I2L-MeshNet: Image-to-Lixel Prediction Network for Accurate 3D Human Pose and Mesh Estimation from a Single RGB Image

Gyeongsik Moon and Kyoung Mu Lee

ECE & ASRI, Seoul National University, Korea
{mks0601,kyoungmu}@snu.ac.kr

Abstract. Most of the previous image-based 3D human pose and mesh estimation methods estimate parameters of the human mesh model from an input image. However, directly regressing the parameters from the input image is a highly non-linear mapping because it breaks the spatial relationship between pixels in the input image. In addition, it cannot model the prediction uncertainty, which can make training harder. To resolve the above issues, we propose I2L-MeshNet, an image-to-lixel (line+pixel) prediction network. The proposed I2L-MeshNet predicts the per-lixel likelihood on 1D heatmaps for each mesh vertex coordinate instead of directly regressing the parameters. Our lixel-based 1D heatmap preserves the spatial relationship in the input image and models the prediction uncertainty. We demonstrate the benefit of the image-to-lixel prediction and show that the proposed I2L-MeshNet outperforms previous methods. The code is publicly available [1].

1 Introduction

3D human pose and mesh estimation aims to simultaneously recover 3D semantic human joint and 3D human mesh vertex locations. This is a very challenging task because of complicated human articulation and 2D-to-3D ambiguity. It can be used in many applications such as virtual/augmented reality and human action recognition.

SMPL [25] and MANO [39] are the most widely used parametric human body and hand mesh models, respectively, which can represent various human poses and identities. They produce 3D human joint and mesh coordinates from pose and identity parameters. Recent deep convolutional neural network (CNN)-based studies [18, 21, 37] for the 3D human pose and mesh estimation are based on the model-based approach, which trains a network to estimate SMPL/MANO parameters from an input image. On the other hand, there have been few methods based on model-free approach [9, 22], which estimates mesh vertex coordinates directly. They obtain the 3D pose by multiplying a joint regression matrix, included in the human mesh model, to the estimated mesh.

[1] `https://github.com/mks0601/I2L-MeshNet_RELEASE`

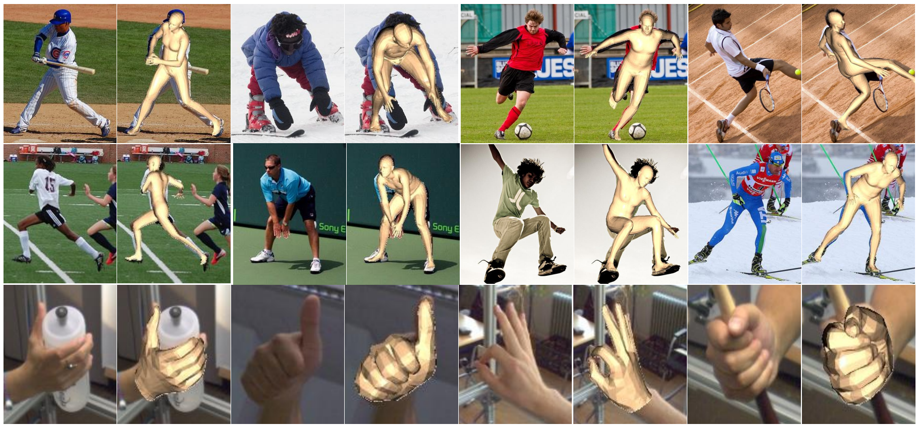


Fig. 1. Qualitative results of the proposed I2L-MeshNet on MSCOCO [24] and FreiHAND [8] datasets.

Although the recent deep CNN-based methods perform impressive, when estimating the target (*i.e.*, SMPL/MANO parameters or mesh vertex coordinates), all of the previous 3D human pose and mesh estimation works break the spatial relationship among pixels in the input image because of the fully-connected layers at the output stage. In addition, their target representations cannot model the uncertainty of the prediction. The above limitations can make training harder, and as a result, reduce the test accuracy as addressed in [29,42]. To address the limitations, recent state-of-the-art 3D human pose estimation methods [29, 30, 41], which localize 3D human joint coordinates without mesh vertex coordinates, utilize the *heatmap* as the target representation of their networks. Each value of one heatmap represents the likelihood of the existence of a human joint at the corresponding pixel positions of the input image and discretized depth value. Therefore, it preserves the spatial relationship between pixels in the input image and models the prediction uncertainty.

Inspired by the recent state-of-the-art heatmap-based 3D human pose estimation methods, we propose I2L-MeshNet, image-to-lixel prediction network that naturally extends heatmap-based 3D human pose to heatmap-based 3D human pose and mesh. Likewise voxel (volume+pixel) is defined as a quantized cell in three-dimensional space, we define *lixel (line+pixel)* as a quantized cell in one-dimensional space. Our I2L-MeshNet estimates per-lixel likelihood on 1D heatmaps for each mesh vertex coordinates, therefore it is based on the model-free approach. The previous state-of-the-art heatmap-based 3D human pose estimation methods predict 3D heatmap of each human joint. Unlike the number of human joints, which is around 20, the number of mesh vertex is much larger (*e.g.*, 6980 for SMPL and 776 for MANO). As a result, predicting 3D heatmaps of all mesh vertices becomes computationally infeasible, which is beyond the limit of modern GPU memory. In contrast, the proposed lixel-based 1D heatmap has an efficient memory complexity, which has a linear relationship with the heatmap resolution. Thus, it allows our system to predict heatmaps with sufficient resolution, which is essential for dense mesh vertex localization.

For more accurate 3D human pose and mesh estimation, we design the I2L-MeshNet as a cascaded network architecture, which consists of PoseNet and MeshNet. The PoseNet predicts the lixel-based 1D heatmaps of each 3D human joint coordinate. Then, the MeshNet utilizes the output of the PoseNet as an additional input along with the image feature to predict the lixel-based 1D heatmaps of each 3D human mesh vertex coordinate. As the locations of the human joints provide coarse but important information about the human mesh vertex locations, utilizing it for 3D mesh estimation is natural and can increase accuracy substantially.

Our I2L-MeshNet outperforms previous 3D human pose and mesh estimation methods on various 3D human pose and mesh benchmark datasets. Figure 1 shows 3D human body and hand mesh estimation results on publicly available datasets.

Our contributions can be summarized as follows.

- We propose I2L-MeshNet, a novel image-to-lixel prediction network for 3D human pose and mesh estimation from a single RGB image. Our system predicts lixel-based 1D heatmap that preserves the spatial relationship in the input image and models the uncertainty of the prediction.
- Our efficient lixel-based 1D heatmap allows our system to predict heatmaps with sufficient resolution, which is essential for dense mesh vertex localization.
- We show that our I2L-MeshNet outperforms previous state-of-the-art methods on various 3D human pose and mesh datasets.

2 Related works

3D human body and hand pose and mesh estimation. Most of the current 3D human pose and mesh estimation methods are based on the model-based approach, which predict parameters of pre-defined human body and hand mesh models (*i.e.*, SMPL and MANO, respectively). The model-based methods can be trained only from groundtruth human joint coordinates without mesh vertex coordinates because the model parameters are embedded in low dimensional space. Early model-based methods [4] iteratively fit the SMPL parameters to estimated 2D human joint locations. More recent model-based methods regress the body model parameters from an input image using CNN. Kanazawa et al. [18] proposed an end-to-end trainable human mesh recovery (HMR) system that uses the adversarial loss to make their output human shape is anatomically plausible. Pavlakos et al. [37] used 2D joint heatmaps and silhouette as cues for predicting accurate SMPL parameters. Omran et al. [32] proposed a similar system, which exploits human part segmentation as a cue for regressing SMPL parameters. Xu et al. [45] used differentiable rendering to supervise human mesh in the 2D

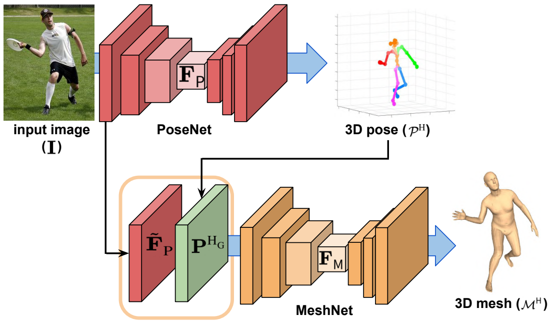

input image
$(\mathbf{I})$
PoseNet
$\mathbf{F}_{\mathrm{P}}$
3D pose ($\mathcal{P}^{\mathrm{H}}$)
$\tilde{\mathbf{F}}_{\mathrm{P}}$
$\mathbf{P}^{\mathrm{H}_\mathrm{G}}$
$\mathbf{F}_{\mathrm{M}}$
MeshNet
3D mesh ($\mathcal{M}^{\mathrm{H}}$)

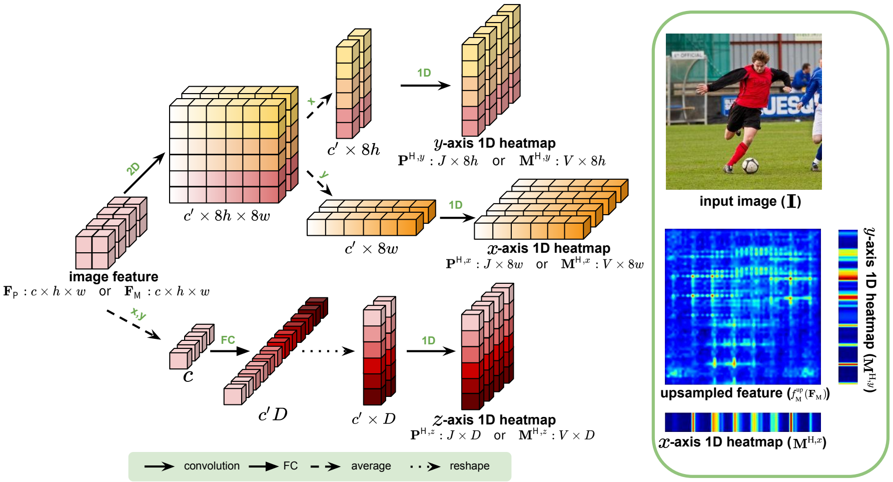
Fig. 2. Overall pipeline of the proposed I2L-MeshNet.

image space. Pavlakos et al. [35] proposed a system that uses multi-view color consistency to supervise a network using multi-view geometry. Baek et al. [3] trained their network to estimate the MANO parameters using a differentiable renderer. Boukhayma et al. [5] trained their network that takes a single RGB image and estimates MANO parameters by minimizing the distance of the estimated hand joint locations and groundtruth. Kolotouros et al. [21] introduced a self-improving system consists of SMPL parameter regressor and iterative fitting framework [4].

On the other hand, the model-free approach estimates the mesh vertex coordinates directly instead of regressing the model parameters. Due to the recent advancement of the iterative human body and hand model fitting frameworks [4, 8, 34], pseudo-groundtruth mesh vertex annotation on large-scale datasets [8, 14, 24, 26] became available. Those datasets with mesh vertex annotation motivated several model-free methods that require mesh supervision. Kolotouros et al. [22] designed a graph convolutional human mesh regression system. Their graph convolutional network takes a template human mesh in a rest pose as input and outputs mesh vertex coordinates using image feature from ResNet [12]. Ge et al. [9] proposed a graph convolution-based network which directly estimates vertices of hand mesh. Recently, Choi et al. [7] proposed a graph convolutional network that recovers 3D human pose and mesh from a 2D human pose.

Unlike all the above model-based and model-free 3D human pose and mesh estimation methods, the proposed I2L-MeshNet outputs 3D human pose and mesh by preserving the spatial relationship between pixels in the input image and modeling uncertainty of the prediction. Those two main advantageous are brought by designing the target of our network to the lixel-based 1D heatmap. This can make training much stable, and the system achieves much lower test error.

Heatmap-based 3D human pose estimation. Most of the recent state-of-the-art 2D and 3D human pose estimation methods use heatmap as a prediction target, which preserves the spatial relationship in the input image and models the uncertainty of the prediction. Tompson et al. [42] proposed to estimate the


2D
x
y
x,y
FC
1D
1D
1D
$c' \times 8h \times 8w$
$c' \times 8h$
$c' \times 8w$
c
$c'D$
$c' \times D$
image feature
$\mathbf{F}_{\mathrm{P}} : c \times h \times w$ or $\mathbf{F}_{\mathrm{M}} : c \times h \times w$
y-axis 1D heatmap
$\mathbf{P}^{\mathrm{H},y} : J \times 8h$ or $\mathbf{M}^{\mathrm{H},y} : V \times 8h$
x-axis 1D heatmap
$\mathbf{P}^{\mathrm{H},x} : J \times 8w$ or $\mathbf{M}^{\mathrm{H},x} : V \times 8w$
z-axis 1D heatmap
$\mathbf{P}^{\mathrm{H},z} : J \times D$ or $\mathbf{M}^{\mathrm{H},z} : V \times D$
convolution
FC
average
reshape
input image ($\mathbf{I}$)
upsampled feature ($f^{\mathrm{up}}_{\mathrm{M}}(\mathbf{F}_{\mathrm{M}})$)
y-axis 1D heatmap ($\mathbf{M}^{\mathrm{H},y}$)
x-axis 1D heatmap ($\mathbf{M}^{\mathrm{H},x}$)

(a) network architecture to estimate lixel-based 1D heatmaps (b) visualized feature map and 1D heatmaps

Fig. 3. Network architecture to predict lixel-based 1D heatmaps and visualized examples of feature maps and the 1D heatmaps.

Gaussian heatmap instead of directly regressing coordinates of human body joints. Their heatmap representation helps their model to perform 2D human pose estimation more accurate and motivated many heatmap-based 2D human pose methods [6,31,44]. Pavlakos et al. [36] and Moon et al. [29] firstly proposed to use 3D heatmaps as a prediction target for 3D human body pose and 3D hand pose estimation, respectively. Especially, Moon et al. [29] demonstrated that under the same setting, changing prediction target from coordinates to heatmap significantly improves the 3D hand pose accuracy while requires much less amount of the learnable parameters. Recently, Moon et al. [30] achieved significantly better 3D multi-person pose estimation accuracy using 3D heatmap compared with previous coordinate regression-based methods [38].

3 I2L-MeshNet

Figure 2 shows the overall pipeline of the proposed I2L-MeshNet. I2L-MeshNet consists of PoseNet and MeshNet, which will be described in the following subsections.

3.1 PoseNet

The PoseNet estimates three lixel-based 1D heatmaps of all human joints $\mathcal{P}^{\mathrm{H}} = \{\mathbf{P}^{\mathrm{H},x}, \mathbf{P}^{\mathrm{H},y}, \mathbf{P}^{\mathrm{H},z}\}$ from the input image $\mathbf{I}$. $\mathbf{P}^{\mathrm{H},x}$ and $\mathbf{P}^{\mathrm{H},y}$ are defined in x- and y-axis of the image space, while $\mathbf{P}^{\mathrm{H},z}$ is defined in root joint (*i.e.*, pelvis or wrist)-relative depth space. For this, PoseNet extracts image feature $\mathbf{F}_{\mathrm{P}} \in \mathbb{R}^{c \times h \times w}$ from the input image by ResNet [12]. Then, three upsampling modules increases the spatial size of $\mathbf{F}_{\mathrm{P}}$ by 8 times, while changing channel dimension from $c = 2048$ to $c' = 256$. Each upsampling module consists of deconvolutional layer, 2D batch normalization layer [13], and ReLU function. The upsampled features are used to compute lixel-based 1D human pose heatmaps, as illustrated in Figure 3 (a). We obtain x- and y-axis 1D human pose heatmaps as follows:

$$\mathbf{P}^{\mathrm{H},x} = f_{\mathrm{P}}^{\mathrm{1D},x}(\mathrm{avg}^{y}(f_{\mathrm{P}}^{\mathrm{up}}(\mathbf{F}_{\mathrm{P}}))) \quad \text{and} \quad \mathbf{P}^{\mathrm{H},y} = f_{\mathrm{P}}^{\mathrm{1D},y}(\mathrm{avg}^{x}(f_{\mathrm{P}}^{\mathrm{up}}(\mathbf{F}_{\mathrm{P}}))), \tag{1}$$

where $f_{\mathrm{P}}^{\mathrm{up}}$ denotes the three upsampling modules of the PoseNet. avg^{i} and $f_{\mathrm{P}}^{\mathrm{1D},i}$ denote i-axis marginalization by averaging and a 1-by-1 1D convolution that changes channel dimension from c' to J for i-axis 1D human pose heatmap estimation, respectively.

We obtain z-axis 1D human pose heatmaps as follows:

$$\mathbf{P}^{\mathrm{H},z} = f_{\mathrm{P}}^{\mathrm{1D},z}(\psi(f_{\mathrm{P}}(\mathrm{avg}^{x,y}(\mathbf{F}_{\mathrm{P}})))), \tag{2}$$

where f_{P} and $\psi \colon \mathbb{R}^{c'D} \rightarrow \mathbb{R}^{c' \times D}$ denote a building block and and reshape function, respectively. The building block consists of a fully-connected layer, 1D batch normalization layer, and ReLU function, and it changes the activation size from c to $c'D$. D denotes depth discretization size and is equal to $8h = 8w$. We convert the discretized heatmaps of $\mathcal{P}^{\mathrm{H}}$ to continuous coordinates $\mathbf{P}^{\mathrm{C}} = [\mathbf{p}^{\mathrm{C},x}, \mathbf{p}^{\mathrm{C},y}, \mathbf{p}^{\mathrm{C},z}] \in \mathbb{R}^{J \times 3}$ by soft-argmax [41].

3.2 MeshNet

The MeshNet has a similar network architecture with that of the PoseNet. Instead of taking the input image $\mathbf{I}$, MeshNet takes a pre-computed image feature from the PoseNet $\tilde{\mathbf{F}}_{\mathrm{P}}$ and 3D Gaussian heatmap $\mathbf{P}^{\mathrm{H}_{\mathrm{G}}} \in \mathbb{R}^{J \times D \times 8h \times 8w}$. $\tilde{\mathbf{F}}_{\mathrm{P}}$ is the input of the first residual block of the PoseNet whose spatial dimension is $8h \times 8w$. $\mathbf{P}^{\mathrm{H}_{\mathrm{G}}}$ is obtained from $\mathbf{P}^{\mathrm{C}}$ as follows:

$$\mathbf{P}^{\mathrm{H}_{\mathrm{G}}}(j, z, y, x) = \exp\left(-\frac{(x - \mathbf{p}_{j}^{\mathrm{C},x})^2 + (y - \mathbf{p}_{j}^{\mathrm{C},y})^2 + (z - \mathbf{p}_{j}^{\mathrm{C},z})^2}{2\sigma^2}\right), \tag{3}$$

where $\mathbf{p}_{j}^{\mathrm{C},x}$, $\mathbf{p}_{j}^{\mathrm{C},y}$ and $\mathbf{p}_{j}^{\mathrm{C},z}$ are jth joint x-, y-, and z-axis coordinates from $\mathbf{P}^{\mathrm{C}}$, respectively. σ is set to 2.5.

From $\mathbf{P}^{\mathrm{H}_{\mathrm{G}}}$ and $\tilde{\mathbf{F}}_{\mathrm{P}}$, we obtain image feature $\mathbf{F}_{\mathrm{M}}$ as follows:

$$\mathbf{F}_{\mathrm{M}} = \mathrm{ResNet}_{\mathrm{M}}(f_{\mathrm{M}}(\psi(\mathbf{P}^{\mathrm{H}_{\mathrm{G}}}) \oplus \tilde{\mathbf{F}}_{\mathrm{P}})), \tag{4}$$

where $\psi \colon \mathbb{R}^{J \times D \times 8h \times 8w} \rightarrow \mathbb{R}^{JD \times 8h \times 8w}$ and $\oplus$ denote reshape function and concatenation along the channel dimension, respectively. f_{M} is a convolutional block that consists of a 3-by-3 convolutional layer, 2D batch normalization layer, and ReLU function. It changes the channel dimension of the input to the input channel dimension of the first residual block of the ResNet. $\mathrm{ResNet}_{\mathrm{M}}$ is the ResNet starting from the first residual block.

From the $\mathbf{F}_\mathrm{M}$, MeshNet outputs three lixel-based 1D heatmaps of all mesh vertices $\mathcal{M}^\mathrm{H} = \{\mathbf{M}^{\mathrm{H},x}, \mathbf{M}^{\mathrm{H},y}, \mathbf{M}^{\mathrm{H},z}\}$ in an exactly the same manner with that of PoseNet, as illustrated in Figure 3 (a). Likewise heatmaps of PoseNet, $\mathbf{M}^{\mathrm{H},x}$ and $\mathbf{M}^{\mathrm{H},y}$ are defined in x- and y-axis of the image space, while $\mathbf{M}^{\mathrm{H},z}$ is defined in root joint-relative depth space. We obtain x- and y-axis 1D human mesh heatmaps as follows:

$$\mathbf{M}^{\mathrm{H},x} = f_\mathrm{M}^{\mathrm{1D},x}(\mathrm{avg}^y(f_\mathrm{M}^\mathrm{up}(\mathbf{F}_\mathrm{M}))) \quad \text{and} \quad \mathbf{M}^{\mathrm{H},y} = f_\mathrm{M}^{\mathrm{1D},y}(\mathrm{avg}^x(f_\mathrm{M}^\mathrm{up}(\mathbf{F}_\mathrm{M}))), \tag{5}$$

where f_M^up denotes the three upsampling modules of the MeshNet. $f_\mathrm{M}^{\mathrm{1D},i}$ denote a 1-by-1 1D convolution that changes channel dimension from c' to V for i-axis 1D human mesh heatmap estimation, respectively. Figure 3 (b) shows visualized $f_\mathrm{M}^\mathrm{up}(\mathbf{F}_\mathrm{M})$, $\mathbf{M}^{\mathrm{H,x}}$, and $\mathbf{M}^{\mathrm{H,y}}$.

We obtain z-axis 1D human mesh heatmaps as follows:

$$\mathbf{M}^{\mathrm{H},z} = f_\mathrm{M}^{\mathrm{1D},z}(\psi(f_\mathrm{M}(\mathrm{avg}^{x,y}(\mathbf{F}_\mathrm{M})))), \tag{6}$$

where f_M and $\psi\colon \mathbb{R}^{c'D} \rightarrow \mathbb{R}^{c' \times D}$ denote a building block and and reshape function, respectively. The building block consists of a fully-connected layer, 1D batch normalization layer, and ReLU function, and it changes the activation size from c to $c'D$. Likewise we did in the PoseNet, we convert the discretized heatmaps of $\mathcal{M}^\mathrm{H}$ to continuous coordinates $\mathbf{M}^\mathrm{C} = [\mathbf{m}^{\mathrm{C},x}, \mathbf{m}^{\mathrm{C},y}, \mathbf{m}^{\mathrm{C},z}] \in \mathbb{R}^{V \times 3}$ by soft-argmax [41].

3.3 Final 3D human pose and mesh

The final 3D human mesh $\mathbf{M}$ and pose $\mathbf{P}$ are obtained as follows:

$$\mathbf{M} = \Pi(\mathbf{T}^{-1}\mathbf{M}^\mathrm{C} + \mathbf{R}) \quad \text{and} \quad \mathbf{P} = \mathcal{J}\mathbf{M}, \tag{7}$$

where Π, $\mathbf{T}^{-1}$, and $\mathbf{R} \in \mathbb{R}^{1 \times 3}$ denote camera back-projection, inverse affine transformation (*i.e.*, 2D crop and resize), and z-axis offset whose element is a depth of the root joint, respectively. $\mathbf{R}$ is obtained from RootNet [30]. We use normalized camera intrinsic parameters if not available following Moon et al. [30]. $\mathcal{J} \in \mathbb{R}^{J \times V}$ is a joint regression matrix defined in SMPL or MANO model.

3.4 Loss functions

PoseNet pose loss. To train the PoseNet, we use $L1$ loss function defined as follows:

$$L_\mathrm{pose}^\mathrm{PoseNet} = \|\mathbf{P}^\mathrm{C} - \mathbf{P}^{\mathrm{C}*}\|_1, \tag{8}$$

where $*$ indicates groundtruth. z-axis loss becomes zero if z-axis groundtruth is unavailable.

MeshNet pose loss. To train the MeshNet to predict mesh vertex aligned with body joint locations, we use $L1$ loss function defined as follows:

$$L_\mathrm{pose}^\mathrm{MeshNet} = \|\mathcal{J}\mathbf{M}^\mathrm{C} - \mathbf{P}^{\mathrm{C}*}\|_1, \tag{9}$$

where $*$ indicates groundtruth. z-axis loss becomes zero if z-axis groundtruth is unavailable.

Mesh vertex loss. To train the MeshNet to output mesh vertex heatmaps, we use $L1$ loss function defined as follows:

$$L_{\text{vertex}} = \|\mathbf{M}^{\text{C}} - \mathbf{M}^{\text{C}*}\|_1, \tag{10}$$

where $*$ indicates groundtruth. z-axis loss becomes zero if z-axis groundtruth is unavailable.

Mesh normal vector loss. Following Wang et al. [43], we supervise normal vector of predicted mesh to get visually pleasing mesh result. The $L1$ loss function for normal vector supervision is defined as follows:

$$L_{\text{normal}} = \sum_{f} \sum_{\{i,j\}\subset f} \left| \left\langle \frac{\mathbf{m}_i^{\text{C}} - \mathbf{m}_j^{\text{C}}}{\|\mathbf{m}_i^{\text{C}} - \mathbf{m}_j^{\text{C}}\|_2}, n_f^* \right\rangle \right|, \tag{11}$$

where f and n_f indicate a mesh face and unit normal vector of face f, respectively. $\mathbf{m}_i^{\text{C}}$ and $\mathbf{m}_j^{\text{C}}$ denote ith and jth vertex coordinates of $\mathbf{M}^{\text{C}}$, respectively. n_f^* is computed from $\mathbf{M}^{\text{C}*}$, where $*$ denotes groundtruth. The loss becomes zero if groundtruth 3D mesh is unavailable.

Mesh edge length loss. Following Wang et al. [43], we supervise edge length of predicted mesh to get visually pleasing mesh result. The $L1$ loss function for edge length supervision is defined as follows:

$$L_{\text{edge}} = \sum_{f} \sum_{\{i,j\}\subset f} = \left| \|\mathbf{m}_i^{\text{C}} - \mathbf{m}_j^{\text{C}}\|_2 - \|\mathbf{m}_i^{\text{C}*} - \mathbf{m}_j^{\text{C}*}\|_2 \right|, \tag{12}$$

where f and $*$ indicate mesh face and groundtruth, respectively. $\mathbf{m}_i^{\text{C}}$ and $\mathbf{m}_j^{\text{C}}$ denote ith and jth vertex coordinates of $\mathbf{M}^{\text{C}}$, respectively. The loss becomes zero if groundtruth 3D mesh is unavailable.

We train our I2L-MeshNet in an end-to-end manner using all the five loss functions as follows:

$$L = L_{\text{pose}}^{\text{PoseNet}} + L_{\text{pose}}^{\text{MeshNet}} + L_{\text{vertex}} + \lambda L_{\text{normal}} + L_{\text{edge}}, \tag{13}$$

where $\lambda = 0.1$ is a weight of L_{normal}. For the stable training, we do not backpropagate gradients before $\mathbf{P}^{\text{H}_\text{G}}$.

4 Implementation details

PyTorch [33] is used for implementation. The backbone part is initialized with the publicly released ResNet-50 [12] pre-trained on the ImageNet dataset [40], and the weights of the remaining part are initialized by Gaussian distribution with $\sigma = 0.001$. The weights are updated by the Adam optimizer [20] with a mini-batch size of 48. To crop the human region from the input image, we use groundtruth bounding box in both of training and testing stages following previous works [18,21,22]. When the bounding box is not available in the testing stage, we trained and tested Mask R-CNN [11] to get the bounding box. The cropped human image is resized to 256×256, thus $D = 64$ and $h = w = 8$. Data augmentations including scaling (±25%), rotation (±60°), random horizontal flip, and color jittering (±20%) is performed in training. The initial learning rate is set to 10^{-4} and reduced by a factor of 10 at the 10^{th} epoch. We train our model for 12 epochs with three NVIDIA RTX 2080Ti GPUs, which takes 36 hours for training. Our I2L-MeshNet runs at a speed of 25 frames per second (fps).

5 Experiment

5.1 Datasets and evaluation metrics

Human3.6M. Human3.6M [14] contains 3.6M video frames with 3D joint coordinate annotations. Because of the license problem, previously used groundtruth SMPL parameters of the Human3.6M are inaccessible. Alternatively, we used SMPLify-X [34] to obtain groundtruth SMPL parameters. Please see the supplementary material for a detailed description of SMPL parameters of the Human3.6M. MPJPE and PA MPJPE are used for the evaluation [30], which is Euclidean distance (mm) between predicted and groundtruth 3D joint coordinates after root joint alignment and further rigid alignment, respectively.

3DPW. 3DPW [26] contains 60 video sequences captured mostly in outdoor conditions. We use this dataset only for evaluation on its defined test set following Kolotouros et al. [21]. The same evaluation metrics with Human3.6M (*i.e.*, MPJPE and PA MPJPE) are used, following Kolotouros et al. [21].

FreiHAND. FreiHAND [8] contains real-captured 130K training images and 4K test images with MANO pose and shape parameters. The evaluation is performed at an online server. Following Zimmermann et al. [8], we report PA MPVPE, PA MPJPE, and F-scores.

MSCOCO. MSCOCO [24] contains large-scale in-the-wild images with 2D bounding box and human joint coordinates annotations. We fit SMPL using SMPLify-X [34] on the groundtruth 2D poses, and used the fitted meshes as groundtruth 3D meshes. This dataset is used only for the training.

MuCo-3DHP. MuCo-3DHP [28] is generated by compositing the existing MPI-INF-3DHP 3D [27]. 200K frames are composited, and half of them have augmented backgrounds. We used images of MSCOCO dataset that do not include humans to augment the backgrounds following Moon et al. [30]. This dataset is used only for the training.

5.2 Ablation study

All models for the ablation study are trained and tested on Human3.6M. As Human3.6M is the most widely used large-scale benchmark, we believe this dataset is suitable for the ablation study.

targets	spatial	uncertainty	MPJPE	no. param.	GPU mem.
SMPL param.	✗	✗	100.3	91M	4.3 GB
xyz coord.	✗	✗	114.3	117M	5.4 GB
xyz lixel hm. wo. spatial	✗	✓	92.6	82M	4.5 GB
xyz lixel hm. (ours)	✓	✓	**86.2**	**73M**	**4.6 GB**

Table 1. The MPJPE, the number of parameters, and the GPU memory usage comparison between various target representations on Human3.6M.

targets	mem. complx.	resolution	MPJPE	GPU mem.
xyz voxel hm.	$\mathcal{O}(VD^3)$	8×8×8	102.8	4.3 GB
		16×16×16	-	OOM
xy pixel hm. + z lixel hm.	$\mathcal{O}(VD^2)$	8×8, 8	97.9	3.5 GB
		32×32, 32	89.4	5.7 GB
		64×64, 64	-	OOM
xyz lixel hm. (ours)	$\mathcal{O}(VD)$	8, 8, 8	100.2	3.4 GB
		32, 32 ,32	94.8	4.0 GB
		64, 64, 64	**86.2**	**4.6 GB**

Table 2. The MPJPE and the GPU memory usage comparison between various heatmap representations on Human3.6M.

Benefit of the heatmap-based mesh estimation. To demonstrate the benefit of the heatmap-based mesh estimation, we compare models with various target representations of the human mesh, such as SMPL parameters, vertex coordinates, and heatmap. Table 1 shows MPJPE, the number of parameters, and the GPU memory usage comparison between models with different targets. The table shows that our heatmap-based mesh estimation network achieves the lowest errors while using the smallest number of the parameters and consuming small GPU memory.

The superiority of our heatmap-based mesh estimation network is in two folds. First, it can model the uncertainty of the prediction. To validate this, we trained two models that estimate the camera-centered mesh vertex coordinates directly and estimates lixel-based 1D heatmap of the coordinates using two fully-connected layers. Note that the targets of the two models are the same, but their representations are different. As the first network regresses the coordinates directly, it cannot model the uncertainty on the prediction, while the latter one can because of the heatmap target representation. However, both do not preserve the spatial relationship in the input image because of the global average pooling and the fully-connected layers. As the second and third row of the table show, modeling uncertainty on the prediction significantly decreases the errors while using a smaller number of parameters. In addition, it achieves lower errors than the SMPL parameter regression model, which is the most widely used target representation but cannot model the uncertainty.

settings	3D pose	MPJPE	PA MPJPE
MeshNet	✗	86.2	59.8
PoseNet+MeshNet (ours)	✓	**81.8**	**58.0**
MeshNet	GT	25.5	17.1

Table 3. The MPJPE and PA MPJPE comparison between various network cascading strategies on Human3.6M.

Second, it preserves the spatial relationship between pixels in the input image. The final model estimates the x- and y-axis heatmaps of each mesh vertex in a fully-convolutional way, thus preserves the spatial relationship. It achieves the best performance with the smallest number of the parameters while consuming similar GPU memory usage compared with SMPL parameter regression method that requires the least amount of GPU memory.

In Table 1, all models have the same network architecture with our I2L-MeshNet except for the final output prediction part. We removed PoseNet from all models, and the remaining MeshNet directly estimates targets from the input image $\mathbf{I}$. Except for the last row (ours), all settings output targets using two fully-connected layers. We followed the training details of [18,21] for the SMPL parameter estimation.

Lixel-based vs. pixel-based vs. voxel-based heatmap. To demonstrate the effectiveness of the lixel-based 1D heatmap over other heatmap representations, we train three models that predict lixel-based, pixel-based, and voxel-based heatmap, respectively. We used the same network architecture (*i.e.*, MeshNet of the I2L-MeshNet) for all settings except for the final prediction part. Their networks directly predict the heatmaps from the input image. x-, y-, and z-axis of each heatmap represents the same coordinates. Table 2 shows memory complexity, heatmap resolution, MPJPE and GPU memory usage comparison between models that predict different target representations of human mesh. The table shows that our lixel-based one achieves the lowest error while consuming small GPU memory usage.

Compared with the pixel-based and voxel-based heatmap, our lixel-based one consumes much less amount of GPU memory under the same resolution. The $8 \times 8 \times 8$ voxel-based heatmap requires similar GPU memory usage with that of $64, 64, 64$ lixel-based one, and we found that enlarging the voxel-based heatmap size from it is not allowed in current GPU memory limit (*i.e.*, 12 GB). The pixel-based heatmap is more efficient than the voxel-based one; however still much inefficient than our lixel-based one, which makes enlarging from $32 \times 32, 32$ impossible. This inefficient memory usage limits the heatmap resolution; however, we found that the heatmap resolution is critical for dense mesh vertex localization. On the other hand, the memory complexity of our lixel-based heatmap is a linear function with respect to D; thus, we can predict high-resolution heatmap for each mesh vertex. The memory efficiency will be more important when a high-resolution human mesh model is used.

methods	Human3.6M		3DPW	
	MPJPE	PA MPJPE	MPJPE	PA MPJPE
HMR [18]	153.2	85.5	300.4	137.2
GraphCMR [22]	78.3	59.5	126.5	80.1
SPIN [21]	72.9	51.9	113.1	71.7
I2L-MeshNet (Ours)	**55.7**	**41.7**	**95.4**	**60.8**

Table 4. The MPJPE and PA MPJPE comparison on Human3.6M and 3DPW. All methods are trained on Human3.6M and MSCOCO.

methods	MPJPE	PA MPJPE
SMPLify [4]	-	82.3
Lassner [23]	-	93.9
HMR [18]	88.0	56.8
NBF [32]	-	59.9
Pavlakos [37]	-	75.9
Kanazawa [19]	-	56.9
GraphCMR [22]	-	50.1
Arnab [2]	77.8	54.3
SPIN [21]	-	**41.1**
I2L-MeshNet (Ours)	**55.7**	41.7

Table 5. The MPJPE and PA MPJPE comparison on Human3.6M. Each method is trained on different datasets.

methods	MPJPE	PA MPJPE
HMR [18]	-	81.3
Kanazawa [19]	-	72.6
GraphCMR [22]	-	70.2
Arnab [2]	-	72.2
SPIN [21]	-	59.2
I2L-MeshNet (Ours)	**93.2**	**58.6**
I2L-MeshNet (Ours) + SMPL regress	99.6	62.9

Table 6. The MPJPE and PA MPJPE comparison on 3DPW. Each method is trained on different datasets.

Under the same resolution, the combination of pixel-based heatmap and lixel-based heatmap achieves the best performance. We think that estimating the voxel-based heatmap involves too many parameters at a single output layer, which makes it produce high errors. In addition, lixel-based heatmap inherently involves spatial ambiguity arises from marginalizing the 2D feature map to 1D, which can be a possible reason for worse performance than the combined one.

Benefit of the cascaded PoseNet and MeshNet. To demonstrate the benefit of the cascaded PoseNet and MeshNet, we trained and tested three networks using various network cascading strategy. First, we removed PoseNet from the I2L-MeshNet. The remaining MeshNet directly predicts lixel-based 1D heatmap of each mesh vertex from the input image. Second, we trained I2L-MeshNet, which has cascaded PoseNet and MeshNet architecture. Third, to check the upper bound accuracy with respect to the output of the PoseNet, we fed the groundtruth 3D human pose instead of the output of the PoseNet to the MeshNet in both training and testing stage. Table 3 shows utilizing the output of the PoseNet (the second row) achieves better accuracy compared with using only MeshNet (the first row) to estimate the human mesh. Interestingly, passing the groundtruth 3D human pose to the MeshNet (the last row) significantly improves the performance compared with all the other settings. This indicates

methods	PA MPVPE	PA MPJPE	F@5 mm	F@15 mm	GT scale
Hasson et al. [10]	13.2	-	0.436	0.908	✓
Boukhayma et al. [5]	13.0	-	0.435	0.898	✓
FreiHAND [8]	10.7	-	0.529	0.935	✓
I2L-MeshNet (Ours)	**7.6**	**7.4**	**0.681**	**0.973**	✗

Table 7. The PA MPVPE, PA MPJPE, and F-scores comparison between state-of-the-art methods and the proposed I2L-MeshNet on FreiHAND. The checkmark denotes a method use groundtruth information during inference time.

that improving the 3D human pose estimation network can be one important way to improve 3D human mesh estimation accuracy.

5.3 Comparison with state-of-the-art methods

Human3.6M and 3DPW. We compare the MPJPE and PA MPJPE of our I2L-MeshNet with previous state-of-the-art 3D human body pose and mesh estimation methods on Human3.6M and 3DPW test set. As each previous work trained their network on different training sets, we report the 3D errors in two ways.

First, we train all methods on Human3.6M and MSCOCO and report the errors in Table 4. The previous state-of-the-art methods [18, 21, 22] are trained from their officially released codes. The table shows that our I2L-MeshNet significantly outperforms previous methods by a large margin on both datasets.

Second, we report the 3D errors of previous methods from their papers and ours in Table 5 and Table 6. Each network of the previous method is trained on the different combinations of datasets, which include Human3.6M, MSCOCO, MPII [1], LSP [15], LSP-Extended [16], UP [23], and MPI-INF-3DHP [27]. We used MuCo-3DHP for the additional training dataset for the evaluation on 3DPW dataset. We also report the 3D errors from a additional SMPL parameter regression module following Kolotouros et al. [22]. The tables show that the performance gap between ours and the previous state-of-the-art method [21] is significantly reduced.

The reason for the reduced performance gap is that previous model-based state-of-the-art methods [18, 21] can get benefit from many in-the-wild 2D human pose datasets [15, 16, 24] by a 2D pose-based weak supervision. As the human body or hand model assumes a prior distribution between the human model parameters (*i.e.*, 3D joint rotations and identity vector) and 3D joint/mesh coordinates, the 2D pose-based weak supervision can provide gradients in depth axis, calculated from the prior distribution. Although the weak supervision still suffers from the depth ambiguity, utilizing in-the-wild images can be highly beneficial because the images have diverse appearances compared with those of the lab-recorded 3D datasets [14, 27, 28]. On the other hand, model-free approaches, including the proposed I2L-MeshNet, do not assume any prior distribution, therefore hard to get benefit from the weak supervision. Based on the two comparisons, we can draw two important conclusions.

- The model-free approaches achieve higher accuracy than the model-based ones when trained on the same datasets that provide groundtruth 3D human poses and meshes.
- The model-based approaches can achieve higher accuracy by utilizing additional in-the-wild 2D pose data without requiring the 3D supervisions.

We think that a larger number of accurately aligned in-the-wild image-3D mesh data can significantly boost the accuracy of the model-free approaches. The iterative fitting [4,34], neural network [17], or their combination [21] can be used to obtain more data. This can be an important future research direction, and we leave this as future work.

FreiHAND. We compare MPVPE and F-scores of our I2L-MeshNet with previous state-of-the-art 3D human hand pose and mesh estimation methods [5,8,10]. We trained Mask R-CNN [11] on FreiHAND train images to get the hand bounding box of test images. Table 7 shows that the proposed I2L-MeshNet significantly outperforms all previous works without groundtruth scale information during the inference time. We additionally report MPJPE in the table.

6 Conclusion

We propose a I2L-MeshNet, image-to-lixel prediction network for accurate 3D human pose and mesh estimation from a single RGB image. We convert the output of the network to the lixel-based 1D heatmap, which preserves the spatial relationship in the input image and models uncertainty of the prediction. Our lixel-based 1D heatmap requires much less GPU memory usage under the same heatmap resolution while producing better accuracy compared with a widely used voxel-based 3D heatmap. Our I2L-MeshNet outperforms previous 3D human pose and mesh estimation methods on various 3D human pose and mesh datasets. We hope our method can give useful insight to the following model-free 3D human pose and mesh estimation approaches.

Acknowledgments

This work was supported by IITP grant funded by the Ministry of Science and ICT of Korea (No. 2017-0-01780), and Hyundai Motor Group through HMG-SNU AI Consortium fund (No. 5264-20190101).

References

1. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2D human pose estimation: New benchmark and state of the art analysis. In: CVPR (2014)

2. Arnab, A., Doersch, C., Zisserman, A.: Exploiting temporal context for 3D human pose estimation in the wild. In: CVPR (2019)
3. Baek, S., In Kim, K., Kim, T.K.: Pushing the envelope for RGB-based dense 3D hand pose estimation via neural rendering. In: CVPR (2019)
4. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In: ECCV (2016)
5. Boukhayma, A., de Bem, R., Torr, P.H.: 3D hand shape and pose from images in the wild. In: CVPR (2019)
6. Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J.: Cascaded pyramid network for multi-person pose estimation. In: CVPR (2018)
7. Choi, H., Moon, G., Lee, K.M.: Pose2Mesh: Graph convolutional network for 3D human pose and mesh recovery from a 2D human pose. ECCV (2020)
8. Christian Zimmermann, Duygu Ceylan, J.Y.B.R.M.A., Brox, T.: FreiHAND: A dataset for markerless capture of hand pose and shape from single RGB images. In: ICCV (2019)
9. Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., Yuan, J.: 3D hand shape and pose estimation from a single RGB image. In: CVPR (2019)
10. Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: CVPR (2019)
11. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
13. Ioffe, S., Szegedy, C.: Batch Normalization: Accelerating deep network training by reducing internal covariate shift. ICML (2015)
14. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. TPAMI (2014)
15. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: BMVC (2010)
16. Johnson, S., Everingham, M.: Learning effective human pose estimation from inaccurate annotation. In: CVPR (2011)
17. Joo, H., Neverova, N., Vedaldi, A.: Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation. arXiv preprint arXiv:2004.03686 (2020)
18. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: CVPR (2018)
19. Kanazawa, A., Zhang, J.Y., Felsen, P., Malik, J.: Learning 3D human dynamics from video. In: CVPR (2019)
20. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. ICLR (2014)
21. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In: ICCV (2019)
22. Kolotouros, N., Pavlakos, G., Daniilidis, K.: Convolutional mesh regression for single-image human shape reconstruction. In: CVPR (2019)
23. Lassner, C., Romero, J., Kiefel, M., Bogo, F., Black, M.J., Gehler, P.V.: Unite the people: Closing the loop between 3D and 2D human representations. In: CVPR (2017)
24. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)

25. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM TOG (2015)
26. von Marcard, T., Henschel, R., Black, M.J., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3D human pose in the wild using imus and a moving camera. In: ECCV (2018)
27. Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., Theobalt, C.: Monocular 3D human pose estimation in the wild using improved cnn supervision. In: 3DV (2017)
28. Mehta, D., Sotnychenko, O., Mueller, F., Xu, W., Sridhar, S., Pons-Moll, G., Theobalt, C.: Single-shot multi-person 3D pose estimation from monocular RGB. In: 3DV (2018)
29. Moon, G., Chang, J.Y., Lee, K.M.: V2V-PoseNet: Voxel-to-voxel prediction network for accurate 3D hand and human pose estimation from a single depth map. In: CVPR (2018)
30. Moon, G., Chang, J.Y., Lee, K.M.: Camera distance-aware top-down approach for 3D multi-person pose estimation from a single RGB image. In: ICCV (2019)
31. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: ECCV (2016)
32. Omran, M., Lassner, C., Pons-Moll, G., Gehler, P., Schiele, B.: Neural Body Fitting: Unifying deep learning and model based human pose and shape estimation. In: 3DV. IEEE (2018)
33. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
34. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3D hands, face, and body from a single image. In: CVPR (2019)
35. Pavlakos, G., Kolotouros, N., Daniilidis, K.: TexturePose: Supervising human mesh estimation with texture consistency. In: ICCV (2019)
36. Pavlakos, G., Zhou, X., Derpanis, K.G., Daniilidis, K.: Coarse-to-fine volumetric prediction for single-image 3D human pose. In: CVPR (2017)
37. Pavlakos, G., Zhu, L., Zhou, X., Daniilidis, K.: Learning to estimate 3D human pose and shape from a single color image. In: CVPR (2018)
38. Rogez, G., Weinzaepfel, P., Schmid, C.: LCR-Net: Localization-classification-regression for human pose. In: CVPR (2017)
39. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM TOG (2017)
40. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. IJCV (2015)
41. Sun, X., Xiao, B., Wei, F., Liang, S., Wei, Y.: Integral human pose regression. In: ECCV (2018)
42. Tompson, J.J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation. In: NeurIPS (2014)
43. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2Mesh: Generating 3D mesh models from single RGB images. In: ECCV (2018)
44. Xiao, B., Wu, H., Wei, Y.: Simple baselines for human pose estimation and tracking. In: ECCV (2018)
45. Xu, Y., Zhu, S.C., Tung, T.: DenseRaC: Joint 3D pose and shape estimation by dense render-and-compare. In: ICCV (2019)