Learning 3D Human Shape and Pose from Dense Body Parts

Hongwen Zhang, Jie Cao, Guo Lu, Wanli Ouyang, *Senior Member, IEEE,* and Zhenan Sun, *Senior Member, IEEE*

Abstract—Reconstructing 3D human shape and pose from monocular images is challenging despite the promising results achieved by the most recent learning-based methods. The commonly occurred misalignment comes from the facts that the mapping from images to the model space is highly non-linear and the rotation-based pose representation of body models is prone to result in the drift of joint positions. In this work, we investigate learning 3D human shape and pose from dense correspondences of body parts and propose a Decompose-and-aggregate Network (DaNet) to address these issues. DaNet adopts the dense correspondence maps, which densely build a bridge between 2D pixels and 3D vertices, as intermediate representations to facilitate the learning of 2D-to-3D mapping. The prediction modules of DaNet are decomposed into one global stream and multiple local streams to enable global and fine-grained perceptions for the shape and pose predictions, respectively. Messages from local streams are further aggregated to enhance the robust prediction of the rotation-based poses, where a position-aided rotation feature refinement strategy is proposed to exploit spatial relationships between body joints. Moreover, a Part-based Dropout (PartDrop) strategy is introduced to drop out dense information from intermediate representations during training, encouraging the network to focus on more complementary body parts as well as neighboring position features. The efficacy of the proposed method is validated on both indoor and real-world datasets including Human3.6M, UP3D, COCO, and 3DPW, showing that our method could significantly improve the reconstruction performance in comparison with previous state-of-the-art methods. Our code is publicly available at https://hongwenzhang.github.io/dense2mesh.

Index Terms—3D human shape and pose estimation, decompose-and-aggregate network, position-aided rotation feature refinement, part-based dropout.

1 Introduction

Reconstructing human shape and pose from a monocular image is an appealing yet challenging task, which typically involves the prediction of the camera and parameters of a statistical body model (e.g. the most commonly used SMPL [1] model). Fig. 1(a) shows an example of the reconstructed result. The challenges of this task come from the fundamental depth ambiguity, the complexity and flexibility of human bodies, and variations in clothing and viewpoint, etc. Classic optimization-based approaches [2], [3] fit the SMPL model to 2D evidence such as 2D body joints or silhouettes in images, which involve complex non-linear optimization and iterative refinement. Recently, regression-based approaches [4], [5], [6], [7] integrate the SMPL model within neural networks and predict model parameters directly in an end-to-end manner.

Though great progress has been made, the direct prediction of the body model from the image space is still complex and difficult even for deep neural networks. In this work, we propose to adopt IUV maps as intermediate representations to facilitate the learning of the mapping from images to models. As depicted in Fig. 1(b), compared with other 2D representations [4], [6], [7], the IUV map could provide more rich information, because it encodes the dense correspondence between foreground pixels on 2D images and vertices on 3D meshes. Such a dense semantic map not only contains essential information for shape and pose estimation from RGB images, but also eliminates the interference of unrelated factors such as appearance, clothing, and illumination variations.

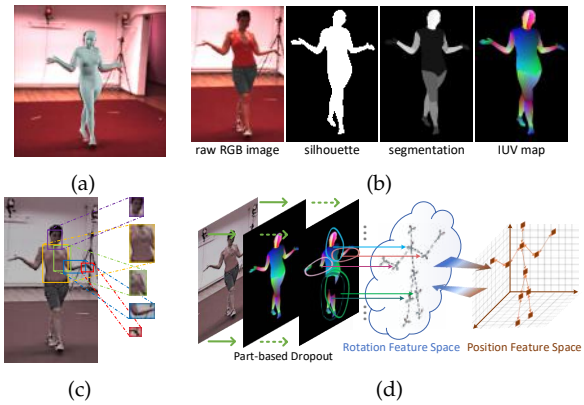


Fig. 1. Illustration of our main ideas. (a) A human image with a parametric body model. (b) Comparison of the raw RGB image, silhouette, segmentation, and IUV map. (c) Local visual cues are crucial for the perception of joint rotations. (d) Our DaNet learns 3D human shape and pose from IUV maps with decomposed perception, aggregated refinement, and part-based dropout strategies.

The representation of 3D body model [1], [8] can be factorized into the shape and pose components, depicting the model at different scales. The body shape gives an identity-dependent description about the model, while

- *H. Zhang, J. Cao, and Z. Sun are with CRIPAC, NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, and aslo with the University of Chinese Academy of Sciences, Beijing 101408, China. E-mail: hongwen.zhang@cripac.ia.ac.cn; jie.cao@cripac.ia.ac.cn; znsun@nlpr.ia.ac.cn. (Corresponding author: Zhenan Sun.)*
- *G. Lu is with the Beijing Institute of Technology, Beijing 100081, China. E-mail: guo.lu@bit.edu.cn.*
- *W. Ouyang is with the University of Sydney, NSW 2006, Australia. E-mail: wanli.ouyang@sydney.edu.au.*

the body pose provides more detailed descriptions about the rotation of each body joint. Previous regression-based methods [5], [7] typically predict them simultaneously using global information from the last layer of the neural network. We observe that the detailed pose of body joints should be captured by local visual cues instead of global information. As shown in Fig. 1(c), we can estimate the rotations of those visible body joints only based on local visual cues, while the information from other body joints and background regions would be irrelevant.

For the rotation-based pose representation of commonly used body models [1], [8], small rotation errors accumulated along the kinematic chain could lead to large drift of position at the leaf joint. Moreover, the rotation estimation is error-prone for those occluded body joints since their perceptions are less reliable under occlusions. Hence, it is crucial to utilize information from visible body joints and the prior about the structure of human bodies. As shown in previous work [9], [10], the structural information at the feature level is helpful for more robust and accurate pose estimation. However, it is non-trivial to apply these feature refinement methods to our case due to the weak correlation between rotation-based poses of different joints. For instance, the shoulder, elbow, and wrist are three consecutive body joints, and one can hardly infer the relative rotation of wrist w.r.t. the elbow given the relative rotation of elbow w.r.t. the shoulder. On the other hand, we observe that the 3D locations of body joints have stronger correlations than the rotation of body joints. For instance, the positions of shoulder, elbow, and wrist are strongly constrained by the length of the arm.

Based on the observations above, we propose a Decompose-and-aggregate Network (DaNet) to learn 3D human shape and pose from dense correspondences of body parts. As illustrated in Fig. 1(d), DaNet utilizes IUV maps as the intermediate information for more efficient learning, and decomposes the prediction modules into multiple streams considering that the prediction of different parameters requires the receptive fields with different sizes. To robustly predict the rotations of body joints, DaNet aggregates messages from different streams and refines the rotation features via an auxiliary position feature space to exploit the spatial relationships between body joints. For better generalization, a Part-based Dropout (PartDrop) strategy is further introduced to drop out dense information from intermediate representations during training, which could effectively regularize the network and encourage it to learn features from complementary body parts and leverage information from neighboring body joints. As will be validated in our experiments, all the above new designs could contribute to better part-based learning and improve the reconstruction performance. To sum up, the main contributions in this work are listed as follows.

- We comprehensively study the effectiveness of adopting the IUV maps in both global and local scales, which contains densely semantic information of body parts, as intermediate representations for the task of 3D human pose and shape estimation.
- Our reconstruction network is designed to have decomposed streams to provide global perception for the camera and shape prediction while detailed perception for pose prediction of each body joint.
- A part-based dropout strategy is introduced to drop dense information from intermediate representations during training. Such a strategy can encourage the network to learn features from complementary body parts, which also has the potential for other structured image understanding tasks.
- A position-aided rotation feature refinement strategy is proposed to aggregate messages from different part features. It is more efficient to exploit the spatial relationship in an auxiliary position feature space since the correlations between position features are much stronger.

An early version of this work appeared in [11]. We have made significant extensions to our previous work in three main aspects. First, the methodology is improved to be more accurate and robust thanks to several new designs, including the part-based dropout strategy for better generalization performance and the customized graph convolutions for more efficient and better feature mapping and refinement. Second, more extensive evaluations and comparisons are included to validate the effectiveness of our method, including evaluations on additional datasets and comparisons of the reconstruction errors across different human actions and model surface areas. Third, more discussions are provided in our ablation studies, including comprehensive evaluations on the benefit of adopting IUV as intermediate representations and in-depth analyses on the refinement upon the rotation feature space and position feature space.

The remainder of this paper is organized as follows. Section 2 briefly reviews previous work related to ours. Section 3 provides preliminary knowledge about the SMPL model and IUV maps. Details of the proposed network are presented in Section 4. Experimental results and analyses are included in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

2.1 3D Human Shape and Pose Estimation

Early pioneering work on 3D human model reconstruction mainly focuses on the optimization of the fitting process. Among them, [12], [13] fit the body model SCAPE [8] with the requirement of ground truth silhouettes or manual initialization. Bogo et al. [2] introduce the optimization method SMPLify and make the first attempt to automatically fit the SMPL model to 2D body joints by leveraging multiple priors. Lassner et al. [3] extend this method and improve the reconstruction performance by incorporating silhouette information in the fitting procedure. These optimization-based methods typically rely on accurate 2D observations and the prior terms imposed on the shape and pose parameters, making the procedure time-consuming and sensitive to the initialization. Alternatively, recent regression-based methods employ neural networks to predict the shape and pose parameters directly and learn the priors in a data-driven manner. These efforts mainly focus on several aspects including intermediate representation leveraging, architecture designs, structural information modeling, and re-projection loss designs, etc. Our work makes contributions to the first three aspects above and is also complementary to the work focusing on the re-projection loss designs [4], [14],

[15], reconstruction from videos or multi-view images [16], [17], [18], [19], [20], and detailed or holistic body model learning [21], [22], [23].

2.1.1 Intermediate Representation

The recovery of the 3D human pose from a monocular image is challenging. Common strategies use intermediate estimations as the proxy representation to alleviate the difficulty. These methods can benefit from existing state-of-the-art networks for lower-level tasks. For the recovery of 3D human pose or human model, 2D joint positions [24], [25], [26], [27], silhouette [6], [28], [29], segmentation [7], depth maps [30], [31], joint heatmaps [4], [6], [32], volumetric representation [33], [34], [35], [36], and 3D orientation fields [37], [38] are adopted in literature as intermediate representations to facilitate the learning task. Though the aforementioned representations are helpful for the task, detailed information contained within body parts is missing in these coarse representations, which becomes the bottleneck for fine-grained prediction tasks. Recently, DensePose [39] regresses the IUV maps directly from images, which provides the dense correspondence mapping from the image to the human body model. However, the 3D model cannot be directly retrieved from such a 2.5D projection. In our work, we propose to adopt such a densely semantic map as the intermediate representation for the task of 3D human shape and pose estimation. To the best of our knowledge, we are among the first attempts [15], [40], [41] to leverage IUV maps for 3D human model recovery. In comparison, the major differences between concurrent efforts and ours lie in three aspects: 1) [15], [40], [41] obtain IUV predictions from a pretrained network of DensePose [39], while our work augments the annotations of 3D human pose datasets with the rendered ground-truth IUV maps and imposes dense supervisions on the intermediate representations; 2) [15], [40], [41] only leverage global IUV maps, while our work exploits using IUV maps in both global and local scales; 3) DenseRaC [41] resorts to involving more synthetic IUV maps as additional training data while our work introduces the part-based dropout upon IUV maps to improve generalization. We believe these concurrent work complement each other and enrich the research community.

2.1.2 Architecture Design

Existing approaches to 3D human shape and pose estimation have designed a number of network architectures for more effective learning of the highly nonlinear image-to-model mapping. Tan et al. [42] develop an encoder-decoder based framework where the decoder learns the SMPL-to-silhouette mapping from synthetic data and the encoder learns the image-to-SMPL mapping with the decoder frozen. Kanazawa et al. [5] present an end-to-end framework HMR to reconstruct the SMPL model directly from images using a single CNN with an iterative regression module. Kolotouros et al. [43] enhance HMR with the fitting process of SMPLify [2] to incorporate regression- and optimization-based methods. Pavlakos et al. [6] propose to predict the shape and pose parameters from the estimated silhouettes and joint locations respectively. Sun et al. [44] also leverage joint locations and further involve deep features into the prediction process. Instead of regressing the shape and pose parameters directly, Kolotouros et al. [40] employ a Graph CNN [45] to regress the 3D coordinates of the human mesh vertices, while Yao et al. [46] regress the 3D coordinates in the form of an unwrapped position map. All aforementioned regression-based methods predict the pose in a global manner. In contrast, our DaNet predicts joint poses from multiple streams, hence the visual cues could be captured in a fine-grained manner. Recently, Güler et al. [14] also introduce a part-based reconstruction method to predict poses from the deep features pooled around body joints. In comparison, the pooling operation of our DaNet is performed on intermediate representations, enabling detailed perception for better pose feature learning. Moreover, existing approaches for rotation-based pose estimation do not consider feature refinement, while DaNet includes an effective rotation feature refinement scheme for robust pose predictions.

2.1.3 Structural Information Modeling

Leveraging the articulated structure information is crucial for human pose modeling [47], [48], [49]. Recent deep learning-based approaches to human pose estimation [9], [10], [50], [51], [52] incorporate the structured feature learning in their network architecture designs. All these efforts exploit the relationship between the *position features* of body joints and their feature refinement strategies are only validated on the position-based pose estimation problem. Our work is complementary to them by investigating the refinement for *rotation features* under the context of the rotation-based pose representation, which paves a new way to impose structural constraints upon rotation features. Our solution aggregates the rotation features into a position feature space, where the aforementioned structural feature learning approaches could be easily applied.

For more geometrically reasonable pose predictions, different types of pose priors [53], [54], [55], [56], [57] are also employed as constraints in the learning procedure. For instance, Akhter and Black [53] learn the pose prior in the form of joint angle constraints. Sun et al. [55] design handcrafted constraints such as limb-lengths and their proportions. Similar constraints are exploited in [56] under the weakly-supervised setting. For the rotation-based pose representation in the SMPL model, though it inherently satisfies structure constraints such as limb proportions, the pose prior is still essential for better reconstruction performance. SMPLify [2] imposes several penalizing terms on predicted poses to prevent unnatural results. Kanazawa et al. [5] introduce an adversarial prior for guiding the prediction to be realistic. All these methods consider the pose prior at the *output level*. In our work, we will exploit the relationship at the *feature level* for better 3D pose estimation in the SMPL model.

2.2 Regularization in Neural Networks

Regularization is important to neural networks for better generalization performance. A number of regularization techniques have been proposed to remove features from neural networks at different granularity levels. Among them, dropout [58] is commonly used at the fully connected

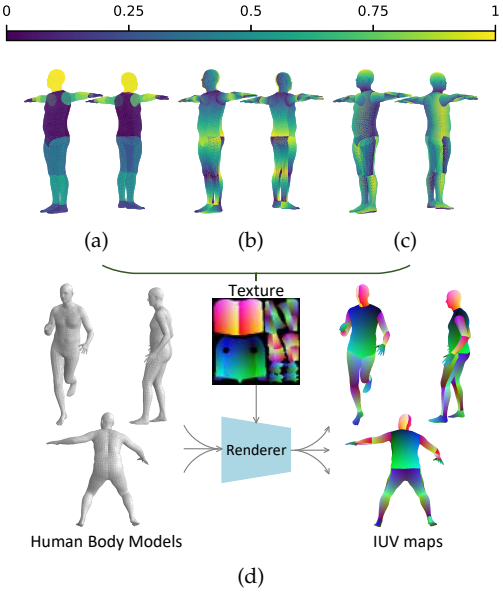


Fig. 2. Illustration of the preparation of ground truth IUV maps. (a)(b)(c) show the *Index*, U, and V values defined in DensePose [39], respectively. Note that the original *Index* values (range from 1 to 24) are also normalized into the $[0, 1]$ interval. (d) Generation of ground truth IUV Maps for 3D human body models.

layers of neural networks to drop unit-wise features independently. The introduction of dropout has inspired the development of other dropping out strategies with structured forms. For instance, SpatialDropout [59] drops channel-wise features across the entire feature map, while DropBlock [60] drops block-wise features in contiguous regions. Different from these techniques, our PartDrop strategy drops part-wise features at the granularity level of semantic body parts. Such a part-wise dropping strategy could remove patterns in a more structured manner and perform better in our learning task. Moreover, our PartDrop strategy is applied on intermediate representations, which is also different from data augmentation methods such as Cutout [61].

3 SMPL Model and IUV Maps

SMPL Model. The Skinned Multi-Person Linear model (SMPL) [1] is one of the widely used statistical human body models, which represents the body mesh with two sets of parameters, i.e., the shape and pose parameters. The shape indicates the model's height, weight and limb proportions while the pose indicates how the model deforms with the rotated skeleton joints. Such decomposition of shape and pose makes it convenient for algorithms to focus on one of these two factors independently. In the SMPL model, the shape parameters $\boldsymbol{\beta} \in \mathbb{R}^{10}$ denote the coefficients of the PCA basis of the body shape. The pose parameters $\boldsymbol{\theta} \in \mathbb{R}^{3K}$ denote the axis-angle representations of the relative rotation of K skeleton joints with respect to their parents in the kinematic tree, where $K = 24$ in the SMPL model. For simplicity, the root orientation is also included as the pose parameters of the root joint in our formulation. Given the pose and shape parameters, the model deforms accordingly and generates a triangulated mesh with $N = 6890$ vertices $\mathcal{M}(\boldsymbol{\theta}, \boldsymbol{\beta}) \in \mathbb{R}^{3\times N}$. The deformation process $\mathcal{M}(\boldsymbol{\theta}, \boldsymbol{\beta})$ is differentiable with respect to the pose $\boldsymbol{\theta}$ and shape $\boldsymbol{\beta}$, which means that the SMPL model could be integrated within a neural network as a typical layer without any learnable weights. After obtaining the final mesh, vertices could be further mapped to sparse 3D keypoints by a pretrained linear regressor.

IUV Maps. Reconstructing the 3D object model from a monocular image is ambiguous, but there are determinate correspondences between foreground pixels on 2D images and vertices on 3D surfaces. Such correspondences could be represented in the form of UV maps, where the foreground pixels contain the corresponding UV coordinate values. In this way, the pixels on the foreground could be projected back to vertices on the template mesh according to a predefined bijective mapping between the 3D surface space and the 2D UV space. For the human body model, the correspondence could have finer granularity by introducing the *Index* of the body parts [39], [62], which results in the IUV maps $\boldsymbol{H} = (\boldsymbol{H}^i|\boldsymbol{H}^u|\boldsymbol{H}^v) \in \mathbb{R}^{(1+P)\times h_{iuv}\times w_{iuv}\times 3}$, where P denotes the number of body parts, h_{iuv} and w_{iuv} denote the height and width of IUV maps. The *Index* channels $\boldsymbol{H}^i$ indicates whether a pixel belongs to the background or a specific body part, while the UV channels $\boldsymbol{H}^u$ and $\boldsymbol{H}^v$ contain the corresponding U, V values of visible body parts respectively. The IUV maps $\boldsymbol{H}$ encode *Index*, U, and V values individually for P body parts in a one-hot manner along $(1 + P)$ ways. The *Index* values for body parts count from 1 and *Index* 0 is reserved for the background. For each body part, the UV space is independent so that the representation could be more fine-grained. The IUV annotation of the human body is firstly introduced in DenseReg [62] and DensePose [39]. Figs. 2(a)(b)(c) show the *Index*, U, and V values on the SMPL model as defined in DensePose [39].

Preparation of IUV Maps for 3D Human Pose Datasets. Currently, there is no 3D human pose dataset providing IUV annotations. In this work, for those datasets providing SMPL parameters with human images, we augment their annotations by rendering the corresponding ground-truth IUV maps based on the same IUV mapping protocol of DensePose [39]. Specifically, we first construct a template texture map from IUV values of each vertex on the SMPL model, and then employ a renderer to generate IUV maps. As illustrated in Fig. 2(d), for each face in the triangulated mesh, the texture values used for rendering is a triplet vector denoting the corresponding *Index*, U, and V values. Then, given SMPL models, the corresponding IUV maps can be generated by existing rendering algorithms such as [63], [64]. Specifically, the renderer takes the template texture map and 3D model as inputs and output a rendered image with the size of $h_{iuv} \times w_{iuv} \times 3$. Afterwards, the rendered image is reorganized as the shape of $(1+P)\times h_{iuv} \times w_{iuv} \times 3$ by converting values into one-hot representations.

4 Methodology

As illustrated in Fig. 3, our DaNet decomposes the prediction task into one global stream for the camera and shape predictions and multiple local streams for joint pose predictions. The overall pipeline involves two consecutive stages. In the first stage, the IUV maps are estimated from global and local perspectives in consideration of the different sizes

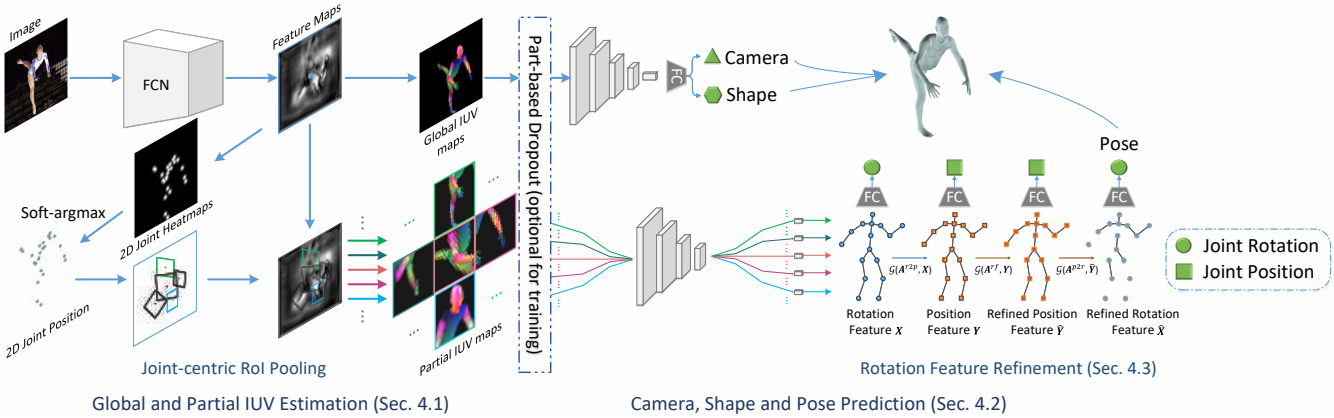


Fig. 3. Overview of the proposed Decompose-and-aggregate Network (DaNet).

of the receptive fields required by the prediction of different parameters. In the second stage, the global and local IUV maps are used for different feature extraction and prediction tasks. The global features are extracted from global IUV maps and then directly used to predict camera and body shape. The rotation features are extracted from partial IUV maps and further fed into the aggregated refinement module before the final prediction of joint poses. During training, the part-based dropout is applied to the estimated IUV maps between the above two stages.

Overall, our objective function is a combination of three objectives:

$$\mathcal{L} = \mathcal{L}_{inter} + \mathcal{L}_{target} + \mathcal{L}_{refine}, \tag{1}$$

where $\mathcal{L}_{inter}$ is the objective for estimating the intermediate representations (Sec. 4.1), $\mathcal{L}_{target}$ is the objective for predicting the camera and SMPL parameters (Sec. 4.2), $\mathcal{L}_{refine}$ is the objective involving in the aggregated refinement module (Sec. 4.3). In the following subsections, we will present the technical details and rationale of our method.

4.1 Global and Partial IUV Estimation

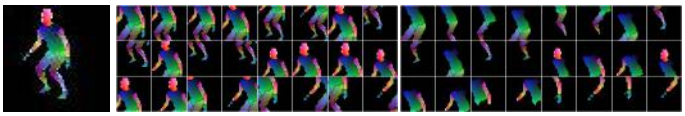
The first stage in our method aims to estimate corresponding IUV maps from input images for subsequent prediction tasks. Specifically, a fully convolutional network is employed to produce $K+1$ sets of IUV maps, including one set of global IUV maps and K sets of partial IUV maps for the corresponding K body joints. The global IUV maps are aligned with the original image through up-sampling, while the partial IUV maps are centered around the body joints. Fig. 4 visualizes a sample of the global and partial IUV maps. The feature maps outputted from the last layer of the FCN would be shared by the estimation tasks of both global and partial IUV maps. The estimation of the global IUV maps is quite straightforward since they could be obtained by simply feeding these feature maps into a convolutional layer. For the estimation of each set of partial IUV maps, a joint-centric RoI pooling would be first performed on these feature maps to extract appropriate sub-regions, which results in K sets of partial feature maps. Then, the K sets of partial IUV maps would be estimated independently from these partial feature maps. Now, we will give details about the RoI pooling process for partial IUV estimation.

Joint-centric RoI Pooling. For pose parameters in the SMPL model, they represent the relative rotation of each body joint with respect to its parent in the kinematic tree. Hence, the perception of joint poses should individually focus on corresponding body parts. In other words, globally zooming, translating the human in the image should have no effect on the pose estimation of body joints. Moreover, the ideal scale factors for the perception of joint poses should vary from one joint to another since the proportions of body parts are different. To this end, we perform joint-centric RoI pooling on feature maps for partial IUV estimation. Particularly, for each body joint, a sub-region of the feature maps is extracted and spatially transformed to a fixed resolution for subsequent partial IUV map estimation and joint pose prediction. In our implementation, the RoI pooling is accomplished by a Spatial Transformer Network (STN) [65]. In comparison with the conventional STNs, the pooling process in our network is learned in an explicitly supervised manner.

As illustrated in Fig. 5(a), the joint-centric RoI pooling operations are guided by 2D joint positions so that each sub-region is centered around the target joint. Specifically, 2D joint heatmaps are estimated along with the global IUV maps in a multi-task learning manner, and 2D joint positions are retrieved from heatmaps using the soft-argmax [66] operation. Without loss of generality, let $\boldsymbol{j}_k$ denote the position of the k-th body joint. Then, the center and scale parameters used for spatial transformation are determined individually for each set of partial IUV maps. Specifically, for the k-th set of partial IUV maps, the center $\boldsymbol{c}_k$ is the position of the k-th joint, while the scale s_k is proportional to the size of the foreground region, i.e.,

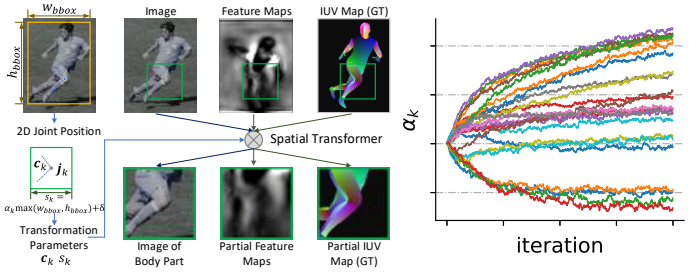
$$\begin{aligned} \boldsymbol{c}_k &= \boldsymbol{j}_k, \\ s_k &= \alpha_k \max(w_{bbox}, h_{bbox}) + \delta, \end{aligned} \tag{2}$$

where α_k and δ are two constants, w_{bbox} and h_{bbox} denote the width and height of the foreground bounding box respectively. In our implementation, the foreground is obtained from the part segmentation (i.e., *Index* channels of estimated IUV maps). Compared with our previous work [11] calculating s_k from 2D joints, the s_ks determined by foreground regions here are more robust to 2D joint localization.



(a) (b) (c)

Fig. 4. Visualization of (a) global, (b) partial, and (c) simplified partial IUV maps.



(a) (b)

Fig. 5. Joint-centric RoI pooling. (a) The RoI pooling is implemented as an STN. (b) The evolution of α_ks of different body joints over learning iterations.

Note that the above constants α_k and δ can be hand-crafted or learned in the STN by taking ground-truth IUV maps as inputs. For learned α_ks, Fig. 5(b) shows how the values of different body joints evolve over learning iterations. It can be observed that α_ks are enlarged for some joints while shrunk for others, which provides more suitable RoI sizes for each body joint.

After obtaining the transformation parameters in Eq. 2, the feature maps extracted from the last layer of fully convolutional network are spatially transformed to a fixed resolution and used to estimate the partial IUV maps, where the corresponding ground-truth ones are also extracted from the ground-truth global IUV maps using the same pooling process.

Considering that the pose of a body joint is only related to its adjacent body parts, we can further simplify partial IUV maps by discarding those irrelevant body parts. For each set of partial IUV maps, we retain specific channels corresponding to those body parts surrounding the target joint. The partial IUV maps before and after the simplification are depicted in Fig. 4(b) and Fig. 4(c) respectively.

Loss Functions. A classification loss and several regression losses are involved in the training of this stage. For both global and partial IUV maps, the loss is calculated in the same manner and denoted as $\mathcal{L}_{iuv}$. Specifically, a classification loss is imposed on the *Index* channels of IUV maps, where a $(1+P)$-way cross-entropy loss is employed to classify a pixel belonging to either background or one among P body parts. For the UV channels of IUV maps, an L_1 based regression loss is adopted, and is only taken into account for those foreground pixels. In other words, the estimated UV channels are firstly masked by the ground-truth *Index* channel before applying the regression loss. For the 2D joint heatmaps and 2D joint positions estimated for RoI pooling, an L_1 based regression loss is adopted and denoted as $\mathcal{L}_{roi}$. Overall, the objective in the IUV estimation stage involves two main losses:

$$\mathcal{L}_{inter} = \lambda_{iuv}\mathcal{L}_{iuv} + \lambda_{roi}\mathcal{L}_{roi}, \tag{3}$$

where λ_{iuv} and λ_{roi} are used to balance the two terms.

4.2 Camera, Shape and Pose Prediction

After obtaining the global and partial IUV maps, the camera and shape parameters would be predicted in the global stream, while pose parameters would be predicted in the local streams.

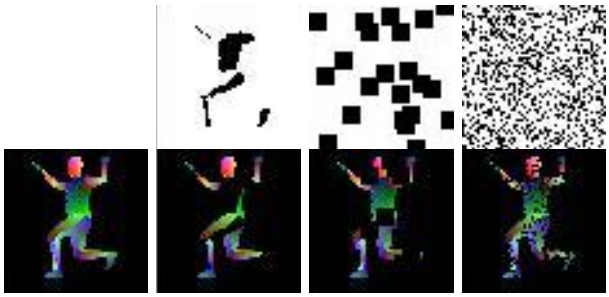
The global stream consists of a ResNet [67] as the backbone network and a fully connected layer added at the end with 13 outputs, corresponding to the camera scale $s \in \mathbb{R}$, translation $\boldsymbol{t} \in \mathbb{R}^2$ and the shape parameters $\boldsymbol{\beta} \in \mathbb{R}^{10}$. In the local streams, a tailored ResNet acts as the backbone network shared by all body joints and is followed by K residual layers for rotation feature extraction individually. For the k-th body joint, the extracted rotation features would be refined (see Sec. 4.3) and then used to predict the rotation matrix $\boldsymbol{R}_k \in \mathbb{R}^{3\times3}$ via a fully connected layer. Here, we follow previous work [6], [7] to predict the rotation matrix representation of the pose parameters $\boldsymbol{\theta}$ rather than the axis-angle representation defined in the SMPL model. Note that using other rotation representations such as the 6D continuous representation [68] is also feasible. An L_1 loss is imposed on the predicted camera, shape, and pose parameter, and we denote it as $\mathcal{L}_{smpl}$.

Following previous work [5], [6], [7], we also add additional constraint and regression objective for better performance. For the predicted rotation matrix, we impose an orthogonal constraint loss $\mathcal{L}_{orth} = \sum_{k=0}^{K-1} \left\| \boldsymbol{R}_k \boldsymbol{R}_k^{\mathrm{T}} - \boldsymbol{I} \right\|_2$ upon the predicted rotation matrices $\{\boldsymbol{R}_k\}_{k=0}^{K-1}$ to guarantee their orthogonality. Moreover, given the predicted SMPL parameters, the performance could be further improved by adding supervision explicitly on the resulting model $\mathcal{M}(\boldsymbol{\theta}, \boldsymbol{\beta})$. Specifically, three L_1 based loss functions are used to measure the difference between the ground-truth positions and the predicted ones. The corresponding losses are denoted as $\mathcal{L}_{vert}$ for vertices on 3D mesh, $\mathcal{L}_{3Dkp}$ for sparse 3D human keypoints, and $\mathcal{L}_{reproj}$ for the reprojected 2D human keypoints, respectively. For the sparse 3D human keypoints, the predicted positions are obtained via a pretrained linear regressor by mapping the mesh vertices to the 3D keypoints defined in human pose datasets. Overall, the objective in this prediction stage is the weighted sum of multiple losses:

$$\begin{aligned}\mathcal{L}_{target} = {} & \lambda_{smpl}\mathcal{L}_{smpl} + \lambda_{orth}\mathcal{L}_{orth} \\ & + \lambda_{point}\left(\mathcal{L}_{vert} + \mathcal{L}_{3Dkp} + \mathcal{L}_{reproj}\right),\end{aligned} \tag{4}$$

where λ_{smpl}, λ_{orth}, and λ_{point} are balance weights.

Part-based Dropout. Our approach learn the shape and pose from the IUV intermediate representation, which contains dense correspondences of the body parts. Following previous work on data augmentation [61] and model regularization [58], [60], we introduce a Part-based Dropout (PartDrop) strategy to drop out semantic information from intermediate representations during training. PartDrop has a dropping rate γ as the probability of dropping values in the estimated IUV maps. In contrast to other dropping out strategies such as Dropout [58] and DropBlock [60], the proposed PartDrop strategy drops features in contiguous regions at the granularity level of body parts. Specifically, for each training sample, the index subset $\mathcal{I}_{drop}$ of the body



(a) (b) (c) (d)

Fig. 6. Comparison of different dropping out strategy. (a) Original IUV map. (b)(c)(d) PartDrop (ours), DropBlock [60] and Dropout [58] drop IUV values in part-wise, block-wise, and unit-wise manners, respectively. The corresponding binary masks are shown on the top row.

parts to be dropped is randomly selected from $\{1, 2, \ldots, P\}$ with the probability of γ. Then, for both global and partial IUV maps, the estimated IUV values of selected body parts are dropped out by setting corresponding body parts as zeros:

$$\boldsymbol{H}[p, :, :, :] = 0, \text{ for } p \in \mathcal{I}_{drop}, \tag{5}$$

where $\boldsymbol{H}[p, :, :, :]$ denotes IUV maps with the part index of p.

PartDrop is motivated by the observation that the estimated IUV maps on real-world images typically have errors on irregular parts in challenging cases such as heavy occlusions. Fig. 6 visualizes how PartDrop, DropBlock [60], and Dropout [58] drop values in part-wise, block-wise, and unit-wise manners. It can be observed that, Dropout leaves obvious pepper-like artifacts after dropping, DropBlock introduces unwanted square patterns, while PartDrop brings much less visual artifacts in the resulting IUV maps. In comparison with DropBlock and Dropout, the proposed PartDrop can remove semantic information from foreground areas in a more structured manner, which consequently enforces the neural network to learn features from complementary body parts and improves its generalization.

4.3 Rotation Feature Refinement

In our approach, the rotation features extracted in local streams are aggregated to exploit spatial relationships among body joints. As illustrated in Fig. 7(a), the position-aided rotation feature refinement involves three consecutive steps, namely rotation feature collection, position feature refinement, and refined feature conversion. Specifically, the rotation features are first collected into the position feature space where the feature refinement is performed. After that, the rotation feature refinement is accomplished by converting the refined position features back to the rotation feature space. All these three steps are performed by customized graph convolution layers. In particular, we consider the following graph-based convolution layer $\mathcal{G}(\cdot)$ that employs one popular formulation of the Graph Convolution Networks as proposed in Kipf et al. [45].

$$\boldsymbol{Z}_{out} = \mathcal{G}(\boldsymbol{A}, \boldsymbol{Z}_{in}) = \sigma(\hat{\boldsymbol{A}}\boldsymbol{Z}_{in}\boldsymbol{W}), \tag{6}$$

where $\boldsymbol{Z}_{in}$ and $\boldsymbol{Z}_{out}$ are input and output features respectively, $\sigma(\cdot)$ is the activation function, $\boldsymbol{W}$ is the parameters of convolution kernels, $\hat{\boldsymbol{A}}$ denotes the row-normalized matrix of the graph adjacency matrix $\boldsymbol{A}$, i.e., $\hat{\boldsymbol{A}} = \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}}$ if $\boldsymbol{A}$ is a symmetric matrix, and otherwise $\hat{\boldsymbol{A}} = \boldsymbol{D}^{-1}\boldsymbol{A}$, where $\boldsymbol{D}$ is the diagonal node degree matrix of $\boldsymbol{A}$ with $\boldsymbol{D}_{ii} = \sum_j \boldsymbol{A}_{ij}$. For simplicity, we also refer to the graph with adjacency matrix of $\boldsymbol{A}$ as graph $\boldsymbol{A}$.

Step 1: Rotation Feature Collection. Note that the rotation of each body joint could be viewed as sequential data along the kinematic chain. This is inspired by the fact that the human could act in a recurrent manner according to the kinematic tree shown in Fig.7(b). The position of a specific body joint can be calculated from the collection of the relative rotations and bone lengths of those joints belonging to the same kinematic chain. At the feature level, we propose to learn the mapping from rotation feature space to position feature space. To that end, one graph convolution layer is customized to gather information from body joints along the kinematic chain and learn such mapping. Formally, let $\boldsymbol{X} \in \mathbb{R}^{K\times C}$ denote the rotation features extracted from K sets of partial IUV maps with C being the feature dimension. The position features $\boldsymbol{Y} \in \mathbb{R}^{K\times C}$ of K joints is obtained by feeding $\boldsymbol{X}$ to the graph convolution, i.e.,

$$\boldsymbol{Y} = \mathcal{G}(\boldsymbol{A}^{r2p}, \boldsymbol{X}), \tag{7}$$

where the graph with adjacency matrix $\boldsymbol{A}^{r2p}$ is customized as a collection graph for mapping rotation features into the position feature space, in which $\boldsymbol{A}^{r2p}_{ij} = 1$ if the j-th joint is one of the ancestors of the i-th joint along the kinematic chain, and otherwise $\boldsymbol{A}^{r2p}_{ij} = 0$. The adjacency matrix $\boldsymbol{A}^{r2p}$ of the collection graph is depicted in Fig. 7(c).

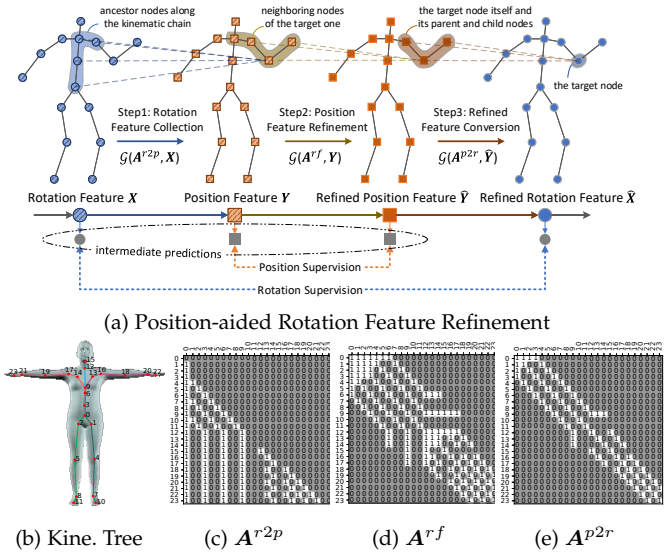
Step 2: Position Feature Refinement. Since there are strong spatial correlations among neighboring body joints, utilizing such structured constraints could effectively improve the features learned at each joint. Towards this goal, a graph-based convolution network is employed to exploit spatial relationships between joints. Specifically, the position features $\boldsymbol{Y}$ are fed into L graph convolution layers with the following layer-wise formulation:

$$\boldsymbol{Y}^{(l)} = \mathcal{G}(\boldsymbol{A}^{rf}, \boldsymbol{Y}^{(l-1)}), \tag{8}$$

where $\boldsymbol{Y}^{l}$ denotes the position features obtained from the l-th layer with $\boldsymbol{Y}^{0} = \boldsymbol{Y}$, and the graph with adjacency matrix $\boldsymbol{A}^{rf} = \boldsymbol{I} + \tilde{\boldsymbol{A}}^{rf}$ serves as a refinement graph for feature refinement, in which $\tilde{\boldsymbol{A}}^{rf}_{ij} = 1$ if the i-th and j-th joints are neighboring, and otherwise $\tilde{\boldsymbol{A}}^{rf}_{ij} = 0$. After graph convolutions, the refined position features $\hat{\boldsymbol{Y}}$ are obtained by adding $\boldsymbol{Y}^{L}$ with the original position features $\boldsymbol{Y}$ in a residual manner, i.e., $\hat{\boldsymbol{Y}} = \boldsymbol{Y} + \boldsymbol{Y}^{L}$. Fig. 7(d) shows an example of the adjacency matrix $\boldsymbol{A}^{rf}$ which considers both one-hop and two-hop neighbors. Note that $\boldsymbol{A}^{rf}$ could have various forms according to the neighbor definition of body joints.

Inspired by previous work [52], [69], we also add a learnable edge weighting mask on the graph convolution of this step since messages from different joints should have different contributions to the feature refinement of the target joint. In this way, we have the adjacency matrix $\boldsymbol{A}^{rf}$ improved as

$$\boldsymbol{A}^{rf} = \boldsymbol{I} + \boldsymbol{M} \circ \tilde{\boldsymbol{A}}^{rf}, \tag{9}$$



(a) Position-aided Rotation Feature Refinement

(b) Kine. Tree (c) $\boldsymbol{A}^{r2p}$ (d) $\boldsymbol{A}^{rf}$ (e) $\boldsymbol{A}^{p2r}$

Fig. 7. Illustration of the aggregated refinement module. (a) Three steps of the proposed refinement strategy. (b) The kinematic tree with $K = 24$ joints in the SMPL model. The pelvis joint with 0 index is the root node of the tree. Joints belonging to the same kinematic chain are linked by the line with the same color. (c)(d)(e) Adjacency matrices of the graphs used in three steps for the feature collection, refinement, and conversion.

where $\circ$ denotes the element-wise product, $\boldsymbol{M} \in [0,1]^{K\times K}$ is the learnable edge weighting matrix serving as an attention mask of the graph to balance the contributions of neighboring features to the target feature.

Step 3: Refined Feature Conversion. The last step of refinement is to convert the refined features back to the original rotation feature space. Since the rotation and position of body joints are two mutual representation of 3D human pose, after the refinement of position features, the rotation features can be refined accordingly[1]. Specifically, for the k-th body joint, its rotation features can be refined by aggregating messages from the refined position features of three consecutive body joints, i.e., the joint itself and its parent and child joints. Similar to the first step, the mapping from position features to rotation features is also learned via a graph-based convolution layer, where the difference lies in the adjacency matrix of the graph. Formally, the refined position features $\hat{\boldsymbol{Y}}$ are fed into the graph to obtain features in the rotation space, resulting in the refined rotation features $\hat{\boldsymbol{X}}$ for the final prediction of joint pose parameters, i.e.,

$$\hat{\boldsymbol{X}} = \mathcal{G}(\boldsymbol{A}^{p2r}, \hat{\boldsymbol{Y}}), \tag{10}$$

where the graph with adjacency matrix $\boldsymbol{A}^{p2r} = \boldsymbol{I} + \tilde{\boldsymbol{A}}^{p2r}$ is customized as a conversion graph for mapping position features to rotation features, in which $\tilde{\boldsymbol{A}}^{p2r}_{ij} = 1$ if the j-th joint is the parent or child joint of the i-th joint, and otherwise $\tilde{\boldsymbol{A}}_{ij} = 0$. The adjacency matrix $\boldsymbol{A}^{p2r}$ of the conversion graph is depicted in Fig. 7(e).

Supervision in Refinement. The rotation and position feature spaces are built under corresponding supervisions during training. As illustrated in Fig. 7(a), the rotation features $\boldsymbol{X}$ and $\hat{\boldsymbol{X}}$ are used to predict joint rotations, while the position features $\boldsymbol{Y}$ and $\hat{\boldsymbol{Y}}$ are used to predict joint positions. L_1 based rotation and position supervisions are imposed on these predictions correspondingly, which compose the objective $\mathcal{L}_{refine}$ involved in the refinement procedure. Note that these intermediate predictions are unnecessary during testing.

1. Strictly speaking, the joint rotations can not be fully retrieved from the joint positions due to the fewer DoFs specified in position-based poses. This issue is mild at the feature level since features could be more redundant.

5 Experiments

5.1 Implementation Details

The FCN for IUV estimation in our framework adopts the architecture of HRNet-W48 [70], which is one of the most recent state-of-the-art networks for dense estimation tasks. The FCN receives the 224×224 input and produces 56×56 feature maps for estimating global and local IUV maps with the same resolution. The IUV estimation network is initialized with the model pretrained on the COCO keypoint detection dataset [71], which is helpful for robust joint-centric RoI pooling and partial IUV estimation. Two ImageNet-pretrained ResNet-18 [67] are employed as the backbone networks for global and rotation feature extraction respectively. During training, data augmentation techniques, including color jittering and flipping, are applied randomly to input images. Random rotation is used when in-the-wild datasets are involved for training. The IUV estimation task is first trained for 5k iterations before involving the parameter prediction task. The α_ks in Eq. (2) are first learned using ground-truth IUV maps as inputs and then frozen as constants for other experiments, while δ is empirically set to 0.1. The hyper-parameters λs are decided based on the scales of values in objectives. The dropping rate γ for PartDrop is adopted as 0.3 in our experiments. For more robust pose recovery from the estimated partial IUV, we perform random jittering on the estimated 2D joint position and the scale of partial IUV maps during training. Following previous work [5], [43], the predicted poses are initialized from the mean pose parameters. For faster runtime, the local streams are implemented to run in a parallel manner. Specifically, the partial IUV maps of all body joints are concatenated batch-wise and then fed into the backbone feature extractor. Moreover, individual rotation feature extraction is implemented based on group convolution. By default, we adopt the Adam [72] optimizer with an initial learning rate of 1×10^{-4} to train our model, and reduce the learning rate to 1×10^{-5} after 30k iterations. The learning process converges after around 60k iterations and takes about 25 hours on a single TITAN Xp GPU. During testing, due to the fundamental depth-scale ambiguity, we follow previous work [5], [7] to center the person within the image and perform scaling such that the inputs have the same setting as training. Our experiments are implemented in PyTorch [73]. More implementation details could be found in the publicly available code.

5.2 Datasets and Evaluation Metrics

Human3.6M. Human3.6M [74] is a large-scale dataset which consists of 3.6 millions of video frames captured in the controlled environment, and currently the most commonly used benchmark dataset for 3D human pose estimation.

Kanazawa et al. [5] generated the ground truth SMPL parameters by applying MoSH [75] to the sparse 3D MoCap marker data. Following the common protocols [5], [6], [33], we use five subjects (S1, S5, S6, S7, S8) for training and two subjects (S9, S11) for evaluation. We also down-sample the original videos from 50fps to 10fps to remove redundant frames, resulting in 312,188 frames for training and 26,859 frames for testing.

UP-3D. UP-3D [3] is a collection dataset of existing 2D human pose datasets (i.e., LSP [76], LSP-extended [77], MPII HumanPose [78], and FashionPose [79]), containing 5,703 images for training, 1,423 images for validation, and 1,389 images for testing. The SMPL parameter annotations of these real-world images are augmented in a semi-automatic way by using an extended version of SMPLify [3].

COCO. The COCO dataset [71] contains a large scale of images and person instances labeled with 17 keypoints. Based on the COCO dataset, DensePose-COCO [39] further provides the dense correspondences from 2D images to the 3D surface of the human body model for 50K humans. Different from our rendered IUV maps, the correspondence annotations in DensePose-COCO only consist of approximately 100-150 points per person, which are a sparse subset of the foreground pixels of human images. In our experiments, we discard those persons without 2D keypoint annotations, resulting in 39,210 samples for training. Since there are no ground-truth shape and pose parameters for COCO, we evaluate our method quantitatively on the keypoint localization task using its validation set, which includes 50,197 samples.

3DPW. The 3DPW dataset [80] is a recent in-the-wild dataset providing accurate shape and pose ground truth annotations. This dataset captured IMU-equipped actors in challenging outdoor scenes with various activities. Following previous work [16], [43], we do not use its data for training but perform evaluations on its defined test set only. There are 35,515 samples extracted from videos for testing.

Fitted SMPL labels from SPIN. Kolotouros et al. [43] proposed SPIN to incorporate a fitting procedure within the training of a SMPL regressor. The regressor provided better initialization for the fitting of human models to 2D keypoints, and the resulting SMPL parameters could be more accurate than those fitted in a static manner. For evaluation on 3DPW [80], our model would be supervised with the final fitted SMPL labels from SPIN [43] for in-the-wild datasets including LSP [76], LSP-Extended [77], MPII [78], COCO [71], and MPI-INF-3DHP [81].

Evaluation Metrics. Following previous work [6], [15], [34], for evaluating the reconstruction performance, we adopt the mean Per-vertex Error (PVE) as the primary metric, which is defined as the average point-to-point Euclidean distance between the predicted model vertices and the ground truth model vertices. Besides the PVE metric, we further adopt PVE-S and PVE-P as secondary metrics for separately evaluate the shape and pose prediction results. The PVE-S computes the per-vertex error with the pose parameters of ground truth and predicted models set as zeros (i.e., models under the rest pose [1]), while the PVE-P computes the analogous per-vertex error with the shape parameters set as zeros. For the Human3.6M dataset, the widely used Mean Per Joint Position Error (MPJPE) and the MPJPE after rigid alignment of the prediction with ground truth using Procrustes Analysis (MPJPE-PA) are also adopted to quantitatively evaluate the 3D human pose estimation performance. The above three metrics will be reported in millimeters (mm) by default.

For the keypoint localization task on COCO, the commonly-used metric is the Average Precision (AP) defined by its organizers[2]. The keypoint localization AP is calculated based on the Object Keypoint Similarity (OKS), which plays the same role as the IoU in object detection. We report results using the mean AP, and the variants of AP including AP_{50} (AP at OKS = 0.50), AP_{75} (AP at OKS = 0.75), AP_M for medium objects, and AP_L for large objects.

5.3 Comparison with State-of-the-art Methods

Table 1
Quantitative comparison with state-of-the-art methods on the Human3.6M dataset.

Method	PVE	MPJPE	MPJPE-PA
Zhou et al. [54]	-	107.3	-
Tung et al. [4]	-	-	98.4
SMPLify [2]	202.0	-	82.3
SMPLify++ [3]	-	-	80.7
Pavlakos et al. [6]	155.5	-	75.9
HMR [5]	-	88.0	56.8
NBF [7]	-	-	59.9
Xiang et al. [38]	-	65.6	-
Arnab et al. [17]	-	77.8	54.3
CMR [40]	-	-	50.1
HoloPose [14]	-	60.3	46.5
TexturePose [19]	-	-	49.7
DenseRaC [41]	-	76.8	48.0
SPIN [43]	-	-	**41.1**
DaNet-LSTM [11]	75.1	61.5	48.6
Ours	**66.5**	**54.6**	42.9

5.3.1 Comparison on the Indoor Dataset.

Evaluation on Human3.6M. We evaluate the 3D human mesh recovery as well as pose estimation performance for quantitative comparison on Human3.6M, where our model is trained on its training set. Table 1 reports the comparison results with previous methods that output more than sparse 3D keypoint positions. For regression-based methods in Table 1, different architectures have been designed to predict the shape and pose parameters. Among them, HMR [5] adopts a single CNN and an iterative regression module to produce all parameters. Pavlakos et al. [6] decompose the shape and pose prediction tasks, while their pose parameters are predicted from 2D joints positions. NBF [7] adopts segmentation as the intermediate representation and learns all parameters from it. CMR [40] directly regresses 3D meshes with a graph-based convolutional network. All these architectures estimate pose parameters through a single stream with an exception that HoloPose [14] regresses poses using a part-based model. As can be seen from Table 1, our network significantly outperforms the above-mentioned architectures. It's worth noting that the methods reported in Table 1 are not strictly comparable since they may use

2. https://cocodataset.org/#keypoints-eval

Table 2
Quantitative comparison of MPJPE-PA across different actions on the Human3.6M dataset.

Method	Direct.	Discuss	Eating	Greet	Phone	Photo	Pose	Purch.	Sitting	SitingD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
Pavlakos et al. [33]	47.5	50.5	48.3	49.3	50.7	55.2	46.1	48.0	61.1	78.1	51.1	48.3	52.9	41.5	46.4	51.9
Martinez et al. [24]	39.5	43.2	46.4	47.0	51.0	56.0	41.4	40.6	56.5	69.4	49.2	45.0	49.5	38.0	43.1	47.7
SMPLify [2]	62.0	60.2	67.8	76.5	92.1	77.0	73.0	75.3	100.3	137.3	83.4	77.3	79.7	86.8	81.7	82.3
HMR [5]	53.2	56.8	50.4	62.4	54.0	72.9	49.4	51.4	57.8	73.7	54.4	50.0	62.6	47.1	55.0	57.2
CMR [40]	41.8	44.8	42.6	46.6	45.9	57.2	40.8	40.6	52.2	66.0	46.6	42.8	51.7	36.9	44.6	48.2
SPIN [43]	37.6	42.4	**38.8**	42.6	**40.4**	**45.9**	36.1	36.7	48.7	58.6	41.2	37.9	46.6	**33.8**	**38.4**	41.1
DaNet-LSTM [11]	43.3	48.8	50.6	48.3	47.3	55.5	41.6	42.7	53.8	61.5	47.4	43.2	53.3	40.8	47.9	48.6
Ours	37.9	44.3	41.2	43.3	42.1	48.7	36.2	38.9	47.4	53.7	41.1	39.9	46.0	34.6	41.3	42.9
Ours-6D	**35.7**	**40.4**	39.0	**40.3**	40.5	47.4	**35.1**	**34.9**	**45.2**	**51.7**	**39.6**	**37.8**	**43.4**	34.4	39.8	**40.5**

Fig. 8. Qualitative comparison of reconstruction results on the UP-3D dataset.

different datasets for training. Among existing state-of-the-art approaches, we have a very competitive result which is only inferior to SPIN in Table 1. SPIN has the same architecture as HMR except that it uses the 6D continuous representation [68] for 3D rotations. SPIN aims to incorporate regression- and optimization-based methods, while our work focuses on the design of a stronger regressor. Hence, our method is complementary to SPIN since we can combine them together by simply plugging our network into SPIN.

For more comprehensive comparison, Table 2 reports pose estimation performance across different actions on Human3.6M. Compared with SPIN and other methods, our method can be more robust to challenging actions such as Sitting and Sitting Down. We believe these benefits come from our decomposition design which enables our network to capture more detailed information for joint poses and produce more accurate reconstruction results. We can also see from the last row of Table 2 that, by simply replacing rotation matrices with the 6D representations [68] for pose parameters as SPIN do, our method can achieve results on par with or even better than SPIN.

Table 3
Quantitative comparison of PVE with state-of-the-art methods on the UP-3D dataset.

Method	LSP	MPII	FashionPose	Full
SMPLify++ [3]	174.4	184.3	108.0	169.8
HMR [5]	-	-	-	149.2
NBF [7]	-	-	-	134.6
Pavlakos et al. [6]	127.8	110.0	106.5	117.7
BodyNet [34]	102.5	-	-	-
Rong et al. [15]	-	-	-	122.2
DaNet-LSTM [11]	90.4	83.0	61.8	83.7
Ours	**88.5**	**82.1**	**60.8**	**82.3**

5.3.2 *Comparison on In-the-wild Datasets*

Reconstructing 3D human model on real-world images is much more challenging due to factors such as extreme poses

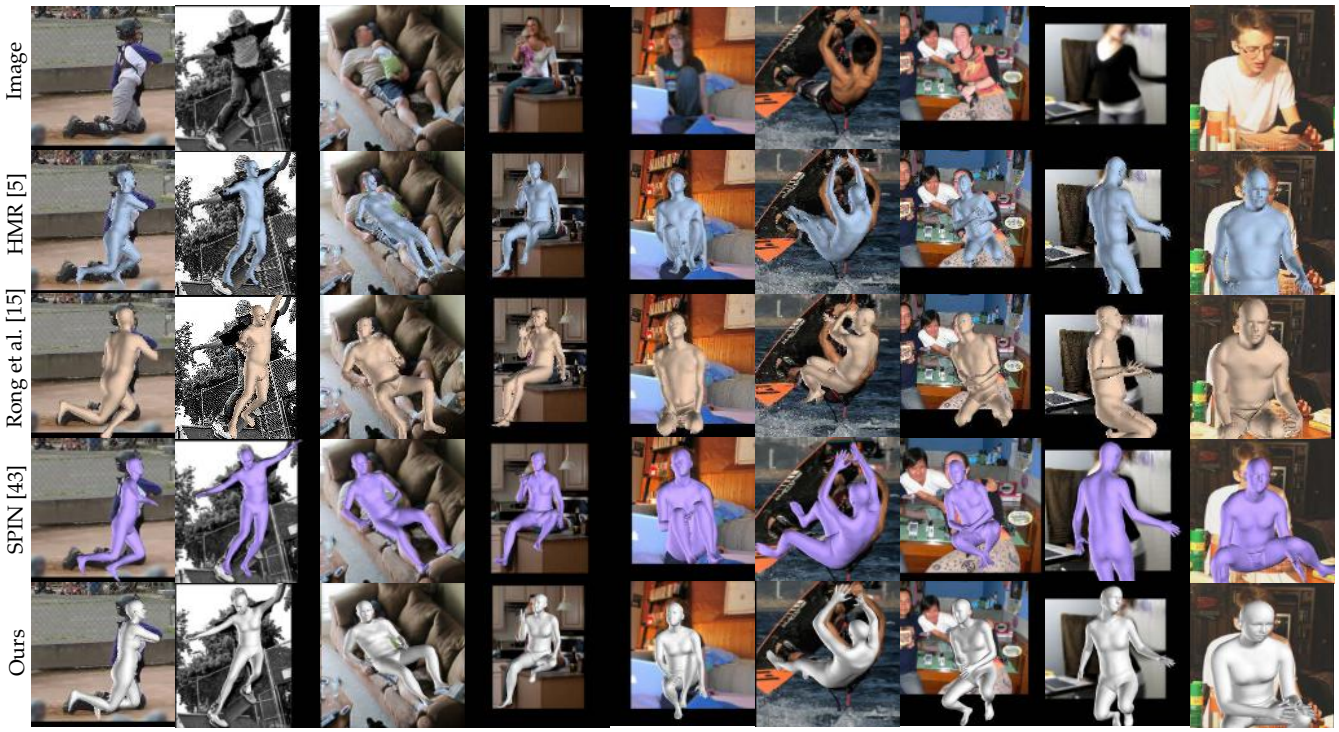


Fig. 9. Qualitative comparison of reconstruction results on the COCO dataset.

Table 4
Quantitative comparison of keypoint localization AP with state-of-the-art methods on the COCO validation set. Results of HMR, CMR, and SPIN are obtained based on their publicly released code and models.

Method	AP	AP_{50}	AP_{75}	AP_M	AP_L
OpenPose [82]	65.3	85.2	71.3	62.2	70.7
SimpleBaseline [83]	74.3	89.6	81.1	70.5	79.7
HRNet [84]	**76.3**	**90.8**	**82.9**	**72.3**	**83.4**
HMR [5]	18.9	47.5	11.7	21.5	17.0
CMR [40]	9.3	26.9	4.2	11.3	8.1
SPIN [43]	17.3	39.1	13.5	19.0	16.6
SPIN-HRNet [43]	21.2	45.3	18.0	22.5	20.9
DaNet-LSTM [11]	28.5	58.7	24.6	30.8	27.1
DaNet-GCN	31.9	65.5	27.5	33.2	31.2
+ Dropout	30.6	64.6	25.7	32.0	30.0
+ DropBlock	32.0	66.9	27.4	33.8	30.9
+ PartDrop (Ours)	**33.8**	**68.6**	**29.9**	**36.0**	**32.3**

and heavy occlusions. In our network, the aggregated refinement module and PartDrop training strategy are proposed to enhance its robustness and generalization. We conduct evaluation experiments on UP-3D, COCO, and 3DPW to demonstrate the efficacy of our method.

Evaluation on UP-3D. For comparison on the UP-3D dataset, we report quantitative results in the PVE of the reconstructed meshes in Table 3. In comparison with previous methods, our method outperforms them across all subsets of UP-3D by a large margin. Our closest competitor BodyNet [34] has the PVE value of 102.5 on LSP, while ours is 88.5. Moreover, BodyNet [34] uses both 2D and 3D estimation as the intermediate representation, which is much more time-consuming than ours. Reconstruction results on UP-3D are visualized in Fig. 8. Compared with other methods, our DaNet could produce more satisfactory results under challenging scenarios.

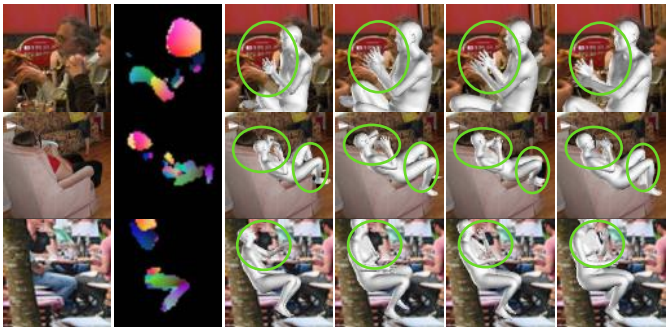


Fig. 10. Comparison of different dropping out strategies in challenging cases. From left to right: input images, estimated IUV maps, results of models trained without dropping, with Dropout, DropBlock, and PartDrop strategies.

Evaluation on COCO. For evaluation on COCO, we train our model on the mixture of training data from DensePose-COCO and Human3.6M datasets, and perform both qualitative and quantitative comparison on the COCO validation set. We first show qualitative reconstruction results in Fig. 9, and make comparisons with HMR [5], Rong et al. [15], and SPIN [43]. As we can see, our method has better generalization in real-world scenarios with more accurate and well-aligned reconstruction performances. Our method can produce reasonable results even in cases of extreme poses, occlusions, and incomplete human bodies, while competitors fail or produce visually displeasing results.

To perform quantitative evaluations on COCO, we project keypoints from the estimated SMPL models on the image plane, and compute the Average Percision (AP) based

Table 5
Quantitative comparison with state-of-the-art methods on the 3DPW dataset.

	Method	PVE	MPJPE	MPJPE-PA
Temporal	Kanazawa et al. [16]	139.3	116.5	72.6
	Doersch et al. [85]	-	-	74.7
	Arnab et al. [17]	-	-	72.2
	Sun et al. [44]	-	-	69.5
	VIBE [20]	113.4	93.5	56.5
Frame-based	HMR [5]	-	130.0	76.7
	CMR [40]	-	-	70.2
	Rong et al. [15]	152.9	-	-
	SPIN [43]	114.8	96.9	59.2
	SPIN-HRNet [43]	112.4	95.4	58.5
	DaNet-LSTM [11]	114.6	92.2	56.9
	Ours	**110.8**	**85.5**	**54.8**

Fig. 11. Qualitative comparison of reconstruction results on the 3DPW dataset.

on the keypoint similarity with the ground truth annotations. We report keypoint localization APs of different approaches in Table 4, where we also include 2D human pose estimation approaches [70], [82], [83] for comparison. It can be seen that, in terms of keypoint localization results, approaches for 3D human mesh recovery lag far behind those for 2D human pose estimation. Among approaches for human mesh recovery, our model achieves significantly higher APs than previous ones. Compared with the recent state-of-the-art method SPIN [43], our model improves the mean AP and AP_{50} by 16.5% and 29.5%, respectively. We attribute such remarkable improvements to our decompose-and-aggregate design. To validate this, we upgrade the backbone of SPIN to HRNet-W64-C [70], a more powerful classification network, and denote it as SPIN-HRNet. As shown in Table 4, though SPIN-HRNet has a stronger backbone with more parameters than our whole network, it brings much less gains over SPIN (3.9% improvement in mean AP from 17.3% to 21.2%). In contrast, our network decomposes the perception tasks and aggregates them efficiently, making our SMPL regressor more effective to handle challenging cases in real-world scenes.

Table 4 also presents the comparison of our approach against our previous model DaNet-LSTM [11]. DaNet-LSTM has the same network architecture with ours except that its aggregation procedure is performed sequentially along kinetic chains via LSTM. Based on DaNet-LSTM, we introduce the graph-based aggregation module and PartDrop strategy in this work. The graph-based aggregation performs feature refinement in parallel for all body parts, while PartDrop regularizes the network and encourages learning features from complementary body parts. These newly introduced designs can help to improve the robustness and generalization of our model. As shown in Table 4, both two new components contribute to higher performance in this challenging dataset. By replacing the LSTM-based aggregation module with the graph-based one, our DaNet-GCN obtains a 6.8% improvement in AP_{50}. By adopting the PartDrop strategy for training, we further have a 3.1% improvement in AP_{50}. Taking these two updates together, our approach improves the AP_{50} by 9.9% over DaNet-LSTM from 58.7% to 68.6%. We can also see from Table 4 that other dropping out strategies such as Dropout and DropBlock do not work well as PartDrop and even degrade the performance. One intuitive explanation for this is that our PartDrop can better imitate the corrupted IUV maps in challenging cases. As we can observe from Fig. 10 that the body parts are missing irregularly from the estimated IUV maps due to occlusions. PartDrop helps to produce more natural and well-aligned results in comparison with its alternatives.

Evaluation on 3DPW. In Table 5, we report the results of our approach and other state-of-the-art approaches on the 3DPW test set. Here, we use the same datasets and training strategy as SPIN [43] and do not use any data from 3DPW for training. Besides, the valid SMPL parameters fitted in SPIN are adopted as ground-truth labels for those in-the-wild training datasets. As shown in Table 5, our approach reduces the MPJPE-PA by 4.4 mm over SPIN to 54.8 mm, achieving the best performance among frame-based and even temporal approaches. Table 5 also includes SPIN-HRNet for comparison, where we can see that there is only a 0.7 mm reduction in MPJPE-PA over SPIN. Fig. 11 depicts the qualitative results of our approach. We can observe that our model has better generalization performances on 3DPW in comparison with SPIN.

5.3.3 Running Time

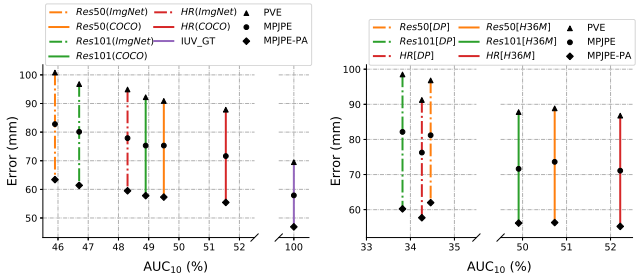
During inference, our method takes about 93ms on a Titan Xp GPU, where the IUV estimation accounts for 60ms while the parameter prediction accounts for the rest 33ms. The running time and platform of different models are included in Table 6 for comparison. Numbers are obtained from respective literature or evaluated using their official implementation. Overall, our method has a moderate computation cost among regression-based reconstruction methods.

Table 6
Comparison of running time (ms) with state-of-the-art models.

Method	Run Time	GPU
HMR [5]	40	GTX 1080 Ti
Pavlakos et al. [6]	50	Titan X
NBF [7]	110	Titan Xp
BodyNet [34]	280	Modern GPU
CMR [40]	33	RTX 2080 Ti
DenseRaC [41]	75	Tesla V100
Ours	93	Titan Xp

Table 7
Performance of approaches adopting different intermediate representations on the Human3.6M dataset.

Method	PVE	MPJPE	MPJPE-PA
ConvFeat	98.9	82.5	60.3
Segmentation	90.4	74.6	57.1
IUV	**87.8**	**71.6**	**55.4**



(a) (b)

Fig. 12. Reconstruction performance on Human3.6M versus the IUV estimation quality for approaches adopting IUV estimators with different architectures and training strategies. (a) Higher IUV estimation qualities generally contribute to better reconstruction performance. IUV estimators are all trained on Human3.6M but initialized with different models. (b) The IUV estimators trained on Human3.6M with dense supervisions have higher IUV estimation qualities. IUV estimators are all pretrained on COCO and then trained on different datasets. Different IUV estimators are denoted as $\dagger(\star)$ or $\dagger[*]$, where $\dagger$ is the architecture, $\star$ and $*$ denote the pretrained and training datasets. IUV_GT stands for taking ground-truth IUV as input. $ImgNet$, DP, and $H36M$ abbreviate ImageNet, DensePose-COCO, and Human3.6M, respectively.

5.4 Ablation Study

To evaluate the effectiveness of the key components proposed in our method, we conduct ablation experiments on Human3.6M under various settings. We will begin with our baseline network by removing the local streams, aggregated refinement module, and PartDrop strategy in our method. In other words, the baseline simply uses the global stream of DaNet to predict all parameters. Moreover, it adopts ResNet101 [67] as the backbone network for parameter predictions such that the model size of the baseline is comparable to that of the networks used in ablation experiments.

5.4.1 Intermediate Representation

To show the superiority of adopting the IUV map as the intermediate representation, our baseline network adopts its alternatives for the shape and pose prediction tasks. Specifically, the IUV maps are replaced by the convolutional feature maps outputted from the last layer of the FCN or the part segmentation (i.e., *Index* channels of IUV maps). Note that there is actually no intermediate representation for the approach adopting feature maps as "intermediate representation". As observed from Table 7, the approach adopting IUV maps as intermediate representations achieves the best performance. In our experiments, we found that the approach without using any intermediate representation is more prone to overfitting to the training set.

Effect of IUV Estimation Quality. We further conduct experiments to investigate the impact of the quality of dense estimation on the final shape and pose prediction performance. To this end, different architectures or initializations of the IUV estimators are adopted in ablation experiments to produce IUV maps with different qualities. Specifically, the IUV estimator adopts the pose estimation networks [83] built upon ResNet-50 and ResNet-101 as alternative architectures, and these models are pretrained on ImageNet [86] or COCO [71]. Following the protocol of DensePose [39], we measure the quality of dense correspondence estimations via the pointwise evaluation [39], where the area under the curve at the threshold of 10cm (i.e., AUC_{10}) is adopted as the metric. Fig. 12(a) reports the reconstruction results of ablation approaches versus their qualities of IUV estimations. As we can see, networks with better IUV estimations consistently achieve better reconstruction performance. To investigate the performance upper bound of adopting IUV maps as intermediate representations, we also report the results of the approach using ground truth IUV maps as input with the removal of the IUV estimator. As shown in the rightmost result of Fig. 12(a), the approach learning from the ground truth IUV maps achieves much better performance than using the estimated one outputted from networks, which means that there is still a large margin for improvement by adopting IUV maps as intermediate representations.

In contrast to the concurrent work [15], [40], [41] obtaining IUV maps from the pretrained network of DensePose [39], our approach augments the annotation of Human3.6M with the rendered IUV maps so that our IUV estimator can be trained on Human3.6M with dense supervision, which enables our network to have a higher quality of IUV estimation. To verify this, the IUV estimator is firstly trained on DensePose-COCO or Human3.6M, and then frozen to generate IUV maps for the training of the reconstruction task on Human3.6M. As can be seen from Fig. 12(b), approaches with the IUV estimators trained on Human3.6M consistently achieve better performances on both IUV estimation and model reconstruction tasks.

5.4.2 Decomposed Perception

The decomposed perception provides fined-grained information for detailed pose estimation. To validate the effectiveness of such a design, we report the performance of the approaches using one-stream and multiple streams in Table 8, where the *D-Net* denotes the variant of our DaNet without using the aggregated refinement module and PartDrop strategy. Results in PVE-S and PVE-P are also reported in Table 8 for separately studying the efficacy of the decomposed design on the shape and pose predictions. It can be seen that the reconstruction performance metric PVE is actually dominated by the PVE-P metric. Comparison of the first and second rows in Table 8 shows that using multiple streams has barely effects on the shape prediction but brings a significant improvement in the pose prediction (i.e., the PVE-P value drops more than 14%). We also report results to validate the use of different ratios α_k and the simplification of partial IUV maps. In the 3rd and 4th rows of Table 8, *D-Net-ES* adopts equal scales with all α_ks set to 0.5, while *D-Net-AP* adopts partial IUV maps with all body parts. As can be seen, such modifications degrade the performance, which is due to two facts that (i) the proportions of body parts are different and (ii) the rotational states of different body joints are relatively independent and

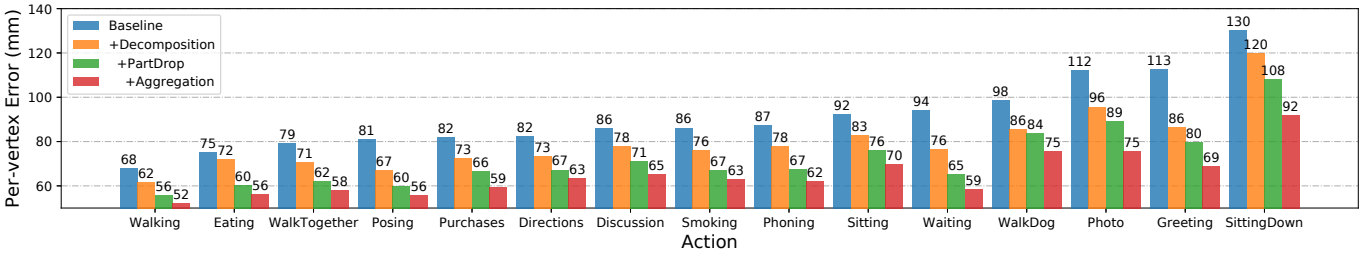


Fig. 13. Reconstruction performance of ablation approaches across different actions on the Human3.6M dataset.

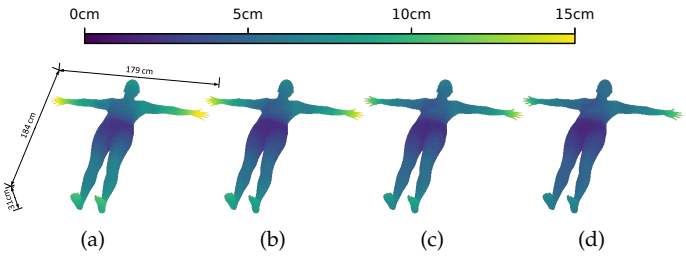


Fig. 14. Comparison of the average per-vertex error upon the model surface for ablation approaches on the Human3.6M dataset. (a) The baseline approach using one stream only. (b) The approach using multiple streams for decomposed perception. (c) The approach using decomposed perception and PartDrop strategies. (d) Our final approach with the aggregated refinement.

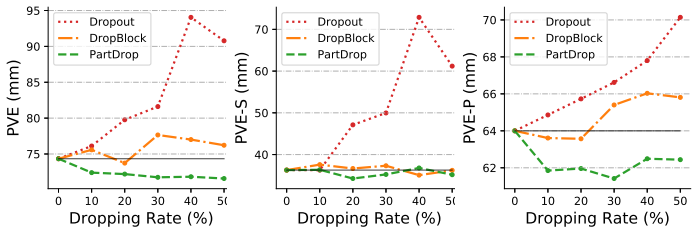


Fig. 15. Comparison of reconstruction performance for approaches using different dropping out strategies on the Human3.6M dataset. (a)(b)(c) report results with metrics of PVE, PVE-S, and PVE-P to reveal the quality of the full model recovery, shape recovery, and pose recovery across different dropping rates, respectively.

Table 8
Performance of approaches using different perception strategies on the Human3.6M dataset.

Method	PVE	PVE-S	PVE-P	MPJPE	MPJPE-PA
Baseline	87.8	38.0	76.3	71.6	55.4
D-Net	**74.3**	**36.3**	**64.0**	**61.8**	**48.5**
D-Net-ES	76.1	36.6	65.5	63.1	49.8
D-Net-AP	76.8	36.8	65.8	63.4	49.5

involving irrelevant body parts could disturb the inference of the target joint rotations.

To visualize the reconstruction performance on different body areas, Fig. 14 depicts the average per-vertex error with respect to the surface areas of the human model. As shown in Fig. 14(a), for the baseline network, the per-vertex errors of limb parts (hands, feet) are much higher than that of the torso. By comparing Figs. 14(a) and 14(b), we can conclude that our decomposed perception design alleviates the above issue and achieves much better reconstruction performance on limb parts. Reconstruction performances across different actions on Human3.6M are also reported in Fig. 13 for comprehensive evaluations. We can see that the decomposed perception design reduces reconstruction errors consistently for all actions.

5.4.3 *Part-based Dropout*

The proposed Part-based Dropout (PartDrop) strategy drops IUV values in contiguous regions at the granularity level of body parts. Such a dropping out strategy can effectively regularize the neural network by removing semantic information from foreground areas of intermediate representations. In this subsection, we conduct experiments to validate its effectiveness and evaluate the impact of the dropping rate on the reconstruction performance.

To validate the superiority of our PartDrop strategy, we adopt DropBlock [60] and Dropout [58] as alternative strategies to drop values from intermediate representations during training. For DropBlock, following the setting of [60], the size of the block to be dropped is set to 7 in our experiments. For fair comparison, only the foreground pixels are involved in counting the dropping rate. Fig. 15 reports the performance of the full model reconstruction as well as its shape and pose components under different strategies across different dropping rates. It can be seen that the performance gains brought by dropping out strategies mainly come from the pose prediction tasks since the evaluation metric PVE is dominated by its pose component PVE-P. Among three strategies, Dropout is the worst and its performance deteriorates quickly when increasing the rate of dropping out. DropBlock works better than Dropout and brings marginal gains when the dropping rate is less than 20%. Though we can see from the PVE-S curves in Fig. 15(b) that DropBlock has comparable results with PartDrop on shape prediction when the dropping rate is larger than 40%, its pose prediction results degrade significantly as shown in Fig. 15(c). We hypothesize that the removal of a large area of block makes DropBlock similar to PartDrop for the global perception but does harm to the local perception for pose prediction. Compared with these two alternatives, the proposed PartDrop is more robust to the dropping rate and achieves the best results at a dropping rate around 30%. The above comparison of unit-wise, block-wise, and part-wise dropping strategies suggest that removing features in a structured manner is crucial to our reconstruction task, where PartDrop performs best among them. The efficacy of

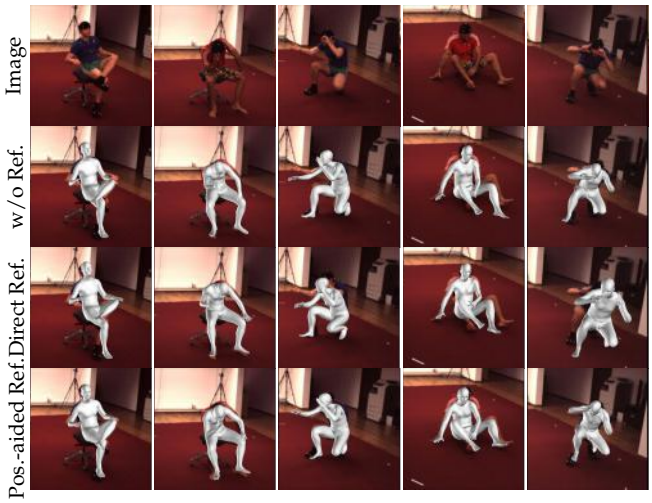


Fig. 16. Example results of approaches without refinement, or using direct / position-aided refinement strategies.

Table 9
Performance of approaches using different feature refinement strategies on the Human3.6M dataset.

Refinement Strategy	PVE	MPJPE	MPJPE-PA
w/o Ref.	71.7	59.1	46.1
Direct Ref.	70.3	58.1	45.5
Pos.-implicit Ref.	69.2	56.5	44.7
Pos.-aided Ref.	**66.5**	**54.6**	**42.9**

PartDrop can be also validated from the reconstruction error reduction shown in Fig. 13 and Fig. 14(c).

5.4.4 Aggregated Refinement

Our aggregated refinement module is proposed to impose spacial structure constraints upon rotation-based pose features. As observed from Fig. 14(d) and Fig. 13, the aggregation in DaNet effectively reduces the reconstruction errors across all surface areas and human actions considerably.

A straightforward strategy to refine the feature would be conducting refinement between the rotation features directly. In such a *direct* refinement strategy, the first and third steps of our refinement procedure are removed and the rotation features are directly refined by the graph convolution layers of the second step. The features outputted from the last refinement layer are also added with the original rotation features in a residual manner and then used to predict joint rotations. For fair comparison, the refinement layer number of the direct strategy is equal to the number of the layers involved in the three steps of the position-aided strategy.

Rotation Feature Space vs. Position Feature Space. The proposed position-aided refinement strategy performs refinement in the position feature space instead of the rotation feature space. The graphs $\boldsymbol{A}^{r2p}$ and $\boldsymbol{A}^{p2r}$ of the first and last refinement steps are customized to connect the rotation and position feature spaces. The graph $\boldsymbol{A}^{r2p}$ collects rotation features to the position feature space, while the graph $\boldsymbol{A}^{p2r}$ converts position features back to the rotation feature space. To validate their functions, we discard position supervisions from the objective $\mathcal{L}_{refine}$ during refinement. We refer to this strategy as the *position-implicit* refinement strategy since the position feature space is built in an implicit manner. The only difference between the direct and position-implicit refinement strategies is that, in the latter one, there are two mapping operations performed before and after the refinement. We report the results of the approaches using direct, position-implicit, position-aided strategies in Table 9 for comparison. It can be seen that the position-implicit strategy achieves inferior results than the position-aided strategy but better results than the direct strategy, which means that the implicit position space still works better than the rotation space for feature refinement. Example results of the approach using the direct or position-aided refinement strategy are also depicted in Fig. 16 for comparison. We can see that the position-aided refinement helps to handle challenging cases and produce more realistic and well-aligned results, while the direct refinement brings marginal to no improvement.

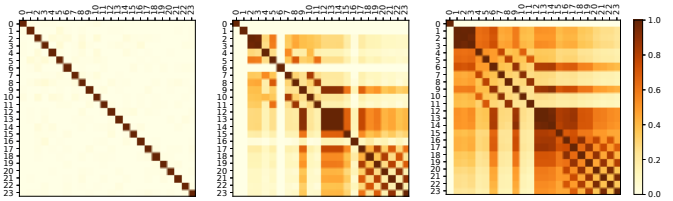


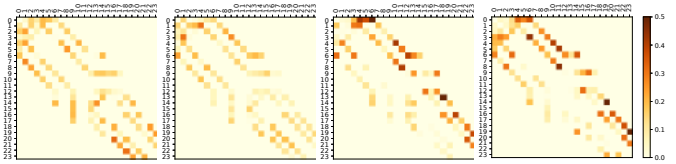
Fig. 17. Correlation matrices of the features extracted from (a) rotation, (b) implicit position, and (c) position feature spaces.

The reason behind the inferior performance of the direct refinement is that the correlation between rotation features is weak, and the messages of neighboring rotation features are generally irrelevant to refine the target rotation feature. Our refinement module builds an auxiliary position feature space for feature refinement, making it much more efficient than that in the original rotation feature space. To verify this, we extract the features before refinement from the rotation, implicit position, and position spaces of the three strategies mentioned above, and compute the correlations between features of different body joints. Fig. 17 shows the comparison of correlation matrices of these three types of features. As observed from Fig. 17(a), the correlation matrix of rotation features approximates to an identity matrix, meaning that the correlations between the rotation features of different joints are rather weak even for two adjacent joints. By contrast, for implicit position features in Fig. 17(b) and position features in Fig. 17(c), the correlations between features of adjacent joints are much higher, making it more feasible to refine features with the messages from neighboring joints.

Benefit from Learnable Graph Edge and PartDrop. The learnable edge weighting matrix $\boldsymbol{M}$ of the refinement graph contributes to better balancing the importance of neighboring messages, while the PartDrop strategy helps to encourage the network to leverage more information from neighboring joints. To verify their effectiveness during feature refinement, Table 10 reports the results of the ablation approaches incrementally adopting the learnable edge in the refinement graph and the PartDrop strategy,

Table 10
Ablation study of using learnable graph edge and PartDrop strategies on the Human3.6M dataset.

Method	PVE	MPJPE	MPJPE-PA
D-Net	74.3	61.8	48.5
+ PartDrop	71.7	59.1	46.1
D-Net+Direct	72.1	59.4	46.9
+ LearntEdge	72.7	59.6	47.0
+ PartDrop	70.3	58.1	45.5
D-Net+Pos.-aided	70.8	57.1	45.9
+ LearntEdge	68.9	55.8	44.9
+ PartDrop	**66.5**	**54.6**	**42.9**



(a) (b) (c) (d)

Fig. 18. Visualization of learned edge weighting matrices under different training settings. (a)(b) Direct refinement without and with PartDrop. (c)(d) Position-aided refinement without and with PartDrop.

where *D-Net+Direct* and *D-Net+Pos.-aided* adopt the refinement module with the direct and position-aided strategy, respectively. It can be seen that, for the direct refinement, the performance gains mainly come from the PartDrop strategy. In contrast, for the position-aided refinement, the performance gains are attributed to both the learnable edge and the PartDrop strategy. Fig. 18 depicts the learned edge weighting matrices of different ablation approaches. As observed, the learned edge weighting matrices of the direct refinement are relatively flat with lower values. When using the PartDrop strategy, the learnable values of most edges in the refinement graph rise for the position-aided refinement, while such a phenomenon is not observed for the direct refinement. We conjecture that the PartDrop strategy brings gains from two perspectives. First, PartDrop regularizes the backbone feature extractor to focus on more complementary regions in intermediate representations for better feature exploitation. Second, PartDrop encourages the refinement module to borrow more information from neighbors in the position feature space for better feature refinement.

6 Conclusion

In this work, a Decompose-and-aggregate Network is proposed to learn 3D human shape and pose from dense correspondences of body parts with the decomposed perception, aggregated refinement, and part-based dropout strategies. All these new designs contribute to better part-based learning and effectively improve the reconstruction performance by providing well-suited part perception, leveraging spatial relationships for part pose refinement, and encouraging the exploitation of complementary body parts. Extensive experiments have been conducted to validate the efficacy of key components in our method. In comparison with previous ones, our network can produce more accurate results, while being robust to extreme poses, heavy occlusions, and incomplete human bodies, etc. In future work, we may explore integrating dense refinement [14] to further improve the shape and pose recovery results.

Acknowledgments

The authors would like to thank the associate editor and reviewers for their helpful comments to improve this manuscript. This work was supported in part by the National Natural Science Foundation of China (Grant No. U1836217, 61806197) and the National Key Research and Development Program of China (Grant No. 2017YFC0821602). This work was done when H. Zhang visited the University of Sydney with the support of the Joint Ph.D. Training Program of the University of Chinese Academy of Sciences.

References

[1] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM Transactions on Graphics*, vol. 34, no. 6, pp. 248:1–248:16, 2015.

[2] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image," in *Proceedings of the European Conference on Computer Vision*. Cham: Springer, 2016, pp. 561–578.

[3] C. Lassner, J. Romero, M. Kiefel, F. Bogo, M. J. Black, and P. V. Gehler, "Unite the people: Closing the loop between 3d and 2d human representations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6050–6059.

[4] H.-Y. Tung, H.-W. Tung, E. Yumer, and K. Fragkiadaki, "Self-supervised learning of motion capture," in *Advances in Neural Information Processing Systems*, 2017, pp. 5236–5246.

[5] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7122–7131.

[6] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis, "Learning to estimate 3d human pose and shape from a single color image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 459–468.

[7] M. Omran, C. Lassner, G. Pons-Moll, P. Gehler, and B. Schiele, "Neural body fitting: Unifying deep learning and model based human pose and shape estimation," in *International Conference on 3D Vision*. IEEE, 2018, pp. 484–494.

[8] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "Scape: shape completion and animation of people," in *ACM Transactions on Graphics*, vol. 24, no. 3. ACM, 2005, pp. 408–416.

[9] X. Chen and A. L. Yuille, "Articulated pose estimation by a graphical model with image dependent pairwise relations," in *Advances in Neural Information Processing Systems*, 2014, pp. 1736–1744.

[10] X. Chu, W. Ouyang, H. Li, and X. Wang, "Structured feature learning for pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4715–4723.

[11] H. Zhang, J. Cao, G. Lu, W. Ouyang, and Z. Sun, "Danet: Decompose-and-aggregate network for 3d human shape and pose estimation," in *Proceedings of the 27th ACM International Conference on Multimedia*. ACM, 2019, pp. 935–944.

[12] L. Sigal, A. Balan, and M. J. Black, "Combined discriminative and generative articulated pose and non-rigid shape estimation," in *Advances in Neural Information Processing Systems*, 2008, pp. 1337–1344.

[13] P. Guan, A. Weiss, A. O. Balan, and M. J. Black, "Estimating human shape and pose from a single image," in *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 2009, pp. 1381–1388.

[14] R. A. Guler and I. Kokkinos, "Holopose: Holistic 3d human reconstruction in-the-wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 884–10 894.

[15] Y. Rong, Z. Liu, C. Li, K. Cao, and C. C. Loy, "Delving deep into hybrid annotations for 3d human recovery in the wild," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5340–5348.

[16] A. Kanazawa, J. Y. Zhang, P. Felsen, and J. Malik, "Learning 3d human dynamics from video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5614–5623.

[17] A. Arnab, C. Doersch, and A. Zisserman, "Exploiting temporal context for 3d human pose estimation in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3395–3404.

[18] J. Liang and M. C. Lin, "Shape-aware human pose and shape reconstruction using multi-view images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4352–4362.

[19] G. Pavlakos, N. Kolotouros, and K. Daniilidis, "Texturepose: Supervising human mesh estimation with texture consistency," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 803–812.

[20] M. Kocabas, N. Athanasiou, and M. J. Black, "Vibe: Video inference for human body pose and shape estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5253–5263.

[21] H. Joo, T. Simon, and Y. Sheikh, "Total capture: A 3d deformation model for tracking faces, hands, and bodies," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8320–8329.

[22] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 975–10 985.

[23] H. Zhu, X. Zuo, S. Wang, X. Cao, and R. Yang, "Detailed human shape estimation from a single image by hierarchical mesh deformation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4491–4500.

[24] J. Martinez, R. Hossain, J. Romero, and J. J. Little, "A simple yet effective baseline for 3d human pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2640–2649.

[25] B. X. Nie, P. Wei, and S.-C. Zhu, "Monocular 3d human pose estimation by predicting depth on joints," in *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 2017, pp. 3467–3475.

[26] F. Moreno-Noguer, "3d human pose estimation from a single image via distance matrix regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2823–2832.

[27] K. Lee, I. Lee, and S. Lee, "Propagating lstm: 3d pose estimation based on joint interdependency," in *Proceedings of the European Conference on Computer Vision*. Cham: Springer, 2018, pp. 119–135.

[28] E. Dibra, H. Jain, C. Öztireli, R. Ziegler, and M. Gross, "Hs-nets: Estimating human body shape from silhouettes with convolutional neural networks," in *International Conference on 3D Vision*. IEEE, 2016, pp. 108–117.

[29] B. M. Smith, V. Chari, A. Agrawal, J. M. Rehg, and R. Sever, "Towards accurate 3d human body reconstruction from silhouettes," in *International Conference on 3D Vision*. IEEE, 2019, pp. 279–288.

[30] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman *et al.*, "Efficient human pose estimation from single depth images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2821–2840, 2013.

[31] V. Gabeur, J.-S. Franco, X. Martin, C. Schmid, and G. Rogez, "Moulding humans: Non-parametric 3d human shape estimation from single images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 2232–2241.

[32] B. Tekin, P. Márquez-Neila, M. Salzmann, and P. Fua, "Learning to fuse 2d and 3d image cues for monocular body pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3941–3950.

[33] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Coarse-to-fine volumetric prediction for single-image 3d human pose," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7025–7034.

[34] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid, "Bodynet: Volumetric inference of 3d human body shapes," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 20–36.

[35] A. S. Jackson, C. Manafas, and G. Tzimiropoulos, "3d human body reconstruction from a single image via volumetric regression," in *Proceedings of the European Conference on Computer Vision*, 2018.

[36] Z. Zheng, T. Yu, Y. Wei, Q. Dai, and Y. Liu, "Deephuman: 3d human reconstruction from a single image," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7739–7749.

[37] C. Luo, X. Chu, and A. L. Yuille, "Orinet: A fully convolutional network for 3d human pose estimation," in *British Machine Vision Conference*, 2018, p. 92.

[38] D. Xiang, H. Joo, and Y. Sheikh, "Monocular total capture: Posing face, body, and hands in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 965–10 974.

[39] R. Alp Güler, N. Neverova, and I. Kokkinos, "Densepose: Dense human pose estimation in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7297–7306.

[40] N. Kolotouros, G. Pavlakos, and K. Daniilidis, "Convolutional mesh regression for single-image human shape reconstruction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4501–4510.

[41] Y. Xu, S.-C. Zhu, and T. Tung, "Denserac: Joint 3d pose and shape estimation by dense render-and-compare," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7760–7770.

[42] V. Tan, I. Budvytis, and R. Cipolla, "Indirect deep structured learning for 3d human body shape and pose prediction," in *British Machine Vision Conference*, 2017, pp. 1–11.

[43] N. Kolotouros, G. Pavlakos, M. J. Black, and K. Daniilidis, "Learning to reconstruct 3d human pose and shape via model-fitting in the loop," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 2252–2261.

[44] Y. Sun, Y. Ye, W. Liu, W. Gao, Y. Fu, and T. Mei, "Human mesh recovery from monocular images via a skeleton-disentangled representation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5349–5358.

[45] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations*, 2017, pp. 1–14.

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

[47] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Poselet conditioned pictorial structures," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 588–595.

[48] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 1385–1392.

[49] S. Zuffi and M. J. Black, "The stitched puppet: A graphical model of 3d human shape and pose," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3537–3546.

[50] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," in *Advances in Neural Information Processing Systems*, 2014, pp. 1799–1807.

[51] H.-S. Fang, Y. Xu, W. Wang, X. Liu, and S.-C. Zhu, "Learning pose grammar to encode human body configuration for 3d pose estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, pp. 6821–6828.

[52] L. Zhao, X. Peng, Y. Tian, M. Kapadia, and D. N. Metaxas, "Semantic graph convolutional networks for 3d human pose regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3425–3435.

[53] I. Akhter and M. J. Black, "Pose-conditioned joint angle limits for 3d human pose reconstruction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1446–1455.

[54] X. Zhou, X. Sun, W. Zhang, S. Liang, and Y. Wei, "Deep kinematic pose regression," in *Proceedings of the European Conference on Computer Vision*. Cham: Springer, 2016, pp. 186–201.

[55] X. Sun, J. Shang, S. Liang, and Y. Wei, "Compositional human pose regression," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2602–2611.

[56] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei, "Towards 3d human pose estimation in the wild: a weakly-supervised approach," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 398–407.

[57] W. Yang, W. Ouyang, X. Wang, J. Ren, H. Li, and X. Wang, "3d human pose estimation in the wild by adversarial learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5255–5264.

[58] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[59] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 648–656.

[60] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 727–10 737.

[61] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[62] R. Alp Guler, G. Trigeorgis, E. Antonakos, P. Snape, S. Zafeiriou, and I. Kokkinos, "Densereg: Fully convolutional dense shape regression in-the-wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6799–6808.

[63] M. M. Loper and M. J. Black, "Opendr: An approximate differentiable renderer," in *Proceedings of the European Conference on Computer Vision*. Cham: Springer, 2014, pp. 154–169.

[64] H. Kato, Y. Ushiku, and T. Harada, "Neural 3d mesh renderer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3907–3916.

[65] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 2017–2025.

[66] X. Sun, B. Xiao, F. Wei, S. Liang, and Y. Wei, "Integral human pose regression," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 529–545.

[67] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[68] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.

[69] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, pp. 7444–7452.

[70] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[71] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proceedings of the European Conference on Computer Vision*. Cham: Springer, 2014, pp. 740–755.

[72] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 2015.

[73] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.

[74] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1325–1339, 2014.

[75] M. Loper, N. Mahmood, and M. J. Black, "Mosh: Motion and shape capture from sparse markers," *ACM Transactions on Graphics*, vol. 33, no. 6, p. 220, 2014.

[76] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation." in *British Machine Vision Conference*, 2010, pp. 12.1–12.11.

[77] ——, "Learning effective human pose estimation from inaccurate annotation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 1465–1472.

[78] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3686–3693.

[79] M. Dantone, J. Gall, C. Leistner, and L. Van Gool, "Body parts dependent joint regressors for human pose estimation in still images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2131–2143, 2014.

[80] T. von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering accurate 3d human pose in the wild using imus and a moving camera," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 601–617.

[81] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt, "Monocular 3d human pose estimation in the wild using improved cnn supervision," in *International Conference on 3D Vision*. IEEE, 2017, pp. 506–516.

[82] Z. Cao, G. H. Martinez, T. Simon, S.-E. Wei, and Y. A. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[83] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 466–481.

[84] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5693–5703.

[85] C. Doersch and A. Zisserman, "Sim2real transfer learning for 3d human pose estimation: motion to the rescue," in *Advances in Neural Information Processing Systems*, 2019, pp. 12 949–12 961.

[86] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.