Supplementary Material for DeepHuman: 3D Human Reconstruction from a Single Image

A. Overview

This supplementary document provides technical details that are supplementary to the main paper. We first describe how we obtain the dense semantic representation given an image of a person in Sec.B. We then present details about the network implementation in Sec.C. Details about the data capture system as well as our THuman dataset are presented in Sec.D. We elaborate the comparison setup in Sec.E. In Sec.F, we add a comparison experiment against two relevant methods, both of which require multiview images as input. Finally we show a failure case in Sec.G.

B. SMPL Estimation and Dense Semantic Representation

Our method starts from SMPL estimation from the input image, and generates a dense semantic representation from the estimated SMPL model. To estimate a body from the image, we exploit the state-of-the-art methods HMR[6] and SMPLify[3], which are both capable of inferring the shape and pose parameters of SMPL[9] from a single image. HMR infers SMPL model with a neural network, while SMPLify aligns SMPL model with the keypoint detection results through non-convex optimization. We found that the two methods have complementary characteristics: the predictions of HMR are always plausible but not well-aligned with the color image, while SMPLify aligns the SMPL model with detected keypoints very well but relies on initialization to output plausible results. Therefore, we combine HMR and SMPLify to obtain an SMPL estimation as accurately as possible. Specifically, we first use HMR to obtain an initial SMPL estimation, and then improve its accuracy using SMPLify. Before applying SMPLify, we use AlphaPose[17] to detect keypoints on the image.

The estimated shape and pose parameters determine a polygon mesh representation of the body through linear shape blending and pose skinning[9]. However, it is hard to feed the polygon mesh into a deep neural network. Therefore, inspired by "Vitruvian Manifold" [14], we introduce a dense semantic representation generated from SMPL. Specifically, we predefine a *semantic code* $C(v)$ for a vertex v on SMPL according to its spatial coordinate at rest pose:

$$C(v) = \left(\frac{x_0(v) - x_{min}}{x_{max} - x_{min}}, \frac{y_0(v) - y_{min}}{y_{max} - y_{min}}, \frac{z_0(v) - z_{min}}{z_{max} - z_{min}} \right), \tag{1}$$

where $(x_0(v), y_0(v), z_0(v))$ is the spatial coordinate of v on a SMPL in mean shape at rest pose, and $[x_{min}, x_{max}] \times [z_{min}, z_{max}] \times [z_{min}, z_{max}]$ is the bounding of that SMPL model. Given a SMPL model corresponding to a human image, we render the semantic code onto the image plane to obtain a semantic map $\mathbf{M}_s$ and generate a semantic volume $\mathbf{V}_s$ by first voxelizing the SMPL model into the voxel grid and then propagating the semantic codes into the occupied voxels. $\mathbf{M}_s$ and $\mathbf{V}_s$ make up the dense semantic representation for the input image.

We use semantic maps/volumes instead of binary mask because we believe that such a dense semantic representation can provide the CNN clues about the correspondences between the 2D image plane and 3D space. To be more specific, a pixel/voxel with a semantic code can be mapped to a corresponding point on the SMPL surface that has an identical code. In this way we obtain a bidirectional relationship between the volume and the image using the SMPL surface as a bridge.

C. Network Implementation Details

The volume-to-volume network $\mathcal{H}$ takes as input a semantic volume with $128 \times 192 \times 128$ resolution, and outputs an occupancy volume with the same shape. The image encoder $\mathcal{G}$ concatenates as input the given RGB image and the corresponding semantic map, both of which have a resolution of 192×128. Our normal refinement U-Net $\mathcal{R}$ takes as input the concatenation of the RGB image, semantic map and upsampled normal projection result, and the input/output resolution of $\mathcal{R}$ is 384×256. The architecture details are shown in Tab.1.

During network training, the parameters are set to $\lambda_{FS} = \lambda_{SS} = 0.1, \lambda_N = 0.01, \gamma = 0.7$. We exploit a two-stage training procedure: first pre-train the vol2vol network and the normal refinement network, and then fine-tune them jointly with the combined loss. We used Adam [8] with default parameters as the optimizer. The learning rate is fixed

Table 1. Network Architecture Details.

Net	Layer	Kernel	Stride	Output
$\mathcal{G}$	conv+lrelu	4	2	$96 \times 64 \times 8$
	conv+lrelu	4	2	$48 \times 32 \times 16$
	conv+lrelu	4	2	$24 \times 16 \times 32$
	conv+lrelu	4	2	$12 \times 8 \times 64$
	conv+lrelu	4	2	$6 \times 4 \times 128$
$\mathcal{H}$	conv+lrelu	4	2	$64 \times 96 \times 64 \times 8$
	conv+lrelu	4	2	$32 \times 48 \times 32 \times 16$
	conv+lrelu	4	2	$16 \times 24 \times 16 \times 32$
	conv+lrelu	4	2	$8 \times 12 \times 8 \times 64$
	conv+lrelu	4	2	$4 \times 6 \times 4 \times 128$
	transconv+lrelu	4	2	$8 \times 12 \times 8 \times 64$
	transconv+lrelu	4	2	$16 \times 24 \times 16 \times 32$
	transconv+lrelu	4	2	$32 \times 48 \times 32 \times 16$
	transconv+lrelu	4	2	$64 \times 96 \times 64 \times 8$
	transconv+lrelu	4	2	$128 \times 192 \times 128 \times 4$
	conv+sigmoid	3	1	$128 \times 192 \times 128 \times 1$
$\mathcal{R}$	conv+lrelu	4	2	$192 \times 128 \times 16$
	conv+lrelu	4	2	$96 \times 64 \times 32$
	conv+lrelu	4	2	$48 \times 32 \times 32$
	conv+lrelu	4	2	$24 \times 16 \times 32$
	conv+lrelu	4	2	$12 \times 8 \times 32$
	transconv+lrelu	4	2	$24 \times 16 \times 32$
	transconv+lrelu	4	2	$48 \times 32 \times 32$
	transconv+lrelu	4	2	$96 \times 64 \times 32$
	transconv+lrelu	4	2	$192 \times 128 \times 16$
	transconv+lrelu	4	2	$384 \times 256 \times 8$
	conv+tanh	3	1	$384 \times 256 \times 3$

* The term "conv" is convolution for short, "transconv" is transposed convolution and "lrelu" is Leakly ReLU.

to 2e-4 during the whole training procedure, and the batch size is set to be 4. Training our network takes about 1 day for 18 epochs on a single TITAN X GPU. Given a single image and its SMPL estimation, it takes about 147 ms to execute our network. Visible mesh refinement (Line 297) using a non-optimized solver takes 4 min, which could be reduced to seconds if parallelizing the solver on GPU.

D. Data Capture System and THuman Dataset

Our capture system is based on the single-view RGB-D DoubleFusion [18] technique. DoubleFusion utilizes a double-layer representation and incorporates a motion prior derived from the SMPL [9]. It simultanuously solves skeleton motions and non-rigid deformation according to the depth observation at the current frame. After getting the motion field, depth pixels in the current frame are fused into a reference volume as described in [12]. As the observed surface is gradually fused and deformed, the shape and pose parameters of the body layer are also gradually optimized through volumetric shape-pose optimization. In this way the two layers can benefit from each other, leading to robust tracking and accurate reconstruction.

The available DoubleFusion technique performs only robust fusion of detailed surface geometries. To obtain full-body texture, we can directly perform color or albedo fusion in a similar way to depth fusion. However, the fused texture blurs when fast body motion occurs. Thus we develop a two-stage capture procedure. In the first stage, the subject actors are required to rotate slowly and perform some basic surface completion motions to obtain a surface geometry that is as complete as possible and clear texture recovery of the surface as well. After that, in the second stage, we disable geometry fusion and texture update, but still perform the non-rigid surface registration based on the input depth information. In this way, we still capture non-rigid motion details of the subject's surface.

In order to obtain human mesh data under natural but diverse poses, our system presents to the subject a reference pose randomly sampled from MOSH[10] dataset every 6 seconds and requires the performer to imitate the reference pose in the second stage. Note that the 6-second interval is usually long enough for subjects to recognize the presented pose and prepare for imitation. At the end of every 6-second interval, the system automatically saves the RGBD image, the 3D surface mesh and its corresponding SMPL model in the current live pose. After data capture, we post-process the raw meshes through hole filling [7], remeshing [5] and isolated artifact removal.

After approximately 70 hours of data capture using only one capture setup, we achieve capturing and reconstruction of 230 subject characters, with each character corresponding to about 30 poses. This data leads to 7000 data items in our THuman dataset; some examples are shown in Fig. 1. Each item contains a textured surface mesh, a RGBD image from the Kinect sensor, and an accompanying well-aligned SMPL model. Note that the topology of the textured models is not the same for the variety of body shapes and clothing styles.

As mentioned in the main paper, the training corpus are synthesized using the textured surface meshes and the accompanied SMPL models. We use the textured surface meshes to generate color images and ground-truth occupancy volumes, and use the accompanied SMPL models to generate semantic maps and volumes. To augment the training data, we apply random perturbations to the shape and pose parameters of the SMPL models. We also apply random cropping and random brightness adjustments to color images during network training.

E. Comparison Experiment Details

E.1. Competing Approaches

We compare our method against three state-of-the-art deep learning based approaches for single view 3D human reconstruction: HMR[6], BodyNet[16] and SiCloPe[11]. To eliminate the effect of dataset bias, we fine-tuned the pre-trained model of HMR[6] and BodyNet[16] with the same training data as we use to train our network. Since SiCloPe is not open-source, we are unable to finetune it and hence

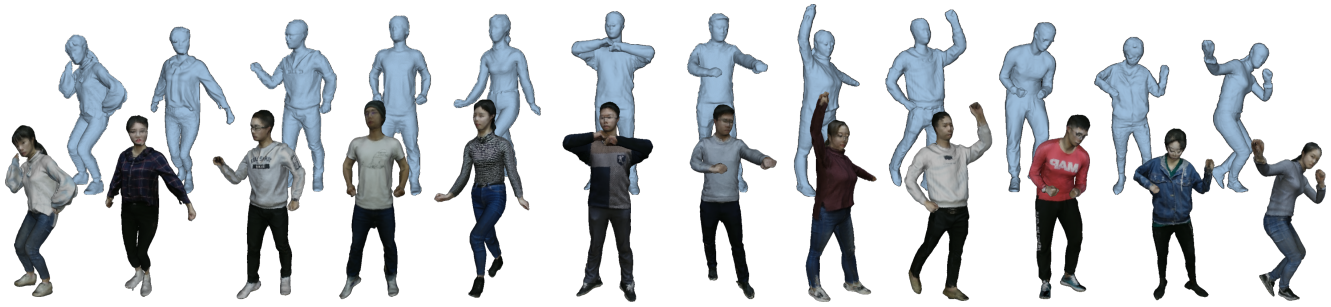

Figure 1. Example meshes sampled from our dataset.

only present qualitative comparison in the comparison section.

(1) **HMR**. In [6], Kanazawa et al. proposed a neural network to directly regress the shape and pose parameter of SMPL from an RGB image. The output of HMR is a 75-D vector, which can be used to generate a triangular mesh of SMPL through linear shape blending and pose skinning[9]. It is the state-of-the-art among available methods for single-view pose and shape estimation[6, 3, 4, 13, 15]. We fine-tuned the pretrained HMR model using the color images and the corresponding ground-truth shape/pose parameters in our synthetic training data.

(2) **BodyNet**. BodyNet is a neural network for direct inference of volumetric body shape from a single image. The output of BodyNet is a $128 \times 128 \times 128$ occupancy volume with similar definition in Sec.3. BodyNet is the most related work to this paper. We fine-tuned the whole network of BodyNet using the color images and the ground-truth occupancy volumes in our training set.

(3) **SiCloPe**. SiCloPe[11] is another voxel-based method, but it recovers certain details by synthesizing multiview silhouettes of the subject given the input silhouette and the 3D skeleton pose of the subject.

E.2. Comparison Metrics

(1) For **qualitative comparison**, we feed all the networks with the same images and convert the network output into a triangular mesh. The results are shown in Fig.5 in the main paper.

(2) The **quantitative comparison** is conducted on the testing set of our synthetic data. We convert the output of HMR to occupancy volumes with a resolution of $128 \times 192 \times 128$. We also upsampled the output of BodyNet by a factor of 1.5 using trilinear interpolation, and then crop the volume to make it have the same resolution. After that we use the mean Intersection-over-Union (IoU score) between predicted 3D volumes and their ground-truth as the comparison metric. It should be noted that the predicted volume and the ground-truth may be unaligned along the z-axis because of depth ambiguities. Therefore, we shift the predicted volume along z-axis to search for the best alignment (i.e., to maximize IoU score between the ground-truth volumes and the predict ones), and regard the maximum IoU score as the final score.

We also consider using the Chamfer Distance (CD) or Earth Movers Distance (EMD) as an additional metric. However, both metrics are computationally heavy as they require distance calculation between two high resolution meshes containing large number of points. Therefore, we did not use CD or EMD for evaluation in our experiments.

(a) (b)

Figure 2. Comparison against [2](a) and [1](b). For each comparison, from left to right: input image, result of the compared method, and our result. Please zoom in to view the detailed surfaces that we reconstructed.

F. More comparison

In this section, we compare our method against [2] and [1]. Note that they both require multiple images or a video sequence from a camera as input, while our method can reconstruct human under various poses using only a single image. We carry out the comparison in Fig.2 using the example data in their open-sourced projects. Note that our method achieves more reasonable shape reconstructions on these A-pose inputs.

G. Failure Case

As mentioned in the main paper, our method relies on HMR and SMPLify to generate a dense semantic representation from SMPL model. Consequently, we cannot give an accurate reconstruction if the estimation of SMPL model is erroneous. Here we show an example in Fig.3. However,

Figure 3. A failure case. HMR gives wrong prediction of the subject's upper body pose (middle), which results into wrong reconstruction by our network (right).

the reliance on SMPL estimation ensures our robustness. Furthermore, the last two years have witnessed a rapid development in this topic for better human shape and pose estimation from a single image. We believe that the dependency on SMPL estimation will not be a bottleneck in the future.

References

[1] T. Alldieck, M. Magnor, B. L. Bhatnagar, C. Theobalt, and G. Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 2019.

[2] T. Alldieck, M. A. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll. Video based reconstruction of 3d people models. In *IEEE CVPR*, 2018.

[3] 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 *ECCV*, pages 561–578, 2016.

[4] P. Guan, A. Weiss, A. O. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *IEEE ICCV*, pages 1381–1388, 2009.

[5] W. Jakob, M. Tarini, D. Panozzo, and O. Sorkine-Hornung. Instant field-aligned meshes. *ACM Trans. Graph (Proc. SIGGRAPH ASIA)*, 34(6), Nov. 2015.

[6] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *IEEE CVPR*, 2018.

[7] M. Kazhdan and H. Hoppe. Screened poisson surface reconstruction. *Acm Trans. Graph*, 32(3):1–13, 2013.

[8] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[9] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015.

[10] M. M. Loper, N. Mahmood, and M. J. Black. MoSh: Motion and shape capture from sparse markers. *ACM Tran. Graph. (Proc.SIGGRAPH Asia)*, 33(6):220:1–220:13, Nov. 2014.

[11] R. Natsume, S. Saito, Z. Huang, W. Chen, C. Ma, H. Li, and S. Morishima. Siclope: Silhouette-based clothed people. *CoRR*, abs/1901.00049, 2019.

[12] R. A. Newcombe, D. Fox, and S. M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *IEEE CVPR*, 2015.

[13] V. Tan, I. Budvytis, and R. Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. In *BMVC*, 2017.

[14] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon. The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation. In *IEEE CVPR*. IEEE, June 2012.

[15] H. Tung, H. Tung, E. Yumer, and K. Fragkiadaki. Self-supervised learning of motion capture. In *NIPS*, pages 5242–5252, 2017.

[16] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid. BodyNet: Volumetric inference of 3D human body shapes. In *ECCV*, 2018.

[17] Y. Xiu, J. Li, H. Wang, Y. Fang, and C. Lu. Pose Flow: Efficient online pose tracking. In *BMVC*, 2018.

[18] T. Yu, Z. Zheng, K. Guo, J. Zhao, Q. Dai, H. Li, G. Pons-Moll, and Y. Liu. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. In *IEEE CVPR*, June 2018.