Learning Human Mesh Recovery in 3D Scenes

Zehong Shen Zhi Cen Sida Peng Qing Shuai Hujun Bao Xiaowei Zhou[†]
State Key Lab of CAD&CG, Zhejiang University

Abstract

We present a novel method for recovering the absolute pose and shape of a human in a pre-scanned scene given a single image. Unlike previous methods that perform scene-aware mesh optimization, we propose to first estimate absolute position and dense scene contacts with a sparse 3D CNN, and later enhance a pretrained human mesh recovery network by cross-attention with the derived 3D scene cues. Joint learning on images and scene geometry enables our method to reduce the ambiguity caused by depth and occlusion, resulting in more reasonable global postures and contacts. Encoding scene-aware cues in the network also allows the proposed method to be optimization-free, and opens up the opportunity for real-time applications. The experiments show that the proposed network is capable of recovering accurate and physically-plausible meshes by a single forward pass and outperforms state-of-the-art methods in terms of both accuracy and speed. Code is available on our project page: https://zju3dv.github.io/sahmr/.

1. Introduction

Monocular human mesh recovery (HMR), i.e., estimating pose and shape parameters of a parametric human model from a single image, has gained significant attention in recent years. To better capture and understand human behaviors, many recent works [1–5] propose to address the problem of scene-aware HMR which involves human-scene interaction constraints when recovering human meshes, given the 3D geometry of the scene scanned by range sensors [5–7], as well as the camera pose of the input image relative to the scene, which may enable more applications in video surveillance, household robots, and motion analysis in gyms and clinics.

Most existing methods propose using scene-aware optimization to fit the human mesh into a pre-scanned scene. They optimize a parametric human model iteratively, e.g., SMPL [8], to minimize scene penetration, chamfer distance of contact regions, and the 3D-2D re-projection error. However, optimization tends to be slow at inference time and is sensitive to initialization and hyperparameters, failing to respond in low-latency applications. As illustrated in Fig. 1, an optimization-based method PROX [5] takes 18.4s to fit a human model into the scene, while the incorrect position and pose still occur.

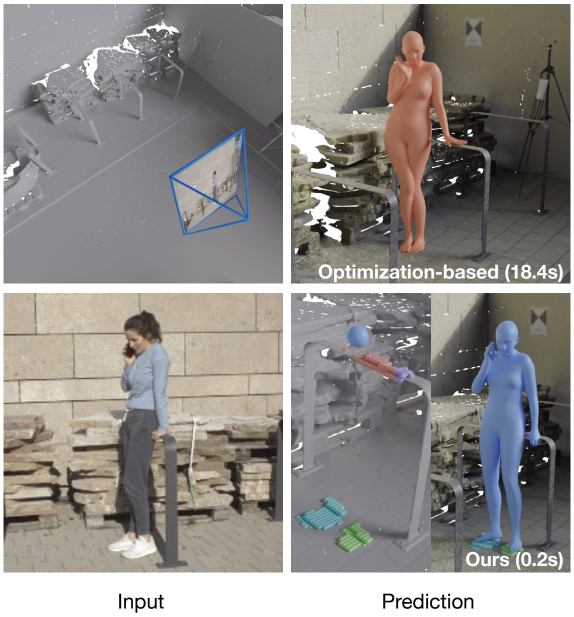

Figure 1. **Comparison between the optimization-based method and the proposed method.** Optimization-based methods typically fit a parametric human model iteratively by minimizing 2D reprojection error and scene conflicts. In contrast, the proposed method utilizes a single forward pass of the network to estimate the global position (blue ball), contact scene points (colored scene points), and a scene-aware human mesh. This design leads to improvements in both efficiency and accuracy.

Recent works [9–13] propose to recover human mesh with neural networks trained on large-scale datasets [14–17]. Specifically, the networks learn a mapping from an input image to a human mesh in the canonical coordinates. Applying these methods in the scene-aware HMR

[†] Corresponding author.

task still requires post-processing optimization, where the global translation and the human poses are refined in accordance with the given scene. However, the monocular prediction is conditioned on the input image solely, omitting the joint distribution of human pose and scene geometry, and therefore tends to suffer from depth ambiguity and occlusion. As a result, the optimization-based post-processing could be easily deteriorated by the erroneous initial poses and may even worsen the initial prediction.

In this work, we propose a Scene-Aware Human Mesh Recovery network (SA-HMR), the first learning-based approach that predicts the absolute position and mesh of a human in the scene by a single forward pass. The overall pipeline is illustrated in Fig. 2. Given the input image and scene point cloud, we first use a sparse 3D CNN to estimate dense scene contacts and absolute human position, where the scene contact estimation is treated as a point cloud labeling task, and the human position prediction is presented as a voting vector field refinement task. The predicted dense contact points are centered by the human position and passed to a scene network in the human mesh recovery step. Specifically, we enhance a pretrained monocular HMR network METRO [12] by cross-attention with the proposed scene network in parallel. In this way, SA-HMR learns a joint distribution of human pose and scene geometry, resulting in more reasonable postures, contacts, and global positions, as illustrated in Fig. 1. Learning scene-aware cues in the network also avoids scene-aware optimization as post-processing and achieves fast inference speed.

We evaluate the proposed method on the RICH [6] and PROX [5] datasets of indoor and outdoor scenes. The experimental results show that SA-HMR is not only effective in recovering absolute positions and meshes that are in accordance with the given scene, but also significantly faster than the optimization-based baselines.

In summary, we make the following contributions:

- The first optimization-free framework for scene-aware human mesh recovery from a single image and a prescanned scene.
- The cross-attention design for enhancing a pretrained HMR network with a parallel scene network, enabling joint learning on the human pose and scene geometry.
- Superior performance compared to optimization-based baselines in terms of both accuracy and speed.

2. Related Work

Monocular Human Mesh Recovery. Most existing approaches formulate the monocular HMR task as recovering the mesh of statistical human body models, e.g. SMPL and SMPL-X [8, 18, 19], where recent works can be divided into optimization-based and learning-based approaches.

The optimization-based approaches fit a parametric human model by minimizing the 3D-2D re-projection error of body joints and energy terms of heuristic priors iteratively, which are represented by SMPLify [20] that fits SMPL [8]. More recently, SMPLify-X [19] proposes a variational pose prior and fits a more expressive SMPL-X. Pose-NDF [21] proposes to represent the manifold of plausible human poses with a neural field. While optimization-based methods are general in their mathematical formulation, they are usually sensitive to hyperparameters and require much time for inference.

The learning-based methods utilize deep neural networks to predict either the parameters [9, 10, 22, 23] or the mesh vertices [11–13] of the SMPL model. HMR [9] is the pioneering work in predicting SMPL parameters, and SPIN [10] improves upon it using an optimization loop. For predicting SMPL mesh vertices, GraphCMR [11] deforms a template human mesh using graph neural network, while METRO [12] uses transformers, and [13] uses graph hierarchy to further improve the performance. However, for scene-aware HMR, the vertices of the human and scene meshes in contact are close in Euclidean space, making methods that regress parameters unsuitable due to errors that accumulate along the kinematic chains. Therefore, the proposed method is built on the works that predict mesh vertices. More details can be found in Sec. 3.

Scene-aware Human Mesh Recovery. PROX [5] is a seminal work that uses scene constraints to reduce the depth and occlusion ambiguity in monocular HMR. It achieves this by adding two energy terms of human-scene contact and penetration in the optimization process [19]. In addition, scene-aware pose generative models [2, 24] can also be used as prior terms in the scene-aware HMR task. Other recent works in this area include MoCapDeform [4], which considers deformable scene objects, LEMO [1], which uses temporal information, and HULC [3], which uses consecutive frames and dense contacts prediction on both the scene and human body. In contrast to these works, the proposed method is optimization-free and requires only a single forward pass.

In a broader topic of capturing humans in a scene-aware manner, [25–27] propose using a simulator and dynamic model, where a pre-defined agent is controlled to interact with the scene, and [28,29] consider human-object arrangement by first predicting the human and object and then performing global optimization.

Attention in Transformers. Attention is a key mechanism in Transformers [30]. It allows a set of query features to fuse the most relevant information from another set of key-value features. When query and key-value features come from the same source, it is called self-attention, otherwise cross-attention. In HMR, METRO [12] uses self-attention to re-

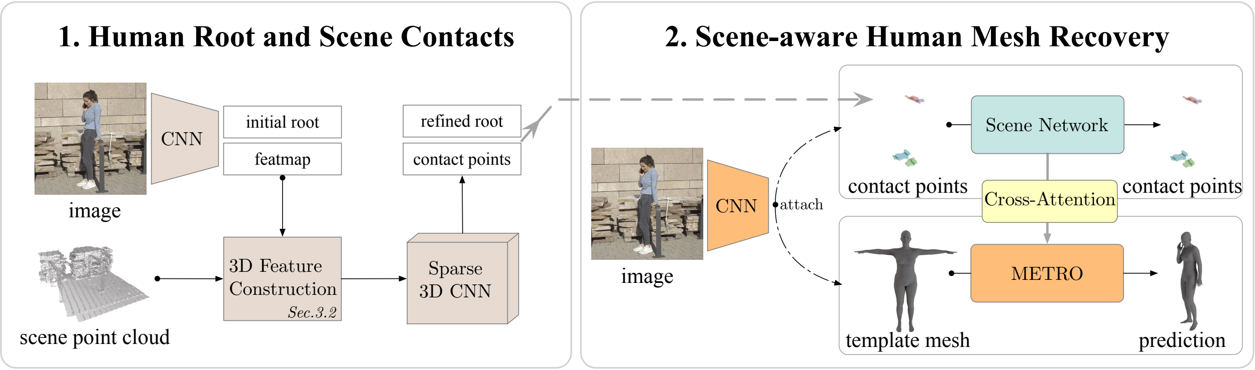

Figure 2. **Overview of the proposed SA-HMR. 1.** The human root and scene contact estimation module (Sec. 3.2) that first predicts the initial root and then refines the root with 3D scene cues using a sparse 3D CNN. The module also predicts contact labels [5] for each scene point. Please refer to Sec. 3.2 for a detailed definition of the 3D feature construction module. **2.** The scene-aware human mesh recovery module (Sec. 3.3) that enhances the pretrained METRO [12] network with a parallel scene network. The scene network takes the predicted contact scene points as input, and uses cross-attention to pass messages to the intermediate features of the METRO network.

duce occlusion ambiguity by establishing non-local feature exchange between visible and invisible parts of a template human mesh. In feature matching, SuperGlue [31] uses cross-attention to make the corresponding image features more similar. Predator [32] uses cross-attention in matching two sets of point clouds.

Inspired by feature matching, the proposed method uses cross-attention to potentially make the features of the human and scene that are in contact more similar, resulting in better contact and more reasonable postures.

3. Methods

Given a calibrated image and a pre-scanned scene point cloud (Sec. 3.1), SA-HMR first estimates the absolute human root position and scene contacts (Sec. 3.2), and then recovers the human mesh with the contact points by enhancing a pretrained METRO network (Sec. 3.3). An overview of the proposed method is presented in Fig. 2.

3.1. Preliminaries

Human Representation. We use SMPL [8] as the human representation. The SMPL is a parametric model that uses the body joint rotations, root translation, and body shape coefficients to compute the body mesh. Following [11, 12], we directly predict the SMPL mesh vertices $V = \mathbb{R}^{6890\times3}$, and use H36M [14] joint regression matrix $M \in \mathbb{R}^{14\times6890}$ to compute 3D joints $J \in \mathbb{R}^{14\times3}$ from the vertices for quantitative evaluation, $J = MV$.

Scene and Image Representation. We assume that the scene is pre-scanned with range sensors, as in RICH [6] and PROX [5], and the image is calibrated and localized in the scene, i.e. with known intrinsic and extrinsic parameters $\{(f, c_x, c_y), (R_c, t_c)\}$. Following METRO [12], we detect a squared bounding box around the target human and resize the cropped region as the input image $I \in \mathbb{R}^{224\times224\times3}$. Based on the camera parameters and the bounding box, we select scene points that fall within the visual frustum as the input scene point cloud $S \in \mathbb{R}^{N_S\times3}$.

Human-Scene Contact. Following PROX [5], we use seven regions of the SMPL mesh that are most likely to be contacted. The details are provided in the supplementary material. Using these seven categories and one for not being in contact, we perform a segmentation task on the scene point cloud.

3.2. Human Root and Scene Contacts

Given an image I bounding the human, we propose using a 2D convolutional neural network (CNN) to extract image features F and predict the initial human root r. Based on the scene points S, we unproject the image features F to 3D, resulting in $\hat{F}$. Additionally, we calculate point-wise offset vectors O that point from a voxelized scene point cloud to the initial root. By taking $\hat{F}$ and O as input, a sparse 3D CNN predicts the segmentation of scene contacts and the refined offsets, which are then converted to the refined human root r^*. An overview of this process is presented in the left column of Fig. 2.

Initial Root. We predict the initial root r=(X, Y, Z) in a 2.5D manner following SMAP [33]. Specifically, we use a CNN to predict the 2D heatmap and a normalized depth map of the root. Then, the 2D position (x, y) is obtained by applying `argmax` to the heatmap, and the corresponding normalized depth value $\tilde{Z}$ is retrieved from the depth map. Finally, using the intrinsic parameters f, c_x, c_y and image size w, the 3D root position is computed:

$$Z = \tilde{Z}\frac{f}{w} \quad (1)$$

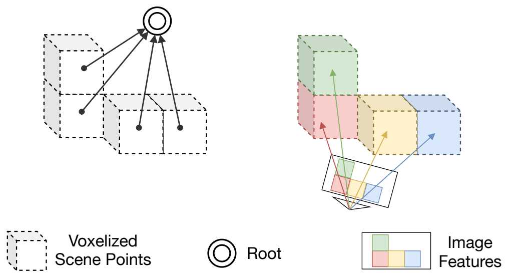

Figure 3. **3D feature construction.** We voxelize a scene point cloud to a sparse volume. The initial feature of each voxel consists of two parts, which are the offset vector pointing from the voxel center to the human root and the unprojected image features.

$$X = \frac{x - c_x}{f} \cdot Z, \quad Y = \frac{y - c_y}{f} \cdot Z \tag{2}$$

From our observation, estimating (x, y) achieves good results with a mean squared error of fewer than two pixels across datasets. However, estimating $\tilde{Z}$ is relatively challenging, possibly due to the variations in human shapes.

3D Feature Construction. Based on the initial root r and image features F, we construct the 3D features on the voxelized scene point cloud, which is illustrated in Fig. 3.

First, we select regions of interest around the initial root r in the point cloud. Specifically, we treat r as an anchor and keep points within a radius γ_1. Since Z has more uncertainty than (x, y), we sample two additional anchors along the z-axis, whose distance to r is γ_2. Next, we construct a sparse volume $\bar{S}$ by voxelizing these points with voxel size s_{vox}, where the center of each voxel is denoted as $\bar{s}_i$.

For each voxel i, the feature consists of the offset vector o_i and the unprojected image feature $\hat{f}_i$. Specifically, o_i is a vector pointing from the voxel center to the human root:

$$o_i = r - \bar{s}_i \tag{3}$$

$\hat{f}$ is computed by projecting voxel center $\bar{s}_i$ onto the image using camera parameters and bilinearly sampling the image feature map F.

Estimating Refined Root and Scene Contacts. We use sparse 3D CNN [34] to process the constructed 3D features and learn to improve the root estimation and predict the scene contacts. Specifically, the output of each voxel includes an updated offset vector o_i^*, confidence c_i, and segmentation indicating the contact category. We compute the refined root r^*:

$$r^* = \sum_i c_i \cdot (o_i^* + \bar{s}_i). \tag{4}$$

There are 8 categories of contact points, including 7 most probable regions on the body that would be contacted [5] and 1 category of not being in contact. We take the category of the highest score as the prediction for the voxel and set the dense point cloud belonging to the voxel with that category. The contact points $\hat{S}_{seg3d} \in \mathbb{R}^{\hat{N}_S \times 3}$ serve as the input for the mesh recovery module.

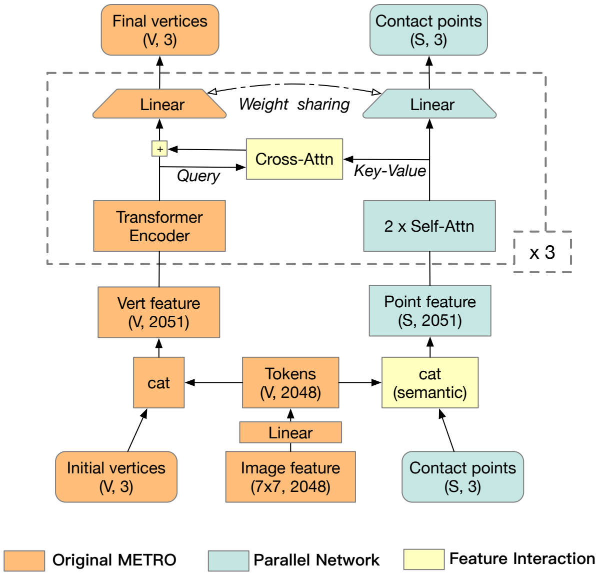

Figure 4. **Enhancing METRO with a parallel network.** The orange parts are the original METRO [12] network, where the residual connection and positional encoding are omitted for simplicity. The blue parts are the proposed parallel network which takes predicted contact scene points as input. The yellow parts indicate feature interaction between METRO and the parallel network.

3.3. Scene-aware Human Mesh Recovery

Since the training data of scene-aware human mesh recovery is limited, we build our model upon a network named METRO [12] that is pre-trained on large-scale data of monocular human mesh recovery. METRO processes feature based on the self-attention mechanism, and our approach enhances METRO by adding a parallel scene network, which provides a cross-attention-based mechanism that enables METRO to notice important scene details and achieve scene-aware human mesh recovery.

METRO consists of a CNN backbone and multiple Transformer encoders. It first extracts global CNN features, then combines the feature to the vertices of a zero posed SMPL mesh, and finally predicts a posed mesh with shape through the transformers. The part of the transformer is illustrated in the orange part of Fig. 4.

Enhancing METRO with Cross-Attention. We improve METRO with a scene network, which makes the predicted human vertices to be close to the corresponding contact scene points $\hat{S}_{seg3d}$ (Sec. 3.2).

As illustrated in Fig. 4, we add a parallel network, which has a similar architecture as the transformer of METRO,

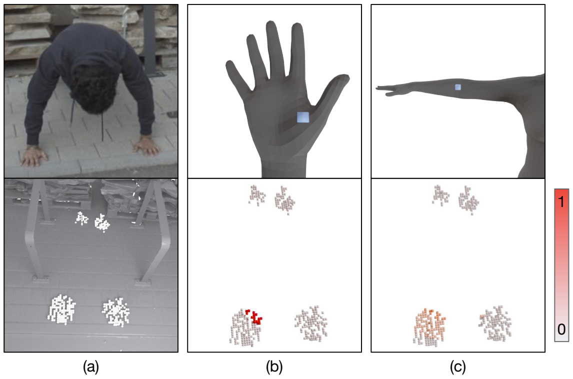

Figure 5. **Visualization of the cross-attention from a body vertex (blue point) to the predicted dense scene contacts (white points).** (a) The image and the predicted dense scene contacts. (b) The hand vertex is in contact with the scene according to the image, and its feature is similar to the nearby scene point features, enabling the final vertex prediction to be close to the corresponding scene surface. (c) The arm vertex is not in contact with the scene, where feature similarities tend to be evenly distributed. The feature similarities are normalized to the same range of 0→1.

to extract features of scene contact points and output the point positions like an autoencoder. Specifically, we first use METRO's CNN backbone to extract the image feature and map it to a set of vertex tokens by a fully-connected layer. In METRO, these tokens are directly concatenated with the positions of initial human vertices. However, the number of scene contact points is not the same as the tokens', and there is no one-to-one correspondence between them. To resolve this issue, we perform the average pooling to vertex tokens based on the contact categories defined on the SMPL mesh vertices [5], resulting in 7 tokens. Then, we append each scene contact point with the corresponding aggregated token based on the predicted category in Sec. 3.2. Intuitively, this helps the cross-attention to focus on the semantically corresponding parts. To be invariant to the global translation, the scene contact points are zero-centered by the predicted root r^*.

Motivated by recent feature matching methods [31, 32], we propose to use cross-attention to pass features from scene contact points to human vertices. The cross-attention and self-attention share the same underlying mechanism, both of which first compute the similarity of query and key, and then use the weighted sum to fuse features. When the query and key are from the same source features, it is self-attention, and otherwise is cross-attention. In practice, we use linear attention operator [35] to improve efficiency. A visualization of cross attention over human vertices and scene points is in Fig. 5. The detailed network architecture is provided in the supplementary material.

Note that, we use a weight-sharing regressor layer with METRO, which regresses from point-wise features to point position (x, y, z). Therefore, to get a similar (x, y, z), the input features of this layer should also be similar. This strategy implicitly aligns the features of human vertices and scene points when their final predictions are near in 3D space, thus facilitating the cross-attention to find correspondences between human vertices and scene points.

3.4. Training Loss

Root and Contact. The loss function L_{RC} for the root and contact estimation is defined as:

$$L_{\text{RC}} = L_{\text{R2D}} + w_{\text{RZ}} \cdot L_{\text{RZ}} + L_{\text{ROV}} + L_{\text{R3D}} + L_{\text{C}} \tag{5}$$

where L_{R2D} is the MSE loss on the root heatmap; L_{RZ}, L_{ROV}, and L_{R3D} are the L1 losses on the relative depth, offset vectors, and the 3D root, respectively; L_{C} is the cross-entropy loss for contact categories of voxel points, where we additionally train an auxiliary task of 2D contact segmentation similar to the voxel points.

Human Mesh Recovery. The loss function L_{HMR} for the mesh recovery is defined as:

$$L_{\text{HMR}} = L_{\text{V}} + L_{\text{J}} + L_{\text{CP}} + L_{\text{GV}} \tag{6}$$

where L_{V}, L_{J}, L_{CP}, and L_{GV} are the L1 losses on the translation-aligned human vertices, human joints, reconstructed contact points, and global human vertices, respectively. More details are in the supplementary material.

3.5. Implementation Details

We train two modules separately. For the root and contact module, the CNN is HRNet-stride-4 [36] with METRO initialization, the sparse 3D CNN is SPVCNN [34, 37] with random initialization. We use linear layers to align intermediate feature dimensions. γ_1 is 1.25 m, γ_2 is 0.5 m, s_{vox} is 5^3 cm^3, and w_{RZ} is 10. The contact threshold is 7 cm. We flip images for augmentation. The module is trained with an initial learning rate of 3.75e-5 and a batch size of 24. It converges after 30 epochs of training on one V100 GPU. For the mesh recovery module, the METRO network is initialized as pretrained and the scene network is randomly initialized. The initial learning rate is 7.5e-6 and the batch size is 24. It converges after 30 epochs of training.

4. Experiments

4.1. Datasets

We train and evaluate the proposed method on RICH [6] and PROX [5] datasets separately.

RICH [6] captures multi-view video sequences in 6 outdoor and 2 indoor scenes. It provides images, reconstructed bodies, scene scans, and human-scene contact labels annotated on SMPL vertices. We skip frames including multiple

Method	Learning-based	Optimization	Scene-aware	G-MPJPE↓	G-MPVE↓	PenE↓	ConFE↓	MPJPE↓	MPVE↓
Dataset GT [6]		✓	✓	/	/	9.8	10.8	/	/
SMPLify-X [19]		✓		482.0	483.7	35.7	43.4	166.9	177.6
PROX [5]		✓	✓	**390.1**	**397.2**	15.5	24.1	164.1	175.8
POSA [2]		✓	✓	427.8	434.0	21.1	27.0	177.2	188.4
PLACE [38]		✓	✓	395.9	403.0	16.1	24.8	163.8	175.4
METRO [12]† + SA-Opt [6,29]	✓	✓	✓	563.1	561.3	**7.4**	**14.8**	**102.7**	**112.8**
METRO [12]	✓			678.6	679.4	52.2	56.9	129.6	134.5
METRO [12]†	✓			511.7	509.7	33.6	37.6	98.8	107.9
SA-HMR	✓		✓	**264.6**	**272.7**	**14.9**	**19.0**	**93.9**	**103.0**

Table 1. **Evaluation on the RICH [6] dataset.** METRO† indicates that the model is finetuned on the dataset. SA-Opt indicates scene-aware optimization, with contact estimation from BSTRO [6] and loss formulation from PROX [5] and PHOSA [29]. The proposed SA-HMR achieves the overall best results and is significantly faster than the methods that require optimization.

Method	G-MPJPE↓	G-MPVE↓	PenE↓	MPJPE↓	MPVE↓
Dataset GT [5]	/	/	9.6	/	/
SMPLify-X [19]	216.0	222.6	49.3	**100.7**	**112.8**
PROX [5]	172.0	178.5	**10.7**	101.1	114.0
POSA [2]	172.3	180.9	16.6	108.5	119.4
PLACE [38]	**168.1**	**176.7**	12.3	100.8	113.7
METRO [12]	283.2	277.7	62.4	137.0	147.2
METRO [12]†	265.6	262.7	67.5	117.1	128.5
SA-HMR	**150.4**	**160.0**	**26.9**	**111.1**	**122.5**

Table 2. **Evaluation on the PROX [5] dataset.** The proposed method achieves the best performance in global metrics.

subjects, remove the first 45 frames of each video to avoid static starting pose, and skip frames where the subjects' 2D bounding boxes are not inside the images. Then we downsample the train / val / test splits to 2 / 1 / 1 fps, resulting in 15360 / 3823 / 3316 frames.

PROX [5] captures monocular RGBD videos in 12 indoor scenes. We use RGB images and scene scans. It is a challenging dataset where severe occlusions exist in most frames. We use the qualitative set for training and the quantitative set for testing. In order to get better training annotations, we additionally use HuMoR [26] which utilizes motion prior and optimizes a sequence of frames. Then, we manually remove the failed frames that are not consistent with the images and scenes. Finally, the training split contains 4852 frames.

4.2. Metrics

We evaluate quantitatively in terms of human mesh recovery and human-scene contact.

Human Mesh Recovery. We report the Global Mean-Per-Joint-Position-Error (**G-MPJPE**) and Global Mean-Per-Vertex-Error (**G-MPVE**) in the scene coordinate system, which calculates the average L2 distances between predicted and ground truth joints/vertices. Additionally, we report the translation-aligned metrics **MPJPE** and **MPVE**.

Human-scene Contact. We report the Penetration Error (**PenE**) and Contact Failure Error (**ConFE**). PenE measures the total distance that SMPL vertices penetrate the scene mesh:

$$\text{PenE} = \sum_{i=1}^{V} \mathbb{1}_{x<0}[sdf(v_i, \mathcal{S})] \cdot |sdf(v_i, \mathcal{S})|, \tag{7}$$

where V is the number of SMPL vertices, $sdf\ (v_i, \mathcal{S})$ is the signed distance from vertex v to scene $\mathcal{S}$, and $\mathbb{1}_{x<0}[\cdot]$ is an indicator function that returns 1 when the condition is met, and 0 otherwise. ConFE measures contact quality when the ground truth contact label is available:

$$\begin{aligned}\text{ConFE} = \sum_{i=1}^{V} (&C_{gt}(v_i) \cdot |sdf(v_i, \mathcal{S})| \\ &+ (1 - C_{gt}(v_i)) \cdot \mathbb{1}_{x<0}[sdf(v_i, \mathcal{S})] \cdot |sdf(v_i, \mathcal{S})|),\end{aligned} \tag{8}$$

where $C_{gt}(v)$ equals 1 if v is labeled as in contact, and 0 otherwise. In order to obtain a good result of ConFE, the body vertices in contact should be close to the scene surface, while vertices not in contact should avoid penetration.

4.3. Main Results

Baselines. Optimization: SMPLify-X [19] uses RGB only, PROX [5] extends it with losses of human-scene contact and penetration, POSA [2] and PLACE [38] extend PROX with scene-aware pose priors, and METRO [12]†+SA-Opt stands for post-processing a finetuned METRO with scene-aware optimization, which will be explained later. Learning-based: METRO predicts canonical human mesh vertices and a weak-perspective camera. We solve the transformation from human to camera coordinate by minimizing joint re-projection error with a PnP solver [39]. The METRO† is finetuned with the same training protocol as SA-HMR.

Since METRO does not consider scenes, we additionally optimize the global pose and scale by minimizing

Dataset	Initial RtErr	Refined RtErr	Final RtErr
RICH	510.8	284.7	**246.5**
PROX	364.2	132.3	**111.8**

Table 3. **Ablation study of root estimation.** The human root position errors (RtErr) in mm are reported.

scene-aware losses, including re-projection error, human-scene penetration, contact distance [6], and ordinal depth error [29], following the key ideas of PROX and PHOSA [29], which is denoted as METRO†+SA-Opt.

Results. For the RICH dataset, Tab. 1 shows that SA-HMR notably outperforms other baselines in terms of the G-MPJPE and G-MPVE by a significant margin, demonstrating the effectiveness of the proposed pipeline. The joint learning on both image and scene geometry also improves the metrics of local pose and human-scene contact. We use open-sourced code for SMPLify-X and PROX, and implement POSA and PLACE upon PROX. Optimization methods approximately cost 18s for a single fitting, which is much slower compared to 0.2s of SA-HMR. We provide qualitative results comparing to baselines in Fig. 6.

For the PROX dataset, SA-HMR outperforms all baselines in terms of global accuracy as illustrated in Tab. 2. Since the pseudo ground truth of the training set is still not of high quality, as well as a domain gap exists in the test set where the subject wears a MoCap suit, our method falls a little behind in local accuracy and scene penetration. And we do not report ConFE, since the ground truth contact label is not available. Nevertheless, the clear improvement compared to the most relevant model METRO† has demonstrated the effectiveness of the proposed method.

We also observe that while considering the scene geometry is critical for estimating the global position and improving physical plausibility, it may not fully resolve the ambiguity of the local pose, where multiple physically plausible solutions may still exist. For example, the RGB-only method SMPLify-X and the scene-aware method PROX perform similarly in MPJPE and MPVE.

4.4. Ablation Study

Root and Contact Module. Tab. 3 shows that the predicted human root position is improved progressively by the refinement and scene-aware HMR modules, where the initial prediction [33] is improved 44% / 52% in RICH, and 64% / 69% in PROX. The offset representation helps to improve erroneous initial root prediction that is not consistent with the scene surface. For scene contact estimation, the precision / recall is 0.57 / 0.53 on RICH, and 0.45 / 0.24 on PROX. We observe that the contacts are difficult to predict, which aligns with the conclusion of a recent work HULC [3]. More visualizations are presented in Fig. 7.

Method	G-MPJPE	G-MPVE	MPJPE	MPVE	CErr↓
w/o parallel	304.8	312.9	98.5	108.9	10.2
Ours	**264.6**	**272.7**	**93.9**	**103.0**	**8.9**

Table 4. **Ablation study of the parallel network** on the RICH dataset. The compared variant fuses features of contact points at the early stage.

Root	Contact	MPJPE	MPVE	CErr↓
/	/	98.8	107.9	10.5
Est.	Est.	93.9	103.0	8.9
GT	Est.	89.2	98.1	7.9
Est.	GT	90.4	99.2	8.3
GT	GT	**76.7**	**84.6**	**5.2**

Table 5. **Ablation study of the scene-aware mesh recovery module.** We validate the upper bound of the proposed method on the RICH dataset.

Mesh Recovery Module. As shown in Tab. 4, the parallel network that uses cross-attention outperforms a variant that fuses the pointnet features of the contact points to the METRO network in the early stage. The CErr indicates the error of contact mesh vertices in the translation-aligned coordinates. In Tab. 5, we validate the upper bound of the mesh recovery module. We replace the intermediate estimation of root and contact, and find a steady improvement in the pose and shape accuracy.

Running Time. SA-HMR runs at 170 ms with a peak memory cost of 1852 MB for a 224×224 image and a scene point cloud of 2 cm resolution on a V100 GPU. Specifically, the root and contact module takes 92 ms (CNN 50 ms, SPVCNN 42 ms), the mesh recovery module takes 75 ms (CNN 49 ms, Transformer 26 ms), and the intermediate processing takes 3 ms.

5. Conclusion

This work addresses the challenge of estimating the human mesh from an RGB image with the consideration of the scene geometry. Our key idea is to inject 3D scene cues into a monocular human mesh recovery network to recover the absolute human pose and shape in the scene. To this end, our approach first predicts the 3D human location and dense human-scene contacts with a sparse 3D CNN. We then develop a transformer to extract features from contact scene points and feed them into the pose estimation network using cross-attention. Experiments demonstrate that our approach achieves state-of-the-art performance on the RICH and PROX datasets.

Acknowledgement. This work was supported by the Key Research Project of Zhejiang Lab (No. K2022PG1BB01), NSFC (No. 62172364), and Information Technology Center and State Key Lab of CAD&CG, Zhejiang University.

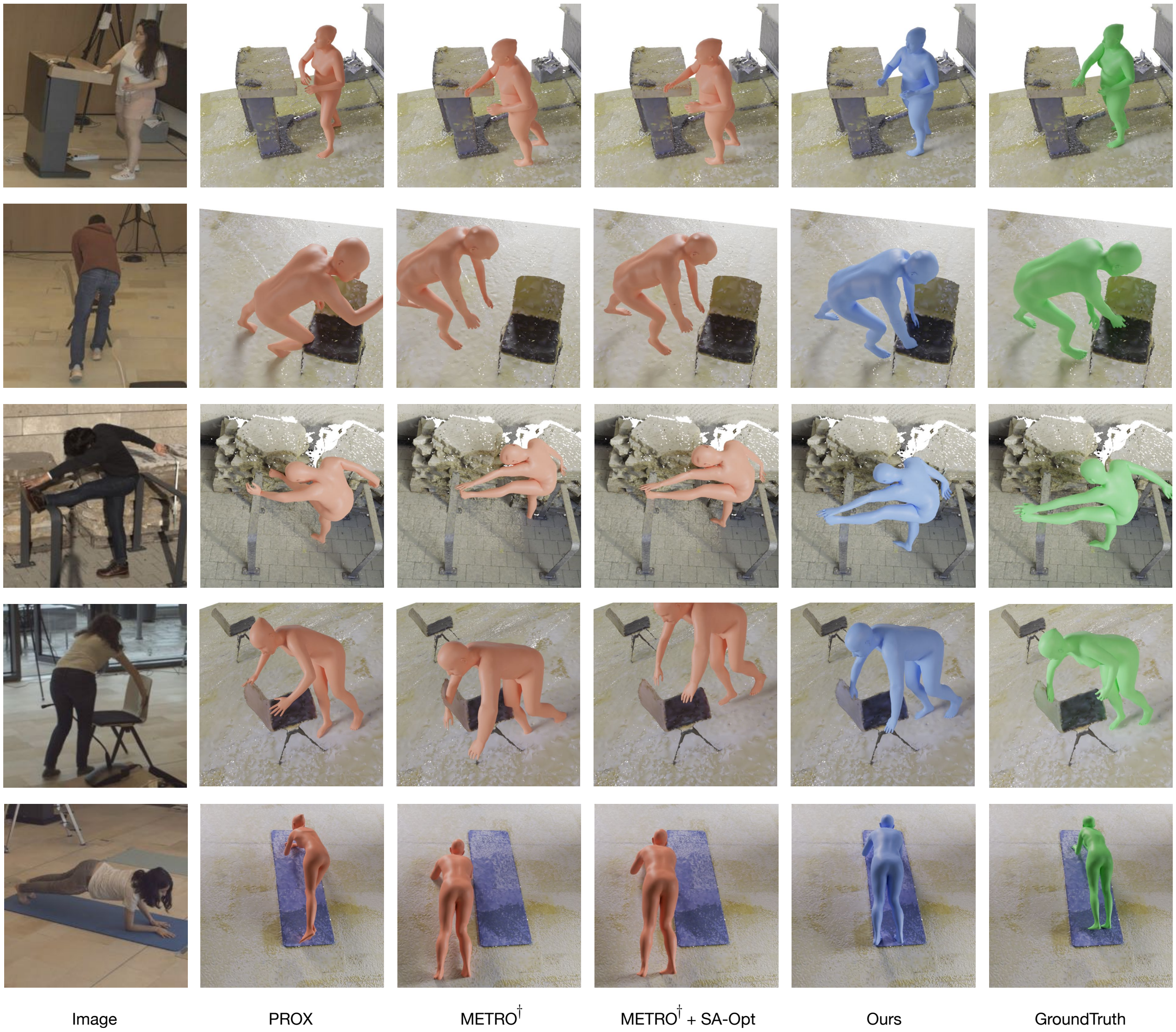

Figure 6. **Qualitative results on the RICH [6] dataset.** We compare the proposed method to PROX [5], finetuned METRO [12], finetuned METRO with scene-aware optimization, and ground truth. The leftmost column shows the input images. The proposed method recovers the global positions and human-scene contact more accurately because of the 3D learning on human root refinement and dense scene contact labeling tasks.

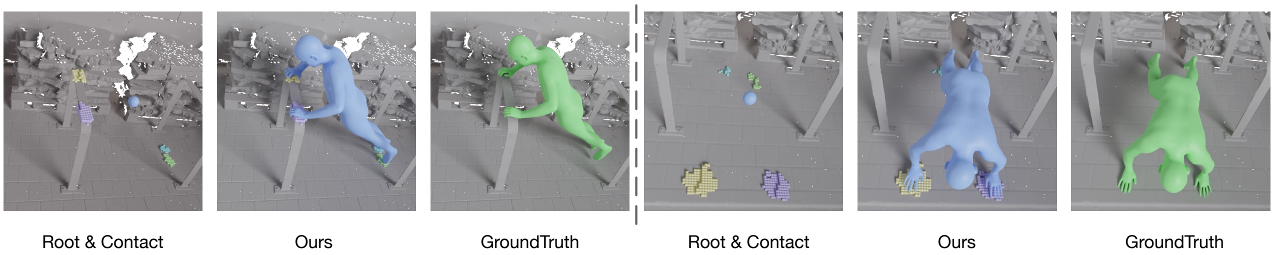

Figure 7. **Qualitative visualization of the estimated root locations and dense scene contacts.** In both examples, the estimated contact points provide accurate position and scene structure for the following step of mesh recovery. The reconstructed human mesh is in good contact with the corresponding scene regions.

References

[1] Siwei Zhang, Yan Zhang, Federica Bogo, Marc Pollefeys, and Siyu Tang. Learning motion priors for 4d human body capture in 3d scenes. In *ICCV*, 2021. 1, 2

[2] Mohamed Hassan, Partha Ghosh, Joachim Tesch, Dimitrios Tzionas, and Michael J. Black. Populating 3d scenes by learning human-scene interaction. In *CVPR*, 2021. 1, 2, 6

[3] Soshi Shimada, Vladislav Golyanik, Zhi Li, Patrick Pérez, Weipeng Xu, and Christian Theobalt. HULC: 3d human motion capture with pose manifold sampling and dense contact guidance. In *ECCV*, 2022. 1, 2, 7

[4] Zhi Li, Soshi Shimada, Bernt Schiele, Christian Theobalt, and Vladislav Golyanik. MoCapDeform: Monocular 3d human motion capture in deformable scenes. In *3DV*, 2022. 1, 2

[5] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. Resolving 3d human pose ambiguities with 3d scene constraints. In *ICCV*, 2019. 1, 2, 3, 4, 5, 6, 8

[6] Chun-Hao P. Huang, Hongwei Yi, Markus Höschle, Matvey Safroshkin, Tsvetelina Alexiadis, Senya Polikovsky, Daniel Scharstein, and Michael J. Black. Capturing and inferring dense full-body human-scene contact. In *CVPR*, 2022. 1, 2, 3, 5, 6, 7, 8

[7] Siwei Zhang, Qianli Ma, Yan Zhang, Zhiyin Qian, Marc Pollefeys, Federica Bogo, and Siyu Tang. Egobody: Human body shape, motion and social interactions from head-mounted devices. In *ECCV*, 2022. 1

[8] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *TOG*, 2015. 1, 2, 3

[9] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 1, 2

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

[11] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, 2019. 1, 2, 3

[12] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, 2021. 1, 2, 3, 4, 6, 8

[13] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *ICCV*, 2021. 1, 2

[14] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 2013. 1, 3

[15] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *ECCV*, 2018. 1

[16] 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*, 2014. 1

[17] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014. 1

[18] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *TOG*, 2017. 2

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

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

[21] Garvita Tiwari, Dimitrije Antić, Jan Eric Lenssen, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Pose-NDF: Modeling human pose manifolds with neural distance fields. In *ECCV*, 2022. 2

[22] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human model fitting towards in-the-wild 3d human pose estimation. In *3DV*, 2021. 2

[23] Muhammed Kocabas, Chun-Hao P Huang, Joachim Tesch, Lea Müller, Otmar Hilliges, and Michael J Black. Spec: Seeing people in the wild with an estimated camera. In *ICCV*, 2021. 2

[24] Yan Zhang, Mohamed Hassan, Heiko Neumann, Michael J. Black, and Siyu Tang. Generating 3d people in scenes without people. In *CVPR*, 2020. 2

[25] Soshi Shimada, Vladislav Golyanik, Weipeng Xu, and Christian Theobalt. Physcap: Physically plausible monocular 3d motion capture in real time. *TOG*, 2020. 2

[26] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J Guibas. Humor: 3d human motion model for robust pose estimation. In *ICCV*, 2021. 2, 6

[27] Zhengyi Luo, Shun Iwase, Ye Yuan, and Kris Kitani. Embodied scene-aware human pose estimation. In *NeurIPS*, 2022. 2

[28] Zhenzhen Weng and Serena Yeung. Holistic 3d human and scene mesh estimation from single view images. In *CVPR*, 2021. 2

[29] Jason Y Zhang, Sam Pepose, Hanbyul Joo, Deva Ramanan, Jitendra Malik, and Angjoo Kanazawa. Perceiving 3d human-object spatial arrangements from a single image in the wild. In *ECCV*, 2020. 2, 6, 7

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[31] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 3, 5

[32] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *CVPR*, 2021. 3, 5

[33] Jianan Zhen, Qi Fang, Jiaming Sun, Wentao Liu, Wei Jiang, Hujun Bao, and Xiaowei Zhou. SMAP: Single-shot multi-person absolute 3d pose estimation. In *ECCV*, 2020. 3, 7

[34] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *ECCV*, 2020. 4, 5

[35] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *ICML*, 2020. 5

[36] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 5

[37] Haotian Tang, Zhijian Liu, Xiuyu Li, Yujun Lin, and Song Han. TorchSparse: Efficient Point Cloud Inference Engine. In *MLSys*, 2022. 5

[38] Siwei Zhang, Yan Zhang, Qianli Ma, Michael J. Black, and Siyu Tang. PLACE: Proximity learning of articulation and contact in 3d environments. In *3DV*, 2020. 6

[39] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 6