Supplementary Material – Everybody Is Unique: Towards Unbiased Human Mesh Recovery

Ren Li
https://liren2515.github.io/page/
Srikrishna Karanam
https://karanams.github.io/
Meng Zheng
https://www.linkedin.com/in/meng-zheng-27ab35144
Terrence Chen
https://www.linkedin.com/in/terrencechen
Ziyan Wu
http://wuziyan.com/

United Imaging Intelligence
Cambridge MA, USA
{first.last}@uii-ai.com

1 Sample Data

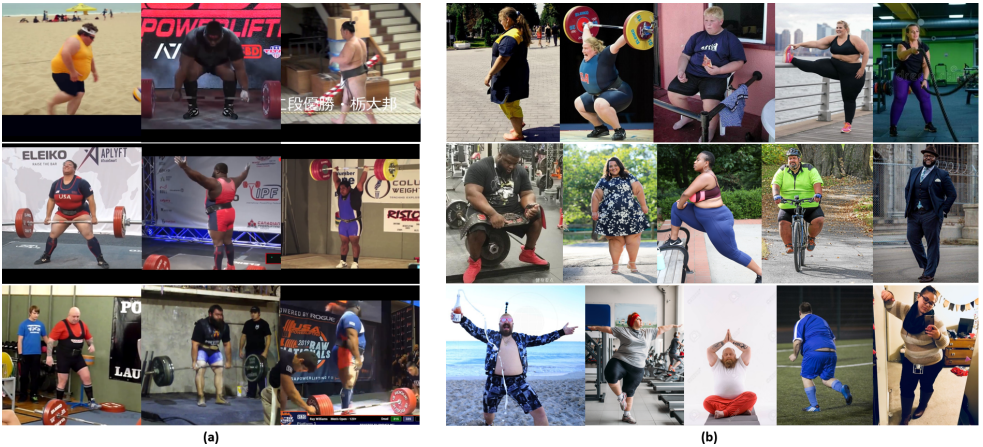

Figure 1: Sample images from (a) SSP-3D [13] and (b) LargeSize.

Figure 1 shows sample images from SSP-3D dataset [13] and our LargeSize dataset.

© 2021. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.
This work was done during Ren Li's internship with UII. Corresponding author: Srikrishna Karanam.

2 Problems with EFT

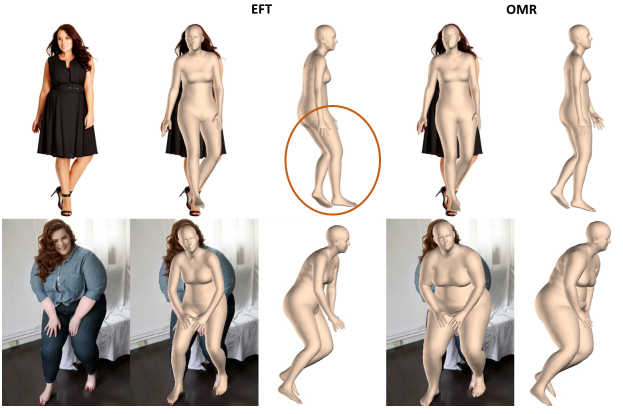

Figure 2: The comparison of mesh fitting results with EFT [9] and our proposed OMR.

Figure 2 shows some fitting results with EFT [9], which illustrate the problmes mentioned in Section 3.2, i.e. the risk of overfitting the 2D cost function and biased estimation for obese data. However, our proposed OMR addresses these limitations.

3 Implementation Details

3.1 Mesh Vertices to Segmentation Masks

Given the mesh vertices, we use a differentiable renderer SoftRas [19], which fuses the probabilistic contributions of all mesh triangles with respect to the rendered pixels, and a predefined texture map to generate the binary body part segmentation masks $\boldsymbol{S} \in \mathbb{R}^{H \times W \times D}$, where H, W and D are the height, width and the number of body part, respectively.

3.2 Loss Items

We use the Geman-McClure error function [2] to measure the 2D re-projection loss L_{2D} for both P-iteration and Q-iteration steps:

$$L_{\text{2D}}(\hat{\mathbf{x}}, \mathbf{x}) = \frac{\sigma^2 * (\hat{\mathbf{x}} - \mathbf{x})^2}{\sigma^2 + (\hat{\mathbf{x}} - \mathbf{x})^2}, \tag{1}$$

where $\sigma = 100$, and $\hat{\mathbf{x}}$ and $\mathbf{x}$ are the predicted 2D joints and their corresponding ground truth.

The pose prior $L_\theta(\theta)$ is implemented following prior work [11] as:

$$L_\theta(\theta) = ||Z(\theta)||_2^2, \tag{2}$$

where $Z(\theta)$ is the latent representation learned by a variational autoencoder.

The 2D shape loss L_{shape} is inspired by the intersection-over-union (IoU) loss used in the segmentation task, e.g., [12]. Given the estimated binary mask $\hat{\mathbf{S}}_i$ and the ground truth mask

$\mathbf{S}_i$ for part i, their intersection I_i and union U_i can be computed by

$$I_i = \sum_{m,n} \hat{\mathbf{S}}_i^{m,n} \cdot \mathbf{S}_i^{m,n}, \tag{3}$$

$$U_i = \sum_{m,n} \hat{\mathbf{S}}_i^{m,n} + \mathbf{S}_i^{m,n} - \hat{\mathbf{S}}_i^{m,n} \cdot \mathbf{S}_i^{m,n}, \tag{4}$$

where (m,n) is the pixel position of the mask. Therefore, our 2D shape loss can be formulated as $L_{\text{shape}} = \sum_{i=1}^{6} 1 - I_i/U_i$ to sum over all 6 body parts, which gives Eqn. 3 in the main paper.

3.3 Optimization

We use the Adam optimizer [6] for both P-iteration and Q-iteration steps. The learning rate for the Q-iteration step is set to 1e-3, whereas the learning rate of P-iteration step is set to 1e-6. The number of iterations for each single P-iteration and Q-iteration steps is 20. All our implementation is in PyTorch [10].

3.4 SSP-3D [13] Validation Set

To select data with extreme shape parameters, we measure the PVE-T between the mean shape parameters and those of each sample in SSP-3D. The data whose PVE-T is no less than 22.5 mm will be included into the validation set.

4 Additional Experimental Results

4.1 Quantitative Results

Table 1 shows results on the SSP-3D dataset where we use HKMR [3] as the base model. Similarly, We can observe the proposed L_{shape} consistently reduces both PA-MPJPE and PVE-T errors across all the methods, and the proposed OMR outperforms both SMPLify [1] and EFT [4]. Note that these results correspond to the same experiment as Table 1 in the main paper, where we showed results with SPIN [7].

SSP-3D	MPJPE-PA (mm)	PVE-T (mm)
HKMR	57.53	34.54
SMPLify w/o L_{shape}	55.02	28.92
SMPLify+L_{shape}	53.31	27.20
EFT w/o L_{shape}	55.29	30.17
EFT+L_{shape}	54.88	29.59
OMR w/o L_{shape}	52.67	29.20
OMR+L_{shape}	**49.77**	**18.72**

SSP-3D (PVE-T)	Torso	Legs	Arms	Head
HKMR	52.10	25.04	36.93	31.19
SMPLify w/o L_{shape}	47.05	19.41	32.50	23.70
SMPLify+L_{shape}	43.99	18.68	30.30	22.02
EFT w/o L_{shape}	49.10	20.01	34.22	24.86
EFT+L_{shape}	48.17	19.59	33.12	24.94
OMR w/o L_{shape}	47.68	19.25	32.30	25.00
OMR+L_{shape}	**24.46**	**16.04**	**19.84**	**16.38**

Table 1: SMPLify [1] vs. EFT [4] vs. proposed OMR on SSP-3D using HKMR [3] as the base model. These results correspond to the same experiment as Table 1 in the main paper, where we showed results with SPIN [7].

Table 2 shows results on Human3.6M dataset for the generalized model fitting evaluation, where CMR [8] and HKMR [3] are the base models. The proposed OMR is still able to have

a steady decrease in both MPJPE and PA-MPJPE, which reaches to the lowest error. Note that these results correspond to the same experiment as Table 2 in the main paper, where we showed results with SPIN [7].

Human3.6M	MPJPE	PA-MPJPE
CMR [8]	76.04	50.46
SMPLify - 20	84.44	57.41
SMPLify - 100	101.92	63.38
EFT - 20	70.32	47.43
EFT - 100	74.62	46.77
OMR (1P1Q)	70.03	47.10
OMR (5P4Q)	**68.89**	**44.56**

Human3.6M	MPJPE	PA-MPJPE
HKMR [3]	65.03	46.53
SMPLify - 20	69.57	44.61
SMPLify - 100	81.66	50.18
EFT - 20	58.41	39.53
EFT - 100	66.62	41.81
OMR (1P1Q)	59.15	39.25
OMR (5P4Q)	**56.97**	**38.60**

Table 2: SMPLify vs. EFT vs. OMR on Human3.6M using the CMR and HKMR base models. All numbers in mm. These results correspond to the same experiment as Table 2 in the main paper, where we showed results with SPIN [7].

SSP-3D	MPJPE	PA-MPJPE	PVE-T	mIoU
SPIN [9]	92.03	53.57	35.68	0.6570
SPIN - SMPLify	106.50	59.45	32.65	0.6889
SPIN - EFT	88.60	55.14	33.05	0.6823
SPIN - OMR	**84.48**	**50.16**	**27.36**	**0.7088**
HKMR [3]	98.02	57.53	34.54	0.6647
HKMR - SMPLify	107.55	60.63	32.62	0.6799
HKMR - EFT	92.53	56.31	32.59	0.6825
HKMR - OMR	**88.81**	**52.43**	**27.23**	**0.7028**

Human3.6M	Protocol #1		Protocol #2	
	MPJPE	PA-MPJPE	MPJPE	PA-MPJPE
SPIN [9]	65.60	44.1	62.23	41.1
SPIN - SMPLify	65.12	45.2	61.39	42.6
SPIN - EFT	63.40	44.3	60.00	41.5
SPIN - OMR	**61.95**	**43.7**	**58.51**	**41.0**
HKMR [3]	64.02	45.9	59.62	43.2
HKMR - SMPLify	65.91	47.3	62.22	44.4
HKMR - EFT	64.04	46.3	62.22	43.4
HKMR - OMR	**62.70**	**45.6**	**59.36**	**42.9**

Table 3: Improving baseline models by retraining with annotations generated by our method: Results on SSP-3D and Human3.6M with SPIN and HKMR as base models. These results correspond to the same experiment as Table 3 in the main paper, where we showed a subset of these results with SPIN [7] and HKMR [3].

Tables 3 and 4 shows the performance improvement obtained by using OMR-generated parameters for the training of HMR [5], CMR [8], SPIN [7], and HKMR [3]. Note that these results correspond to the same experiment as Table 3 in the main paper, where we showed a subset of these results with SPIN [7] and HKMR [3] as the base models.

Finally, in Table 5, we show results of retraining HMR, CMR, SPIN, and HKMR on our internal LargeSize data, where one can note substantial performance improvements with the proposed OMR.

4.2 Qualitative Results

In Figure 3, we show some representative mesh fitting results using the proposed OMR on both the generic standard benchmark dataset and our LargeSize dataset. Figure 4 and 5 show the improved mesh estimations with HKMR [3] and SPIN [7] on 3DPW [14] and LargeSize when compared to their corresponding baseline versions.

SSP-3D	MPJPE	PA-MPJPE	PVE-T	mIoU
HMR [8]	102.34	67.91	31.41	0.6477
HMR - OMR	**99.54**	**59.83**	**30.87**	**0.6599**
CMR [9]	135.51	67.11	730.62	0.6651
CMR - OMR	**93.61**	**56.19**	**30.08**	**0.6862**

Human3.6M	Protocol #1		Protocol #2	
	MPJPE	PA-MPJPE	MPJPE	PA-MPJPE
HMR [8]	87.97	58.1	88.0	56.8
HMR - OMR	**77.73**	**56.1**	**74.2**	**53.8**
CMR [8]	74.7	51.9	71.9	50.1
CMR - OMR	**67.0**	**47.9**	**64.7**	**45.7**

Table 4: Improving baseline models by retraining with annotations generated by our method: Results on SSP-3D and Human3.6M with HMR and CMR as base models. These results correspond to the same experiment as Table 3 in the main paper, where we showed results with SPIN [7] and HKMR [3].

	FB Seg.		Part Seg.		PVE-T
	acc.	f1	acc.	f1	
HMR [5]	93.89	89.35	91.97	71.37	36.87
HMR - OMR	**94.71**	**91.01**	**92.99**	**76.21**	**20.24**
CMR [9]	93.93	89.84	91.83	72.26	24.93
CMR - OMR	**94.45**	**90.42**	**92.90**	**76.30**	**22.05**
SPIN [7]	93.76	89.01	92.20	73.25	29.24
SPIN - SMPLify	94.93	91.37	93.41	77.94	20.42
SPIN - EFT	94.73	90.89	93.27	77.91	21.55
SPIN - OMR	**95.78**	**93.13**	**94.32**	**81.01**	**15.20**
HKMR [3]	94.42	90.35	92.82	75.77	30.73
HKMR - SMPLify	94.76	90.98	93.23	76.97	20.30
HKMR - EFT	94.57	90.55	93.10	76.92	21.67
HKMR - OMR	**95.82**	**93.00**	**94.29**	**80.57**	**13.98**

Table 5: Improving baseline models by retraining with annotations generated by our method on our LargeSize dataset.

References

[1] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, 2016.

[2] Stuart Ganan and D McClure. Bayesian image analysis: An application to single photon emission tomography. *Amer. Statist. Assoc*, pages 12–18, 1985.

[3] Georgios Georgakis, Ren Li, Srikrishna Karanam, Terrrence Chen, Jana Kosecka, and Ziyan Wu. Hierarchical kinematic human mesh recovery. In *ECCV*, 2020.

[4] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation. *arXiv preprint arXiv:2004.03686*, 2020.

[5] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018.

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Figure 3: OMR mesh fits on (a) non-obese standard benchmark data and (b) LargeSize data.

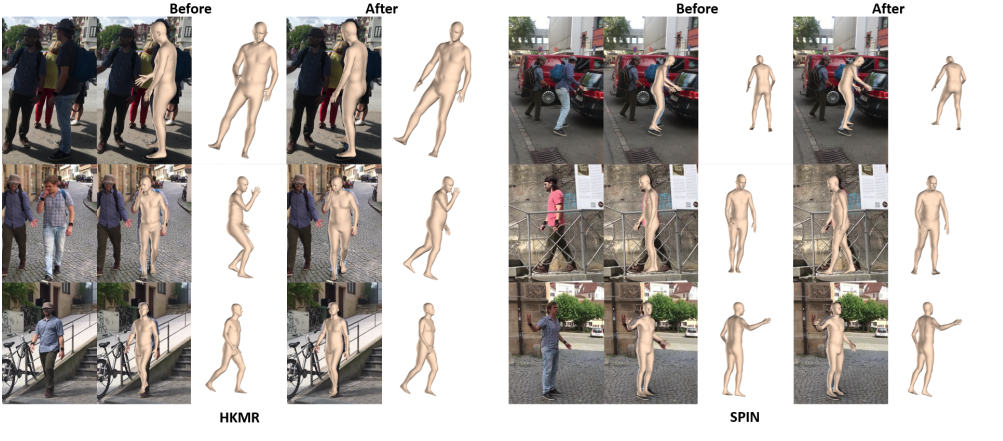

Figure 4: Mesh estimation results on 3DPW [14] of HKMR and SPIN before and after retraining with OMR-generated parameters.

Figure 5: Mesh estimation results on LargeSize of HKMR and SPIN before and after retraining with OMR-generated parameters.

[7] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, 2019.

[8] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, 2019.

[9] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7708–7717, 2019.

[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[11] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10975–10985, 2019.

[12] M. A. Rahman and W. Yang. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International Symposium on Visual Computing*, 2016.

[13] Akash Sengupta, Ignas Budvytis, and Roberto Cipolla. Synthetic training for accurate 3d human pose and shape estimation in the wild. *arXiv preprint arXiv:2009.10013*, 2020.

[14] Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, sep 2018.