Feature Bridging Networks for 3D Human Body Shape Estimation from a Single Depth Map

Naoshi Kaneko*, Mei Oyama†, Masaki Hayashi‡, Seiya Ito§, and Kazuhiko Sumi*
* Department of Integrated Information Technology, Aoyama Gakuin University, Kanagawa, Japan
†Innovation/Research and Development Division, Ricoh Company, Ltd., Kanagawa, Japan
‡Department of Electronics and Electrical Engineering, Keio University, Kanagawa, Japan
§Graduate School of Science and Engineering, Aoyama Gakuin University, Kanagawa, Japan
*{kaneko, sumi}@it.aoyama.ac.jp †mei.oyama@jp.ricoh.com
‡mhayashi@aoki-medialab.jp §ito.seiya@vss.it.aoyama.ac.jp

***Abstract*—This paper describes a novel deep neural network architecture to reconstruct an accurate human body shape from a single depth map. The proposed method utilizes a statistical parametric body shape model, which represents a wide variety of body shape with low-dimensional body shape parameters. We formulate the body shape reconstruction as a regression problem of the body shape parameters. One of the biggest challenges of the single-image shape reconstruction lies in a gap between input and output modalities. This is because an input depth map only contains a surface of a human body, while the output is a full 3D body shape model. To bridge this gap, we utilize dedicated two deep neural networks ShapeEncoder and DepthMapEncoder, which respectively process the 3D body model and the depth map. These two networks are bridged with a learned latent body feature space to enable accurate single-image body shape estimation. Furthermore, the proposed method also uses body joint positions estimated from the depth map to further improve the performance. The proposed approach is evaluated on real depth maps taken from 30 subjects and achieves significant performance improvements over the existing methods.**

***Contribution*—Multiple deep neural networks form a novel feature bridging architecture to achieve significant performance improvements.**

***Keywords*—human body shape estimation; convolutional neural networks; feature representation; depth map**

I. Introduction

Each person has a unique body shape. Body shape is a fundamental and important factor for human recognition, as the shapes provide various informations including personality impression [1], health state [2], gender attribution [3], and so on. Also, the body size of a person is crucial to provide personalized services. However, due to the complexity and expensiveness of equipment, scanning time, and the requirement for nakedness, current body scanning systems are not suitable for personal use. If a body shape can be estimated from images of a dressed human, various applications, such as clothing recommendation, become easier and more effective.

Human body shape estimation has been extensively studied in the field of computer vision. The problem aims to recover an accurate 3D human body mesh model from image inputs. Estimation methods can be broadly divided into two categories: multi-view and single-view approaches.

Multi-view approaches have a longer research history than single-view approaches. Multi-view methods use multiple images taken from different viewpoints as inputs. Some early studies use multi-view silhouettes and estimate the body shape by using a technique called shape-from-silhouettes [4]. Later, due to the emergence of consumer depth sensors, RGB-D image-based approaches were actively proposed [5], [6].

On the other hand, single image-based methods require minimal image input. Thus, they are better suited for personal use. This paper focuses on methods that use a single still image as input [7]–[11]. To recover accurate 3D shapes, most approaches employ statistical human body shape models [12], [13]. Statistical shape models support mesh deformation with learned, low-dimensional body shape parameters. These models have a so-called template model, which can represent various body shapes by changing the parameters. Thus, the body shape estimation problem is formulated as an estimation of the low-dimensional body shape parameters, instead of directly deriving the actual shape of a mesh model. This formulation reduces the complexity of the problem.

The input representations for single-image methods are broadly categorized into two types: joint coordinates and images. In joint-based algorithms, first 2D joint coordinates are estimated from an input image with human pose estimation algorithms, then statistical models are deformed to minimize 2D joint errors between projected model joints and the estimated 2D coordinates [7], [9], [11]. As the joint coordinates describe a person's limb lengths well, these methods provide a 3D model with relatively accurate limb lengths (e.g. body height or arm lengths). In image-based methods, the silhouette of a person is commonly used as input [8], [10]. These approaches estimate a 3D model that fits to the outline of a person. Thus, they can estimate a model with relatively better circumference estimates (e.g. waist or hip).

A common problem of these methods is that they only use 2D joint positions or silhouettes. These 2D features imply input-level depth ambiguity, e.g. it is hard to estimate "how

978-1-7281-9331-1/20/$31.00

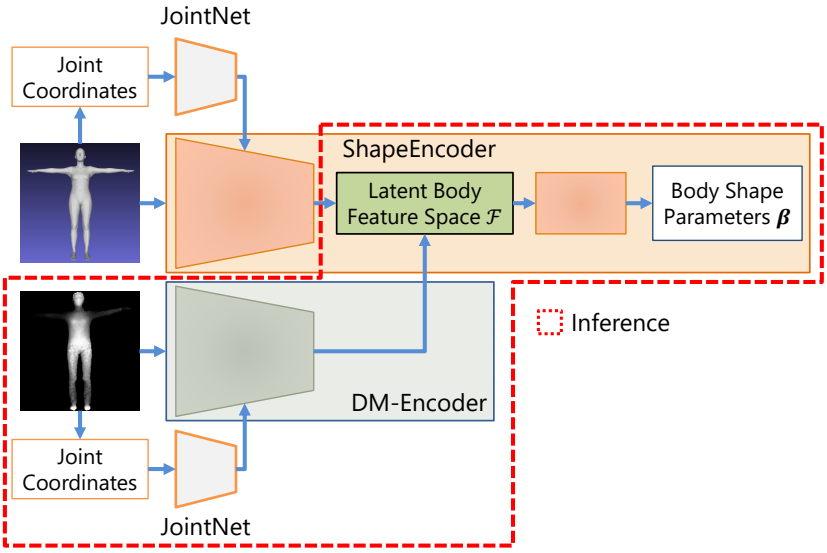

Fig. 1: **Overview of the proposed network architecture.** The proposed method uses two deep neural networks, ShapeEncoder and DepthMapEncoder (DM-Encoder). ShapeEncoder is trained with 3D mesh models to capture latent representations of the 3D body shapes. This latent representations are stored in high-dimensional space, which we call latent body feature space $\mathcal{F}$. DM-Encoder learns mappings from depth maps to the corresponding latent representations. At inference time, by taking a depth map as an input, DM-Encoder maps the input depth map to the latent representation. Then, ShapeEncoder estimates the body shape parameters $\boldsymbol{\beta}$ of a statistical body model from the latent representation.

much potbelly" a person has, just from a 2D image or joint coordinates. Also, all of the above methods except the CNN-based method [10] employ iterative optimization algorithms. Thus the inference takes a long time to convergence. On the other hand, CNN-based methods can provide instantaneous estimation [10]. However, CNNs have their own problem, which is known as the locality of convolutional operations [14]. This characteristic leads to larger errors in non-local parts, such as body height or leg length.

To overcome the above limitations, this paper proposes a human body shape estimation method based on deep neural networks. There are three key design choices: 1) using a single depth map; 2) employing two deep neural networks; and 3) incorporating joint coordinates.

First, the proposed method employs a single depth map as an input. Since depth maps store pixel-level metric distances, they can represent a person's surface shape. This representation reduces the input-level ambiguity and provides more detailed information about a subject's shape.

Second, the shape estimator is composed of two deep neural networks, namely ShapeEncoder and DepthMapEncoder (DM-Encoder), which are bridged with a learned latent body feature space. The key choice of the network design is to bridge the modality gap between target 3D mesh models and 2.5D input depth maps. Our insight is that the problem of 3D human body shape estimation is a multi-modal learning problem. The first modality is a 2.5D input depth map and the second modality is a 3D body shape model. Through the rapid development of deep neural networks, researchers have shown that different modalities generally require different network architectures to achieve good performance. Therefore, we have designed the network in this way, i.e., using 2D CNN for the first modality and 3D network for the second modality. The proposed method uses a two-stage training process. In the first stage, ShapeEncoder is trained with 3D mesh models to capture useful, reduced latent representations of the 3D body shapes. Then in the second stage, DM-Encoder takes depth maps corresponding to the 3D models, and learns mappings from the depth maps to the latent representation. At the inference time, the mapped feature is used to estimate the body shape parameters of a statistical body model.

Third, the proposed method also uses 3D joint coordinates estimated from the depth map to further improve the performance. To this end, we introduce an auxiliary network, JointNet, to extract features from the joint coordinates. Fig. 1 illustrates the overview of the proposed network architecture.

The proposed approach is evaluated on real depth maps taken from 30 subjects (15 males and 15 females) and shows the improved performance over the existing approaches. Note that, since the proposed method is intended for personal use, the evaluation is conducted with T-pose, assuming that the users are cooperative.

II. Related Work

Guan et al. [7] proposed a method to recover a 3D human body model from a single RGB image. Taking 2D human joint positions given by a user as inputs, the method first estimates a 3D pose from the 2D joints, and transforms a parametric human body model based on that pose. Then the

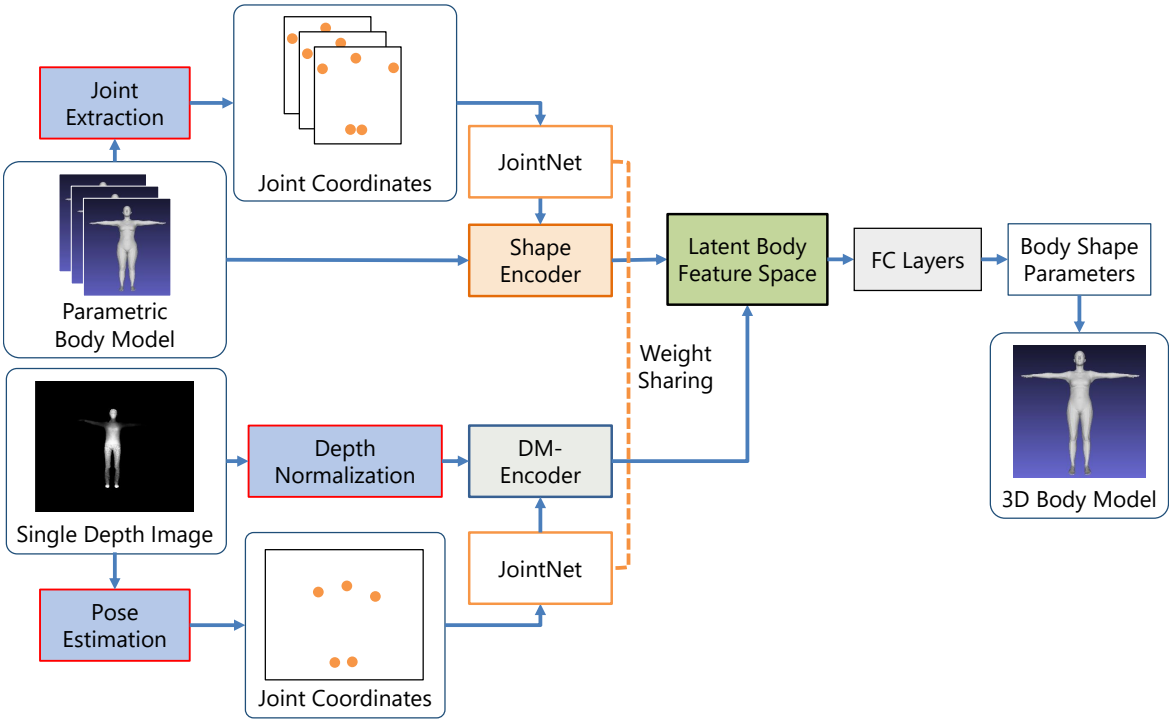

Fig. 2: **Process flow of the proposed method.** First, synthetic body mesh models of males and females are generated by using parametric body model SMPL. Next, the vertex and joint coordinates of the generated mesh models are fed to ShapeEncoder. ShapeEncoder is trained to estimate the corresponding body shape parameters that generated the mesh model. Through the training, the network acquires a latent body feature space in its intermediate layer. Finally, DM-Encoder is trained to map a depth map and the estimated joint coordinates to the corresponding latent feature vector. Before feeding the joint coordinates to ShapeEncoder and DM-Encoder, these coordinates are fed to auxiliary network JointNet to extract features.

transformed model is refined with a silhouette and edges of human regions in an RGB image. Hasler et al. [8] proposed a method that takes a person's silhouette as an input. This method also requires users to input corresponding points between the silhouette and a template human body model. The method solves an optimization problem to minimize the correspondence error. The above methods require manual user inputs. Bogo et al. [9] proposed SMPLify, a fully automatic 3D posture and body shape estimation method using an RGB image as an input. A parametric body model skinned multi-person linear model (SMPL) [13] is employed in this method. SMPLify first estimates a human's 2D joint positions from an RGB image [15], and performs iterative fitting to minimize the 2D distance between the 2D joint position and the projected joint position of the SMPL. Later, the authors extended SMPLify to SMPLify-X [11]. SMPLify-X takes 2D joint positions estimated by OpenPose [16] and iteratively fit an expressive SMPL (SMPL-X) model to the given joints. These methods succeed to estimate 3D pose and body shape from a single image. However, these methods are computationally expensive, because they solve iterative optimization problems to fit a 3D human model.

On the other hand, recent methods estimate a 3D human body model in a short time by using CNNs. Dibra et al. [10] showed that a 3D body model can be directly estimated from a single silhouette image by using two types of CNNs. The first CNN learns a feature space of 3D body models from 3D shape descriptors (heat kernel signatures (HKS) [17]), and the second CNN maps an input silhouette image to the learned feature space. Once the networks are trained, the inference time is less than 1 second when using a GPU. This method shares several design principles with our proposed method, i.e. fast inference with neural networks, single image input, and a combination of two networks. However, as described before, there are problems with 2D silhouette ambiguity and local convolutional operations. Also, Dibra et al. conducted a quantitative evaluation only on synthesized data, and did not report the estimation accuracy on real-world data.

The proposed method solves these problems by representing the surface shape of a human with depth maps, and by using 3D joint coordinates as global features. In addition, its performance was evaluated with real human data of 15 males and 15 females each.

III. Method

Fig. 2 illustrates the process flow of the proposed method. The proposed method takes a single depth map of a person as an input and estimates the body shape parameters of a parametric body model to estimate a 3D human body shape model.

We use the skinned multi-person linear model (SMPL) [13] as the parametric body model. In the SMPL, a body model M is defined as a function of the body shape parameters $\boldsymbol{\beta}$ and the pose parameters $\boldsymbol{\theta}$, defined as $M(\boldsymbol{\beta}, \boldsymbol{\theta})$. M is represented as a mesh model having 6,890 vertices and 23 joints. Note that, since this work focuses on the body shape estimation of a T-posed person, we set $\boldsymbol{\theta}$ to $\mathbf{0}$, which represents a T-pose.

The proposed method is composed of two different deep neural networks, namely ShapeEncoder and DepthMapEncoder (DM-Encoder), which are trained individually. Furthermore, ShapeEncoder and DM-Encoder are equipped with an auxiliary network, JointNet, to extract additional features. To train the above three networks, we use three data modalities: 1) mesh vertices of a body model (for ShapeEncoder); 2) projected depth map of a body model (for DM-Encoder); and 3) 3D body joint coordinates (for JointNet).

We purely use synthetic data to train these three networks. First, by randomly varying the body shape parameters $\boldsymbol{\beta}$ of the SMPL, gender-specific synthetic mesh models of 2,500 males and 2,500 females are generated. From the generated mesh models, we extract 6,890 mesh vertices and five 3D body joint coordinates. Then, we use a simulated time-of-flight camera [18] to capture the corresponding depth maps of the mesh models.

The proposed method is trained with a two-stage process. In the first stage, we train ShapeEncoder. The extracted vertices and joint coordinates of a mesh model are respectively fed as inputs to ShapeEncoder and JointNet. JointNet extracts useful features from the joint coordinates and passes the extracted features to ShapeEncoder. ShapeEncoder is trained to estimate the corresponding body shape parameters $\boldsymbol{\beta}$ that generated the mesh model. Through the training, the network acquires a latent body feature space $\mathcal{F}$ in its intermediate layer. This latent space $\mathcal{F}$ is represented as a 256-dimensional vector, i.e., $\mathcal{F} = (f_1, f_2, ..., f_{256})$. This process may be referred to as a kind of representation learning.

In the second stage, DM-Encoder is trained. This stage uses the all three modalities. First, we freeze the parameters of trained ShapeEncoder and JointNet. Then, the vertices and joint coordinates of a mesh model are fed to the frozen networks. The networks map the inputs to the latent body feature space $\mathcal{F}$. This space is used as a bridge between the two modalities, i.e., a 2.5D input depth map and a 3D body model. Specifically, DM-Encoder is trained to map an input depth map to the corresponding latent body feature space, which is located in ShapeEncoder. We call this design feature bridging networks, which bridge the features from different modalities at the intermediate network layer. Note that, similar to ShapeEncoder, JointNet also passes the extracted features to DM-Encoder.

During the inference, the proposed architecture feeds a depth map along with 3D joint coordinates to DM-Encoder. DM-Encoder maps the inputs to the latent feature space, then the latter part of ShapeEncoder estimates $\boldsymbol{\beta}$ from the mapped features. An output body model is generated by morphing the template SMPL model with the estimated $\boldsymbol{\beta}$.

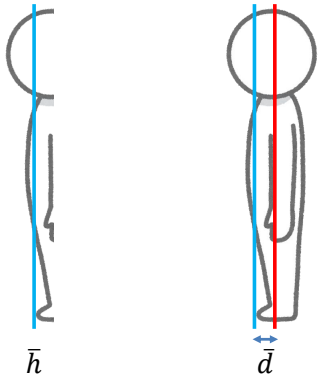

Fig. 3: **Depth value adjustment.** Front surface scanned with a depth camera (left) and a 3D body model (right) are depicted. The blue line shows the average depth value of the depth map, while the red line indicates the depth center of the 3D model.

A. Training Data

To train the proposed networks, mesh models of various body shapes and corresponding depth maps are required. In this paper, mesh models randomly generated with SMPL [13] are used as training data. In SMPL, the shape of a model is controlled by a 10-dimensional vector $\boldsymbol{\beta} = (\beta_1, \beta_2, ..., \beta_{10})$, where $\boldsymbol{\beta} = \mathbf{0}$ represents the template shape. The training samples are generated by randomly changing each parameter $\beta_{1\text{–}10}$, drawn from a normal distribution with zero mean and variance of σ^2. Since β_1 and β_2 have a strong effect on shape deformation, they are drawn with $\sigma^2 = 0.7$, while others are drawn with $\sigma^2 = 1.0$. All the generated models have the same number of mesh vertices (6,890) with the same indices. The proposed approach extracts $k = 5$ joint coordinates, namely head, hand tips, and foot tips, which represent global features in T-poses. The models are captured with a simulated time-of-flight camera (BlenSor [18]) to obtain depth maps.

B. Depth Map Normalization

Since depth maps contain per-pixel distances between the camera and the subject, the values may change according to the camera location. Thus, depth map normalization is performed before feeding the depth maps to the networks. First, the height H of the human body in a depth map is calculated as the distance between the head top and toes, and a human region of $H \times H$ is cropped and scaled to 256×256 pixels. As illustrated in Fig. 3, a depth sensor measures only the front surface of a person, while a 3D body model has full surfaces. Thus, the depth values are adjusted so that the reference value of the depth is at the center of the person. This operation is performed by $\bar{p}_i = p_i - (\bar{h} + \bar{d})$, where p_i is the depth value at pixel i, $\bar{h}$ is the average depth value of the input depth map, and $\bar{d}$ is the average difference between the center of the 3D model and the corresponding depth map values. Lastly, min-max normalization is applied to the adjusted depth values, with a range of ± 30 cm.

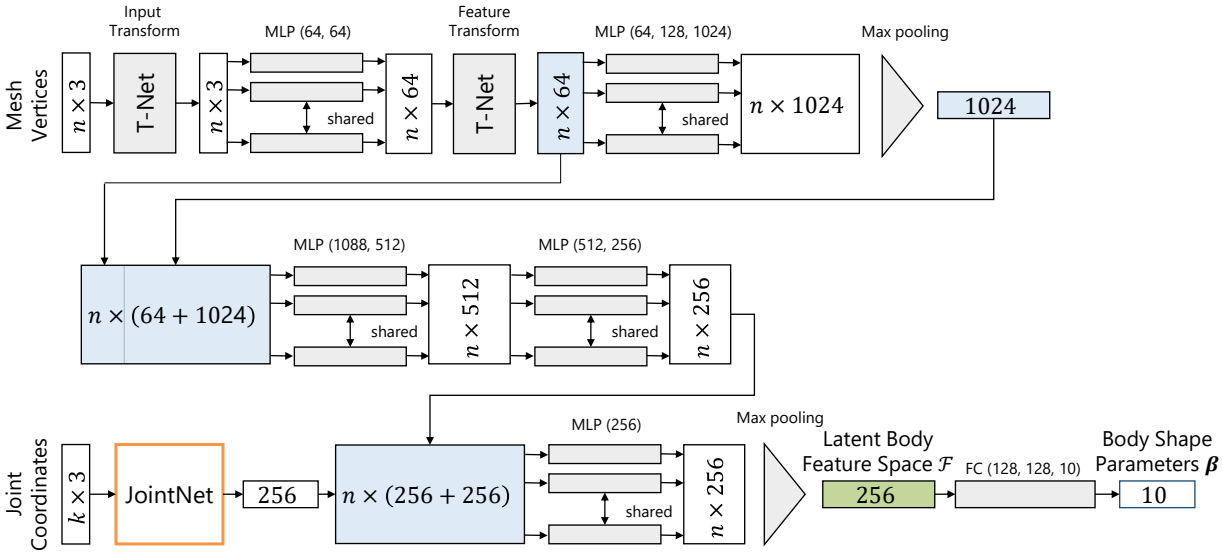

Fig. 4: **Architecture of ShapeEncoder.** This architecture is based on PointNet [19], which is designed to process point cloud data. The numbers in the figure denote the feature dimensions of each layer. In the proposed method, the number of mesh vertices n is 6,890 and the number of joint coordinates k is 5.

C. Network Architecture

The proposed method is composed of two main networks, ShapeEncoder and DM-Encoder, and one auxiliary network, JointNet. ShapeEncoder takes the vertex and joint coordinates of a mesh model generated from SMPL, and estimates the corresponding body shape parameters $\boldsymbol{\beta}$. Compared to the method of Dibra et al. [10], which used HKS descriptors extracted from mesh models, the proposed network avoids the use of hand-crafted features. This is because deriving HKS is computationally expensive, and it is hard to apply data augmentation to HKS. ShapeEncoder is based on PointNet [19], which is designed to process point cloud data. Fig. 4 shows the architecture of ShapeEncoder.

DM-Encoder is a network that encodes depth maps into the latent body feature space. We utilize a 164-layered Residual Network (ResNet-164) [20] as DM-Encoder. Features from joint coordinates are fused just before the last global average pooling of the network.

The auxiliary network JointNet extracts features from joint coordinates. In the proposed architecture, input data from two different modalities, that is, depth maps and vertex coordinates of the mesh model, are processed by two types of networks. On the other hand, joint coordinates can be obtained from both of the two modalities, and have the same properties, regardless of their origin. Therefore, JointNet is designed as a common part in both ShapeEncoder and DM-Encoder, and shares weight parameters as shown in Fig. 2. This design allows the architecture to extract features shared between the two networks. The structure of JointNet is a small version of ShapeEncoder, where 256-dimensional features are extracted from the layer placed just before the first max pooling layer of ShapeEncoder.

D. Training

The proposed method is trained with 5,000 synthetic mesh models and depth maps (2,500 males and 2,500 females) generated as described in Section III-A. For depth normalization, $\bar{d} = 77.46$ mm, computed from the training samples, is used. ShapeEncoder is trained with Adam optimizer [21] with mini-batches of size 16 to minimize mean squared error (MSE) between the estimated body shape parameter $\hat{\boldsymbol{\beta}}$ and the ground truth $\boldsymbol{\beta}$:

$$MSE(\boldsymbol{\beta}, \hat{\boldsymbol{\beta}}) = \frac{1}{m_\beta} \sum_{i=1}^{m_\beta} (\beta_i - \hat{\beta}_i)^2, \tag{1}$$

where $m_\beta = 10$ as described in Sec. III-A.

To train DM-Encoder, the objective is mapping a depth map to the learned latent representation. DM-Encoder is trained to minimize MSE between the mapped feature vector and the corresponding ShapeEncoder's representation.

$$MSE(\mathcal{F}, \hat{\mathcal{F}}) = \frac{1}{m_f} \sum_{i=1}^{m_f} (f_i - \hat{f}_i)^2, \tag{2}$$

where the dimensionality $m_f = 256$. Nesterov's accelerated gradient method [22] with mini-batches of size 2 is employed as the optimizer. Each training is run for 250 epochs.

IV. Experiments

To evaluate the performance, the proposed method was compared against the existing single image-based approaches, namely SMPLify [9], the method of Dibra et al. [10], and SMPLify-X [11]. The result for SMPLify [9] was estimated with the authors' public imlementation[1]. The input poses

[1]http://smplify.is.tue.mpg.de/

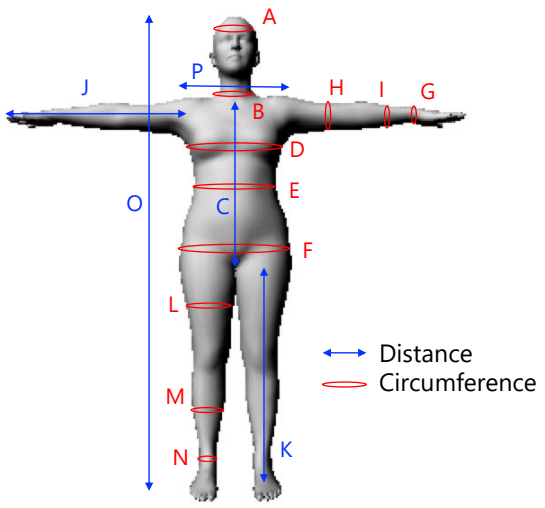

Fig. 5: **Measurements in the experiment.** A total of 16 measurements were used, consisting of 5 distance measures (blue) and 11 circumferences (red).

were derived from the pretrained ResNet-152 model[2] of DeeperCut [23]. Since SMPLify estimates the focal length of a camera during its optimization, we tested two different settings; 1) purely estimated focal length (default); and 2) given calibrated focal length. The given calibrated focal length brings better initialization of the optimization while it requires camera calibration in advance. On the other hand, there is no public implementation of the method of Dibra et al. [10]. Therefore, we reimplemented this method and refer it to as HKSNet. HKSNet was trained with the same dataset as the proposed method (explained in Sec. III-A). We created the input silhouette images for HKSNet by binarizing the depth maps in the dataset. For SMPLify-X, we used the authors' public implementation[3]. We used OpenPose[4] to estimate the input poses. For a fair comparison with the proposed method, we explicitly gave a subject's gender to SMPLify-X and fit an SMPL model instead of an SMPL-X model.

A. Dataset

Since there was no publicly-available dataset for single-depth shape estimation, an experiment was conducted on a newly created dataset containing real-world depth maps of 15 males and 15 females each. The depth maps were taken using Kinect V2, a time-of-flight camera from Microsoft Corporation. Subjects of the experiment were requested to be tightly clothed and to keep T-pose during image acquisition. For the evaluation, a total of 16 measurements were used as in [10], consisting of 5 distance measures (e.g. body height or arm length) and 11 circumferences (e.g. hip or waist). Fig. 5 visualizes the 16 measurements.

[2]https://github.com/eldar/deepcut-cnn
[3]https://github.com/vchoutas/smplify-x
[4]https://github.com/CMU-Perceptual-Computing-Lab/openpose

B. Results

Table I shows the mean measurement errors between the estimated models and the ground truth measurements. On the average distance, circumference, and overall errors, the proposed method consistently achieved the smallest error. Also, the proposed full model (ShapeEncoder + DM-Encoder + JointNet) outperformed the conventional methods on the majority of the 16 measurements. In the case where the joint coordinates are not used (ShapeEncoder + DM-Encoder), only the circumference error is reduced. This indicates that the detailed difference in body shapes can be expressed by using depth maps. By using the joint coordinates (ShapeEncoder + DM-Encoder + JointNet), the distance error was also reduced, and the smallest error among the compared methods was achieved. The results of SMPLify indicate that the given calibrated focal length greatly contributed to the reduction of the distance errors. However, it required manual camera calibration in advance. On the other hand, the proposed method does not require such a preliminary calibration.

V. Conclusions

This paper proposed a 3D body shape estimation method using a single depth map. The proposed method incorporates two deep neural networks, namely ShapeEncoder and DM-Encoder. ShapeEncoder learns a compact latent representation of 3D body models generated from a statistical body model. DM-Encoder maps input depth maps to the latent space, and body shape parameters to deform the template model are estimated by the latter part of ShapeEncoder. To incorporate pose information, JointNet is introduced, which gives the two networks shared auxiliary features. The proposed method was evaluated on real depth maps of 30 subjects, and the results showed significant performance improvements over previous approaches.

References

[1] M. S. Allen and E. E. Walter, "Personality and body image: A systematic review," *Body Image*, vol. 19, pp. 79–88, 2016.
[2] J. Stevens, J. Cai, E. R. Pamuk, D. F. Williamson, M. J. Thun, and J. L. Wood, "The effect of age on the association between body-mass index and mortality," *New England Journal of Medicine*, vol. 338, no. 1, pp. 1–7, 1998.
[3] K. L. Johnson and L. G. Tassinary, "Perceiving sex directly and indirectly: Meaning in motion and morphology," *Psychological Science*, vol. 16, no. 11, pp. 890–897, 2005.
[4] I. Mikić, M. Trivedi, E. Hunter, and P. Cosman, "Human body model acquisition and tracking using voxel data," *International Journal of Computer Vision*, vol. 53, no. 3, pp. 199–223, 2003.
[5] A. Weiss, D. Hirshberg, and M. J. Black, "Home 3D body scans from noisy image and range data," in *ICCV*, pp. 1951–1958, 2011.
[6] M. Zeng, L. Cao, H. Dong, K. Lin, M. Wang, and J. Tong, "Estimation of human body shape and cloth field in front of a Kinect," *Neurocomputing*, vol. 151, pp. 626–631, 2015.
[7] P. Guan, A. Weiss, A. O. Balan, and M. J. Black, "Estimating human shape and pose from a single image," in *ICCV*, pp. 1381–1388, 2009.
[8] N. Hasler, H. Ackermann, B. Rosenhahn, T. Thormählen, and H.-P. Seidel, "Multilinear pose and body shape estimation of dressed subjects from image sets," in *CVPR*, pp. 1823–1830, 2010.
[9] 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*, pp. 561–578, 2016.

TABLE I: **Mean estimation errors [cm].** † indicates a method using given calibrated focal length. We abbreviate the proposed ShapeEncoder, DM-Encoder, and JointNet as SE, DE, and JN, respectively. The proposed full model (ShapeEncoder + DM-Encoder + JointNet) achieved the smallest error on the average distance, circumference, and overall measurements. On the majority of the 16 measurements, the full model also outperformed the conventional methods.

Measurements	SMPLify [9]	SMPLify† [9]	HKSNet	SMPLify-X [11]	SMPLify-X† [11]	SE + DE	SE + DE + JN
Avg. Distance	13.29	5.27	5.91	7.79	12.48	6.15	**4.89**
Avg. Circum.	7.64	7.84	6.88	4.99	6.04	4.78	**4.22**
Overall Avg.	9.41	7.03	6.57	5.87	8.05	5.21	**4.43**
A. Head circum.	**2.02**	2.62	3.57	4.51	8.13	4.92	4.19
B. Neck circum.	3.39	**2.76**	3.77	5.30	8.55	5.54	5.21
C. Shoulder-blade/crotch length	10.65	8.11	5.63	**5.58**	6.66	6.59	5.83
D. Chest circum.	20.28	22.31	17.17	10.41	**7.39**	8.61	10.15
E. Waist circum.	18.39	22.91	16.88	9.03	**7.56**	8.83	9.46
F. Pelvis circum.	17.95	14.43	13.23	8.78	10.83	7.80	**5.20**
G. Wrist circum.	1.93	1.83	1.34	1.10	2.06	1.00	**0.76**
H. Bicep circum.	6.08	7.03	6.09	3.41	2.98	3.36	**2.61**
I. Forearm circum.	2.63	2.79	2.43	1.87	3.09	2.02	**1.16**
J. Arm length	7.05	5.63	6.70	6.79	15.52	5.49	**4.68**
K. Inside leg length	21.07	**4.93**	5.68	10.41	8.30	7.27	7.33
L. Thigh circum.	5.35	4.68	5.75	5.68	8.54	6.12	**4.25**
M. Calf circum.	4.04	3.50	3.59	2.96	4.05	2.81	**2.13**
N. Ankle circum.	2.02	1.39	1.82	1.85	3.21	1.62	**1.35**
O. Overall height	23.89	**4.66**	8.66	13.30	27.82	8.80	5.32
P. Shoulder breadth	3.78	3.00	2.89	2.87	4.11	2.60	**1.31**

[10] E. Dibra, H. Jain, C. Oztireli, R. Ziegler, and M. Gross, "Human shape from silhouettes using generative HKS descriptors and cross-modal neural networks," in *CVPR*, pp. 5504–5514, 2017.

[11] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3D hands, face, and body from a single image," in *CVPR*, pp. 10 975–10 985, 2019.

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

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

[14] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, pp. 7794–7803, 2018.

[15] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele, "DeepCut: Joint subset partition and labeling for multi person pose estimation," in *CVPR*, pp. 4929–4937, 2016.

[16] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *CVPR*, pp. 7291–7299, 2017.

[17] J. Sun, M. Ovsjanikov, and L. Guibas, "A concise and provably informative multi-scale signature based on heat diffusion," *Computer Graphics Forum*, vol. 28, no. 5, pp. 1383–1392, 2009.

[18] M. Gschwandtner, R. Kwitt, A. Uhl, and W. Pree, "BlenSor: Blender sensor simulation toolbox," in *ISVC*, pp. 199–208, 2011.

[19] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *CVPR*, pp. 77–85, 2017.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *ECCV*, pp. 630–645, 2016.

[21] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[22] Y. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$," *Soviet Mathematics Doklady*, vol. 27, no. 2, pp. 372–376, 1983.

[23] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "DeeperCut: A deeper, stronger, and faster multi-person pose estimation model," in *ECCV*, pp. 34–50, 2016.