Multi-RoI Human Mesh Recovery with Camera Consistency and Contrastive Losses

Yongwei Nie[1], Changzhen Liu[1], Chengjiang Long[2], Qing Zhang[3], Guiqing Li[1], and Hongming Cai[1]

[1] South China University of Technology, China
[2] Meta Reality Labs, USA
[3] Sun Yat-sen University, China
{nieyongwei}@scut.edu.cn

Abstract. Besides a 3D mesh, Human Mesh Recovery (HMR) methods usually need to estimate a camera for computing 2D reprojection loss. Previous approaches may encounter the following problem: both the mesh and camera are *not* correct but the combination of them can yield a low reprojection loss. To alleviate this problem, we define multiple RoIs (region of interest) containing the same human and propose a multiple-RoI-based HMR method. Our key idea is that with multiple RoIs as input, we can estimate multiple local cameras and have the opportunity to design and apply additional constraints between cameras to improve the accuracy of the cameras and, in turn, the accuracy of the corresponding 3D mesh. To implement this idea, we propose a RoI-aware feature fusion network by which we estimate a 3D mesh shared by all RoIs as well as local cameras corresponding to the RoIs. We observe that local cameras can be converted to the camera of the full image through which we construct a local camera consistency loss as the additional constraint imposed on local cameras. Another benefit of introducing multiple RoIs is that we can encapsulate our network into a contrastive learning framework and apply a contrastive loss to regularize the training of our network. Experiments demonstrate the effectiveness of our multi-RoI HMR method and superiority to recent prior arts.

Keywords: Human mesh recovery · RoI · SMPL · Camera estimation

1 Introduction

Since the seminar work of HMR (Human Mesh Recovery) by [19], more and more work attempts to estimate 3D mesh of a human from a single image, for its potential value in VR/AR, virtual try-on and simulative-coaching, etc.

Most of previous work, inspired by [19], treats this task as a regression problem [6,19,27,29,54,63]. They first detect the human from an original full image and use the detected boundingbox to crop the RoI (region of interest) of the human and feed it to a neural network for estimating the target SMPL [37] mesh together with a local camera. The camera is used to project the mesh to the 2D RoI plane, such that the projected mesh can be compared with 2D evidences

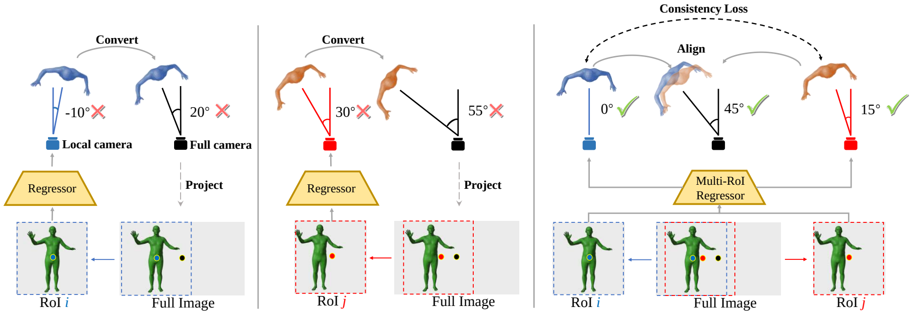

Fig. 1: (a) Extracted RoI i is fed to a regressor but it wrongly estimates a local camera which sees the mesh in -10° while the accurate local camera shall see it in 0°. Consequently, when further converted to full camera, it will wrongly see the mesh in 20° instead of groundtruth 45°. **(b)** As with RoI j, the full camera derived from incorrectly estimated local camera (30°) sees the mesh in 55°. Both (a) and (b) will mislead the 2D-projection loss to output incorrect 3D mesh due to the false projection. **(c)** We feed multiple RoIs into the network simultaneously and estimate local cameras of the RoIs. Both local cameras can be converted to the full camera from the perspective of which the 3D mesh should be aligned. We use this observation to establish pairwise consistency losses between local cameras to obtain accurate local cameras (0° and 15°).

(e.g., poses and human joints) in the given RoI to compute the so-called reprojection loss. However, the reprojection loss may be deceived by the mesh and camera. That is, when the mesh and camera are incorrect, their combination may still yield a low reprojection error.

Through the above analysis, we find that without accurate camera for projection during training, 2D-reprojection loss will be misled, making the network learn incorrect mesh configurations (*e.g.*, wrong global orientation or incorrect joint rotations). To improve the accuracy of mesh, network needs to estimate more accurate camera parameters. Previous approaches for improving cameras can be classified into two categories. The first kind of methods improve the camera projection model. For example in [54], the usually adopted weak-perspective camera is replaced with a perspective-distorted camera model with which the distortion in close-up images can be modeled. In [23, 29], the 3D mesh is projected onto the full image and the reprojection loss is computed in the whole view of the full image. This is because the original RoI-view camera has ambiguity on reasoning about the global orientation of the 3D mesh, while the full-view camera model can resolve this ambiguity. Although this kind of methods can reduce the structural error brought by the inappropriate camera model, they cannot guarantee the camera parameters they estimated are accurate, and still cannot prevent the mesh and camera from deceiving the network. The second kind of methods directly design and impose additional constraints on the camera parameters. For example, the work of [25] collects ground-truth camera rotations and trains a standalone camera estimation network with the estimated cameras being supervised by the ground-truth data.

We propose a different method for improving the accuracy of cameras by imposing additional constraints in a self-supervised manner. Our main finding is that we can extract multiple RoIs of a human by slightly translating and resizing the original RoI of the human. For each of the RoIs, we then compute the camera projecting the 3D mesh onto the corresponding RoI, which is referred to as a local camera. According to [23, 29], the local cameras of RoIs can be converted to the camera of full image coordinate system. Apparently, all RoIs share the same full camera. We then use the full camera as the intermediate bridge to build pairwise consistency losses between local cameras. We illustrates this idea through Figure 1.

With the above motivation, we propose a multiple-RoI-based HMR method. At the core of our method is a RoI-aware feature fusion network. It accepts multiple RoIs of a human as input, equipped with a RoI-guided mechanism extracting and fusing features of the multiple RoIs. We obtain two kinds of fusion features: RoI-shared fusion feature and RoI-specific fusion features. The former is decoded to the 3D mesh shared by all RoIs, and the latter are decoded to parameters of local cameras. We then deduce the pairwise camera consistency losses and impose them on the estimated local cameras to regularize the training of the network. Notably, the introducing of the multiple RoIs allow us to encapsulate our network into the contrastive learning framework as RoIs of the same human shall own similar features, while RoIs of different humans shall output dissimilar features. We propose a contrastive loss to enforce this property, which further improves the performance of our method.

In summary, our method is motivated by a simple intuition about the entanglement of mesh and camera. To solve the problem, we propose to extract multiple RoIs, which is novel in this field as most previous approaches are based on a single RoI. Our contributions are then summarized as:

- We propose a multi-RoI-based HMR method implemented as a RoI-aware feature extraction and fusion network.
- We design two loss functions to guide the training of the network, namely a camera consistency loss and a contrastive loss on the basis of our proposed multiple-RoI setting.
- We conduct comparison and ablation experiments to validate the designs of our method.

2 Related Work

Top-down HMR Methods. Most existing approaches recover human mesh from an image in a top-down manner, *i.e.*, detecting and cropping target person from the image and estimating the human mesh of SMPL-based model [37, 42, 43, 46] in one cropped RoI. There are optimization-based approaches [3, 9, 43], regression-based approaches [19, 20, 26, 45, 53, 65], and hybrid approaches [10, 16, 18, 26–28, 48, 64]. Optimization approaches either directly fit parameters of a

SMPL-based model to 2D joints in the input image [3, 43], or fine-tune a pretrained regression network to match 2D evidences while utilizing priors learned by the network [18]. Different from optimization-based approaches, regression approaches directly train a model to extract features from an input image and map the features to a human mesh model, using CNN [20, 29, 45, 65], GCN [7,22,41], or Transformer [6,8,32,33,51,56]. Many approaches combine regression and optimization methods. For example, work of [18, 26] get an initial prediction through regression-based methods and iteratively optimize the result making it in line with 2D-keypoints reprojection loss. Taking human-kinematics into consideration, work of [27, 28, 48] incorporate Inverse Kinematics Process with the neural network and iteratively update the rotation and location of each joint.

HMR with Multiple Inputs. Considering that HMR is a task with much ambiguity, many methods tend to add more auxiliary information at the input end to better assist the network to reconstruct the accurate body mesh. Some methods manage to estimate the mesh with the aid of extra inputs such as 2D segmentation or silhouettes of the target human [12, 24, 57, 59, 62, 67] which help the network grasp and understand the human bodies in images with those guidance. Work of [36,38,61] try to utilize available sparse 3D markers on surface of the target human before full-body reconstruction and complete the dense human meshes through optimization or interpolation. There are also multi-view methods, by which the ambiguity of HMR is alleviated since multiple view angles and camera parameters are available [30, 44, 47, 49]. A large number of temporal (video-based) methods incorporate auxiliary inputs as well, such as trajectory [11, 60], optical flows [31] and 3D scene point cloud [66]. There is also egocentric work [35] using an extra scaled RoI to aid the network to estimate SMPL poses.

Approaches Improving Cameras. There has been much work focusing on the camera projection model since it is the vital bridge between the 2D image and the 3D mesh. Based on SPIN [26] and Simplify [3], the work of [23] optimizes the full perspective camera of the original full image for the first time. [25] tries to estimate the real camera pitch and yaw angles along with the mesh prediction, instead of using identity rotation like most HMR methods uses. [54] introduces a new dataset and copes with the scenario where people are shown up close in the image, taking the distortion of perspective projection into consideration. CLIFF [29] digs deeper into the full-image reprojection and uses boundingbox information in order to guide the network towards the accurate full camera. We, in this paper, incorporate the theory of full-image projection in [23, 29] and model the pairwise relations between cameras estimated from different RoIs of the same person.

Contrastive learning. Contrastive learning [4, 21] is a type of unsupervised learning that aims to learn a similarity pattern between data samples. The goal of contrastive learning is to pull similar examples closer while pushing others away. This goal is accomplished via a contrastive constraint on the triplet of $(anchor, positive, negative)$, where anchor sample can be one of the samples in the mini batch and positive samples are the similar samples with the anchor while negative samples are the dissimilar ones.

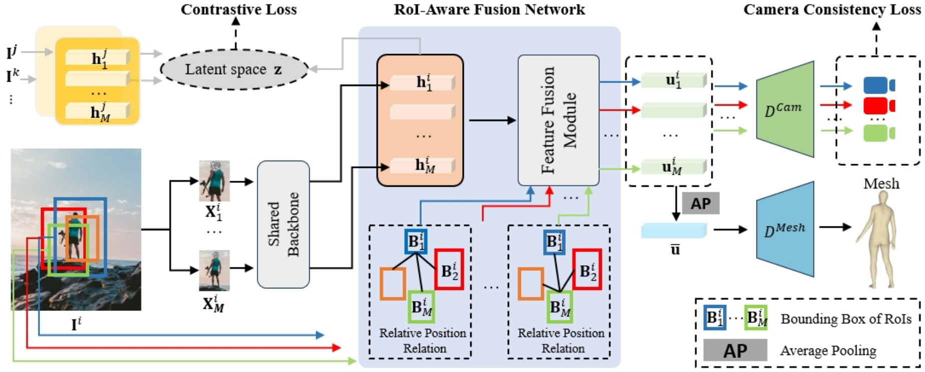

Fig. 2: Overview of our method. Given an image, we extract multiple RoIs of a human, and use a RoI-aware feature fusion network to estimate the 3D mesh of the human together with cameras. We use a camera consistency loss and a contrastive loss to supervise the training of the network.

3 Method

Figure 2 provides the overview of our method. Given a full image $\mathbf{I}^i$, we extract M RoIs $\{\mathbf{X}_m^i\}_{m=1}^M$ of a person in the image by different boundingboxes $\{\mathbf{B}_m^i\}_{m=1}^M$, and use a shared backbone network to extract features $\{\mathbf{h}_m^i\}_{m=1}^M$ from the RoIs. Then, we propose a RoI-aware fusion network to fuse $\{\mathbf{h}_m^i\}_{m=1}^M$ to obtain RoI-specific fusion features $\{\mathbf{u}_m^i\}_{m=1}^M$ and a RoI-shared fusion feature $\bar{\mathbf{u}}$. Each RoI-specific feature is individually decoded by D^{cam} to a local camera, obtaining M local cameras to which we apply camera consistency loss. The RoI-shared feature is decoded by D^{mesh} to the target 3D mesh. We also extract features from RoIs of other objects (*e.g.*, in $\mathbf{I}^j$, $\mathbf{I}^k$) and project all of them into the latent space of $\mathbf{z}$, and finally apply a contrastive loss in $\mathbf{z}$-space.

Formally, the regression task in this paper is formulated as:

$$\theta, \beta, \{\mathbf{C}_m\}_{m=1}^M = f(\{\mathbf{X}_m\}_{m=1}^M, \{\mathbf{B}_m\}_{m=1}^M), \tag{1}$$

where $\theta \in \mathbb{R}^{24\times 3}$ determines the pose of the SMPL mesh, $\beta \in \mathbb{R}^{10}$ determines the shape of the SMPL mesh, and $\mathbf{C_m} = (s_m, t_{x_m}, t_{y_m})$ contains scale s_m and translation parameters (t_{x_m}, t_{y_m}) determining a weak-perspective camera that projects the predicted 3D mesh onto the 2D RoI plane. $\mathbf{B}_m = (c_{x_m}, c_{y_m}, b_m)$, where (c_{x_m}, c_{y_m}) is the location of the boundingbox in the full image, and b_m is the width of the boundingbox.

3.1 RoI-aware Feature Fusion Network

To be specific, given $\{\mathbf{X}_m\}_{m=1}^M$ (the superscript i used in Figure 2 is dropped for simplicity), we use a shared encoder E to extract features from RoIs, *i.e.*, $\mathbf{h}_m = E(\mathbf{X}_m)$ for $m \in [1, M]$. The encoder E can be ResNet50 [14] or HRNet48

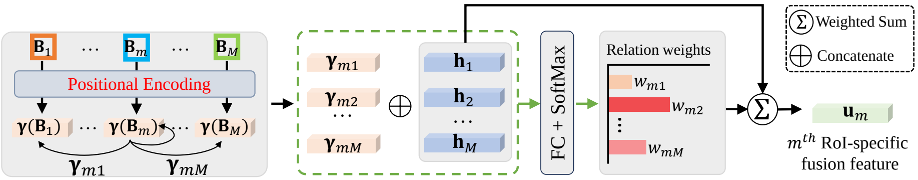

Fig. 3: RoI-aware fusion. To obtain $\mathbf{u}_m$, we consider the relative relation of other boundingboxes to the m^{th} boundingbox. We perform positional encoding to all the boundingboxes and then compute relative position relation γ_{m*} (where $*$ is a number in $[1, M]$). We then concatenate γ_{m*} and the corresponding feature $\mathbf{h}_*$ to compute weight w_{m*}. Finally, $\mathbf{u}_m$ is the weighted sum of $\{\mathbf{h}_m\}_{m=1}^M$ with w_{m*} as the weights.

[50] as employed in previous approaches [2,6,19,29]. After that, we design a RoI-aware fusion network to fuse $\{\mathbf{h}_m\}_{m=1}^M$, obtaining RoI-specific fusion features $\{\mathbf{u}_m\}_{m=1}^M$. Then, we simply average the RoI-specific features by AP (Average Pooling) to obtain the RoI-shared feature $\bar{\mathbf{u}}$.

The core of our network is the feature fusion module. To begin with, the feature $\mathbf{h}_m$ only contains the information about the m^{th} RoI. Since different RoIs contain different visual details about the target person, we fuse all features $\{\mathbf{h}_m\}_{m=1}^M$ together for reasoning about the mesh and cameras. Our fusion method, as illustrated in Figure 3, leverages the boundingbox information $\{\mathbf{B}_m\}_{m=1}^M$, by which we compute the *relative* position relation between the boundingboxes to align the features of different RoIs. Specifically, the relative position relation is simply computed as the pairwise difference between boundingboxes after positional encoding:

$$\gamma_{mn} = \gamma(\mathbf{B}_m) - \gamma(\mathbf{B}_n), \tag{2}$$

where $\gamma(\cdot)$ is the position encoding function [40, 51]:

$$\gamma(p) = (p, \sin(\pi p), \cos(\pi p), \cdots, \sin(2^L \pi p), \cos(2^L \pi p), \tag{3}$$

which is applied to each of the three variables of $\mathbf{B}_m$ (or $\mathbf{B}_n$). We set $L = 32$ in this paper. Then, taking the m^{th} RoI as example, the way to compute the fused feature $\mathbf{u}_m$ is:

$$\begin{aligned} \mathbf{u}_m &= \sum_{n=1}^{M} w_{mn} \mathbf{h}_n, \\ \{w_{mn}\}_{n=1}^M &= \mathrm{Softmax}(\mathrm{Linear}(\{\mathbf{F}_{mn}\}_{n=1}^M)) \\ \mathbf{F}_{mn} &= \mathrm{Concat}(\mathbf{h}_n, \gamma_{mn}), \end{aligned} \tag{4}$$

where $w_{mn} \in [0, 1]$ is a scalar used to fuse features of multiple RoIs. To compute w_{mn}, we first concatenate the feature $\mathbf{h}_n$ and the relative position relation γ_{mn}, and then send the concatenated feature to a linear layer to obtain a scalar, which is finally converted to w_{mn} by a Softmax function. Eventually, we use a D^{cam},

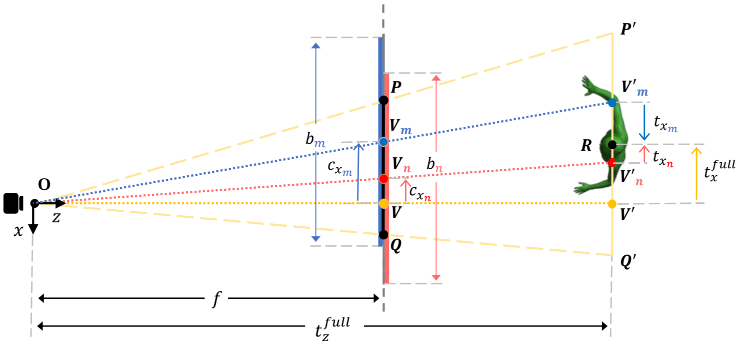

Fig. 4: Conversion between local and full cameras in bird's eye view.

which is composed of FC layers with residual connections as adopted in [19], to compute the local camera $\mathbf{C}_m$ from the feature $\mathbf{u}_m$:

$$\mathbf{C}_m = D^{cam}(\mathbf{u}_m). \tag{5}$$

We employ D^{mesh} similar to D^{cam} to map the averaged feature $\bar{\mathbf{u}}$ to 3D mesh:

$$\theta, \beta = D^{mesh}(\bar{\mathbf{u}}). \tag{6}$$

3.2 Camera Consistency Loss

To build camera consistency loss, local cameras are converted to the coordinate system of full camera which we use to establish the constraints between each pair of local cameras. Formally, let $\mathbf{C}_m = (s_m, t_{x_m}, t_{y_m})$ and $\mathbf{C}_n = (s_n, t_{x_n}, t_{y_n})$ be two local cameras estimated from two RoIs cropped by boundingboxes $\mathbf{B}_m = (c_{x_m}, c_{y_m}, b_m)$ and $\mathbf{B}_n = (c_{x_n}, c_{y_n}, b_n)$, respectively. Let $\mathbf{C}_{full} = (t_x^{full}, t_y^{full}, t_z^{full})$ be the parameters of the full camera. The focal length of the full camera is denoted as f. We refer to Figure 4 to interpret the conversion between these variables.

Figure 4 shows that a 3D mesh at distance t_z^{full} from the camera O is projected onto the image plane in a focal length of f. We assume the 3D human mesh is bounded in a 2m-2m-2m box. From the bird's eye view, we use $P'Q'$ to denote the valid region occupied by the 3D mesh, and the length of $P'Q'$ is 2m, which is projected to PQ on the image plane. The blue line on the image plane indicates the RoI $\mathbf{B}_m$, the length of which is b_m, *i.e.*, the width of the bounding box. Since $\triangle OPQ$ and $\triangle OP'Q'$ are similar, we have:

$$\frac{PQ}{P'Q'} = \frac{f}{t_z^{full}}, \quad i.e., \quad \frac{b_m \cdot s_m}{2} = \frac{f}{t_z^{full}}, \tag{7}$$

where $b_m \cdot s_m$ is the length of PQ. And we get,

$$t_z^{full} = \frac{2 \cdot f}{b_m \cdot s_m}. \tag{8}$$

On the other hand, we have the center of $\mathbf{B}_m$, *i.e.*, V_m, on the image plane. The distance from V (the image center) to V_m is c_{x_m}. Let V'_m be the point on the mesh plane corresponding to V_m, and R be the center of mesh. The distance from V'_m to R is the local camera translation t_{x_m}. The distance from V' to R is the full camera translation t_x^{full}. Since $\triangle OVV_m$ and $\triangle OV'V'_m$ are similar, we have:

$$\frac{VV_m}{V'V'_m} = \frac{f}{t_z^{full}}, \quad i.e., \quad \frac{c_{x_m}}{t_x^{full} - t_{x_m}} = \frac{f}{t_z^{full}}. \tag{9}$$

Combining Eq. 8 with Eq. 9, we get

$$t_x^{full} = t_{x_m} + \frac{2 \cdot c_{x_m}}{b_m \cdot s_m} \tag{10}$$

The above is the relation between local translation t_{x_m} and global translation t_x^{full}. Similarly, the relation along the y axis is:

$$t_y^{full} = t_{y_m} + \frac{2 \cdot c_{y_m}}{b_m \cdot s_m}. \tag{11}$$

From Eq. 10, 11 and 8, we convert the local camera $\mathbf{C}_m$ to the full camera $\mathbf{C}_{full}$ by:

$$t_x^{full} = t_{x_m} + \frac{2 \cdot c_{x_m}}{b_m \cdot s_m}, \quad t_y^{full} = t_{y_m} + \frac{2 \cdot c_{y_m}}{b_m \cdot s_m}, \quad t_z^{full} = \frac{2 \cdot f}{b_m \cdot s_m}. \tag{12}$$

Similarly, we can convert local camera $\mathbf{C}_n$ to the full camera $\mathbf{C}_{full}$ by:

$$t_x^{full} = t_{x_n} + \frac{2 \cdot c_{x_n}}{b_n \cdot s_n}, \quad t_y^{full} = t_{y_n} + \frac{2 \cdot c_{y_n}}{b_n \cdot s_n}, \quad t_z^{full} = \frac{2 \cdot f}{b_n \cdot s_n}. \tag{13}$$

Combining Eq. 12 and 13, we establish the following relations between parameters of local cameras:

$$\begin{cases} t_{x_m} + \frac{2 \cdot c_{x_m}}{b_m \cdot s_m} = t_{x_n} + \frac{2 \cdot c_{x_n}}{b_n \cdot s_n} \\ t_{y_m} + \frac{2 \cdot c_{y_m}}{b_m \cdot s_m} = t_{y_n} + \frac{2 \cdot c_{y_n}}{b_n \cdot s_n} \\ b_m \cdot s_m = b_n \cdot s_n \end{cases} \tag{14}$$

We define

$$\begin{cases} \mathcal{L}_x(m,n) = \left\| \left(t_{x_m} + \frac{2 \cdot c_{x_m}}{b_m \cdot s_m} \right) - \left(t_{x_n} + \frac{2 \cdot c_{x_n}}{b_n \cdot s_n} \right) \right\|_2^2 \\ \mathcal{L}_y(m,n) = \left\| \left(t_{y_m} + \frac{2 \cdot c_{y_m}}{b_m \cdot s_m} \right) - \left(t_{y_n} + \frac{2 \cdot c_{y_n}}{b_n \cdot s_n} \right) \right\|_2^2 \\ \mathcal{L}_s(m,n) = \| b_m \cdot s_m - b_n \cdot s_n \|_2^2 \end{cases} \tag{15}$$

Finally, the local camera consistency loss is defined as:

$$\mathcal{L}_{cam} = \sum_{m,n}^{M} \lambda_x \mathcal{L}_x(m,n) + \lambda_y \mathcal{L}_y(m,n) + \lambda_s \mathcal{L}_s(m,n), \tag{16}$$

where λ_x, λ_y and λ_s are weights of the three regularization terms, which are 0.1, 0.1 and 0.0001, respectively.

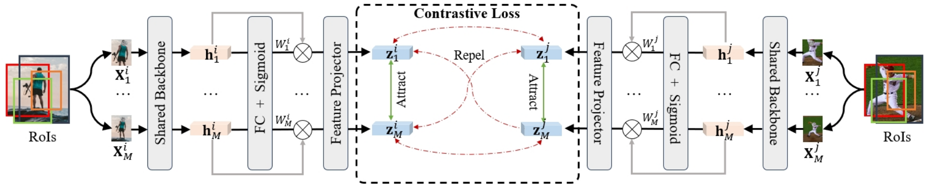

Fig. 5: Contrastive Loss. Taking RoIs $\{\mathbf{X}_m^i | m \in [1, M]\}$ of object i and RoIs $\{\mathbf{X}_m^j | m \in [1, M]\}$ of object j as example, features $\{\mathbf{h}_m^i | m \in [1, M]\}$ and $\{\mathbf{h}_m^j | m \in [1, M]\}$ are first extracted by the shared backbone E from the RoIs, respectively. Then the features are further projected into the latent space $\mathbf{z}$, obtaining $\{\mathbf{z}_m^i | m \in [1, M]\}$ and $\{\mathbf{z}_m^j | m \in [1, M]\}$. The latent features from the same object attract each other, while latent features from different objects repel each other.

3.3 Contrastive Loss

An extra benefit of using multiple RoIs as input is that we can apply a contrastive loss as another regularization term besides the camera consistency loss. At training, we have access to RoIs of different persons. It is natural to require extracting similar features from RoIs of the same person. While for RoIs of different persons, different features should be extracted. The contrastive learning of [4] can be adapted to fulfill this purpose.

Let $\{\mathbf{X}_m^i | m \in [1, M]\}$ be RoIs of object i with $i \in [1, N]$ where N is the number of objects in a training batch. We first extract features from all the RoIs, obtaining $\{\mathbf{h}_m^i | i \in [1, N], m \in [1, M]\}$. Then we further project the features into a latent space $\mathbf{z}$, obtaining latent features $\{\mathbf{z}_m^i | i \in [1, N], m \in [1, M]\}$. Figure 5 illustrates the the mapping process from $\mathbf{X}$ to $\mathbf{z}$. The contrastive loss is defined on all the latent features:

$$\mathcal{L}_{cont} = \sum_{i=1}^{N} \sum_{m=1}^{M} \frac{-1}{M-1} \sum_{n=1, n \neq m}^{M} \log \frac{\exp(\mathbf{z}_m^i \cdot \mathbf{z}_n^i / \tau)}{\sum\limits_{i'=1, i' \neq i}^{N} \sum\limits_{m'=1}^{M} \exp(\mathbf{z}_m^i \cdot \mathbf{z}_{m'}^{i'} / \tau)}, \tag{17}$$

where $\tau = 0.5$ is the temperature parameter. The numerator/denominator aims at (1) minimizing cosine distance between features $\mathbf{z}_m^i$ and $\mathbf{z}_n^i$ from the same object i, and (2) maximizing distance between features $\mathbf{z}_m^i$ and $\mathbf{z}_{m'}^{i'}$ from different objects i and i'.

3.4 Total Training Loss

Besides the the camera consistency loss in Eq. 16 and contrastive loss in Eq. 17, we also adopt the typical losses using GT mesh and 2D joints as supervision, including:

$$\begin{aligned} \mathcal{L}_{smpl} = \left\| \Theta - \hat{\Theta} \right\|, \quad \mathcal{L}_{vert} = \left\| V^{3D} - \hat{V}^{3D} \right\|, \\ \mathcal{L}_{3D} = \left\| J^{3D} - \hat{J}^{3D} \right\|_2^2, \quad \mathcal{L}_{2D} = \sum_{m}^{M} \left\| J_m^{2D} - \hat{J}^{2D} \right\|_2^2, \end{aligned} \tag{18}$$

where $\Theta = (\theta, \beta)$ denotes estimated SMPL parameters and $\hat{\Theta}$ is the ground truth (GT), V^{3D} indicates 3D vertices of human mesh with $\hat{V}^{3D}$ as GT, and J^{3D} denotes the 3D joints of the human with $\hat{J}^{3D}$ as GT. For the 2D reprojection loss, J_m^{2D} is obtained by projecting J^{3D} from 3D to 2D with the full camera deduced from local camera $\mathbf{C}_m$. Following [29], the projected joints are compared with the GT 2D joints $\hat{J}^{2D}$ in the full image. The total loss function is:

$$\mathcal{L}_{total} = \lambda_{cam}\mathcal{L}_{cam} + \lambda_{cont}\mathcal{L}_{cont} + \lambda_{smpl}\mathcal{L}_{smpl} + \lambda_{vert}\mathcal{L}_{vert} + \lambda_{3D}\mathcal{L}_{3D} + \lambda_{2D}\mathcal{L}_{2D}, \quad (19)$$

where λ_* are weights for each loss component and we set them following SPIN [26] except λ_{cont} and λ_{cam}, which are 0.1 and 1, respectively.

3.5 Extraction of RoIs

Given an image, we use methods of [13, 17] to detect boundingboxes of human. Let $\mathbf{B} = (c_x, c_y, b)$ be a boundingbox, we slightly resize/translate the the boundingbox to select multiple RoIs of the human. One can randomly generate the resizing factors or translation offsets. However, experiments (see Supp.) show that fixing these parameters during training gives better results. Specifically, the offset along the x and y axes includes $\{(0.1b, 0), (-0.1b, 0), (0, 0.1b), (0, -0.1b)\}$, and the corresponding resizing factors are $\{1.5, 1.25, 0.8, 0.65\}$. Together with the original boundingbox, we totally extract $M = 5$ RoIs for a person from the full image. More detailed illustrations are provided in supplemental material.

4 Experiments

4.1 Datasets and Metrics

To conduct fair comparison between our method and SOTA methods, we follow the dataset setting used in SOTA works [2, 5, 6, 24, 28, 64]. Specifically, we train our method on a mixture of four datasets including Human3.6M [15], MPI-INF-3DHP [39], COCO [34], and MPII [1].

As for evaluation, we use the test sets of 3DPW [52] and Human3.6M [15]. Following prior works, we finetune our model on 3DPW train set when evaluating on its test set. [1]

We use MPJPE (Mean Per Joint Position Error [15]), PA-MPJPE (Procrustes-Aligned MPJPE [68]) and PVE (the mean Euclidean distance between mesh vertices) as the evaluation metrics.

4.2 Implementation Details

We implement our method using PyTorch. For the shared backbone, we use ResNet-50 [14] extracting features of $d = 2048$ dimensions and HRNet-W48 [50]

[1] The authors Changzhen Liu and Yongwei Nie signed the license and produced all the experimental results in this paper. Meta did not have access to the datasets.

Table 1: Quantitative comparison with SOTA methods. $R50$ (or $R34$) denotes using ResNet [14] as backbone. $H48$ (or $H32$, $H64$) denotes using HRNet [50]. Note that we present the result of Zolly^{H48} trained without synthetic distorted data for fairness, as reported in their paper.

Method	3DPW			Human3.6M	
	MPJPE	PA-MPJPE	PVE	MPJPE	PA-MPJPE
HMR^{R50} [19]'18	116.5	72.6	–	–	56.8
SPIN^{R50} [26]'19	96.9	59.2	116.4	–	41.1
SPEC^{R50} [25]'21	96.4	52.7	–	–	–
PyMAF^{R50} [64]'21	92.8	58.9	110.1	57.7	40.5
PARE^{R50} [24]'21	82.9	52.3	99.7	–	–
PARE^{H32} [24]'21	74.5	46.5	88.6	–	–
HybrIk^{R34} [28]'21	74.1	45.0	86.5	55.4	33.6
FastMETRO^{R50} [6]'22	77.9	48.3	90.6	53.9	37.3
FastMETRO^{H64} [6]'22	73.5	44.6	84.1	52.2	33.7
CLIFF^{R50} [29]'22	71.4	45.4	84.2	50.2	35.9
CLIFF^{H48} [29]'22	69.0	43.0	81.2	47.1	32.7
MPT^{H48} [32]'22	65.9	42.8	79.4	45.3	31.7
PLIKS^{H32} [48]'23	66.9	42.8	82.6	49.3	34.7
BoPR^{H48} [5]'23	65.4	42.5	80.8	–	–
ReFit^{H48} [55]'23	65.8	41.0	–	48.4	32.2
Deformer^{R50} [58]'23	–	–	–	50.7	36.3
Deformer^{H48} [58]'23	72.9	44.3	82.6	44.8	31.6
PyMAF-X^{R50} [63]'23	76.8	46.8	88.7	58.1	40.2
PyMAF-X^{H48} [63]'23	74.2	45.3	87.0	54.2	37.2
NIKI^{H48} [27]'23	71.3	40.6	86.6	–	–
Zolly^{R50} [54]'23	72.5	44.1	84.3	52.7	34.2
Zolly^{H48} w/o PD [54]'23	67.2	40.9	**78.4**	49.4	32.3
Ours^{R50}	**68.2**	**43.2**	**81.9**	**45.4**	**33.0**
Ours^{H48}	**64.1**	**40.4**	78.6	**42.2**	**30.7**

extracting features of $d = 720$ dimensions, and refer to our methods with these backbones as Ours^{R50} and Ours^{H48}, respectively. Following [2,54], the adopted backbones of ResNet-50 and HRNet-W48 are pre-trained on COCO [34] for 2D pose estimation. We train our models with a learning rate of 1e-4 and 5e-5 for ResNet and HRNet backbones respectively, both scheduled by an Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The batchsize for Ours^{R50} is 48 and that for Ours^{H48} is 20. Training with ResNet-50 takes 25 epochs for 1 day and training with HRNet-W48 takes 15 epochs for 2 days on NVIDIA RTX 3090. When finetuning on 3DPW, we fix the learning rate at 1e-5 (for both backbones) to train our models for another 5 epochs. By default, we use $M = 5$ RoIs.

4.3 Comparison to Prior Arts

Table 1 provides quantitative comparisons with SOTA approaches. We compare with IK-based approaches [27,28], iterative fitting approaches [55,63,64], Transformer-based approaches [6,58], and approaches improving camera [29,54], etc. As seen, our method, either with a HRNet backbone or with a ResNet backbone, has better performance on the two evaluation datasets than the corresponding compared approaches. Please pay attention to the comparison between

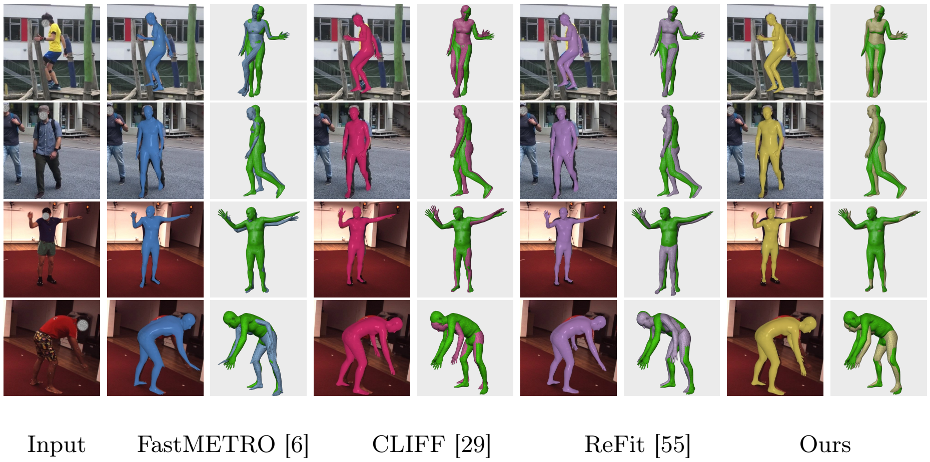

Fig. 6: Qualitative comparison with SOTA approaches. Our results align better with the GT mesh (green) than other results.

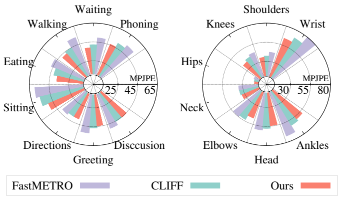

Fig. 7: Per action (left) or joint (right) MPJPE comparison with FastMETRO [6] and CLIFF [29] on Human3.6M.

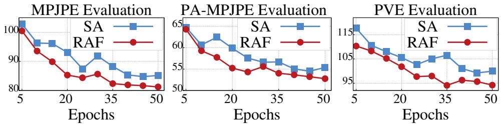

Fig. 8: Comparison between self attention (SA) fusion and our relative-relation-based fusion (RAF). Accuracy at different training epochs are shown.

our method and CLIFF [29], as our method is implemented based on CLIFF. Taking the backbone of HRNet-W48 as an example, the margin on MPJPE between CLIFF and ours is 5mm, which is a large improvement considering CLIFF is a very strong baseline. When compared with Zolly [54] and NIKI [27], our method works well in terms of all the three evaluation metrics, while they are competitive in terms of PA-MPJPE but not MPJPE or PVE. Our method performs well on both testing datasets, while approaches such as PLIKS [48] and ReFit [55] show advantages on 3DPW but not Human3.6M.

Figure 6 shows qualitative comparisons between our method and SOTA approaches. The shown cases are challenging, containing either complex poses or showing occlusions by other body parts. For these cases, our estimated meshes resemble the GT (green color) better than results of the compared approaches. Figure 7 shows the action and per joint comparisons on Human3.6M. Our method outperforms FastMETRO and CLIFF on all kinds of actions and joints.

4.4 Ablation Study

In this section, we conduct ablation studies on the core design ideas of our method. All the following ablation studies are performed on the COCO training dataset and tested on 3DPW following previous literature [24, 29], if not otherwise specified.

Ablation on Design Components. Our method is composed of three major components: RoI-aware fusion module (RAF), camera consistency loss ($\mathcal{L}_{cam}$) and contrastive loss ($\mathcal{L}_{cont}$). To show the effect of each component, we remove each of them at a time while maintaining the other two components (ablations of removing two components are provided in the Supp.). Table 2 shows that removing any component incurs an apparent performance drop. Especially, when we drop $\mathcal{L}_{cam}$, the MPJPE increases by 3mm while PA-MPJPE stays low, indicating that $\mathcal{L}_{cam}$ assists to predict more accurate mesh orientation by improving cameras.

Importance of Relative Relation and Positional Encoding. As discussed in Section 3.1, we rely on relative relation for the RoI-aware feature fusion (also denoted as RAF), and the relation is computed based on the positional encoding (PE) of the bounding boxes. Both of them are critical to our method, as shown in Table 3: (1) $\mathbf{h}_*\oplus$ NULL: concatenating nothing, *i.e.*, using only feature $\mathbf{h}_*$ for computing relation weights, where $*$ is a number in $[1, M]$. (2) $\mathbf{h}_* \oplus \gamma(\mathbf{B}_*)$: simply concatenating PE of the corresponding boundingbox. (3) $\mathbf{h}_* \oplus \gamma_{m*}$: concatenating relative PE γ_{m*} for computing m^{th} fused feature. We also test the above three setups with different length of PE, denoted as L. As seen, using relative PE with $L = 32$ yields the best results.

We also implemented RAF by performing self attention [51] on M tokens of $\{\mathbf{h}_m \oplus \gamma(\mathbf{B}_m)\}_{m=1}^{M}$. Here we can only concatenate PE but not relative PE, since there is only M tokens but we have M^2 relative relations (see supplemental material for details). Results are shown in Figure 8, where our relative-PE based scheme *i.e.*, RAF, outperforms the self-attention approach.

Number of RoIs. We conduct an ablation study that gradually increases the number of input RoIs in Table 4. As seen, the accuracy is consistently increased as the number of input RoIs increases. Experiments of inputting 6 or more RoIs are not conducted due to memory limit. We find that as the RoI number increases, the loss $\mathcal{L}_{2D}$ in Eq. 19 increases while higher regression accuracy can be obtained. This indicates that inputting more RoIs may prevent the network from over-fitting.

Inferring Speed. We also report the inferring speed in Table 4. As the number of RoIs increases, the inferring speed is just slightly decreased. With 5 RoIs, our method processes 55.6 frames per second, which is fast.

4.5 Limitations

Figure 9 shows failure cases, where both our method and CLIFF produce nearly perfect reprojection results without noticeable misalignment in the 2D image. But in the 3D space, some body parts of both methods deviate from the ground

Table 2: Ablation on core components of our method.

RAF	$\mathcal{L}_{cam}$	$\mathcal{L}_{cont}$	MPJPE	PA-MPJPE
✗	✗	✗	87.0	55.8
✗	✓	✓	83.8	54.8
✓	✗	✓	83.1	52.0
✓	✓	✗	83.4	53.3
✓	✓	✓	**80.8**	**51.9**

Table 3: Importance of Relative Relation and Positional Encoding (PE).

Concatenation	L of PE	MPJPE	PA-MPJPE
$\mathbf{h}_* \oplus$ NULL	–	98.7	56.9
$\mathbf{h}_* \oplus \gamma(\mathbf{B}_*)$	$L = 32$	87.6	55.0
$\mathbf{h}_* \oplus \gamma_{m*}$	$L = 0$	81.6	52.9
$\mathbf{h}_* \oplus \gamma_{m*}$	$L = 32$	**80.8**	**51.9**
$\mathbf{h}_* \oplus \gamma_{m*}$	$L = 64$	81.1	52.3

Table 4: Ablation on number of input RoIs. Five is the default.

RoI Num	1 RoI	2 RoIs	3 RoIs	4 RoIs	5 RoIs
MPJPE (mm)	84.6	82.9	81.2	82.2	**80.8**
PA-MPJPE (mm)	54.2	54.4	52.3	53.1	**51.9**
Final $\mathcal{L}_{2D}$ ($\times 1e-3$)	**1.6**	1.8	1.9	2.3	2.4
Inferring speed (fps)	**76.3**	63.5	61.7	57.3	55.6

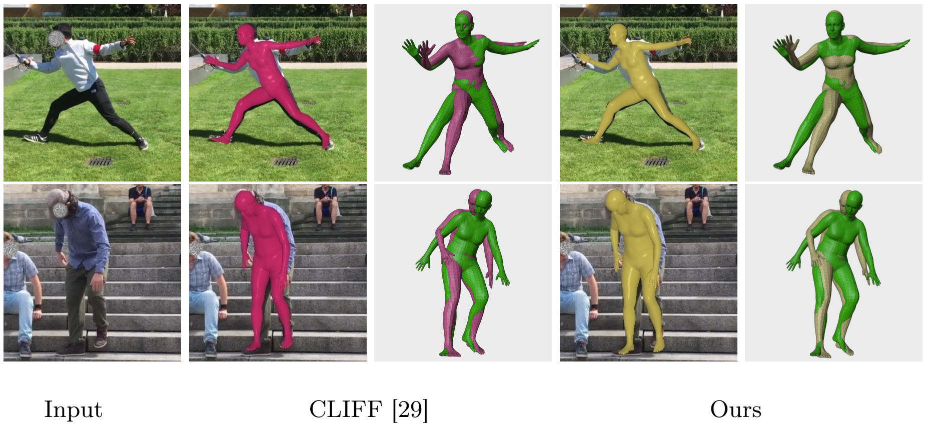

Fig. 9: Failure examples. Both our method and CLIFF produce nearly perfect results in the 2D image, but not in the 3D space.

truth. These examples show that introducing multiple RoIs still cannot solve the ill-posed problem that there may be multiple 3D meshes matching with the same 2D configuration.

5 Conclusion and Future Work

This paper digs into the relation among different RoIs of the same person in an image for human mesh recovery. With the multiple RoIs indicated by different boundingboxes, we are able to design a multi-RoI fusion network to estimate reliable camera parameters, thanks to the additional visual information and pairwise relation provided by the multiple inputs. Specifically, we have exploited using relative-position-relation guided feature fusion, camera consistency loss and contrastive loss to take advantage of the information in multiple inputs as much as possible. We validate the effectiveness of each proposed component using experiments and prove our method has better regression accuracy than current SOTA approaches on popular benchmarks and datasets. In the future, it is valuable to investigate whether the proposed strategies are effective in multi-view or video-based HMR.

References

1. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: Proceedings of the IEEE Conference on computer Vision and Pattern Recognition. pp. 3686–3693 (2014)
2. Black, M.J., Patel, P., Tesch, J., Yang, J.: Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8726–8737 (2023)
3. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14. pp. 561–578. Springer (2016)
4. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: Simclr: A simple framework for contrastive learning of visual representations. In: International Conference on Learning Representations. vol. 2 (2020)
5. Cheng, Y., Huang, S., Ning, J., Shan, Y.: Bopr: Body-aware part regressor for human shape and pose estimation. arXiv preprint arXiv:2303.11675 (2023)
6. Cho, J., Youwang, K., Oh, T.H.: Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In: European Conference on Computer Vision. pp. 342–359. Springer (2022)
7. Choi, H., Moon, G., Lee, K.M.: Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16. pp. 769–787. Springer (2020)
8. Dou, Z., Wu, Q., Lin, C., Cao, Z., Wu, Q., Wan, W., Komura, T., Wang, W.: Tore: Token reduction for efficient human mesh recovery with transformer. arXiv preprint arXiv:2211.10705 (2022)
9. Fan, T., Alwala, K.V., Xiang, D., Xu, W., Murphey, T., Mukadam, M.: Revitalizing optimization for 3d human pose and shape estimation: A sparse constrained formulation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11457–11466 (2021)
10. Fang, Q., Chen, K., Fan, Y., Shuai, Q., Li, J., Zhang, W.: Learning analytical posterior probability for human mesh recovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8781–8791 (2023)
11. Goel, S., Pavlakos, G., Rajasegaran, J., Kanazawa, A., Malik, J.: Humans in 4d: Reconstructing and tracking humans with transformers. In: Proceedings of the IEEE international conference on computer vision (2023)
12. Guan, P., Weiss, A., Balan, A.O., Black, M.J.: Estimating human shape and pose from a single image. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 1381–1388. IEEE (2009)

13. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
14. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. pp. 630–645. Springer (2016)
15. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE transactions on pattern analysis and machine intelligence **36**(7), 1325–1339 (2013)
16. Iqbal, U., Xie, K., Guo, Y., Kautz, J., Molchanov, P.: Kama: 3d keypoint aware body mesh articulation. In: 2021 International Conference on 3D Vision (3DV). pp. 689–699. IEEE (2021)
17. Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Michael, K., Fang, J., Yifu, Z., Wong, C., Montes, D., et al.: ultralytics/yolov5: v7. 0-yolov5 sota realtime instance segmentation. Zenodo (2022)
18. Joo, H., Neverova, N., Vedaldi, A.: Exemplar fine-tuning for 3d human model fitting towards in-the-wild 3d human pose estimation. In: 2021 International Conference on 3D Vision (3DV). pp. 42–52. IEEE (2021)
19. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7122–7131 (2018)
20. Khirodkar, R., Tripathi, S., Kitani, K.: Occluded human mesh recovery. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1715–1725 (2022)
21. Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. Advances in neural information processing systems **33**, 18661–18673 (2020)
22. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
23. Kissos, I., Fritz, L., Goldman, M., Meir, O., Oks, E., Kliger, M.: Beyond weak perspective for monocular 3d human pose estimation. In: Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16. pp. 541–554. Springer (2020)
24. Kocabas, M., Huang, C.H.P., Hilliges, O., Black, M.J.: Pare: Part attention regressor for 3d human body estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11127–11137 (2021)
25. Kocabas, M., Huang, C.H.P., Tesch, J., Müller, L., Hilliges, O., Black, M.J.: Spec: Seeing people in the wild with an estimated camera. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11035–11045 (2021)
26. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2252–2261 (2019)
27. Li, J., Bian, S., Liu, Q., Tang, J., Wang, F., Lu, C.: Niki: Neural inverse kinematics with invertible neural networks for 3d human pose and shape estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12933–12942 (2023)
28. Li, J., Xu, C., Chen, Z., Bian, S., Yang, L., Lu, C.: Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In:

Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3383–3393 (2021)

29. Li, Z., Liu, J., Zhang, Z., Xu, S., Yan, Y.: Cliff: Carrying location information in full frames into human pose and shape estimation. In: European Conference on Computer Vision. pp. 590–606. Springer (2022)
30. Li, Z., Oskarsson, M., Heyden, A.: 3d human pose and shape estimation through collaborative learning and multi-view model-fitting. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 1888–1897 (2021)
31. Li, Z., Xu, B., Huang, H., Lu, C., Guo, Y.: Deep two-stream video inference for human body pose and shape estimation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 430–439 (2022)
32. Lin, K., Lin, C.C., Liang, L., Liu, Z., Wang, L.: Mpt: Mesh pre-training with transformers for human pose and mesh reconstruction. arXiv preprint arXiv:2211.13357 (2022)
33. Lin, K., Wang, L., Liu, Z.: Mesh graphormer. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12939–12948 (2021)
34. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
35. Liu, Y., Yang, J., Gu, X., Guo, Y., Yang, G.Z.: Egohmr: Egocentric human mesh recovery via hierarchical latent diffusion model. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 9807–9813. IEEE (2023)
36. Loper, M., Mahmood, N., Black, M.J.: Mosh: motion and shape capture from sparse markers. ACM Trans. Graph. **33**(6), 220–1 (2014)
37. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. ACM Transactions on Graphics **34**(6) (2015)
38. Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: Amass: Archive of motion capture as surface shapes. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2019), `https://amass.is.tue.mpg.de`
39. Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., Theobalt, C.: Monocular 3d human pose estimation in the wild using improved cnn supervision. In: 2017 international conference on 3D vision (3DV). pp. 506–516. IEEE (2017)
40. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
41. Moon, G., Lee, K.M.: I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16. pp. 752–768. Springer (2020)
42. Osman, A.A., Bolkart, T., Black, M.J.: Star: Sparse trained articulated human body regressor. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16. pp. 598–613. Springer (2020)
43. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10975–10985 (2019)

44. Pavlakos, G., Malik, J., Kanazawa, A.: Human mesh recovery from multiple shots. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1485–1495 (2022)
45. Pavlakos, G., Zhu, L., Zhou, X., Daniilidis, K.: 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 (2018)
46. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. arXiv preprint arXiv:2201.02610 (2022)
47. Sengupta, A., Budvytis, I., Cipolla, R.: Probabilistic 3d human shape and pose estimation from multiple unconstrained images in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16094–16104 (2021)
48. Shetty, K., Birkhold, A., Jaganathan, S., Strobel, N., Kowarschik, M., Maier, A., Egger, B.: Pliks: A pseudo-linear inverse kinematic solver for 3d human body estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 574–584 (2023)
49. Shin, S., Halilaj, E.: Multi-view human pose and shape estimation using learnable volumetric aggregation. arxiv. org. arXiv preprint arXiv:2011.13427 (2020)
50. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5693–5703 (2019)
51. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
52. Von Marcard, T., Henschel, R., Black, M.J., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3d human pose in the wild using imus and a moving camera. In: Proceedings of the European conference on computer vision (ECCV). pp. 601–617 (2018)
53. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2mesh: Generating 3d mesh models from single rgb images. In: Proceedings of the European conference on computer vision (ECCV). pp. 52–67 (2018)
54. Wang, W., Ge, Y., Mei, H., Cai, Z., Sun, Q., Wang, Y., Shen, C., Yang, L., Komura, T.: Zolly: Zoom focal length correctly for perspective-distorted human mesh reconstruction. arXiv preprint arXiv:2303.13796 (2023)
55. Wang, Y., Daniilidis, K.: Refit: Recurrent fitting network for 3d human recovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14644–14654 (2023)
56. Xue, Y., Chen, J., Zhang, Y., Yu, C., Ma, H., Ma, H.: 3d human mesh reconstruction by learning to sample joint adaptive tokens for transformers. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 6765–6773 (2022)
57. Yao, P., Fang, Z., Wu, F., Feng, Y., Li, J.: Densebody: Directly regressing dense 3d human pose and shape from a single color image. arXiv preprint arXiv:1903.10153 (2019)
58. Yoshiyasu, Y.: Deformable mesh transformer for 3d human mesh recovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17006–17015 (2023)
59. Yu, Z., Wang, J., Xu, J., Ni, B., Zhao, C., Wang, M., Zhang, W.: Skeleton2mesh: Kinematics prior injected unsupervised human mesh recovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8619–8629 (2021)

60. Yuan, Y., Iqbal, U., Molchanov, P., Kitani, K., Kautz, J.: Glamr: Global occlusion-aware human mesh recovery with dynamic cameras. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11038–11049 (2022)
61. Zanfir, M., Zanfir, A., Bazavan, E.G., Freeman, W.T., Sukthankar, R., Sminchisescu, C.: Thundr: Transformer-based 3d human reconstruction with markers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12971–12980 (2021)
62. Zhang, H., Cao, J., Lu, G., Ouyang, W., Sun, Z.: Learning 3d human shape and pose from dense body parts. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(5), 2610–2627 (2020)
63. Zhang, H., Tian, Y., Zhang, Y., Li, M., An, L., Sun, Z., Liu, Y.: Pymaf-x: Towards well-aligned full-body model regression from monocular images. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
64. Zhang, H., Tian, Y., Zhou, X., Ouyang, W., Liu, Y., Wang, L., Sun, Z.: Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11446–11456 (2021)
65. Zhang, J., Yu, D., Liew, J.H., Nie, X., Feng, J.: Body meshes as points. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 546–556 (2021)
66. Zhang, S., Ma, Q., Zhang, Y., Aliakbarian, S., Cosker, D., Tang, S.: Probabilistic human mesh recovery in 3d scenes from egocentric views. arXiv preprint arXiv:2304.06024 (2023)
67. Zhang, T., Huang, B., Wang, Y.: Object-occluded human shape and pose estimation from a single color image. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7376–7385 (2020)
68. Zhou, X., Zhu, M., Pavlakos, G., Leonardos, S., Derpanis, K.G., Daniilidis, K.: Monocap: Monocular human motion capture using a cnn coupled with a geometric prior. IEEE transactions on pattern analysis and machine intelligence **41**(4), 901–914 (2018)