Personalized 3D Human Pose and Shape Refinement

Tom Wehrbein[1,2†]   Bodo Rosenhahn[1]   Iain Matthews[2]   Carsten Stoll[2]

[1]Leibniz University Hannover, [2]Epic Games

wehrbein@tnt.uni-hannover.de

Abstract

Recently, regression-based methods have dominated the field of 3D human pose and shape estimation. Despite their promising results, a common issue is the misalignment between predictions and image observations, often caused by minor joint rotation errors that accumulate along the kinematic chain. To address this issue, we propose to construct dense correspondences between initial human model estimates and the corresponding images that can be used to refine the initial predictions. To this end, we utilize renderings of the 3D models to predict per-pixel 2D displacements between the synthetic renderings and the RGB images. This allows us to effectively integrate and exploit appearance information of the persons. Our per-pixel displacements can be efficiently transformed to per-visible-vertex displacements and then used for 3D model refinement by minimizing a reprojection loss. To demonstrate the effectiveness of our approach, we refine the initial 3D human mesh predictions of multiple models using different refinement procedures on 3DPW and RICH. We show that our approach not only consistently leads to better image-model alignment, but also to improved 3D accuracy.

1. Introduction

Reconstructing 3D human pose and shape from RGB images is a long-standing and fundamental problem in computer vision due to its various applications in *e.g.* medicine, sports, AR/VR and animation. Powered by deep CNNs and vision transformers, regression-based methods have made rapid progress and achieve state-of-the-art performance. Given a single image or video sequence, they learn to predict the parameters of a human body model (*e.g.* SMPL [41]) in a data-driven way. Despite the promising results and high efficiency, regression-based methods typically suffer from coarse alignment between predicted meshes and image evidence [79] (see Fig. 1, *top right*). This is often caused by minor joint rotation errors that ac-
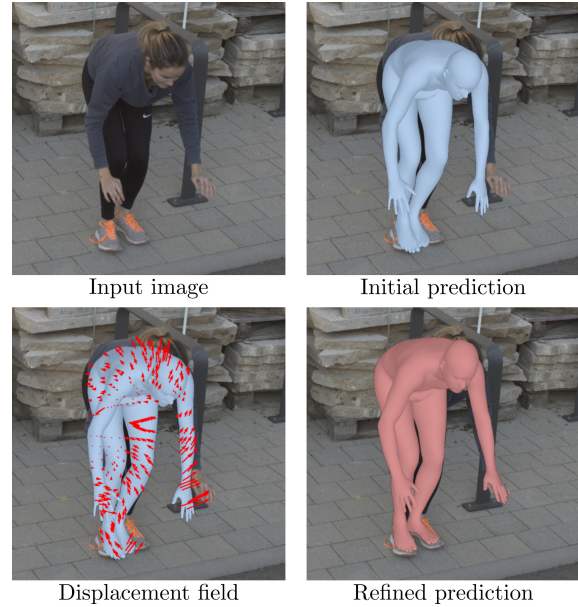

Figure 1. Given an initial 3D human model estimate, we predict per-pixel 2D displacements between renderings of the 3D model and the given image that we subsequently use to refine the initial prediction. For clarity, only a sparse subset of displacement vectors is shown. Image is taken from RICH [22].

cumulate along the kinematic chain, resulting in noticeable drift of joint positions. Furthermore, the non-linear mapping from image features to global body model parameters together with the complex nature of human appearances makes human body representations extremely difficult to regress accurately without any form of error-feedback loop. Nevertheless, high-precision estimates are crucial in various applications, especially when interacting with other people or objects in the (virtual) world.

Recently, methods have been introduced that propose to refine an initial regressed human mesh prediction [18, 27, 33, 34, 39, 54, 61, 79]. They focus on either generating 3D pseudo-annotations [27, 33, 39], adapting a model to out-of-domain videos [54], or on the more general task of improving the 3D accuracy for unconstrained monocular images [18, 34, 61, 79]. All of these methods primarily rely on

---

[†]Work primarily done during an internship at Epic Games.

a data term defined as the reprojection loss between given 2D joints and the projection of the regressed body model joints. Target 2D joints are either obtained by an off-the-shelf 2D pose estimator such as OpenPose [7], or manually annotated in an offline setting. However, the resulting human meshes are extremely sensitive to the quality of the given 2D joints. Joo *et al*. [27] observed that even manually annotated keypoints often contain non-negligible errors, causing artifacts such as foreshortening in 3D. They completely ignore the hip and ankle keypoints, since they found them to be particularly noisy. This sensitivity is even more pronounced when using 2D joint predictions, making it extremely challenging to achieve improvement of the initial 3D body estimate. There are a lot of cases in which the post-processing step even leads to a degradation in 3D accuracy [18, 27, 34, 61]. We introduce a drop-in replacement for sparse 2D keypoints that can be used for refining 3D human mesh predictions without requiring manual annotations, and show that the typical 25 keypoints used by [7] are not sufficient to robustly constrain the full human body.

Instead of using keypoints for refinement, we learn dense 2D correspondences that can be effectively used as image cues for refining estimated 3D human meshes in realistic and challenging in-the-wild scenarios. We leverage initial 3D mesh estimates generated by state-of-the-art regression-based pose estimators [31, 34, 45] and learn 2D displacements between renderings of the predictions and the corresponding images. This allows us to integrate and exploit appearance information of the person and utilize 3D information in the form of depth and normal renderings. By taking into account the initial human mesh prediction, the network only has to learn small displacements while being able to adapt to typical prediction errors. Furthermore, only image displacements need to be learned compared to complex pixel to 3D body surface mappings required by Dense-Pose [18, 19]. Instead of designing a specialized regressor architecture to improve image-model alignment [79], our approach can be combined with any 3D human mesh regressor and benefits from advances in that field, as well as advances in optimization techniques [9, 56]. Using 2D correspondences for refinement leads to better image-model alignment and to improved 3D accuracy. As shown in Fig. 1, even accurate 3D estimates can be further refined.

To evaluate our approach, we refine the initial 3D human mesh predictions of multiple models [31, 34, 45] using different fitting procedures [5, 34, 46] on 3DPW [65] and RICH [22]. We compare the performance of using Open-Pose [7], DensePose [19] and our displacement fields for the reprojection loss in optimization. We show that our displacement fields lead to significantly better performance in almost all settings and metrics. Additionally, we demonstrate that our model is robust to noisy and erroneous texture estimates, as well as to changes in illumination.

To summarize, our main contributions are:

- We present a method to learn per-pixel 2D correspondences between renderings of a 3D human mesh and an image that enables 3D human mesh refinement.

- The appearance information of the person is successfully leveraged to boost prediction accuracy.

- Our 2D displacement fields can refine the estimates of off-the-shelf 3D human mesh regressors and consistently outperform OpenPose keypoints for refinement.

2. Related Work

The de facto approach for monocular 3D human mesh recovery is to estimate the low-dimensional parameters of a statistical body model [4, 41, 46, 69] such as SMPL [41].

Optimization-based approaches have historically been the leading paradigm for model-based 3D human mesh estimation. They rely on classical optimization to iteratively fit the body model parameters to 2D image observations. Pioneering work in this area [17, 20, 55] leveraged 2D keypoints or silhouettes for human body fitting but required manual user intervention. Enabled by advances in 2D human pose estimation [49], Bogo *et al*. [5] introduced SMPLify, the first fully automated approach. SMPLify fits the SMPL model to detected 2D keypoints utilizing multiple strong priors to regularize the optimization. Subsequent work investigated different data terms, *e.g*. silhouettes [36], part orientation fields (POFs) [68], dense correspondences [18] and contact information [42, 59], extended the approach to multi-view and multi-person [11, 23, 76], and devised more efficient optimization pipelines [13]. However, due to their robustness and performance on challenging in-the-wild data, recent methods [27, 33, 34, 39, 46] almost exclusively rely on 2D skeletons estimated by off-the-shelf pose estimators [7]. To better constrain the 3D body during fitting and thus reduce ambiguities, recent work focused on constructing stronger 3D pose priors [10, 34, 46, 61] or on training neural optimizers [9, 56, 75] to predict the parameter updates. In general, optimization-based approaches achieve well-aligned results, but tend to be sensitive to initialization and the quality of the given image cues.

Regression-based approaches [28, 31–34, 37–39, 53, 67, 80] use a deep network to predict 3D body parameters directly. To compensate for the lack of in-the-wild 3D annotations, methods have focused on integrating alternative supervision signals. They often rely on 2D annotations, such as keypoints [28, 64], silhouettes [48, 53, 62], part segmentations [12, 31, 44, 74], or dense correspondences [18, 72, 77, 78, 80], that can be integrated as reprojection losses or leveraged as proxy representations. Regression-based approaches are fast and achieve state-of-the-art reconstruction performance. However, since they lack an error-feedback

loop, they typically suffer from coarse alignment between predicted meshes and image evidence [79]. Recently, aiming at producing well-aligned meshes, Zhang *et al*. [79] introduced PyMAF, a specialized deep regressor with an integrated alignment feedback loop that leverages learned feature pyramids.

Hybrid approaches. To combine the best of both paradigms, recent work has explored hybrid approaches. [48] demonstrated that by initializing SMPLify with their regressed pose parameters the fitting procedure is three times faster and converges to better solutions than vanilla SMPLify. SPIN [33] also uses a regression network to initialize the optimization and leverages the fitted estimates to supervise the network. This approach has been extended to multi-view by [40]. With the goal of generating 3D pseudo-annotations for 2D datasets, EFT [27] updates the network weights for each frame to achieve better reprojection accuracy. In a similar manner, BOA [54] adapts a trained network to out-of-domain streaming videos. All of the above methods exclusively rely on sparse 2D keypoints as image evidences. HoloPose [18] introduces a refinement procedure that penalizes deviations between the regressed body model and DensePose/2D/3D joint predictions. However, while the image alignment improves, the 3D accuracy slightly degrades when using DensePose and/or 2D joints for refinement. Similar observations have been made by [27, 34, 61], emphasizing the difficulty of fitting 3D model estimates to image cues, especially if the image cues are sparse and noisy and the model estimates are already good.

3. Method

Our aim is to construct dense correspondences between an initial human mesh prediction and the given image that can be used to refine the initial prediction and thus improve the accuracy of the 3D mesh. Motivated by progress and applications in 3D human texture estimation [2,3,26,47,50, 71], we aim to exploit the appearance of the person for 3D mesh refinement. Given a short calibration sequence where the person is seen from all sides, an accurate texture map can be computed [2, 3, 26]. If no calibration sequence is available, a rough texture map can be build over time [50]. We argue that for almost all practical applications, first estimating the texture map of the person is non-intrusive and adds very little overhead. Inspired by the 6D object pose estimation refinement approach of Grabner *et al*. [16], we learn pixel-wise 2D displacement fields between the 3D human model renderings and the images similar to optical flow [14, 24, 60, 70]. Unlike Grabner *et al*. [16], we utilize the estimated texture maps to generate RGB renderings. Additionally, we use depth, normal and unique vertex color renderings to explicitly provide 3D information. We train a CNN-based network (Sec. 3.1) that takes as input the model renderings together with the image and outputs a 2D

displacement vector for each rendered pixel. The per-pixel displacements can then be efficiently transformed to per-visible-vertex displacements utilizing information provided by the renderer. Finally, we use the per-vertex vectors to perform 3D model refinement (Sec. 3.2). This way, an ideal geometric reprojection loss of the full human mesh can be minimized. The overall framework is depicted in Fig. 2.

Body representation. We use SMPL [41] to represent the human body. It provides a differentiable function that given pose $\boldsymbol{\theta} \in \mathbb{R}^{72}$ and shape $\boldsymbol{\beta} \in \mathbb{R}^{10}$ parameters outputs a 3D mesh $\mathcal{M}(\boldsymbol{\theta}, \boldsymbol{\beta}) \in \mathbb{R}^{N \times 3}$ with $N = 6890$ vertices. In addition, the 3D body joints $\mathcal{J}_{3D}$ can be expressed as a linear combination of the mesh vertices. A linear regressor W can be pretrained for this task to produce the k joints of interest $\mathcal{J}_{3D} = W\mathcal{M} \in \mathbb{R}^{k \times 3}$.

3.1. Model Design

Given a single RGB image, an initial 3D human mesh and camera estimate and an approximate texture map of the person, we compute per-pixel 2D displacements between the 3D human model renderings and the image. Formally, the per-pixel displacement field $\mathbf{f} : \mathbb{N}^2 \rightarrow \mathbb{R}^2$ maps every valid 2D pixel location $\mathbf{x} \in \mathbb{N}^2$ of the renderings $\mathbf{I}_r$ to its corresponding 2D location $\mathbf{p} = \mathbf{x} + \mathbf{f}(\mathbf{x})$ of the target RGB image $\mathbf{I}_t$. A 2D pixel position is considered valid if a pixel has been rendered at that position, therefore correspondences are not learned for background pixels.

The first step in our pipeline is to render the initial 3D human mesh using the estimated camera parameters and texture map. In addition to RGB, we generate depth, normal and unique vertex color renderings. Thus, important 3D information is provided to the network. The unique per-vertex color attributes are defined as the 3D vertex positions of the neutral SMPL body with mean shape and pose parameters. Given the output of the rasterizer and the 3D model, the unique vertex color rendering is computed by interpolating the vertex color attributes. The different renderings are concatenated along the channel dimension. Next, in order to predict 2D displacements we map the renderings and the input RGB image to a common feature space. We utilize two different feature encoder branches for this task. The architecture of both branches is similar to the first stage of ResNet-50 [21]. The only adjustment we make is to use a stride of 1 for all convolutional layers and to remove all max pooling layers. This maintains the input image size which makes it easier to predict fine-grained displacements. The architecture of the input image and the renderings feature branches only differ in the number of input channels. After mapping both input modalities to the common feature space, we concatenate the feature maps and use a stacked hourglass network [43] with 4 stacks to predict the 2D per-pixel displacement field. We train all network branches end-to-end from scratch. By explicitly predicting the displace-
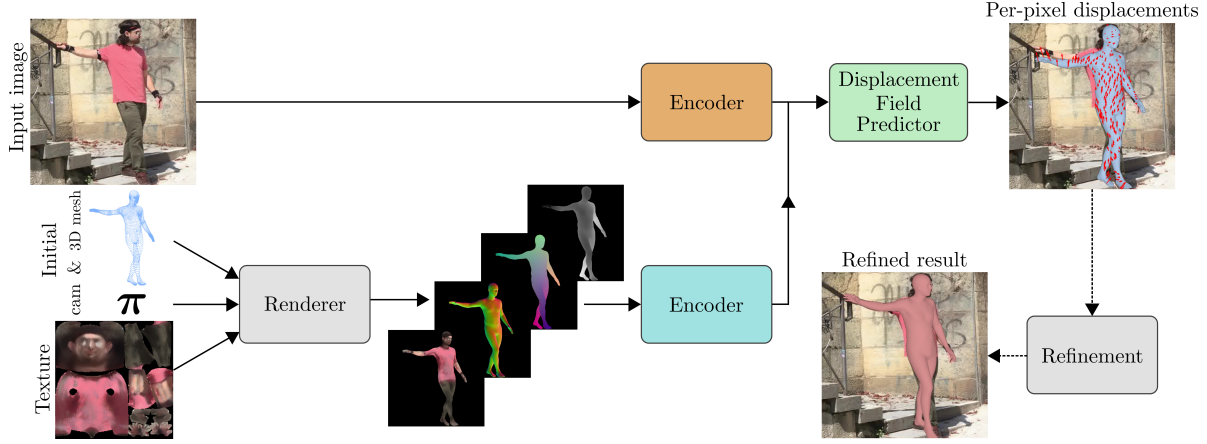
Figure 2. Overview of our proposed approach. Given an image with estimated 3D human mesh, camera parameters $\boldsymbol{\pi}$ and an approximate texture map of the target person, we predict the per-pixel 2D displacement field between the 3D human model renderings and the image. The per-pixel 2D displacements are transformed to per-visible-vertex displacements and can subsequently be used to refine the 3D human model using *e.g.* SMPLify [5]. For clarity, only a sparse subset of displacement vectors is shown. Reference image from 3DPW [65].

ment fields, the network learns to be robust to noisy texture maps and changes in illumination. Additionally, the absence of the scene background in the renderings can be easily dealt with. We found that despite the similarity in task, deep optical flow models (*e.g.* [24, 60, 70]) do not perform well, even when retrained on human mesh data and when two separate feature encoders are used. We hypothesize that the 4D correlation volume build by these methods cannot effectively handle large differences in illumination between the rendered and input image. Furthermore, it is extremely challenging to learn a feature mapping that makes correlating normal and depth features with image features meaningful.

Optimization. We train our model in a fully-supervised manner. Given an image with corresponding ground-truth SMPL pose $\hat{\boldsymbol{\theta}}$ and shape $\hat{\boldsymbol{\beta}}$ parameters and camera projection function $\hat{\boldsymbol{\pi}} : \mathbb{R}^3 \rightarrow \mathbb{R}^2$, together with a second set of SMPL and camera parameters $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, $\boldsymbol{\pi}$, we first obtain the ground-truth 2D per-vertex displacement field $\hat{\mathbf{v}} \in \mathbb{R}^{N \times 2}$ between the projection of both 3D human meshes:

$$\hat{\mathbf{v}} = \hat{\boldsymbol{\pi}}(\mathcal{M}(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\beta}})) - \boldsymbol{\pi}(\mathcal{M}(\boldsymbol{\theta}, \boldsymbol{\beta})). \tag{1}$$

The parameters $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, $\boldsymbol{\pi}$ are either obtained by random perturbations of the corresponding ground-truth parameters or by using the regressed values of some pretrained SMPL prediction model (*e.g.* [31, 34, 45]). Since we supervise on the pixel level, the per-vertex displacement field needs to be transformed to a per-pixel displacement field. This is done by interpolating the per-vertex displacements across the projected triangle surfaces using barycentric coordinates. Formally, the 2D ground-truth displacement of the pixel at position (x, y) is computed as:

$$\hat{\mathbf{f}}_{x,y} = \sum_{i=1}^{3} b_{x,y,i} \cdot \hat{\mathbf{v}}_{\triangle_{\texttt{IndexMap}_{x,y,i}}} \tag{2}$$

where $\triangle_{\texttt{IndexMap}_{x,y,i}}$ indexes the i-th vertex of the triangle visible at (x, y) and $b_{x,y,i}$ is the corresponding barycentric coordinate.

Finally, we supervise our network on the l_1 distance between the ground-truth and predicted 2D per-pixel displacement field:

$$\mathcal{L} = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} m_{x,y} ||\hat{\mathbf{f}}_{x,y} - \mathbf{f}_{x,y}||_1, \tag{3}$$

where $m_{x,y}$ is 1 if a pixel has been rendered at position (x, y) and 0 otherwise. The loss is applied at the end of each stacked hourglass stack and the output of the last layer is used as the final prediction.

3.2. 3D Human Mesh Refinement

SMPLify [5] is a popular optimization-based method that fits the SMPL body model to a set of sparse 2D keypoints. The objective function it minimizes consists of a re-projection term encouraging the 3D body model to explain the observed 2D keypoints and of pose and shape priors that regularize the fit. More specifically, the optimal fit is given by:

$$(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*, \boldsymbol{\pi}^*) = \arg\min_{\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\pi}} \lambda_{2D} \mathcal{L}_{2D} + \\ \lambda_{\boldsymbol{\theta}} \mathcal{L}_{\boldsymbol{\theta}} + \lambda_{\boldsymbol{\beta}} \mathcal{L}_{\boldsymbol{\beta}} + \lambda_{\alpha} \mathcal{L}_{\alpha}, \tag{4}$$

with re-projection term $\mathcal{L}_{2D}$, 3D pose prior $\mathcal{L}_{\boldsymbol{\theta}}$, shape regularizer $\mathcal{L}_{\boldsymbol{\beta}}$ and joint bending term $\mathcal{L}_{\alpha}$. The re-projection term in the original paper calculates the distance between estimated 2D pose keypoints such as [7] and the corresponding projected joint locations of the SMPL model. Since the camera parameters are usually unknown, they must be optimized together with the body parameters. The

bending term $\mathcal{L}_\alpha = \sum_{i\in(\text{elbows,knees})} \exp(\boldsymbol{\theta}_i)$ penalizes unnatural rotations of elbows and knees, the shape regularizer is given as $\mathcal{L}_{\boldsymbol{\beta}} = \|\boldsymbol{\beta}\|^2$ and the 3D pose prior $\mathcal{L}_{\boldsymbol{\theta}}$ is expressed via a Gaussian mixture model. In order to better constrain the 3D body during fitting and thus reduce ambiguities, recent work [10, 34, 46, 61] focused on designing stronger 3D pose priors to replace the GMM. However, the success of fitting the parametric body model depends heavily on the initialization, the balance of data and prior terms and the quality of the sparse 2D keypoints [5, 27, 36].

We approach the problem from a different angle and argue that a main source of ambiguity is the insufficient data term $\mathcal{L}_{2D}$. There can be a lot of different mesh configurations that explain the observed sparse 2D keypoints [34, 66]. Motivated by these limitations, we propose to replace the sparse 2D keypoints with our dense per-pixel displacement fields. To do this, the predicted per-pixel 2D displacement vectors need to be transformed to per-vertex displacements. For vertex i, this is achieved by accumulating the 2D vectors for all pixel positions (x, y) for which the vertex i is a vertex of the triangle visible at (x, y). Formally, the displacement $\mathbf{v}_i$ of vertex i is computed as:

$$\mathbf{v}_i = \frac{1}{\sum_{x,y} b_{x,y,i}} \sum_{x,y} (b_{x,y,i} \cdot \mathbf{f}_{x,y})$$
$$\forall x, y : i \in \triangle_{\texttt{IndexMap}_{x,y}}, \tag{5}$$

where $\mathbf{f}_{x,y}$ is the predicted 2D displacement at pixel position (x, y) and $b_{x,y,i}$ is the barycentric coordinate of vertex i at that position[1]. The indices of the vertices of the triangle visible at (x, y) are given by $\triangle_{\texttt{IndexMap}_{x,y}}$. Finally, we obtain the target 2D vertices $\mathcal{V}_{\text{est}} \in \mathbb{R}^{N\times 2}$ by simply adding the predicted displacement field and the projection of the initial model parameters:

$$\mathcal{V}_{\text{est}} = \mathbf{v} + \tilde{\boldsymbol{\pi}}(\mathcal{M}(\tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\beta}})). \tag{6}$$

Instead of using distance between 2D joint locations we then define the re-projection term of Eq. 4 as:

$$\mathcal{L}_{2D} = \sum_{i\in\text{vertices}} w_i \rho(\boldsymbol{\pi}(\mathcal{M}(\boldsymbol{\theta}, \boldsymbol{\beta}))_i - \mathcal{V}_{\text{est},i}), \tag{7}$$

where w_i equals 1 if vertex i is visible in the rendering and 0 otherwise, and ρ represents a robust Geman-McClure error [15]. In addition to providing significantly more 2D landmarks that better constrain the human body during fitting, optimization is no longer dependent on the potentially slightly inaccurate linear mesh-to-joint regressor W. In the experimental section, we show that using our estimated dense 2D displacement, we are able to consistently improve the fitting results over sparse landmark approaches.

---

[1]Note that we slightly abuse the notation of Eq. 2 and index the barycentric coordinate with the mesh vertex index.

Figure 3. Examples of reconstructed texture maps from Human3.6M, 3DPW and RICH used for training and evaluation. We generate texture maps by back-projecting the image colors from multiple frames to all visible vertices.

4. Experiments

Training. We train our model on the standard training sets of Human3.6M [25], 3DPW [65] and SURREAL [63] using ground-truth camera and SMPL annotations. Since we want to learn 2D displacements between a rendered 3D mesh and the person in the image, a rich set of SMPL and camera parameter predictions per training image is required. For every training image, we precompute SMPL and camera parameter predictions using PARE [31] and ProHMR [34]. We utilize the probabilistic characteristic of ProHMR and sample 64 predictions for each frame. To focus on fine-grained displacements, we additionally use ground-truth pose with PARE predicted shape and camera parameters during training. We perform the rendering on-the-fly at the start of each iteration using nvdiffrast [35]. Since we do not need to keep track of gradients, rendering a batch of 8 only takes around 1 ms and thus causes almost no overhead. For further implementation and training details, we refer the reader to the supplemental material.

Evaluation. We use the test splits of 3DPW and the newly released dataset RICH [22] which contains outdoor and indoor video sequences with highly accurate 3D mesh annotations and subjects with varied body shapes[2]. We focus evaluation on challenging in-the-wild scenes and the generalization capability to unseen body shapes and camera angles. No test subject is seen during training. We report the mean per joint position error (MPJPE) and its scale normalized [52] (N-MPJPE) and Procrustes aligned (PA-MPJPE) variants. The equivalent metrics to evaluate the per vertex error are denoted as PVE, N-PVE and PA-PVE. All metrics are measured in millimeters.

Data preprocessing. Since only SURREAL provides ground-truth textures, we need to compute the texture maps for the subjects in Human3.6M, 3DPW and RICH. Note that no texture calibration sequence is available for each subject and that the focus of this work is not on reconstructing high-quality textures. Therefore, we resort to simply back-projecting the image colors from the reference sequence of a subject to all visible vertices and finally calculate the texture map by taking the median color values. As seen in

---

[2]All datasets were obtained and used only by the authors affiliated with academic institutions.

Method + SMPLify	3DPW						RICH					
	MPJPE ↓	PA-MPJPE ↓	N-MPJPE ↓	PVE ↓	PA-PVE ↓	N-PVE ↓	MPJPE ↓	PA-MPJPE ↓	N-MPJPE ↓	PVE ↓	PA-PVE ↓	N-PVE ↓
ProHMR [34]	95.1	59.5	93.2	109.6	74.9	108.4	126.4	70.3	111.8	152.9	86.1	126.8
+GMM (DP)	101.1	67.7	99.6	118.7	84.4	116.1	129.7	73.7	111.2	155.3	89.2	125.9
+GMM (OP)*	103.2	66.6	101.1	118.3	82.6	117.0	130.6	73.6	115.9	152.7	88.5	130.5
+GMM (OP)	86.1	58.7	83.1	103.6	76.9	101.7	119.0	65.8	101.5	140.7	81.2	115.7
+GMM (*Ours*)	**79.7**	**53.9**	**78.0**	**95.9**	<u>69.7</u>	**94.0**	**108.5**	**63.8**	**90.4**	<u>135.1</u>	**79.1**	**105.6**
+VPoser (DP)	96.2	61.7	94.7	116.8	80.8	114.8	125.3	69.1	108.2	151.4	85.5	123.6
+VPoser (OP)	85.6	58.0	81.4	104.2	77.5	102.1	115.1	<u>65.0</u>	99.7	137.4	81.3	114.6
+VPoser (*Ours*)	84.7	57.4	83.0	103.0	75.3	101.6	<u>110.3</u>	65.7	<u>93.4</u>	137.1	81.7	<u>109.1</u>
+cNF (DP)	90.8	58.9	88.1	103.2	72.7	101.4	118.7	67.2	105.2	140.2	82.1	119.3
+cNF (OP)	88.5	<u>54.6</u>	84.2	103.2	**69.5**	97.6	118.5	65.5	104.9	138.7	80.6	118.8
+cNF (*Ours*)	<u>84.5</u>	54.7	<u>81.1</u>	<u>97.6</u>	69.5	<u>95.5</u>	111.0	65.1	97.5	**132.7**	<u>80.1</u>	111.8
PARE [31]	74.5	46.6	72.9	88.6	61.8	87.2	106.8	55.8	86.6	128.8	69.3	100.1
+GMM (DP)	102.4	69.2	100.9	117.5	86.9	116.9	122.6	67.7	101.4	144.9	81.9	114.4
+GMM (OP)*	94.1	60.4	92.4	108.7	75.6	107.9	124.4	66.5	105.8	144.5	80.5	119.3
+GMM (OP)	80.8	54.4	79.0	97.4	72.9	96.3	112.1	58.3	89.4	131.0	71.6	102.2
+GMM (*Ours*)	<u>65.5</u>	<u>44.5</u>	<u>63.6</u>	<u>79.6</u>	<u>59.4</u>	<u>78.1</u>	<u>95.2</u>	<u>51.6</u>	<u>71.4</u>	<u>117.3</u>	<u>64.7</u>	<u>84.9</u>
+VPoser (DP)	89.7	51.0	88.1	102.5	65.6	101.6	111.7	56.0	91.2	132.2	68.3	103.2
+VPoser (OP)	73.0	45.0	69.8	86.5	59.9	84.3	104.2	52.8	83.1	121.9	65.3	95.0
+VPoser (*Ours*)	**65.2**	**43.5**	**63.4**	**79.3**	**58.0**	**77.6**	**93.9**	**50.7**	**70.9**	**115.1**	**63.0**	**83.9**
HMR+ [28, 45]	83.0	52.1	81.5	98.1	70.8	96.1	119.3	62.4	101.6	144.6	78.5	117.3
+GMM (OP)	83.0	56.2	80.9	100.3	74.9	98.7	115.7	62.5	95.8	134.8	77.1	109.3
+GMM (*Ours*)	<u>75.0</u>	<u>49.4</u>	<u>73.0</u>	<u>89.9</u>	<u>65.7</u>	<u>87.0</u>	<u>107.2</u>	<u>59.4</u>	<u>85.8</u>	<u>132.3</u>	<u>74.2</u>	<u>101.2</u>
+GMM (*Ours*)†	**74.5**	**49.3**	**72.6**	**89.6**	**65.2**	**86.5**	**106.6**	**59.1**	**85.6**	132.4	**73.5**	**100.9**
+VPoser (OP)	79.1	52.0	76.2	96.8	72.7	94.1	113.4	61.2	95.6	133.3	76.9	110.0
+VPoser (*Ours*)	78.4	52.6	76.5	95.6	71.8	93.0	110.8	62.5	90.4	135.8	77.7	106.2
+VPoser (*Ours*)†	78.2	52.5	76.4	95.4	71.4	92.7	110.6	62.4	90.4	135.7	77.2	106.2

Table 1. Detailed results for 3D human mesh refinement using our 2D displacements, OpenPose keypoints and DensePose predictions. The regressed SMPL parameters of ProHMR [34], PARE [31] and HMR+ [45] are refined using SMPLify [5] with GMM [5], VPoser [46] and a conditional Normalizing Flow (cNF) [34] as pose prior. * denotes the default SMPLify implementation of SPIN [33] and † that we fine-tuned our model on training set predictions of HMR+. The unit of all numbers is mm and the best results are in bold.

Fig. 3, the resulting textures often contain visual artifacts and are blurry, especially in the facial area and around the hands. This is caused by imperfect 3D mesh annotations and is particularly noticeable for 3DPW. Furthermore, the reconstructed textures for the subjects in 3DPW are sometimes incomplete since subjects are not always seen from all sides. We leave the exploration of more sophisticated texture reconstruction approaches [2, 3, 26, 71] to future work.

4.1. Quantitative Evaluation

To demonstrate the benefits of using our 2D displacement fields for 3D human mesh refinement, we evaluate SMPLify with three different pose priors on 3DPW and RICH, using the initial SMPL and camera predictions of three different models. We compare the results against fitting with OpenPose (OP) and DensePose (DP) predictions. For the GMM [5] and VPoser [46] pose priors, we use the publicly available SMPLify implementation of SPIN [33] and initialize the fitting process with predictions from ProHMR [34], HMR+ [45], and the state-of-the-art model PARE [31]. However, we noticed that the default implementation consistently leads to very poor results, especially if using OP predictions and if the initialization is already good. This has also been observed by [27, 30, 34, 61]. To improve the results, we modify the default implementation by 1) removing the bending and camera depth prior term, 2) fitting in the full image space instead

of the cropped and 3) using a focal length approximation of $f = \sqrt{w^2 + h^2}$ [30, 39], where w and h are the width and height of the full image. We scale the reprojection loss depending on the size of the person in the image and multiply it by 5.0, 0.4, 0.002 for OP, our 2D displacements and DP respectively. For fitting with the conditional Normalizing Flow [51, 57, 58] (cNF) pose prior of ProHMR [34], we use their publicly available fitting implementation. We only adjust the weight of the reprojection term to account for having significantly more 2D landmarks. We multiply the reprojection loss by 0.04 and 0.002 for our 2D displacements and DP respectively. While it is possible to re-evaluate our displacement prediction network after each iteration, we did not find a significant advantage over evaluating it once.

The results are shown in Table 1. Our predicted 2D displacements lead to the best fitting results in nearly all metrics and settings. The gap to OP and DP fitting is especially large when using the GMM pose prior, showing that due to our dense and accurate displacement fields, a complex pose prior such as VPoser is not necessary to constrain the pose space. While OP fitting with our adjusted SMPLify version significantly improves upon the default implementation, the performance still heavily degrades when using the GMM prior with predictions from PARE. Thus, fitting to OP keypoints is more sensitive to the initialization and has to rely on strong pose priors. Since DP fitting with GMM and VPoser, and the default SMPLify imple-

	3DPW			
Method + SMPLify	MPJPE ↓	PA-MPJPE ↓	N-MPJPE ↓	PVE ↓
ProHMR [34]	95.1	59.5	93.2	109.6
+GMM (OP GT)	69.5	43.7	66.3	81.6
+GMM (Ours GT)	**56.7**	**36.3**	**54.6**	**66.1**
PARE [31]	74.5	46.6	72.9	88.6
+GMM (OP GT)	50.5	33.4	47.6	62.8
+GMM (Ours GT)	**41.0**	**26.5**	**38.2**	**48.5**

Table 2. Refining ProHMR and PARE estimates using SMPLify with our GT 2D per-pixel displacements and GT OP keypoints.

PARE + VPoser	EPE ↓	MPJPE ↓	PA-MPJPE ↓	PVE ↓
PARE [31]	-	74.5	46.6	88.6
texture + $\mathcal{N}(\mathbf{0}, \mathbf{10})$	3.98	66.8	44.0	80.6
texture + $\mathcal{N}(\mathbf{0}, \mathbf{30})$	4.31	68.5	44.7	82.1
brightness + $\mathcal{N}(0, 25)$	3.75	65.4	43.7	79.5
brightness + $\mathcal{N}(0, 50)$	3.84	66.0	44.0	80.1
wrong texture	4.73	71.1	46.1	84.9
w/o texture	4.17	66.9	44.5	80.9
texture only	3.79	65.8	43.8	79.9
Ours (Full)	**3.70**	**65.2**	**43.5**	**79.3**

Table 3. Evaluation of the influence of the texture. Results are for refining PARE estimates on 3DPW using SMPLify with VPoser prior. To assess the robustness of our model to noisy and erroneous texture estimates, we manipulate the textures by adding pixel-wise Gaussian noise, changing the brightness and using textures of wrong subjects.

mentation always lead to a loss in performance, we do not show the numbers for all settings for the sake of visibility. We found that while the DP model is generally good at detecting pixels that belong to the person, the predicted correspondences between the pixels and the 3D SMPL surface often lack in accuracy, particularly at the boundary between body parts. Our displacement fields significantly outperforming the DP predictions for fitting shows the benefit of learning 2D displacements in the image space instead of complex pixel to 3D body surface mappings. Interestingly, using the strong cNF prior of ProHMR leads to improved results even for DP. The image-conditioned prior limits the pose space significantly more than the generic priors and can thus better handle noisy 2D landmarks. However, the prior heavily depends on the estimated conditional pose distribution. Therefore, it is not suitable to use in combination with stronger models such as PARE, since the fitting converges to solutions of similar accuracy as when initialized with ProHMR. Nonetheless, fitting with the cNF prior also works best with our 2D displacements. Due to our dense and accurate 2D displacements, fitting on average works best with the lightweight and simple GMM pose prior.

Although we trained our model only with ProHMR and PARE estimates, it generalizes quite well to predictions of HMR+ as can be seen in Table 1. We can further improve the performance by generating HMR+ estimates for our training images and fine-tuning on them. The results are also shown in Table 1.

To assess the performance upper bound of our approach, we present the metrics for fitting with ground-truth per-pixel fields in Table 2. We compare the results to fitting with the 25 ground-truth joints corresponding to the OpenPose skeleton, which we generate from the ground-truth SMPL mesh using the linear regressor and dataset given camera. Therefore, ground-truth values for occluded joints are used as well. Despite that our displacement fields only regard visible vertices, they still significantly outperform the 25 OpenPose joints in fitting. As expected, the performance gap is particularly large for the quality of the refined 3D meshes as shown by the per-vertex error. Interestingly, we found that the GMM prior consistently outperforms the other two priors when using the 25 GT keypoints.

4.2. Qualitative Evaluation

To better illustrate the degree of improvements, we compare our refined 3D human models with the initial predictions in Fig. 4 and with refinements using OP keypoints in Fig. 5. We use the state-of-the-art model PARE and SMPLify with VPoser. Although the initial estimates are already accurate, our refinement clearly further improves the 3D models. Compared with the refinements using OP keypoints, we achieve significantly better reconstructions of the spine. The sparse OP keypoints cannot effectively capture articulation around this area, resulting in incorrect arching of the back. A larger variety of results and also failure cases can be found in the supplementary material.

4.3. Ablation Studies

To assess the influence of the texture map, we train a model without texture and one with texture only. We present the end-point-error (EPE) of the predicted 2D displacements and the metrics after refinement in Table 3. The appearance information in the textures is successfully leveraged to achieve more accurate displacement predictions. Additionally providing depth, normal and vertex color renderings further boost the performance. Despite the importance of the texture, using the model without texture still leads to noticeably improvements. Thus, if only a single image of a subject is available, it is possible to use the model trained without texture. We also evaluate the robustness of our model to noisy and erroneous texture estimates, as well as to changes in illumination. We want to again emphasize that most of the textures are very inaccurate to begin with (see Fig. 3). For the evaluation, we add pixel-wise Gaussian noise, randomly change the brightness and use the texture of a different subject. As shown in Table 3, our model is extremely robust to changes in illumination and can also handle pixel noise very well. The performance most heavily degrades when using the texture of a wrong subject.
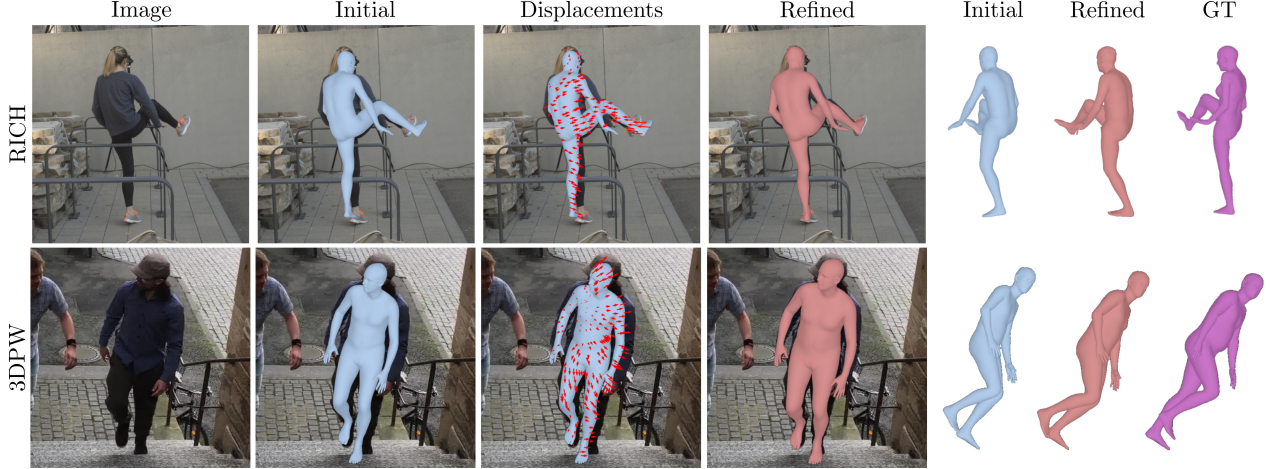
Figure 4. Qualitative results on RICH [22] and 3DPW [65]. From left to right: input images, initial body estimates, our predicted displacement fields, our refined 3D human models and side views of initial, refined and ground-truth bodies.
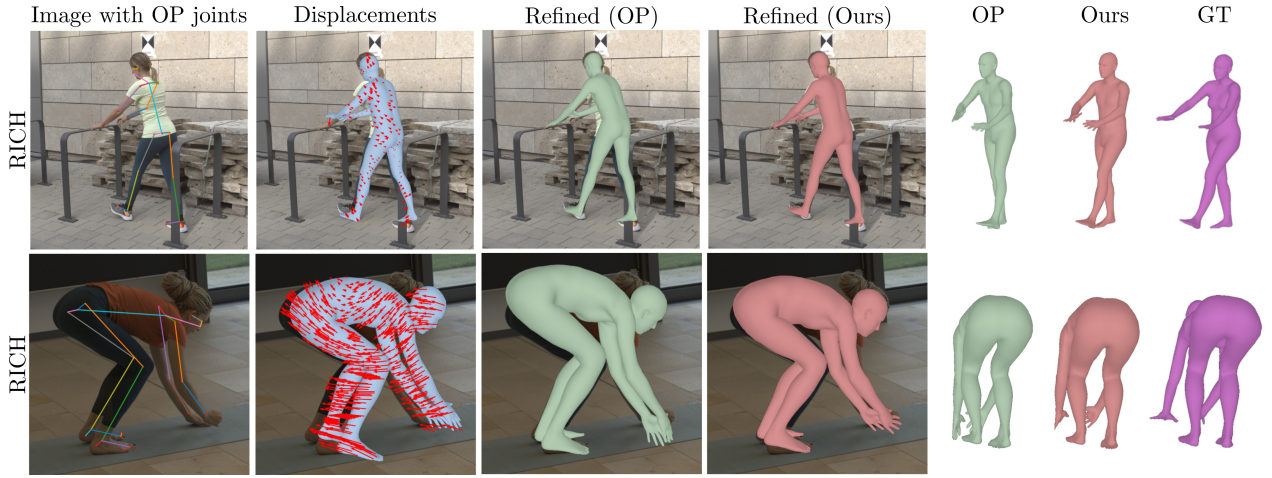

Figure 5. Qualitative comparison on RICH [22]. We compare our refined 3D human models (red) with refinements using OpenPose keypoints (green) and the ground-truth bodies (magenta). Best viewed with zoom and in color.

5. Conclusion

Motivated by the observation that regression-based methods often suffer from coarse alignment between the predicted meshes and image evidences, this work presents an approach to refine initial 3D human mesh estimates using predicted 2D displacement fields. We learn displacement fields between renderings of the 3D model predictions and the images. This allows us to exploit the appearance of the persons in form of rough texture maps and additionally leverage 3D information encoded in normal and depth renderings. Using SMPLify, we demonstrate that dense 2D displacements can be successfully used to improve the image-model alignment and the 3D accuracy of initial 3D model estimates. Experimental results show that our dense displacements outperform OpenPose and DensePose predictions for 3D human pose and shape refinement.

Limitations and future work. Since our model lever-ages texture maps, an obvious limitation is that the texture map must be recalculated when the person changes clothes so as to not lose performance. Exploring an automated way to detect change of clothes and then update the texture could be interesting future research. Furthermore, as SMPL only captures the undressed shape of the body, extremely loose clothing cannot be modeled. To improve our approach for loose and complex clothing, future work could employ the SMPL+D model [3], which extends SMPL by a set of 3D offsets that can be optimized for during the texture calibra-tion sequence. Finally, we aim to apply our approach to multi-view and motion sequences.

References

[1] https://github.com/vchoutas/smplx/tree/master/transfer_model. 12

[2] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *CVPR*, 2019. 3, 6

[3] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *CVPR*, 2018. 3, 6, 8

[4] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: Shape completion and animation of people. *SIGGRAPH*, 2005. 2

[5] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, 2016. 2, 4, 5, 6, 12

[6] Alexander Buslaev, Vladimir I. Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A. Kalinin. Albumentations: Fast and flexible image augmentations. *Information*, 11(2), 2020. 12

[7] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 2, 4, 12, 15

[8] Minghui Chen, Zhiqiang Wang, and Feng Zheng. Benchmarks for corruption invariant person re-identification. In *NeurIPS*, 2021. 12

[9] Vasileios Choutas, Federica Bogo, Jingjing Shen, and Julien Valentin. Learning to fit morphable models. In *ECCV*, 2022. 2

[10] Andrey Davydov, Anastasia Remizova, Victor Constantin, Sina Honari, Mathieu Salzmann, and Pascal Fua. Adversarial parametric pose prior. In *CVPR*, 2022. 2, 5

[11] Zijian Dong, Jie Song, Xu Chen, Chen Guo, and Otmar Hilliges. Shape-aware multi-person pose estimation from multi-view images. In *ICCV*, 2021. 2

[12] Sai Kumar Dwivedi, Nikos Athanasiou, Muhammed Kocabas, and Michael J. Black. Learning to regress bodies from images using differentiable semantic rendering. In *ICCV*, 2021. 2

[13] Taosha Fan, Kalyan Vasudev Alwala, Donglai Xiang, Weipeng Xu, Todd Murphey, and Mustafa Mukadam. Revitalizing optimization for 3d human pose and shape estimation: A sparse constrained formulation. *ICCV*, 2021. 2, 12

[14] Philipp Fischer, Alexey Dosovitskiy, Eddy Ilg, Philip Häusser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. *CVPR*, 2015. 3

[15] Stuart Geman and Donald E. McClure. Statistical methods for tomographic image reconstruction. 1987. 5

[16] Alexander Grabner, Yaming Wang, Peizhao Zhang, Peihong Guo, Tong Xiao, Peter Vajda, Peter M. Roth, and Vincent Lepetit. Geometric correspondence fields: Learned differentiable rendering for 3d pose refinement in the wild. In *ECCV*, 2020. 3

[17] P. Guan, A. Weiss, A. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *ICCV*, 2009. 2

[18] Riza Alp Guler and Iasonas Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *CVPR*, 2019. 1, 2, 3, 12

[19] Riza Alp Guler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018. 2, 12, 16

[20] Nils Hasler, Hanno Ackermann, Bodo Rosenhahn, Thorsten Thormählen, and Hans-Peter Seidel. Multilinear pose and body shape estimation of dressed subjects from image sets. In *CVPR*, pages 1823–1830, 2010. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[22] Chun-Hao P. Huang, Hongwei Yi, Markus Höschle, Matvey Safroshkin, Tsvetelina Alexiadis, Senya Polikovsky, Daniel Scharstein, and Michael J. Black. Capturing and inferring dense full-body human-scene contact. In *CVPR*, 2022. 1, 2, 5, 8, 12, 14, 15, 16, 17

[23] Yinghao Huang, Federica Bogo, Christoph Lassner, Angjoo Kanazawa, Peter V. Gehler, Javier Romero, Ijaz Akhter, and Michael J. Black. Towards accurate marker-less human shape and pose estimation over time. In *International Conference on 3D Vision (3DV)*, 2017. 2

[24] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. FlowFormer: A transformer architecture for optical flow. *ECCV*, 2022. 3, 4

[25] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(7), 2014. 5

[26] Umar Iqbal, Akin Caliskan, Koki Nagano, Sameh Khamis, Pavlo Molchanov, and Jan Kautz. Rana: Relightable articulated neural avatars. *arXiv preprint arXiv:2212.03237*, 2022. 3, 6

[27] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation. In *3DV*, 2020. 1, 2, 3, 5, 6

[28] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 2, 6

[29] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 12

[30] Imry Kissos, Lior Fritz, Matan Goldman, Omer Meir, Eduard Oks, and Mark Kliger. Beyond weak perspective for monocular 3d human pose estimation. In *ECCVW*, 2020. 6

[31] Muhammed Kocabas, Chun-Hao P. Huang, Otmar Hilliges, and Michael J. Black. PARE: Part attention regressor for 3D human body estimation. In *ICCV*, 2021. 2, 4, 5, 6, 7, 12

[32] Muhammed Kocabas, Chun-Hao P. Huang, Joachim Tesch, Lea Müller, Otmar Hilliges, and Michael J. Black. SPEC:

Seeing people in the wild with an estimated camera. In *ICCV*, 2021. 2

[33] Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, 2019. 1, 2, 3, 6, 12

[34] Nikos Kolotouros, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis. Probabilistic modeling for human mesh recovery. In *ICCV*, 2021. 1, 2, 3, 4, 5, 6, 7

[35] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6), 2020. 5

[36] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J. Black, and Peter V. Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *CVPR*, 2017. 2, 5

[37] Jiefeng Li, Siyuan Bian, Qi Liu, Jiasheng Tang, Fan Wang, and Cewu Lu. NIKI: Neural inverse kinematics with invertible neural networks for 3d human pose and shape estimation. In *CVPR*, 2023. 2

[38] Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In *CVPR*, pages 3383–3393, 2021. 2

[39] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. Cliff: Carrying location information in full frames into human pose and shape estimation. In *ECCV*, 2022. 1, 2, 6

[40] Li, Zhongguo and Oskarsson, Magnus and Heyden, Anders. 3D Human Pose and Shape Estimation Through Collaborative Learning and Multi-View Model-Fitting. In *WACV*, 2021. 3

[41] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 2015. 1, 2, 3

[42] Lea Müller, Ahmed A. A. Osman, Siyu Tang, Chun-Hao P. Huang, and Michael J. Black. On self-contact and human pose. In *CVPR*, 2021. 2

[43] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. *ECCV*, 2016. 3

[44] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter V. Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model-based human pose and shape estimation. In *3DV*, 2018. 2

[45] Hui En Pang, Zhongang Cai, Lei Yang, Tianwei Zhang, and Ziwei Liu. Benchmarking and analyzing 3d human pose and shape estimation beyond algorithms. In *NeurIPS*, 2022. 2, 4, 6

[46] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, 2019. 2, 5, 6, 12

[47] Georgios Pavlakos, Nikos Kolotouros, and Kostas Daniilidis. Texturepose: Supervising human mesh estimation with texture consistency. In *ICCV*, 2019. 3

[48] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *CVPR*, pages 459–468, 2018. 2, 3

[49] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *CVPR*, 2016. 2

[50] Jathushan Rajasegaran, Georgios Pavlakos, Angjoo Kanazawa, and Jitendra Malik. Tracking people with 3d representations. *Advances in Neural Information Processing Systems*, 34:23703–23713, 2021. 3

[51] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of Machine Learning Research (PMLR)*, 2015. 6

[52] Helge Rhodin, Jörg Spörri, Isinsu Katircioglu, Victor Constantin, Frédéric Meyer, Erich Müller, Mathieu Salzmann, and Pascal Fua. Learning monocular 3d human pose estimation from multi-view images. *CVPR*, 2018. 5

[53] Akash Sengupta, Ignas Budvytis, and Roberto Cipolla. Synthetic training for accurate 3d human pose and shape estimation in the wild. In *British Machine Vision Conference (BMVC)*, 2020. 2

[54] Guan Shanyan, Xu Jingwei, Wang Yunbo, Ni Bingbing, and Yang Xiaokang. Bilevel online adaptation for out-of-domain human mesh reconstruction. In *CVPR*, 2021. 1, 3

[55] Leonid Sigal, Alexandru Balan, and Michael Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. 2

[56] Jie Song, Xu Chen, and Otmar Hilliges. Human body model fitting by learned gradient descent. In *ECCV*, 2020. 2

[57] Esteban G. Tabak and Cristina V. Turner. A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66(2), 2013. 6

[58] Esteban G. Tabak and Eric Vanden-Eijnden. Density estimation by dual ascent of the log-likelihood. *Communications in Mathematical Sciences*, 8(1), 2010. 6

[59] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *ECCV*, 2020. 2

[60] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 3, 4

[61] Garvita Tiwari, Dimitrije Antic, Jan Eric Lenssen, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Pose-ndf: Modeling human pose manifolds with neural distance fields. In *ECCV*, 2022. 1, 2, 3, 5, 6

[62] Hsiao-Yu Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *Advances in Neural Information Processing Systems 30*, 2017. 2

[63] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, 2017. 5

[64] Ignas Budvytis Vince Tan and Roberto Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2017. 2

[65] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3D human pose in the wild using IMUs and a moving camera. In *ECCV*, 2018. 2, 4, 5, 8, 12, 13, 15, 16, 17

[66] Tom Wehrbein, Marco Rudolph, Bodo Rosenhahn, and Bastian Wandt. Probabilistic monocular 3d human pose estimation with normalizing flows. In *ICCV*, 2021. 5

[67] Wen-Li Wei, Jen-Chun Lin, Tyng-Luh Liu, and Hong-Yuan Mark Liao. Capturing humans in motion: Temporal-attentive 3d human pose and shape estimation from monocular video. In *CVPR*, June 2022. 2

[68] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[69] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020. 2

[70] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *CVPR*, 2022. 3, 4

[71] Xiangyu Xu and Chen Change Loy. 3D human texture estimation from a single image with transformers. In *Proceedings of the IEEE International Conference on Computer Vision*, 2021. 3, 6

[72] Yuanlu Xu, Song-Chun Zhu, and Tony Tung. Denserac: Joint 3d pose and shape estimation by dense render-and-compare. *ICCV*, 2019. 2

[73] Chun-Han Yao, Jimei Yang, Duygu Ceylan, Yi Zhou, Yang Zhou, and Ming-Hsuan Yang. Learning visibility for robust dense human body estimation. In *ECCV*, 2022. 12

[74] Andrei Zanfir, Eduard Gabriel Bazavan, Hongyi Xu, Bill Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Weakly supervised 3d human pose and shape reconstruction with normalizing flows. In *ECCV*, 2020. 2

[75] Andrei Zanfir, Eduard Gabriel Bazavan, Mihai Zanfir, William T. Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Neural descent for visual 3d human pose and shape. In *CVPR*, 2021. 2

[76] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. Monocular 3d pose and shape estimation of multiple people in natural scenes: The importance of multiple scene constraints. In *CVPR*, 2018. 2

[77] Wang Zeng, Wanli Ouyang, Ping Luo, Wentao Liu, and Xiaogang Wang. 3d human mesh regression with dense correspondence. In *CVPR*, 2020. 2

[78] Hongwen Zhang, Jie Cao, Guo Lu, Wanli Ouyang, and Zhenan Sun. Learning 3d human shape and pose from dense body parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2610–2627, 2020. 2

[79] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In *ICCV*, 2021. 1, 2, 3

[80] Tianshu Zhang, Buzhen Huang, and Yangang Wang. Object-occluded human shape and pose estimation from a single color image. In *CVPR*, 2020. 2, 12

Appendix

A. Implementation Details

Training details. We train our model for 600K steps with a batch size of 8 using Adam [29]. The learning rate is set to 5×10^{-4} and we sample training examples from H36M, 3DPW and SURREAL with probabilities 0.4, 0.3 and 0.3. The images are cropped and resized to 224×224 while maintaining the aspect ratio. Additionally, with a probability of 0.5, we add Gaussian noise to the pose, shape and camera parameters. We select the PARE prediction with probability of 0.3 and take a sample from ProHMR with probability of 0.5. To also focus on fine-grained displacements, we use ground-truth pose with PARE predicted shape and camera parameters with probability of 0.2. Following [33], image data augmentation includes random rotations, scaling and channel-wise pixel noise [33]. Besides, we adopt photometric distortion [6] and for H36M and SURREAL self-mixing [8]. The channel-wise pixel noise is also applied on the texture map.

Preprocessing details. To benchmark our approach, we generate predictions using the latest OpenPose [7] version (v1.7.0) and a state-of-the-art DensePose [19] model[3]. For fair comparison, we feed both models with the images cropped around the target subject using the ground-truth bounding boxes. By transforming the DensePose predictions to points on the SMPL body, they can be used for the reprojection loss [18]. For 3DPW, we use the OpenPose detections included in the dataset. Because RICH only provides SMPL-X bodies, we convert the provided model parameters to SMPL using the official implementation [1].

Runtime. The PyTorch implementation of the displacement field prediction network takes on average 26.4 ms to process one frame on a RTX4090. Running our slightly modified SMPLify [5, 33] implementation for 100 iterations with the reconstructed 2D vertices brings no overhead compared to sparse 2D keypoints and takes around 614 ms and 769 ms with the GMM [5] and VPoser [46] prior respectively. Rendering and transforming the per-pixel 2D displacements to per-vertex displacements is in total done in 1 ms. For faster evaluation, we run SMPLify in batch mode. SMPLify with a batch size of 32 takes around 644 ms and 815 ms with GMM and VPoser pose prior respectively. Note that we did not spend any effort optimizing the runtime of our approach. A highly optimized custom implementation can reduce the fitting time to a few milliseconds [13], which would enable our approach to run in real-time. Additionally, by using the refined estimate of the last frame as initialization for the next frame, the 3D pose regressor would only need to be evaluated once.

B. Additional Results

We provide more qualitative refinement results on images from 3DPW [65] and RICH [22] in Fig. 6 and Fig. 7. We use PARE [31] predictions and SMPLify with VPoser. Our approach generalizes well to different scenes and subjects with varied body shapes, can handle poor lighting and challenging poses, and can even improve fine details such as head rotation.

OpenPose comparison. We show additional visual comparisons with refinements using OpenPose keypoints in Fig. 8. Our approach better refines the reconstruction of the back (row 4, 6), better detects barely visible body parts (row 2, 5, 7) and leads to more accurate depth estimates (row 3). Additionally, body parts that are visible in the initial SMPL prediction but not in the image can be correctly pushed to be occluded in the refinements (row 1) using our approach.

DensePose comparison. We visually compare our refined 3D human models with refinements using DensePose predictions in Fig. 9. Each person pixel detected by DensePose is colorized in the image and shown on the ground-truth human body. While Dense-Pose is good at detecting pixels belonging to a person, the predicted correspondences between the pixels and the 3D SMPL surface lack in accuracy. This is especially noticeable at the boundary between body parts, where no pixels are assigned to even though the regions are visible in the image. Our approach computes more accurate dense correspondences, leading to significantly better refined 3D bodies.

Failure cases. In Fig. 10, we show a few examples where our network fails to estimate reasonable 2D displacement vectors. The scenarios range from (a) extreme occlusion, (b) very poor initial body estimates and (c) close interactions and overlap with other subjects. To improve the performance for large occlusions, it could be helpful to learn visibility masks [73, 80] or per-pixel confidence scores. The problem of wrongly associating limbs could be mitigated by integrating more examples of closely interacting persons in the training set. Finally, in some cases our refinement leads to improved image-model alignment but degrades the 3D pose (see Fig. 11). This is due to the depth ambiguity inherent in monocular 3D motion capture and could be alleviated by regarding multiple images or integrating scene constraints.

Garments with complex texture. When evaluating on the 3DPW test subject with the most complex texture pattern using VPoser and PARE as base model, we achieve an MPJPE of 68.1 and a PVE of 81.2, compared to 75.9 and 90.2 when using OP joints and 75.6 and 88.8 with the base model. Note that most real world cases allow for a cooperative setting where the person is first turning around in front of a camera, which would allow accurate texture estimation even for complex patterns.

Figure 6. Additional results from the 3DPW [65] dataset. From left to right: input images, initial body estimates, our predicted displacement fields, our refined 3D human models and side views of initial, refined and ground-truth bodies.
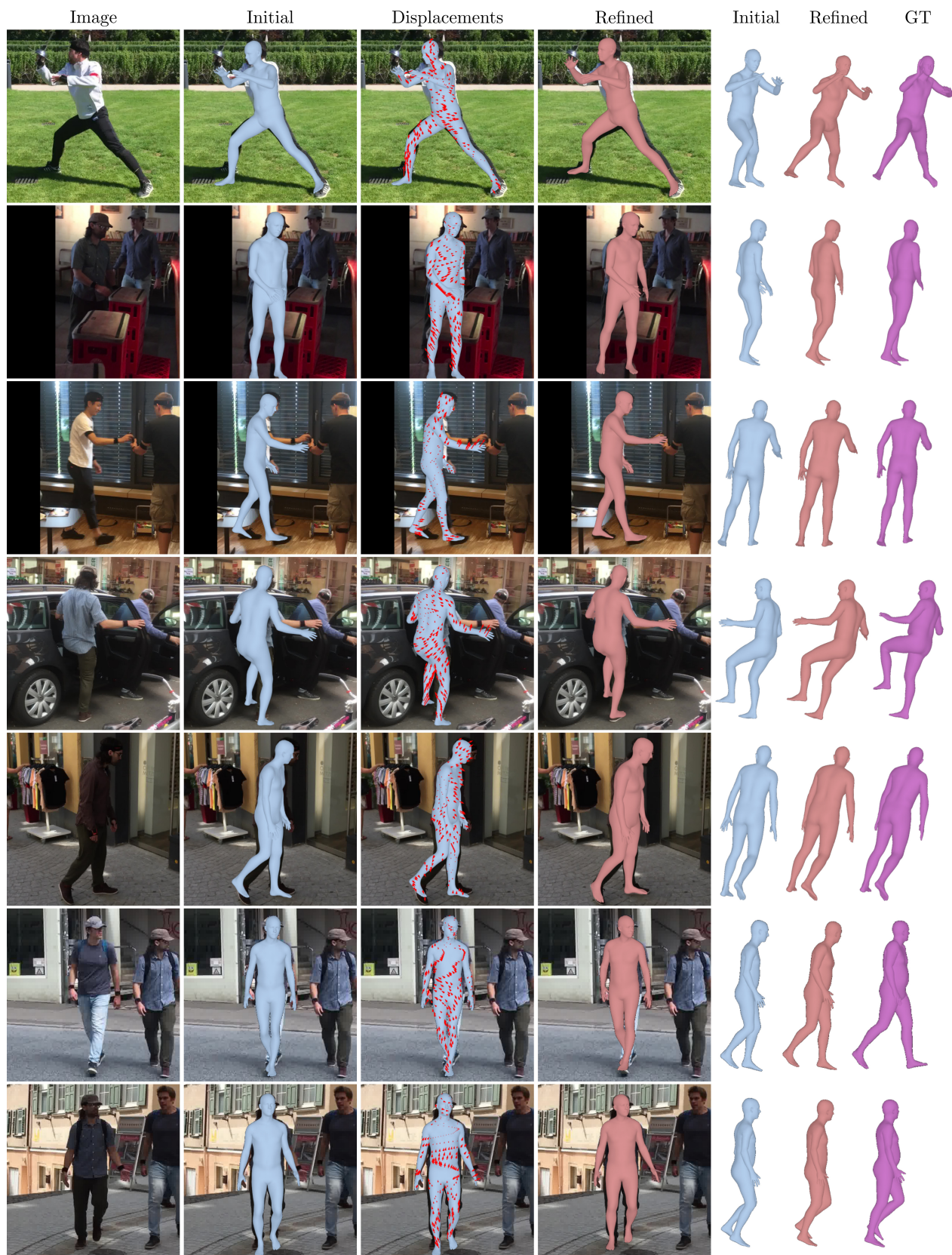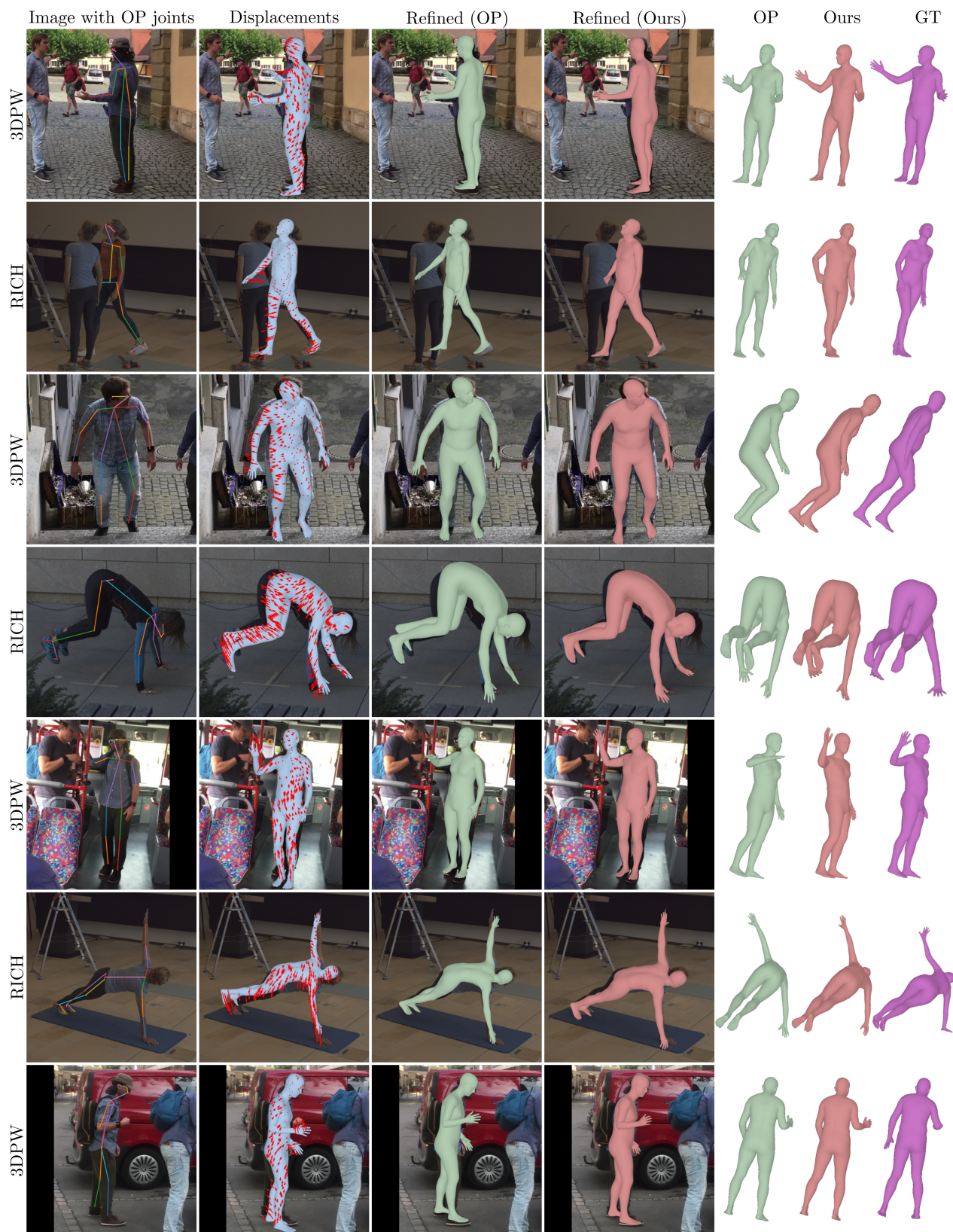
Figure 7. Additional results from the RICH [22] dataset. From left to right: input images, initial body estimates, our predicted displacement fields, our refined 3D human models and side views of initial, refined and ground-truth bodies.

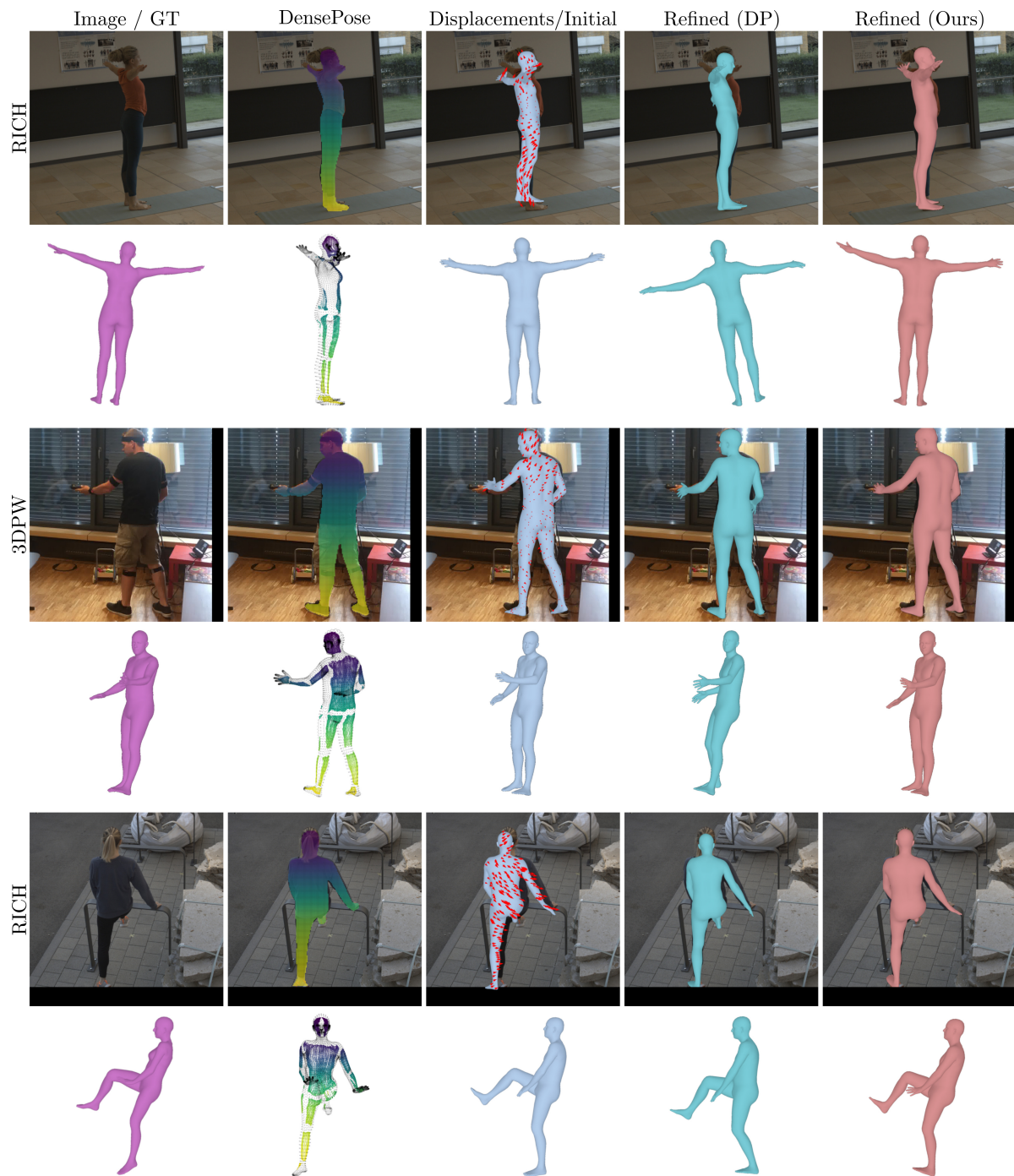Figure 8. Comparison with refinements using OpenPose [7] keypoints on images from 3DPW [65] and RICH [22].

Figure 9. Comparing refinements using DensePose [19] on 3DPW [65] and RICH [22]. Each person pixel detected by DensePose is colorized in the image and shown on the ground-truth human body.

Figure 10. Failure cases of our approach with examples from 3DPW [65] and RICH [22]. (a) Large occlusions may lead to wrong displacement estimates. (b) If the initial estimate is too far away, displacements may not be enough to fit the model. (c) In some cases of close interactions and overlap with other actors the model may wrongly associate limbs.
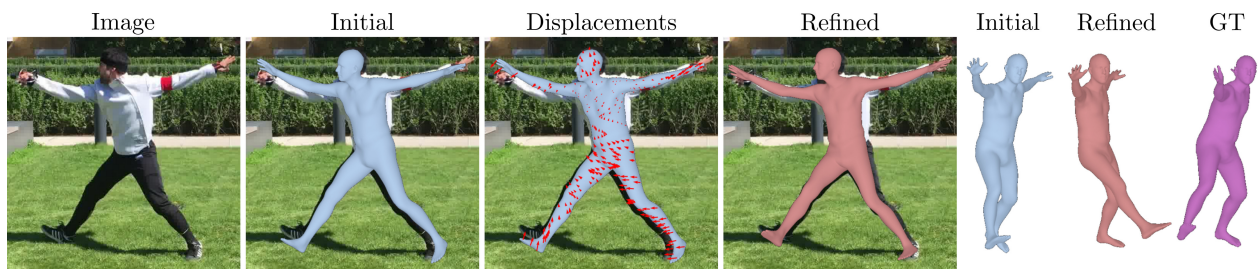

Figure 11. In some cases the 2D alignment may be improved by our approach while leading to a worse 3D pose. Example from 3DPW [65].