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 a monocular image 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 the body model 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 vertexes, 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 adjacent position features. The effectiveness of our method is validated on both in-door and real-world datasets including the Human3.6M, UP3D, and DensePose-COCO datasets. Experimental results show that the proposed method significantly improves the reconstruction performance in comparison with previous state-of-the-art methods. Our code will be made 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. Traditional 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, learning-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 UUV 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 vertexes 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.

The representation of 3D body model [1], [8] can be factorized into the shape and pose parameters, depicting the model at different scales. The shape parameters give an overall description about the model such as the height and weight, while the pose

Fig. 1. Illustration of our main ideas. (a) A human image with the reconstructed 3D model. The rotation-based pose representation of the body model is prone to result in drift of joint positions. (b) Comparison of the raw RGB image, silhouette, segmentation, and IUV map. (c) Local visual cues are crucial for joint rotation status perception. (d) Our DaNet learns 3D human shape and pose from IUV maps with decomposed perception and aggregated refinement.

• H. Zhang, J. Cao, and Z. Sun are with CRIPAC, NLPR, CAS CEBSIT, Institute of Automation, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, Beijing, China. E-mail: hongwen.zhang@cripac.ie.ac.cn; jie.cao@cripac.ie.ac.cn; zn.sun@nlpr.ie.ac.cn. (Corresponding author: Zhenan Sun.)
• G. Lu is with Shanghai Jiao Tong University, Shanghai, China. E-mail: luguo2014@sjtu.edu.cn.
• W. Ouyang is with the University of Sydney, Sydney, Australia. E-mail: wanli.ouyang@sydney.edu.au.
A part-based dropout strategy is introduced to drop dense information from intermediate representations for the task of 3D human pose and shape estimation. Previous learning-based methods 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 rotation status 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, 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 works, the structural information at the feature level is helpful for more robust and accurate pose estimation results. 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 in consideration that the prediction of different parameters requires the receptive fields with different sizes. To robustly predict the rotation status of body joints, DaNet aggregates messages from different streams and refines the rotation feature via an auxiliary position feature space to exploit the spatial relationship between body joints. For better generalization performances, 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 adjacent body joints.

To sum up, the main contributions in this work are listed as follows.

- This work comprehensively studies 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 position features of different body joints. 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 performances and the customized graph convolutions for faster and better feature mapping and refinement. Second, more extensive evaluation and comparison 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 works 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 the 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 attempts 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], and detailed or holistic body model learning [20, 21, 22].

#### 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 3D human pose estimation, 2D joint postions [23, 24, 25, 26], volumetric representation [27], joint heat maps [28],...
and 3D orientation fields \cite{29} are adopted in literature as intermediate representations to facilitate the learning task. Similarly, for 3D human model reconstruction, silhouette \cite{6, 30, 31}, joint heatmap \cite{4, 6}, segmentation \cite{7}, depth map \cite{32}, 3D volumetric representation \cite{33, 34, 35}, and 3D orientation field \cite{36} have also been adopted in existing methods as proxy representations. 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 the subsequent prediction. Recently, DensePose \cite{37} 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 dense 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 \cite{15, 38, 39} to investigate learning 3D human shape and pose from IUV maps via CNN. In comparison, the major differences between concurrent efforts and ours lie in three aspects: 1) \cite{15, 38, 39} obtain IUV predictions from a pretrained network of DensePose \cite{37}, 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) \cite{15, 38, 39} only leverage global IUV maps, while our work exploits using IUV maps in both global and local scales; 3) DenseRaC \cite{39} 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. \cite{40} 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 fixed decoder. Kanazawa et al. \cite{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. \cite{41} enhance HMR with the fitting process of SMPLify \cite{2} to incorporate regression- and optimization-based methods. Pavlakos et al. \cite{6} propose to predict the shape and pose parameters from the estimated silhouettes and joint locations respectively. Sun et al. \cite{42} 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. \cite{38} employ a Graph CNN \cite{43} to regress the 3D coordinates of the human mesh vertices, while Yao et al. \cite{44} regress the 3D coordinates in the form of an unwrapped position map. All aforementioned learning-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, Gler et al. \cite{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 accurate human pose estimation \cite{45, 46}. Recent deep learning-based approaches to human pose estimation \cite{9, 10, 47, 48, 49} 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 approach is complementary to them by investigating the refinement strategy for rotation features under the context of rotation-based pose representation. We further show that the spatial relationship between body joints is a good intermediate space for refining the rotation features. Our approach aggregates the rotation features into the 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 \cite{50, 51, 52, 53, 54} are also employed as constraints in the learning procedure. For instance, Akhter and Black \cite{50} learn the pose prior in the form of joint angle constraints. Sun et al. \cite{52} design handcrafted constraints such as limb-lengths and their proportions. Similar constraints are exploited in \cite{53} under the strongly-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 \cite{2} imposes several penalizing terms on predicted poses to prevent unnatural results. Kanazawa et al. \cite{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 \cite{55} is commonly used at the fully connected layers of neural networks to drop unit-wise features independently. The introduction of dropout has inspired the development of dropping out strategies with structured forms. For instance, SpatialDropout \cite{56} drops channel-wise features across the entire feature map, while DropBlock \cite{57} 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 \cite{55}.

3 SMPL Model and IUV Maps

SMPL Model. The Skinned Multi-Person Linear model (SMPL) \cite{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 $\beta \in \mathbb{R}^{10}$ denotes the coefficients of the PCA basis of body shape. The pose parameters $\theta \in \mathbb{R}^{3K}$ denotes 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 $\mathcal{M}(\theta, \beta) \in \mathbb{R}^{3 \times N}$. The deformation process $\mathcal{M}(\theta, \beta)$ is differentiable with respect to the pose $\theta$ and shape $\beta$, 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 pixels on 2D images and vertexes on 3D surfaces. Such correspondence 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 vertexes 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 $I$ of the body parts [37], [59], which results in the IUV maps $H = (H^i|H^u|H^v) \in \mathbb{R}^{p+1 \times h_{iuv} \times w_{iuv} \times 3}$, where $p$ denotes the number of body parts, $h_{iuv}$ and $w_{iuv}$ denote the height and width of IUV maps. $H^i$ indicates whether a pixel belongs to the background or a specific body part, while $H^u$ and $H^v$ contain the corresponding $U$, $V$ values of visible body parts respectively. For each body part, the UV space is independent so that the representation could be more fine-grained. Such a IUV annotation of the human body is firstly introduced in DenseReg [59] and DensePose [37]. Figs. 2(a)(b)(c) show the Index, $U$, and $V$ values on the SMPL model as defined in DensePose [37].

**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 adding the corresponding ground-truth IUV maps based on the same IUV mapping protocol of DensePose [37]. 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 $(u, v, i)$ denoting the corresponding $U$, $V$ and $I$ values. Then, given SMPL models, the corresponding IUV maps can be obtained from existing rendering algorithms such as [60], [61]. 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 $(p + 1) \times h_{iuv} \times w_{iuv} \times 3$ by converting values into one-hot representations.

### 4 Methodology

As illustrated in Fig. 2(e) our DaNet decomposes the prediction task into a global stream for the camera and shape prediction and multiple local streams for joint pose prediction. The overall pipeline involves two consecutive stages, where the IUV maps are firstly estimated from the fully convolution network and then taken as inputs for subsequent parameter prediction.

In the first stage, the IUV maps are estimated from global and local perspectives in consideration of the different sizes of the receptive fields required by the prediction of different parameters. In the second stage, the global and local IUV maps are used for separate tasks. The global IUV maps are used for extracting global features, which are directly used to predict camera and body shape. The partial IUV maps are used for extracting the rotation features, which are further refined and then used to predict joint poses.

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

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

where $\mathcal{L}_{\text{inter}}$ is the objective for estimating the intermediate representations (Sec. 4.1), $\mathcal{L}_{\text{target}}$ is the objective for predicting the camera and SMPL parameters (Sec. 4.2), $\mathcal{L}_{\text{refine}}$ is the
objective involving in the feature refinement procedure (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. 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, the joint-centric RoI pooling would first be applied on these feature maps to extract an appropriate sub-region, which results in partial feature maps. Then, the \( K \) sets of partial IUV maps would be estimated independently from the resulting \( K \) sets of partial feature maps.

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 factor for the perception of joint pose 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, sub-regions of the feature maps are 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) \([62]\). In comparison with the conventional STNs, the pooling process in our network is learned in an explicitly supervised manner.

As illustrated in Fig. 3(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 \([63]\) operation. Without loss of generality, let \( j_k \) 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 \( c_k \) is the position of the \( k \)-th joint, while the scale \( s_k \) is proportional to the size of the foreground region, i.e.,

\[
\begin{align*}
    c_k &= j_k, \\
    s_k &= \alpha_k \max(w_{bbox}, h_{bbox}) + \delta,
\end{align*}
\]

where \( \alpha_k \) and \( \delta \) 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 \( s_k \)s determined by foreground regions here are more robust to 2D joint localization.

Note that the above constants \( \alpha_k \) and \( \delta \) can be handcrafted or learned in the STN by taking ground-truth IUV maps as inputs. For learned \( \alpha_k \)s, Fig. 3(b) shows how the values of different body joints evolve over learning iterations. It can be observed that \( \alpha_k \)s 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.

Simplified Partial IUV Maps. Considering that the pose of a body joint is only related to its adjacent body parts, we further introduce the simplified 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. By doing so, the resulting simplified partial IUV maps are much cleaner, which eliminates interference from irrelevant body parts. 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 \( L_{IUV} \). Specifically, a classification loss is imposed on the index \( I \) channels of IUV maps, where the \( K + 1 \)-way cross-entropy loss is employed to classify a pixel belonging to either background or one among the \( K \) 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 \( I \) 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 \( L_{roi} \). Overall, the objective in the IUV estimation stage involves
two main losses:

\[ L_{\text{inter}} = \lambda_{\text{iuw}} L_{\text{iuw}} + \lambda_{\text{roi}} L_{\text{roi}}, \]  

where \( \lambda_{\text{iuw}} \) and \( \lambda_{\text{roi}} \) 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 [64] 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 \( t \in \mathbb{R}^2 \) and the shape parameters \( \beta \in \mathbb{R}^{10} \). 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 \( R_k \in \mathbb{R}^{3 \times 3} \) via a fully connected layer. Here, we follow previous work [6, 7] to predict the rotation matrix representation of the pose parameters \( \theta \) rather than the axis-angle representation defined in the SMPL model. An \( L_1 \) loss is imposed on the predicted camera, shape and pose parameter, and we denote it as \( L_{\text{smpl}} \).

Following previous work [5, 6, 7], we also add additional constraint and regression objective for better performance. For the predicted rotation matrix, it is necessary to make it lie on the manifold of rotation matrices. In our method, we impose an orthogonal constraint loss on the predicted rotation matrix to guarantee its orthogonality. The orthogonal constraint loss for predicted rotation matrices \( \{R_k\}_{k=1}^K \) is denoted as \( L_{\text{orth}} \) and could be written as

\[ L_{\text{orth}} = \sum_{k=1}^K \left\| R_k R_k^T - I \right\|_2. \]  

Given the predicted SMPL parameters, the performance could be further improved by adding supervision explicitly on the resulting model \( M(\theta, \beta) \). Specifically, we use three \( L_1 \) based loss functions to measure the difference between the ground-truth and the predicted ones. The corresponding losses are denoted as \( L_{\text{verx}} \) for vertices on 3D mesh, \( L_{\text{iuv}} \) for sparse 3D human keypoints and \( L_{\text{reproj}} \) 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:

\[ L_{\text{target}} = \lambda_{\text{smpl}} L_{\text{smpl}} + \lambda_{\text{orth}} L_{\text{orth}} + \lambda_{\text{point}} (L_{\text{verx}} + L_{\text{iuv}} + L_{\text{ADKp}} + L_{\text{reproj}}), \]

where \( \lambda_{\text{smpl}} \), \( \lambda_{\text{orth}} \), and \( \lambda_{\text{point}} \) 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. Inspired by previous work on data augmentation [58] and model regularization [55, 57], we propose a Part-based Dropout (PartDrop) strategy to drop out semantic information from the intermediate representation during training. PartDrop has a dropping rate \( \gamma \) as the probability of dropping values in the estimated IUV maps. In contrast to other dropping out strategies such as Dropout [55] and DropBlock [57], the proposed PartDrop strategy drops features in contiguous regions at the granularity level of body parts. Specifically, for each training sample, the index subset \( I_{\text{drop}} \) of the body parts to be dropped is randomly selected from \( \{1, 2, \ldots, p\} \) with the probability of \( \gamma \). 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:

\[ H[c, :, :] = 0, \quad \text{for } c \in I_{\text{drop}}, \]

where \( H[c, :, :] \) denotes IUV maps with the part index of \( c \).

Fig. 5 visualizes how PartDrop, DropBlock [57], and Dropout [55] drop values in part-wise, block-wise, and unit-wise manners. As observed, in comparison with DropBlock and Dropout, the proposed PartDrop can remove semantic information 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, a position-aided rotation feature refinement strategy is proposed to exploit spatial relationships among body joints. As illustrated in Fig. 6(a), the rotation refinement procedure includes three consecutive steps, namely rotation feature to position feature mapping, position feature refinement, and refined feature aggregation. Specifically, the rotation features are first aggregated and converted to the position feature space where the feature refinement is performed. After that, the rotation feature refinement is accomplished by aggregating the messages from the refined position features. All these three steps are implemented using graph-based convolutional networks. In particular, we consider the following graph-based convolution layer \( G(\cdot) \) that employs one popular variant of the GCN formulation as proposed in Kipf et al. [43]:

\[ Z_{\text{out}} = G(A, Z_{\text{in}}) = \sigma(\tilde{A} Z_{\text{in}} W), \]

where \( Z_{\text{in}} \) and \( Z_{\text{out}} \) are input and output features respectively, \( \sigma(\cdot) \) is the activation function, \( W \) is the parameters of convolution kernels, \( \tilde{A} \) denotes the row-normalized adjacency matrix \( A \) of the graph, i.e., \( \tilde{A} = \tilde{D}^{-\frac{1}{2}} A \tilde{D}^{-\frac{1}{2}} \) if \( A \) is a symmetric matrix, and otherwise \( \tilde{A} = \tilde{D}^{-1} A \), where \( \tilde{D} \) is the diagonal node degree matrix of \( A \) with \( \tilde{D}_{ii} = \sum_j A_{ij} \).

**Step 1: Rotation Feature to Position Feature Mapping.** 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. 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 employed to learn such mapping, which aims at gathering information from body joints along the kinematic chain. Formally, let \( X \in \mathbb{R}^{K \times C} \) denote the rotation features extracted from \( K \) sets of partial IUV maps with \( C \) being the feature dimension. The position feature \( Y \in \mathbb{R}^{K \times C} \) of \( K \) joints is obtained by feeding \( X \) to the graph convolution, i.e.,

\[
Y = G(A^{2p}, X),
\]

where \( A^{2p} \) denotes the adjacency matrix of the graph for gathering rotation features to position features, in which \( A_{ij}^{2p} = 1 \) if the \( j \)-th joint is one of the ancestors of the \( i \)-th joint along the kinematic chain, and otherwise \( A_{ij}^{2p} = 0 \). The adjacency matrix \( A^{2p} \) is depicted in Fig. 6(c).

**Step 2: Position Feature Refinement.** Since there are strong correlations among spatially adjacent body joints, utilizing this spatial relationship could effectively improve features learned at each joint. Towards this goal, a graph-based convolution network is utilized to exploit spatial relationships between joints. Specifically, the position feature \( Y \) is fed into \( L \) graph convolution layers with the following layer-wise formulation:

\[
Y^{(l)} = G(A^{lf}, Y^{(l-1)}),
\]

where \( Y^{l} \) denotes the position feature obtained from the \( l \)-th layer with \( Y^{0} = Y \), and \( A^{lf} = I + A^{lf} \) is the adjacency matrix of the graph for feature refinement, in which \( A_{ij}^{lf} = 1 \) if the \( i \)-th and \( j \)-th joints are spatially adjacent, and otherwise \( A_{ij}^{lf} = 0 \). After graph convolutions, the refined position features \( \hat{Y} \) are obtained by adding \( Y^{L} \) with the original position feature \( Y \) in a residual manner, i.e., \( \hat{Y} = Y + Y^{L} \). Fig. 6(d) shows an example of the adjacency matrix \( A^{lf} \), which considers both one-hop and two-hop neighbors. Note that \( A^{lf} \) could have various forms according to the neighbor definition of body joints.

Inspired by previous work [49, 65], we also add a learnable edge importance weighting mask on the graph convolution of this step considering that messages from different joints have different contributions to the feature refinement of the target joint. In this way, we have the adjacency matrix in Eq. 9 improved as

\[
A^{lf} = I + M^{(l)} \odot \tilde{A}^{lf},
\]

where \( M^{(l)} \in [0, 1]^{K \times K} \) is the layer-specific trainable weighting matrix serving as an attention mask of the graph to balance the contributions of joints’ features to their neighboring joints.

**Step 3: Refined Feature Aggregation.** The last step of refinement is to project the feature 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 feature, the rotation feature can be refined accordingly. Specifically, for the \( k \)-th body joint, its rotation features can be refined by aggregating messages from the refined position feature 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 feature \( \hat{Y} \) is fed into the graph to obtain feature in the rotation space, resulting in the refined rotation features \( \hat{X} \) for the final prediction of joint pose parameters, i.e.,

\[
\hat{X} = G(A^{p2r}, \hat{Y}),
\]

where \( A^{p2r} = I + \hat{A}^{p2r} \) is the adjacency matrix of the graph for gathering position feature to rotation feature, in which \( A_{ij}^{p2r} = 1 \) if the \( j \)-th joint is the parent or child joint of the \( i \)-th joint, and otherwise \( A_{ij} = 0 \). The adjacency matrix \( A^{p2r} \) is depicted in Fig. 6(e).

**Supervision in Refinement.** The rotation and position feature spaces are built under corresponding supervisions during training. As illustrated in Fig. 6(a), the rotation features \( X \) and \( \hat{Y} \) are used to predict joint rotations, while the position features \( Y \) and \( \hat{Y} \) are used to predict joint positions. \( L1 \) based rotation and position supervisions are imposed on these predictions correspondingly, which compose the objective \( \ell_{refine} \) involved in the refinement procedure. Note that these intermediate predictions are unnecessary during testing.

### 5 Experiments

#### 5.1 Implementation Details

The FCN for IUV estimation in our framework adopts the architecture of HRNet-W48 [66], which is one of the most recent state-of-the-art networks for dense prediction tasks. The FCN receives the 224 \( \times \) 224 input and produces 56 \( \times \) 56 feature maps for estimating the global and local IUV maps, which have the same resolution of 56 \( \times \) 56. Two ResNet-18 [53] are employed as the backbone networks for global and rotation feature extraction respectively. 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 [67] and run with a TITAN Xp GPU. More details could be found in the supplementary material and publicly available code.
5.2 Datasets and Evaluation Metrics

Human3.6M. Human3.6M [68] 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 [69] to the sparse 3D MoCap marker data. Following the common protocols [5], [6], [27], 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 [70], LSP-extended [71], MPII HumanPose [72], and FashionPose [73]), containing 5703 images for training, 1423 images for validation, and 1389 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].

DensePose-COCO. DensePose-COCO [37] provides the dense correspondences from 2D images to the 3D surface of the human body model for 50K humans appearing in the COCO dataset [74]. 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 and 7,297 samples for evaluation.

Evaluation Metrics. Following previous work [6], [15], [33], 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 ground truth and predicted models’ pose parameters 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 wildly 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 metrics will be reported in millimeters (mm) by default.

5.3 Comparison with State-of-the-art Methods

Comparison on the In-door Dataset. We evaluate the reconstruction as well as 3D human pose estimation performance for quantitative comparison on Human3.6M. Table 1 reports the comparison results with previous methods that output more than sparse 3D keypoint positions. 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 [38] directly regresses 3D shapes with a graph-based convolutional network. All these methods except [14] estimate pose parameters through a single stream and our method outperforms them significantly. Concurrent work [14] predicts pose parameters using a part-based model and has similar results with ours.

| Method        | PVE   | MPJPE | MPJPE-PA |
|---------------|-------|-------|----------|
| Zhou et al.   | 51    | 107.3 | -        |
| Tung et al.   | -     | -     | 98.4     |
| SMPLify       | 202.0 | 82.3  |          |
| SMPLify++     | -     | -     | 80.7     |
| Pavlakos et al. | 155.5 | 75.9  |          |
| HMR           | -     | 88.0  | 56.8     |
| NBF           | -     | 59.9  |          |
| Xiang et al.  | -     | 65.6  |          |
| Arnab et al.  | -     | 77.8  | 54.3     |
| CMR           | -     | -     | 50.1     |
| HoloPose      | -     | 64.3  | 50.6     |
| TexturePose   | -     | -     | 49.7     |
| DenseRaC      | -     | 76.8  | 48.0     |
| DaNet-LSTM    | 75.1  | 61.5  | 48.6     |
| Ours          | 66.5  | 54.6  | 42.9     |

Comparison on In-the-wild Datasets. Reconstructing 3D human model on real world images is much more challenge due to factors such as extreme poses and heavy occlusions. We conduct evaluation experiments on UP-3D and DensePose-COCO to demonstrate the robustness of our method.

For evaluation on UP-3D, we report quantitative results in the PVE of the reconstructed meshes in Table 2. In comparison with previous methods, our method outperforms them across all subsets of UP-3D by a large margin. Our closest competitor BodyNet [33] has the PVE value of 102.5 on LSP, while ours is 88.5. Moreover, BodyNet [33] 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. 7. Compared with other methods, our DaNet could produce more satisfactory results under challenging scenarios, which could be attributed to the proposed aggregation design for rotation feature refinement.

For evaluation on DensePose-COCO, our model is trained on the mixture of training data from both DensePose-COCO and Human3.6M datasets. We only apply PartDrop on samples of Human3.6M considering that PartDrop somewhat imitates the commonly occurred occlusions in samples of DensePose-COCO. Since there is no ground truth human model provided in DensePose-COCO, we only perform qualitative evaluations on this dataset. We show reconstruction examples in Fig. 8 and make comparisons with HMR [5] and Rong et al. [15]. It can be observed that our method has better generalization in real-world scenarios with more accurate and 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.

Running Time. During inference, our method takes about 93ms on a Titan Xp GPU, while the IVU estimation accounts for 60ms while the parameter prediction accounts for the rest 33ms. The running time and platform of other state-of-the-art methods are included in Table 3. Numbers are obtained from respective literature or evaluated using their official implementation. Overall, our method has a moderate computation cost among learning-based reconstruction methods.
Fig. 7. Qualitative comparisons of reconstruction results on the UP-3D dataset.

Table 2
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 [33]         | 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  |

Table 3
Comparison of running time (ms) with state-of-the-art methods.

| Method               | Run Time | GPU          |
|----------------------|----------|--------------|
| HMR [5]              | 40       | GTX 1080 Ti  |
| Pavlakos et al. [6]  | 50       | Titan X      |
| NBF [7]              | 110      | Titan Xp     |
| BodyNet [33]         | 280      | Modern GPU   |
| CMR [38]             | 33       | RTX 2080 Ti  |
| DenseRC [39]         | 75       | Tesla V100   |
| Ours                 | 93       | Titan Xp     |

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 and refinement module in our method. In other words, the baseline uses only the global stream of DaNet to predict all parameters. Moreover, it adopts ResNet101 [64] in the global stream for parameter prediction such that the model size of the baseline is comparable to that of the networks used in the following experiments.

5.4.1 Intermediate Representation

The IUV map acts as a bridge between pixels on 2D images and vertexes on 3D meshes, which facilitates the learning task of the shape and pose prediction network. To show its superiority, we use our baseline network and adopt alternative intermediate representations for the shape and pose prediction tasks. Specifically, the IUV maps are replaced by the 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 4, the approach adopting IUV maps as intermediate representations achieves the best performances. Compared with alternative representations, adopting IUV maps reduces the PVE value from 98.9 of feature maps and 90.4 of part segmentation to 87.8. 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 prediction 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 [75] built upon ResNet-50 and ResNet-101 as alternative architectures, and these models are pretrained on ImageNet [76] or COCO [74]. Following the protocol of DensePose [57], we measure the quality of dense correspondence predictions via the pointwise evaluation [57], where the area
under the curve (AUC) at the threshold of 10cm is adopted as the metric. Fig. 9 reports the reconstruction results of ablation approaches versus their qualities of dense predictions. As can be seen, networks with better dense predictions consistently achieve better reconstruction performance. Moreover, networks initialized with the weight pretrained on COCO always leads to better dense prediction results and reconstruction performances. This could be explained by the fact that better initialization is essential to alleviate the overfitting issue on Human3.6M. To investigate the performance upper bound of adopting IUV maps as intermediate representations, we also report results of the approach using the ground truth IUV maps as input. As shown in the rightmost result of Fig. 9(a), the approach learning from the ground truth IUV maps achieves a 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 the intermediate representation.

In contrast to the concurrent work [15], [38] obtaining IUV maps from the pretrained network of DensePose [37], 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. 9(b) models with the IUV estimators trained on Human3.6M consistently achieve better performances on both IUV estimation and reconstruction tasks.

5.4.2 Decomposed Perception

The decomposed perception provides fine-grained information for detailed pose estimation. To validate the effectiveness of such a design, we report the performances of the approaches using one-stream and multiple streams in Table 5 where the D-Net denotes the variant of our DaNet without using the refinement module and PartDrop strategy. Results in PVE-S and PVE-P are also reported in Table 5 for individually 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 5 shows that using multiple streams has barely effects on the shape prediction but brings a significant improvement on the pose prediction (i.e., the PVE-P value drops more than 14%). We also report results to validate the use of different ratios αk and the simplification of partial IUV maps. In the 3rd and 4th rows
of Table 5, D-Net-ES adopts equal scales with all $\alpha_k$s set to 0.5, while D-Net-AP adopts partial IUV maps with all body parts. It can be seen that such modifications degrade the performances, which is due to two facts that (i) the proportions of body parts are different and (ii) the pose status of different body joints is relatively independent and involving irrelevant body parts could disturb the inference of the target joint poses.

To visualize the reconstruction performance on different body areas, Fig. 11 depicts the average per-vertex error with respect to the surface position of the human model. As shown in Fig. 11(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. 11(a) and 11(b), we can conclude that our decomposed perception design alleviates the above issue and achieve much better reconstruction performance on limb parts. Reconstruction performances across different actions on Human3.6M are also reported in Fig. 10 for comprehensive comparisons. We can see that the decomposed perception design reduces reconstruction errors consistently for all actions.

| 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     |

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 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 [57] and Dropout [55] as alternative strategies to drop values from intermediate representations during training. For DropBlock, following the setting of [57], the size of the block to be dropped is set to 7 in our experiments. Fig. 12 reports the shape and pose reconstruction results of different strategies across different dropping rates. For fair comparisons, only the foreground pixels are involved in counting the dropping rate. It can be seen that the performance gains brought by dropping out strategies mainly come from the pose prediction tasks. 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 30%. The proposed PartDrop overwhelmingly outperforms its counterparts, and achieves the best results at the dropping rate around 30%. The above comparisons 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 PartDrop could be also validated from the reconstruction error reduction shown in Fig. 10 and Fig. 11(c).

![Fig. 11. Comparison of the average per-vertex error upon the model surface for ablation approaches on the Human3.6M dataset.](image)

![Fig. 10. Reconstruction performance of ablation approaches across different actions on the Human3.6M dataset.](image)

![Fig. 12. Comparison of reconstruction performance for approaches using different dropping out strategies on the Human3.6M dataset.](image)
5.4.4 Position-aided Rotation Feature Refinement

Our refinement module is proposed to impose spatial structure constraints upon rotation-based pose features. As observed from Fig. 11(d) and Fig. 10, 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, we remove the first and third steps of our refinement procedure and refine the rotation features directly using the graph convolution layers of the second step. The features outputted from the last graph convolution layers are also added with the original rotation features in a residual manner and then used to predict joint rotation status. For fair comparisons, 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 adjacent matrices $A^{r2p}$ and $A^{p2r}$ of the first and last graphs are customized as mapping matrices to connect the rotation and position feature spaces. The matrix $A^{r2p}$ aggregates rotation features to the position feature space, while the matrix $A^{p2r}$ maps the position features back to the rotation feature space. To validate their functions, we discard position supervisions from the objective $L_{\text{refine}}$ 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 10 for comparisons. 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 approaches using the direct or position-aided refinement strategy are also depicted in Fig. 13 for comparisons. 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.

The reason behind the inferior performances of the direct refinement is that the correlation between rotation features is weak, and the messages of adjacent rotation features are generally irrelevant to refine the target rotation feature. Our aggregation 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, and compute the correlation between features of different body joints. Fig. 14 shows the comparison of correlation matrices of these three types of features. As observed from Fig. 14(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. 14(b) and position features in Fig. 14(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 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

| 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     |

Table 6 Performance of approaches using different feature refinement strategies on the Human3.6M dataset.
information from neighboring joints. To verify their effectiveness during feature refinement, Table 7 reports the results of the ablation approaches incrementally adopting the learnable edge in the refinement graph and the PartDrop strategy. 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. 15 depicts the learned edge importance matrices of different ablation approaches. As observed, the learned edge importance 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 aggregation module to borrow more information from neighbors in the position feature space for better feature refinement.

6 Conclusion

In this work, we investigate learning 3D human shape and pose from dense correspondences of body parts. A Decompose-and-aggregate Network (DaNet) is proposed to adopt IUV maps as the intermediate representation and predict the shape and pose parameters in a decomposition-followed-by-aggregation manner. In DaNet, the decomposed streams enable the network to provide global perception for the camera and shape prediction and fine-grained perception for pose prediction of each body joint. The position-aided aggregation strategy makes it more efficient to exploit the spatial relationship between body joints in the position feature space, since the correlations between position features are stronger than that in the original rotation feature space. A Part-based Dropout (PartDrop) strategy is introduced to encourage the network to leverage more complementary information from the intermediate representations as well as the position feature space. Extensive experiments have been conducted to validate the efficacy of our method.

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, p. 248, 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 European Conference on Computer Vision. 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. 6030–6039.

[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. Li, 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] 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.

[21] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Trzoniak, 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.

[22] 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.

[23] 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.
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.

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.

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, 2018, pp. 119–135.

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.

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.

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.

E. Dibra, H. Jain, C. Öztirelie, 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.

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.

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.

D. 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.

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.

Z. Zheng, T. Yu, 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.

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. 10965–10974.

R. Alp Guler, 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.

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.

Y. Xu, S.-C. Zhu, and T. Tung, “Denserc: 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.

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.

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.

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.

T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in International Conference on Learning Representations, 2017.

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.

L. Fischedich, M. Andriluka, P. Gehler, and B. Schiele, “Poselet conditioned joint prior structures,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 588–595.

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.
Appendix

Training Details

During training, data augmentation techniques, including rotation ±30°, color jittering (±30% channel-wise) and flipping, are applied randomly to input images. The FCN is initialized with the model pre-trained on the COCO keypoint detection dataset [74] for 2D human pose estimation, which is essential for robust 2D joint position localization and partial IUV estimation. The α_k,s in Eq. (2) are learned using ground-truth IUV maps as inputs, while δ is empirically set to 0.1. The hyper-parameters λ 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 prediction from the estimated partial IUV, we perform random jittering on the estimated 2D joint position (±5%) and the scale of partial IUV map (±10%) during training. The IUV estimation task is first trained for 5k iterations before involving the parameter prediction task. We adopt the ADAM [77] optimizer with an initial learning rate of $1 \times 10^{-4}$ to train the model, and reduce the learning rate to $1 \times 10^{-5}$ after 30k iterations. The learning process converges after around 60k iterations. 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. The training process takes about 25 hours on a single TITAN Xp GPU.

Additional Quantitative Comparison

Comparisons on SURREAL. SURREAL [78] is a large scale dataset containing synthetic images with the ground truth labels of SMPL parameters. The accurate ground truth labels make the SURREAL dataset ideal for quantitative evaluations of reconstruction performances. The original dataset provides 1,964 video sequences of 115 subjects for training, and 703 video sequences of 30 subjects for testing. We downsample the video to select one out of every three frames to remove redundancy, resulting in 1.7 million frames as training data. Since the training samples are much abundant in this dataset, we double the number of training iterations by reducing the learning rate after 60k iterations and finishing training after 120k iterations. For testing, we follow the protocol in [33] to select 507 video clips from the test sequences as a sub-set, in which the middle frame of each clip is chosen as testing data. Those frames containing no human body or incomplete human bodies are further discarded, which makes total testing samples of 465 frames.

Table 8 shows the evaluation results of Per-vertex Error (PVE) on the SURREAL dataset. It can be seen that our method outperforms previous methods. Our closest competitor BodyNet [33] is much more time-consuming than ours. Their method predicts volumetric representations of the human body with both 2D and 3D estimation as intermediate representations.

Additional Ablation Study

Effect of Image Features. The IUV maps are the re-projection of the 3D human body models on image planes, containing highly abstract information about the shape and pose of models while eliminating other irrelevant factors. Compared with IUV maps, the image features contain the raw information extracted from RGB images. To evaluate how important the image features are to
our task, we concatenate the image features with the IUV maps and feed them as input for the shape and pose prediction. Our experiments found that image features are not always helpful, which depends on the initialization of networks and the quality of IUV maps. As shown in Table 9, concatenating the image features of the network pretrained on COCO could result in slightly better performances. However, image features are not helpful and even degrade the performance when the network is trained from scratch. The degraded performances are also observed in case of concatenating image features with the ground-truth IUV maps, although the network is pretrained on COCO. Based on these observations, we do not resort to using image feature in our approach and leave it for future work to improve the quality of the estimated IUV maps.

Table 9
Performance of approaches with/without concatenated image features as intermediate representations on the Human3.6 dataset.

| Pre-trained | Input       | PVE  | MPJPE | MPJPE-PA |
|-------------|-------------|------|-------|----------|
| None        | IUV         | 103.5| 85.8  | 64.5     |
|             | IUV+Feat    | 120.7| 100.1 | 73.1     |
| ImageNet    | IUV         | 94.9 | 77.9  | 59.5     |
|             | IUV+Feat    | 94.2 | 77.7  | 56.8     |
| COCO        | IUV         | 87.8 | 71.6  | 55.4     |
|             | IUV+Feat    | 88.9 | 71.4  | 52.2     |
| COCO        | IUV_GT      | 69.5 | 57.9  | 46.9     |
|             | IUV_GT+Feat | 83.5 | 67.4  | 49.9     |

Fig. 16. Comparison of direct and position-aided refinement strategies across different numbers of refinement layers.

Table 10
Performance of approaches using different feature refinement strategies on the Human3.6M dataset.

| Refinement Strategy | PVE  | MPJPE | MPJPE-PA |
|---------------------|------|-------|----------|
| w/o Refinement      | 71.7 | 59.1  | 46.1     |
| LSTM-based          |      |       |          |
| Direct              | 71.9 | 59.0  | 46.0     |
| Position-aided      | 68.4 | 57.0  | 45.1     |
| GCN-based (1-hop)   |      |       |          |
| Direct              | 69.9 | 57.7  | 45.3     |
| Position-aided      | 66.7 | 54.4  | 43.6     |
| GCN-based (1- & 2-hop) |      |       |          |
| Direct              | 70.3 | 58.1  | 45.5     |
| Position-aided      | 66.5 | 54.6  | 42.9     |

Fig. 17 depicts more example results of our method including visualization under side viewpoints. Moreover, failure cases of our method as well as the estimated IUV are also visualized in Fig. 18. These failure cases are typically caused by extreme poses, heavy occlusion/self-occlusion, and poor IUV estimation.
Fig. 17. Successful results of our method. For each example from left to right: Image, Our reconstruction result, Our reconstruction result from a side view. All images come from the DensePose-COCO dataset.
Fig. 18. Erroneous reconstructions of our method. For each example from left to right: Image, estimated IUV, Our reconstruction result, Our reconstruction result from a side view. All images come from the DensePose-COCO dataset.