Human Model Learning from RGB with Depth Assistance

Balamurugan Thambiraja
balamurugan.thambiraja@tum.de

Aljaž Božič
aljaz.bozic@tum.de

Matthias Nießner
niessner@tum.de

Abstract

Model based human pose estimation from an RGB image is a long standing problem in computer vision. Current state-of-the-art methods rely on data-driven priors that are built from limited 3D data captured using complex multi-camera setups. In contrast, we propose RGB-D based self-supervised training of a deep network that (i) learns to predict the pose parameters of a human from an RGB image. (ii) learns to regress shape and person specific shape displacements which which enables use of the method for scale-aware 3D reconstruction task. Specifically, at train time we overfit to shape and delta displacements using RGB-D video. At test time, we predict the pose parameters from a single RGB image. In addition, we also introduce a method to generate quality 3D supervision data by leveraging an existing human pose tracking method and the depth map. Through detailed experiments, we analyze how different components of our method impact performance, especially the proposed data generation pipeline, and present an efficiently trainable framework for 3D human shape and deformation from from RGB images.

1. Introduction

Human performance capture and 3D human model acquisition from RGB involves predicting pose, shape and geometry details of the person in a given image. Such human performance capture systems has various application in the entertainment, VR/AR, reenactment and recently sport analytics.

Traditional optimization based approaches typically optimize an energy function that measures how well a statistical human model like SMPL [14, 17] fits a given image. An example of a typical objective function could be 2D joint loss, where the re-projection of 3D joint are compared with the 2D joint acquired using state-of-the-art human pose detectors [4, 6]. One of the drawback of such methods are the dependence of the initialization to be close the final/true solution. Most recent approaches use CNN based method to learn a mapping to predict parameters of statistical human model from a single RGB image and these methods mostly require paired data for training. Most commonly used supervision loss function such tasks are 2D joint loss, 3D joint loss and silhouette loss. Training data of such kind is very hard to acquire and require rigorous human annotation.

For 3D human model acquisition task, the most common way of acquiring such models are using 3D laser scanner or complicated multi-camera setup. With availability of affordable RGB-D sensors like Microsoft Kinect or Intel Realsense; methods like KinectFusion, DynamicFusion enables capture of static scene, but the dynamic reconstruction are not stable for fast motion and big occlusion, which often happen at natural human motion. Since assuming human prior make problem constrained, line of methods like Double Fusion [25] combine volumetric dynamic reconstruction with data driven statistical body model to simultaneously reconstruct detailed geometry, non-rigid motion from single RGB-D sensor. Thiemo *et al* learns a CNN based mapping function to regress pose, shape and geometry details from RGB using 3D and 2D supervision. Similar to methods in human pose estimation, these method also require quality supervision data, which are hard to acquire.

In human performance capture problem the commonly used supervision signals are 2D joint loss and 3D joint loss [16]. while the 2D joint can be readily obtained from state-of-the-art pose detectors [4, 6], the 3D joint requires more complicated multi-camera setups or scanning labs to generate, which makes them very hard to acquire for any arbitrary sequence that we want. Hence in this work, we propose to utilize the knowledge of the existing human pose detectors and depth to quality 3D supervision data in the form of depth silhouettes. In addition,we also propose the very first self-supervised clothed human model learning from RGB images. In short, our contributions can be summarized as

- A self-supervised learning approach to learn clothed human model from any arbitrary RGB-D sequence.
- A human pose tracking pipeline from RGB images.
- A novel method to generate robust 3D supervision data in the form of depth silhouettes.

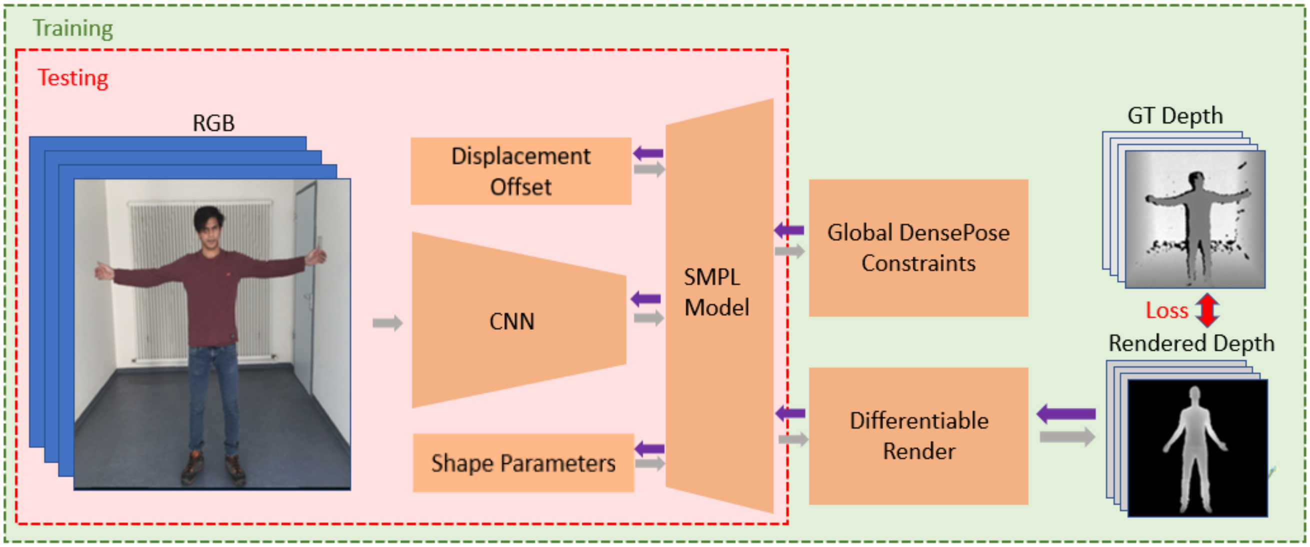

Figure 1. Overview of approach: Our novel CNN learns to predict pose parameters of the SMPL model from RGB images. At training time, our method uses RGB-D frames to learn the mapping function from pixel space to SMPL space. In addition to that, our method estimates the shape and per-vertex offset displacements on top of SMPL model. At test time, our method predicts the pose parameters from the RGB image using the estimated shape and per-vertex offset displacements. Our method uses face barycentric correspondence(Global DensePose constraints) (Fig.2) and depth silhouettes (Fig.3) to train the model and we use differentiable renderer (Section3.5) to render down depth silhouettes during training.

2. Related works

3D Human model acquisition: Human 3D model acquisition is fundamentally challenging task. Methods using complicated multi-camera setups with laser scanners achieve impressive results. However, this multi-camera setup is a limitation for multiple applications. Earlier methods [10, 15, 24] used RGB-D sensors to incremental fuse and reconstruct 3D model to canonical pose. Nonetheless these suffer from drift and occlusions. Tao Yu *et al.* [25] combined volumetric fusion with data driven statistical human model to reconstruct dynamically deforming human model from RGB-D frames. One of the drawbacks of these methods is they cannot handle fast motion. Recently, Thielo *et al.* [2] used a learning based to learn personalized body shape from single RGB camera. While this method really on 3D ground truth and silhouettes for training, our method is based on self-supervision generated using an existing human pose estimators.

Pose estimation: Pose and shape estimation from RGB image is another challenging problem closely related to 3D human model acquisition problem. Earlier methods [20, 1] relied on manual input to fit a parametric model to the RGB images. Bogo *et al.* [3] proposed a full optimization based approach to fit the SMPL [14] to 2D joints estimated using off the shelf pose detector [4]. Another method [13] followed a similar approach with additional silhouette based constraints. Omran *et al.* [16] integrated SMPL model within a CNN leveraging 2D joint detections to directly regress pose and shape parameters from RGB image. Other methods [21, 22, 23, 18] further extended idea with added constraints and self-supervision with varying inferring cues RGB, RGB+Joints [18]. H.F.Tung *et al.* [23] combined supervision from synthetic data with self-supervision from differentiable rendering in an end-to-end framework. Recently, Angjoo *et al.* [11] proposed an approach to learn human dynamics from RGB images, the method used 2D joints as well as 3D joints to train the network to predict SMPL parameters temporally. In this work, we go beyond these approaches by self-supervisely training a method to overfit shape and per-vertex offsets on the RGB-D frames and at test time we use the network to predict pose parameters from a single RGB frame. Compared to other methods, we can use any arbitrary sequence for training and we can also use our approach for scale aware 3D reconstruction.

3. Method

Our goal is to train a deep network from which we can (i) predict pose of human from RGB images (ii) regress shape and person specific shape displacements to capture the human body on top of data-driven human priors. To this end, we train a convolution neural network to learn the 3D human pose from the supervision detailed below. An overview of our method is provided in Fig. 1.

Training: Our network is trained to predict the parameters of the human model represented via SMPL [14] model (Section3.1). Specifically, at train time we estimate

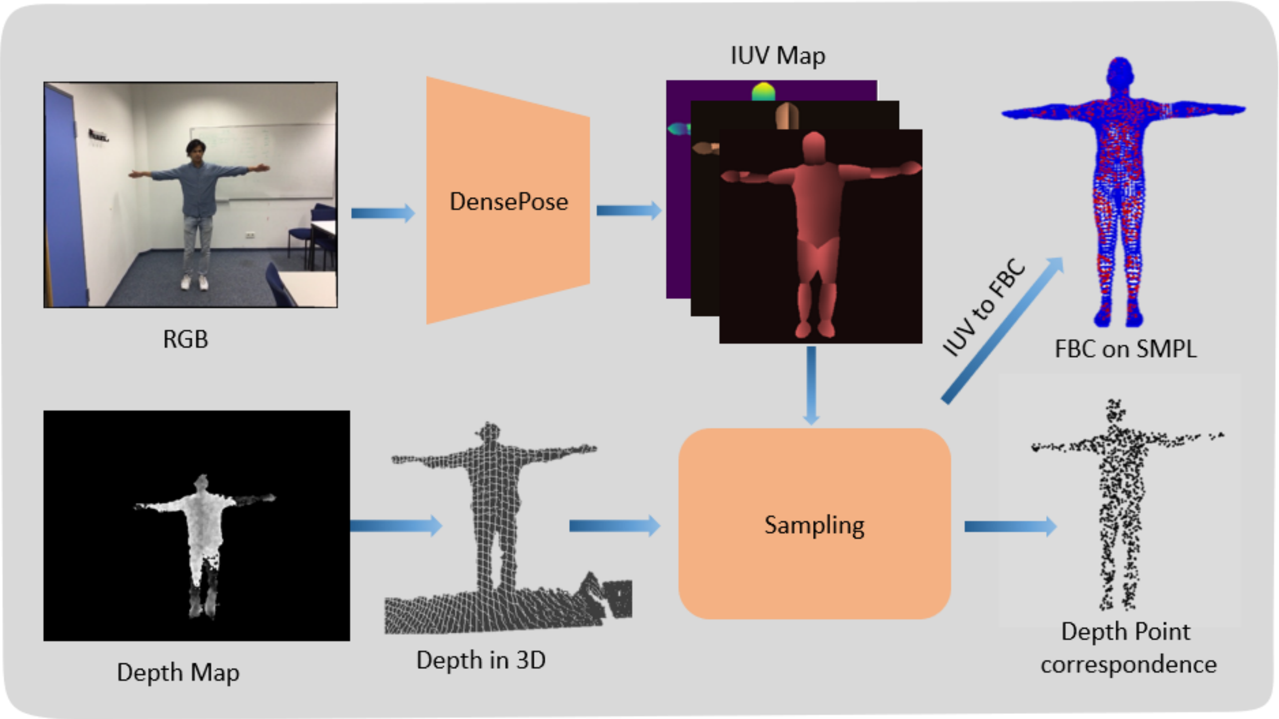

Figure 2. Overview of the face barycentric correspondence generation pipeline: We first extract the IUV maps by running DensePose [7] over our input images. In parallel, we lift the depth map to 3D depth map by back-projection. We sample IUV's from the IUV map and extract the corresponding face barycentric correspondence via IUV to FBC transformation given by the DensePose. Then we generate an one to one correspondence set by back-projecting the absolute (x,y) values of the IUV's in the RGB image to lifted 3D depth map using intrinsic transformation and querying the corresponding the 3D point in the back-projected pixel.

the shape and per-vertex offsets from the RGB-D frames and at test time we predict pose from RGB frames. The network learns per-frame pose parameters for every input frame. It also learns shape parameters and person specific vertex-displacements that are shared across all frames. We train our network in self-supervised manner using (i) Face barycentric correspondence (ii) Depth silhouette, generated using DensePose [7] prediction and depth map, see Section3.3. We also train our model using differentiable renderer to render down the SMPL model into depth map for our loss function, see Section3.5. In Section3.5 we propose a set of the loss functions that are used to train our model.

Testing: At test time, our network is capable of predicting the pose from RGB images using the estimated shape and per-vertex displacements learned during training. In essence, we can use our model for human pose prediction from both single and multi-frame setup.

3.1. Human representation

Similar to previous methods [16, 11, 7, 2], we use SMPL statistical body model [14] for representing undressed body, and we also learn a set of per-vertex offsets D which enable us to model beyond SMPL. SMPL(M) maps pose $\theta \in \mathbb{R}^{72}$ and shape $\beta \in \mathbb{R}^{10}$ parameters to mesh of vertices V = 6890. The whole function is expressed as

$$M(\beta, \theta, \delta, D) = W(T(\beta, \theta, \delta, D), J(\beta), \theta, W) \quad (1)$$

$$T(\beta, \theta, \delta, D) = T + B_s(\beta) + B_p(\theta) + D + \delta \quad (2)$$

where the function $W(.)$ represent linear blend-skinning with weights W, T is the template mesh or SMPL body model in rest pose, $B_s(\beta)$ and $B_p(\theta)$ are the pose and shape blend shapes which add pose and shape specific deformations to the template mesh, D is the per-vertex displacements which are added to the template mesh, δ represent the additional translation we learn to transform the model between camera coordinate and SMPL model coordinate, $J(\beta)$ represent the 3D skeletal joint locations of the SMPL model. In general, SMPL plus offsets are represented as SMPL+D and all together the pipeline is fully differentiable and it allows us to use SMPL as fixed layer in our pipeline.

3.2. Model representation

Given a set of RGB images $I_p = \{I_i\}_{i=1}^N$ of the subject from different sides and the corresponding depth map $Z_p = \{Z_i\}_{i=1}^N$, we learn a mapping function f_{fit} from RGB pixel space to SMPL+D space. Our mapping function is expressed as

$$f_{fit} : (I_p; \Theta, \beta, D) \mapsto (\theta, \delta) \quad (3)$$

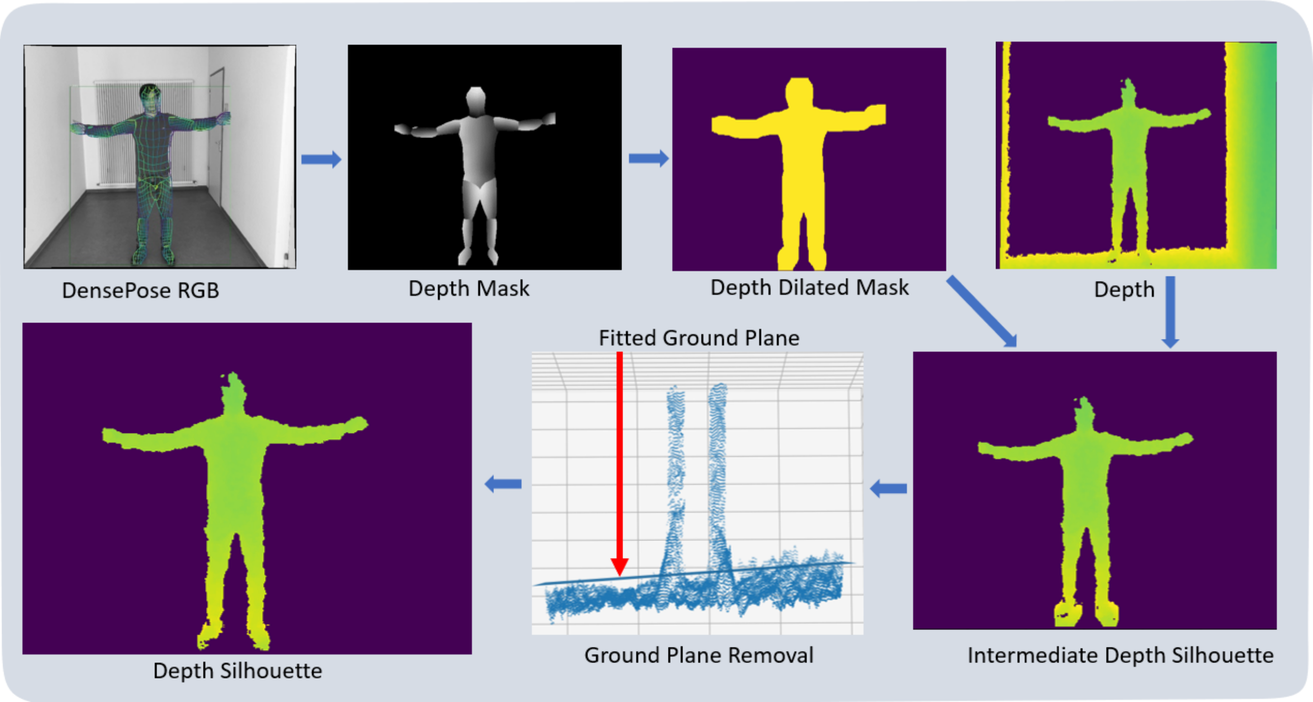

Figure 3. Overview of depth silhouette generation pipeline: We first extract the silhouette mask predicted by the DensePose [7] and we back-project the segemented mask from RGB to Depth space. We then dilate the depth mask to generate depth dilated mask. In parallel, We pre-process the depth image by filtering out the depth pixels which are greater then 3 Meter. We extract an intermediate depth silhouette by element wise product of dilated depth mask and filtered depth map. Finally, we fit a plane to the 3D points near the bottom of the depth image and remove the depth pixels, whose 3D point lie on and below the plane.

where $f_{fit} : \mathbb{R}^{224\times224\times3} \rightarrow \mathbb{R}^{85}$ is the mapping function, Θ is the learnable network parameter and D is the learnable per-vertex displacements applied to SMPL+D model.

3.3. Data generation

As mentioned earlier, we use two different supervision for training our network (i) Face barycentric correspondence (ii) Depth silhouette. The entire data generation is pipeline is described in the below sections.

Data acquisition: In order to obtain RGB-D scans of humans for training and testing, we use a Structure Sensor mounted on an iPad. The RGB stream is captured at a resolution of 1296 × 968 pixels and 30 frames per second; the depth stream is captured at a resolution of 640 × 480 pixels. For every actor, we record two action sequences of 3 Minutes (roughly around 1800 frames each). The first sequence is recorded with the actor standing in rough A-pose, and whereas we record the second sequence with actors performing more complex actions. We further post process and register the RGB with the depth image.

Face barycentric correspondence: As shown in Fig.2, we start by running DensePose [7] over the RGB image and saving the IUV map predicted by it. For every RGB image, DensePose predicts a bounding box of the human in the image and for every pixel inside the bounding box the DensePose uses Mask-RCNN [8] to regress the part index(I) of the body part and their U and V coordinates in the corresponding UV Map. We first sample the IUV's from the predicted bounding box and extract corresponding Face Barycentric coordinate on the SMPL model through an IUV to Face Barycentric coordinate transformation provided by the DensePose. We then take the depth map and back-project the pixels to 3D space and generate the 3D depth map. Then we generate an one to one correspondence set between Face barycentric correspondence and 3D depth map by camera intrinsic transformation to project the absolute UV value of the sampled IUV's to the 3D depth map and query the value of the 3D point in the depth map.

Depth silhouettes: The problem with using DensePose prediction to generate supervision data is misclassification of the pixels near the boundaries. Hence we propose depth silhouettes based supervision to go beyond DensePose. As shown in Fig.3, we start with generating depth mask by projecting the silhouette-mask predicted by the DensePose Mask-RCNN to depth image via intrinsic transformation. We heuristically determine that DensePose classifies pixels near the boundaries of the predicted human silhouette. We use the product of dilated depth mask and Z filtered depth image to generate an intermediate depth silhouette with an assumption that combination of dilating the depth mask and

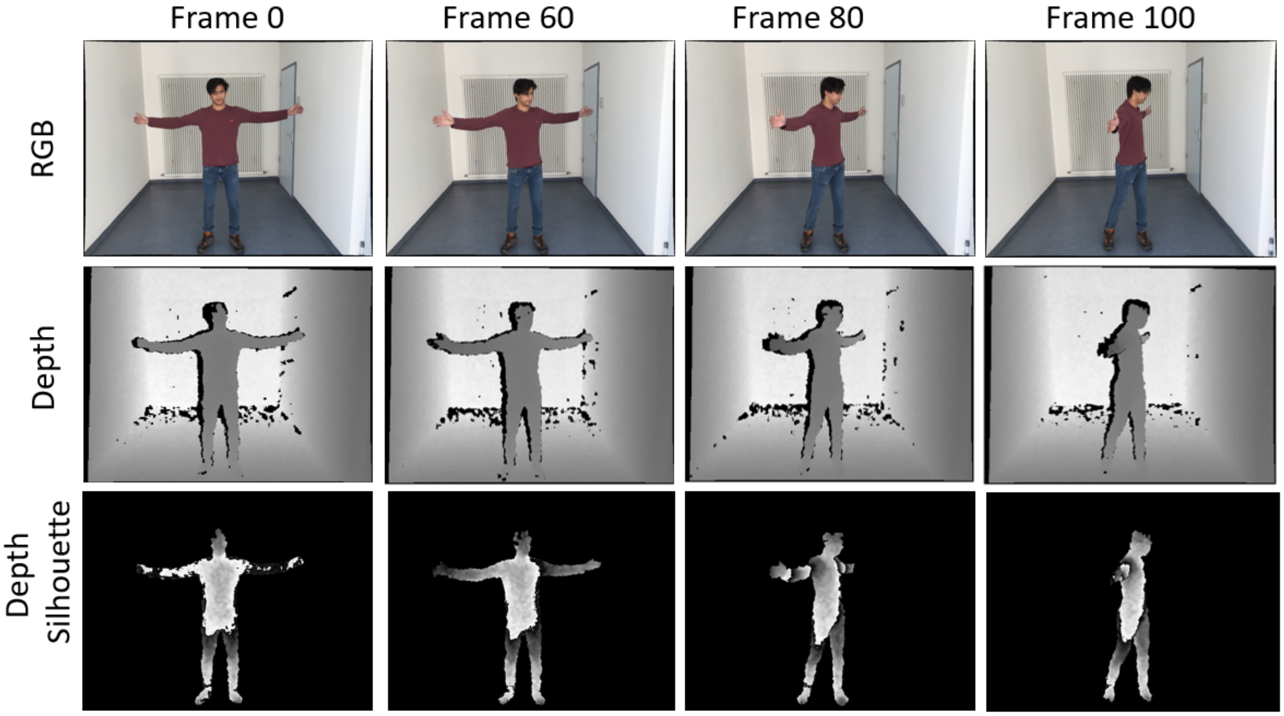

Figure 4. Sample Depth silhouettes

max filtering the depth map to remove background pixel will allow us to extract the true silhouette of the Human in depth space. In intermediate depth silhouette, we can see that depth silhouette overlaps with the pixels in the ground plane. In order to remove the ground pixels, we fit a plane to the 3D point near the ground plane and remove all the pixel that lie on and below the plane. We can see the generated depth silhouette in the Fig. 4.

3.4. Differentiable image formation

We use differentiable renderer to enable end-to-end self-supervised training in our method. We first render down depth map from SMPL+D mesh using the differentiable renderer and apply our silhouette loss on the rendered depth map as described in Section3.5. Our differentiable image formation model is expressed as

$$R_c(M(x)) = \pi(\phi(M(x))) \tag{4}$$

where $\phi(v) = R \cdot v + t$ is the rigid transformation from SMPL model space to camera space, defined by $R \in SO(3)$ and $t \in \mathbb{R}^3$, $\pi : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ is the full perspective camera model that maps from the camera space to screen space. Note, unlike the previous methods [2, 11, 16] we use full perspective camera model instead of weak perspective camera model, which enable us to learn scale-aware mapping from pixel space to SMPL+D space. For differentiable renderer, we modified the implementation of Neural 3D mesh render from Hiroharu Kato [12] to render rectangular images.

3.5. Loss functions

Let x = (θ, δ; Θ, β, D) represent the regressed parameters as well as learnable parameters of the network. Our entire loss function is described in the following expression.

$$E(x) = \lambda_{FBC} \cdot E_{FBC}(x) + \lambda_{sil} \cdot E_{sil}(x) + E_{reg}(x) \tag{5}$$

$$E_{reg}(x) = \lambda_{smo} \cdot E_{smo}(x) + \lambda_{pose} \cdot E_{pose}(x) \tag{6}$$

Face barycentric loss: We use Face Barycentric loss as supervision to fit the SMPL+D model to depth map, given a set of pair correspondence $\{FBC_i, v_i\}_i^N$ for Face Barycentric correspondence (Section3.3)

$$E_{FBC}(x) = \frac{1}{N}\sum_{i=1}^{N} ||\Psi(FBC_i, M(x)) - v_i||^2 \tag{7}$$

where Ψ is function to compute Face Barycentric value from the SMPL+D, $M(x)$ is the function to generate SMPL+D from the regressed parameters as mentioned in the Section3.1, v_i represent the corresponding 3D point from the depth map.

Depth silhouette loss: We use the depth silhouette loss to fit the SMPL+D model densely to the depth map. Given

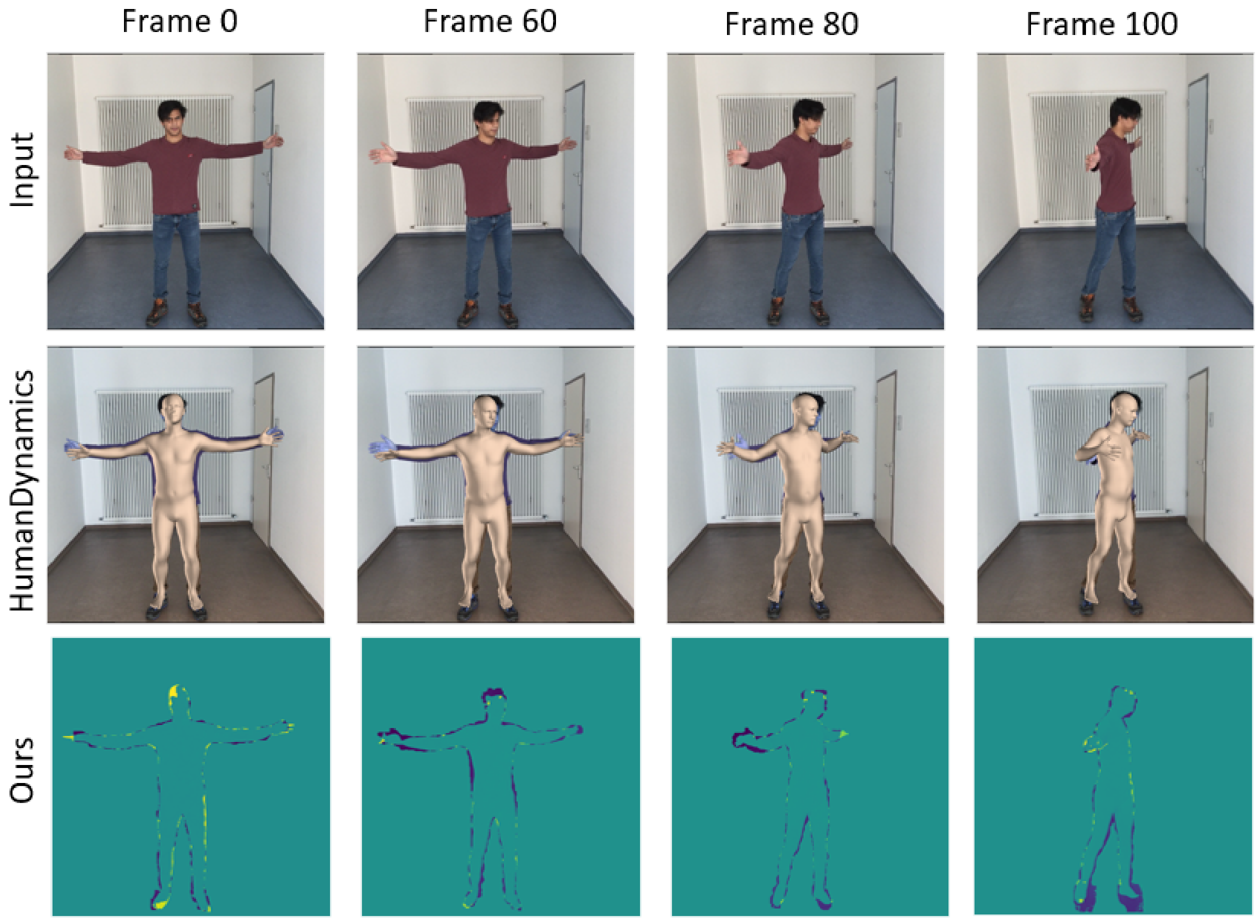

Figure 5. Comparison of our tracking results with HumanDynamics [11]

the set of $\{I_i, Z_i\}_i^N$, the depth silhouette loss is expressed as

$$E_{sil}(x) = \frac{1}{N_p}\sum_{i=1}^{N}||R_c(M(x)) - Z_i||^2 \qquad (8)$$

where R_c is the image formation function mentioned in the Section, Z_i is the depth silhouette generated as mentioned in Section3.3, N_p is the number of valid pixels in the depth silhouettes.

Geometry smoothness: We use the first-order geometry smoothness to enforce smooth deformation on top of SMPL model. The geometry smoothness can be expressed as

$$E_{smo}(x) = \sum_{i=1}^{N}\sum_{j\in h_i}||t_i(x) - t_j(x)||^2 \qquad (9)$$

where h_i the set of neighbouring vertex to the i-th vertex, t_i or t_j represent the vertex displacements from the SMPL+D mesh, N is the number of vertices in the SMPL model.

Invalid pose penalizer: We use V-Poser function similar to SMPL-X [17] to penalize the invalid poses and limit the regressed SMPL parameters within the valid human pose manifold. The VPoser is trained using a variational autoencoder that to learn a latent representation of human pose and it regularizes the distribution of the latent code to a normal distribution.

4. Experiments

In this section, we first describe our dataset, network architecture and training scheme. The following sections details the result of our experiments and how different components of our approach impact the performance.

4.1. Experimental setup

Dataset As we mentioned in earlier Section3.3, we first capture RGB-D frames from a subject and preprocess them. For every sequences, we record roughly around 1800 frames and sample 1200 frames without motion blur. We train our method on the first 1000 frames and evaluate on the later 200 frames.

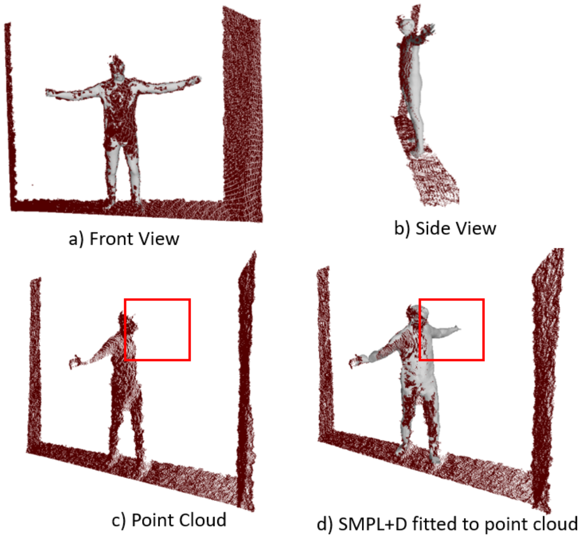

Figure 6. Evaluation of our model fitting in 3D space.

Network architecture In this section, we provide the details of the convolutional network architecture of our f_{fit} mapping function. Similar to the previous methods [2, 11], we use ResNet-18 [9] based encoder block and we modify the final fully connected layer to regress 85 parameters as detailed in Section3.2. In addition to the network parameters(Θ), we also have a learnable per-vertex displacments(D), which are initialized to zero at start of training. Like network parameters, the per-vertex displacements are optimized by propagating gradients from our loss functions.

Training scheme The proposed method, starting from our CNN block, fixed SMPL layer (Section3.1), renderer (Section3.5), is fully end-to-end differentiable. We train our network in an incremental scheme similar to [5, 2]. We experimented with various training schemes and empirically determined a scheme that produced the best results for us, where we first train our network with Face Barycentric correspondence, after 500 epochs we introduce Depth silhouettes loss and after next 500 epochs we add the per-vertex offset optimization. We have also empirically determined that training in an incremental scheme prevents the optimizer from converging to local minima.

4.2. Evaluation

We qualitatively compare the tracking result with current state-of-the-art pose estimation pipeline [11]. In Fig.5, we demonstrate our tracking results on subject S1 and note this particular sequence is not seen by the network during training. As we can see from the Fig.5, our network can better fitting the SMPL+D on comparison with HumanDynamics. Further, we use pixel accuracy and IoU(Intersection over Union) on the silhouettes to quantitatively evaluate our fitting on 2D space.We use grabcut [19] generate the ground truth silhouette used for evaluation. To be able to quantitatively compare our sequence, we reuse and adapt the code provided by [11, 7]. The quantitative comparison is shown in the Table 1 The results on the IoU and pixel accuracy are reported on the validation sequence of 600 frames with varying pose. our proposed method outperforms both DensePose and HumanDynamics in the our current setup without fast motions. We further qualitatively compare our fitting with point cloud to demonstrate our fitting in 3D Space and how our network learn geometry from point cloud. Our network captures the shape and pose of the subject in detail, which can seen in Fig.6.

Method	Pixel Accuracy	IoU
HumanDynamics	0.82	0.78
DensePose	0.85	0.73
ours	**0.92**	**0.89**
ours - only FBC	0.80	0.76
ours - without displacement	0.81	0.75
ours - only depth silhouette	-	-

Table 1. Comparison with the state-of-the-art approaches. Our method outperforms both DensePose and HumanDynamics by a significant margin. Note: Grabcut [19] is used to generate ground truth silhouettes for evaluation.

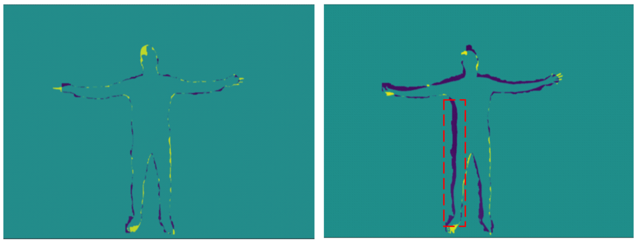

Figure 7. Evaluation of fitting between network trained using Face barycentic loss vs Face barycentic + Depth silhouette loss. As we can see the method trained using Face barycentric loss cannot fit the RGB image.

4.3. Ablation study

We evaluate different components our network to evaluate their effect on fitting. Since the Face barycentric correspondence is generated using DensePose predictions, our network trained only using Face Baryncetric correspondence cannot fully fit on the Silhouettes, which can be seen in Fig.7 and Table. 1. On the contrary, our method trained only using Depth Silhouette loss failed to converge, since optimizing global rigid pose of SMPL+D model to align it with depth silhouette is very hard problem. In particular, in our depth loss formulation we use mask to remove back-

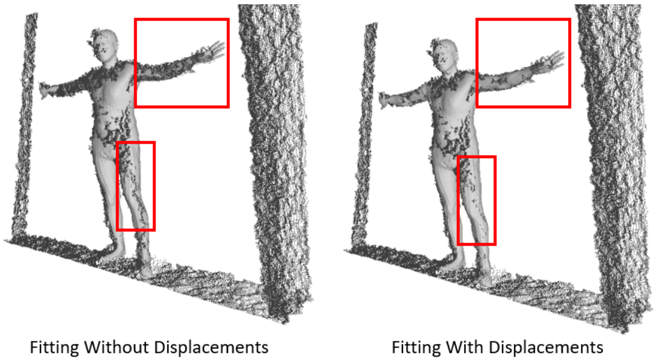

Figure 8. Evaluation of the method with and without per-vertex displacements.

ground pixels from the loss and compute loss over the foreground rendered SMPL+D pixels. Hence, a depth image with full of background pixels will have zero loss. Even though, removing the background masks seems to be the option to overcome this problem. However, in practice without this mask our loss value overshoots and the optimization diverges. Our entire method learns the mapping function from depth and 2D silhouettes, we need to learn per-vertex offset displacements on top of SMPL to better in both 2D and 3D space, which is evident from the Fig.8 and Table 1 (row: 5). Our method without per-vertex displacements is unable to capture the detailed shape of the subject in 3D space. In Fig.8, we can see that our method without per-vertex displacements couldn't fit to the 3D points that are close to surface of the SMPL model; whereas Our method with per-vertex displacements captures these small details via the displacements and fits the SMPL parameters better in 2D and 3D space.

4.4. Limitations

In our method, we depend on the depth silhouettes to overcome the errors introduced by DensePose [7] misclassifications. Sometimes, the depth silhouettes might be corrupted by the invalid depth data. As shown in the Fig. 9, we can see the depth silhouettes shown are corrupted near the head, this can cause our network to unnaturally deform the head of our SMPL+D model during fitting. As a future work, we could use RGB silhouettes to handle the invalid pixels in the depth silhouettes.

5. Conclusion

We have proposed RGB-D based self-supervised training of a deep neural network to learn human pose from RGB-D images, we have demonstrated that our method can predict pose of human from RGB images using the shape and per-vertex displacements during training. We also show that our method learns a scale-aware mapping between RGB pixels and SMPL+D space, which enable us to

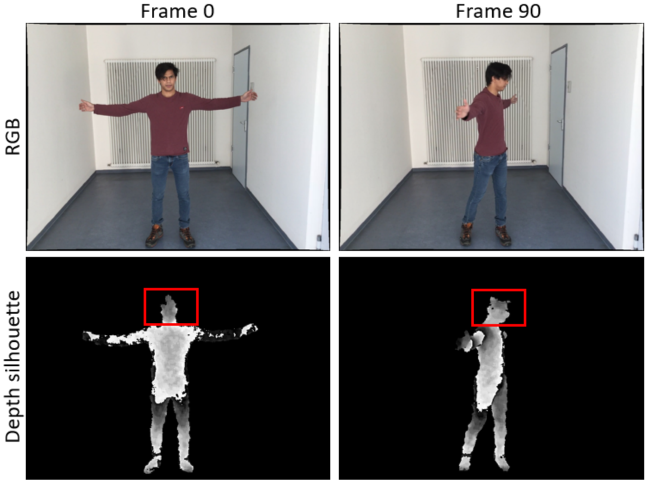

Figure 9. Impact of the invalid depth pixels on the depth silhouettes

use method for 3D reconstruction applications. We also introduced a method to generate quality 3D supervision data using DensePose [7] and depth. Through experiments, we establish that our depth silhouettes eliminate the errors introduced by the DensePose misclassifications and allow us to learn a detailed 3D model from RGB-D images. Further, our method can be easily extended to any RGB-D sequence without any annotation.

References

[1] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'04, page 882–888, USA, 2004. IEEE Computer Society.

[2] T. Alldieck, M. A. Magnor, B. L. Bhatnagar, C. Theobalt, and G. Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 1175–1186. Computer Vision Foundation / IEEE, 2019.

[3] F. Bogo, A. Kanazawa, C. Lassner, P. V. Gehler, J. Romero, and M. J. Black. Keep it SMPL: automatic estimation of 3d human pose and shape from a single image. *CoRR*, abs/1607.08128, 2016.

[4] Z. Cao, G. Hidalgo, T. Simon, S. Wei, and Y. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *CoRR*, abs/1812.08008, 2018.

[5] F. M. Castro, M. J. Marín-Jiménez, N. Guil, C. Schmid, and K. Alahari. End-to-end incremental learning. *CoRR*, abs/1807.09536, 2018.

[6] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu. Rmpe: Regional multi-person pose estimation. In *ICCV*, 2017.

[7] R. A. Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. *CoRR*, abs/1802.00434, 2018.

[8] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[10] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, page 559–568, New York, NY, USA, 2011. Association for Computing Machinery.

[11] A. Kanazawa, J. Y. Zhang, P. Felsen, and J. Malik. Learning 3d human dynamics from video. *CoRR*, abs/1812.01601, 2018.

[12] H. Kato, Y. Ushiku, and T. Harada. Neural 3d mesh renderer. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[13] C. Lassner, J. Romero, M. Kiefel, F. Bogo, M. J. Black, and P. V. Gehler. Unite the people: Closing the loop between 3d and 2d human representations. *CoRR*, abs/1701.02468, 2017.

[14] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015.

[15] R. A. Newcombe, D. Fox, and S. M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[16] M. Omran, C. Lassner, G. Pons-Moll, P. V. Gehler, and B. Schiele. Neural body fitting: Unifying deep learning and model-based human pose and shape estimation. *CoRR*, abs/1808.05942, 2018.

[17] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[18] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis. Learning to estimate 3d human pose and shape from a single color image. *CoRR*, abs/1805.04092, 2018.

[19] C. Rother, V. Kolmogorov, and A. Blake. "grabcut" – interactive foreground extraction using iterated graph cuts. *ACM TRANS. GRAPH*, pages 309–314, 2004.

[20] L. Sigal, A. Balan, and M. J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, page 1337–1344, Red Hook, NY, USA, 2007. Curran Associates Inc.

[21] V. Tan, I. Budvytis, and R. Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. In *BMVC*, 2017.

[22] M. Trumble, A. Gilbert, A. Hilton, and J. P. Collomosse. Deep autoencoder for combined human pose estimation and body model upscaling. *CoRR*, abs/1807.01511, 2018.

[23] H. F. Tung, H. Tung, E. Yumer, and K. Fragkiadaki. Self-supervised learning of motion capture. *CoRR*, abs/1712.01337, 2017.

[24] T. Whelan, M. Kaess, H. Johannsson, M. Fallon, J. J. Leonard, and J. Mcdonald. Real-time large-scale dense rgb-d slam with volumetric fusion. *Int. J. Rob. Res.*, 34(4–5):598–626, Apr. 2015.

[25] T. Yu, Z. Zheng, K. Guo, J. Zhao, Q. Dai, H. Li, G. Pons-Moll, and Y. Liu. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. In *The IEEE International Conference on Computer Vision and Pattern Recognition(CVPR)*. IEEE, June 2018.