

Deep3DPose: Realtime Reconstruction of Arbitrarily Posed Human Bodies from Single RGB Images

Liguo Jiang, Miaopeng Li, Jianjie Zhang, Congyi Wang, Juntao Ye, Xinguo Liu, and Jinxiang Chai

Abstract—We introduce an approach that accurately reconstructs 3D human poses and detailed 3D full-body geometric models from single images in realtime. The key idea of our approach is a novel end-to-end multi-task deep learning framework that uses single images to predict five outputs simultaneously: foreground segmentation mask, 2D joints positions, semantic body partitions, 3D part orientations and *uv* coordinates (*uv* map). The multi-task network architecture not only generates more visual cues for reconstruction, but also makes each individual prediction more accurate. The CNN regressor is further combined with an optimization based algorithm for accurate kinematic pose reconstruction and full-body shape modeling. We show that the realtime reconstruction reaches accurate fitting that has not been seen before, especially for wild images. We demonstrate the results of our realtime 3D pose and human body reconstruction system on various challenging in-the-wild videos. We show the system advances the frontier of 3D human body and pose reconstruction from single images by quantitative evaluations and comparisons with state-of-the-art methods.

Index Terms—Realtime RGB-based motion capture, multi-task regression, 3D human body and shape reconstruction.

1 INTRODUCTION

CREATING natural-looking human characters with realistic motions is essential for many applications, including movies, video games, robotics, sports training, medical analytics and social behavior recognition, and so on. Using expensive and special equipment, such as multi-cameras and reflective markers based motion capture systems, this task can be achieved without too much pain for scenes that do not impose many restrictions. Yet the inconvenient accessibility to such equipment has limited the flourishing of 3D human motion related applications.

The ideal and most convenient way is to use off-the-shelf RGB cameras to capture live performance and create 3D motion data. The minimal requirement of a single RGB camera is particularly appealing, as it offers the lowest cost, easy setup, and the potential of converting huge volume of Internet videos into a large-scale 3D human body corpus. Recent years have seen much research efforts being devoted to estimating not only the skeletal motion but also body pose and shape. Yet reconstructing 3D pose and shape from a single RGB camera is still a challenging and underconstrained problem with inherent ambiguities, especially in wild uncontrolled environment and in realtime. Therefore the state-of-the-art results are often vulnerable to ambiguities in the video (e.g., occlusions, cloth deformation, and illumination changes), degeneracy in camera motion, and a lack of discernible features on a human body. Moreover,

methods that achieve realtime, robust as well as accurate performance have rarely been seen common so far.

We introduce an approach that is capable of obtaining accurate 3D human poses and body shape from single wild images in realtime. When applied to video sequences, our system outputs temporally consistent bodies in motion at more than 20 Hz on a desktop computer. The power of our method comes from a convolutional neural network (CNN) which leverages a multi-task architecture that is able to outputs five results simultaneously: foreground mask, 2D joint positions, body partition, 3D part orientation fields (POFs) and *uv* coordinates. Body partition index and *uv* coordinates indicate part-specific *uv* coordinates, which is called IUV [1]. While none of existing networks support so many tasks at one time, this architecture makes it possible to refine multiple predictions recurrently. The regressed results are fed into a kinematic skeleton pose and body geometry fitting optimizer and outputs a camera-relative full 3D posed body mesh. The success of our approach also relies on the expansion of publicly available training datasets. While it is feasible to annotate a small number of labels in 2D images, upgrading to a large number of 3D representation becomes impractical. The new data is collected with our in-house cost-efficient, marker-less and scalable data acquisition system, and is preprocessed efficiently.

The power of our system is demonstrated by reconstructing 3D human poses and shapes for a wide variety of subjects from monocular video sequences. We have tested our realtime system on both live video streams and the Internet videos, demonstrating its accuracy and robustness under a variety of uncontrolled illumination conditions and backgrounds, as well as significant variations on races, shapes, poses, clothes across individuals. We show that our system can reconstruct bodies with realistic poses for highly dynamic motions such as figure skating (Fig. 1), low

- L. Jiang, J. Ye are with NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China and School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China.
E-mail: jiangliguo2015@ia.ac.cn and yejuntao@gmail.com.
- J. Zhang and C. Wang are with Xmov, Shanghai, China.
- M. Li and X. Liu are with State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou, China.
- J. Chai is with Texas A&M University.

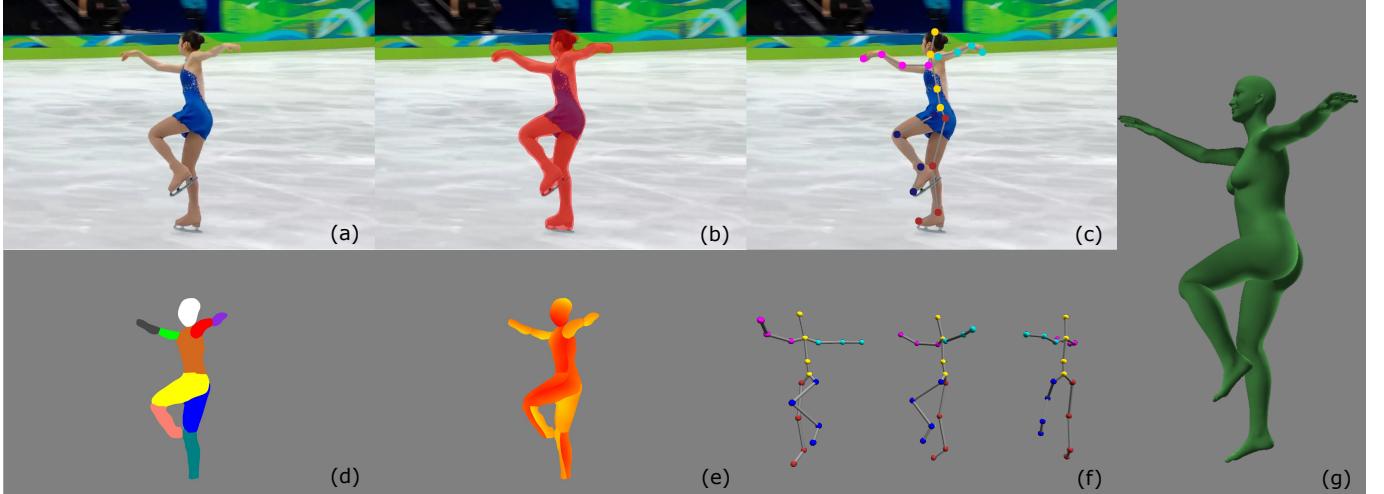


Fig. 1: Given an image (a), our regression network produces five outputs simultaneously: foreground segmentation mask (b), 2D joints positions (c), body partition (d), and a *uv* map (e), 3D part orientations (applied to a mean skeleton) (f). These outputs further guide the generation of a full-body model (g). The whole process runs in realtime on a desktop computer.

energy motions such as walking, and motions with human-environment interaction such as sitting and standing up. We evaluate the importance of each key component of our algorithm, by dropping off each component in the reconstruction. We advance the state-of-the-art realtime reconstruction of 3D human poses and detailed geometric body meshes from single images, and offer comparisons with alternative solutions [2]–[5].

The highlights of our 3D reconstruction system are

- **Realtime.** Thanks to our specially designed neural network, we are able to regress multiple human structural features from single images in realtime. We further feed network outputs to an efficient 3D human pose and body geometry fitting optimizer, and achieve realtime reconstruction performance.
- **Fully automatic and robust.** With the abundant regression outputs per-frame, reconstruction can be achieved from one single image, independent of any pre-initialized state. This makes reconstruction from videos no longer suffers from the headache of reinitialization. Our system is also robust to illumination variation, as well as clothing diversity.
- **Accuracy.** Our realtime system achieves reconstruction quality that is even more accurate than most offline or video-based methods in wild images. This achievement is mainly due to three points: (1) a novel multi-task deep learning network predicts abundant features, which boosts each other; (2) an efficient reconstruction process that seamlessly integrates all the visual features obtained from the deep learning network. (3) the augmentation to existing training dataset with our newly collected data.

2 RELATED WORK

The research on human body reconstruction from single RGB images is traced back to skeleton joints estimation, from 2D to 3D, and has achieved significant advances in recent years. This line of work has further boosted the

interest for simultaneous pose and shape estimation. We will focus our review on 2D pose estimation, 3D pose and body reconstruction from single images.

2D Pose Estimation. Nowadays image based single-person estimation [6]–[8] has achieved great improvement due to the success of CNN. These methods usually regress a probability map for each joint, designating the probability of a joint being located on each image pixel. The image-to-surface correspondence (IUV), represented by human part partition and *uv* coordinates map, was proposed in Densepose [1], and it is more effective and expressive than positioning just sparse 2D joints. By predicting the (u, v) coordinates and body part index for each pixel, a dense correspondence between pixels and points on a 3D mesh is defined. Our goal is different from these 2D or dense pose estimation methods in that we focus on 3D pose and geometry model reconstruction.

3D Pose estimation. Other than regressing 2D pose or dense pose from single images, many people attempt to estimate 3D pose directly from images. Most recent works can be divided into two categories: the *one-stage method* and the *two-stage method*. In the two-stage methods [9]–[12], the task of 3D pose estimation is decoupled into 2D joint detection and 3D coordinate regression. However, due to ambiguity of 3D estimation from 2D joints, these methods not only overlook certain image features having 3D cues, but also are very sensitive to the results of 2D pose estimation. To overcome the ambiguity in lifting 2D to 3D, priors are introduced in some works. Pavlakos et al. [13] further annotated the ordinal depth relation in the COCO dataset [14] and the MPII dataset [15], and proposed to estimate not only the 2D pose but also the ordinal depth relation as the extra input for lifting 2D to 3D. They achieved much better results. Different from [13], joint limits and bone lengths are introduced as constraints in [16]. The one-stage methods usually use a single cropped image as the input to a CNN, and directly obtain root-relative 3D joint positions [18]–[20], parent-relative joint positions [21], or voxel joint probability map [22], [23]. In the Vnect method

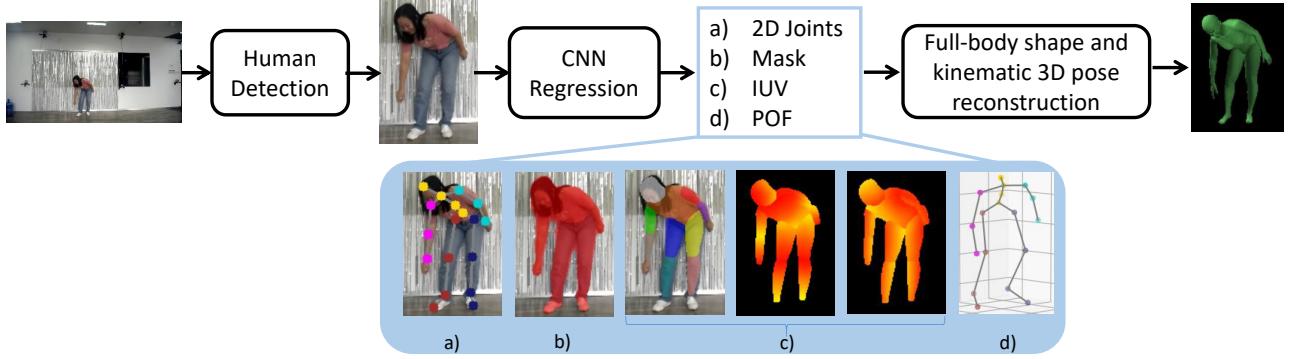


Fig. 2: System overview. The CNN outputs 2D joints, foreground mask, IUV (including body partition and *u*-map and *v*-map) and POF (Part Orientation Field).

[3], a fully convolutional network structure is proposed to directly regress location maps, in order to decrease the dependency on tight bounding boxes for human. More importantly, VNect gets global coordinates rather than root-relative coordinates, and produces real-time performance. The OriNet [24] decouples bone lengths and bone orientations by representing 3D poses with 3D orientations of limbs, which are very suitable for motion control. We also adopt the representation of 3D orientations of limbs. Yet different from above 3D pose estimation network, we design an end-to-end network to regress a foreground mask, 2D joint positions, body partition, *uv* coordinates and 3D part orientations simultaneously. Please note that existing works address only one or a subset of the tasks that we address. Multi-task learning usually boosts the quality of each individual output due to the correlation among tasks, and our experiments witness this fact. On the other hand, while their goal focuses on 3D pose estimation only, we further reconstruct human body geometry automatically. With these image features and the geometry model, we are able to obtain much more accurate pose reconstruction with strong temporal fitting.

Model-based Pose Estimation. Our method is related to one set of *model-based pose estimation* methods. Such approaches consider a parametric model of the human body, like SCAPE [25], SMPL [26] and SMPL-X [27], and the goal is to reconstruct a full 3D body pose and shape. These approaches can be further divided into *model-based optimization* and *model-based regression*.

In the first category, [28] relies on annotated 2D ground truth, including joint landmarks and body silhouettes, to optimize the parameters of the SCAPE model through minimizing errors of the reprojected evidence. With the SMPLify approach [4], this procedure was made automatic by replacing annotated 2D joints with 2D pose estimator. The whole process is then independent of user interference. Moreover, inter-penetration constraints are introduced to decrease the depth ambiguity when lifting 2D joints to 3D. The human shape estimation in SMPLify, however, relies on 2D joints only and does not constrain the body shape completely. To address this issue, UP3D [29] further extends the SMPLify method by adding human silhouette to estimate human shape parameters, with the pipeline being still automatic. Because of the binary representation

of the human silhouette, as well as the introduction of cloth intervention in this method, the body shape is still not sufficiently constrained. To overcome these issues, two mechanisms have been introduced by our method. The first one is the IUV, which is albeit more expensive but provides a dense correspondence between an image and a 3D model. The second one is the 3D limb orientation, which makes the reconstruction of human shape and pose more precise.

Among the model-based regression methods, HMR [30] uses a weakly supervised approach to regress the SMPL parameters directly from images, relying on 2D keypoints reprojection and a pose prior learnt in an adversarial manner. Instead of regressing SMPL parameters directly, CMR [31] builds a structure with Graph-CNN to model the connection of adjacent vertices of a human body mesh, and the 3D coordinates of mesh vertices are directly regressed. EFT [32], on the other hand, attempts to enrich wild images with missing SMPL parameters. By fine-tuning the HMR for each wild image, a few iterations to minimize errors of the 2D projection, and the current SMPL parameters are obtained. Treating these parameters as the ground truth for wild images, the original HMR is fine-tuned for the whole wild datasets. SPIN [2] adopts a similar idea. Rather than fine-tuning the HMR to get the ground truth for wild images, SPIN use the optimization-based method, like SMPLify [4], to refine the result to be used by HMR as ground truth. Instead of directly regressing highly non-linear shape and pose parameters from an image, we regress multiple image features, and get body shape and pose by a well-designed optimization formulation. The experiments show that our reconstruction results are much more accurate, and also stable on image sequences.

Model-based tracking. Our work is also related to model-based tracking of 3D human poses using a single RGB camera. Usually this type of method pre-defines a human skeleton/body on initialization, and the 3D pose is updated by minimizing the inconsistency between the hypothesized poses and observed 2D measurements [33]. This method, on one hand, needs a careful initialization; on the other hand the optimization is prone to get stuck in local minima, leading to track failures in the coming frames. What is different in our method is that the body model is reconstructed automatically, and is not sensitive to initialization under abundant constraints. Accurate results

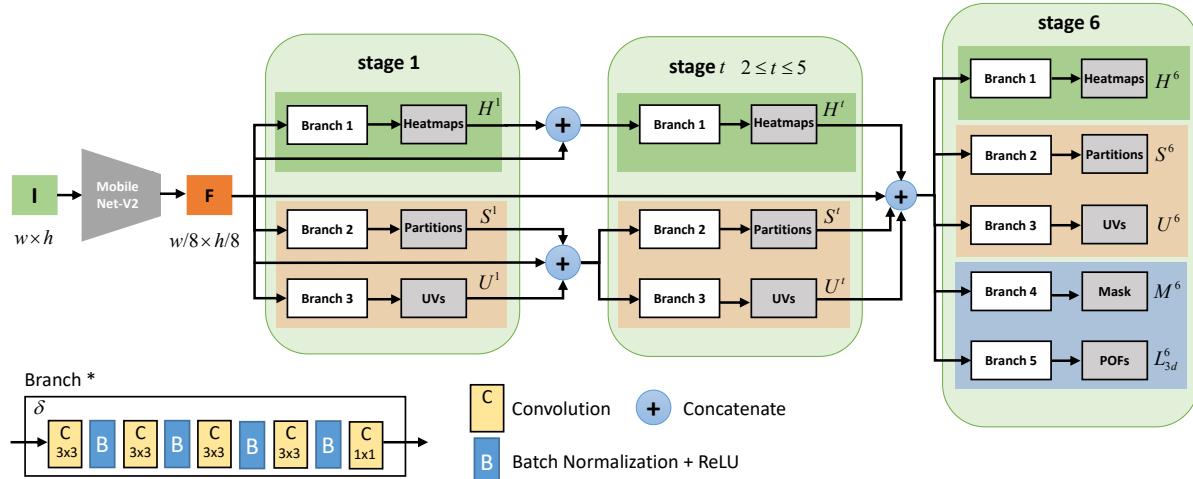


Fig. 3: Architecture of our multi-task network.

are obtained from single images, therefore it is also robust on image sequences.

Several other works combine the power of regression and fitting, as we do. The Total Capture method [34] regresses 2D joint positions and 3D limb orientations with a network, and then optimize human face, body, and hands with a unified model Adam [35]. Our method outputs more predictions in realtime (e.g. the IUV and foreground mask), thus gives more accurate reconstruction of body and pose. More importantly, the goal of our system design is realtime, therefore we want to discuss more about realtime systems here. VNect [3] is the first system that captures kinematic skeleton using a single RGB camera. It uses a fully convolutional network to regress 2D pose and 3D root-relative joint positions, and then fit for a kinematic skeleton. PhysCap [36] adds environment constraints for VNect to ensure the biophysical plausibility of human postures. In contrast, our multi-task network outputs more features thus more expressive geometry models can be reconstructed, which is manifested by the experimental results. Based on human skeleton pose capture, some methods further capture non-rigid deformation of clothing using optimization such as MonoPerfCap [37] and LiveCap [38], or regression such as DeepCap [39]. However, the dependency on a pre-scanned, pre-reconstructed and pre-rigged subject-specific model limits the usability of these methods, while our system is fully automatic.

3 METHOD OVERVIEW

Our method takes as input a single RGB image with a single person, and outputs a 3D human body whose pose and shape are in accordance with the person in image. This method consists of two parts: a neural network that regress measurements of human anatomical structures, and an optimization model that utilizes the network outputs to build a 3D body mesh. For the regression part (§ 4), a human is first detected from image by YOLO [40], outputting a bounding box. Then the cropped image is fed into a convolutional neural network (CNN) to get five outputs,

namely foreground segmentation mask, 2D joints positions, body partition, uv coordinates, and 3D part orientations (which is also encoded as part orientation field [24], [34]). For the optimization part (§ 5), our method reconstructs body pose and shape by fitting a deformable human model. We show that with as many as five features being integrated into the optimization pipeline, the reconstruction reaches an accurate fitting that has not been seen before. The whole pipeline is illustrated in Fig. 2. The success of our approach also relies on the enlargement of publicly available training datasets. We describe how the new data is collected and preprocessed with our in-house acquisition system in § 6.

4 THE TRAINING NETWORK

As mentioned, the key of our method is a multi-task CNN regressor for predicting five human anatomical structures: foreground segmentation mask, 2D joints positions, body partition, uv coordinates and 3D part orientations. The motivation behind this multi-task architecture is that more outputs gives more visual cues to be used for reconstruction. Actually this architecture refines multiple predictions recurrently, as a result each individual prediction turns to be more accurate. This is not a surprise, as the power of multi-task learning is that efficiency and prediction accuracy can be improved by learning multiple objectives from a shared representation [41].

Our multi-task regressor is a fully convolutional network. More specifically, given a RGB image $I \in R^{3 \times w \times h}$, a feed-forward network simultaneously predicts a set of 2D joint confidence maps $H \in R^{J \times w \times h}$ (where $J = 18$ is the number of joints to predict), human mask probability map $M \in R^{w \times h}$, human part probability map plus uv map, and 3D Part Orientation Fields $L \in R^{3O \times w \times h}$, where $O = 17$ is the number of body parts.

We use the term *IUV map* to indicate human partition probability map $S \in R^{(C+1) \times w \times h}$ (for $C = 24$ partitions and one background) and *uv* coordinates $U \in R^{2C \times w \times h}$, as did in [1]. To our knowledge, no previous works have ever regressed so many outputs as we do.

4.1 Multi-task CNN Regression

Fig. 3 illustrates the structure of our multi-task network. It is inspired by architectures like [42]–[44], which refine the predictions recurrently. An image is first encoded by a convolutional network, generating a set of image features F , which are then passed over to the first estimation for each individual task at stage 2. We get coarse predictions for joint confidence maps H^1 , IUV maps (body partition S^1 and uv coordinates U^1) in stage 1. In the successive stages, the network takes as input the image feature F , the results of previous stages of the same type. We formulate the procedure as follows:

$$H^t = \delta_H^t(Cat(F, H^{t-1})) \quad (1)$$

$$S^t = \delta_S^t(Cat(F, S^{t-1}, U^{t-1})) \quad (2)$$

$$U^t = \delta_U^t(Cat(F, S^{t-1}, U^{t-1})) \quad (3)$$

where $2 \leq t \leq 5$ is the stage index, and $\delta(\cdot)$ is the mapping for Branch $*$, as defined as four Conv3×3-BN-ReLU blocks and one Conv1×1 task-specified regressor. $Cat(\cdot)$ is the concatenation operation. In stage 6, the joint confidence map and IUV map from the previous stage is concatenated and treated as input to predict not only the joint and IUV, but also two additional terms: the mask M and the part orientation maps L_{3d} .

Loss Term. To guide the training of the multi-task network, we apply losses for predictions at each stage, specifically L_2 losses for the confidence maps H , POFs and UV maps. Note for the UV map, we only take into account a body part if the pixel is located inside it. When training part partition, a standard multi-class cross-entropy loss is used. Note that due to the difference of human part areas, we balance the supervision for part segmentation classification by the weight w_c for each human part c , so that the network would not over-fit body parts of large area. The balance weight w_c is inversely proportional to the part area, as in [45]. Our IUV (body partition and UV maps) ground truth for hands is inaccurate, because we fit a statistic model into a skeleton without finger joints, so we just ignore the IUV loss for hands. The segmentation mask is trained by binary cross-entropy loss.

Implementation. The training of our multi-task network consists of three phases. (1) First, we pre-train our network for the 2D joint detection task with in-the-wild image dataset for stage 1 ~ 5, ignoring other tasks, which gives better generalization performance. Our 2D joint detection task is trained with an initial learning rate of 10^{-3} and is reduced every 200,000 iters by a factor $\gamma = 0.333$, as [46] does. (2) Second, we combine our newly collected dataset with 2D joint dataset, and apply a mix-training strategy for other tasks while freezing the weights of feature extractor and 2D joint detector for 100,000 iters by a learning rate of 5×10^{-4} . Note that our newly collected data has all desired ground truth for every task. Fig. 4 shows a few images in it, including the original captures and the augmented ones through background replacement. Data augmentation with background replacement greatly increases the generalization of in-the-wild images. (3) Finally, we unfreeze the weights of well-trained 2D task and feature extractors, but apply a smaller learning rate (multiplied by 0.1) for these



Fig. 4: Our training dataset contains original captured images, as well as augmented images with background replacement.

weights. Our full-task training takes 1,000,000 iterations with a learning rate of 5×10^{-4} reduced every 200,000 iters by a factor $\gamma = 0.333$. We employ a rotation augmentation ($\pm 30^\circ$), a scaling augmentation (0.75-1.25) and left-right flipping (only for in-the-wild dataset) for training. We use the Caffe framework [47] for network training, and use the Adadelta solver [48]. The performance of each task is strongly dependent on the relative weighting between the loss of each task [49]. And in our experiment, we set loss weights as follows: $w = 0.5$ for UV , $w = 0.05$ for body partition, $w = 0.5$ for heatmap, $w = 1.0$ for foreground mask and $w = 1.0$ for POFs. To balance accuracy and efficiency, we use MobileNet-V2 [50] as our feature extractor. Note that we removed the downsampling operations in last two blocks (by replacing stride=2 in downsampling convolution with stride=1), and maintained the size of the final feature map to be 1/8 of the input image, which is 224-by-224.

5 AUTOMATIC KINEMATIC POSE RECONSTRUCTION AND FULL-BODY SHAPE MODELING

Our work sets a new mark in terms of level-of-detail that previous work did not reach. This mainly attributes to our innovative kinematic pose reconstruction and full-body shape modeling. The network output (as in § 4) is of low-quality and noisy, and may not be compatible with input images or with human kinematic constraints. Directly using such information leads to inaccurate reconstruction of human motion. To refine the network output, we develop a novel algorithm that accurately reconstructs the human motion as well as a subject-specific full-body mesh model.

5.1 Human Full-body Representation

Similar to SMPL, we approximate the human full-body geometry with a skinned mesh that is driven by an articulated skeleton model using Linear Blend Skinning (LBS). Our skeleton has 45 degree of freedoms (DOFs), 6 of which

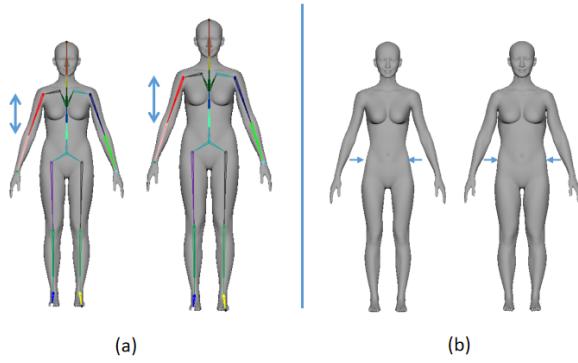


Fig. 5: Human shape variations: (a) bone length variation; (b) body part thickness variation.

are for global position and orientation, and 39 are for joint angles (note that a joint may be of 1, 2 or 3 DOFs). We built a female mesh model of 28,109 vertices (or 56,142 triangular faces), carrying more geometric details than the SMPL model of 6,890 vertices. The female model is elaborately rigged and parameterized such that the shape can be easily controlled.¹

Following a relatively mature process we build a parametric human full-body geometry model (Fig. 5). The model is controllable in two aspects: (1) skeleton scales, which encode coarse-level variation like the overall and the per-bone scales, (2) mesh vertex offsets, which encode fine-level shape variation, such as thickness of a limb. The parametric human full-body model can be represented as:

$$\mathbf{H}_i(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta}) = \sum_{j=0}^{n-1} w_{ij} T_j(\boldsymbol{\theta}) \hat{\mathbf{v}}'_{ij}, \quad (4)$$

$$\hat{\mathbf{v}}'_{ij} = \mathbf{S}(\boldsymbol{\alpha}) \otimes (\mathbf{Q}(\boldsymbol{\beta}) \oplus \hat{\mathbf{v}}_{ij}),$$

where $\mathbf{H}_i(\cdot)$ is the coordinate of the i -th vertex of the mesh model, $\hat{\mathbf{v}}'_{ij}$ is the result after applying the scaling and offsetting to $\hat{\mathbf{v}}_{ij}$ (which is the i -th vertex represented in the local coordinate frame of the j -th bone), $\mathbf{Q}(\boldsymbol{\beta}) \oplus$ describes the vertex offsetting, $\mathbf{S}(\boldsymbol{\alpha}) \otimes$ describes the bone scaling, the shape parameters $\boldsymbol{\alpha} \in R^8$ and $\boldsymbol{\beta} \in R^{26}$ provide a low-dimensional representation of human bone scale variances and vertex offset variances across individuals respectively, $\boldsymbol{\theta}$ is the pose for deformation, $T_j(\boldsymbol{\theta})$ is the transformation of the j -th bone for pose $\boldsymbol{\theta}$, w_{ij} is a sparse weight map for deformation, n is the number of bones.

5.2 Kinematic Pose Reconstruction

Given the subject-specific full-body mesh model (obtained in § 5.3) and the network observations (2D pose from 2D joints probability maps, 3D part orientations from the 3D limb orientation fields, the human mask, the IUV map for frame image I_i), our goal is to estimate a human pose $\boldsymbol{\theta}$ which best matches the network observations. We estimate

1. Yet currently we do not have a decently parameterized male model on par with the female model. For scenes with a male subject, we either use motion-retargeting to drive a male model mesh but without body dimension adjustment (e.g. Fig. 17(b), or just blindly use the female model (e.g. one case in the accompanying video).

the human pose $\boldsymbol{\theta}$ by minimizing the following objective function:

$$\arg \min_{\boldsymbol{\theta}} (w_{data} E_{data} + w_{prior} E_{prior} + w_{temporal} E_{temporal}). \quad (5)$$

where E_{data} is the data term penalizing the registration error between the synthesized human model and the observation. E_{prior} is the prior term that penalizes invalid human pose configuration, and $E_{temporal}$ is the pose smoothness term that penalizes the jerkiness in the motion, which is only used for video application. While searching for the solution in an iterative manner, it is possible (and recommended) to use the pose $\boldsymbol{\theta}_{prev}$ from the previous frame as the initial guess. We will describe each term in detail in the following subsections.

5.2.1 The Data Term

The data term E_{data} evaluates how well the current human pose $\boldsymbol{\theta}$ matches the network observations by the analysis-by-synthesis strategy. Given the human pose $\boldsymbol{\theta}$, we first apply *skeleton subspace deformation* to synthesize a full-body mesh model. And then we compute the registration error between the network observations and the synthesized human model. The data term is defined as

$$E_{data}(\boldsymbol{\theta}) = w_{2d} E_{2d} + w_{3d} E_{3d} + w_{iuv} E_{iuv} + w_{mask} E_{mask}. \quad (6)$$

Here E_{2d} and E_{3d} are alignment constraints based on predicted 2D joints positions and 3D limb orientations, respectively. E_{iuv} penalizes the registration error between the synthesized *uv* map and the observed *uv* map, and E_{mask} penalizes error between the synthesized and the observed mask.

Sparse 2D alignment. To minimize the discrepancy between estimated 2D joints \hat{P}_{2d} and the projections of the 3D joints from the human body model, we incorporate \hat{P}_{2d} into the following reprojection constraint:

$$E_{2d}(\boldsymbol{\theta}) = \sum_i \|\Pi(J_{3d}^i(\boldsymbol{\theta})) - \hat{P}_{2d}^i\|_2^2. \quad (7)$$

Here $J_{3d}^i(\boldsymbol{\theta})$ is the i -th joint in the skeleton, and Π is the 3D-to-2D projection matrix according to known intrinsic camera parameters.

Sparse 3D alignment. Since many 3D poses share the same reprojection of 2D pose in single image, and it is hard to infer a 3D pose only with above mentioned reprojection constraint. Therefore we add another 3D constraint

$$E_{3d}(\boldsymbol{\theta}) = \sum_{(m,n) \in B} \|(\|J_{3d}^m - J_{3d}^n\|_2 \cdot \hat{\mathbf{O}}_{m,n} - (J_{3d}^m - J_{3d}^n))\|_2^2. \quad (8)$$

Here $\hat{\mathbf{O}}_{m,n}$ is the estimated 3D direction of limb (m, n) , which is the mean value along the segment from joint J^m to J^n .

Why do we use 3D direction? Various representations have been put forward to denote a 3D pose, including 3D joint positions, 2D joints position plus root-relative depth, and 3D limb directions and so on. We adopt 3D limb direction due to its two advantages. First, limb orientation is scale-invariant and dataset independent, which helps resolve scale ambiguity and generalizes easily to diversity data. Second, because an auto-reconstruction of human

shape is done before tracking, there is no need to worry too much about limb length ratios in the following frames.

Dense IUV alignment. In addition to the coarse level shape variations caused by bone length and pose change, another data term E_{iuv} constrains fine-level shape variations, such as the thickness of limbs. To this end, we construct a dense pixel-to-surface correspondence represented as an IUV map. Each pixel of an IUV image has a body part index i , and a (uv) coordinate that maps a pixel to a unique point on the surface of a body model. Given an IUV map predicted by the neural network, we select some reliable pixels p which satisfies $\delta^{(1)}(p) = 1$ for the function defined as

$$\delta^{(1)}(p) = \begin{cases} 1, & \text{if } \text{Max}\{\mathbf{v}(p)\} - \text{Max2nd}\{\mathbf{v}(p)\} > \phi \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

where $\mathbf{v}(p)$ is the segmentation probability vector of pixel p . In our experiment, we set threshold $\phi = 0.5$. Interpolating these image-to-surface points, a more accurate human model is obtained by minimizing

$$E_{iuv}(\boldsymbol{\theta}) = \sum_p \delta^{(1)}(p) \|\Pi(M(\boldsymbol{\theta}, p)) - \hat{I}(p)\|_2^2, \quad (10)$$

where $M(\boldsymbol{\theta}, p)$ is the synthesized mesh vertex corresponding to pixel p , and Π is the 3D-to-2D projection matrix according to known intrinsic camera parameters.

Mask Term. The foreground segmentation mask term is to penalize the inconsistency between the mask of synthesized human mesh model and the mask from network observations.

$$E_{mask}(\boldsymbol{\theta}) = \sum_v \delta^{(2)}(v) \|\Pi(v) - q_v\|_2^2 \quad (11)$$

where v is the mesh vertex, Π is the 3D-to-2D projection matrix according to known intrinsic camera parameters, $\delta^{(2)}(v)$ indicating whether v is outside the observed mask or not, q_v is the corresponding 2D image position for v obtained from the distance map of the observed mask.

5.2.2 The Prior Term

To make joints of a human skeleton to be physically meaningful, two different priors, the pose space prior and the joint limit, are defined and used to form the prior term

$$E_{prior} = w_{pose_prior} E_{pose_prior} + w_{jt_limit} E_{jt_limit}.$$

Pose Space Prior. We construct individual PCA models for each body part (e.g., shoulders, arms, spines, legs and feet) via CMU mocap database². With those part-wise PCA models, we are able to constraint the solution space into the physically meaningful area by minimizing the following objective function:

$$E_{pose_prior}(\boldsymbol{\theta}) = \|P_k^T(P_k(\boldsymbol{\theta} - \boldsymbol{\mu})) + \boldsymbol{\mu} - \boldsymbol{\theta}\|_2^2, \quad (12)$$

where $\boldsymbol{\mu}$ is the mean vector of the PCA model, and P_k is the first k principle components of the PCA model. Here k is chosen to retain 95% of original variations.

Joint Limit. The joint limit term is added to penalize invalid joint poses that exceed the range of joint movement.

2. <http://mocap.cs.cmu.edu/resources.php>

Every joint angle θ_i , $i = 7, 8 \dots 45$ should stay within $[\theta_i^l, \theta_i^u]$. The joint limit term E_{jt_limit} can be represented as

$$E_{jt_limit}(\boldsymbol{\theta}) = \sum_{i=7}^{45} (\delta^{(3)}(\theta_i < \theta_i^l) \|\theta_i - \theta_i^l\|^2 + \delta^{(3)}(\theta_i > \theta_i^u) \|\theta_i - \theta_i^u\|^2), \quad (13)$$

where the binary function $\delta^{(3)}(x)$ is 1 if x is true, and is 0 if x is false.

5.2.3 Temporal Smoothness Term

We add a smoothness term to penalize the pose jerkiness between two consecutive frames. This smoothness term is defined as:

$$E_{temporal}(\boldsymbol{\theta}) = \|\boldsymbol{\theta} - \boldsymbol{\theta}_{prev}\|_2^2 \quad (14)$$

5.2.4 Optimization

As Eq. 5 is represented as a sum of squares, we can efficiently solve it by Gauss-Newton method. Since every term is differentiable, we can directly compute the Jacobian matrix $J(\boldsymbol{\theta})$ and then follow the standard Gauss-Newton step to solve $\delta\boldsymbol{\theta}$ and update current $\boldsymbol{\theta}$

$$\begin{aligned} J(\boldsymbol{\theta})^T J(\boldsymbol{\theta}) \delta\boldsymbol{\theta} &= J(\boldsymbol{\theta})^T r(\boldsymbol{\theta}), \\ \boldsymbol{\theta} &= \boldsymbol{\theta} + \delta\boldsymbol{\theta}, \end{aligned} \quad (15)$$

where $r(\boldsymbol{\theta})$ is the residual vector formed by concatenating each term.

Parameter values. In our implementation, the weights w_{data} , w_{prior} and $w_{temporal}$ in Eq. 5 are set to 1.0, 0.5 and 5.0 in our experiments. Weight settings in Eq. 6 are $w_{2d} = 20.0$, $w_{3d} = 300.0$, $w_{iuv} = 2.0$ and $w_{mask} = 1.0$. The weights in E_{prior} are $w_{pose_prior} = 0.002$ and $w_{jt_limit} = 5.0$. The weight for temporal smoothness term is $w_{temporal} = 0.003$.

5.3 Full-body Shape Modeling

Now we describe how to reconstruct a full-body mesh model for a subject using the network observations. Note in the case that the input is a video stream, the shape reconstruction is done only once, for the first frame.

5.3.1 Shape Reconstruction

Given an image I , our goal is to reconstruct a subject-specific human body model $\mathbf{H}(\alpha, \beta, \boldsymbol{\theta})$, with the derived image positions for 2D joints, the segmentation mask, body partition, 3D part orientation, and the dense correspondence (uv map) for human parts. We formulate the reconstruction as a non-linear optimization problem

$$\arg \min_{\alpha, \beta, \boldsymbol{\theta}} (w_{data} E_{data} + w_{pose_prior} E_{pose_prior} + w_{shape_prior} E_{shape_prior}), \quad (16)$$

where E_{data} is the data term that describes the registration error between the synthesized model and the network observations that inherits from § 5.2, E_{pose_prior} is the pose prior term that inherits from § 5.2, E_{shape_prior} is the shape prior term that penalizes the deviation of shape parameters from the human shape in the database.

The shape prior term. We model the shape prior distribution with multiple single-variable Gaussian models

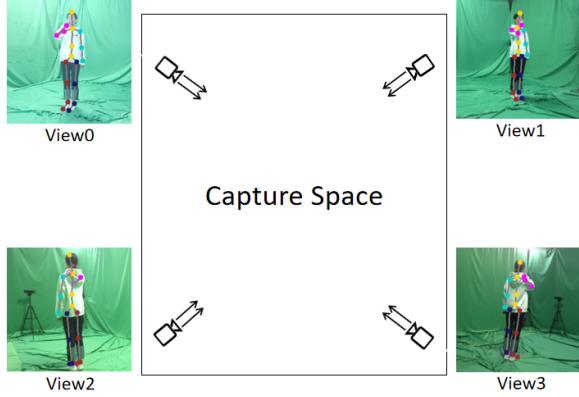


Fig. 6: The capture space.

that is learnt from the human shape database. We define E_{shape_prior} as the deviation distance with parameters

$$E_{shape_prior} = \left(\sum_i |\alpha^i - \mu_\alpha^i|/\sigma_\alpha^i + \sum_j |\beta^j - \mu_\beta^j|/\sigma_\beta^j \right),$$

where μ is the mean value and σ is the standard deviation.

5.3.2 Optimization

Directly optimizing Eq. 16 is not efficient and often falls into local minima, because the pose and the shape are coupled. To address this issue, we decouple the optimization into two sub-optimization problems: pose optimization and shape optimization. In each iteration, we first fix shape parameters and optimize pose parameters, and then vice versa.

Pose estimation. In this step, we optimize the pose parameter θ while fixing the shape parameter α, β . This process is identical to that in § 5.2.

Shape estimation. In this step, we optimize the shape parameter α, β while keeping the pose parameter θ fixed. And therefore the optimization problem can be represented by

$$\arg \min_{\alpha, \beta} (w_{data} E_{data} + w_{shape_prior} E_{shape_prior}). \quad (17)$$

Following the optimization method in § 5.2.4, we can solve the shape parameters α and β .

Parameter values. For shape reconstruction, the energy terms E_{data} and E_{pose_prior} are the same as in § 5.2. The shape prior weight we use is $w_{shape_prior} = 0.1$.

6 TRAINING DATA

As one major highlight of our work, we complement existing datasets by building a dataset with a large number of actors, everyday clothing appearances, a broad range of motions. The data capture setup eases the appearance augmentation and extends the captured variability. This gives a potential to bring significant boost to accuracy and generalizability of the learnt models. In this section we describe the capture environment and the recording process, as well as the processing of the captured data.

6.1 Experimental Setup

Our laboratory setup is shown in Fig. 6, where data is captured by four digital video cameras. The designated laboratory area is about $4m \times 4m$, and within it we obtain effective capture court of approximately $2.5m \times 2.5m$, where each subject is fully visible to all cameras. Four cameras are placed at four corners of the court. The floor and the wall are mantled with green curtains, making it easy for automatic segmentation of the foreground body. The total number of actors screened are over 300, covering a broad range of ages, body shapes and pose extensions. Our dataset has much more subjects than in any of existing 3D datasets. Each person is asked to do certain daily life motions as well as sport motions, for about three minutes. To eliminate redundancy between consecutive video frames, frame images are further filtered, and only 500 to 1,000 images are sampled for each actor. The sampling is achieved by clustering frame images according to the similarity of human poses.

6.2 Shape Reconstruction from Measurements

To build a highly accurate parametric body model for each actor, we take some body measurements while an actor is standing still in A-pose. A set of measurements $M \in R^{44}$ (including but not limited to lengths of limbs, shoulder and back, girths of chest, wrist, hip and stomach, etc.), are tailoring measured with a ruler. Our parametric model has $8 + 26 = 34$ shape parameters, more sophisticated than the SMPL model, which has only 10 shape parameters. With the measurements, parameters α and β are computed by minimizing an energy function

$$E(\alpha, \beta) = w_1 E_{geoDist} + w_2 E_{shape_prior}, \quad (18)$$

where $E_{geoDist} = \sum_{i=1}^{44} \|f_i(\alpha, \beta) - M_i\|_2$ assesses the error of geodesic distance on the parameterized human body, E_{shape_prior} determines the shape prior error, with respect to mixed Gaussian distribution. The above energy is minimized with the Particle Swarm Optimization method [51], [52].

Parameter values. We set $w_1 = 8.0$ for $E_{geoDist}$ and $w_2 = 0.8$ for E_{shape_prior} in Eq. 18.

6.3 Pose Reconstruction

With four multi-view images for each frame, a 3D pose is reconstructed, according to the following steps.

- 1) Estimate 18 joints on each image from each camera, with a 2D joint estimation method [43].
- 2) Validate the consistency of the estimated 2D joints from multi-views (see the following explanation).
- 3) Segment foreground body from background automatically (green curtains setting).
- 4) Solve the human motion by extending Eq. 5 into multi-views, while dropping off the IUV constraints and the 3D constraints from Eq. 6 (As both constraints are not available when constructing our datasets).
- 5) Recover the 3D body mesh with the shape from measurements and the pose from multi-view images, shown in Fig. 7.



Fig. 7: We reconstruct body mesh from multi-views as ground truth.

In Step 2, it is very common that in a single-view image some joints could be invisible due to occlusion. Therefore we propose a cross validation scheme based on multiple views. For a certain joint, we choose any pair of cameras and use the two 2D estimations to build the 3D joint position. The 3D joint is then re-projected with respect to the view-ports of the other two cameras, creating two projected joints. Each projected joint is compared against the estimated 2D joint under the same view-port, and their Euclidean distance is calculated. Only if the distances in two view-ports are both below a certain threshold (18 pixels in our experiment), this set of four 2D estimations for this joint is considered to be consistent and reliable. If four view-ports fail to reach consistency, we check if any three of them do. If that happens, the 2D joint estimation in the three view-ports are treated as reliable. If such three view-ports can not be found, we have to ask for help from the previous frame. The 3D joint from the previous frame is compared with the reconstructed 3D joint from any pair of cameras, and the distances are calculated. The minimal distance designates the pair of cameras and their estimations are treated as reliable.

6.4 IUV Maps

The image-to-surface correspondence (IUV map) proposed in Densepose [1] for human body is an essential mechanism for mapping a 2D image into a 3D geometry. We improved this idea with a more solid implementation, and it works well for persons with loose clothes. In Densepose certain pixels are manually sampled on each human part, and their corresponding points on the meshed surface are manually marked as well. Annotators are asked to determine the body silhouette if it is covered by clothes, and mark around 100 points for each human. In this situation the burden is heavy and errors prone to happen, especially when a human instance wears a large/loose skirt.

We adopt a quite different scheme for computing the part label and the (u, v) coordinate for each pixel, requiring much less human intervention. The reconstructed 3D mesh by measurements is re-projected according to the view-ports of the cameras, creating four images. As each mesh vertex has a (u, v) coordinate and a part label, it is trivial to compute (u, v) and label for each pixel of these images. Due to the topological complexity of human meshes, we follow

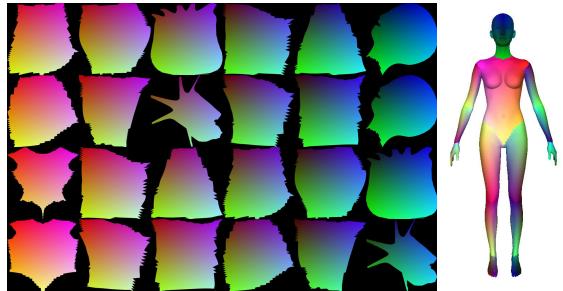


Fig. 8: The IUV maps of the 24 body parts.

Component	Time (ms)
Human detection (YOLO)	8.554
Image Preprocess	0.312
Multi-task Prediction	18.269
Pose Reconstruction	10.232
Others	11.678
Shape Recon (only for 1st frame)	328.227

TABLE 1: Running time of each component in our system.

Densepose and segment a human mesh into 24 parts, and define a uv -field on each part, as shown in Fig. 8.

7 RESULTS AND EVALUATIONS

We demonstrate the power and effectiveness of our system, by reconstructing 3D human bodies from both live streams and in-the-wild videos of various scenes (§ 7.1). We quantitatively compare the accuracy of our results with state-of-the-art 3D pose and/or mesh reconstruction methods (§ 7.2). Our method is also qualitatively compared against four most related state-of-the-art methods (§ 7.3). In § 7.4, we evaluate different parts the key components of our system by dropping off each term at one time for both multi-task network and optimization procedure. Our results are best seen in the accompanying video.

Computational time. Our system runs at 20 fps on a desktop computer for the current implementation. Table 1 reports the detailed timing for each component in our processing pipeline. All execution time is collected on a computer with an Intel i7 CPU and a nvidia Geforce GTX 2080Ti GPU. Apart from the shape reconstruction, which is done only once in the first frame, the total processing time for one cycle is under 50 ms.

7.1 Test on live streams and in-the-wild videos

Our system reconstructs 3D human poses and full-body geometry models from single images in realtime, and the reconstructed 3D poses is retargeted to animate a character in realtime (See Fig. 10). Our technology has potentials in applications such as game character control, embodied VR, sport motion analysis and reconstruction of community video.

We also test our method on various in-the-wild videos, showing its robustness to different actors with different body shapes and clothes, even under significant occlusion,

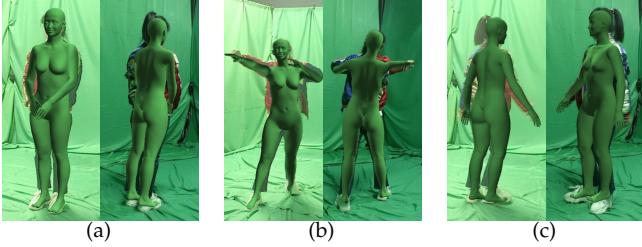


Fig. 9: On the left of each subfigure is the input view from which the 3D model is reconstructed. On the right, the model is rendered and overlaid in a different view.



Fig. 10: The reconstructed 3D poses from our system can be retargeted to animate a character in realtime.

lighting and background changes. Fig. 11 shows some excerpted frames. Besides, we also present other-view overlay results in Fig. 9 by using our multi-view test dataset, which reflects the accuracy of human reconstruction achieved by our method. Please refer to the accompanying video for more vivid results.

7.2 Quantitative Evaluation

We evaluate the performance of our method on two popular test benchmarks: 3DPW [53] (outdoor scenes) and Human3.6M [54] (indoor scenes). Following the standard protocol for 3D pose estimation [22] in Human3.6M, we use 5 subjects (S1, S5, S6, S7 and S8) for training, and the rest 2 subjects (S9 and S11) for testing. As for Human3.6M, we get ground truth parameters for training images using MoSh [56] from the raw 3D Mocap markers like [30]. As for 3DPW, we only use its testing dataset for evaluation as previous works [2]. Note that Human3.6M and 3DPW have different skeleton configurations from ours, we therefore learn a linear regressor for a mapping which maps our mesh vertices to 17 joints defined in Human3.6M, as did in [30]. For evaluation, we adopt averaged skeleton dimensions computed from the training set to rescale our reconstruction human, as did in [22].

The results are shown in Table 2. Our single image based method is even competitive to the video-based VIBE [5] on Human3.6M and 3DPW. The results also show that our newly collected data improves the performance further, especially on 3DPW (wild), with improved wild generalization.

Method	H36M-P1 ↓	H36M-PA ↓	3DPW-PA ↓
Mehta <i>et al.</i> [3]	80.5	-	-
Pavlakos <i>et al.</i> [13]	56.2	41.80	-
Sun <i>et al.</i> [23]	49.6	40.60	-
Zhou <i>et al.</i> [57]	39.9	32.1	-
Bogo <i>et al.</i> [4]	82.3	-	-
Kanazawa <i>et al.</i> [30]	87.97	56.8	76.7
Xiang <i>et al.</i> [34]	58.3	-	-
kolotouros <i>et al.</i> [31]	74.7	50.1	70.2
kolotouros <i>et al.</i> [2]	-	44.3	59.2
Joo <i>et al.</i> [58]	-	45.2	55.7
Kocabas <i>et al.</i> [5]	65.6	41.4	51.9
Ours (wild image + H36M)	66.3	47.2	64.1
Ours (wild image + H36M + ours)	63.7	41.8	53.2

TABLE 2: Quantitative Evaluation on Human3.6M (indoor) and 3DPW (outdoor). The number of H36M-P1 is the Mean Per Joint Position Error (MPJPE) in millimeter on Human3.6M, while the number of H36m-PA is the MPJPE on Human3.6M after procrustes alignment (PA). And 3DPW-PA is the MPJPE on 3DPW after PA. Our method achieves competitive performance against previous works.

Methods	PCKh@0.5 ↑	MPJPE-P1 ↓
2D	96.3	-
2D + IUVs	97.5	-
2D + POFs	96.2	51.8
2D + IUVs + POFs	97.8	49.8

TABLE 3: Ablation experiments on the effect of multi-task learning. In the experiment, we modify Fig. 3 to set different task combinations. Our results shows that IUVs is beneficial to 2D joint detection and part orientation predictions.

7.3 Comparisons with state-of-the-art methods

To show the efficiency of our method, we compare against two state-of-the-art regression-based methods, one is single frame method, SPIN [2], the other is video-based method, VIBE [5]. Fig. 12 shows the result of a side-by-side comparison. It is obvious that our method achieves better image-model alignments than SPIN and VIBE. Regression-based methods usually achieves global image-model alignments quickly, but at the cost of low quality. This type of nonlinear prediction is also uneasy to control, due to the mutual effect between human pose and shape. We decouple them, and since 2D joint positions and dense image-to-surface correspondence (IUV map) offer better image-model alignments, and 3D part orientation helps to avoid depth ambiguity, our method produces more accurate body reconstruction than SPIN and VIBE.

Furthermore, SPIN does not guarantee temporal stability because it regresses different body shapes from different images in a sequence, while VIBE fixes it, but it is not in realtime. Please refer to Fig. 12 and the accompanying video for the comparison results.

Our method is also compared against a recent realtime system, VNect [3], though it captures poses only. We compare not only the raw network outputs, but also the final fitting results (see Fig. 13). It is obvious that our method achieves better pose reconstruction than VNect. Two reasons account for this. Firstly, our multi-task network produces more accurate 3D joint positions, as shown in Fig. 13(c). Secondly, VNect initializes bone lengths for forearm and upper arm to an improper ratio, as seen in Fig. 13(b), while



Fig. 11: Our reconstructed 3D bodies with different shapes and clothes, under occlusion, lighting and background variations.

Methods	H36M-P1↓	H36M-PA↓	3DPW-PA↓
2D	121.3	103.4	116.8
2D + mask	118.3	105.1	110.9
2D + mask + 3D	64.7	43.4	51.8
2D + mask + 3D + IUV	63.7	41.8	51.1
2D + mask + 3D + IUV + temporal	65.9	42.4	53.5

TABLE 4: Ablation study on the importance of each energy term in optimization. We report the MPJPE/MPJPE-PA on Human3.6M (indoor) and 3DPW (outdoor) on 5 experiments. All five experiments share the same network outputs but differ in the energy terms in optimization.

our initialization matches with person in image very well. VNect initializes skeleton by averaging 3D joint positions from the CNN output at the beginning, which is thus very sensitive to single CNN outputs (3D joint positions). We instead utilize more image features, including 2D joints, 3D part orientation and IUV maps, to reconstruct human bodies more accurately and robustly. Please refer to the accompanying video for more comparison results.

SMPLify [4] is a somewhat hybrid method: it fits the SMPL model by optimizing regressed 2D joints without user intervention. Fig. 14 shows the results given by SMPLify and our method. At least three issues about SMPLify can be interpreted from the figure. First, SMPLify is more vulnerable to depth ambiguity than our method, as can be seen from images in the first row. This is because SMPLify relies on 2D joint reprojection alone for model fitting, and this is insufficient to robustly establish a 3D pose. Second, there is no constraints for foot orientation in SMPLify. Last, it is easy for SMPLify to fall into a local minima, as shown in second row. Our method has hardly convergence problem. For a single image, the initial shape from the average of our human model. We use it and predicted 2D joint positions and 3D bone directions to roughly estimate the root position and joint angles as initial pose. The optimization problem is



(a) Comparison our method with SPIN [2]



(b) Comparison our method with VIBE [5]

Fig. 12: From left to right: the input, SPIN/VIBE and our results. Our method produces better image-model alignments than SPIN and VIBE.

over-constrained.

7.4 Ablation Study

We have designed a realtime multi-task network that regresses more features than any other deep learning based methods. We believe that more features offer more visual cues that facilitate pose and shape reconstruction. In this section we justify this belief with experiments, evaluating the role of each regressed features in both the CNN regression and the body reconstruction process.

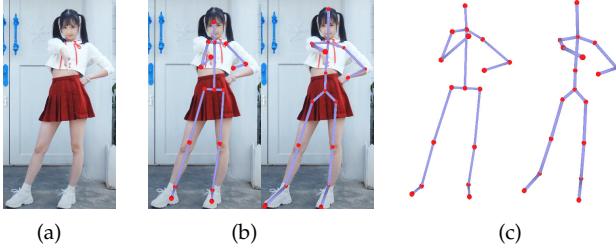


Fig. 13: Comparison of our results with VNect. (a) the input image; (b) the final results of VNect (left) and our method (right); (c) the raw network predictions of VNect (left) and our method (right).

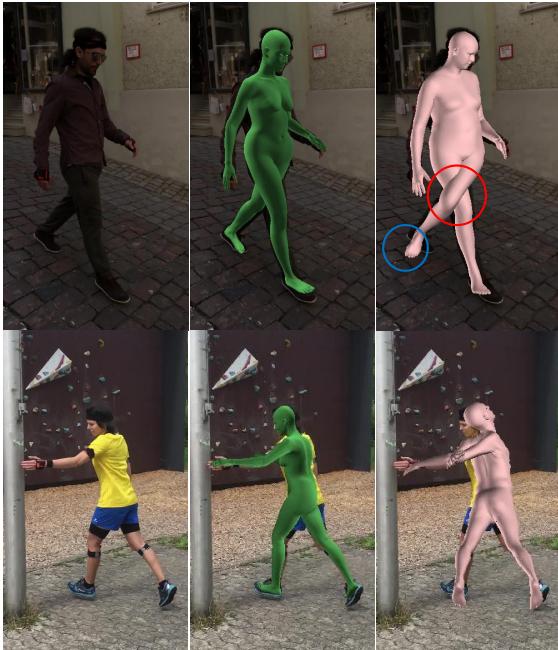


Fig. 14: Comparison of our method (middle) with SMPLify (right).

Quantitative analysis for multi-task network To evaluate the importance of our multi-task design, we modify the network in Fig. 3 into 4 structures with different configurations: a) 2D joint detection only, b) 2D joint detection + IUV branch, c) 2D joint detection + POFs, d) 2D joint detection + IUV + POFs. All these networks are trained with the same training dataset, and tested on our validation dataset, which contains 11 different subjects not in the training dataset. For metrics of 2D joint positions, we report PCKh@0.5 [7] (the higher the better). For 3D part orientation, we scale the predicted 3D part orientation by the ground-truth limb length to obtain the 3D joint positions, then align the root joint position and compute the MPJPE (the lower the better). The results are reported in Table 3, which shows the power of mutual promotion of multi-task. We found that IUV information improves the accuracy of 3D POF orientation. The reason is that IUV maps provide the part occlusion relationship which conveys some 3D information. IUV maps are usually more abstract and more powerful than 2D landmarks in representing a human, and it is used

by [59] as input.

Quantitative analysis for optimization Table 4 shows the quantitative performance, which reveals the importance of each energy term. We compare results under 5 different energy term settings: a) 2D position only; b) 2D position + mask; c) 2D position + mask + 3D part orientation; d) 2D position + mask + 3D part orientation + IUV; e) 2D position + mask + 3D part orientation + IUV + temporal. We report MPJPE on Human3.6M and 3DPW. The results show that every energy term in our optimization is beneficial for human pose reconstruction. The result with temporal term shows higher errors, because it ensures temporal smooth rather than the consistency with the network output cues.

The mask term in optimization. We evaluate the importance of the foreground segmentation mask by comparing the reconstructed bodies with and without this term. Fig. 15 clearly shows the role of the mask when predicted 2D joints and 3D orientation are inaccurate, especially when some joints are missing.

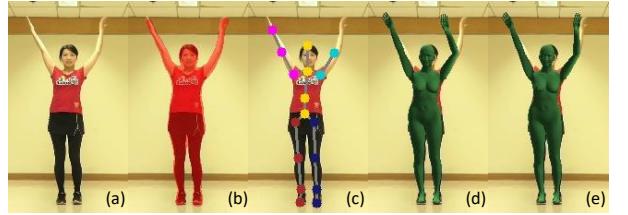


Fig. 15: Importance of the mask term. Given an image (a), the network predicts a foreground segmentation mask (b) and 2D joints (c) (note the left-wrist is missing). If the mask is not used, the reconstructed body is problematic (d); The mask helps build a correct pose (e).

The 3D part Orientation term in optimization. Fig. 16 shows an example with and without the 3D orientation term. The use of the 3D orientation term significantly reduces the reconstruction ambiguity of 3D poses.



Fig. 16: Importance of the 3D part orientation term. (left) input image; (middle) result with 3D part orientation term; (right) result without 3D part orientation term.

The IUV term in optimization. An IUV map plays important roles in both 3D body geometry reconstruction and pose estimation. We evaluate the importance of IUV by dropping off this term in shape reconstruction and pose estimation, respectively. Fig. 17(a) shows a side-by-side comparison while reconstructing an over-weighted lady. Using IUV



(a) Shape reconstructed with the IUV term (middle) manifests the body weight better than that without the IUV term (right).



(b) Pose reconstructed with (middle) and without (right) the IUV term.

Fig. 17: The importance of IUV term for shape and pose reconstruction.

term gives more accurate body model, because IUV terms impose detailed geometry model constraints from dense correspondences.

For pose reconstruction, Fig. 17(b) shows that reconstructed examples with and without IUV term. It is obvious that IUV term helps recover more accurate result, especially for body orientation.

7.5 Limitations

With no exceptions, our method suffers from several limitations. First, we observed failure cases when a significant part of the target person is either occluded by other objects or out of image boundary. Occlusion is the biggest issue and it imposes more challenge for RGB camera than depth camera based methods. Second, our method also fails for complicated or uncommon poses, particularly those in sports videos, such as gymnastics and skydiving. The main reason is that such data is not adequate in training dataset. Third, our system does not have specific hand pose detector (as did in [60]) and each hand is associated with only one joint, therefore the reconstructed hands are sometimes incorrectly oriented. Finally, our CNN does not handle multiple bodies at this moment, but can easily extended to support this. Solving the above mentioned problems points to interesting future directions.

8 CONCLUSIONS

We have presented a method for reconstructing the 3D pose and shape of a human, in a stable and consistent manner, from a single RGB video stream at more than 20 Hz. Our approach employs a multi-task CNN that regresses five human anatomical features simultaneously, which are further coupled with a kinematic pose reconstruction and shape modeling algorithm, producing a temporally stable 3D reconstruction of the full-body. In contrast to most existing approaches, our approach can operate on any input image fully automatically, without strict prescribed bounding boxes, and independent of expensive initialization. We test and evaluate our system in a variety of challenging real-time scenarios, including live streaming from commercial cameras, as well as in community videos. Results demonstrate that our approach compares to offline state-of-the-art monocular RGB methods qualitatively and advances the real-time 3D body reconstruction methods with a significant step.

REFERENCES

- [1] R. A. Güler, N. Neverova, and I. Kokkinos, "Densepose: Dense human pose estimation in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7297–7306.
- [2] 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/CVF International Conference on Computer Vision*, 2019, pp. 2252–2261.
- [3] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt, "VNect: Real-time 3D human pose estimation with a single RGB camera," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 44, 2017.
- [4] 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*, 2016, pp. 561–578.
- [5] M. Kocabas, N. Athanasiou, and M. J. Black, "Vibe: Video inference for human body pose and shape estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5253–5263.
- [6] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4724–4732.
- [7] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European conference on computer vision*, 2016, pp. 483–499.
- [8] X. Nie, J. Feng, Y. Zuo, and S. Yan, "Human pose estimation with parsing induced learner," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2100–2108.
- [9] 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.
- [10] H. Ci, C. Wang, X. Ma, and Y. Wang, "Optimizing network structure for 3d human pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2262–2271.
- [11] 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/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3425–3435.
- [12] 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*, vol. 32, no. 1, 2018.
- [13] G. Pavlakos, X. Zhou, and K. Daniilidis, "Ordinal depth supervision for 3D human pose estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7307–7316.
- [14] 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 *European Conference on Computer Vision (ECCV)*, Zürich, 2014, oral.

- [15] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.
- [16] 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.
- [17] R. Dabral, A. Mundhada, U. Kusupati, S. Afaque, A. Sharma, and A. Jain, "Learning 3d human pose from structure and motion," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 668–683.
- [18] 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 *2017 International Conference on 3D Vision (3DV)*, 2017, pp. 506–516.
- [19] 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.
- [20] I. Habibie, W. Xu, D. Mehta, G. Pons-Moll, and C. Theobalt, "In the wild human pose estimation using explicit 2d features and intermediate 3d representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10905–10914.
- [21] 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.
- [22] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Coarse-to-fine volumetric prediction for single-image 3D human pose," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7025–7034.
- [23] X. Sun, B. Xiao, F. Wei, S. Liang, and Y. Wei, "Integral human pose regression," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 529–545.
- [24] C. Luo, X. Chu, and A. Yuille, "Orinet: A fully convolutional network for 3d human pose estimation," *arXiv preprint arXiv:1811.04989*, 2018.
- [25] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "SCAPE: Shape completion and animation of people," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 408–416, 2005.
- [26] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM transactions on graphics (TOG)*, vol. 34, no. 6, p. 248, 2015.
- [27] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3D hands, face, and body from a single image," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [28] P. Guan, A. Weiss, A. O. Bălan, and M. J. Black, "Estimating human shape and pose from a single image," in *IEEE International Conference on Computer Vision*, 2009, pp. 1381–1388.
- [29] 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 *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6050–6059.
- [30] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7122–7131.
- [31] 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.
- [32] H. Joo, N. Neverova, and A. Vedaldi, "Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation," *arXiv preprint arXiv:2004.03686*, 2020.
- [33] C. Bregler, J. Malik, and K. Pullen, "Twist based acquisition and tracking of animal and human kinematics," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 179–194, 2004.
- [34] 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.
- [35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [36] S. Shimada, V. Golyanik, W. Xu, and C. Theobalt, "Physcap: Physically plausible monocular 3d motion capture in real time," *ACM Transactions on Graphics*, vol. 39, no. 6, dec 2020.
- [37] W. Xu, A. Chatterjee, M. Zollhöfer, H. Rhodin, D. Mehta, H.-P. Seidel, and C. Theobalt, "MonoPerfCap: Human performance capture from monocular video," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 2, p. 27, 2018.
- [38] M. Habermann, W. Xu, M. Zollhöfer, G. Pons-Moll, and C. Theobalt, "LiveCap: Real-time human performance capture from monocular video," *ACM Trans. Graph.*, vol. 38, no. 2, pp. 14:1–14:17, 2019.
- [39] M. Habermann, W. Xu, M. Zollhoefer, G. Ponsmoll, and C. Theobalt, "Deepcap: Monocular human performance capture using weak supervision," *arXiv: Computer Vision and Pattern Recognition*, 2020.
- [40] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
- [41] R. Caruana, *Learning to learn*. Springer, 1998, ch. "Multitask learning", pp. 95–133.
- [42] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7291–7299.
- [43] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," *IEEE conference on computer vision and pattern recognition*, 2016.
- [44] A.-I. Popa, M. Zanfir, and C. Sminchisescu, "Deep multitask architecture for integrated 2D and 3D human sensing," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [45] 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.
- [46] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using part affinity fields," *IEEE conference on computer vision and pattern recognition*, 2018.
- [47] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.
- [48] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.
- [49] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7482–7491.
- [50] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
- [51] Y. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, vol. 3. IEEE, 1999, pp. 1945–1950.
- [52] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Efficient model-based 3D tracking of hand articulations using kinect," in *The British Machine Vision Conference (BMVC)*, 2011, pp. 1–11.
- [53] 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 (ECCV)*, September 2018.
- [54] 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, 2013.
- [55] Y. Cai, L. Ge, J. Cai, and J. Yuan, "Weakly-supervised 3D hand pose estimation from monocular RGB images," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 666–682.
- [56] M. Loper, N. Mahmood, and M. J. Black, "Mosh: Motion and shape capture from sparse markers," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 6, pp. 1–13, 2014.
- [57] K. Zhou, X. Han, N. Jiang, K. Jia, and J. Lu, "Hemlets pose: Learning part-centric heatmap triplets for accurate 3d human pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2344–2353.
- [58] H. Joo, N. Neverova, and A. Vedaldi, "Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation," *arXiv preprint arXiv:2004.03686*, 2020.

- [59] 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.
- [60] 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.



Juntao Ye was awarded his B.Eng from Harbin Engineering University in 1994, MSc from Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences in 2000, and his PhD in Computer Science from The University of Western Ontario, Canada, in 2005. He is currently an associate professor with National Laboratory of Pattern Recognition of the Institute of Automation, Chinese Academy of Sciences. His research interests include graphics, particularly physically-based simulation of cloth and fluid, as well as image/video processing.



Liguo Jiang received the B.Eng degree in software engineering from Chongqing University in 2015. He is currently working toward the PhD degree in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include deep learning, human motion capture and cloth simulation.



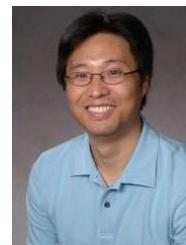
Miaopeng Li is a Ph.D. student at the State Key Lab of CAD&CG, Zhejiang University, China. She received her bachelor degree from Northwestern Polytechnical University in 2016. Her research interests include marker-less human motion capture, human pose estimation, 3D reconstruction and their applications.



Xinguo Liu received the BS and PhD degrees in applied mathematics from Zhejiang University in 1995 and 2001, respectively. He is a professor at the School of Computer Science and Technology, Zhejiang University. He was with Microsoft Research Asia in Beijing during 2001–2006, and then joined in Zhejiang University. His main research interests are in graphics and vision, particularly geometry processing, realistic and image-based rendering, and 3D reconstruction.



Jianjie Zhang received PhD degree in computer science from Texas A&M University (TAMU). He is currently a R&D director in Xmov ai Inc. His primary research is in the area of computer graphics and vision, including human body modeling and tracking, human body dynamics simulation, human face modeling and tracking and etc.



Jinxiang Chai received PhD degree in computer science from Carnegie Mellon University(CMU). He is currently an associate professor in the Department of Computer Science and Engineering at Texas A&M University. His primary research is in the area of computer graphics and vision with broad applications in other disciplines such as virtual and augmented reality, robotics, human computer interaction, and biomechanics. He received an NSF CAREER award for his work on theory and practice of



Congyi Wang received the PhD degree in computer science from Institute of Computing Technology, Chinese Academy of Sciences in Jan 2017. Since 2018, he has been a research scientist at XMov, a startup company aiming at AI powered virtual production line. His research interests include computer animation, computer graphics, computer vision and speech signal processing.

Bayesian motion synthesis.