International Journal of Computer Vision (2021) 129:2499–2512
https://doi.org/10.1007/s11263-021-01488-2

Deep Unsupervised 3D Human Body Reconstruction from a Sparse set of Landmarks

Meysam Madadi[1] · **Hugo Bertiche**[2] · **Sergio Escalera**[2]

Received: 1 September 2020 / Accepted: 3 June 2021 / Published online: 15 June 2021

Abstract
In this paper we propose the first deep unsupervised approach in human body reconstruction to estimate body surface from a sparse set of landmarks, so called *DeepMurf*. We apply a denoising autoencoder to estimate missing landmarks. Then we apply an attention model to estimate body joints from landmarks. Finally, a cascading network is applied to regress parameters of a statistical generative model that reconstructs body. Our set of proposed loss functions allows us to train the network in an unsupervised way. Results on four public datasets show that our approach accurately reconstructs the human body from real world mocap data.

Keywords Human body reconstruction · Mocap data · Unsupervised deep learning · Attention model · Cascading

1 Introduction

Reconstruction of 3D human body has a great applicability in many domains including pose and shape retargeting, movie editing, videogame industry and virtual reality, just to name a few. This is a particularly challenging problem, since the solution must deal with 3D joint locations and orientations along with subject specific body surface. Besides, data acquisition and annotation is an expensive process, specially for supervised approaches.

Given the difficulty of acquiring 3D ground truth body surface, this problem is tackled unsupervisedly in different ways in the literature. On the one hand, image-based approaches (Kanazawa et al. 2018; Omran et al. 2018) try to reconstruct 3D body from 2D data gathered from images. However, this is known as ill-posed since depth is lost in the projection to the image plane and 2D-to-3D reconstruction can have multiple solutions for the same 2D data. Some authors propose multi-camera setups (Joo et al. 2018; Rhodin et al. 2016; Mehrizi et al. 2018) to cope with this problem. On the other hand, mocap-based pose estimation has become a standard procedure in domain specific applications like movie and videogame industry. However, standard procedures are not able to reconstruct body surface. In this regard, Loper et al. (2014) showed that a sparse set of physical landmarks attached to the body is enough to reconstruct the whole body surface.

In this paper we focus on mocap-based solutions. Prior works Loper et al. (2014) and Mahmood et al. (2019) are based on regular optimization techniques in which a statistical body model is fit to the input landmarks. These approaches are able to reconstruct the body surface accurately, though, in a significant amount of time. Besides, optimization is applied in several steps, e.g., body shape is optimized first and later pose is conditioned on shape. This further prevents these approaches to be utilized in on-the-fly applications. On the contrary, in this paper we propose a deep unsupervised approach to reconstruct 3D body surface from single frame mocap data which is fast in training and testing time, able to generate accurate results. Working on single frames, rather than sequences, relaxes the network from the need of enormous temporal data, allowing our approach to be trained on small custom datasets. To the best of our knowledge this is the first time a deep unsupervised approach is applied to this problem.

Specifically, we build our approach based on a widely used statistical generative model (SMPL) (Loper et al. 2015). SMPL receives body pose and shape parameters, updates a

Communicated by Javier Romero.

✉ Meysam Madadi
mmadadi@cvc.uab.es

[1] Computer Vision Center and Universitat de Barcelona, Barcelona, Spain

[2] Universitat de Barcelona and Computer Vision Center, Barcelona, Spain

template mesh and generates body surface through forward kinematic. The goal of this paper is to estimate SMPL pose and shape parameters from mocap data through deep learning such that generated surface best fits the input 3D landmarks. There are several challenges in this task: (1) landmarks are noisy and sometimes missing, (2) although SMPL is differentiable, it is sensitive to noise, and (3) the space of human pose and shape is highly non-linear. These challenges produce many local minima and an efficient architecture and training procedure are crucial for model generalization. Given the aforementioned challenges, one goal in this paper is to design a network that can work well in small size custom datasets.

We inspire our architecture from SMPLR (Madadi et al. 2018), in which a set of landmarks are predicted from input RGB images and used to estimate body pose and shape. SMPLR is a supervised approach which is not applicable on the problem at hand. Our main contribution is to redesign SMPLR for unsupervised training. Specifically, our contributions are as follows. Our proposed architecture is composed of a denoising autoencoder (similar to SMPLR) to recover missing landmarks, an attention model to estimate joints from landmarks, and a cascading regression model to estimate body pose and shape parameters. We train the model unsupervisedly, that is, 3D joints and body pose and shape parameters are unknown. To do so we propose several loss functions including regularization on pose and shape parameters, landmarks-to-surface loss and denosing autoencoder and attention model loss functions. We also propose a novel unposing layer which helps to generalize better when there is a low amount of data. Finally, we provide an extensive analysis of the architecture and loss functions on four public datasets (Varol et al. 2017; Mahmood et al. 2019; Hoyet et al. 2012; Lab 2000). Particularly, our approach can deal with missing landmarks, accurately estimates joints and body surface (including hands) and performs well when trained on small datasets.

2 Related Works

Human pose and shape recovery has been an extensively studied field of research in the recent years. There is a large literature on the topic which could be classified according to considered input data: image, IMUs, mocap or combinations, or according to methodology: energy optimization, database search or deep learning. Regarding deep learning, we could further subdivide it into supervised and unsupervised learning. Nonetheless, to the best of our knowledge, we are the first ones to try a direct mapping from sparse mocap sensor data to body surface through unsupervised deep learning.

2.1 Modalities

Image This category is the most extensive. On one hand, we have multi-view setups (Joo et al. 2018; Rhodin et al. 2016; Mehrizi et al. 2018), which can be tackled through energy optimization or deep learning. Multi-view data significantly reduces problem complexity, but it requires a constrained scenario, which limits applicability. On the other hand, we find monocular RGB approaches. Deep learning methodologies are predominant in current literature, outperforming traditional, non-deep, strategies. Volumetric body prediction is a highly complex task, so most works rely on parametric models such as SMPL (Loper et al. 2015) to predict 3D body model. This is done by direct regression (Kanazawa et al. 2018) or through intermediate representations (Pavlakos et al. 2018; Omran et al. 2018; Madadi et al. 2018; Varol et al. 2018; Mehta et al. 2020). We also find works that rely on depth maps (Bogo et al. 2015; Achilles et al. 2016), though, as with multi-view, it simplifies the problem while requiring a specific setup. Works based on RGB data are prone to be domain-dependant, impairing their generalization and applicability. In this work we propose recovering body pose and shape only from mocap-like data which yields a domain-independant model.

IMUs Inertial Measurement Units are a very common sensor, found in smartphones, gaming devices and airplanes. They provide acceleration and orientation data, but no location. Due to their availability, many researchers proposed methodologies to obtain body pose from sparse IMUs. Authors of von Marcard et al. (2017) propose an energy optimization approach to obtain SMPL parameters from temporal IMU data. These works (Slyper and Hodgins 2008; Tautges et al. 2011) compare input data against a prerecorded database. Schwarz et al. (2009) propose a Gaussian Process Regression to map IMU data to full body pose, though, it has generalization problems. Finally, Huang et al. (2018) apply deep learning to the recordings of 6 IMU sensors to predict body pose with a RNN. While IMUs are cheap, their lack of location data renders them useless to determine subject body shape and often temporal data is required to solve pose ambiguities. These drawbacks are not present for mocap sensors.

Mocap This data contains the location of a set of sparse landmarks evenly placed through body surface. They are the film and animation industry standards on motion capture because of the accuracy of measurements, while on the other hand, require an specific multi-view setup for correct location tracking. Similar to IMU-based approaches, we find energy optimization methods, such as MoSH (Loper et al. 2014) or Park and Hodgins (2006), where body pose and shape are obtained from sparse mocap markers. Although energy optimization methods generate good results, they do not achieve real-time performance. Other approaches have also been explored (Chai and Hodgins 2005; Liu et al. 2011),

where input data is compared against pre-recorded databases. These approaches mainly suffer from lack of generalization. Instead of sparse landmarks, some works (Groueix et al. 2018; Prokudin et al. 2019; Bhatnagar et al. 2020a, b) predict body (or outfit) surface from dense point clouds. These approaches perform as registration techniques to find correspondences. Our work proposes a mapping from sparse mocap data to SMPL pose and shape parameters through deep learning, which, to the best of our knowledge, has not been previously explored.

RGB+IMU/Mocap Some works like (Von Marcard et al. 2016; von Marcard et al. 2018) use combination of modalities, RGB plus IMUs in this case, to improve accuracy on pose prediction by complementing the drawbacks of each data type with the other. Similarly, in Trumble et al. (2017), propose an RGB multi-view plus IMUs setup, from which volumetric probabilistic visual hull data is extracted and fed to a 3DCNN for human 3D pose prediction. Zhao et al. (2012) use markers (mocap-like data) and kinect sensor data (RGB-D) to obtain accurate 3D hand models as an optimization problem.

2.2 Unsupervised Approaches

Aforementioned deep-based strategies work in a supervised scenario. Unsupervised learning has not been widely explored yet in human pose and shape recovery. As proposed in Kudo et al. (2018), Chen et al. (2019), 3D pose can be mapped from RGB in an unsupervised manner only after decomposing the problem in 2D joint detection from an image and 2D-to-3D joint lifting. The 2D joints are assumed to be accurate enough and the lifting part is learnt without direct supervision by back-projecting predictions and ensuring consistency w.r.t. 2D *annotations*. Human shape recovery without supervision has not been tackled yet, but in Insafutdinov and Dosovitskiy (2018), propose an unsupervised learning of an RGB-to-3D mapping for generic objects, also based on back-projection and comparison w.r.t. visual evidence.

3 Unsupervised Surface Reconstruction from Landmarks

Let $\mathcal{X} = \{\mathbf{L}\}_{i=1}^{n}$ be a dataset where $\mathbf{L}_i \in \mathbb{R}^{l\times 3}$ is a set of l landmarks in 3D coordinates for a given frame. Landmarks have several properties: (1) the set $\mathbf{L}$ is ordered, that is, $j-$th landmark is always attached to the same location on the body, (2) landmarks are sparse, i.e. they cover a few locations on the body ($l < 100$ in this paper), and (3) landmarks are noisy with missing data. There are two types of noise in the landmarks: (1) small perturbation in the attachment location and (2) large measurement error that may cause some landmarks appear far from where they should be. Finally, there is no guarantee that there will always be l landmarks attached to the body. Therefore, l is the number of all possible landmarks in the dataset $\mathcal{X}$. In this paper we use the same set of landmarks as in Loper et al. (2014) and assume missing or corrupted landmarks are known beforehand. We define valid landmarks as a mask by matrix $\mathbf{M} \in \{0, 1\}^{l\times 3}$.

The goal is to train a neural network on the dataset $\mathcal{X}$ to output a dense set of 3D body surface points $\mathbf{T}_{out} \in \mathbb{R}^{p\times 3}$ that best fits the data. To do so, we apply a hybrid approach which combines deep learning with SMPL, a statistical generative model, that reconstructs body surface. SMPL receives axis-angle pose parameters $\theta \in \mathbb{R}^{24\times 3}$ along with shape coefficients $\beta \in \mathbb{R}^{10}$ to generate body surface $\mathbf{T}_{out}$ with $p = 6890$. Therefore, the network is simplified to regress SMPL parameters from input landmarks. Regressing SMPL parameters allows us to apply pose and shape retargeting in custom applications, as shown in Fig. 1. The original SMPL implementation has two separate models for male and female. As a common practice, we use a neutral gender model that simplifies the training.

The pipeline of the architecture is shown in Fig. 2. First, we apply a denoising autoencoder to recover missing and noisy landmarks. Then an attention network is used to estimate body joints. Finally, recovered landmarks and estimated joints are concatenated as input to a cascading network to regress SMPL pose and shape parameters. Next, we explain details of each part and how we train the network unsupervisedly.

3.1 Recover Missing Landmarks

Missing landmarks are unavoidable during the setup and capturing process. Therefore a neural network must be able to deal with a variable number of landmarks per frame (or per sequence of frames). A standard neural network (e.g. a MLP network) requires ordered and fixed size arrays. By knowing landmark labels and matrix $\mathbf{M}$, we can fill valid landmarks in each frame to form the input matrix $\mathbf{L}$. Then missing or erroneous landmarks are set to zero.

Although neural networks are able to handle data arrays with few zero values, there is no guarantee they implicitly learn the patterns of missing information. It has been shown that denoising autoencoders (Vincent et al. 2010) are useful tools to learn local representations of corrupted data. Therefore, given a large dataset, it is possible to estimate a missing landmark in a frame from neighbor frames in the representation space. In this paper we apply a denoising autoencoder network (DAE, as proposed in Madadi et al. (2018)) by the usage of fully connected layers, dropout and skip connections, as shown in Fig. 2. We train DAE for one frame using L1 loss as:

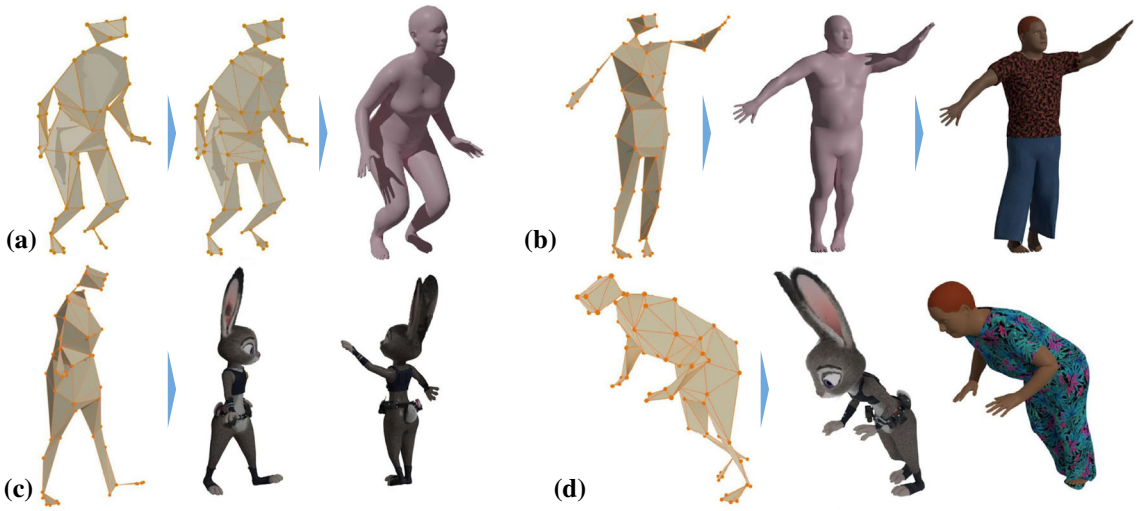

Fig. 1 Applications of landmarks to body surface reconstruction. The input to the system is a sparse set of landmarks. **a** Missing landmarks in the input are recovered and then body surface is reconstructed. **b** Garments are simulated on top of the body surface and rendered with textures. **c** Any subject can be reconstructed and re-posed from estimated pose and shape parameters. **d** More examples of retargeting and garment simulation. The edges are added to the landmarks for visualization purposes

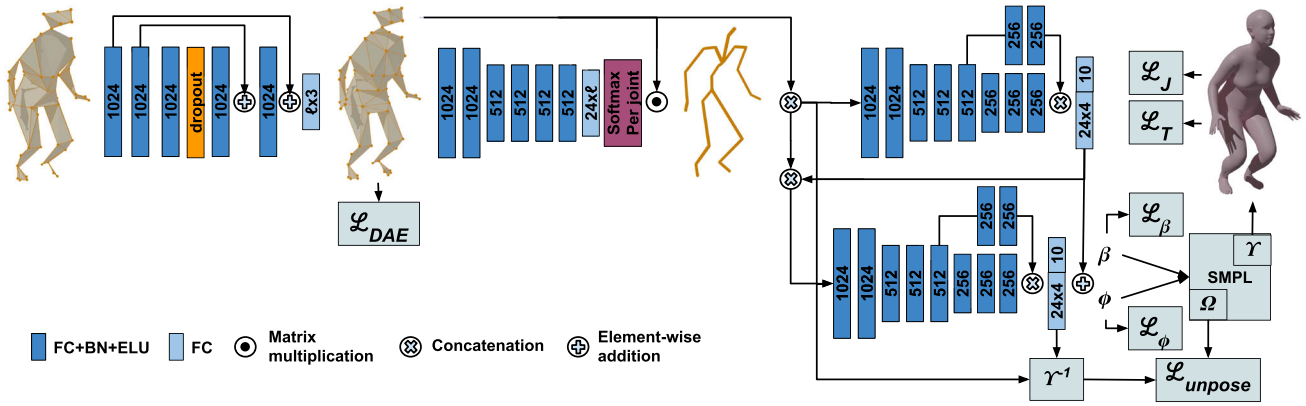

Fig. 2 Architecture pipeline. First missing landmarks are recovered in denoising autoencoder network. Then the attention network is used to estimate body joints. Both landmarks and body joints are fed to the cascading network which regress shape (β) and pose (ϕ in terms of quaternion) parameters. Finally, the body surface is generated by SMPL. Our proposed loss functions allow us to train the network unsupervisedly

$$\mathcal{L}_{DAE} = \frac{1}{l \times 3} \sum_{j=1}^{l \times 3} \mathbf{M}_j |\mathbf{L}_j - \hat{\mathbf{L}}_j|, \tag{1}$$

where $\hat{\mathbf{L}}$ is the output of DAE as the set of estimated landmarks. Note that we normalize $\mathbf{L}$ beforehand by subtracting its mean value. Therefore, we make $\mathbf{L}$ translation invariant.

3.2 Regress Pose and Shape Parameters

Our goal is to estimate SMPL pose and shape parameters from landmarks. SMPL pose is defined by axis-angle rotation of each joint w.r.t. its parent joint in the skeleton kinematic tree. Therefore, SMPL pose is a combination of local representations for each joint. This is while the landmark coordinates are represented globally. We believe a standard MLP network has difficulties to implicitly learn a direct mapping from these global landmarks to local relative axis-angle rotations, as we show in the experiments. This mapping suffers from many local minima regardless of the capacity of the network. This behavior is observed in image-based body reconstruction domain as well. Kanazawa et al. (2018) directly mapped images, as global representations, to SMPL parameters trying to handle local minima through adversarial training. Later, Madadi et al. (2018) showed that a two-step mapping could significantly improve the results, i.e. by first

mapping to, easier to extract, intermediate representations and then mapping them to SMPL parameters. Similarly, we first extract body joints from landmarks as complementary information since they are basis coordinates for these relative rotations, and use them along with landmarks to predict SMPL parameters. We do this process unsupervisedly in a unified pipeline. However, one can apply standard mocap pose estimators for this task.

Fortunately, body joint locations can be interpolated from surface vertices. In the case of sparse landmarks, the accuracy of interpolated joints depends on the placement of the landmarks. We use the same set of landmarks as in Loper et al. (2014), where an optimization is applied for the placement and importance of landmarks. It is also a standard procedure in mocap data to place at least one landmark around main body joints (e.g. wrist, elbow, hip, knee and ankle). We found standard mocap landmarks are rich enough to approximate the joints. To do so, we design an attention network (ATN) which receives updated landmarks $\hat{\mathbf{L}} = \mathbf{M} \odot \mathbf{L} + (1 - \mathbf{M}) \odot \hat{\mathbf{L}}$ and outputs joints $\mathbf{J}_{in} \in \mathbb{R}^{m \times 3}$ where $\odot$ is the hadamard product and $m = 24$ is the number of SMPL joints. The architecture can be seen in Fig. 2. $\mathbf{J}_{in}$ is computed as:

$$\mathbf{J}_{in} = \sigma(\mathbf{A}) \cdot \hat{\mathbf{L}}, \tag{2}$$

where $\mathbf{A} \in \mathbb{R}^{m \times l}$ is the last reshaped layer of ATN and σ is the softmax over the second dimension of $\mathbf{A}$. Let J^0_{in} be the estimated root joint location in the kinematic tree. We subtract both $\mathbf{J}_{in}$ and $\hat{\mathbf{L}}$ from J^0_{in} to have translation invariant data.

Now, θ and β can be regressed from $\mathbf{J}_{in}$ and $\hat{\mathbf{L}}$. First, we explain the modifications we apply to the pose parameters. In the SMPL pipeline, axis-angles θ are converted to rotation matrices through Rodrigues formulation. It is known that axis-angles are not unique and Rodrigues function is not one-to-one. This is problematic in the training due to its instability and convergence to wrong values. In the literature, axis-angles are replaced with rotation matrices, bypassing Rodrigues function. However, we believe this is not suitable either, because there is no guarantee the regression network yields valid rotation matrices. That means predicted matrices are not orthonormal. Instead, we propose to replace axis-angles θ with quaternions $\phi \in \mathbb{R}^{m \times 4}$ which are known to be unique and can be easily converted to rotation matrices.

Any proposed network must be able to efficiently map between two highly nonlinear spaces, i.e. from $\mathbf{J}_{in}$ and $\hat{\mathbf{L}}$ to ϕ and β. To cope with this, we propose a cascade of sequential networks $\{\Psi_0, \Psi_1, \ldots, \Psi_c\}$. All Ψ networks have a similar architecture without sharing weights. Let $\{\phi, \beta\} = \Psi_0(\mathbf{J}_{in}, \hat{\mathbf{L}}; \boldsymbol{\omega}_0)$ be our first pose and shape regressor where $\boldsymbol{\omega}_0$ is its trainable parameters. Then each Ψ_i , $i \neq 0$, is defined as $\{\phi, \beta\} = \Psi_i(\mathbf{J}_{in}, \hat{\mathbf{L}}, \Psi_{i-1}; \boldsymbol{\omega}_i) + \Psi_{i-1}$. The architecture with one cascade can be seen in Fig. 2.

3.3 How do we Train the Network?

In this section, we explain details of the applied loss functions and training procedure. We note that the only available information are $\mathbf{L}$ and $\mathbf{M}$ and the network must learn $\mathbf{J}_{in}$, β and ϕ unsupervisedly.

Regularization on β and ϕ Due to the lack of ground truth data on β and ϕ, we do not know the real distribution of these parameters. However, we can define upper and lower bounds on them. This is particularly important to teach the network to be aware of valid parameters, since SMPL is sensitive to noise and can converge to invalid parameters. Pose regularization for one frame is defined as:

$$\mathcal{L}_\phi = \frac{1}{m \times 4} \sum max(0, \mathbf{B}_l - \phi) + max(0, \phi - \mathbf{B}_u), \tag{3}$$

where $\mathbf{B}_l, \mathbf{B}_u \in \mathbb{R}^{m \times 4}$ are pose lower and upper bounds. We set $\mathbf{B}_l$ and $\mathbf{B}_u$ manually by checking valid angles of each joint in SMPL and converting them to quaternions. For shape regularization, we apply a standard $L1$ norm to force shape parameters close to zero, as well as keeping a hard boundary on shape:

$$\mathcal{L}_\beta = \frac{1}{10} \sum max(0, |\beta| - 5) + |\beta|. \tag{4}$$

Joints and surface loss Let $\mathbf{J}_{out} \in \mathbb{R}^{m \times 3}$, along with $\mathbf{T}_{out}$, be SMPL outputs of joints and body surface vertices. We define a loss on $\mathbf{J}_{out}$ based on an observation: $\mathbf{J}_{in}$ error is way lower than $\mathbf{J}_{out}$. So it can be used as a teacher to $\mathbf{J}_{out}$ in the loss (computed for one frame):

$$\mathcal{L}_J(\mathbf{J}_{in}, \mathbf{J}_{out}) = \frac{1}{m \times 3} \sum |\mathbf{J}_{in} - \mathbf{J}_{out}|. \tag{5}$$

To fit the surface on landmarks one must take several challenges into account: (1) landmarks are in a distance to the surface, and (2) landmarks have perturbation in their placement on the body. To cope with the first challenge, Loper et al. (2014) apply a loss to keep a landmark-to-surface distance higher than a threshold. We believe this loss is unnecessary as long as we update SMPL template vertices. Specifically, we add a vector of size 1 cm[1] to each SMPL template vertex in the direction of vertex normal. We do this once just in the training and save the network from extra complexity. To cope with the second challenge, we apply a soft landmark-to-surface assignment. That is, for each landmark we manually select and fix a patch of SMPL vertices that the landmark may appear in. Then, the nearest vertex in the patch to the landmark is the candidate for the computation of loss:

[1] We find 1 cm adequate for the experimented datasets, though can be adjusted for custom datasets.

$$\mathcal{L}_T(\hat{\mathbf{L}}, \mathbf{T}_{out}) = \frac{1}{l \times 3} \sum_{i=1}^{l} \sum_{j=1}^{3} \min_{k \in \rho_i} |\hat{\mathbf{L}}^{i,j} - \mathbf{T}_{out}^{k,j}|, \tag{6}$$

where ρ_i is a patch of assigned indices to i-th landmark. Although this loss can yield some offset error in low resolution meshes, it works well in practice and has a low complexity.

Inverse kinematic loss SMPL is a multi-valued function, that is, there are multiple valid solutions for a given body surface. SMPL is also sensitive to noise in the loss (due to the noise in the landmarks) and it hurts back-propagated gradients. To handle this problem, we force the network to provide unique solutions for SMPL, i.e. to have a one-to-one function. To do so, we assume SMPL shaped body surface in rest pose (computed in the forward path) as a canonical surface. Then, we want joints and landmarks (i.e. $\mathbf{J}_{in}$ and $\hat{\mathbf{L}}$), inverted through backward SMPL, to be perfectly similar to the canonical surface. This helps the network to be aware of geometry and improves generalization. Formally, we define this loss as follows.

Let $\{\mathbf{J}_t, \mathbf{T}_t\} = \Omega(\beta, R(\phi); \mathbf{T}_t^*, \mathcal{W})$ be the SMPL function that produces shaped vertices $\mathbf{T}_t$ and joints $\mathbf{J}_t$ from template vertices $\mathbf{T}_t^*$. We note that all variables with subscript t have a rest pose. $R(\phi)$ converts quaternions to rotation matrices and $\mathcal{W}$ is a precomputed set of SMPL parameters including blend shape functions and blend weights. Also, let $\mathbf{a}_{out} = \Upsilon(R(\phi), \mathbf{a}_t; \mathbf{W})$ be the forward kinematic function that transforms any given joints or landmarks $\mathbf{a}_t$ to the final posed form $\mathbf{a}_{out}$ where $\mathbf{W} \in \mathbb{R}^{6890 \times m}$ is the set of blend weights. Finally, let Υ^{-1} be the inverse kinematic function which must be able to unpose any given joints or landmarks. Then, we define inverse kinematic loss as:

$$\mathcal{L}_{unpose} = \mathcal{L}_J(\Upsilon^{-1}(R(\phi), \mathbf{J}_{in}), \mathbf{J}_t) + \mathcal{L}_T(\Upsilon^{-1}(R(\phi), \hat{\mathbf{L}}), \mathbf{T}_t). \tag{7}$$

Inverse kinematic function Forward kinematic function Υ is part of SMPL pipeline. Here, we explore the details of inverse kinematic function Υ^{-1}. The unposing procedure is different between joints and landmarks. We first explain this process for joints. This is done by recursive unposing of the joints in the kinematic tree through $\mathbf{R} = R^{\intercal}(\phi)$ where $R^{\intercal}$ is the transpose of rotation matrices for each joint. We show indexing operator by superscript indices, e.g. $\mathbf{J}_{in}^i$ means i-th joint.

$$\mathbf{J}_r = \left[\mathbf{J}_{in}^i - \mathbf{J}_{in}^{\kappa_i}\right]_{i=1}^{m}, \tag{8}$$

$$\mathbf{G}_r = \left[\mathbf{R}^{\kappa_i} \cdot \mathbf{G}^{\kappa_i}\right]_{i=2}^{m}, \tag{9}$$

$$\mathbf{J}_t = \left[\mathbf{G}_r^i \cdot \mathbf{J}_r^{i-1} + \mathbf{J}_t^{\kappa_i}\right]_{i=2}^{m}. \tag{10}$$

where $\kappa \in \mathbb{R}^m$ is the kinematic relationship between joints, i.e. κ_i is the parent index of i-th joint. $\mathbf{G} \in \mathbb{R}^{m \times 4 \times 4}$ is a set of m transformation matrices computed from $R(\phi)$ and $\mathbf{J}_t$ (similar to SMPL). Since this procedure is recursive, $\mathbf{J}_t^1$ is set by $\mathbf{J}_{in}^1$ and $\mathbf{G}^1$ is set by an identity matrix.

Then, we compute $\hat{\mathbf{L}}_t$ as:

$$\mathbf{G}' = \left[\mathbf{R}^i \cdot \mathbf{G}^i\right]_{i=1}^{m}, \tag{11}$$

$$\mathbf{O} = \mathbf{J}_t - \left[\mathbf{G}'^i \cdot \mathbf{J}_{in}^i\right]_{i=1}^{m}, \tag{12}$$

$$\hat{\mathbf{L}}_t = \left[[\mathbf{W} \cdot \mathbf{G}']^i \cdot \hat{\mathbf{L}}^i\right]_{i=1}^{l} + \mathbf{W} \cdot \mathbf{O}. \tag{13}$$

An example is shown in Fig. 3d. The linear transformation in Eq. 13 does not provide a smooth unposed surface causing an offset error in some landmarks. Therefore, we propose an approximation to unpose the landmarks which is accurate and stable during training. Specifically, we compute unposed landmarks $\hat{\mathbf{L}}_t$ as[2]:

$$\hat{\mathbf{L}}_t = \Upsilon^{-1}(\hat{\mathbf{L}}) + \mathbf{T}_t^*(\tilde{\rho}) - \Upsilon^{-1}(\Upsilon(\mathbf{T}_t^*(\tilde{\rho}))), \tag{14}$$

where $\tilde{\rho}$ is the set of indices of the median vertex for each landmark patch and $\mathbf{T}_t^*(\tilde{\rho}) = \{\mathbf{T}_t^{*i} : i \in \tilde{\rho}\}$. In Eq. 14 we update unposed landmarks ($\Upsilon^{-1}(\hat{\mathbf{L}})$) by summing to a correction offset. This offset is computed by the aid of a known reference body ($\mathbf{T}_t^*$ in this case). We apply a nested forward-backward kinematic function to $\mathbf{T}_t^*$ and subtract the results from the reference body. The result in Fig. 3c shows this is a valid approximation improving the unposing procedure.

Training procedure We train the network incrementally. We first train DAE, ATN and Ψ_0 end-to-end by $\mathcal{L}_1$ loss:

$$\mathcal{L}_1 = \lambda_1 \mathcal{L}_{DAE} + \lambda_2 \mathcal{L}_\beta + \lambda_3 \mathcal{L}_\phi + \lambda_4 \mathcal{L}_J + \lambda_5 \mathcal{L}_T + \lambda_6 \mathcal{L}_{unpose}, \tag{15}$$

$$\mathcal{L}_2 = \lambda_2 \mathcal{L}_\beta + \lambda_3 \mathcal{L}_\phi + \lambda_4 \mathcal{L}_J + \lambda_5 \mathcal{L}_T + \lambda_6 \mathcal{L}_{unpose}, \tag{16}$$

where $\{\lambda_i\}_{i=1}^6$ are balancing terms set empirically as 1, 0.1, 1, 0.1, 10 and 2, respectively. We then freeze DAE, ATN and Ψ_0, and train Ψ_1 by $\mathcal{L}_2$ loss. We freeze Ψ_1 and train next cascades likewise.

4 Experiments

In this section, we first describe training details and considered datasets for the experiments. Then, we provide an extensive analysis of the proposed architecture components and loss functions. Finally, we show proof-of-concept real applications of mocap to body surface reconstruction.

[2] We discard $R(\phi)$ for simplicity of reading.

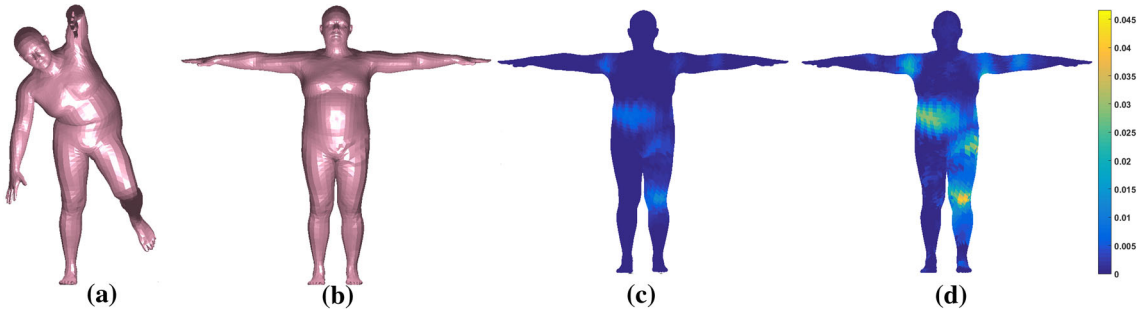

Fig. 3 An example of the proposed inverse kinematic. **a** A given body, **b, c** our proposed unposing and its per vertex error heatmap, and **d** error heatmap for the standard recursive inverse kinematic (Eq. 13)

4.1 Training Details

The code was implemented on Tensorflow and the model was trained on a TITAN Xp GPU. All networks were trained by Adam optimizer with learning rate 0.001 (and default optimizer parameters), from scratch with Xavier initializer, batch size 256 and dropout keeping probability 0.8. The network could converge in less than 6K training steps. Processing time took 1.02s in training for 1 step with batch size 256, and 0.41s and 0.013s in testing for 1 step with batch size 256 and 1, respectively.

4.2 Datasets

SURREAL (Varol et al. 2017) It is composed of 68K videos of rendered humans on top of fixed RGB background. This is a synthetic dataset of humans generated with SMPL model, thus containing exact annotations. We use this dataset for ablation study. The dataset contains millions of frames. In this paper, we randomly subsample 88K and 27K frames from the training and validation set, respectively. This dataset does not provide landmarks. Therefore we create them artificially. We use the 67 landmarks defined in Loper et al. (2014) for this dataset. For each patch ρ_i associated to i-th landmark, we select a random point on the patch surface and move it in the direction of its normal using a random distance in the range of [8..10] mm.

MOSH-SSM (Mahmood et al. 2019) This is a recently published mocap dataset of around 4.5K frames captured from two females. Each frame has an accurate 3D scanned data. In overall, 73 landmarks have been used in the whole dataset (a subset of Loper et al. 2014 landmarks) and in average 10 landmarks are missing in each frame. This dataset has a small variability in pose. We are interested in this dataset due to its availability of synchronized scanned bodies and real-world mocap data challenges.

CMU (Lab 2000) This is a widely used large mocap dataset captured from 96 subjects with more than 1.9K motions. This dataset contains 41 landmarks and the rate of missing landmarks is low.

TCD Hands (Hoyet et al. 2012) We use this dataset to analyze our approach in predicting expressive human body, i.e. body plus hands. There is just one subject in this dataset performing 62 motions. We use 46 standard landmarks on the body plus 8 landmarks on each hand (proposed in Hoyet et al. 2012, i.e. 4 landmarks for thumb and 4 fingertips).

Each standard landmark has an alphabetical code to recognize it. We define a dictionary of landmark codes and their corresponding SMPL vertex indices for each patch. This way we can easily switch between datasets as long as landmark codes follow the standard labels.

4.3 Results

In this section, we study ablative results of our proposed approach on SURREAL validation set. We also show how our method performs on a real world scenario as in MOSH-SSM dataset and we compare to Mahmood et al. (2019) on this dataset. Finally, we show qualitative results on both datasets.

4.3.1 Ablation Study

Our base model is ATN + Ψ_0 trained with $\mathcal{L}_2$. We call this model *DeepMurf*. We then explain the results by adding or removing different components to/from *DeepMurf*. To evaluate on SURREAL dataset, we report average per joint/vertex Euclidean error on $\mathbf{J}_{in}$, $\mathbf{J}_{out}$ and $\mathbf{T}_{out}$ in millimeters. The results are shown in Tables 1 and 2.

Impact of attention model ATN Attention model brings several advantages: (1) an accurate estimation of joints $\mathbf{J}_{in}$ (as in *DeepMurf* with an error of 35.9 mm), and (2) applicability of additional loss functions, i.e. $\mathcal{L}_J$ and $\mathcal{L}_{unpose}$. By omitting $\mathcal{L}_J$ from *DeepMurf* training, one can observe that the surface error is increased by 8 mm (6th row in Table 1). As an additional experiment, we omit ATN, and consequently $\mathcal{L}_J$ and $\mathcal{L}_{unpose}$, from *DeepMurf* and train Ψ_0 directly by feeding landmarks $\hat{\mathbf{L}}$ to the network. As a result, surface

Table 1 Ablation results on SURREAL dataset

Method	$\mathbf{J}_{in}$	$\mathbf{J}_{out}$	$\mathbf{T}_{out}$
Preprocessing + *DeepMurf* + Ψ_1	16.8	19.2	22.7
DeepMurf + Ψ_1 (cascading)	34.7	41.6	47.2
DeepMurf + Ψ_1 − $\mathcal{L}_{unpose}$ (without inverse kinematic loss)	97.6	42.1	47.6
DeepMurf	35.9	56.8	64.6
DeepMurf, $\mathcal{L}_T$ with hard assignment	34.4	61.7	71.1
DeepMurf − $\mathcal{L}_J$	44.3	65.3	72.7
DeepMurf − $\mathcal{L}_\phi$	42.6	63.1	77.3
DeepMurf − $\mathcal{L}_\beta$	44.8	70.1	80.4
$\Psi_0 + \mathcal{L}_2 - \mathcal{L}_J - \mathcal{L}_{unpose}$ (without attention model)	–	85.3	92.9
DeepMurf (trained on 256 samples)	50.8	102.9	121.5
DeepMurf − $\mathcal{L}_{unpose}$ (trained on 256 samples)	127.4	121.7	136.5

The errors are in millimeters. *DeepMurf* = ATN + $\Psi_0 + \mathcal{L}_2$

Table 2 The impact of training with missing landmarks on SURREAL validation set

Method	τ	DAE	$\mathbf{J}_{in}$	$\mathbf{J}_{out}$	$\mathbf{T}_{out}$
Preprocessing + *DeepMurf* + Ψ_1	0	–	16.8	19.2	22.7
Preprocessing + DAE + *DeepMurf* + $\Psi_1 + \mathcal{L}_{DAE}$	0.1	8.1	23.7	28.2	33.2
	0.3	7.8	33.3	33.6	41.5
	0.5	6.8	37.4	39.4	45.7
DeepMurf	0	–	35.9	56.8	64.6
DAE + *DeepMurf* + $\mathcal{L}_{DAE}$	0.1	11.2	48.4	59.7	68.3
	0.3	11.9	69.7	77.1	87.5
	0.5	12.4	85.3	87.5	100.4

τ is the rate of missing landmarks

error is increased by more than 28 mm (9th row in Table 1). This shows that the proposed ATN has a huge impact on the results and *DeepMurf* without ATN is not able to properly learn useful information just from landmarks to map them to the pose and shape parameters.

Impact of regularization loss on pose and shape parameters We omit $\mathcal{L}_\phi$ or $\mathcal{L}_\beta$ from $\mathcal{L}_2$ loss and train *DeepMurf*. As a result (7th and 8th rows in Table 1), the error is increased by around 13 mm and 16 mm for $\mathcal{L}_\phi$ and $\mathcal{L}_\beta$, respectively. This is mainly due to the sensitivity of SMPL to the noise and convergence to invalid parameters in the backpropagation. As one can see, omitting $\mathcal{L}_\beta$ has more impact on the results than $\mathcal{L}_\phi$.

Soft versus hard landmark-to-surface assignment *DeepMurf* has a surface error of 64.6 mm. When *DeepMurf* is trained with a hard landmark-to-surface assignment in $\mathcal{L}_T$, the surface error is increased by more than 6 mm (5th row in Table 1). A hard assignment means each landmark is always associated with a fixed vertex on the SMPL surface. A hard assignment introduces some noise in the loss and does not lead to the most optimum solution. We also applied a chamfer distance as $\mathcal{L}_T$. However, it had a high complexity and did not converge well. These results reveal that our soft assignment works well in practice and can cope with the challenges in the data.

Impact of cascading In our proposed cascading, each block learns the error of the previous block. As one can see (2nd row in Table 1), an additional cascading block Ψ_1 to *DeepMurf* improves the surface error by more than 17 mm. We have observed more cascading blocks were not as effective as Ψ_1 improving the error around 1 mm. We note that *incrementally* training cascading approach is important for performance gains. Training cascading network end-to-end from scratch performs similar to *DeepMurf*. We show some qualitative images of cascading model on SURREAL dataset in Fig. 4.

Impact of inverse kinematic loss We study inverse kinematic loss $\mathcal{L}_{unpose}$ in two ways. Firstly, we omit $\mathcal{L}_{unpose}$ from *DeepMurf* + Ψ_1 network[3]. As it is visible in Table 1 (3rd row), $\mathcal{L}_{unpose}$ has a huge impact on $\mathbf{J}_{in}$. Omitting $\mathcal{L}_{unpose}$ from cascading network increases $\mathbf{J}_{in}$ error by around 63 mm reducing performance of ATN. However, $\mathcal{L}_{unpose}$ does not have much impact on the surface error. Secondly, we omit $\mathcal{L}_{unpose}$ from *DeepMurf* and train the model for 500 epochs on a very small dataset (256 samples). We want to study

[3] Note that *DeepMurf* − $\mathcal{L}_{unpose}$ behaves similarly.

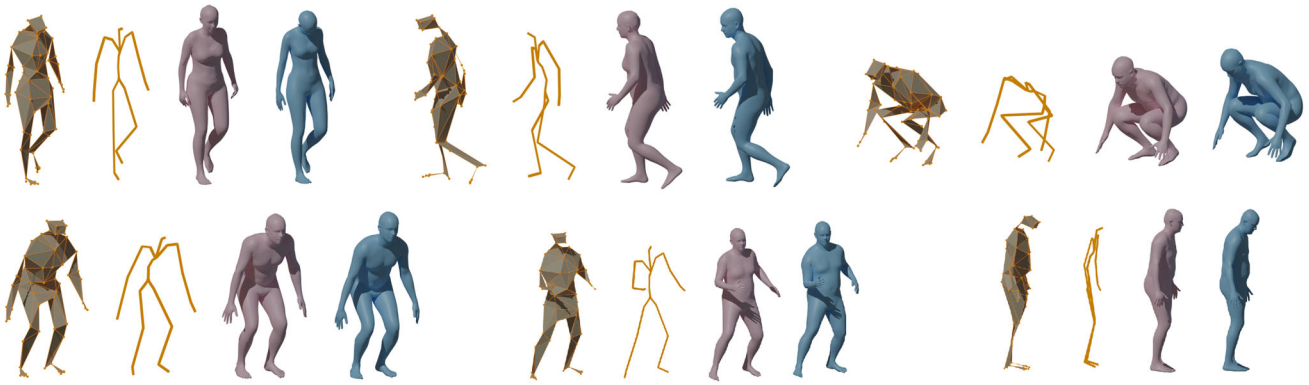

Fig. 4 This figure shows the different stages of the cascading model for some samples of SURREAL dataset. First, we see the input landmarks. Next, the estimated joints obtained through ATN. Finally, the estimated 3D human model (pink) along the ground truth (blue) (Color figure online)

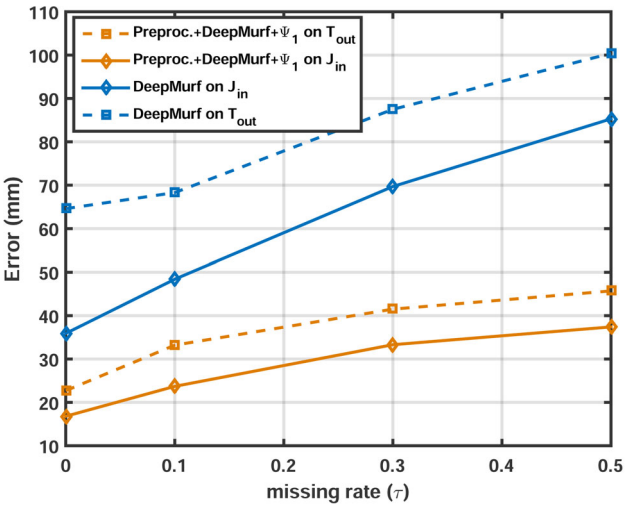

Fig. 5 The impact of training with missing landmarks on SURREAL validation set

generalization ability of the proposed loss on small custom datasets. As a result (can be seen in the last two rows), $\mathcal{L}_{unpose}$ helps to gain 15 mm improvement on the surface error. Interestingly, *DeepMurf* can still perform well to estimate $\mathbf{J}_{in}$ (error of 50.8 mm) trained on such a small dataset.

Impact of global orientation Data normalization and augmentation are two common preprocessing techniques applied in deep learning to boost performance. In this paper we mainly focus on the data normalization. In the previous experiments, we applied a translation invariant solution by subtracting the input landmarks from the mean point. Here, we explore an additional preprocessing to make the network rotation invariant. To do so, we rigidly (without scaling) align all landmarks in the dataset to a reference set of landmarks, e.g. the template landmarks. More specifically, we apply procrustes analysis to compute a rotation matrix and translation vector to transform landmarks $\hat{\mathbf{L}}$. We then train the cascading network on the aligned data as before. At test time, estimated surface is transformed back to the original orientation. The results are shown in the first row of Table 1. As it can be seen in the comparison of the first two rows, the surface error is reduced by 48% (24.5 mm).

Impact of missing landmarks To study the impact of missing landmarks, we train DAE + *DeepMurf* and Preprocessing + DAE + *DeepMurf* + Ψ_1 (including $\mathcal{L}_{DAE}$ in the training) with different rates of missing landmarks, that is, we randomly select 90%, 70% and 50% of the landmarks in the batch and assign zero to the rest. This is repeated for each step. This means in average 7, 20 and 33 landmarks are dropped for each frame in each setup. The results can be seen in Table 2. Interestingly, DAE error on missing landmarks (3rd column) is not strongly correlated to the rate of missing landmarks (τ). This error is around 7.5 mm and 11.8 mm in average for the cascading and baseline models, respectively. Also, the error on the surface ($\mathbf{T}_{out}$) and the input joints ($\mathbf{J}_{in}$) is polynomial with a degree below 1 w.r.t. the τ. This means the error will not increase much in higher rates of missing landmarks. This can be seen in Fig. 5. We note that the model trained with preprocessing is more resistant against the missing landmarks than the default model *DeepMurf*. These results show that DAE is effective against missing landmarks to be used in real world applications of the proposed surface recovery.

Qualitative comparison We qualitatively compare different methods in the ablation study in Fig. 6. As expected, according to Table 1, the attention model has a high impact on the quality of the results. Interestingly, body shape has more impact on the error than pose, e.g. in the extreme shapes. Finally, by applying the preprocessing, we can generate a near perfect body surface.

4.3.2 MOSH-SSM Results

This dataset does not have any split regarding training-testing set. Therefore, we randomly split the data in 50/50% ratio and train our cascading model DAE + *DeepMurf* + Ψ_1 (includ-

Fig. 6 Qualitative ablation results on SURREAL validation dataset. Connections are added to the landmarks for visualization purposes

ing preprocessing). To evaluate our approach on this dataset, we compute average scan-to-model distance as in Mahmood et al. (2019). To do so, for each frame we randomly sample 10K points from the ground truth scan and for each point take its nearest neighbor vertex on the estimated $\mathbf{T}_{out}$. Finally, per point Euclidean distances are averaged over the whole dataset. The scans and landmarks are not very well aligned in this dataset. Therefore, to apply scan-to-model distance we first align predictions with scans by fitting a translation vector through CPD algorithm (Myronenko and Song 2010).

We evaluate three different models on this dataset and compare with Mosh++ (Mahmood et al. 2019) in Table 3. In the second row, we train the model with a set of 46 standard landmarks. This model has the highest error among others (24.5 mm) due to a reduced set of landmarks. However, it still performs well. In the next row in the table, we train the net-

Table 3 Quantitative results on Mosh-SSM dataset

Method	Scan-to-model error (mm)
Mosh++ (Mahmood et al. 2019)	18.1
Best *DeepMurf* (46 landmarks)	24.5
Best *DeepMurf* (67 landmarks)	19.9
Best *DeepMurf* (67 landmarks) + Temporal smoothing	19.8

Fig. 7 Qualitative comparison with Mosh++ (Mahmood et al. 2019) on MOSH-SSM dataset. Left: input landmarks, middle: our predictions with 67 landmarks, and right: Mosh++ predictions

work with 67 landmarks (as in Mahmood et al. 2019) which shows more than 21% improvement against 46 landmarks. This is while the difference error with Mosh++ is 1.8 mm. This model is trained and tested on single frames and temporal smoothing is not applied. In the next experiment we apply temporal smoothing as a post-processing. To do so, we set shape parameters as the average over the whole sequence. Regarding the pose parameters, we check for jittering based on angle difference between previous and next frames for each joint and axis. We empirically select threshold 0.1 for this task. As a result, the error is improved by 0.1 mm which shows predictions on single frames are temporally consistent. Finally, we show qualitative results in Fig. 7.

4.3.3 CMU Results

We analyze our approach on CMU dataset qualitatively in Fig. 8. As one can see, we predict similar surfaces to Mosh++ on this dataset with variable motion and subjects. We achieve this performance in few miliseconds vs several minutes of Mosh++. Since we train a neutral SMPL body, without specifically defining gender, our model converges mostly to a female body using standard landmarks. Similar to Mosh++, one can train gender specific SMPL models when gender specific landmarks are not available.

4.3.4 TCD Hands Results

In this dataset, we analyze the applicability of our approach for the case of expressive humans, specifically body and hands. Handling body and hand pose together in a single network is a challenging task due to unbalanced landmarks, different motion and level of details between body and hands. To do so, we update *DeepMurf* with SMPLH model (Mahmood et al. 2019). We train the network by setting high weights on lower arm joints and landmarks in $\mathcal{L}_\phi$ and $\mathcal{L}_T$, respectively. We train the network with the same capacity as before. The results can be seen in Fig. 9. As it can be seen, our approach shows promising results being applicable for expressive humans.

4.4 Applications

Due to the popularity of mocap data in movie editing and videogame industry, a fast, accurate simulation and rendering can save a lot of working hours of animators. Furthermore,

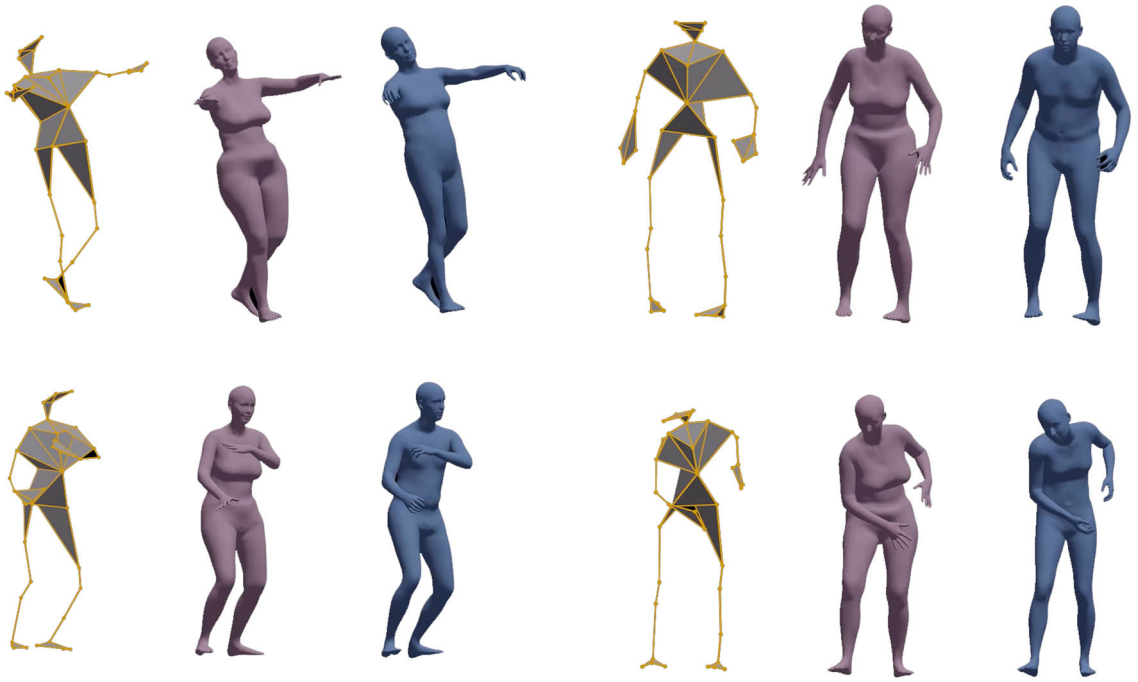

Fig. 8 Qualitative comparison with Mosh++ (Mahmood et al. 2019) on CMU dataset. Left: input landmarks, middle: our predictions, and right: Mosh++ predictions

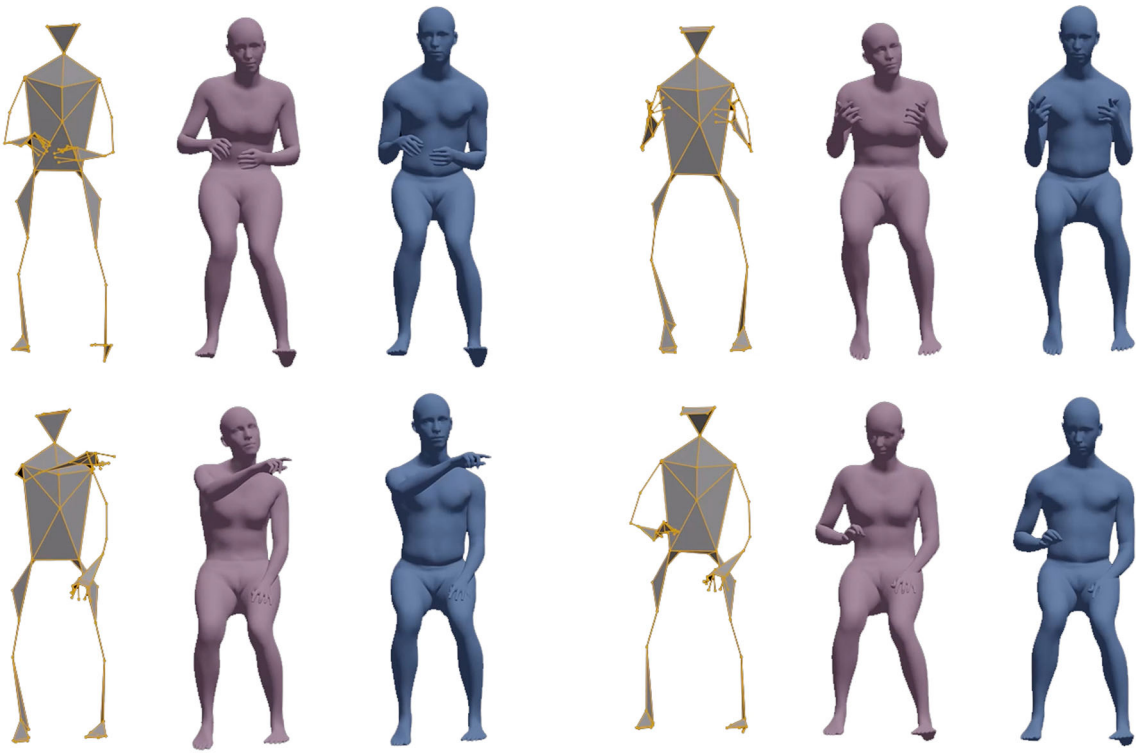

Fig. 9 Qualitative comparison with Mosh++ (Mahmood et al. 2019) on TCD Hands dataset. Left: input landmarks, middle: our predictions, and right: Mosh++ predictions

the deep model can be embedded into graphic engines for specific applications. In this section, we propose different applications for our deep-based mocap-to-surface estimation, shown in Fig. 1. A basic application can be shape or pose modification. We also perform garment simulation on top of the reconstructed body. Furthermore, we apply retargeting to a bunny avatar. This can be done by replacing rigged SMPL template by any other rigged template consistent with SMPL inline functionality or replacing SMPL with any other generative model. All of this is done by just inputing a set of sparse landmarks to the application.

4.5 Results in a Sequence

In the supplementary video, we also show the results of our best model in a video along with this document. We show four example sequences from each dataset in the order of CMU, MOSH-SSM and TCD Hands. Note that the results are shown without temporal smoothing.

5 Conclusions

We presented a deep unsupervised approach for estimation of body surface from sparse mocap data. We applied a denoising autoencoder network able to recover missing landmarks accurately. Our proposed attention model estimated body joints from landmarks and we showed it has a high impact on the accuracy of the generated surface. Attention model also allowed us to apply several loss functions improving the model performance including an unposing layer useful to learn body geometry. We also designed a cascading regression which helped to improve the error by 17 mm. Our quantitative and qualitative results on four datasets show applicability of our approach in real world problems (including expressive humans) with a surface error less than 20 mm.

Although we showed promising results for expressive humans, there is still room for improvement. Also, we did not model soft-tissue on *DeepMurf*, which can be an important source of landmarks noise. Body soft-tissue dynamics can be modeled from body pose and shape. We will explore these ideas as the future work.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11263-021-01488-2.

Funding This work is partially supported by ICREA under the ICREA Academia programme, and by the Spanish project PID2019-105093GB-I00 (MINECO/FEDER, UE) and CERCA Programme/Generalitat de Catalunya, and by Amazon Research Awards ARA.

Availability of data and material The authors have used public datasets in this work.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

Achilles, F., Ichim, A.E., Coskun, H., Tombari, F., Noachtar, S., & Navab, N. (2016). Patient mocap: Human pose estimation under blanket occlusion for hospital monitoring applications. In *International conference on medical image computing and computer-assisted intervention* (pp. 491–499). Springer.

Bhatnagar, B.L., Sminchisescu, C., Theobalt, C., & Pons-Moll, G. (2020). Combining implicit function learning and parametric models for 3D human reconstruction. In *ECCV*

Bhatnagar, B. L., Sminchisescu, C., Theobalt, C., & Pons-Moll, G. (2020). LoopReg: Self-supervised learning of implicit surface correspondences, pose and shape for 3D human mesh registration. In *Neural information processing systems (NeurIPS)*.

Bogo, F., Black, M. J., Loper, M., & Romero, J. (2015). Detailed full-body reconstructions of moving people from monocular RGB-D sequences. In *Proceedings of the IEEE international conference on computer vision* (pp. 2300–2308).

Chai, J., & Hodgins, J. K. (2005). Performance animation from low-dimensional control signals. In *ACM SIGGRAPH 2005 papers* (pp. 686–696).

Chen, C. H., Tyagi, A., Agrawal, A., Drover, D., Stojanov, S., & Rehg, J. M. (2019). Unsupervised 3D pose estimation with geometric self-supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5714–5724).

Groueix, T., Fisher, M., Kim, V. G., Russell, B., & Aubry, M. (2018). 3D-coded: 3D correspondences by deep deformation. In *ECCV*.

Hoyet, L., Ryall, K., McDonnell, R., & O'Sullivan, C. (2012). Sleight of hand: Perception of finger motion from reduced marker sets. In *Proceedings of the ACM SIGGRAPH symposium on interactive 3D graphics and games* (pp. 79–86).

Huang, Y., Kaufmann, M., Aksan, E., Black, M. J., Hilliges, O., & Pons-Moll, G. (2018). Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Transactions on Graphics (TOG)*, *37*(6), 1–15.

Insafutdinov, E., & Dosovitskiy, A. (2018). Unsupervised learning of shape and pose with differentiable point clouds. In *Advances in neural information processing systems* (pp. 2802–2812).

Joo, H., Simon, T., & Sheikh, Y. (2018). Total capture: A 3D deformation model for tracking faces, hands, and bodies. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8320–8329).

Kanazawa, A., Black, M. J., Jacobs, D. W., & Malik, J. (2018). End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7122–7131).

Kudo, Y., Ogaki, K., Matsui, Y., & Odagiri, Y. (2018). Unsupervised adversarial learning of 3D human pose from 2D joint locations. Preprint retrieved from arXiv:1803.08244

Lab, C. G. (2000). CMU graphics lab motion capture. http://mocap.cs.cmu.edu

Liu, H., Wei, X., Chai, J., Ha, I., & Rhee, T. (2011). Realtime human motion control with a small number of inertial sensors. In *Symposium on interactive 3D graphics and games* (pp. 133–140).

Loper, M., Mahmood, N., & Black, M. J. (2014). Mosh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics (TOG)*, *33*(6), 1–13.

Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., & Black, M. J. (2015). SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (TOG)*, *34*(6), 1–16.

Madadi, M., Bertiche, H., & Escalera, S. (2018). SMPLR: Deep SMPL reverse for 3D human pose and shape recovery. Preprint retrieved from arXiv:1812.10766.

Mahmood, N., Ghorbani, N., Troje, N. F., Pons-Moll, G., & Black, M. J. (2019). AMASS: Archive of motion capture as surface shapes. In *Proceedings of the IEEE international conference on computer vision* (pp. 5442–5451).

Mehrizi, R., Peng, X., Tang, Z., Xu, X., Metaxas, D., & Li, K. (2018). Toward marker-free 3D pose estimation in lifting: A deep multi-view solution. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)* (pp. 485–491). IEEE.

Mehta, D., Sotnychenko, O., Mueller, F., Xu, W., Elgharib, M., Fua, P., Seidel, H. P., Rhodin, H., Pons-Moll, G., & Theobalt, C. (2020). XNect: Real-time multi-person 3D motion capture with a single RGB camera. Vol. 39. https://doi.org/10.1145/3386569.3392410

Myronenko, A., & Song, X. (2010). Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(12), 2262–2275.

Omran, M., Lassner, C., Pons-Moll, G., Gehler, P., & Schiele, B. (2018). Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *2018 international conference on 3D vision (3DV)* (pp. 484–494). IEEE.

Park, S. I., & Hodgins, J. K. (2006). Capturing and animating skin deformation in human motion. *ACM Transactions on Graphics (TOG)*, *25*(3), 881–889.

Pavlakos, G., Zhu, L., Zhou, X., & Daniilidis, K. (2018). Learning to estimate 3D human pose and shape from a single color image. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 459–468).

Prokudin, S., Lassner, C., & Romero, J. (2019). Efficient learning on point clouds with basis point sets. In *Proceedings of the IEEE international conference on computer vision* (pp. 4332–4341).

Rhodin, H., Robertini, N., Casas, D., Richardt, C., Seidel, H. P., & Theobalt, C. (2016). General automatic human shape and motion capture using volumetric contour cues. In *European conference on computer vision* (pp. 509–526). Springer.

Schwarz, L. A., Mateus, D., & Navab, N. (2009). Discriminative human full-body pose estimation from wearable inertial sensor data. In *3D physiological human workshop* (pp. 159–172). Springer.

Slyper, R., & Hodgins, J. K. (2008). Action capture with accelerometers. In *Symposium on computer animation* (pp. 193–199).

Tautges, J., Zinke, A., Krüger, B., Baumann, J., Weber, A., Helten, T., et al. (2011). Motion reconstruction using sparse accelerometer data. *ACM Transactions on Graphics (ToG)*, *30*(3), 1–12.

Trumble, M., Gilbert, A., Malleson, C., Hilton, A., & Collomosse, J. (2017). Total capture: 3D human pose estimation fusing video and inertial sensors. In *BMVC* (Vol. 2, p. 3).

Varol, G., Ceylan, D., Russell, B., Yang, J., Yumer, E., Laptev, I., & Schmid, C. (2018). Bodynet: Volumetric inference of 3D human body shapes. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 20–36).

Varol, G., Romero, J., Martin, X., Mahmood, N., Black, M. J., Laptev, I., & Schmid, C. (2017). Learning from synthetic humans. In *CVPR*

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P. A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, *11*, 3371–3408.

von Marcard, T., Henschel, R., Black, M. J., Rosenhahn, B., & Pons-Moll, G. (2018). Recovering accurate 3D human pose in the wild using IMUs and a moving camera. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 601–617).

von Marcard, T., Rosenhahn, B., Black, M. J., & Pons-Moll, G. (2017). Sparse inertial poser: Automatic 3D human pose estimation from sparse IMUs. In *Computer graphics forum* (Vol. 36, pp. 349–360). Wiley Online Library.

Von Marcard, T., Pons-Moll, G., & Rosenhahn, B. (2016). Human pose estimation from video and IMUs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(8), 1533–1547.

Zhao, W., Chai, J., & Xu, Y.Q. (2012). Combining marker-based mocap and RGB-D camera for acquiring high-fidelity hand motion data. In *Proceedings of the ACM SIGGRAPH/eurographics symposium on computer animation* (pp. 33–42). Eurographics Association