

Expressive Body Capture: 3D Hands, Face, and Body from a Single Image

Supplementary Material

Georgios Pavlakos^{*1,2}, Vasileios Choutas^{*1}, Nima Ghorbani¹, Timo Bolkart¹, Ahmed A. A. Osman¹,
Dimitrios Tzionas¹, and Michael J. Black¹

¹MPI for Intelligent Systems, Tübingen, DE, ²University of Pennsylvania, PA, USA

{gpavlakos, vchoutas, nghorbani, tbolkart, aosman, dtzionas, black}@tuebingen.mpg.de

1. Qualitative results

Comparison of SMPL, SMPL+H & SMPL-X: In Section 4.2 of the main paper, in Table 1 we present a quantitative comparison between different models with different modeling capacities. In Fig. A.1 we present a similar comparison for SMPL (left), SMPL+H (middle) and SMPL-X (right) for an image of the EHB dataset. For fair comparison we fit all models with a variation of SMPLify-X to a single RGB image. The figure reflects the same findings as Table 1 of the paper, but qualitatively; there is a clear increase in expressiveness from left to right, as model gets richer from body-only (SMPL) to include hands (SMPL+H) or hands and face (SMPL-X).

Holistic vs part models: In Section 4.2 and Fig. 5 of the main paper we compare our holistic SMPL-X model to the hand-only approach of [24] on EHB. Figure A.2 shows a similar qualitative comparison, this time on the data of [24]. To further explore the benefit of holistic reasoning, we also focus on the head and we compare SMPL-X fitting to a head-only method by fitting FLAME [16] to 2D keypoints similar to our method. The context of the full body stabilizes head estimation for occlusions or non-frontal views (see Fig. A.3). This benefit is also quantitative, where the holistic SMPL-X improves over the head-only fitting by 17% in our EHF dataset in terms of vertex-to-vertex error.

Failure cases: Figure A.4 shows some representative failure cases; depth ambiguities can cause wrong estimation of torso pose or wrong ordinal depth estimation of body parts due to the simple 2D re-projection data term. Furthermore, occluded joints leave certain body parts unconstrained, which currently leads to failures. We plan to address this in future work, by employing a visibility term in the objective.

2. Collision Penalizer

In Section 3.4 of the paper we describe the collision penalizer. For technical details and visualizations the reader

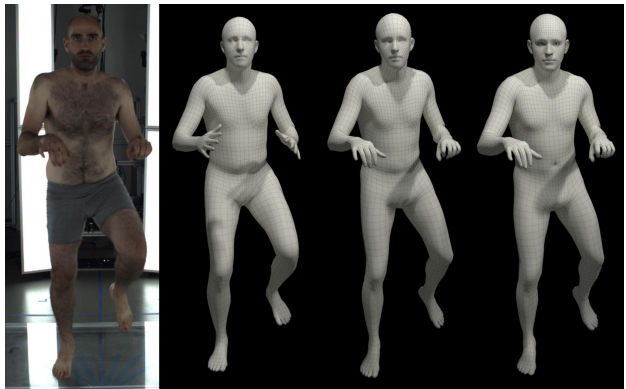


Figure A.1. Comparison of SMPL (left), SMPL+H (middle) and SMPL-X (right) on the EHB dataset, using the male models. For fair comparison we fit all models with a variation of SMPLify-X to a single RGB image. The results show a clear increase in *expressiveness* from left to right, as model gets richer from body-only (SMPL) to include hands (SMPL+H) or hands and face (SMPL-X).

is redirected to [4, 28], but for the sake of completion we include the mathematical formulation also here.

We first detect a list of colliding triangles \mathcal{C} by employing Bounding Volume Hierarchies (BVH) [27] and compute local conic 3D distance fields $\Psi : \mathbb{R}^3 \rightarrow \mathbb{R}_+$ defined by the triangles \mathcal{C} and their normals $n \in \mathbb{R}^3$. Penetrations are then penalized by the depth of intrusion, efficiently computed by the position in the distance field. For two colliding triangles f_s and f_t intrusion is bi-directional; the vertices $v_t \in \mathbb{R}^3$ of f_t are the *intruders* in the distance field Ψ_{f_s} of the *receiver* triangle f_s and are penalized by $\Psi_{f_s}(v_t)$, and vice-versa. Thus, the collision term $E_{\mathcal{C}}$ is defined as

$$E_{\mathcal{C}}(\theta) = \sum_{(f_s(\theta), f_t(\theta)) \in \mathcal{C}} \left\{ \sum_{v_s \in f_s} \| -\Psi_{f_t}(v_s)n_s \|^2 + \sum_{v_t \in f_t} \| -\Psi_{f_s}(v_t)n_t \|^2 \right\}. \quad (1)$$

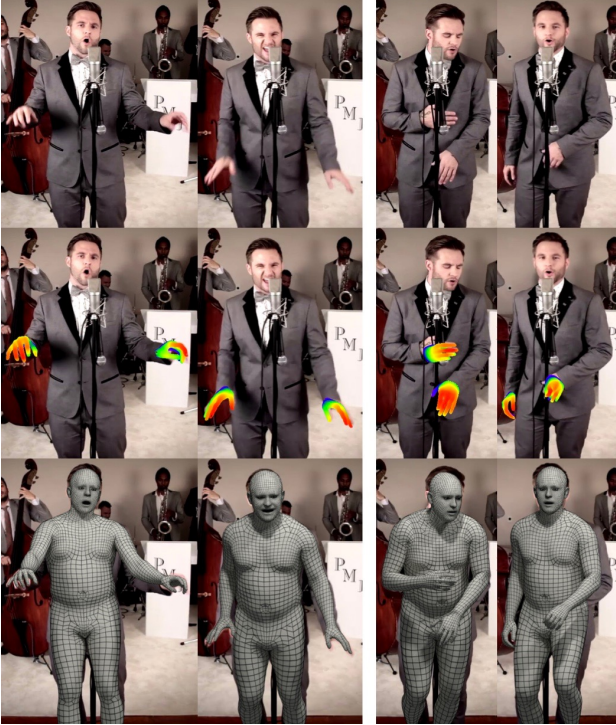


Figure A.2. Comparison of the hands-only approach of [24] (middle row) against SMPLify-X with the male SMPL-X (bottom row). Both approaches depend on OpenPose [23]. In case of good 2D detections both perform well (left group). In case of noisy detections (right group) fitting a holistic model is more robust.

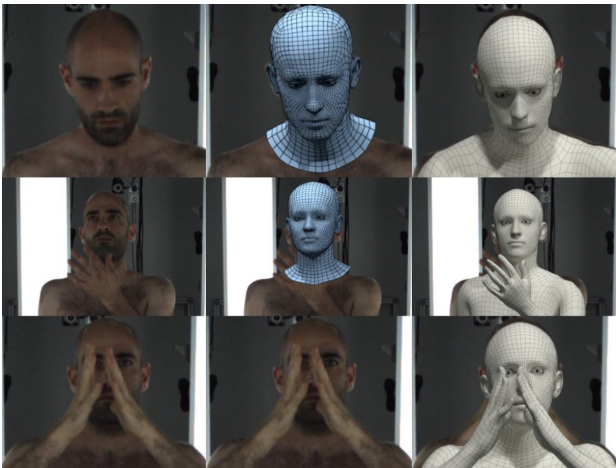


Figure A.3. Fitting SMPL-X (right) versus FLAME (middle). For minimal occlusions and frontal views (top) both methods perform well. For moderate (middle) or extreme (bottom) occlusions the body provides crucial context and improves fitting (bottom: missing FLAME model indicates a complete fitting failure).

For the case where f_t is the *intruder* and f_s is the *receiver* (similarly for the opposite case) the cone for the distance



Figure A.4. Failure cases for SMPLify-X with the female SMPL-X for expressive RGB images similar to the ones of Figures 1 and 2 of the main paper. In the left case, 2D keypoints are reasonable, but due to depth ambiguities the torso pose is wrong, while the head shape is under-estimated. In the right case, the arms and hands are occluded and due to lack of constraints the arm and hand pose is wrong. The ordinal depth for feet is estimated wrongly, while similarly to the left case the torso pose and head shape are not estimated correctly. *Left*: Input RGB image. *Middle*: Intermediate 2D keypoints from OpenPose. *Right*: SMPL-X fittings overlaid on the RGB image.

field Ψ_{f_s} is defined as

$$\Psi_{f_s}(v_t) = \begin{cases} |(1 - \Phi(v_t))\Upsilon(n_{f_s} \cdot (v_t - \mathbf{o}_{f_s}))|^2 & \Phi(v_t) < 1 \\ 0 & \Phi(v_t) \geq 1 \end{cases} \quad (2)$$

where $\mathbf{o}_{f_s} \in \mathbb{R}^3$ is the circumcenter and $r_{f_s} \in \mathbb{R}_{>0}$ the radius of the circumcircle for the *receiver* triangle. The term

$$\Phi(v_t) = \frac{\|(v_t - \mathbf{o}_{f_s}) - (n_{f_s} \cdot (v_t - \mathbf{o}_{f_s}))n_{f_s}\|}{-\frac{r_{f_s}}{\sigma}(n_{f_s} \cdot (v_t - \mathbf{o}_{f_s})) + r_{f_s}} \quad (3)$$

projects the vertex v_t onto the axis of the cone defined by the triangle normal n_{f_s} and going through the circumcenter \mathbf{o}_{f_s} . It then measures the distance to it, scaled by the radius of the cone at this point. If $\Phi(v) < 1$ the vertex is inside the cone and if $\Phi(v) = 0$ the vertex is on the axis. The term

$$\Upsilon(x) = \begin{cases} -x + 1 - \sigma & x \leq -\sigma \\ -\frac{1-2\sigma}{4\sigma^2}x^2 - \frac{1}{2\sigma}x + \frac{1}{4}(3 - 2\sigma) & x \in (-\sigma, +\sigma) \\ 0 & x \geq +\sigma \end{cases} \quad (4)$$

measures how far the projected point is from the circumcenter to define the intensity of penalization. For $\Upsilon(x) < 0$ the projected point is behind the triangle. For $x \in (-\sigma, +\sigma)$ the penalizer is quadratic, while for $x > |\sigma|$ it becomes linear. The parameter σ also defines the field of view of the cone. In contrast to [4, 28] that use *mm* unit and $\sigma = 0.5$, we use *m* unit and $\sigma = 0.0001$. For the resolution of our meshes, we empirically find that this value allows for both penalizing penetrations, as well as for not over-penalizing in case of self-contact, *e.g.* arm resting on knee.

As seen in Fig. A.5, for certain parts of the body, like the eyes, toes, armpits and crotch, as well as neighboring parts

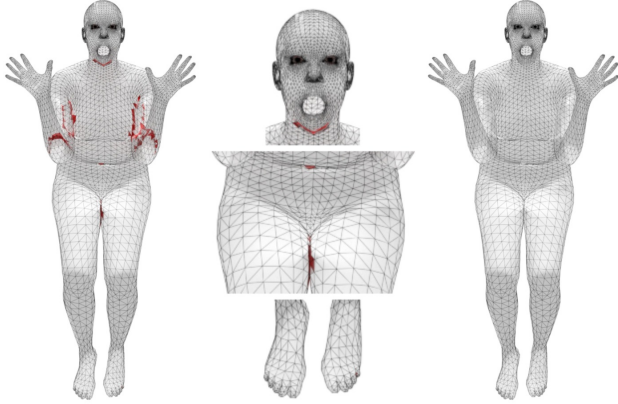


Figure A.5. For certain parts of the body, like the eyes, toes, armpits and crotch, as well as neighboring parts in the kinematic chain, there is either always or frequently self-contact. The triangles for which collisions are detected are highlighted with red (left, middle). Since the model does not model deformations due to contact, for simplicity we just ignore collisions for these areas (right).

in the kinematic chain, there is either always or frequently self-contact. For simplicity, since the model does not model deformations due to contact, we simply ignore collisions for neighboring parts in these areas. Our empirical observations suggest that collision detection for the other parts resolves most penetrations and helps prevent physically implausible poses. Figure A.6 shows the effect of the collision penalizer, by including or excluding it from optimization, and depicts representative success and failure cases.

For computational efficiency, we developed a custom PyTorch wrapper operator for our CUDA kernel based on the highly parallelized implementation of BVH [14].

3. Optimization

In Section 3.6 of the paper we present the main information about optimizing our objective function, while in the following we present omitted details.

To keep optimization tractable, we use a PyTorch implementation and the Limited-memory BFGS optimizer (L-BFGS) [22] with strong Wolfe line search. We use a learning rate of 1.0 and 30 maximum iterations. For the annealing scheme presented in Section 3.6 we take the following three steps. We start with high regularization to mainly refine the global body pose, ($\gamma_b = 1, \gamma_h = 0, \gamma_f = 0$) and gradually increase the influence of hand keypoints to refine the pose of the arms ($\gamma_b = 1, \gamma_h = 0.1, \gamma_f = 0$). After converging to a better pose estimate, we increase the influence of both hands and facial keypoints to capture expressivity ($\gamma_b = 1, \gamma_h = 2, \gamma_f = 2$). Throughout the above steps the weights $\lambda_\alpha, \lambda_\beta, \lambda_\epsilon$ in the objective function E start with high regularization that progressively lowers to allow for better fitting. The only exception is λ_C that progressively



Figure A.6. Effect of the collision penalizer. The colliding triangles are highlighted to show penetrations at the end of optimization with SMPLify-X without (middle) and with (right) the collision term in the objective function. The top row shows a successful case, where optimization resolves most collisions and converges in a physically plausible pose that reflects the input image. The bottom row shows a failure case, for which arm crossing causes a lot of collisions due to self-touch. The final pose (right) is still physically plausible, but optimization gets trapped in a local minima and the pose does not reflect the input image.

increases while the influence of hands and facial keypoints gets stronger in E_J , thus bigger pose changes and more collisions are expected.

Regarding the weights of the optimization, they are set empirically and the exact parameters for each stage of the optimization will be released with our code. For more intuition we performed sensitivity analysis by perturbing each weight λ separately by up to $\pm 25\%$. This resulted to relative changes smaller than 6% in the vertex-to-vertex error metric, meaning that our approach is robust for significant weight ranges and not sensitive to fine-tuning. The detailed results are presented in Fig. A.7.

4. Quantitative evaluation on “Total Capture”

In the main paper we present a curated dataset called *Expressive hands and faces dataset (EHF)* with ground-truth shape for bodies, hands and faces together.

Since the most relevant model is Frank [13], we also use the “Total Capture” dataset [8] of the authors, focusing on the “PtCloudDB” part that includes pseudo ground-truth for

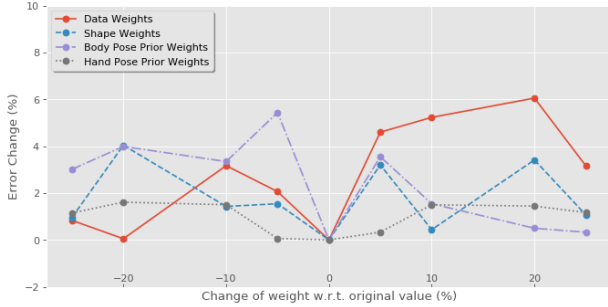


Figure A.7. Sensitivity of the weights for the different terms of the optimization. Each weight λ is perturbed separately up to $\pm 25\%$. The relative changes in the vertex-to-vertex error are smaller than 6%, indicating that our approach is robust for significant weight ranges and not sensitive to fine-tuning.

		SMPLify-X using	
Error Joints	Alignment Joints	GT 2D	pred 2D
Body	Body	92.6	117.5
Body+H+F	Body	101.2	136.2
Body+H+F	Body+H+F	71.2	93.4

Table A.1. Quantitative results on the selected frames from CMU Panoptic Studio, using SMPLify-X on the 2D re-projection of the ground-truth 3D joints, and the 2D joints detected by OpenPose respectively. The numbers are mean 3D joint errors after Procrustes alignment. First, we evaluate the error on the body-only keypoints after Procrustes alignment with the ground-truth body-only keypoints (row 1). Then, we consider the same alignment using body-only keypoints, but we evaluate the joint error across all the body+hands+face keypoints (row 2). Finally, we align the prediction using all body+hands+face keypoints and we report the mean error across all of them (row 3).

all body, face and hands. This pseudo ground-truth is created with triangulated 3D joint detection from multi-view with OpenPose [23]. We curate and pick 200 images, according to the degree of visibility of the body in the image, interesting hand poses and facial expressions. In the following, we refer to this data as “total hands and faces” (THF) dataset. Figure A.8 shows qualitative results on part of THF. For each group of images the top row shows a reference RGB image, the middle row shows SMPLify-X results using pseudo ground-truth OpenPose keypoints (projected on 2D for use by our method), while the bottom row shows SMPLify-X results using 2D OpenPose keypoints estimated with [23]. Quantitative results for this dataset are reported in Table A.1.

5. Quantitative evaluation on Human3.6M

In the main manuscript (Table 1), we demonstrated that evaluating the reconstruction accuracy using 3D body joints is not representative of the accuracy and the detail of a

Method	Mean (mm)	Median (mm)
SMPLify [5]	82.3	69.3
SMPLify-X	75.9	60.8

Table A.2. Quantitative results on the Human3.6M dataset [10]. The numbers are mean 3D joint errors after Procrustes alignment. We use the evaluation protocol of [5].

method’s reconstruction. However, many approaches do evaluate quantitatively based on 3D body joints metrics, so here we compare our results with SMPLify [5] to demonstrate that our approach is not only more natural, expressive and detailed, but the results are also more accurate in the common metrics. In Table A.2 we present our results using the Human3.6M [10] dataset. We follow the same protocol as [5] and we report results after Procrustes alignment with the ground-truth 3D pose. Even though there are several factors that improve our approach over SMPLify and this experiment does not say which is more important (we direct the reader to the ablative study in Table 2 of the main manuscript for this), we still outperform the original SMPLify using this crude metric based on 3D joints.

6. Qualitative evaluation on MPII

In Fig. A.14 we present qualitative results on the MPII dataset [3]. For this dataset we also include some cases with low resolution, heavily occluded or cropped people.

7. Model

In Section 3.1 of the main manuscript we describe the SMPL-X model. The model shape space is trained on the CAESAR database [26]. In Fig. A.9 we present the percentage of explained variance as a function of the number of PCA components used. All models explain more than 95% of the variance with 10 principle components.

We further evaluate the model on a held out set of 180 alignments of male and female subjects in different poses. The male model is evaluated on the male alignments, the female model is evaluated on the female alignments, while the gender neutral is evaluated on both male and female alignments. We report the model alignment vertex-to-vertex (v2v) mean absolute error as a function of the number of principle components used, shown in Fig. A.10.

8. VPoser

In Section 3.3 of the main manuscript we introduce a new parametrization of the human pose and a prior on this parameterization, also referred to as VPoser. In this Section we present further details on the data preparation and implementation.



Figure A.8. Qualitative results on some of the data of the “total capture” dataset [8], focusing on the “PtCloudDB” part that includes pseudo ground-truth for all body, face and hands. We curate and pick 200 images, according to degree of body coverage in the image and interesting hand poses and facial expressions. We refer to this data as “total hands and faces” dataset (THF). *Top row*: Reference RGB image. *Middle row*: SMPLify-X results using pseudo ground-truth OpenPose keypoints (3D keypoints of [8] estimated from multi-view and projected on 2D). *Bottom row*: SMPLify-X results using 2D OpenPose keypoints estimated with [23]. *Gray* color depicts the gender-specific model for confident gender detections. *Blue* is the gender-neutral model that is used when the gender classifier is uncertain.

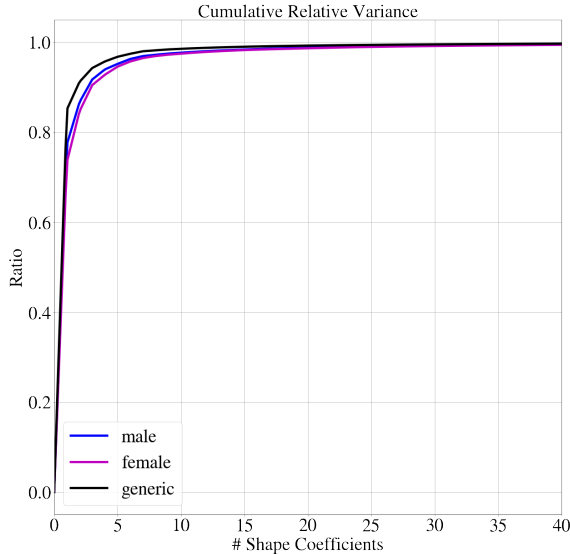


Figure A.9. Cumulative relative variance of the CAESAR dataset explained as a function of the number of shape coefficients for three SMPL-X models: male, female, gender neutral model.

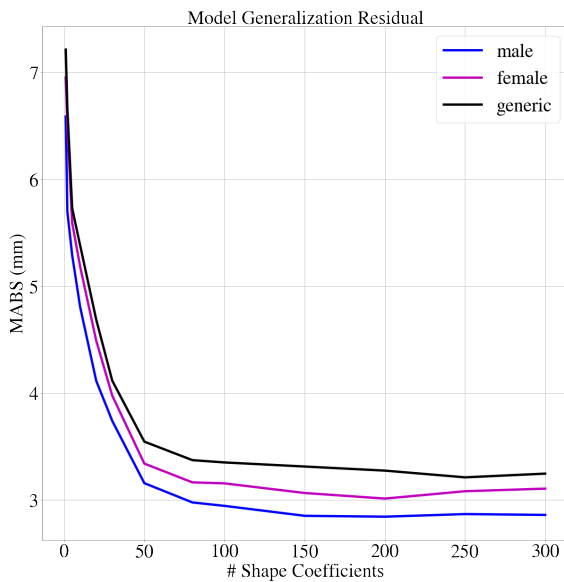


Figure A.10. Evaluating SMPL-X generalization on a held out test set of male and female 3D alignments.

8.1. Data preparation

We use SMPL pose parameters extracted from human motion sequences of CMU [7], Human3.6M [10], and PosePrior [2] as our dataset. These parameters were originally released by [19, 21]. Subsequently, we hold out parameters for Subjects 9 and 11 of Human3.6M as our test set. We randomly select 5% of the training set as our validation set and use that to make snapshots of the model with minimum validation loss. We choose matrix rotations for our pose parameterization.

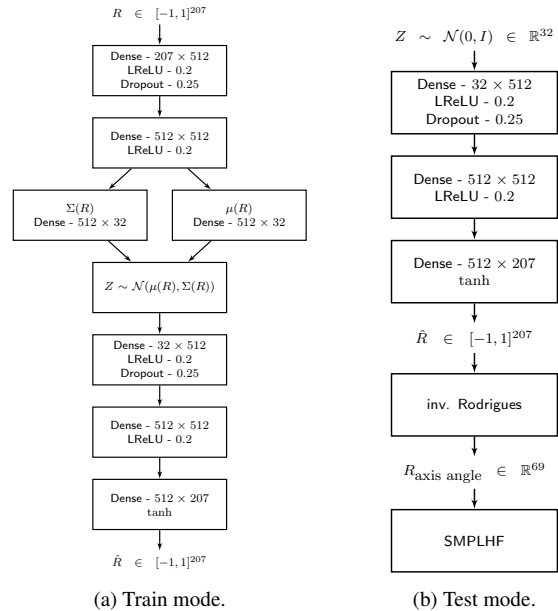


Figure A.11. VPoser model in different modes. For training the network consists of an encoder and a decoder. For testing we use the latent code instead of the body pose parameters, *i.e.* θ_b , of SMPL-X, which are described in Section 3.1 of the main paper. By “inverse Rodrigues” we note the conversion from a rotation matrix to an axis-angle representation for posing SMPL-X.



Figure A.12. Gender classifier results on the test set. From left to right column: Successful predictions, predictions discarded due to low confidence (< 0.9), failure cases.

8.2. Implementation details

For implementation we use TensorFlow [1] and later port the trained model and weights to PyTorch [25]. Figure A.11 shows the network architecture during training and test time. We use only fully-connected layers, with LReLU [20] non-linearity and keep the encoder and decoder symmet-

ric. The encoder has two dense layers with 512 units each, and then one dense layer for mean and another for variance of the VAE’s posterior Normal distribution. The decoder weights have the same shape as the encoder, only in reverse order. We use the ADAM solver [15], and update the weights of the network to minimize the loss defined in Eq. 5 of the main manuscript. We empirically choose the values for loss weights as: $c_1 = 0.005$, $c_2 = 1.0 - c_2$, $c_3 = 1.0$, $c_4 = 1.0$, $c_5 = 0.0005$. We train for 60 epochs for each of the following learning rates: $[5e-4, 1e-4, 5e-5]$.

After training, the latent space describes a manifold of physically plausible human body poses, that can be used for efficient 2D-to-3D lifting. Figure A.13 shows a number of random samples drawn from the latent space of the model.

9. Gender classifier

Figure A.12 shows some qualitative results of the gender classifier on the test set.

9.1. Training data

For training data we employ the LSP [11], LSP-extended [12], MPII [3], MS-COCO [18], LIP [17] datasets, respecting their original train and test splits. To curate our data for gender annotations we collect tight crops around persons and keep only the ones for which there is at least one visible joint with high confidence for the head, torso and for each limb. We further reject crops with size smaller than 200×200 pixels. The gathered samples are annotated with gender labels using Amazon Mechanical Turk. Each image is annotated by two Turkers and we keep only the ones with consistent labels.

9.2. Implementation details

For implementation we use Keras [6] with TensorFlow [1] backend. We use a pretrained ResNet18 [9] for feature extraction and append fully-connected layers for our classifier. We employ a cross entropy loss, augmented with an L2 norm on the weights. Each data sample is resized to 224×224 pixels to be compatible with the ResNet18 [9] architecture. We start by training the final fully-connected layers for two epochs with each of the following learning rate values $[1e-3, 1e-4, 1e-5, 1e-6]$. Afterwards, the entire network is finetuned end-to-end for two epochs using these learning rates $[5e-5, 1e-5, 1e-6, 1e-7]$. Optimization is performed using Adam [15].

Disclosure: MJB has received research gift funds from Intel, Nvidia, Adobe, Facebook, and Amazon. While MJB is a part-time employee of Amazon, his research was performed solely at, and funded solely by, MPI. MJB has financial interests in Amazon and Meshcapade GmbH.

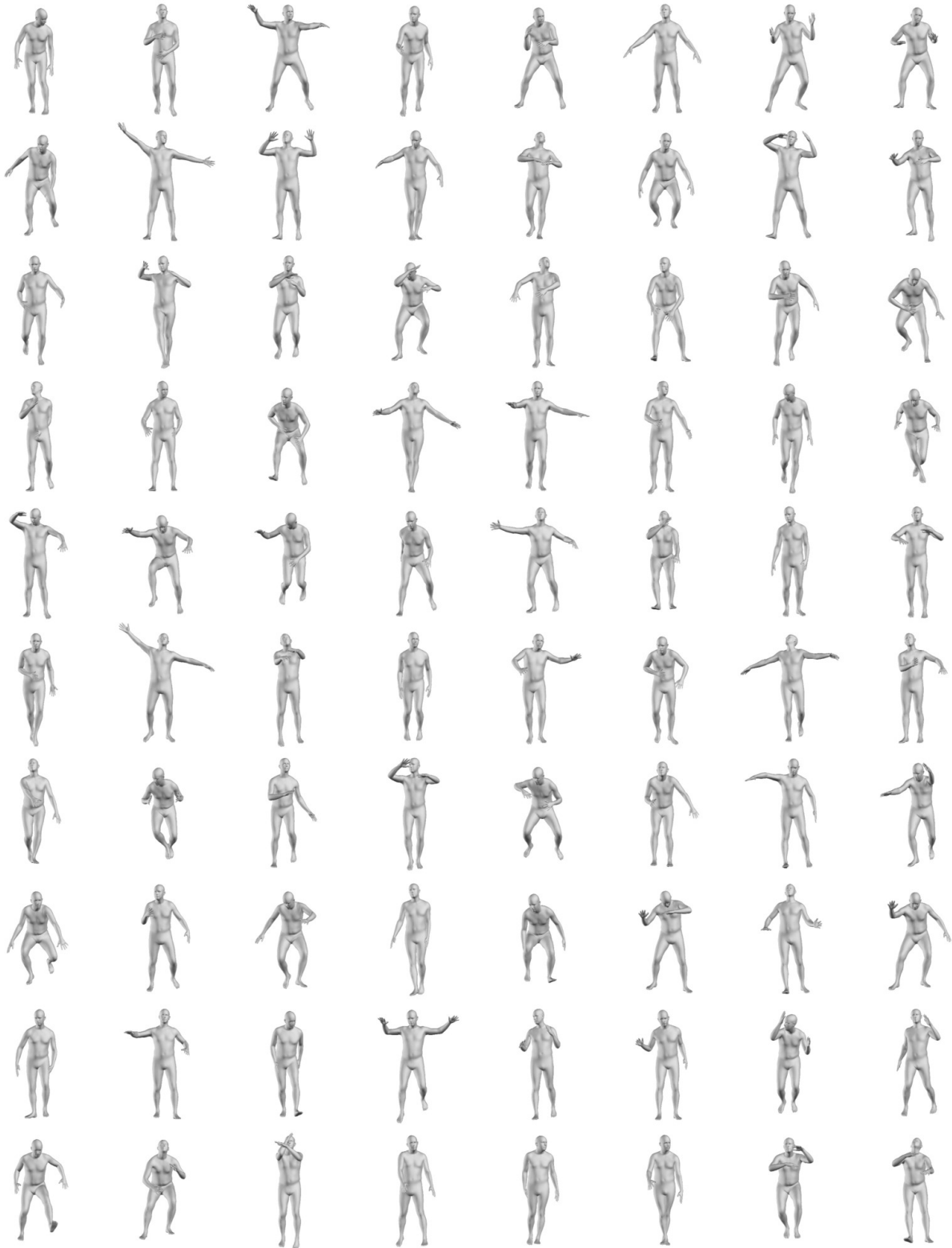


Figure A.13. Random pose samples from the latent space of VPoser. We sample from a 32 dimensional normal distribution and feed the value to the decoder of VPoser; shown in Figure A.11b. SMPL is then posed with the decoder output, after conversion to an axis-angle representation.

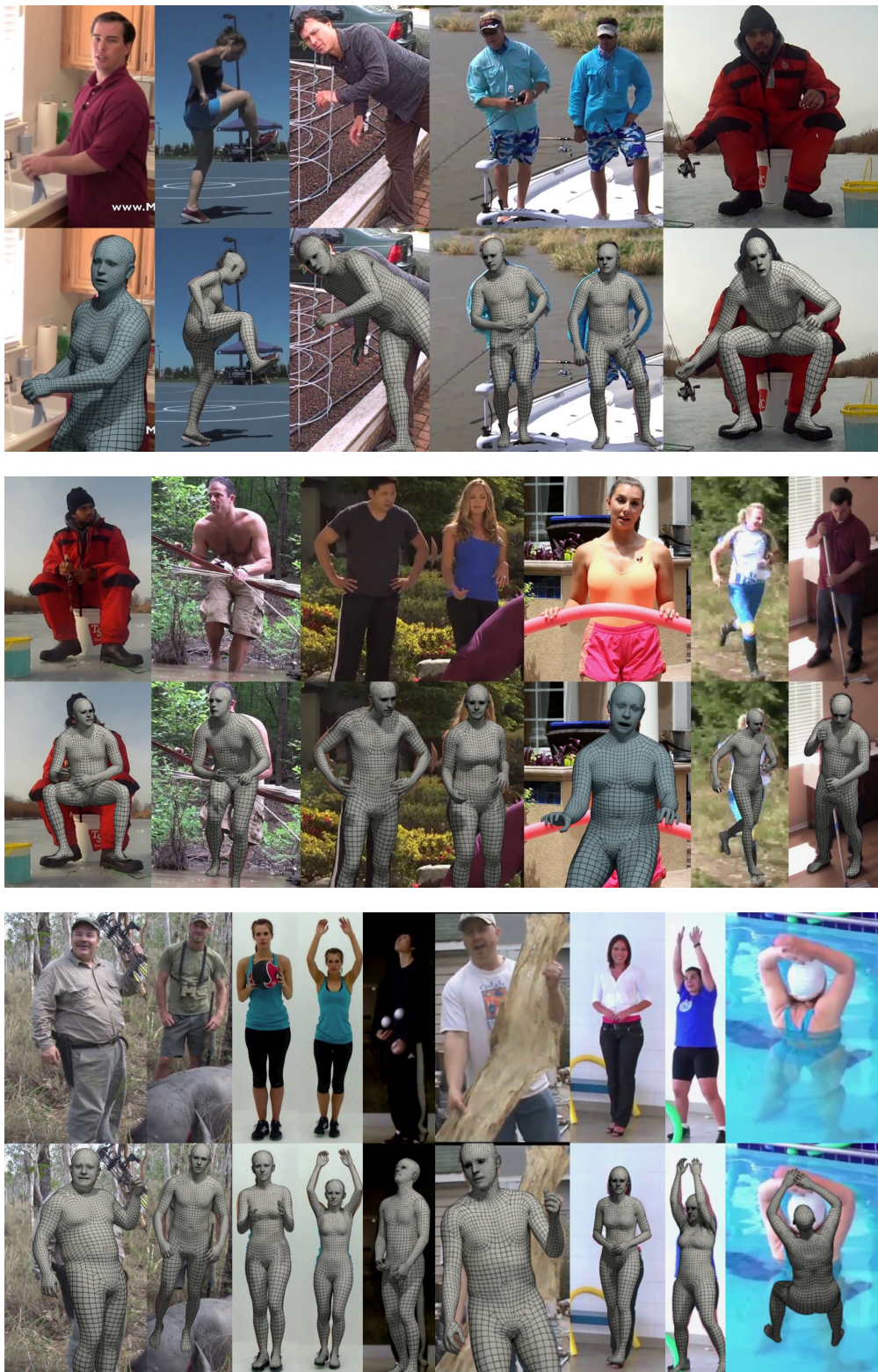


Figure A.14. Qualitative results of SMPLify-X with SMPL-X on the MPII dataset [3]. In this figure we also include images with some heavily occluded or cropped bodies. *Gray* color depicts the gender-specific model for confident gender detections. *Blue* is the gender-neutral model that is used when the gender classifier is uncertain or when cropping does not agree with the filtering criterion described in subsection 9.1.



Figure A.15. Results of SMPLify-X fitting for the LSP dataset. For each group of images we compare two body priors; the top row shows a reference RGB image, the bottom row shows results of SMPLify with VPoser, while the middle row shows results for which VPoser is replaced with the GMM body pose prior of SMPLify [5]. To eliminate factors of variation, for this comparison we use the gender neutral SMPL-X model.

References

- [1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016. 6, 7
- [2] Ijaz Akhter and Michael J. Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *CVPR*, 2015. 6
- [3] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2D human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014. 4, 7, 9
- [4] Luca Ballan, Aparna Taneja, Juergen Gall, Luc Van Gool, and Marc Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, 2012. 1, 2
- [5] 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. 4, 10
- [6] François Chollet et al. Keras. <https://keras.io>, 2015. 7
- [7] CMU. CMU MoCap dataset. 6
- [8] Total Capture Dataset. <http://domedb.perception.cs.cmu.edu>. 3, 5
- [9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 7
- [10] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *PAMI*, 36(7):1325–1339, 2014. 4, 6
- [11] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, 2010. 7
- [12] Sam Johnson and Mark Everingham. Learning effective human pose estimation from inaccurate annotation. In *CVPR*, 2011. 7
- [13] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3D deformation model for tracking faces, hands, and bodies. In *CVPR*, 2018. 3
- [14] Tero Karras. Maximizing parallelism in the construction of BVHs, Octrees, and K-d trees. In *Proceedings of the Fourth ACM SIGGRAPH / Eurographics Conference on High-Performance Graphics*, pages 33–37, 2012. 3
- [15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 7
- [16] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics (TOG)*, 36(6):194, 2017. 1
- [17] Xiaodan Liang, Chunyan Xu, Xiaohui Shen, Jianchao Yang, Si Liu, Jinhui Tang, Liang Lin, and Shuicheng Yan. Human parsing with contextualized convolutional neural network. In *ICCV*, 2015. 7
- [18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 7
- [19] Matthew M. Loper, Naureen Mahmood, and Michael J. Black. MoSh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 33(6):220:1–220:13, Nov. 2014. 6
- [20] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *ICML Workshops*, 2013. 6
- [21] N. Mahmood, N. Ghorbani, Troje N.F., G. Pons-Moll, and M.J. Black. AMASS: Archive of motion capture as surface shapes. In *preparation*, 2019. 6
- [22] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006. 3
- [23] OpenPose. <https://github.com/CMU-Perceptual-Computing-Lab/openpose>. 2, 4, 5
- [24] Paschalis Panteleris, Iason Oikonomidis, and Antonis Argiros. Using a single RGB frame for real time 3D hand pose estimation in the wild. In *WACV*, 2018. 1, 2
- [25] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pyTorch. In *NIPS-W*, 2017. 6
- [26] Kathleen M. Robinette, Sherri Blackwell, Hein Daanen, Mark Boehmer, Scott Fleming, Tina Brill, David Hoeflerlin, and Dennis Burnsides. Civilian American and European Surface Anthropometry Resource (CAESAR) final report. Technical Report AFRL-HE-WP-TR-2002-0169, US Air Force Research Laboratory, 2002. 4
- [27] Matthias Teschner, Stefan Kimmerle, Bruno Heidelberger, Gabriel Zachmann, Laks Raghupathi, Arnulph Fuhrmann, Marie-Paule Cani, François Faure, Nadia Magnenat-Thalmann, Wolfgang Strasser, and Pascal Volino. Collision detection for deformable objects. In *Eurographics*, 2004. 1
- [28] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient points and physics simulation. *IJCV*, 118(2):172–193, 2016. 1, 2