Layered-Garment Net: Generating Multiple Implicit Garment Layers from a Single Image (Supplementary Document)

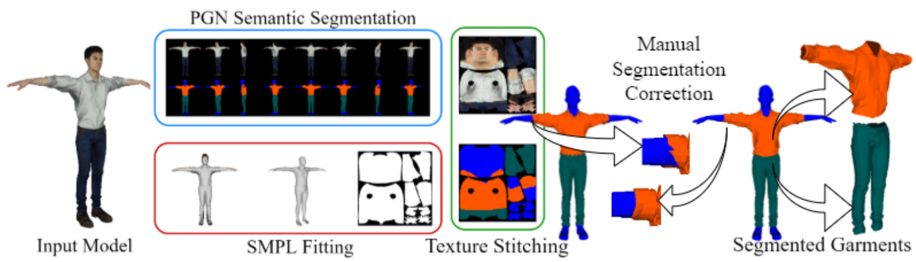

Fig. 1. Garment Segmentation Pipeline.

A Dataset Preparation Details

In this section we show the details of data processing steps. We perform the SMPL fitting, semantic segmentation, layer relationship definition and generate garment meshes for each object. We purchase 142 rigged human models from AXYZ.

SMPL Fitting Given a 3D human model, the first step is to deformed it to T-pose, then we align SMPL model to it by optimizing the shape β and pose θ parameters. A good alignment is essential to generate correct garment meshes. SMPL is a naked body model. Since our dataset covers various outfits, it makes our fitting task even more challenging. We minimize the fitting energy which combines the 3D joint error and Chamfer distance to get a good estimation of body shape and pose. The details of fitting can be found in [1].

Garment Segmentation The detailed steps of garment segmentation is illustrated in Fig. 1. Because all the objects are rigged models and compatible with Unity, following the mapping between SMPL joints and Unity humanoid skeletons, we could control object poses with SMPL's parameter θ and get rendered images. In this step, we first generate eight images from different views for each object, by putting it roughly in the view center and at a distance of around 2 meters away from the camera, and run semantic segmentation on them. We use the trained model in [2] to segment the rendered images. As we focus on the garments, only labels of Upper-Clothes, Pants, Coat, Dress and Skirt are included.

Then, we follow the texture stitching method in [3] to generate the segmentation texture map. From the SMPL fitting model and its texture map, we could get the object's segmented mesh. However, this initial segmentation is not always accurate. We perform manual correction to the segmentation results when necessary.

Fig. 2. Example models of each garment class.

Finally, we extract garments according to the face color of object's triangular mesh, and classify them to different garment classes as Tab. 1 shows. Fig. 2 shows some example garment models of each class.

Covering Relationship To generate the combination of layered garment data set, we first pick up three garments from classes defined in Tab. 1, arranged in combinations. Subsequently, we use the iterative approach as shown in Alg. 1 for each layer i, for $i > 0$, with layer 0 being human body and n total layers of garments over human body.

Algorithm 1 Iterative Garment Layer Update

```
for i from 1 to n do
    for j from 0 to i − 1 do
        while no intersecting triangles do
            select vertices of layer_i inside layer_j
            bring the selected vertices outside layer_j
            select triangles on layer_i intersecting layer_j
            subdivide the selected triangles
        end while
    end for
end for
```

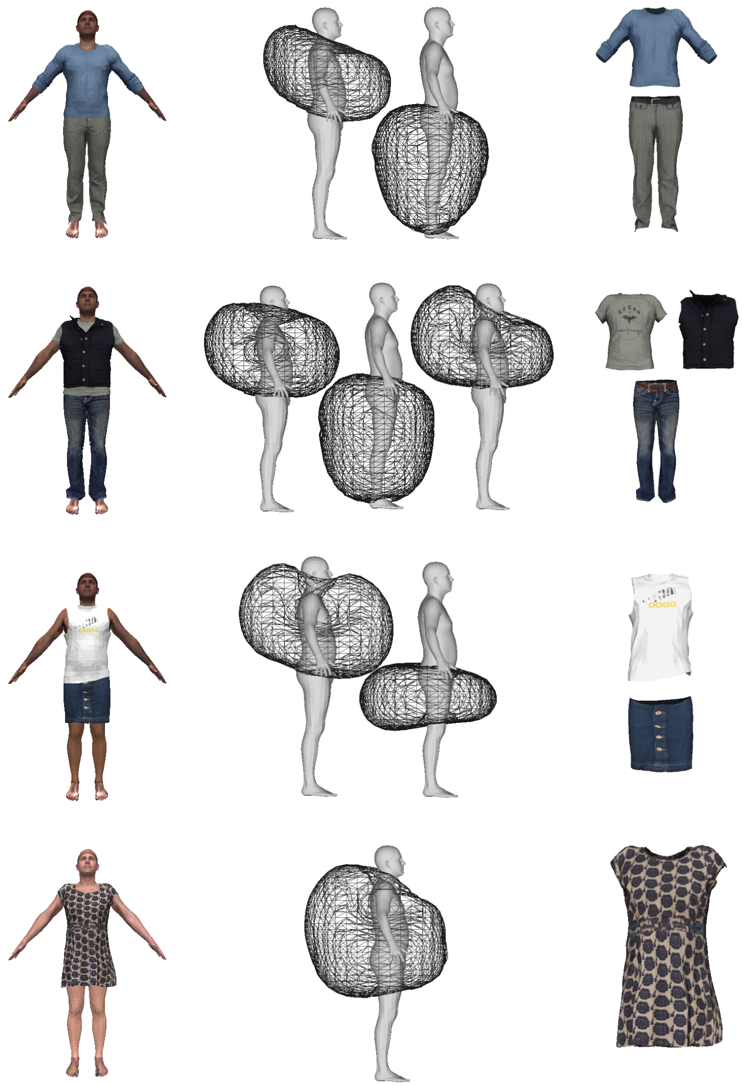

Fig. 3. Some example garment data consisting of different combinations of garments from different classes. We show the input model (left) with layers of garment on human body following a covering relationship, garment indication field 0.5 level iso-surface (middle) for each garment, and individual garments (right).

Class	Subclass	Number
Shirt	None-sleeve (NS)	1
	Short-sleeve (SS)	16
	Long-sleeve (LS)	16
Coat	None-sleeve (NC)	1
	Short-sleeve (SC)	1
	Long-sleeve (LC)	15
Pant	Short (SP)	8
	Long (LP)	14
Dress	None-sleeve (ND)	8
	Short-sleeve (SD)	2
	Long-sleeve (LD)	7
Skirt	Skirt	12

Garment Combinations			**Number**
Layer 1 — Layer 2 — Layer 3			
Pant	Shirt		4032
Shirt	Pant		4032
Shirt	Pant	Coat	2800
Skirt	Shirt		588
Dress			63
			11515

Table 1. Garment Classes in our training data set.

We finally get around $12k$ combinations of different one-layered, two-layered and three-layered garment data over 7 different poses of naked human. These extracted garments form the training data. We use a shirt, a pant and a coat unavailable in training data, over 10 different poses, and form the testing data. This test set is divided into two parts: 7 instances in same poses as training data as shown in Fig. 4, and 3 instances in different poses as shown in Fig. 5.

Some examples of garments in our data, along with their underlying 0.5-level iso-surface of Garment Indication Fields has been shown in Fig. 3.

B Qualitative comparison

We show our reconstruction results on our synthetic test data, as mentioned in Appendix A, in 7 training poses in Fig. 4, and on 3 new poses in Fig. 5. These reconstruction results show generalization of our model on different garments and different poses.

We show our reconstruction results on publically available data sets: Digital Wardrobe [4], BUFF [5] and SIZER [6] data sets in Fig. 6. We show full clothed human body reconstruction for BUFF data set, and individual garment reconstruction for Digital Wardrobe and SIZER data sets.

Fig. 5. Our reconstruction results on garments and poses from outside training set.

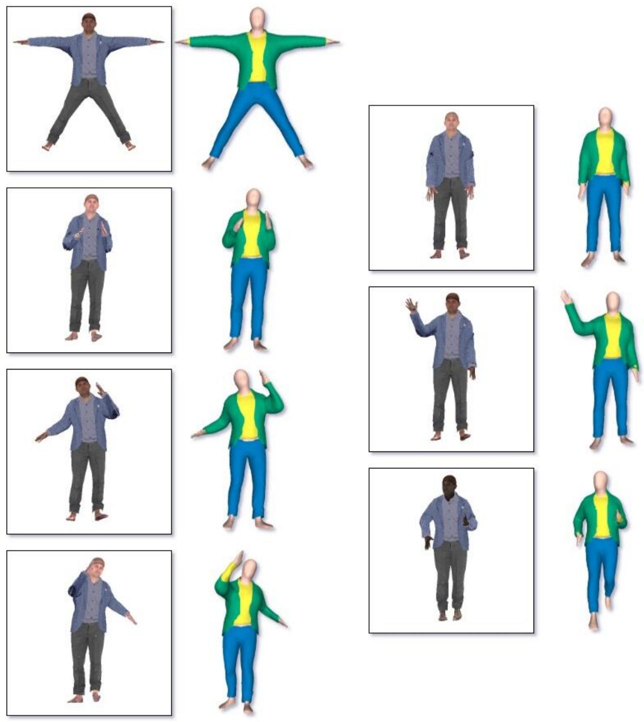

Fig. 4. Our reconstruction results on garments outside training set, posed in poses inside trained set.

We show the comparison of our results with state-of-the-art approaches [7, 8] on some images from Google Images in Fig. 7.

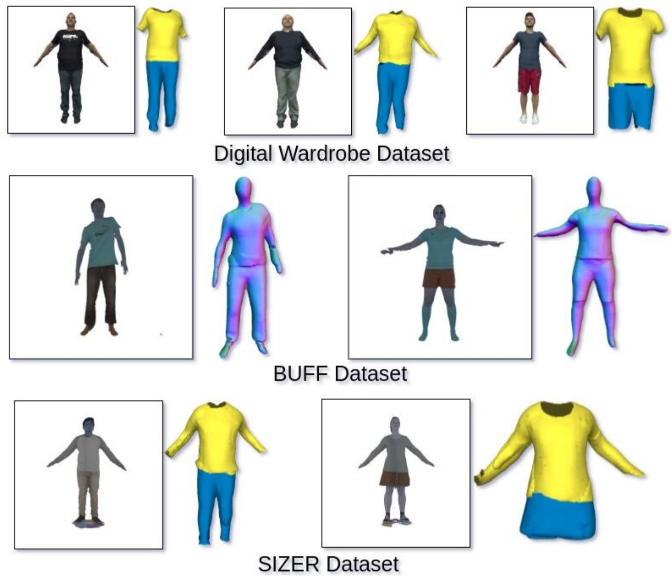

Fig. 6. Our results on different data sets. We reconstruct full human body for BUFF data set, and individual garments for Digital Wardrobe and SIZER data sets.

We finally show comparison of our reconstruction results with state-of-the-art approaches [7, 8] on some challenging cases from real world images in Fig. 8. We observe that since the limbs (hands and/or legs) of the subjects in these images are close to torso, the reconstruction of layer 0 naked human body fails and have some artifacts. However, SMPLicit fails to reconstruct accurate garment results, and BCNet fails to reconstruct multiple-layer of garments in challenging poses. Hence, our method outperforms them in these challenging approaches as well.

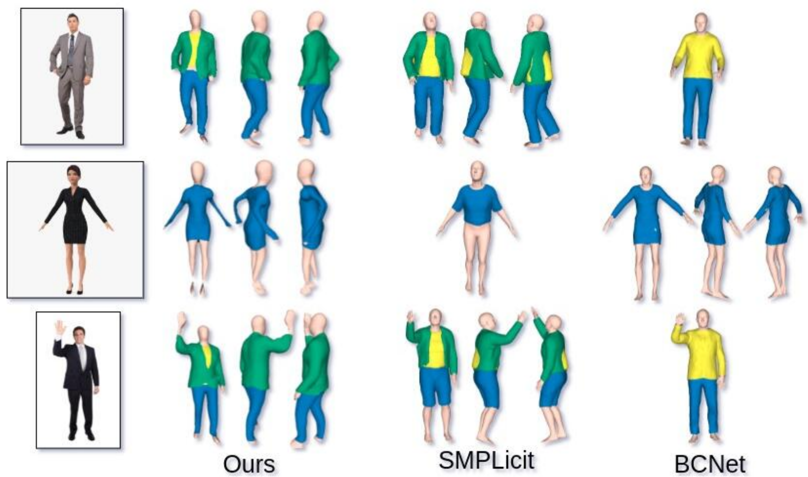

Fig. 7. Multi-layer garment reconstruction results (front, left and right views), and comparison with SMPLicit [8] and BCNet [7] on some web-scraped images. Source: Google Images

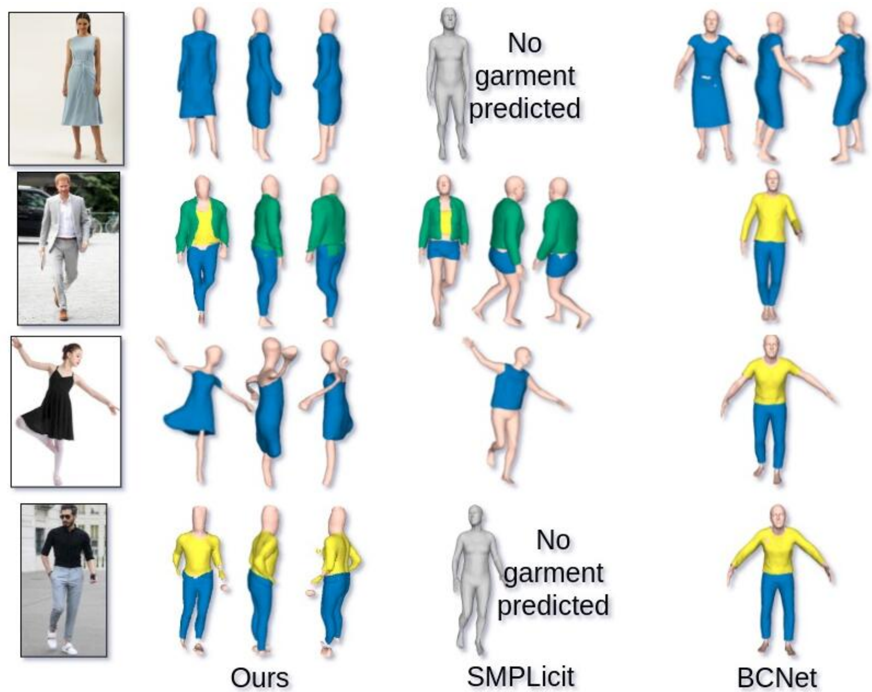

Fig. 8. Some failure cases. As mentioned in limitations, LGN fails for challenging poses from real world images. Source: Google Images

References

1. Ni, S., Luo, R., Zhang, Y., Budagavi, M., Dickerson, A.J., Nagar, A., Guo, X.: Scan2avatar: Automatic rigging for 3d raw human scans. In: ACM SIGGRAPH 2020 Posters. (2020) 1–2
2. Gong, K., Liang, X., Li, Y., Chen, Y., Yang, M., Lin, L.: Instance-level human parsing via part grouping network. CoRR **abs/1808.00157** (2018)
3. Alldieck, T., Magnor, M., Bhatnagar, B.L., Theobalt, C., Pons-Moll, G.: Learning to reconstruct people in clothing from a single rgb camera. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2019) 1175–1186
4. Bhatnagar, B.L., Tiwari, G., Theobalt, C., Pons-Moll, G.: Multi-garment net: Learning to dress 3d people from images. In: Proceedings of the IEEE/CVF international conference on computer vision. (2019) 5420–5430
5. Zhang, C., Pujades, S., Black, M.J., Pons-Moll, G.: Detailed, accurate, human shape estimation from clothed 3d scan sequences. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
6. Tiwari, G., Bhatnagar, B.L., Tung, T., Pons-Moll, G.: Sizer: A dataset and model for parsing 3d clothing and learning size sensitive 3d clothing. In: European Conference on Computer Vision (ECCV), Springer (2020)

7. Jiang, B., Zhang, J., Hong, Y., Luo, J., Liu, L., Bao, H.: Bcnet: Learning body and cloth shape from a single image. In: European Conference on Computer Vision, Springer (2020) 18–35
8. Corona, E., Pumarola, A., Alenya, G., Pons-Moll, G., Moreno-Noguer, F.: Smplicit: Topology-aware generative model for clothed people. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2021) 11875–11885