KITRO: Refining Human Mesh by 2D Clues and Kinematic-tree Rotation

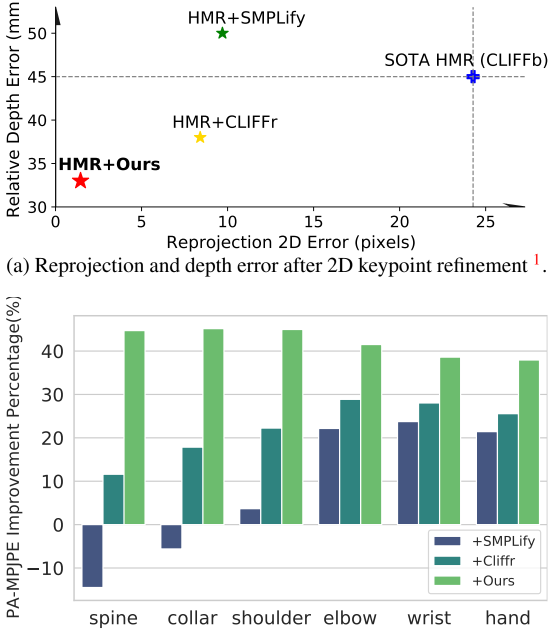
Fengyuan Yang   Kerui Gu   Angela Yao
National University of Singapore
{fyang, keruigu, ayao}@comp.nus.edu.sg

Abstract

2D keypoints are commonly used as an additional cue to refine estimated 3D human meshes. Current methods optimize the pose and shape parameters with a reprojection loss on the provided 2D keypoints. Such an approach, while simple and intuitive, has limited effectiveness because the optimal solution is hard to find in ambiguous parameter space and may sacrifice depth. Additionally, divergent gradients from distal joints complicate and deviate the refinement of proximal joints in the kinematic chain. To address these, we introduce ***Ki****nematic-****T****ree* ***Ro****tation (****KITRO****), a novel mesh refinement strategy that explicitly models depth and human kinematic-tree structure. KITRO treats refinement from a bone-wise perspective. Unlike previous methods which perform gradient-based optimizations, our method calculates bone directions in closed form. By accounting for the 2D pose, bone length, and parent joint's depth, the calculation results in two possible directions for each child joint. We then use a decision tree to trace binary choices for all bones along the human skeleton's kinematic-tree to select the most probable hypothesis. Our experiments across various datasets and baseline models demonstrate that KITRO significantly improves 3D joint estimation accuracy and achieves an ideal 2D fit simultaneously. Our code available at:* https://github.com/MartaYang/KITRO.

1. Introduction

3D human pose and shape estimation, also called Human Mesh Recovery (HMR), is relevant for augmented and virtual reality applications. Statistical human shape models like SMPL [21] have greatly simplified the HMR task. However, state-of-the-art methods [9, 13, 18] still suffer from the "misalignment problem" [15] where the predicted 3D mesh does not align well with the 2D image evidence. The most advanced HMR model [18] still has a 24 pixels reprojection error on the 3DPW dataset [32], as shown in Fig. 1a. On the other hand, estimating 2D human pose is more advanced [2, 33, 36] and yields robust 2D keypoints, even under challenging scenarios such as occlusion, varying lighting, and extreme poses. As such, a standard strategy for improving HMR is to leverage 2D keypoints as a cue to refine the estimated 3D meshes. Prior methods [1, 9, 18] optimize the pose and shape parameters with a reprojection loss on the 2D keypoints with a standard gradient descent. Such an approach, while intuitive and straightforward, is not always effective.



(a) Reprojection and depth error after 2D keypoint refinement [1].

(b) Improvement of joints along the kinematic chain of the arm.

Figure 1. (a) Our method has the lowest reprojection and depth errors. (b) Competing methods exhibit a decrease in improvement and even deterioration progressing along the kinematic chain from the hand to the spine; our approach exhibits the opposite but maintains a significant and competitive improvement for all joints.

A primary reason is the inherent depth ambiguity in 2D projection, as multiple poses and shapes can fit the same 2D evidence. As such, optimizing solely on 2D reprojection does not account for depth ambiguity so is unlikely to find the optimal solution and may even increase the depth error (see HMR+SMPLify [1] in Fig. 1a). Secondly, existing

[1] CLIFF [18] proposes both a base model and a 2D keypoint refinement; we distinguish the two as 'CLIFFb' and 'CLIFFr'.

methods optimize all the body joints collectively through gradient descent. Yet the gradient updates at different joints may be incongruent. Updates at the distal joints far down the kinematic chain, such as the wrist or hands, are backpropagated to the proximal joints closer to the root, like the shoulder or collar. Divergent gradients can complicate the update of proximal joints, limiting the refinement improvements, to the extent that it may even harm the overall accuracy (see Fig. 1b).

In this work, we offer the key insight that joint depth can be solved explicitly in closed-form. Provided with 2D keypoints, the depth of the parent joint, and the 3D length of the bone connecting the two joints, the problem can be narrowed down into two solutions. Of these two solutions, one corresponds to the bone pointing toward the camera, and the other corresponds to the bone pointing away from the camera as shown in Fig. 3. In this way, the depth ambiguity in the solution space can be largely reduced.

Having two possible depths for every joint naturally forms a binary tree progressing along kinematic chains in the human body (see Fig. 4). Any path traversing the tree is a hypothesis and we can formulate overall pose refinement as a selection problem of finding an optimal path. Such a strategy has a distinct advantage in that it can equally improve proximal and distal joints.

Based on these insights, we propose a novel plug-and-play human mesh refinement strategy which we call Kinematic-tree Rotation (KITRO). KITRO explicitly models joint depth and solves for bone direction as a swing rotation in closed form, ensuring an excellent fit to 2D keypoints with lower depth errors. KITRO employs a decision tree to handle divergences among joints in the kinematic-tree, effectively tracing and selecting the most probable hypotheses for stable and consistent improvements across the kinematic chain. Our experimental results, on various datasets and base models, demonstrate significantly higher accuracy compared to existing methods.

We highlight our key contributions as follows:

- KITRO's explicit depth modeling and closed-form calculation diminish ambiguities in solution space, enhancing depth accuracy and obtaining ideal 2D fit simultaneously.
- KITRO's decision-tree-based hypotheses tracing and selection for joint rotations encompasses the entire kinematic-tree, benefiting both proximal and distal joints.
- Our method's effectiveness, especially large gains in pose accuracy, is validated across datasets and base models.

2. Related Works

Human Mesh Recovery (HMR). HMR methods are either non-parametric, in that they directly estimate the 3D mesh vertices [19, 20, 22, 27, 30, 37] or parametric, using statistical models [3, 10, 15, 18, 26, 28]. In using a statistical model like SMPL [21], the HMR task simplifies to an estimation on the model parameters. However, estimating the parameters can be highly challenging under monocular settings despite the advancements in network architecture [29, 31, 35] and learning paradigms [17, 23, 24]. More recently, there are also hybrid approaches combining parametric and non-parametric [16].

Human Mesh Refinement with 2D Keypoints. One curious observation is that estimated meshes are often poorly aligned to the 2D image evidence. As such, optimization methods have been proposed to leverage 2D keypoints as an additional cue to refine estimated 3D human meshes. A simple and intuitive way for refinement is simply to update the mesh model parameters, *i.e.* the SMPL parameters with an additional 2D reprojection loss [1, 6, 15]. However, as 2D image evidence is an ambiguous cue for 3D estimates, directly update the SMPL parameters tends to result in unnatural poses, even if they are better aligned to the provided 2D keypoints. As such, methods like EFT [9] and CLIFF [18] use the 2D reprojection loss to update the HMR estimation network weights instead. Such a refinement, however, is dataset-specific and thus loses generalization. Additionally, finding ways to avoid overfitting or underfitting remains a challenging limitation in these fine-tuning methods. Our work also focuses on mesh refinement; however, instead of optimization with a 2D reprojection loss, we opt to solve for the refined solution in closed form.

3. Preliminaries

SMPL. The SMPL model [21] is a commonly used 3D statistical shape model of the human body. It maps pose parameters $\boldsymbol{\theta} \in \mathbb{R}^{P\times 3}$ and shape parameters $\boldsymbol{\beta} \in \mathbb{R}^{10}$ to a 3D mesh of the body $\mathbf{V} \in \mathbb{R}^{N\times 3}$, where $N=6890$ and $P=24$ are the number of vertices and body joints respectively. We denote the $\mathbf{J}(\boldsymbol{\beta}, \boldsymbol{\theta}) \in \mathbb{R}^{P\times 3}$ as the function that maps pose parameters $\boldsymbol{\theta}$ and shape parameters $\boldsymbol{\beta}$ to joint locations. According to human kinematic-tree shown in Fig. 4, we define the Euclidean distance between child joint $\mathbf{J}^c(\boldsymbol{\beta}, \boldsymbol{\theta})$ and its parent joint $\mathbf{J}^p(\boldsymbol{\beta}, \boldsymbol{\theta})$ as the bone length:

$$bl_{3D}^{(p,c)} = \|\mathbf{J}^p(\boldsymbol{\beta}, \boldsymbol{\theta}) - \mathbf{J}^c(\boldsymbol{\beta}, \boldsymbol{\theta})\|_2 . \quad (1)$$

Swing-Twist Decomposition. The pose parameters $\boldsymbol{\theta}$ are the relative rotation of each joint, where the rotation of the pelvis [2] joint $\boldsymbol{\theta}^0$ serves as the global rotation of the human body. The standard SMPL model uses the axis-angle $\{\boldsymbol{\theta}^0, \cdots, \boldsymbol{\theta}^{23}\}$ to represent the rotation of each joint and rotates the joints along the kinematic chain from the pelvis to the end joints. An alternative [16] proposes to represent each joint rotation in rotation matrix form $\boldsymbol{\theta}_R^i \in \mathbb{SO}(3)$ with the swing-twist decomposition: $\boldsymbol{\theta}_R^i = R_{sw}R_{tw}$, where $\boldsymbol{\theta}_R^i$ is equivalent rotation matrix form of $\boldsymbol{\theta}^i$. Given the bone

[2] Also known as root joint, we use them interchangeably in this paper.

direction, the swing rotation which is 2 degrees of freedom can be derived in closed form from Rodrigues' formula (full details in Sec. A of the Supplementary).

Human Mesh Recovery and Refinement. Monocular HMR models take an image $\mathbf{X} \in \mathbb{R}^{H\times W}$ as input and predicts SMPL parameters $(\boldsymbol{\theta}, \boldsymbol{\beta})$ and the camera translation $\mathbf{t} \in \mathbb{R}^3$ as part of the camera extrinsics to project the 3D mesh onto the image plane. Mesh refinement methods [1, 9, 18] uses additional 2D keypoints $\mathbf{j} \in \mathbb{R}^{P\times 2}$ to refine some estimated SMPL and camera translation parameters $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\beta}}, \hat{\mathbf{t}})$. The refinement process can be formalized as:

$$\left(\hat{\boldsymbol{\theta}}', \hat{\boldsymbol{\beta}}', \hat{\mathbf{t}}'\right) = \text{Refine}(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\beta}}, \hat{\mathbf{t}}, \mathbf{j}). \tag{2}$$

The refinement as given in Eq. 2 can also be applied iteratively such as SMPLify [1], by using the refined outputs $(\hat{\boldsymbol{\theta}}', \hat{\boldsymbol{\beta}}', \hat{\mathbf{t}}')$ as the input estimates to continue the refinement process. A commonly used approach for refinement is to minimize the 2D reprojection loss of the estimated 3D joints $\mathbf{J}(\boldsymbol{\beta}, \boldsymbol{\theta})$ with respect the provided 2D joints $\mathbf{j}$:

$$\mathcal{L}_{j2D} = \min_{\boldsymbol{\theta},\boldsymbol{\beta},\mathbf{t}} \|\pi\left(\mathbf{J}(\boldsymbol{\beta}, \boldsymbol{\theta}) + \mathbf{t}\right) - \mathbf{j}\|_2, \tag{3}$$

where π indicates the camera projection function. These parameters can be optimized jointly [9, 18] or fix some parameters and optimize others [1]. It is worth mentioning that when fixing $\boldsymbol{\theta}$ and β, the optimal $\mathbf{t}$ can be solved in closed-form because it can be formulated as a least squares problem.

Camera Intrinsics and Extrinsics. In our work, we relax the weakly-perspective assumption and adopt a full-perspective camera model like prior studies [11, 14, 34]. Consistent with these works [14, 18], we estimate the camera focal length as $f = \sqrt{H^2 + W^2}$. Therefore, the camera intrinsic can be formulated as:

$$K = \begin{bmatrix} f & 0 & W/2 \\ 0 & f & H/2 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

Following the convention in HMR [5, 10, 12, 13, 15], the camera rotation is set as I and absorbed into the global orientation of the human (*i.e.*, $\boldsymbol{\theta}^0$) predicted by the HMR model. Thus, the translation $\mathbf{t}$ is the only camera extrinsic that needs to be estimated.

4. Method

As illustrated in Fig. 2, our refinement framework iteratively updates the estimates for the camera and shape (Sec. 4.1) as well as the pose (Sec. 4.2, Sec. 4.3). Unlike previous works [1, 9, 18], which update pose and shape parameters jointly, we perform an individual update while keeping the others fixed based on the previously refined value. This allows us to focus on the pose refinement where we explicitly model the depth in a closed-form manner and choose the most likely hypothesis by using a decision tree along the kinematic-tree.

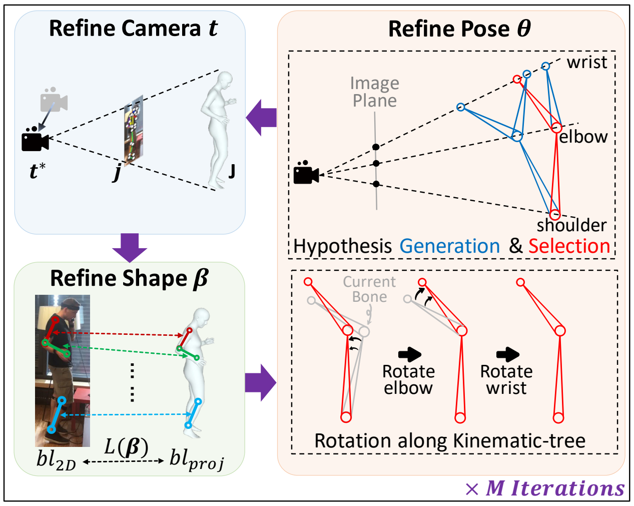


Figure 2. Our framework overview. Starting with an initial human mesh and 2D keypoints, our iterative refinement operates over camera, shape, and pose. The process involves localizing camera translation for 2D-3D joint alignment, optimizing shape for more aligned bone length, and generating and selecting bone direction hypotheses for rotation refinement along the kinematic-tree.

4.1. Camera and Shape Refinement

As can be seen in Fig. 2, the given 2D keypoints together with the current human mesh can definitely provide some information and constraints for the camera translation and human shape. These constraints can help us to refine these two factors better.

Camera Translation Adjustment. We estimate the camera translation based on the 2D reprojection loss given by Eq. 3. Minimizing the loss is equivalent to solving a least-squares optimization for the camera translation [15]:

$$\mathbf{t}^* = \arg\min_{\mathbf{t}} \|\pi(\mathbf{J}(\boldsymbol{\beta}, \boldsymbol{\theta}) + \mathbf{t}) - \mathbf{j}\|_2, \tag{5}$$

where $\mathbf{t}^*$ can be found with SVD. Note that $\mathbf{t}^*$ is the optimal solution for a given $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$.

From an iterative update perspective, we find it more effective to update $\mathbf{t}$ with a moving average. For current camera translation $\mathbf{t}$, the updated $\mathbf{t}'$ is given as:

$$\mathbf{t}' = (\mathbf{t}^* + \mathbf{t})/2. \tag{6}$$

The reason for Eq. 6 is that $\mathbf{t}^*$ can be affected by the noise of $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$. In that sense, the moving average keeps the historical information from the original HMR prediction, acting like a good regularizer.

Shape Optimization. Previous works update $\boldsymbol{\beta}$ based on the 2D reprojection of the individual joints $\mathbf{J}(\boldsymbol{\beta}, \boldsymbol{\theta})$. We

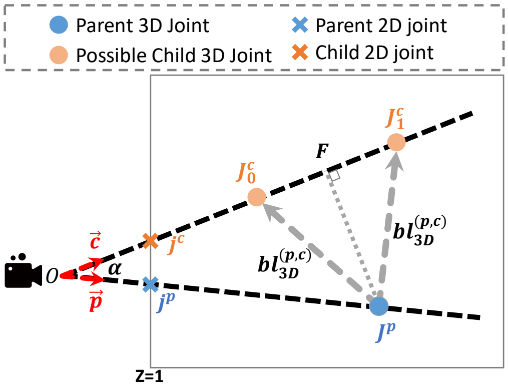


Figure 3. Calculation of two bone direction solutions of the (p, c) joint pair, based on 2D keypoints, bone length, and the depth of the parent joint. One points towards the camera, and the other away.

consider the refinement of $\boldsymbol{\beta}$ from a *bone length* perspective. Consider a parent joint indexed by p, with estimated 3D joint $\mathbf{J}^p(\boldsymbol{\beta},\boldsymbol{\theta})$ and provided 2D joint $\mathbf{j}^p$; similarly, for a child joint indexed by c, consider $\mathbf{J}^c(\boldsymbol{\beta},\boldsymbol{\theta})$. The projected 2D bone length between the (p,c) joint pair is defined as the Euclidean distance between the two joints:

$$bl^{(p,c)}_{proj}(\boldsymbol{\beta}) = \|\pi(\mathbf{J}^p(\boldsymbol{\beta},\boldsymbol{\theta})+\mathbf{t}') - \pi(\mathbf{J}^c(\boldsymbol{\beta},\boldsymbol{\theta})+\mathbf{t}')\|_2 \,. \tag{7}$$

For clarity, we specify that the projected bone length $bl^{(p,c)}_{proj}(\boldsymbol{\beta})$ depends only on $\boldsymbol{\beta}$, as we treat $\boldsymbol{\theta}$ and $\mathbf{t}'$ as fixed constants while we update the shape parameter $\boldsymbol{\beta}$. The bone length of the provided 2D joints can be defined similarly as

$$bl^{(p,c)}_{2D} = \|\mathbf{j}^p - \mathbf{j}^c\|_2 \,. \tag{8}$$

To estimate the shape loss, we consider an L1-norm between the projected bone length in Eq. 7 and the given bone lenght of Eq. 8 over all the bones or p, c combinations in the human body:

$$L(\boldsymbol{\beta}) = \sum_{p,c} \left| bl^{(p,c)}_{proj}(\boldsymbol{\beta}) - bl^{(p,c)}_{2D} \right| . \tag{9}$$

Unlike the camera parameter $\mathbf{t}$, the optimal shape parameter $\boldsymbol{\beta}$ cannot be solved for in closed form. As such, we estimate the updated shape parameter $\boldsymbol{\beta}'$ with gradient-descent based optimization of the loss in Eq. 9:

$$\boldsymbol{\beta}' = \boldsymbol{\beta} - \eta \nabla_{\boldsymbol{\beta}} L(\boldsymbol{\beta}) \quad \text{for } T \text{ steps}, \tag{10}$$

where η is the learning rate and optimization takes T steps.

4.2. Pose Hypothesis Generation

3D Bone Direction Calculation. To refine the pose parameters $\boldsymbol{\theta}$, we opt to work with the swing-twist decomposed form of the joint rotation, rather than in the standard axis-angle representations of SMPL. This is key in our refinement approach because it allows us to estimate the swing angle in closed form directly from the provided 2D keypoints. Consider the 2D keypoints $\mathbf{j}^p$ and $\mathbf{j}^c$ of a bone (p,c) as defined in the SMPL kinematic tree. Their corresponding 3D joints $\mathbf{J}^p(\boldsymbol{\beta}',\boldsymbol{\theta})$ and $\mathbf{J}^c(\boldsymbol{\beta}',\boldsymbol{\theta})$ are located along the rays $\vec{p}$ and $\vec{c}$ from the camera to the 2D keypoints on the image plane. And O denotes the camera location which is the negative of camera translation $O = -\mathbf{t}'$ since camera rotation is identity. As shown in Fig. 3, directions of these two rays are:

$$\vec{p} = K^{-1} \times \mathbf{j}^p_h, \quad \vec{c} = K^{-1} \times \mathbf{j}^c_h, \tag{11}$$

where subscript h denotes homogenized coordinates and K is the camera intrinsic matrix defined in Sec. 3. Based on the direction of the rays $\vec{p}$ and $\vec{c}$, their intersection angle α satisfies the following:

$$\cos\alpha = \frac{\vec{p}\cdot\vec{c}}{\|\vec{p}\|\|\vec{c}\|}, \quad \sin\alpha = \frac{\|\vec{p}\times\vec{c}\|}{\|\vec{p}\|\|\vec{c}\|}. \tag{12}$$

Now, as we know the 3D bone length $bl^{(p,c)}_{3D}$ defined in Eq. 1 and the depth of the parent joint from the camera $|OJ^p| = \left\|\mathbf{J}^p(\boldsymbol{\beta}',\boldsymbol{\theta})+\mathbf{t}'\right\|_2$, we can directly solve for the child joint in closed form. Specifically, $|FJ^c_0| = |FJ^c_1| = \sqrt{\left(bl^{(p,c)}_{3D}\right)^2 - (|OJ^p|\sin\alpha)^2}$ based on Fig. 3. Therefore, we can calculate the child joint J^c specified by the vector $\overrightarrow{J^pJ^c}$ from the parent joint J^p, and the depth $|OJ^p|$ from the camera O:

$$\begin{aligned} \overrightarrow{J^pJ^c} &= \underbrace{|OJ^p|\cdot(\cos\alpha\cdot\vec{c}-\vec{p})}_{\overrightarrow{J^pF}} \pm \underbrace{|FJ^c_0|\cdot\vec{c}}_{\overrightarrow{FJ^c_0}}, \\ |OJ^c| &= |OJ^p|\cdot\cos\alpha \pm |FJ^c_0|. \end{aligned} \tag{13}$$

There are two possibilities for the child joint, as indicated by the ± sign in the solutions in Eq. 13. One solution points towards the camera ($\overrightarrow{J^pJ^c_0}$ in Fig. 3) while the other points away ($\overrightarrow{J^pJ^c_1}$). These two solutions directly illustrate the inherent depth ambiguity. Depending on the accuracy of the estimated terms, the square root term may become negative (might arise during decision tree calculations in the next step); for numerical stability, we rectify it to 0.

From Bone Direction to Full Body Pose Hypothesis. Note that the solution for the bone direction in Eq. 13 depends on the depth of the parent joint $|OJ^p|$. As the parent joint's depth has two possible solutions as well, and the depth dependency propagates through the kinematic tree, the hypotheses for possible poses naturally form binary trees. Fig. 4 shows an example of the left leg, beginning at the pelvis root node, with child nodes for the hip, knee, ankle, and toe. For the entire body pose, there are 5 trees, representing the arms, legs, and torso. A full-body pose hypothesis is then represented by a path through each of the

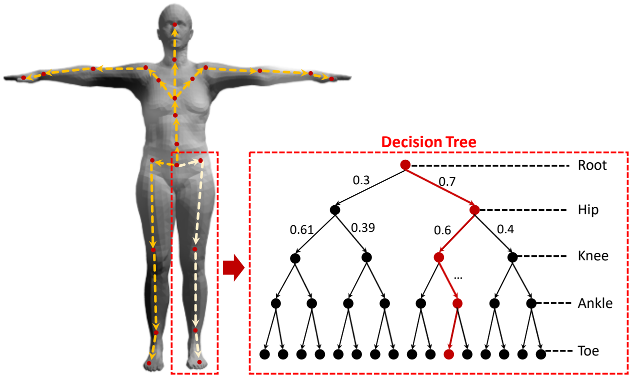


Figure 4. Human kinematic-tree in SMPL [21] and ours decision tree example tracing all hypotheses for leg joints. In our method, local solution certainties are used as edge weights, with the largest global path product representing the most probable pose.

trees. It's worth mentioning that while we could select solutions for each joint greedily, our approach leverages a decision tree naturally formed along the kinematic chain since it enables more accurate selections by considering all potential outcomes.

4.3. Pose Hypothesis Selection

Decision Tree Formulation. From the binary trees, we wish to select an optimal path. Yet without additional information or cues, it is challenging to know how optimality should be measured. As such, we rely only on the initial HMR estimate, and compute the cosine similarity between the relative bone rotations for the two closed-form solutions $R_{rel}(\overrightarrow{J^p J^c_k}|\phi(p))$ for $k = \{0, 1\}$ and the relative bone rotation predicted by the original HMR model $\boldsymbol{\theta}^p_{\text{HMR}}$:

$$\text{cos\_sim}(\overrightarrow{J^p J^c_k}|\phi(p)) = \frac{cos\left\langle R_{rel}(\overrightarrow{J^p J^c_k}|\phi(p)), \boldsymbol{\theta}^p_{\text{HMR}}\right\rangle + 1}{2} \tag{14}$$

where $\phi(p) = (\phi_0, \cdots, \phi_i, \cdots, \phi_p)$ represents the path from the root to the current parent joint p. Each $\phi_i \in \{0, 1\}$ denotes the two possible solutions for each bone with $\mathbf{J}_i$ as the child joint along the chain, except ϕ_0 since the root joint is fixed after camera refinement for each iteration. We directly apply the Softmax of these two cosine similarities as the weights for edges in the decision tree as shown in Fig. 4:

$$w(e^{(p,c)}_k|\phi(p)) = \frac{\exp(\text{cos\_sim}(\overrightarrow{J^p J^c_k}|\phi(p)))}{\sum_{k'=0}^{1} \exp(\text{cos\_sim}(\overrightarrow{J^p J^c_{k'}}|\phi(p)))}. \tag{15}$$

Intuitively, Eq. 15 measures the relative consistency between calculated and original prediction. It relies on the assumption that the original HMR prediction, although not precise at exact bone direction, is sufficiently accurate at indicating whether the bone points towards or away from the camera. We verify this assumption empirically in Sec. B of the Supplementary.

The computational complexity for calculating the solutions in the binary trees depends only on the depth, as nodes within any given depth of the trees can be computed in parallel. By constructing the binary trees and estimating the weights, we obtain a global view on the pose feasibility. We choose as the optimal path the one with the highest node products. The final pose is defined by the optimal selection path:

$$\phi^* = \arg\max_{\phi} \prod_{(p,c)} w(e^{(p,c)}_{\phi_c}|\phi(p)), \tag{16}$$

where $\phi = \{\phi_0, \cdots, \phi_{23}\}$ is a pose hypothesis for all 23 bones of a human body.

Pose Parameter Update. We perform an update on current $\boldsymbol{\theta}$ according to the selection chain ϕ^* in Eq. 16. The update is soft, weighted by the edge weight $w(e^{(p,c)}_{\phi^*_c}|\phi^*(p))$ calculated in Eq. 15 which acts like a certainty term for choosing ϕ^*_c instead of the other (*i.e.*, $1 - \phi^*_c$):

$$\lambda^{(p,c)} = w(e^{(p,c)}_{\phi^*_c}|\phi^*(p)). \tag{17}$$

The final bone direction $\vec{\mathbf{b}}^{(p,c)}_{new}$ is updated as the weighted sum of the selected solution $\overrightarrow{J^p J^c_{\phi^*_c}}$ and current bone direction $\vec{\mathbf{b}}^{(p,c)} = \mathbf{J}^c(\boldsymbol{\beta}', \boldsymbol{\theta}) - \mathbf{J}^p(\boldsymbol{\beta}', \boldsymbol{\theta})$:

$$\vec{\mathbf{b}}^{(p,c)}_{new} = \lambda^{(p,c)} \cdot \overrightarrow{J^p J^c_{\phi^*_c}} + (1 - \lambda^{(p,c)}) \cdot \vec{\mathbf{b}}^{(p,c)}. \tag{18}$$

Finally, the pose parameter $\boldsymbol{\theta}^p$ of the parent joint p can be updated based on the swing rotation $R^{(p,c)}_{sw}$ which rotates $\vec{\mathbf{b}}^{(p,c)}$ to $\vec{\mathbf{b}}^{(p,c)}_{new}$ simply by the Rodrigues' rotation formula. In the special case of the root joint and the third spine joint (*i.e.*, 'Spine3'), where there are three children denoted as c_0, c_1, c_2, we compute a rotation matrix $R^{(p,c)}_{sw}$ that optimally rotates the vectors $\{\vec{\mathbf{b}}^{(p,c_0)}, \vec{\mathbf{b}}^{(p,c_1)}, \vec{\mathbf{b}}^{(p,c_2)}\}$ to best align with $\{\vec{\mathbf{b}}^{(p,c_0)}_{new}, \vec{\mathbf{b}}^{(p,c_1)}_{new}, \vec{\mathbf{b}}^{(p,c_2)}_{new}\}$ by SVD [16]. Finally, by applying above rotation matrix $R^{(p,c)}_{sw}$ to update parent joint p's rotation $\boldsymbol{\theta}^p_R$, we obtain the refined pose parameter:

$$\boldsymbol{\theta}^{p\,\prime}_R = (\prod_{i \in KC(\tilde{p})} \boldsymbol{\theta}^i_R)^T \cdot R^{(p,c)}_{sw} \cdot \prod_{i \in KC(p)} \boldsymbol{\theta}^i_R, \tag{19}$$

where $\tilde{p}$ is the parent joint of p, and $KC(p)$ is the kinematic chain from root to joint p. Proof of correctness is given in Sec. C of the Supplementary. Updating all joints by Eq. 19 along kinematic-tree gets the refined $\boldsymbol{\theta}'$.

The updates in Eq. 6, Eq. 10, and Eq. 19 specify one refinement iteration for $\mathbf{t}$, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. We can continue to iterate by using the refined parameters as initial estimates for a total of M iterations.

4.4. Implementation Details

Here we elaborate on more implementation details. For the shape refinement, we employ Adam optimizer to optimize

Algorithm 1 Ours Human Mesh Refinement

```
Require: Initial pose $\boldsymbol{\theta}_0$, shape $\boldsymbol{\beta}_0$, camera translation $\mathbf{t}_0$,
    2D keypoints $\mathbf{j}$, iterations $M$, kinematic-tree $KT$.
Ensure: Refined $\boldsymbol{\theta}_M$, $\boldsymbol{\beta}_M$, $\mathbf{t}_M$.
for $m = 0 \rightarrow M - 1$ do
    $\mathbf{t}^* \leftarrow$ Best $\mathbf{t}$ aligned $\mathbf{j}$ and $\mathbf{J}(\boldsymbol{\beta}_m, \boldsymbol{\theta}_m)$    ▷ Eq. 5
    $\mathbf{t}_{m+1} \leftarrow (\mathbf{t}^* + \mathbf{t}_m)/2$    ▷ Eq. 6
    $\boldsymbol{\beta}_{m+1} \leftarrow$ Adam optimize $\boldsymbol{\beta}_m$ by $L(\boldsymbol{\beta})$    ▷ Eq. 9
    $DecisionTree \leftarrow$ Binary solutions along $KT$
    $\phi^* \leftarrow$ Optimal path in $DecisionTree$    ▷ Eq. 16
    $\boldsymbol{\theta}_{m+1} \leftarrow$ Update bone rotation base on $\phi^*$    ▷ Eq. 19
end for
```

Eq. 9 for $T = 10$ steps with a learning rate $\eta = 0.1$ in each iteration. And the whole iteration number is $M = 10$. Ours overall pseudo-code is shown in Algorithm 1.

5. Experiments

5.1. Dataset and Metrics

3DPW dataset serves as a rigorous outdoor benchmark tailored for 3D pose and shape estimation. We follow [8, 9, 18] and use the ground truth 2D keypoint annotations from 3DPW as refinement inputs. **Human3.6M** is an indoor 3D human mesh dataset. Consistent with preceding works [4, 10, 15], we use subjects S9 and S11 for testing. Similarly, we use the ground truth 2D keypoints for refinement inputs.

We use three standard 3D pose and shape metrics: (1) **MPJPE** measures the average Euclidean distance between the ground truth and the predicted joint positions, only considering alignment at the pelvis, (2) **PA-MPJPE** which is the MPJPE error after further aligning the predicted pose to the ground truth with Procrustes aligned and (3) **PVE** measuring the average Euclidean distance between the predicted and the ground truth mesh vertices after pelvis alignment.

5.2. Ablation Study

We adopted CLIFFb [18] as the baseline HMR model. The ablation study upon our full model for three refinement factors, camera, shape, and pose, are shown in Tab. 1.

Camera Refinement. The first segment in Tab. 1 shows the impact when we do not refine the camera estimate and fix it to the HMR estimate (first row) and do not use the moving average and directly replace the camera estimate according to the optimal translation of Eq. 5 without Eq. 6 (second row). Performance drop shows the effectiveness of the camera translation refinement and the moving average as a good regularizer. The third row, using a fixed focal length of 5000, shows an obvious performance drop, indicating the limitations of a weak perspective camera model.

Shape Refinement. The second segment in Tab. 1 shows the impact when the shape parameter $\boldsymbol{\beta}$ is not updated (first row in this segment) vs. optimizing $\boldsymbol{\beta}$ according to the standard 2D projection loss of the keypoints (second row) and our proposed loss on the 2D bone length projection (as per Eq.9). This comparison reveals that our loss is more effective than $\mathcal{L}_{j2D}$ in our method. The ablation for refinement step T and learning rate η are given in Sec. D of the Supplementary, they are all not highly sensitive factors.

Table 1. Ablation study for camera, shape, and pose refinement on 3DPW across three segments. The bottom segment presents our full model's results. 'DT' stands for Decision Tree.

Method	PA-MPJPE ↓	MPJPE ↓	PVE ↓
fixed $\mathbf{t}$ as $\mathbf{t}_0$	28.53	46.68	57.76
hard-update $\mathbf{t}$ (Eq. 5)	27.99	50.21	61.11
Large focal length	32.72	52.46	63.61
fixed $\boldsymbol{\beta}$ as $\boldsymbol{\beta}_0$	35.00	69.02	84.25
$\boldsymbol{\beta}$ from $\mathcal{L}_{j2D}$ of Eq. 3	32.60	52.60	64.45
fixed $\boldsymbol{\theta}$ as $\boldsymbol{\theta}_0$	44.57	80.03	95.71
greedy + hard-update	34.32	54.91	67.05
greedy + soft-update	29.24	45.94	56.57
DT + hard-update	32.18	52.15	63.82
KITRO (**ours**)	**27.67**	**43.53**	**53.44**

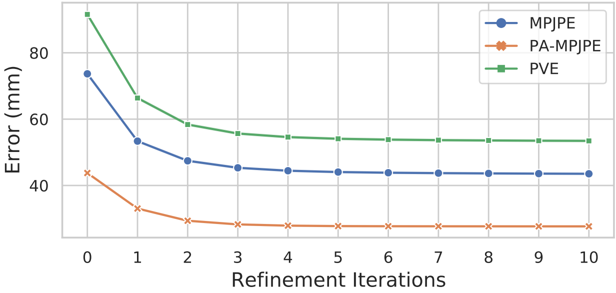


Figure 5. Impact of Refinement Iterations for MPJPE, PA-MPJPE, and MPVPE on 3DPW. Three line plots demonstrate a graceful decrease and quick convergence of error metrics.

Pose Refinement. The third segment in Tab. 1 shows the impact when the pose refinement is removed (first row in this segment) as well as alternative designs on the decision tree (vs greedy selection, second and third row) and the weighted update of Eq. 18 (vs hard-update, second and fourth row). The decreased performance in these designs shows the global perspective offered by our decision tree and the necessity of reweighting for a moderated update.

Parameter Impact. Across the three refined parameters, $\boldsymbol{\theta}$ has the biggest impact; removing it from the refinement leads to a ≈60% increase in these three evaluation errors. In contrast, $\boldsymbol{\beta}$ has a ≈30% increase, and the camera has the least impact of $< 1\%$. This demonstrates the crucial of pose and shape refinement in our method, with the KITRO design for pose refinement notably enhancing performance. Camera refinement, however, exhibits less sensitivity.

Iterative Updates. Fig. 5 plots the three evaluation metrics with respect to the number of refinement iterations. The errors gracefully decrease with increasing interactions, converging to the final result after 4-5 iterations. Fig. 6 shows

Table 2. SOTA comparison on 3DPW and Human3.6M. Baseline HMR model for human mesh initialization in the first segment, refinement methods in the second segment including our results at the bottom. * indicates our reproduced results. † marks a different 3DPW evaluation protocol using extra gender information in data preprocessing. All our reproductions ensure the same protocol and fair comparison.

	3DPW			Human3.6M	
Method	PA-MPJPE ↓	MPJPE ↓	PVE ↓	PA-MPJPE ↓	MPJPE ↓
CLIFFb† [18]	43.0	69.0	81.2	-	-
CLIFFb*	43.76	73.67	91.58	36.16	55.18
DynaBOA [8]	40.4	65.5	82.0	-	-
Pose2Mesh [4]	34.6	65.1	-	35.3	51.1
CLIFFb + CLIFFr† [18]	32.8	52.8	61.5	-	-
CLIFFb + SMPLify*	36.11	66.67	79.91	28.07	45.19
CLIFFb + CLIFFr*	32.04	55.83	71.95	25.88	42.79
CLIFFb + KITRO **(ours)**	**27.67 (4.3↓)**	**43.53 (12.3↓)**	**53.44 (18.5↓)**	**21.04 (4.8↓)**	**34.50 (8.3↓)**

Table 3. Results using alternative HMR base models and rotation representations on 3DPW. * indicates our reproduced results.

Method	PA-MPJPE ↓	MPJPE ↓	PVE ↓
SPIN [15]	59.97	102.12	130.62
SPIN + SMPLify*	47.99	87.06	102.28
SPIN + **ours**	**42.46**	**67.12**	**80.25**
EFTb [9]	54.71	94.02	116.23
EFTb + SMPLify*	44.69	82.39	96.50
EFTb + **ours**	**32.34**	**49.14**	**59.28**
PRoM [7]	42.0	67.6	79.2
PRoM + SMPLify*	35.67	65.21	78.02
PRoM + **ours**	**26.72**	**42.18**	**51.25**

qualitative examples of body poses over the iterations. The initial predictions are misaligned with the visual evidence from the image; our method progressive refines the parameters to match the evidence, while preserving natural poses.

Different HMR Models and Representation. Our method is plug-and-play on top of any HMR method. We test our method on other commonly used base models such as SPIN [15], EFT [9] and a different rotation representation PRoM [7]. As shown in Tab. 3, our method can also make large improvements and outperform previous refinement methods with large margins, especially in MPJPE and PVE which are indicators for global orientation and shape.

5.3. Comparison with the State-of-the-art

In this subsection, we compare our approach with the SOTA human mesh refinement methods, as presented in Tab. 2. The upper segment of Tab. 2 first represents our HMR base model CLIFFb [18] as stated previously. The bottom segment ensures a fair comparison as all utilize ground truth 2D keypoints. By comparing with optimization-based methods in the second segment (*e.g.*, CLIFFr [18] and SMPLify [1] adapted by ours), our KITRO demonstrates a significant enhancement over them (about 15% improvement in PA-MPJPE, 20% improvement in MPJPE, 25% improvement in PVE than SOTA refinement method [18]). Notably, our method achieves a 27.67mm PA-MPJPE joint error on the 3DPW dataset, which is comparably close to the 26mm accuracy obtained in the dataset's creation process [32].

Table 4. Our refinement results on top of CLIFFr on 3DPW.

Method	PA-MPJPE ↓	MPJPE ↓	PVE ↓
CLIFFb	43.76	73.67	91.58
+ CLIFFr	32.04	55.83	71.95
+ CLIFFr + **ours**	**26.21**	**46.96**	**57.53**

This advancement is attributed to our explicit depth and kinematic-tree modeling, which consistently yields a precisely refined pose fitting with 2D clues more ideally.

We also compare our visualization result with other methods as shown in Fig. 6. Based on the same initialized human pose, our approach aligns more accurately with 2D evidence, as evident in the upright position of the left hand in the first row, unlike the incorrect direction seen with other methods. Additionally, while SPMLify [1] and CLIFFr [18] may correctly fit 2D evidence, depth ambiguities often lead to penetration issues, as in the second row in Fig. 6. Our explicit depth modeling effectively avoids such problems.

5.4. Analysis

In this subsection, we analyze KITRO in three aspects: its enhancement on top of other refinement methods; its adaptability to similar parametric models like MANO [25] in hand mesh recovery; and its overall improvement coverage quantified by the improvement samples proportion.

Refinement on Top of Refinement. Previous subsection shows the result on top of HMR base models (*e.g.*, CLIFFb, SPIN, EFTb). Here we further examine whether our method can get extra improvement on top of other refinement methods. Tab. 4 shows KITRO can further enhance the refinement result from SOTA refinement method CLIFFr, showing KITRO can make gains they are unable to achieve.

Generalizing to Other Parametric Model. Beyond the SMPL model, our method's versatility is further illustrated through its application to the MANO model [25], a commonly used parametric model for hand mesh recovery. Lacking camera prediction in the pre-trained MANO baseline, we utilized ground truth camera translation for these experiments and ensured the same setting for other adapted refinement methods for fair comparison. The results shown

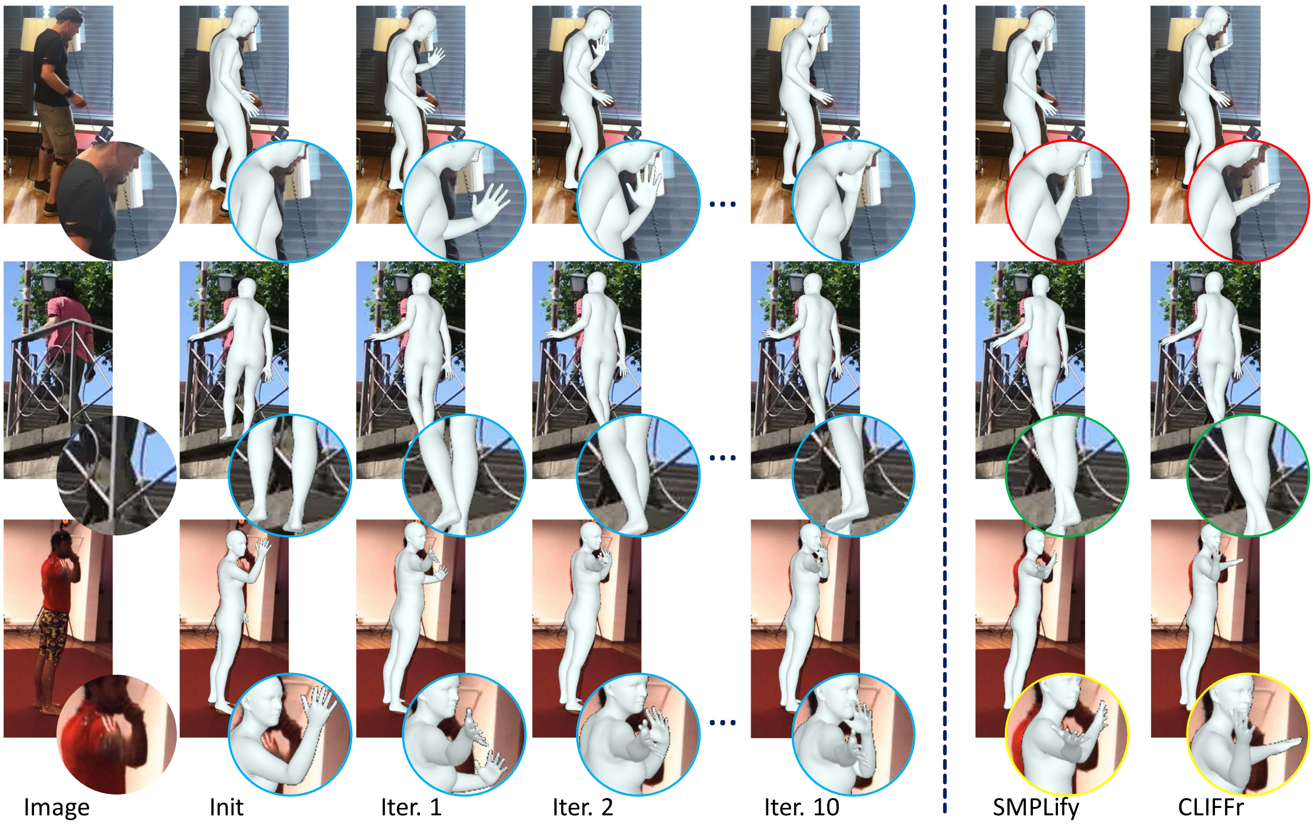


Figure 6. Refinement result over iterations (left part) and comparison with other human mesh refinement methods (right part). Left: The blue circles highlight the refinement progress where our method iteratively refines the misaligned bone direction to fit with 2D. Right: the red circles (first row) highlight the refinement of other methods still misaligned with the 2D; the green ones (second row) highlight the penetration problem caused by their depth inaccuracy; and the yellow circles (third row) have both above problems.

Table 5. Result of our method adopted to MANO model on Freihand. PA-PVE denotes the PVE result after Procrustes aligned.

Method	PA-MPJPE ↓	PA-PVE ↓
Baseline (MANO)*	7.82	8.01
Baseline + SMPLify*	4.66	5.07
Baseline **+ ours**	**4.06**	**4.58**

in Tab. 5 demonstrate that our method not only refines human mesh predictions effectively but is also suitable for other parametric models like MANO.

Comprehensive Improvement Across Joints and Samples. Our method not only enhances average performance but also ensures equitable and widespread effectiveness across joints and samples. As depicted in Fig. 1b, it effectively enhances both proximal and distal joints, while other methods may sacrifice the proximal ones. Additionally, a significant portion of individual samples improved after our refinement: 88% on 3DPW and 92% on Human3.6M as shown in Sec. E of the Supplementary. This underscores our method's comprehensive efficacy.

6. Limitations

KITRO relies on the initial predicted mesh as a reference for hypothesis selection, which can lead to deviations in cases of poor initial mesh predictions. This is particularly evident in scenarios of person misidentification errors [3], where the initial 3D mesh is totally mismatched with the 2D evidence. More discussion is in Sec. F of the Supplementary. In addition, similar to other human mesh refinement methods, KITRO relies on the accuracy of the input 2D pose. A detailed discussion and evaluation on the detected and noisy 2D keypoints can be found in Sec. G of the Supplementary.

7. Conclusion

Motivated by the inadequate depth accuracy and suboptimal proximal joint performance in existing human mesh refinement methods, we propose KITRO. By explicitly modeling joint depth in closed-form solution along the human kinematic-tree, we can then use the decision tree to trace all hypotheses and select the most likely one. KITRO demonstrates superior accuracy and comprehensive improvement across various datasets and baseline models.

Acknowledgement

This research / project is supported by the Ministry of Education, Singapore, under its MOE Academic Research Fund Tier 2 (STEM RIE2025 MOE-T2EP20220-0015).

[3] Predicting the wrong person in an image with multiple individuals.

References

[1] 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 *European Conference on Computer Vision (ECCV)*, pages 561–578. Springer, 2016. 1, 2, 3, 7

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7291–7299, 2017. 1

[3] Hanbyel Cho, Yooshin Cho, Jaesung Ahn, and Junmo Kim. Implicit 3d human mesh recovery using consistency with pose and shape from unseen-view. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21148–21158, 2023. 2

[4] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *European Conference on Computer Vision (ECCV)*, pages 769–787. Springer, 2020. 6, 7

[5] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3d human pose and shape from a video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[6] Kerui Gu, Rongyu Chen, and Angela Yao. On the calibration of human pose estimation. *arXiv preprint arXiv:2311.17105*, 2023. 2

[7] Kerui Gu, Zhihao Li, Shiyong Liu, Jianzhuang Liu, Songcen Xu, Youliang Yan, Michael Bi Mi, Kenji Kawaguchi, and Angela Yao. Learning unorthogonalized matrices for rotation estimation. *arXiv preprint arXiv:2312.00462*, 2023. 7

[8] Shanyan Guan, Jingwei Xu, Michelle Zhang He, Yunbo Wang, Bingbing Ni, and Xiaokang Yang. Out-of-domain human mesh reconstruction via dynamic bilevel online adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(4):5070–5086, 2022. 6, 7

[9] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human model fitting towards in-the-wild 3d human pose estimation. In *2021 International Conference on 3D Vision (3DV)*, pages 42–52. IEEE, 2021. 1, 2, 3, 6, 7

[10] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 3, 6

[11] Imry Kissos, Lior Fritz, Matan Goldman, Omer Meir, Eduard Oks, and Mark Kliger. Beyond weak perspective for monocular 3d human pose estimation. In *European Conference on Computer Vision Workshops (ECCVW)*, pages 541–554. Springer, 2020. 3

[12] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[13] Muhammed Kocabas, Chun-Hao P Huang, Otmar Hilliges, and Michael J Black. Pare: Part attention regressor for 3d human body estimation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 11127–11137, 2021. 1, 3

[14] 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 *IEEE International Conference on Computer Vision (ICCV)*, pages 11035–11045, 2021. 3

[15] 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 *IEEE International Conference on Computer Vision (ICCV)*, 2019. 1, 2, 3, 6, 7

[16] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3383–3393, 2021. 2, 5

[17] S. Li, L. Ke, K. Pratama, Y.-W. Tai, C.-K. Tang, and K.-T. Cheng. Cascaded deep monocular 3d human pose estimation with evolutionary training data. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020. 2

[18] 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 *European Conference on Computer Vision (ECCV)*, pages 590–606. Springer, 2022. 1, 2, 3, 6, 7

[19] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *IEEE International Conference on Computer Vision (ICCV)*, pages 12939–12948, 2021. 2

[20] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1954–1963, 2021. 2

[21] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (TOG)*, 34(6):248:1–248:16, 2015. 1, 2, 5

[22] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *European Conference on Computer Vision (ECCV)*, pages 752–768. Springer, 2020. 2

[23] Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Neuralannot: Neural annotator for 3d human mesh training sets. *arXiv preprint arXiv:2011.11232*, 2020. 2

[24] Lea Muller, Ahmed AA Osman, Siyu Tang, Chun-Hao P Huang, and Michael J Black. On self-contact and human pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9990–9999, 2021. 2

[25] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017. 7

[26] Yu Rong, Ziwei Liu, Cheng Li, Kaidi Cao, and Chen Change Loy. Delving deep into hybrid annotations for 3d human

recovery in the wild. In *IEEE International Conference on Computer Vision (ICCV)*, pages 5340–5348, 2019. 2

[27] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 84–93, 2020. 2

[28] Akash Sengupta, Ignas Budvytis, and Roberto Cipolla. Synthetic training for accurate 3d human pose and shape estimation in the wild. *arXiv preprint arXiv:2009.10013*, 2020. 2

[29] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5693–5703, 2019. 2

[30] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *European Conference on Computer Vision (ECCV)*, pages 20–38. Springer International Publishing, 2018. 2

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017. 2

[32] 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 *European Conference on Computer Vision (ECCV)*, pages 601–617, 2018. 1, 7

[33] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 43(10): 3349–3364, 2020. 1

[34] Wenjia Wang, Yongtao Ge, Haiyi Mei, Zhongang Cai, Qingping Sun, Yanjun Wang, Chunhua Shen, Lei Yang, and Taku Komura. Zolly: Zoom focal length correctly for perspective-distorted human mesh reconstruction. *arXiv preprint arXiv:2303.13796*, 2023. 3

[35] T. Xu and W. Takano. Graph stacked hourglass networks for 3d human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[36] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:38571–38584, 2022. 1

[37] P. Yao, Z. Fang, F. Wu, Y. Feng, and J. Li. Densebody: Directly regressing dense 3d human pose and shape from a single color image. *arXiv preprint arXiv:1903.10153*, 2019. 2