Weakly Supervised 3D Human Pose and Shape Reconstruction with Normalizing Flows *Supplementary Material*

Andrei Zanfir, Eduard Gabriel Bazavan, Hongyi Xu
William T. Freeman, Rahul Sukthankar, and Cristian Sminchisescu

Google Research
{andreiz,egbazavan,hongyixu,wfreeman,sukthankar,sminchisescu}@google.com

In this annex we provide additional implementation detail and quantitative insight for the different models used in the paper.

1 Architectures

We describe the architectures used to construct a normalizing flow prior. We illustrate the case where the input is represented in terms of 6D rotation variables. We assume 23 joints (corresponding to the SMPL kinematic hierarchy), each with 6 dimensions representing each rotation. Hence, the total dimension of the body pose representation is 138. For other rotation representations (e.g. angle-axis, rotation matrix) the same procedure applies.

Low-capacity version We test a low-capacity normalizing flow architecture with the following structure: FC138-PReLU-FC138-PreLU-FC138-PreLU-FC138-PReLU-FC138, with a total of $95,914$ trainable parameters. In comparison, VPoser[2] uses $344,190$ parameters. Note that in the backward pass from latent to ambient space we do not use matrix inversions – the fully connected layers are applied in a standard way.

Real-NVP version We also use a more complex normalizing flow architecture, which replaces the PreLU activation unit with a Real-NVP step. The structure is then FC138-RNVP-FC138-RNVP-FC138-RNVP-FC138-RNVP-FC138-RNVP-FC138, with a total of $331,462$ trainable parameters. For the Real-NVP unit, we use a simple FC128-Tanh-FC128-Tanh-FC69 architecture.

Training We use a custom TensorFlow implementation for all architectures. In training, the batch size is set to 64, and we use ADAM optimization with an initial learning rate of $1e{-}4$ and an exponential decay rate of 0.99 at every $10,000$ steps. The training is stopped after $200,000$ steps. For the AMASS dataset, this corresponds to ≈ 4 epochs.

2 Translation Estimation from 2d Keypoints

For all of our experiments, we assume a perspective camera model. In this case, one unknown is the global model translation, $\mathbf{T}$, which has to be either predicted

or estimated. Unfortunately, predicting a 3d translation directly is difficult with neural networks. In order to circumvent this, we propose the following solution: given a posed mesh $\mathbf{M}(\boldsymbol{\theta}, \boldsymbol{\beta})$, with skeleton joints $\mathbf{J}_{3d} = \{\mathbf{J}_i^{3d}, i = 1 \dots N_j\}$, projected skeleton joints $\mathbf{J}_{2d} = \{\mathbf{J}_i, i = 1 \dots N_j\}$ and detected 2d joint locations $\{\widehat{\mathbf{J}}_i\}$, we rewrite the keypoint alignment error as:

$$\begin{aligned} L_{KA} &= \frac{1}{N_j} \sum_i \|\mathbf{J}_i - \widehat{\mathbf{J}}_i\|_2 \\ &= \frac{1}{N_j} \sum_i \|\Pi(\mathbf{J}_i^{3d} + \mathbf{T}) - \widehat{\mathbf{J}}_i\|_2 \end{aligned} \tag{1}$$

where Π is the perspective projection operator. By relaxing the operator to a weak-perspective one, Π_W, we can solve for translation directly, by using least-squares:

$$\mathbf{T}^* = \arg\min_{\mathbf{T}} \frac{1}{N_j} \sum_i \|\Pi_W(\mathbf{J}_i^{3d} + \mathbf{T}) - \widehat{\mathbf{J}}_i\|_2^2 \tag{2}$$

Note that (2) is used only to predict the global translation, whereas (1) is used afterwards to compute the keypoint alignment loss, based on the estimated $\mathbf{T}^*$. Gradients will flow to all the variables of the network, through both layers implementing the above operations.

3 Normalizing Flows and VPoser on 3DPW

In this experiment, we compare our light-version normalizing flow prior (trained on AMASS) with the prior of [2], on 500 random images sampled from the 3DPW dataset. In this study, the 2d keypoints and semantic segmentation are predictions from a deep-neural network we trained, and images have ground-truth 3d meshes which permits evaluation. We fit the SMPL model in the same conditions (starting from 4 globally rotated 0-mean latent space kinematic initializations, using both KA and KA+BA losses), for both priors, and report errors in fig. 1.

4 Self-supervised Learning on COCO and OpenImages

In order to further explore the effect of additional self-supervision to our training process, we extended the set of in-the-wild images with a subset of OpenImages[1]. OpenImages contains various annotations from which we used the ones related to people (bounding boxes) in various shapes and poses. We kept the images on which the keypoint detector component of our network predicted enough keypoints with high confidence. Using the 2D keypoints and the semantic segmentation predictions from the network we extended the training data with up to 70,000 samples, including the initial COCO data mentioned in the

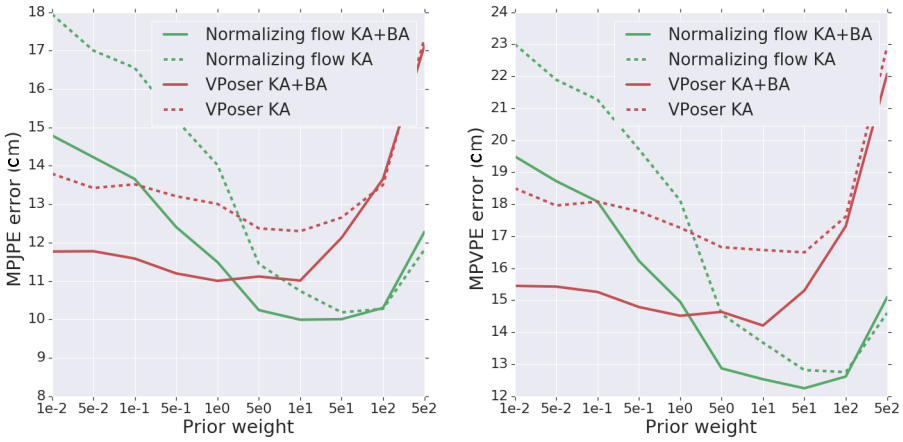

Fig. 1. Reconstruction errors, MPJPE (left plot) and MPVPE (right), for different priors and loss functions. Notice that normalizing flow priors reduce the reconstruction error in all cases.

paper. We gradually increased the amount of data from 10% up to 100% used to further train the network and we show results on the 3DPW test set. As can be seen in table 1, the 2D joint error is decreasing and the overlap mIOU metric is increasing showing that with more self supervision the predictions of the network get better. As can be seen in the paper this also leads to better 3D predictions.

Method	2D Joints Error (pixels)	mIOU
FS	9.13	42.5
WS+KA+BA-10%	7.2	45.5
WS+KA+BA-30%	6.27	47.86
WS+KA+BA-60%	**5.4**	47.06
WS+KA+BA-100%	5.8	**50.0**
WS+KA-10%	6.91	41.8
WS+KA-30%	6.0	42.2
WS+KA-60%	5.7	43.0
WS+KA-100%	5.6	44.0

Table 1. Self supervised experiments on the 3DPW test set using COCO and OpenImages data for additional training. FS identifies the model trained only using full supervision. WS is the model trained weakly supervised. KA and BA denote the keypoint, respectively the body part alignment losses. We gradually increased the amount of self supervised data used for refining the network, which was initially trained fully supervised. We observe that the 2D joint error (measured in pixels) is decreasing as we add more data. As expected, the mIOU metric is increasing–more so in the case where the KA+BA loss is used denoting better alignment. Usually decreases in KA correlate with increases in BA, although this is not always the case– one can expect that a certain lack of calibration between the 2d detected skeletons and the 3d SMPL counterpart, or an aggressive maximization of overlap when clothing makes it difficult to correctly segment body parts, could lead to potential inconsistencies between the trends of the two losses. In practice we find the model image alignment to be much better for BA than for KA, with 3d reconstructions that are perceptually good.

References

1. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Duerig, T., Ferrari, V.: The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. arXiv:1811.00982 (2018)
2. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A., Tzionas, D., Black, M.: Expressive body capture: 3d hands, face, and body from a single image. In: CVPR (2019)