tang, S.: Generating 3D people in scenes without people. In: CVPR (2020) 1
69. Zhou, K., Bhatnagar, B.L., Lenssen, J.E., Pons-Moll, G.: Toch: Spatio-temporal object correspondence to hand for motion refinement. In: European Conference on Computer Vision (ECCV). Springer (October 2022) 4
70. Zou, S., Zuo, X., Qian, Y., Wang, S., Xu, C., Gong, M., Cheng, L.: 3D human shape reconstruction from a polarization image. In: ECCV (2020) 1