Generalizing Neural Human Fitting to Unseen Poses With Articulated SE(3) Equivariance
Supplementary Material

Haiwen Feng[1] Peter Kulits[1] Shichen Liu[2] Michael J. Black[1] Victoria Abrevaya[1]
[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]University of Southern California
{hfeng,kulits,black,vabrevaya}@tuebingen.mpg.de, {liushich}@usc.edu

A. Additional Results

In this section we provide additional results, as well as ablation studies that demonstrate the impact of our design choices.

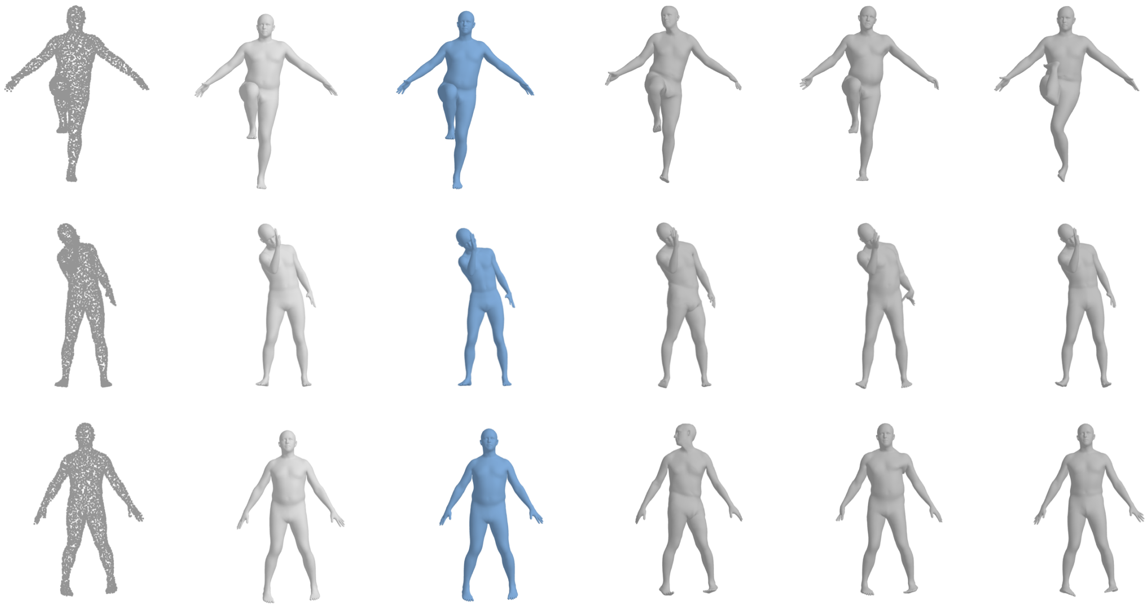
Qualitative Registration Results. Figure S.2 shows qualitative results for out-of-distribution testing data, and Figure S.1 shows results for the in-distribution case. IP-Net, PTF, and LoopReg all fail under difficult poses, resulting in unnatural rotations for some body parts. Poses that are particularly far from the distribution, such as standing on the arms, result in very unnatural shapes. In contrast, our method can handle such poses well despite never having seen similar ones during training. It is worth noting that LoopReg uses the same self-supervised objective during training and during optimization, and refines the learned correspondences at test time by overfitting to the input. Hence, we see here that even such a test-time optimization strategy is not sufficient when the initial poses are far from the correct result.

Qualitative Segmentation Results. In Figure S.3 we show additional results for part segmentation on out-of-distribution data, along with comparisons to the segmentations obtained by IP-Net, PTF, and LoopReg. Note that IP-Net and LoopReg predict part segmentation for 14 body parts, where, for example, the two shoulder blades, the three spine regions and the hip are all merged into one torso part (here, in red), or the neck is merged into the head region (here, in olive), making it an easier problem. Our method produces accurate segmentations even for these difficult OOD cases, while PTF, IP-Net, and LoopReg struggle to predict the segmentation, particularly in the regions with out-of-distribution pose. For example, in the second row, IP-Net's part segmentation confuses left and right, resulting in a flipped torso with the belly facing up.

Raw Scan Data. We evaluated our method on the raw scans from the DFaust testing set (in-distribution), without any fine-tuning or re-training. Our model obtains 88.3% accuracy for part segmentation, 3.62cm vertex-to-vertex error, and 4.37cm MPJPE error, which is still better than most other methods on clean data. A qualitative example of these results is shown in Figure S.4. Here we see that our estimations are still accurate for out-of-distribution poses, despite the out-of-distribution noise.

Impact of the Number of Input Points. We show in Table S.1 the results of our model when the input is 500, 1000, 2500, and 5000 points. We see here that our method can already perform reasonably well for in-distribution data for 1000 input points, with a segmentation accuracy that is on par with competitors that use 5000 points as input (91.2% for IP-Net, Table 1 in main paper). The segmentation accuracy does not differ much when moving to the OOD case. The model has lower performance in terms of V2V and MPJPE for a lower number of points on OOD data, however it still outperforms all the competitors (Table 2 in main paper). This shows that our model does not require a significant number of points in order to obtain accurate results, both for in- and out-of-distribution data.

Baselines Without a Pose Prior. PTF and IP-Net use a pose prior to regularize the pose space when fitting to SMPL. In the main paper we tested these methods with default parameters, which include the use of the pose prior. To make sure that this does not negatively affect the final outcome, we evaluate PTF and IP-Net without the pose prior. The results are shown in Table S.2, where we observe that the pose prior does not have a substantial effect on the output.



(a) Input (b) Ground truth (c) ArtEq (Ours) (d) IP-Net (e) PTF (f) LoopReg

Figure S.1: Qualitative results for in-distribution poses. From left to right: (a) input point cloud, (b) ground-truth SMPL mesh, (c) our results, (d) IP-Net [1], (e) PTF [5] and (f) LoopReg [2].

# points	OOD			ID		
	Seg. ↑	V2V ↓	MPJPE ↓	Seg.	V2V ↓	MPJPE ↓
500	80.5	7.33	8.63	82.9	4.55	5.27
1000	89.7	4.85	5.83	92.1	2.27	2.80
2500	93.0	4.09	4.59	95.4	1.01	1.22
5000	94.1	3.62	4.23	96.2	0.98	1.26

Table S.1: Our results for different numbers of input points, in terms of segmentation accuracy (“Seg.”), vertex-to-vertex error (“V2V”), and mean joint position error (“MPJPE”).

B. Permutation Equivariance of the Self-Attention Mechanism

As we have mentioned in the main paper, a function (network) $f : V \to W$ is said to be equivariant with respect to a group $\mathcal{G}$ if, for any transformation $\mathcal{T} \in \mathcal{G}, f(\mathcal{T}\boldsymbol{X}) = \mathcal{T}f(\boldsymbol{X})$, $\boldsymbol{X} \in V$. Here we elaborate on how the self-attention function f_{SA} is equivariant to the permutation group $\mathcal{T}(\boldsymbol{X}) = \boldsymbol{X}P_\pi$, where P_π denotes the permutation matrix of π, and π denotes the permutation of the input tensor's elements (in our case, the permutation over the group element dimension). The self-attention function f_{SA} is defined as $f_{SA}(\mathbf{X}) = \boldsymbol{W}_v\boldsymbol{X} \cdot \mathrm{softmax}\left((\boldsymbol{W}_k\boldsymbol{X})^T \cdot \boldsymbol{W}_q\boldsymbol{X}\right)$, then

$$
\begin{aligned}
& f_{SA}\left(\mathcal{T}(\boldsymbol{X})\right) \\
&= \boldsymbol{W}_v\mathcal{T}(\boldsymbol{X}) \cdot \mathrm{softmax}\left((\boldsymbol{W}_k\mathcal{T}(\boldsymbol{X}))^T \cdot \boldsymbol{W}_q\mathcal{T}(\boldsymbol{X})\right) \\
&= \boldsymbol{W}_v\boldsymbol{X}P_\pi \cdot \mathrm{softmax}\left((\boldsymbol{W}_k\boldsymbol{X}P_\pi)^T \cdot \boldsymbol{W}_q\boldsymbol{X}P_\pi\right) \\
&= \boldsymbol{W}_v\boldsymbol{X}P_\pi \cdot \mathrm{softmax}\left(P_\pi^T\,(\boldsymbol{W}_k\boldsymbol{X})^T \cdot \boldsymbol{W}_q\boldsymbol{X}P_\pi\right) \\
&= \boldsymbol{W}_v\boldsymbol{X}\left(P_\pi P_\pi^T\right) \cdot \mathrm{softmax}\left((\boldsymbol{W}_k\boldsymbol{X})^T \cdot \boldsymbol{W}_q\boldsymbol{X}\right)P_\pi \\
&= \boldsymbol{W}_v\boldsymbol{X} \cdot \mathrm{softmax}\left((\boldsymbol{W}_k\boldsymbol{X})^T \cdot \boldsymbol{W}_q\boldsymbol{X}\right)P_\pi \\
&= \mathcal{T}\left(f_{SA}(\boldsymbol{X})\right),
\end{aligned}
\tag{S.1}
$$

where we used the property $\mathrm{softmax}(P \cdot A \cdot P^T) = P \cdot \mathrm{softmax}(A) \cdot P^T$, for a permutation matrix P and an arbitrary matrix A, to go from the third to the fourth line (refer to the proof in [6]). Hence, we have proved that the self-

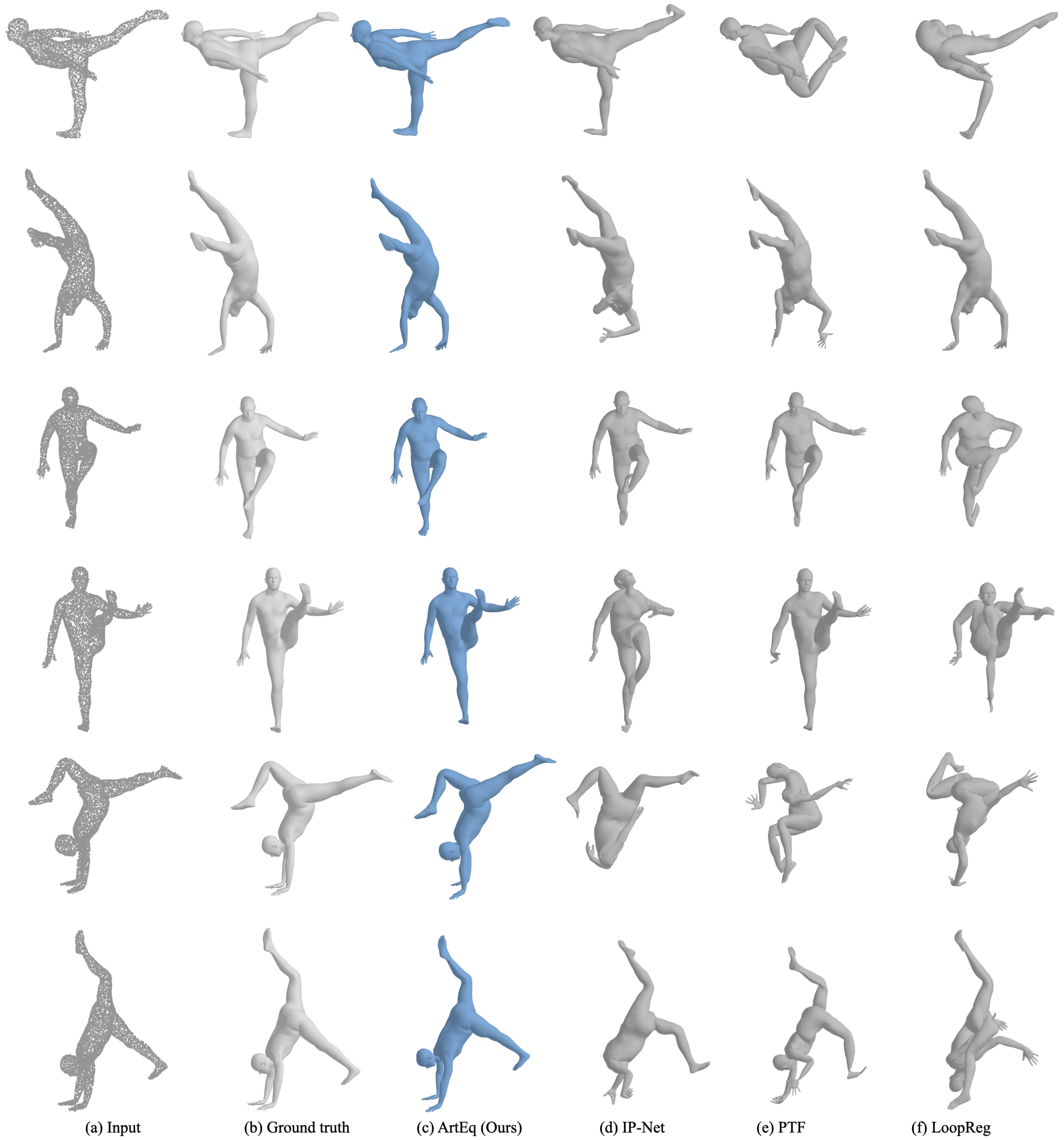
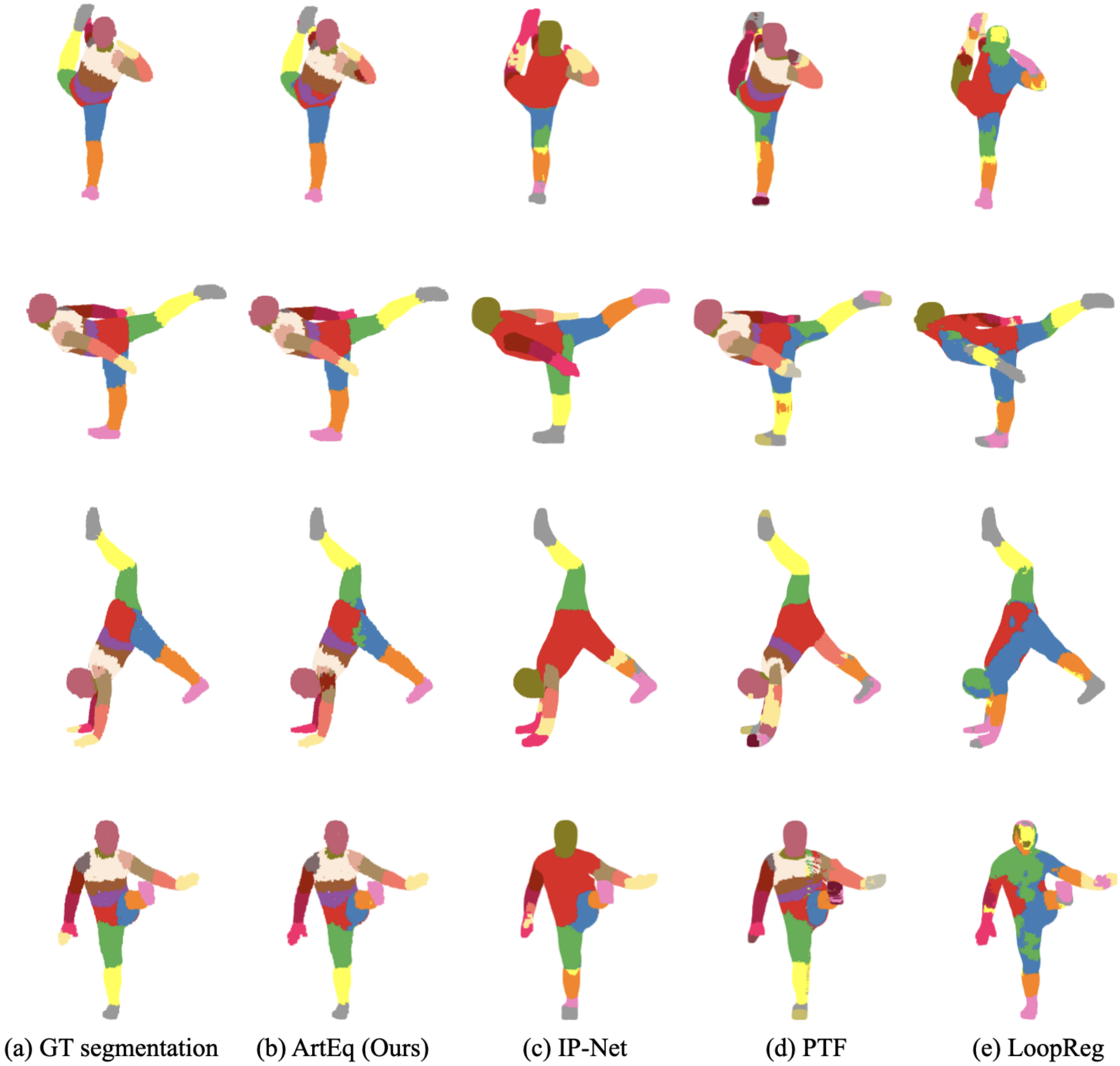


Figure S.2: Qualitative results for out-of-distribution poses. From left to right: (a) input point cloud, (b) ground-truth SMPL mesh, (c) our results, (d) IP-Net [1], (e) PTF [5] and (f) LoopReg [2].

attention function is equivariant to the permutation operation over the discretized SO(3) group elements.



(a) GT segmentation (b) ArtEq (Ours) (c) IP-Net (d) PTF (e) LoopReg

Figure S.3: Qualitative results for part segmentation. From left to right: (a) ground-truth segmentation, (b) our results, (c) IP-Net [1], (d) PTF [5], and (e) LoopReg [2].

Method	Pose Prior	OOD		ID	
		V2V ↓	MPJPE ↓	V2V ↓	MPJPE ↓
IP-Net	✓	7.57	9.41	5.98	6.42
IP-Net		7.67	9.55	6.04	6.50
PTF	✓	6.42	7.56	3.05	3.53
PTF		6.46	7.62	3.13	3.66
Ours		**3.62**	**4.23**	**0.98**	**1.26**

Table S.2: Comparison with IP-Net and PTF with and without using their pose prior.

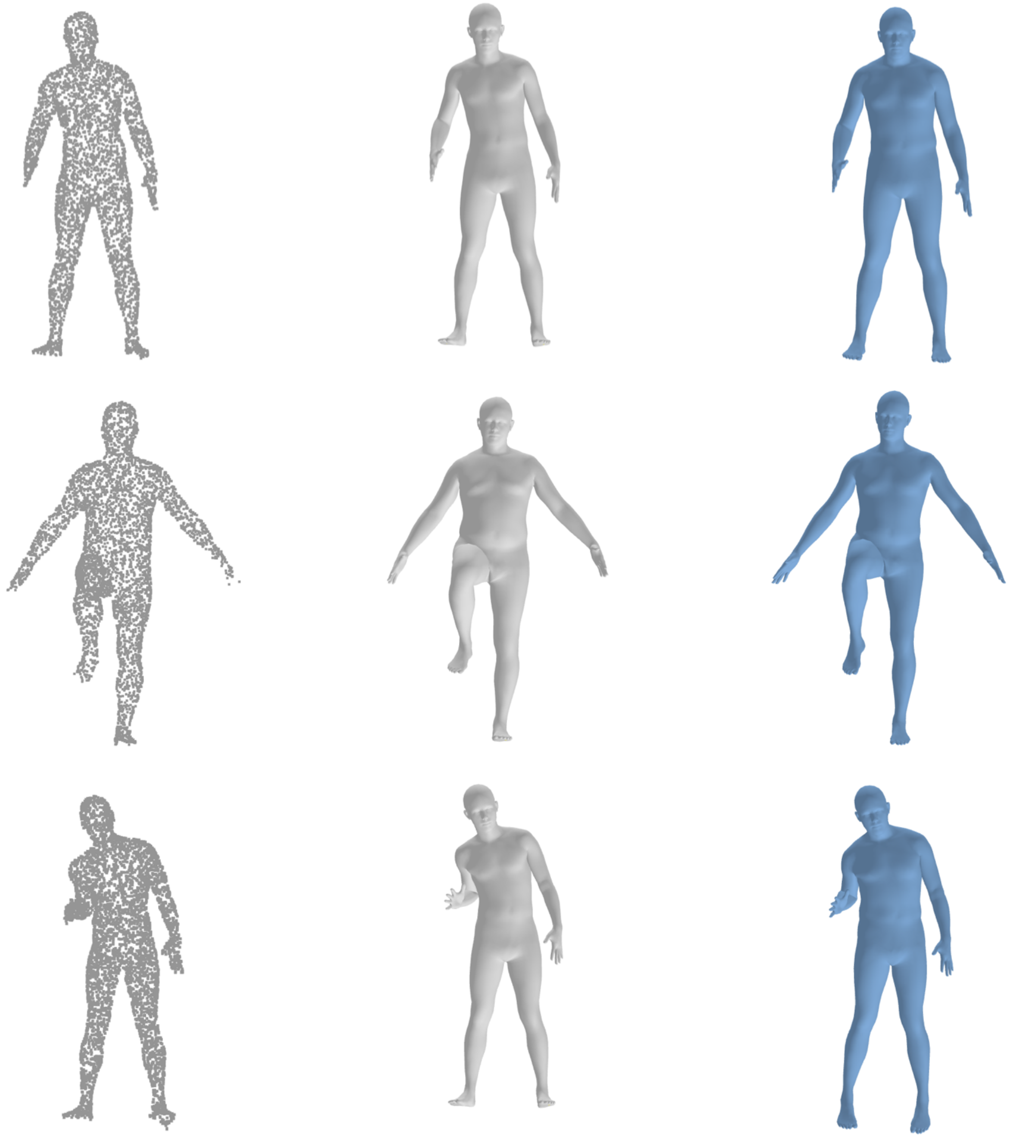


Figure S.4: Qualitative results on the raw scans from DFAUST testing set. From left to right: (a) input point cloud, (b) ground-truth SMPL mesh, (c) our results.

C. Method Details

Architecture. The local SO(3) feature extractor has two SPConv layers and a nearest neighbor feature propagation layer [4]. Each SPConv layer has a kernel size of 0.4 and a stride downsampling factor of 2, therefore, the input point cloud with shape [B, N, 3] will be processed as [B, N/4, 64, 60] where the last dimension is the group element obtained by SO(3) discretization, and $C = 64$ is the feature dimension. For each input point, the feature propagation layer finds the top 3 spatial nearest neighbors of the downsampled point-wise features, and interpolates these features weighed by their pairwise distance, resulting in an output of size [B, N, 64, 60].

To obtain the chordal mean weights we attach to the self-attention layers an element-wise MLP (3 layers with ReLU, sizes [64,64,1]), since self-attention does not contain non-linear activations. Similarly, we attach a 2-layer MLP on the flattened part features [B, 20*6] to obtain the final SMPL shape code.

Part Segmentation. We consider here 20 body parts, merging the fingers into hands, and toes into feet. This is because the AMASS DFAUST dataset does not contain finger or toe motion.

Averaging Rotations by Calculating the Chordal L2 Mean. Given two rotations R and S, the chordal L2 distance is defined as $d_{chord}(R,S) = \|R-S\|_F$ where $\|\cdot\|_F$ is the Frobenius norm of the matrix, which is related to the angular distance between R and S [3]. The chordal L2 mean of a set of rotations is then defined as the matrix that minimizes the chordal distance to all rotations in the set. In our case, if $\mathbf{w}_{k,j}$ is the weight for part k and group j, then the weighted average for part k over the $|\mathcal{G}| = 60$ rotation symmetries is

$$\underset{\hat{\mathcal{R}}_k \in SO(3)}{\arg\min} \sum_{j=1}^{|\mathcal{G}|} d_{chord}(\mathbf{w}_{k,j} \cdot \mathcal{R}(\mathbf{g}_j), \hat{\mathcal{R}}_k) \tag{S.2}$$

where $\mathcal{R}(\mathbf{g}_j)$ is the rotation matrix of $\mathbf{g}_j$, and $\mathbf{g}_j$ is a group element. In practice, $\hat{\mathcal{R}}_k$ can be obtained in closed-form by using singular value decomposition. We refer the readers to [3] for more details.

Loss Function. We train both stages of the network with the following loss function:

$$\lambda_1 \mathcal{L}_{pose} + \lambda_2 \mathcal{L}_{shape} + \lambda_3 \mathcal{L}_{verts} + \lambda_4 \mathcal{L}_{joint} + \lambda_5 \mathcal{L}_{part}, \tag{S.3}$$

where

- $\mathcal{L}_{pose} = ||\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}||^2$ is the MSE loss between predicted pose coefficients $\tilde{\boldsymbol{\theta}}$ and ground-truth pose coefficients $\boldsymbol{\theta}$.
- $\mathcal{L}_{shape} = ||\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}||^2$ is the MSE loss between predicted shape coefficients $\tilde{\boldsymbol{\beta}}$ and ground-truth shape coefficients $\boldsymbol{\beta}$.
- $\mathcal{L}_{verts} = ||\mathcal{W}\left(M(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\theta}}) - M(\boldsymbol{\beta}, \boldsymbol{\theta})\right)||^2$ is the weighted MSE loss between the reconstructed SMPL mesh vertices and the ground-truth registration, using the per-vertex weights $\mathcal{W}$, where the vertices corresponding to body markers are assigned a weight of 2.0, and the other vertices a weight of 1.0.
- $\mathcal{L}_{joint} = ||\mathcal{T}(\mathcal{J}(\tilde{\boldsymbol{\beta}}), \tilde{\boldsymbol{\theta}}) - \mathcal{T}(\mathcal{J}(\boldsymbol{\beta}), \boldsymbol{\theta})||^2$ is the MSE loss between the predicted joint positions of the SMPL mesh (posed) and the ground-truth joint positions.
- $\mathcal{L}_{part} = \text{cross-entropy}(\alpha(\mathbf{x}_i, \mathbf{p}_k), \alpha_{gt}(\mathbf{x}_i, \mathbf{p}_k))$ is the cross-entropy loss between the predicted part segmentation and the ground-truth part segmentation of the point cloud.

We use $\lambda_1 = 5$, $\lambda_2 = 50$, $\lambda_3 = 100$, $\lambda_4 = 100$, $\lambda_5 = 5$.

References

[1] Bharat Lal Bhatnagar, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Combining implicit function learning and parametric models for 3D human reconstruction. In *European Conference on Computer Vision (ECCV)*. Springer, August 2020. 2, 3, 4

[2] Bharat Lal Bhatnagar, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. LoopReg: Self-supervised learning of implicit surface correspondences, pose and shape for 3D human mesh registration. *Conference on Neural Information Processing Systems (NeurIPS)*, 33:12909–12922, 2020. 2, 3, 4

[3] Richard Hartley, Jochen Trumpf, Yuchao Dai, and Hongdong Li. Rotation averaging. *International Journal of Computer Vision (IJCV)*, 103:267–305, 2013. 6

[4] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *Conference on Neural Information Processing Systems (NeurIPS)*, 30, 2017. 5

[5] Shaofei Wang, Andreas Geiger, and Siyu Tang. Locally aware piecewise transformation fields for 3D human mesh registration. In *Computer Vision and Pattern Recognition (CVPR)*, pages 7639–7648, 2021. 2, 3, 4

[6] Jiancheng Yang, Qiang Zhang, Bingbing Ni, Linguo Li, Jinxian Liu, Mengdie Zhou, and Qi Tian. Modeling point clouds with self-attention and Gumbel subset sampling. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3323–3332, 2019. 2