Learning to reconstruct 3D human pose and shape via model-fitting in the loop

1. Instruction

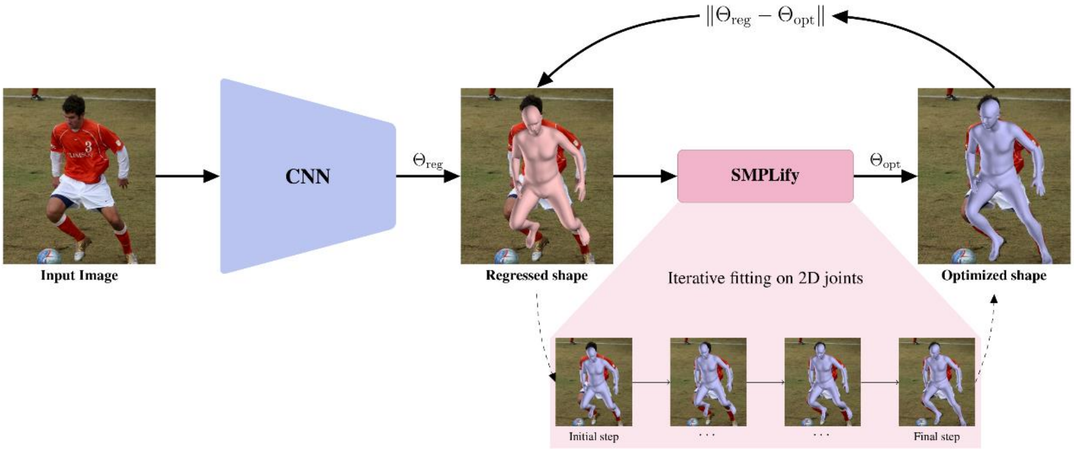
	Regression-based	Optimization-based
Strength	Can take all pixel values into consideration instead relying only on a sparse set of 2D location	Typically get a good fit
Weakness	A large amount of data is necessary to properly train the network	Slow and sensitive about choice of initialization

- **SPIN (SMPL oPtimization IN the loop)**

1) Deep network is used to regress the parameters of SMPL parametric model

2) These regressed values initialize the iterative fitting

3) Parameters of fitted model are used as supervision for network closing the loop between regression and optimization method



- **Self-improving model**

1) In early stage, network produce result which iterative fitting is prone to make error. Then more examples are provided to network as supervision by iterative fitting module and network produce more meaningful shapes. Eventually, this lead

optimization to more meaningful shapes.

2) Iterative fitting only requires 2D key point to fit model. This makes network can be trained even when no image with corresponding 3D GT is available because 3D supervision will be provided by optimization module.
3) Network is trained with explicit 3D supervision instead of weaker 2D reprojection errors and this improve the regression performance

2. Related work

- Optimization based method

  Early works estimate the parameters of the SCAPE model using silhouettes or key points and often there was some user intervention needed.

  Recently, fully automatic approach, SMPLify was introduced. It use off-the-shelf key point detector and fits SMPL to 2D key point detection.

  Works have demonstrated fits for more expressive models in multi-view setting.

  This work uses SMPLify to supply direct supervision for neural network.

- Regression based method

  Due to lack of images with full 3D shape GT, the majority of these works have focused on alternative supervision signals to train the deep networks. Most of them rely heavily on 2D annotation and they are providing only weak supervision for the network.

  This work uses strong model-based supervision which is direct supervision on the model parameters and output mesh. This work uses fitting routine in training loop to provide strong supervision signal to train the network.

- Iterative fitting meets direct regression

  Early optimization methods required a good initial estimate which could be obtained by discriminative approach.

  1) *Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black,*

and Peter V Gehler. Unite the people:Closing the loop between 3D and 2D human representations. In CVPR, 2017.

➔ Used SMPLify to get good model fit.

2) *Gregory Rogez, PhilippeWeinzaepfel, and Cordelia Schmid. LCR-Net++: Multi-person 2D and 3D pose detection in natural images. PAMI, 2019.*
   ➔ Used 3D pose pseudo annotations for training.
3) *Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3D human pose and shape from a single color image. In CVPR, 2018.*
   ➔ Used initial prediction from their network to initialize and anchor the SMPLify optimization routine.
4) *Gul Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. BodyNet: Volumetric inference of 3D human body shapes. In ECCV, 2018.*
   ➔ Proposed Extension of SMPLify to fit SMPL on the regressed volumetric representation of their network.

This work proposed more tighter collaboration by incorporating the fitting method within training loop.

3. Technical approach
   1) SMPL model
   Provide function

$$\mathrm{M}(\theta,\beta) = M \ \in R^{N\times 3} \ (N = 6890\ vertices)$$

$$\theta : \text{pose parameter}, \beta : \text{shape parameter}$$

Body joints X of model can be defined as a linear combination of mesh vertices.
A linear regressor W can be pre-trained with k joints of interests.

$$\mathrm{X} \ \in R^{k\times 3} = WM$$

   2) Regression network
   Use deep neural network. This architecture has same design with
   Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In CVPR, 2018
   with the only difference that proposed by
   Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In CVPR, 2019
   for the 3D rotations since it empirically observed faster convergence.

Forward pass of new image provides the regressed prediction for

$$model\,parameter : \ \Theta_{reg} = \{\,\theta_{reg}\,, \beta_{reg}\}$$

$$cameraparameter: \Pi_{reg}$$

These parameters allow us to estimate the 2D projection of the joints

$$J_{reg} = \Pi_{reg}(X_{reg})$$

Prediction of this work allows us to generate the mesh corresponding to the regressed parameters

$$M_{reg} = M(\theta_{reg}, \beta_{reg})$$

Common supervision is provided using a reprojection loss on the joints

$$L_{2D} = \left\| J_{reg} - J_{gt} \right\|$$

$$J_{gt}: \ GT\ 2D\ joints$$

This supervisory signal is very weak.

In this work, put an extra burden on the network, forcing to search in the parameter space for a valid pose that agrees with the GT 2D locations.

3) Optimization routine

Iterative fitting routine follows the SMPLify work by

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.

Total objective is

$$E_J(\beta, \theta; K, J_{est}) + \lambda_\theta E_\theta(\theta) + \lambda_a E_a(\theta) + \lambda_\beta E_\beta(\beta)$$

$$\beta, \theta: parameters\ of\ SMPL\ model, J_{est}: detected\ 2D\ joints,$$
$$K: camera\ parameter$$

$E_J(\beta, \theta; K, J_{est}):$
$penalty\ on\ the\ weighted\ 2D\ distance\ between\ J_{est}\ and\ projected\ SMPL\ joints.$

$E_\theta(\theta):$
$mixture\ of\ Gaussians\ pose\ prior\ trained\ with\ shapes\ fitted\ on\ marker\ data$

$E_a(\theta): pose\ prior\ penalizing\ unnatural\ rotations\ of\ elbows\ and\ knees$

$E_\beta(\beta): quadratic\ penalty\ on\ the\ shape\ coefficient$

SMPLify involves an optimization over the camera translation and body orientation, while keeping the model pose and shape fixed. After estimating camera translation, SMPLify attempts to minimize objective using single optimization stage instead of 4-stage optimization because it enough to converge to good fit. Also, instead of estimating initial translation using triangle similarity, this work use predicted camera translation from network.

Another modification is run SMPLify in parallel.

4) SPIN

During training loop, an image is forwarded through the network providing the regressed parameters Θ_{reg}. These parameters are used to initialize the optimization routine. Given reasonable initial estimate, routine can be accelerated.

Let $\Theta_{opt} = \{\theta_{opt}, \beta_{opt}\}$ is set of model parameters produced by iterative fitting. These values are explicitly optimized $M_{opt} = M(\theta_{opt}, \beta_{opt})$ and reprojected joints J_{opt}. We can directly supervise network function.

$$L_{3D} = \left|\left|\Theta_{reg} - \Theta_{opt}\right|\right| \text{ (on parameter level)}$$

$$or$$

$$L_M = \left|\left|M_{reg} - M_{opt}\right|\right| \ (on\ mesh\ level)$$

Instead of forcing the network to identify a set of parameters that satisfy the joints reprojection, this work supplies it directly with a parametric solution.

SPIN is self-improving. Good initial network estimate Θ_{reg} will lead the optimization to a better fit $\Theta_{opt,}$ while a good fit from iterative routine will provide even better supervision to the network.

Since optimization routine uses only 2D joints for fitting and network relies on this routine for model-based supervision, this work applicable even in cases where no image with corresponding 3D GT is available for training.

5) Implementation details

During SMPLify, some cases can get bad failure. This make training unstable. This work use criterion to reject supervision.

- ➔ Simple thresholding based on the joint reprojection error
- ➔ Avoid training with improbable values for the shape parameters.
  When SMPLify returns shape values outside of range, this work only supervises β parameters with simple L_2 loss. (push close to mean shape)

Also, this work incorporated dictionary.

- ➔ Can keep track of the best fit which have seen for it over all epochs.
- ➔ Dictionary is initially populated with SMPLify fits, process done before training starts.

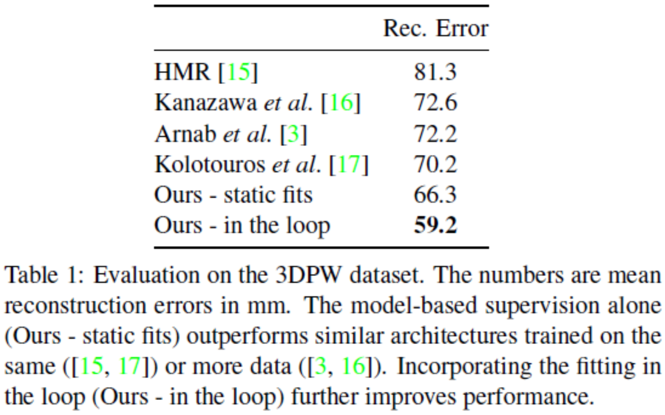
4. Empirical evaluation

1) Dataset

- Train using Human3.6M, MPI-INF-3DHP, LSP and report result on Human3.6M, MPI-INF-3DHP, LSP, 3DPW.

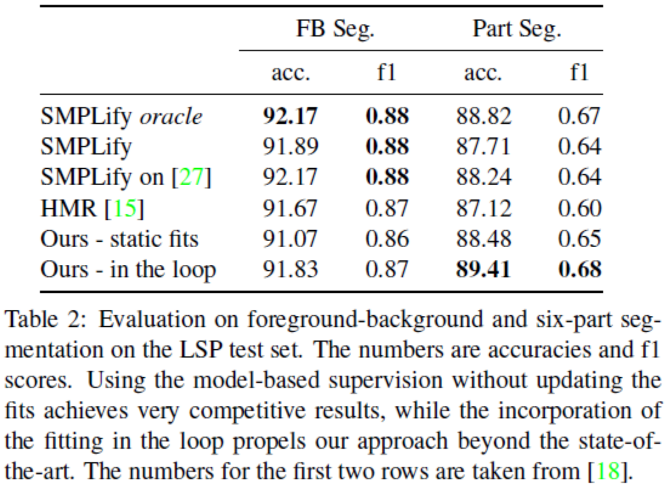
-Also, incorporate training data with 2D annotation from other dataset (LSP-Extended,. MPII, COCO).

2) Quantitative evaluation



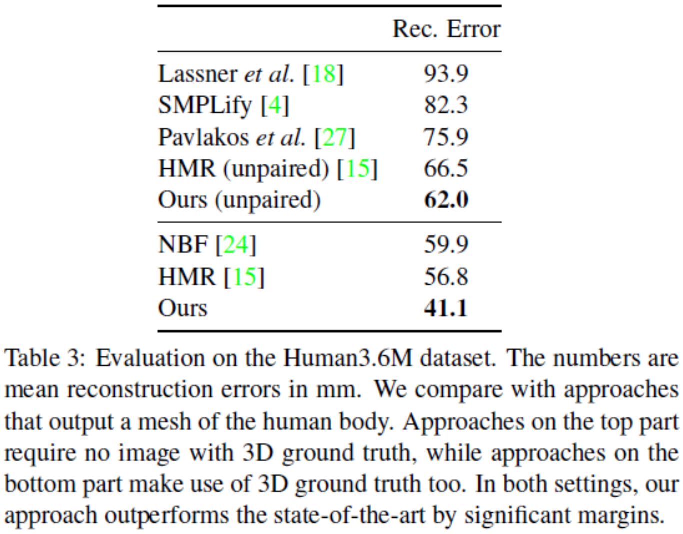
	Rec. Error
HMR [15]	81.3
Kanazawa *et al.* [16]	72.6
Arnab *et al.* [3]	72.2
Kolotouros *et al.* [17]	70.2
Ours - static fits	66.3
Ours - in the loop	**59.2**

Table 1: Evaluation on the 3DPW dataset. The numbers are mean reconstruction errors in mm. The model-based supervision alone (Ours - static fits) outperforms similar architectures trained on the same ([15, 17]) or more data ([3, 16]). Incorporating the fitting in the loop (Ours - in the loop) further improves performance.



	FB Seg.		Part Seg.	
	acc.	f1	acc.	f1
SMPLify *oracle*	**92.17**	**0.88**	88.82	0.67
SMPLify	91.89	**0.88**	87.71	0.64
SMPLify on [27]	92.17	**0.88**	88.24	0.64
HMR [15]	91.67	0.87	87.12	0.60
Ours - static fits	91.07	0.86	88.48	0.65
Ours - in the loop	91.83	0.87	**89.41**	**0.68**

Table 2: Evaluation on foreground-background and six-part segmentation on the LSP test set. The numbers are accuracies and f1 scores. Using the model-based supervision without updating the fits achieves very competitive results, while the incorporation of the fitting in the loop propels our approach beyond the state-of-the-art. The numbers for the first two rows are taken from [18].



	Rec. Error
Lassner *et al.* [18]	93.9
SMPLify [4]	82.3
Pavlakos *et al.* [27]	75.9
HMR (unpaired) [15]	66.5
Ours (unpaired)	**62.0**
NBF [24]	59.9
HMR [15]	56.8
Ours	**41.1**

Table 3: Evaluation on the Human3.6M dataset. The numbers are mean reconstruction errors in mm. We compare with approaches that output a mesh of the human body. Approaches on the top part require no image with 3D ground truth, while approaches on the bottom part make use of 3D ground truth too. In both settings, our approach outperforms the state-of-the-art by significant margins.

	Absolute			Rigid Alignment		
	PCK	AUC	MPJPE	PCK	AUC	MPJPE
HMR (unpaired) [15]	59.6	27.9	169.5	77.1	40.7	113.2
Ours (unpaired)	**66.8**	**30.2**	**124.8**	**87.0**	**48.5**	**80.4**
Mehta *et al.* [22]	75.7	39.3	117.6	-	-	-
VNect [23]	**76.6**	**40.4**	124.7	83.9	47.3	98.0
HMR [15]	72.9	36.5	124.2	86.3	47.8	89.8
Ours	76.4	37.1	**105.2**	**92.5**	**55.6**	**67.5**

Table 4: Evaluation on the MPI-INF-3DHP dataset. The comparison is under different metrics before (left) and after (right) rigid alignment. Our approach outperforms the previous baselines. (For PCK and AUC, higher is better, while for MPJPE, lower is better).