Neural Body Fitting: Unifying Deep Learning and Model Based Human Pose and Shape Estimation - Supplementary Material

Mohamed Omran[1] Christoph Lassner[2*] Gerard Pons-Moll[1] Peter V. Gehler[2*]
Bernt Schiele[1]
[1]Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
[2]Amazon, Tübingen, Germany
{mohomran, gpons, schiele}@mpi-inf.mpg.de, {classner, pgehler}@amazon.com

1. Further Qualitative Results

One of our findings is the high correlation between input segmentation quality and output fit quality. We provide some additional qualitative examples that illustrate this correlation. In Fig. 1, we present the four worst examples from the validation set in terms of 3D joint reconstruction error when we use our trained part segmentation network; in Fig. 2, we present the worst examples when the network is trained to predict body model parameters given the ground truth segmentations. This does not correct all estimated 3D bodies, but the remaining errors are noticeably less severe.

2. Training Details

We present examples of paired training examples and ground truth in Fig 3.

Segmentation Network We train our own TensorFlow implementation of a RefineNet [4] network (based on ResNet-101) to predict the part segmentations. The images are cropped to 512x512 pixels, and we train for 20 epochs with a batch size of 5 using the Adam [3] optimizer. Learning rate and weight decay are set to 0.00002 and 0.0001 respectively, with a polynomial learning rate decay. Data augmentation improved performance a lot, in particular horizontal reflection (which requires re-mapping the labels for left and right limbs), scale augmentation (0.9 - 1.1 of the original size) as well as rotations (up to 45 degrees). For training the segmentation network on UP-3D we used the 5703 training images. For Human3.6M we subsampled the videos, only using every 10th frame from each video, which results in about 32000 frames. Depending on the amount of data, training the segmentation networks takes about 6-12 hours on a Volta V100 machine.

Fitting Network For the fitting network we repurpose a ResNet-50 network pretrained on ImageNet to regress the SMPL model parameters. We replace the final pooling layer with a single fully-connected layer that outputs the 10 shape and 216 pose parameters. We train this network for 75 epochs with a batch size of 5 using the Adam optimizer. The learning rate is set to 0.00004 with polynomial decay and we use a weight decay setting of 0.0001. We found that an L1 loss on the SMPL parameters was a little better than an L2 loss. We also experimented with robust losses (e.g. Geman-McClure [2] and Tukey's biweight loss [1]) but did not observe benefits. Training this network takes about 1.5 hours for the UP-3D dataset and six hours for Human3.6M.

Data Augmentation At test-time we cannot guarantee that the person will be perfectly centered in the input crop, which can lead to degraded performance. We found it thus critical to train both the segmentation network and the fitting network with strong data augmentation, especially by introducing random jitter and scaling. For the fitting network, such augmentation has to take place prior to training since it affects the SMPL parameters. We also mirror the data, but this requires careful mirroring of both the part labels as well as the SMPL parameters. This involves remapping the parts, as well as inverting the part rotations.

References

[1] V. Belagiannis, C. Rupprecht, G. Carneiro, and N. Navab. Robust optimization for deep regression. 2015. 1

[2] S. Geman and D. McClure. Statistical methods for tomographic image reconstruction. *Bulletin of the International Statistical Institute*, 52(4):5–21, 1987. 1

[3] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. 2014. 1

[4] G. Lin and I. R. Anton Milan, Chunhua Shen. Refinenet: Multi-path refinement networks for high-

Figure 1: Worst examples from the validation set in terms of 3D error given imperfect segmentations.

Figure 2: Worst examples from the validation set in terms of 3D error given perfect segmentations.

resolution semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

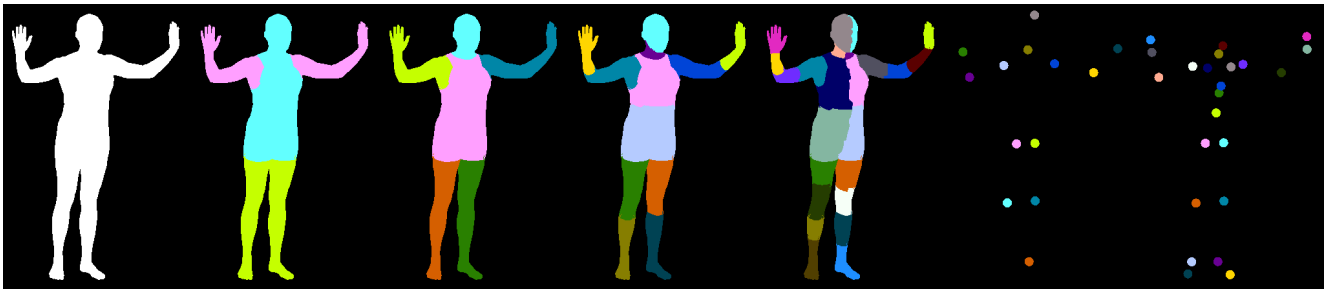

Figure 3: Example training images annotations illustrating different types and granularities.