3D Human Mesh Estimation from Virtual Markers

Xiaoxuan Ma[1] Jiajun Su[1] Chunyu Wang[3 *] Wentao Zhu[1] Yizhou Wang[1, 2, 4]

[1] School of Computer Science, Center on Frontiers of Computing Studies, Peking University
[2] Inst. for Artificial Intelligence, Peking University
[3] Microsoft Research Asia
[4] Nat'l Eng. Research Center of Visual Technology

{maxiaoxuan, sujiajun, wtzhu, yizhou.wang}@pku.edu.cn, chnuwa@microsoft.com

Abstract

Inspired by the success of volumetric 3D pose estimation, some recent human mesh estimators propose to estimate 3D skeletons as intermediate representations, from which, the dense 3D meshes are regressed by exploiting the mesh topology. However, body shape information is lost in extracting skeletons, leading to mediocre performance. The advanced motion capture systems solve the problem by placing dense physical markers on the body surface, which allows to extract realistic meshes from their non-rigid motions. However, they cannot be applied to wild images without markers. In this work, we present an intermediate representation, named virtual markers, which learns 64 landmark keypoints on the body surface based on the large-scale mocap data in a generative style, mimicking the effects of physical markers. The virtual markers can be accurately detected from wild images and can reconstruct the intact meshes with realistic shapes by simple interpolation. Our approach outperforms the state-of-the-art methods on three datasets. In particular, it surpasses the existing methods by a notable margin on the SURREAL dataset, which has diverse body shapes. Code is available at https://github.com/ShirleyMaxx/VirtualMarker.

1. Introduction

3D human mesh estimation aims to estimate the 3D positions of the mesh vertices that are on the body surface. The task has attracted a lot of attention from the computer vision and computer graphics communities [3, 10, 18, 24, 26, 29, 34, 36, 41, 49] because it can benefit many applications such as virtual reality [14]. Recently, the deep learning-based methods [7, 18, 28] have significantly advanced the accuracy on the benchmark datasets.

Figure 1. Mesh estimation results on four examples with different body shapes. Pose2Mesh [7] which uses 3D skeletons as the intermediate representation fails to predict accurate shapes. Our virtual marker-based method obtains accurate estimates.

The pioneer methods [18, 49] propose to regress the pose and shape parameters of the mesh models such as SMPL [35] directly from images. While straightforward, their accuracy is usually lower than the state-of-the-arts. The first reason is that the mapping from the image features to the model parameters is highly non-linear and suffers from image-model misalignment [28]. Besides, existing mesh datasets [15, 27, 37, 52] are small and limited to simple labo-

*Corresponding author

ratory environments due to the complex capturing process. The lack of sufficient training data severely limits its performance.

Recently, some works [25, 38] begin to formulate mesh estimation as a dense 3D keypoint detection task inspired by the success of volumetric pose estimation [42, 43, 45, 48, 57, 63]. For example, in [25, 38], the authors propose to regress the 3D positions of all vertices. However, it is computationally expensive because it has more than several thousand vertices. Moon and Lee [38] improve the efficiency by decomposing the 3D heatmaps into multiple 1D heatmaps at the cost of mediocre accuracy. Choi *et al*. [7] propose to first detect a sparser set of skeleton joints in the images, from which the dense 3D meshes are regressed by exploiting the mesh topology. The methods along this direction have attracted increasing attention [7, 28, 53] due to two reasons. First, the proxy task of 3D skeleton estimation can leverage the abundant 2D pose datasets which notably improves the accuracy. Second, mesh regression from the skeletons is efficient. However, important information about the body shapes is lost in extracting the 3D skeletons, which is largely overlooked previously. As a result, different types of body shapes, such as lean or obese, cannot be accurately estimated (see Figure 1).

The professional marker-based motion capture (mocap) method MoSh [34] places physical markers on the body surface and explore their subtle non-rigid motions to extract meshes with accurate shapes. However, the physical markers limit the approach to be used in laboratory environments. We are inspired to think whether we can identify a set of landmarks on the mesh as virtual markers, *e.g*., elbow and wrist, that can be detected from wild images, and allow to recover accurate body shapes? The desired virtual markers should satisfy several requirements. First, the number of markers should be much smaller than that of the mesh vertices so that we can use volumetric representations to efficiently estimate their 3D positions. Second, the markers should capture the mesh topology so that the intact mesh can be accurately regressed from them. Third, the virtual markers have distinguishable visual patterns so that they can be detected from images.

In this work, we present a learning algorithm based on archetypal analysis [12] to identify a subset of mesh vertices as the virtual markers that try to satisfy the above requirements to the best extent. Figure 2 shows that the learned virtual markers coarsely outline the body shape and pose which paves the way for estimating meshes with accurate shapes. Then we present a simple framework for 3D mesh estimation on top of the representation as shown in Figure 3. It first learns a 3D keypoint estimation network based on [45] to detect the 3D positions of the virtual markers. Then we recover the intact mesh simply by interpolating them. The interpolation weights are pre-trained in the representation learning step and will be adjusted by a light network based on the prediction confidences of the virtual markers for each image.

We extensively evaluate our approach on three benchmark datasets. It consistently outperforms the state-of-the-art methods on all of them. In particular, it achieves a significant gain on the SURREAL dataset [51] which has a variety of body shapes. Our ablation study also validates the advantages of the virtual marker representation in terms of recovering accurate shapes. Finally, the method shows decent generalization ability and generates visually appealing results for the wild images.

2. Related work

2.1. Optimization-based mesh estimation

Before deep learning dominates this field, 3D human mesh estimation [2, 27, 34, 40, 58] is mainly optimization-based, which optimizes the parameters of the human mesh models to match the observations. For example, Loper *et al*. [34] propose MoSh that optimizes the SMPL parameters to align the mesh with the 3D marker positions. It is usually used to get GT 3D meshes for benchmark datasets because of its high accuracy. Later works propose to optimize the model parameters or mesh vertices based on 2D image cues [2, 11, 27, 40, 58]. They extract intermediate representations such as 2D skeletons from the images and optimize the mesh model by minimizing the discrepancy between the model projection and the intermediate representations such as the 2D skeletons. These methods are usually sensitive to initialization and suffer from local optimum.

2.2. Learning-based mesh estimation

Recently, most works follow the learning-based framework and have achieved promising results. Deep networks [18, 24, 26, 36, 49] are used to regress the SMPL parameters from image features. However, learning the mapping from the image space to the parameter space is highly nonlinear [38]. In addition, they suffer from the misalignment between the meshes and image pixels [60]. These problems make it difficult to learn an accurate yet generalizable model.

Some works propose to introduce proxy tasks to get intermediate representations first, hoping to alleviate the learning difficulty. In particular, intermediate representations of physical markers [59], IUV images [55, 60–62], body part segmentation masks [23, 27, 39, 50] and body skeletons [7, 28, 47, 53] have been proposed. In particular, THUNDR [59] first estimates the 3D locations of physical markers from images and then reconstructs the mesh from the 3D markers. The physical markers can be interpreted as a simplified representation of body shape and pose. Although it is very accurate, it cannot be applied to wild images without markers. In contrast, body skeleton is a popular human representation

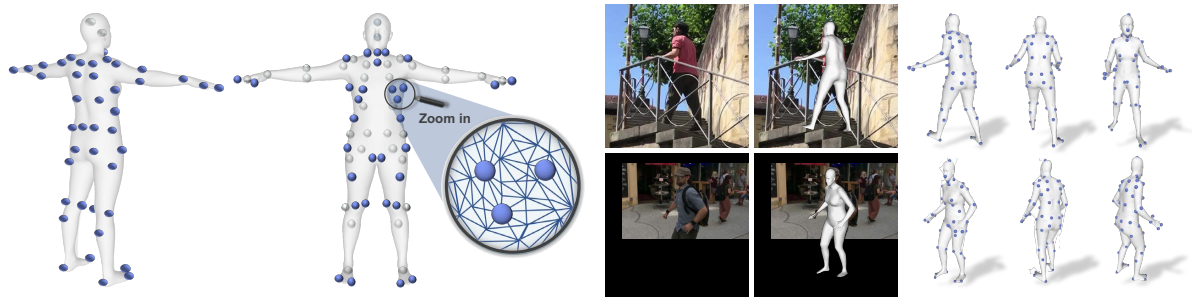

Figure 2. **Left:** The learned virtual markers (blue balls) in the back and front views. The grey balls mean they are invisible in the front view. The virtual markers act similarly to physical body markers and approximately outline the body shape. **Right:** Mesh estimation results by our approach, from left to right are input image, estimated 3D mesh overlayed on the image, and three different viewpoints showing the estimated 3D mesh with our intermediate predicted virtual markers (blue balls), respectively.

that can be robustly detected from wild images. Choi *et al*. [7] propose to first estimate the 3D skeletons, and then estimate the intact mesh from them. However, accurate body shapes are difficult to be recovered from the oversimplified 3D skeletons.

Our work belongs to the learning-based class and is related to works that use physical markers or skeletons as intermediate representations. But different from them, we propose a novel intermediate representation, named *virtual markers*, which is more expressive to reduce the ambiguity in pose and shape estimation than body skeletons and can be applied to wild images.

3. Method

In this section, we describe the details of our approach. First, Section 3.1 introduces how we learn the virtual marker representation from mocap data. Then we present the overall framework for mesh estimation from an image in Section 3.2. At last, Section 3.3 discusses the loss functions and training details.

3.1. The virtual marker representation

We represent a mesh by a vector of vertex positions $\mathbf{x} \in \mathbb{R}^{3M}$ where M is the number of mesh vertices. Denote a mocap dataset such as [15] with N meshes as $\widehat{\mathbf{X}} = [\mathbf{x}_1, ..., \mathbf{x}_N] \in \mathbb{R}^{3M \times N}$. To unveil the latent structure among vertices, we reshape it to $\mathbf{X} \in \mathbb{R}^{3N \times M}$ with each column $\mathbf{x}_i \in \mathbb{R}^{3N}$ representing all possible positions of the i^{th} vertex in the dataset [15].

The rank of $\mathbf{X}$ is smaller than M because the mesh representation is smooth and redundant where some vertices can be accurately reconstructed by the others. While it seems natural to apply PCA [17] to $\mathbf{X}$ to compute the eigenvectors as virtual markers for reconstructing others, there is no guarantee that the virtual markers correspond to the mesh vertices, making them difficult to be detected from images. Instead, we aim to learn K virtual markers $\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_K] \in \mathbb{R}^{3N \times K}$ that try to satisfy the following two requirements to the greatest extent. First, they can accurately reconstruct the intact mesh $\mathbf{X}$ by their linear combinations: $\mathbf{X} = \mathbf{ZA}$, where $\mathbf{A} \in \mathbb{R}^{K \times M}$ is a coefficient matrix that encodes the spatial relationship between the virtual markers and the mesh vertices. Second, they should have distinguishable visual patterns in images so that they can be easily detected from images. Ideally, they can be on the body surface as the meshes.

Type	Formula	Reconst. Error (mm) ↓
Original	$\|\|\mathbf{X} - \mathbf{XBA}\|\|_F^2$	11.67
Symmetric	$\|\|\mathbf{X} - \mathbf{X}\widetilde{\mathbf{B}}^{sym}\widetilde{\mathbf{A}}^{sym}\|\|_F^2$	10.98

Table 1. The reconstruction errors using the original and the symmetric sets of markers on the H3.6M dataset [15], respectively. The errors are small indicating that they are sufficiently expressive and can reconstruct all vertices accurately.

We apply archetypal analysis [4, 12] to learn $\mathbf{Z}$ by minimizing a reconstruction error with two additional constraints: (1) each vertex $\mathbf{x}_i$ can be reconstructed by convex combinations of $\mathbf{Z}$, and (2) each marker $\mathbf{z}_i$ should be convex combinations of the mesh vertices $\mathbf{X}$:

$$\min_{\substack{\boldsymbol{\alpha}_i \in \Delta_K \, for \, 1 \le i \le M, \\ \boldsymbol{\beta}_j \in \Delta_M \, for \, 1 \le j \le K}} ||\mathbf{X} - \mathbf{XBA}||_F^2, \tag{1}$$

where $\mathbf{A} = [\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_M] \in \mathbb{R}^{K \times M}$, each $\boldsymbol{\alpha}$ resides in the simplex $\Delta_K \triangleq \{\boldsymbol{\alpha} \in \mathbb{R}^K \text{ s.t. } \boldsymbol{\alpha} \succeq 0 \text{ and } ||\boldsymbol{\alpha}||_1 = 1\}$, and $\mathbf{B} = [\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_K] \in \mathbb{R}^{M \times K}$, $\boldsymbol{\beta}_j \in \Delta_M$. We adopt Active-set algorithm [4] to solve objective (1) and obtain the learned virtual markers $\mathbf{Z} = \mathbf{XB} \in \mathbb{R}^{3N \times K}$. As shown in [4, 12], the two constraints encourage the virtual markers $\mathbf{Z}$ to unveil the latent structure among vertices, therefore they learn to be close to the extreme points of the mesh and located on the body surface as much as possible.

Post-processing. Since human body is left-right symmetric, we adjust $\mathbf{Z}$ to reflect the property. We first replace each $\mathbf{z}_i \in \mathbf{Z}$ by its nearest vertex on the mesh and obtain $\widetilde{\mathbf{Z}} \in \mathbb{R}^{3 \times K}$. This step allows us to compute the left or right counterpart

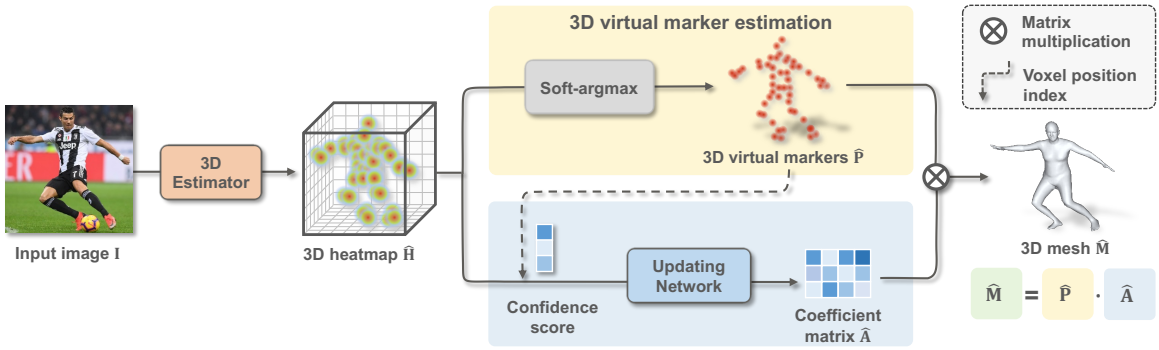

Figure 3. Overview of our framework. Given an input image $\mathbf{I}$, it first estimates the 3D positions $\hat{\mathbf{P}}$ of the virtual markers. Then we update the coefficient matrix $\hat{\mathbf{A}}$ based on the estimation confidence scores $\mathbf{C}$ of the virtual markers. Finally, the complete human mesh can be simply recovered by linear multiplication $\hat{\mathbf{M}} = \hat{\mathbf{P}}\hat{\mathbf{A}}$.

of each marker. Then we replace the markers in the right body with the symmetric vertices in the left body and obtain the symmetric markers $\widetilde{\mathbf{Z}}^{sym} \in \mathbb{R}^{3\times K}$. Finally we update $\mathbf{B}$ and $\mathbf{A}$ by minimizing $||\mathbf{X} - \mathbf{X}\widetilde{\mathbf{B}}^{sym}\widetilde{\mathbf{A}}^{sym}||_F^2$ subject to $\widetilde{\mathbf{Z}}^{sym} = \mathbf{X}\widetilde{\mathbf{B}}^{sym}$. More details are elaborated in the supplementary.

Figure 2 shows the virtual markers learned on the mocap dataset [15] after post-processing. They are similar to the physical markers and approximately outline the body shape which agrees with our expectations. They are roughly evenly distributed on the surface of the body, and some of them are located close to the body keypoints, which have distinguishable visual patterns to be accurately detected. Table 1 shows the reconstruction errors of using original markers $\mathbf{XB}$ and the symmetric markers $\mathbf{X}\widetilde{\mathbf{B}}^{sym}$. Both can reconstruct meshes accurately.

3.2. Mesh estimation framework

On top of the virtual markers, we present a simple yet effective framework for end-to-end 3D human mesh estimation from a single image. As shown in Figure 3, it consists of two branches. The first branch uses a volumetric CNN [45] to estimate the 3D positions $\hat{\mathbf{P}}$ of the markers, and the second branch reconstructs the full mesh $\hat{\mathbf{M}}$ by predicting a coefficient matrix $\hat{\mathbf{A}}$:

$$\hat{\mathbf{M}} = \hat{\mathbf{P}}\hat{\mathbf{A}}. \tag{2}$$

We will describe the two branches in more detail.

3D marker estimation. We train a neural network to estimate a 3D heatmap $\hat{\mathbf{H}} = [\hat{\mathbf{H}}_1, ..., \hat{\mathbf{H}}_K] \in \mathbb{R}^{K\times D\times H\times W}$ from an image. The heatmap encodes per-voxel likelihood of each marker. There are $D \times H \times W$ voxels in total which are used to discretize the 3D space. The 3D position $\hat{\mathbf{P}}_z \in \mathbb{R}^3$ of each marker is computed as the center of mass of the corresponding heatmap $\hat{\mathbf{H}}_z$ [45] as follows:

$$\hat{\mathbf{P}}_z = \sum_{d=1}^{D}\sum_{h=1}^{H}\sum_{w=1}^{W}(d,h,w)\cdot\hat{\mathbf{H}}_z(d,h,w). \tag{3}$$

The positions of all markers are represented as $\hat{\mathbf{P}} = [\hat{\mathbf{P}}_1, \hat{\mathbf{P}}_2, \cdots, \hat{\mathbf{P}}_K]$.

Interpolation. Ideally, if we have accurate estimates for all virtual markers $\hat{\mathbf{P}}$, then we can recover the complete mesh by simply multiplying $\hat{\mathbf{P}}$ with a fixed coefficient matrix $\widetilde{\mathbf{A}}^{sym}$ with sufficient accuracy as validated in Table 1. However, in practice, some markers may have large estimation errors because they may be occluded in the monocular setting. Note that this happens frequently. For example, the markers in the back will be occluded when a person is facing the camera. As a result, inaccurate markers positions may bring large errors to the final mesh if we directly multiply them with the fixed matrix $\widetilde{\mathbf{A}}^{sym}$.

Our solution is to rely more on those accurately detected markers. To that end, we propose to update the coefficient matrix based on the estimation confidence scores of the markers. In practice, we simply take the heatmap score at the estimated positions of each marker, *i.e.* $\hat{\mathbf{H}}_z(\hat{\mathbf{P}}_z)$, and feed them to a single fully-connected layer to obtain the coefficient matrix $\hat{\mathbf{A}}$. Then the mesh is reconstructed by $\hat{\mathbf{M}} = \hat{\mathbf{P}}\hat{\mathbf{A}}$.

3.3. Training

We train the whole network end-to-end in a supervised way. The overall loss function is defined as:

$$\mathcal{L} = \lambda_{vm}\mathcal{L}_{vm} + \lambda_c\mathcal{L}_{conf} + \lambda_m\mathcal{L}_{mesh}. \tag{4}$$

Virtual marker loss. We define $\mathcal{L}_{vm}$ as the L_1 distance between the predicted 3D virtual markers $\hat{\mathbf{P}}$ and the GT $\hat{\mathbf{P}}^*$ as follows:

$$\mathcal{L}_{vm} = \|\hat{\mathbf{P}} - \hat{\mathbf{P}}^*\|_1. \tag{5}$$

Note that it is easy to get GT markers $\hat{\mathbf{P}}^*$ from GT meshes as stated in Section 3.1 without additional manual annotations.

Confidence loss. We also require that the 3D heatmaps have reasonable shapes, therefore, the heatmap score at the

voxel containing the GT marker position $\hat{\mathbf{P}}_z^*$ should have the maximum value as in the previous work [16]:

$$\mathcal{L}_{conf} = -\sum_{z=1}^{K} log(\hat{\mathbf{H}}_z(\hat{\mathbf{P}}_z^*)). \quad (6)$$

Mesh loss. Following [38], we define $\mathcal{L}_{mesh}$ as a weighted sum of four losses:

$$\mathcal{L}_{mesh} = \mathcal{L}_{vertex} + \mathcal{L}_{pose} + \mathcal{L}_{normal} + \lambda_e \mathcal{L}_{edge}. \quad (7)$$

– **Vertex coordinate loss.** We adopt L_1 loss between predicted 3D mesh coordinates $\hat{\mathbf{M}}$ with GT mesh $\hat{\mathbf{M}}^*$ as:

$$\mathcal{L}_{vertex} = \|\hat{\mathbf{M}} - \hat{\mathbf{M}}^*\|_1. \quad (8)$$

– **Pose loss.** We use L_1 loss between the 3D landmark joints regressed from mesh $\hat{\mathbf{M}}\mathcal{J}$ and the GT joints $\hat{\mathbf{J}}^*$ as:

$$\mathcal{L}_{pose} = \|\hat{\mathbf{M}}\mathcal{J} - \hat{\mathbf{J}}^*\|_1, \quad (9)$$

where $\mathcal{J} \in \mathbb{R}^{M \times J}$ is a pre-defined joint regression matrix in SMPL model [2].

– **Surface losses.** To improve surface smoothness [54], we supervise the normal vector of a triangle face with GT normal vectors by $\mathcal{L}_{normal}$ and the edge length of the predicted mesh with GT length by $\mathcal{L}_{edge}$:

$$\mathcal{L}_{normal} = \sum_f \sum_{\{i,j\} \subset f} \left| \left\langle \frac{\hat{\mathbf{M}}_i - \hat{\mathbf{M}}_j}{\|\hat{\mathbf{M}}_i - \hat{\mathbf{M}}_j\|_2}, \hat{\mathbf{n}}_f^* \right\rangle \right|,$$
$$\mathcal{L}_{edge} = \sum_f \sum_{\{i,j\} \subset f} \left| \|\hat{\mathbf{M}}_i - \hat{\mathbf{M}}_j\|_2 - \|\hat{\mathbf{M}}_i^* - \hat{\mathbf{M}}_j^*\|_2 \right|. \quad (10)$$

where f and $\hat{\mathbf{n}}_f^*$ denote a triangle face in the mesh and its GT unit normal vector, respectively. $\hat{\mathbf{M}}_i$ denote the i^{th} vertex of $\hat{\mathbf{M}}$. * denotes GT.

4. Experiments

4.1. Datasets and metrics

H3.6M [15]. We use (S1, S5, S6, S7, S8) for training and (S9, S11) for testing. As in [7, 18, 31, 32], we report MPJPE and PA-MPJPE for poses that are derived from the estimated meshes. We also report Mean Per Vertex Error (MPVE) for the whole mesh.

3DPW [52] is collected in natural scenes. Following the previous works [23, 31, 32, 59], we use the train set of 3DPW to learn the model and evaluate on the test set. The same evaluation metrics as H3.6M are used.

SURREAL [51] is a large-scale synthetic dataset with GT SMPL annotations and has diverse samples in terms of body shapes, backgrounds, *etc*. We use its training set to train a model and evaluate the test split following [7].

4.2. Implementation Details

We learn 64 virtual markers on the H3.6M [15] training set. We use the same set of markers for all datasets instead of learning a separate set on each dataset. Following [7, 18, 22, 25, 31, 32, 38, 59], we conduct mix-training by using MPI-INF-3DHP [37], UP-3D [27], and COCO [33] training set for experiments on the H3.6M and 3DPW datasets. We adapt a 3D pose estimator [45] with HRNet-W48 [44] as the image feature backbone for estimating the 3D virtual markers. We set the number of voxels in each dimension to be 64, *i.e.* $D = H = W = 64$ for 3D heatmaps. Following [18, 25, 38], we crop every single human region from the input image and resize it to 256×256. We use Adam [21] optimizer to train the whole framework for 40 epochs with a batch size of 32. The learning rates for the two branches are set to 5×10^{-4} and 1×10^{-3}, respectively, which are decreased by half after the 30^{th} epoch. Please refer to the supplementary for more details.

4.3. Comparison to the State-of-the-arts

Results on H3.6M. Table 2 compares our approach to the state-of-the-art methods on the H3.6M dataset. Our method achieves competitive or superior performance. In particular, it outperforms the methods that use skeletons (Pose2Mesh [7], DSD-SATN [47]), body markers (THUNDR) [59], or IUV image [60, 62] as proxy representations, demonstrating the effectiveness of the virtual marker representation.

Results on 3DPW. We compare our method to the state-of-the-art methods on the 3DPW dataset in Table 2. Our approach achieves state-of-the-art results among all the methods, validating the advantages of the virtual marker representation over the skeleton representation used in Pose2Mesh [7], DSD-SATN [47], and other representations like IUV image used in PyMAF [62]. In particular, our approach outperforms I2L-MeshNet [38], METRO [31], and Mesh Graphormer [32] by a notable margin, which suggests that virtual markers are more suitable and effective representations than detecting all vertices directly as most of them are not discriminative enough to be accurately detected.

Results on SURREAL. This dataset has more diverse samples in terms of body shapes. The results are shown in Table 3. Our approach outperforms the state-of-the-art methods by a notable margin, especially in terms of MPVE. Figure 1 shows some challenging cases without cherry-picking. The skeleton representation loses the body shape information so the method [7] can only recover mean shapes. In contrast, our approach generates much more accurate mesh estimation results.

Method	Intermediate Representation	H3.6M			3DPW		
		MPVE↓	MPJPE↓	PA-MPJPE↓	MPVE↓	MPJPE↓	PA-MPJPE↓
† Arnab *et al*. [1] CVPR'19	2D skeleton	-	77.8	54.3	-	-	72.2
† HMMR [19] CVPR'19	-	-	-	56.9	139.3	116.5	72.6
† DSD-SATN [47] ICCV'19	3D skeleton	-	59.1	42.4	-	-	69.5
† VIBE [22] CVPR'20	-	-	65.9	41.5	99.1	82.9	51.9
† TCMR [6] CVPR'21	-	-	62.3	41.1	102.9	86.5	52.7
† MAED [53] ICCV'21	3D skeleton	-	56.3	38.7	92.6	79.1	45.7
SMPLify [2] ECCV'16	2D skeleton	-	-	82.3	-	-	-
HMR [18] CVPR'18	-	96.1	88.0	56.8	152.7	130.0	81.3
GraphCMR [25] CVPR'19	3D vertices	-	-	50.1	-	-	70.2
SPIN [24] ICCV'19	-	-	-	41.1	116.4	96.9	59.2
DenseRac [55] ICCV'19	IUV image	-	76.8	48.0	-	-	-
DecoMR [60] CVPR'20	IUV image	-	60.6	39.3	-	-	-
ExPose [9] ECCV'20	-	-	-	-	-	93.4	60.7
Pose2Mesh [7] ECCV'20	3D skeleton	85.3	64.9	46.3	106.3	88.9	58.3
I2L-MeshNet [38] ECCV'20	3D vertices	65.1	55.7	41.1	110.1	93.2	57.7
PC-HMR [36] AAAI'21	3D skeleton	-	-	-	108.6	87.8	66.9
HybrIK [28] CVPR'21	3D skeleton	65.7	54.4	34.5	86.5	74.1	45.0
METRO [31] CVPR'21	3D vertices	-	54.0	36.7	88.2	77.1	47.9
ROMP [46] ICCV'21	-	-	-	-	108.3	91.3	54.9
Mesh Graphormer [32] ICCV'21	3D vertices	-	51.2	34.5	87.7	74.7	45.6
PARE [23] ICCV'21	Segmentation	-	-	-	88.6	74.5	46.5
THUNDR [59] ICCV'21	3D markers	-	55.0	39.8	88.0	74.8	51.5
PyMaf [62] ICCV'21	IUV image	-	57.7	40.5	110.1	92.8	58.9
ProHMR [26] ICCV'21	-	-	-	41.2	-	-	59.8
OCHMR [20] CVPR'22	2D heatmap	-	-	-	107.1	89.7	58.3
3DCrowdNet [8] CVPR'22	3D skeleton	-	-	-	98.3	81.7	51.5
CLIFF [30] ECCV'22	-	-	**47.1**	32.7	81.2	69.0	43.0
FastMETRO [5] ECCV'22	3D vertices	-	52.2	33.7	84.1	73.5	44.6
VisDB [56] ECCV'22	3D vertices	-	51.0	34.5	85.5	73.5	44.9
Ours	Virtual marker	**58.0**	47.3	**32.0**	**77.9**	**67.5**	**41.3**

Table 2. Comparison to the state-of-the-arts on H3.6M [15] and 3DPW [52] datasets. † means using temporal cues. The methods are not strictly comparable because they may have different backbones and training datasets. We provide the numbers only to show proof-of-concept results.

Method	Intermediate Representation	MPVE↓	MPJPE↓	PA-MPJPE↓
HMR [18] CVPR'18	-	85.1	73.6	55.4
BodyNet [50] ECCV'18	Skel. + Seg.	65.8	-	-
GraphCMR [25] CVPR'19	3D vertices	103.2	87.4	63.2
SPIN [24] ICCV'19	-	82.3	66.7	43.7
DecoMR [60] CVPR'20	IUV image	68.9	52.0	43.0
Pose2Mesh [7] ECCV'20	3D skeleton	68.8	56.6	39.6
PC-HMR [36] AAAI'21	3D skeleton	59.8	51.7	37.9
* DynaBOA [13] TPAMI'22	-	70.7	55.2	34.0
Ours	Virtual marker	**44.7**	**36.9**	**28.9**

Table 3. Comparison to the state-of-the-arts on SURREAL [51] dataset. * means training on the test split with 2D supervisions. "Skel. + Seg." means using skeleton and segmentation together.

No.	Intermediate Representation	MPVE↓	
		H3.6M	SURREAL
(a)	Skeleton	64.4	53.6
(b)	Rand virtual marker	63.0	50.1
(c)	Virtual marker	**58.0**	**44.7**

Table 4. Ablation study of the virtual marker representation for our approach on H3.6M and SURREAL datasets. "Skeleton" means the sparse landmark joint representation is used. "Rand virtual marker" means the virtual markers are randomly selected from all the vertices without learning. (c) is our method, where the learned virtual markers are used.

4.4. Ablation study

Virtual marker representation. We compare our method to two baselines in Table 4. First, in baseline (a), we replace the virtual markers of our method with the skeleton representation. The rest are kept the same as ours (c). Our method achieves a much lower MPVE than the baseline (a), demonstrating that the virtual markers help to estimate body shapes more accurately than the skeletons. In baseline (b), we randomly sample 64 from the 6890 mesh vertices as virtual markers. We repeat the experiment five times and report the average number. We can see that the result is worse than ours, which is because the randomly selected vertices may not be expressive to reconstruct the other vertices or can not

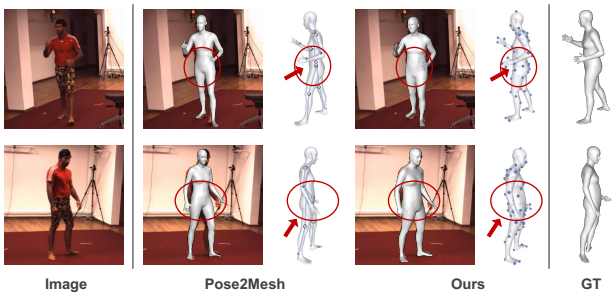

Figure 4. Mesh estimation results of different methods on H3.6M test set. Our method with virtual marker representation gets better shape estimation results than Pose2Mesh which uses skeleton representation. Note the waistline of the body and the thickness of the arm.

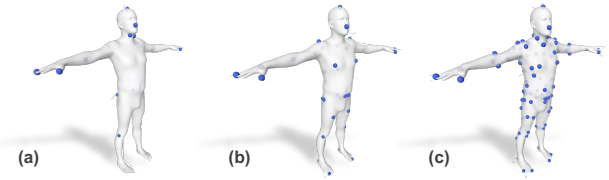

Figure 5. Visualization of the learned virtual markers of different numbers of $K = 16, 32, 96$, from left to right, respectively.

be accurately detected from images as they lack distinguishable visual patterns. The results validate the effectiveness of our learning strategy.

Figure 1 shows some qualitative results on the SURREAL test set. The meshes estimated by the baseline which uses skeleton representation, *i.e*. Pose2Mesh [7], have inaccurate body shapes. This is reasonable because the skeleton is oversimplified and has very limited capability to recover shapes. Instead, it implicitly learns a mean shape for the whole training dataset. In contrast, the mesh estimated by using virtual markers has much better quality due to its strong representation power and therefore can handle different body shapes elegantly. Figure 4 also shows some qualitative results on the H3.6M test set. For clarity, we draw the intermediate representation (blue balls) in it as well.

Number of virtual markers. We evaluate how the number of virtual markers affects estimation quality on H3.6M [15] dataset. Figure 5 visualizes the learned virtual markers, which are all located on the body surface and close to the extreme points of the mesh. This is expected as mentioned in Section 3.1. Table 5 (GT) shows the mesh reconstruction results when we have GT 3D positions of the virtual markers in objective (1). When we increase the number of virtual markers, both mesh reconstruction error (MPVE) and the regressed landmark joint error (MPJPE) steadily decrease. This is expected because using more virtual markers improves the representation power. However, using more

K	GT		Det	
	MPVE↓	MPJPE↓	MPVE↓	MPJPE↓
16	46.8	39.8	58.7	47.8
32	20.1	14.2	58.2	48.3
64	11.0	7.5	**58.0**	**47.3**
96	**9.9**	**5.6**	59.6	48.2

Table 5. Ablation study of the different number of virtual markers (K) on H3.6M [15] dataset. (GT) Mesh reconstruction results when GT 3D positions of the virtual markers are used in objective (1). (Det) Mesh estimation results obtained by our proposed framework when we use different numbers of virtual markers (K).

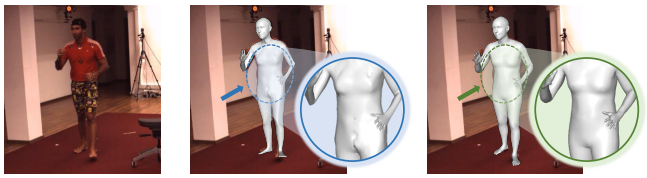

Figure 6. Mesh estimation comparison results when using (a) fixed coefficient matrix $\widetilde{\mathbf{A}}^{sym}$, and (b) updated $\hat{\mathbf{A}}$. Please zoom in to better see the details.

virtual markers cannot guarantee smaller estimation errors when we need to estimate the virtual marker positions from images as in our method. This is because the additional virtual markers may have large estimation errors which affect the mesh estimation result. The results are shown in Table 5 (Det). Increasing the number of virtual markers K steadily reduces the MPVE errors when K is smaller than 96. However, if we keep increasing K, the error begins to increase. This is mainly because some of the newly introduced virtual markers are difficult to detect from images and therefore bring errors to mesh estimation.

Coefficient matrix. We compare our method to a baseline which uses the fixed coefficient matrix $\widetilde{\mathbf{A}}^{sym}$. We show the quality comparison in Figure 6. We can see that the estimated mesh by a fixed coefficient matrix (a) has mostly correct pose and shape but there are also some artifacts on the mesh while using the updated coefficient matrix (b) can get better mesh estimation results. As shown in Table 6, using a fixed coefficient matrix gets larger MPVE and MPJPE errors than using the updated coefficient matrix. This is caused by the estimation errors of virtual markers when occlusion happens, which is inevitable since the virtual markers on the back will be self-occluded by the front body. As a result, inaccurate marker positions would bring large errors to the final mesh estimates if we directly use the fixed matrix.

4.5. Qualitative Results

Figure 7 (top) presents some meshes estimated by our approach on natural images from the 3DPW test set. The

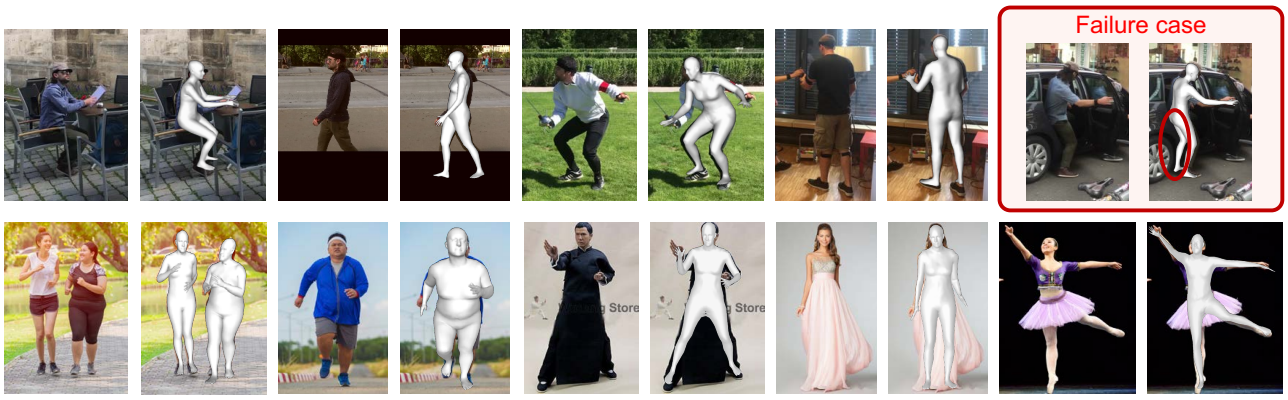

Figure 7. **Top:** Meshes estimated by our approach on images from 3DPW test set. The rightmost case in the dashed box shows a typical failure. **Bottom:** Meshes estimated by our approach on Internet images with challenging cases (extreme shapes or in a long dress).

No.	Method	Fixed $\tilde{\mathbf{A}}^{sym}$	Updated $\hat{\mathbf{A}}$	MPVE↓	MPJPE↓
(a)	Ours (fixed)	✓	✗	64.7	51.6
(b)	Ours	✗	✓	**58.0**	**47.3**

Table 6. Ablation study of the coefficient matrix for our approach on H3.6M dataset. "fixed" means using the fixed coefficient matrix $\tilde{\mathbf{A}}^{sym}$ to reconstruct the mesh.

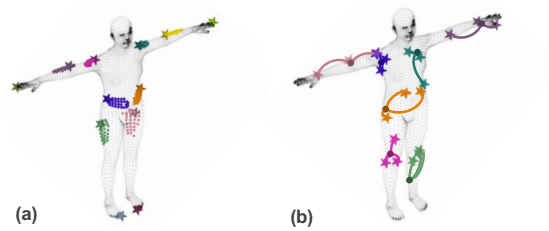

Figure 8. (a) For each virtual marker (represented by a star), we highlight the top 30 most affected vertices (represented by a colored dot) based on average coefficient matrix $\hat{\mathbf{A}}$. (b) For each vertex (dot), we highlight the top 3 virtual markers (star) that contribute the most. We can see that the dependency has a strong locality which improves the robustness when some virtual markers cannot be accurately detected.

rightmost case shows a typical failure where our method has a wrong pose estimate of the left leg due to heavy occlusion. We can see that the failure is constrained to the local region and the rest of the body still gets accurate estimates. We further analyze how inaccurate virtual markers would affect the mesh estimation, *i.e*. when part of human body is occluded or truncated. According to the finally learned coefficient matrix $\hat{\mathbf{A}}$ of our model, we highlight the relationship weights among virtual markers and all vertices in Figure 8. We can see that our model actually learns *local and sparse* dependency between each vertex and the virtual markers, *e.g*. for each vertex, the virtual markers that contribute the most are in a near range as shown in Figure 8 (b). Therefore, in inference, if a virtual marker has inaccurate position estimation due to occlusion or truncation, the dependent vertices may have inaccurate estimates, while the rest will be barely affected. Figure 2 (right) shows more examples where occlusion or truncation occurs, and our method can still get accurate or reasonable estimates robustly. Note that when truncation occurs, our method still guesses the positions of the truncated virtual markers.

Figure 7 (bottom) shows our estimated meshes on challenging cases, which indicates the strong generalization ability of our model on diverse postures and actions in natural scenes. Please refer to the supplementary for more quality results. Note that since the datasets do not provide supervision of head orientation, face expression, hands, or feet, the estimates of these parts are just in canonical poses inevitably. Apart from that, most errors are due to inaccurate 3D virtual marker estimation which may be addressed using more powerful estimators or more diverse training datasets in the future.

5. Conclusion

In this paper, we present a novel intermediate representation *Virtual Marker*, which is more expressive than the prevailing skeleton representation and more accessible than physical markers. It can reconstruct 3D meshes more accurately and efficiently, especially in handling diverse body shapes. Besides, the coefficient matrix in the virtual marker representation encodes spatial relationships among mesh vertices which allows the method to implicitly explore structure priors of human body. It achieves better mesh estimation results than the state-of-the-art methods and shows advanced generalization potential in spite of its simplicity.

Acknowledgement

This work was supported by MOST-2022ZD0114900 and NSFC-62061136001.

References

[1] Anurag Arnab, Carl Doersch, and Andrew Zisserman. Exploiting temporal context for 3d human pose estimation in the wild. In *CVPR*, pages 3395–3404, 2019.

[2] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, pages 561–578, 2016.

[3] Ronan Boulic, Pascal Bécheiraz, Luc Emering, and Daniel Thalmann. Integration of motion control techniques for virtual human and avatar real-time animation. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 111–118, 1997.

[4] Yuansi Chen, Julien Mairal, and Zaid Harchaoui. Fast and robust archetypal analysis for representation learning. In *CVPR*, pages 1478–1485, 2014.

[5] Junhyeong Cho, Kim Youwang, and Tae-Hyun Oh. Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In *ECCV*, 2022.

[6] 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 *CVPR*, pages 1964–1973, 2021.

[7] 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 *ECCV*, pages 769–787, 2020.

[8] Hongsuk Choi, Gyeongsik Moon, JoonKyu Park, and Kyoung Mu Lee. Learning to estimate robust 3d human mesh from in-the-wild crowded scenes. In *CVPR*, pages 1475–1484, June 2022.

[9] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J Black. Monocular expressive body regression through body-driven attention. In *ECCV*, pages 20–40, 2020.

[10] Hai Ci, Mingdong Wu, Wentao Zhu, Xiaoxuan Ma, Hao Dong, Fangwei Zhong, and Yizhou Wang. Gfpose: Learning 3d human pose prior with gradient fields. *arXiv preprint arXiv:2212.08641*, 2022.

[11] Enric Corona, Gerard Pons-Moll, Guillem Alenyà, and Francesc Moreno-Noguer. Learned vertex descent: a new direction for 3d human model fitting. In *ECCV*, pages 146–165. Springer, 2022.

[12] Adele Cutler and Leo Breiman. Archetypal analysis. *Technometrics*, 36(4):338–347, 1994.

[13] Shanyan Guan, Jingwei Xu, Michelle Z He, Yunbo Wang, Bingbing Ni, and Xiaokang Yang. Out-of-domain human mesh reconstruction via dynamic bilevel online adaptation. *IEEE TPAMI*, 2022.

[14] Yinghao Huang, Federica Bogo, Christoph Lassner, Angjoo Kanazawa, Peter V Gehler, Javier Romero, Ijaz Akhter, and Michael J Black. Towards accurate marker-less human shape and pose estimation over time. In *3DV*, pages 421–430, 2017.

[15] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE TPAMI*, 36(7):1325–1339, 2013.

[16] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *ICCV*, pages 7718–7727, 2019.

[17] Ian T Jolliffe. Principal components in regression analysis. In *Principal component analysis*, pages 129–155. Springer, 1986.

[18] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131, 2018.

[19] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *CVPR*, pages 5614–5623, 2019.

[20] Rawal Khirodkar, Shashank Tripathi, and Kris Kitani. Occluded human mesh recovery. In *CVPR*, pages 1715–1725, June 2022.

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[22] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, pages 5253–5263, 2020.

[23] Muhammed Kocabas, Chun-Hao P. Huang, Otmar Hilliges, and Michael J. Black. Pare: Part attention regressor for 3d human body estimation. In *ICCV*, pages 11127–11137, October 2021.

[24] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, pages 2252–2261, 2019.

[25] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, pages 4501–4510, 2019.

[26] Nikos Kolotouros, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis. Probabilistic modeling for human mesh recovery. In *ICCV*, pages 11605–11614, October 2021.

[27] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black, and Peter V Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *CVPR*, pages 6050–6059, 2017.

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

[29] Yong-Lu Li, Liang Xu, Xinpeng Liu, Xijie Huang, Yue Xu, Shiyi Wang, Hao-Shu Fang, Ze Ma, Mingyang Chen, and Cewu Lu. Pastanet: Toward human activity knowledge engine. In *CVPR*, pages 382–391, 2020.

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

[31] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, pages 1954–1963, 2021.

[32] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *ICCV*, pages 12939–12948, 2021.

[33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence

Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014.

[34] Matthew Loper, Naureen Mahmood, and Michael J Black. Mosh: Motion and shape capture from sparse markers. *TOG*, 33(6):1–13, 2014.

[35] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *TOG*, 34(6):1–16, 2015.

[36] Tianyu Luan, Yali Wang, Junhao Zhang, Zhe Wang, Zhipeng Zhou, and Yu Qiao. Pc-hmr: Pose calibration for 3d human mesh recovery from 2d images/videos. In *AAAI*, pages 2269–2276, 2021.

[37] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3DV*, pages 506–516, 2017.

[38] 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 *ECCV*, pages 752–768, 2020.

[39] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *3DV*, pages 484–494. IEEE, 2018.

[40] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, pages 10975–10985, 2019.

[41] Liliana Lo Presti and Marco La Cascia. 3d skeleton-based human action classification: A survey. *Pattern Recognition*, 53:130–147, 2016.

[42] Haibo Qiu, Chunyu Wang, Jingdong Wang, Naiyan Wang, and Wenjun Zeng. Cross view fusion for 3d human pose estimation. In *ICCV*, pages 4342–4351, 2019.

[43] Jiajun Su, Chunyu Wang, Xiaoxuan Ma, Wenjun Zeng, and Yizhou Wang. Virtualpose: Learning generalizable 3d human pose models from virtual data. In *ECCV*, pages 55–71. Springer, 2022.

[44] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, pages 5693–5703, 2019.

[45] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, pages 529–545, 2018.

[46] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people. In *ICCV*, pages 11179–11188, 2021.

[47] Yu Sun, Yun Ye, Wu Liu, Wenpeng Gao, Yili Fu, and Tao Mei. Human mesh recovery from monocular images via a skeleton-disentangled representation. In *ICCV*, pages 5349–5358, 2019.

[48] Hanyue Tu, Chunyu Wang, and Wenjun Zeng. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In *ECCV*, pages 197–212. Springer, 2020.

[49] Hsiao-Yu Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *NIPS*, volume 30, 2017.

[50] Gul Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *ECCV*, pages 20–36, 2018.

[51] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, pages 109–117, 2017.

[52] Timo von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *ECCV*, pages 601–617, 2018.

[53] Ziniu Wan, Zhengjia Li, Maoqing Tian, Jianbo Liu, Shuai Yi, and Hongsheng Li. Encoder-decoder with multi-level attention for 3d human shape and pose estimation. In *ICCV*, pages 13033–13042, 2021.

[54] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*, pages 52–67, 2018.

[55] Yuanlu Xu, Song-Chun Zhu, and Tony Tung. Denserac: Joint 3d pose and shape estimation by dense render-and-compare. In *ICCV*, pages 7760–7770, 2019.

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

[57] Hang Ye, Wentao Zhu, Chunyu Wang, Rujie Wu, and Yizhou Wang. Faster voxelpose: Real-time 3d human pose estimation by orthographic projection. In *ECCV*, pages 142–159. Springer, 2022.

[58] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints. In *CVPR*, pages 2148–2157, 2018.

[59] Mihai Zanfir, Andrei Zanfir, Eduard Gabriel Bazavan, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Thundr: Transformer-based 3d human reconstruction with markers. In *ICCV*, pages 12971–12980, 2021.

[60] Wang Zeng, Wanli Ouyang, Ping Luo, Wentao Liu, and Xiaogang Wang. 3d human mesh regression with dense correspondence. In *CVPR*, pages 7054–7063, 2020.

[61] Hongwen Zhang, Jie Cao, Guo Lu, Wanli Ouyang, and Zhenan Sun. Learning 3d human shape and pose from dense body parts. *IEEE TPAMI*, 44(5):2610–2627, 2022.

[62] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In *ICCV*, pages 11446–11456, 2021.

[63] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenyu Liu, and Wenjun Zeng. Voxeltrack: Multi-person 3d human pose estimation and tracking in the wild. *IEEE TPAMI*, 45(2):2613–2626, 2022.