NCHO: Unsupervised Learning for Neural 3D Composition of Humans and Objects (Supplemental Material)

Taeksoo Kim[1] Shunsuke Saito[2] Hanbyul Joo[1]
[1]Seoul National University [2]Meta Reality Labs
taeksu98@snu.ac.kr shunsuke.saito16@gmail.com hbjoo@snu.ac.kr

1. Implementation details

1.1. Network Architectures

Latent codes assigned to each scan, $\mathbf{z}_{th}$, $\mathbf{z}_{sh}$, and $\mathbf{z}_o$ are 64-dimensional. For $\mathbf{z}_o$, we use its first 5 bits to encode the object category via one-hot encoding and optimize only the last 59 bits during training. The generator G of the human module and the object module generates the $256 \times 256 \times 64$ feature image from a constant vector of size $256 \times 16 \times 16$ via 4 layers of (bilinear upsampler with a scale factor of 2, 2D convolution of kernel size 3 and stride 1, adaIN for conditioning the generator with the latent code $\mathbf{z}$, and leaky ReLU activations). The $256 \times 256 \times 64$ output feature image is split into one $256 \times 256 \times 32$ and two $256 \times 128 \times 32$ to form a tri-plane feature map. Note that the feature map is 128-dimensional along z-axis and 256-dimensional along other axes. The decoder for predicting the occupancy of the human module and the object module is a multi-layer perceptron having the intermediate neuron size of (256, 256, 256, 229, 1) with skip connection from the input features to the 4th layer and nonlinear activations of softplus with $\beta = 100$ except for the last layer that uses sigmoid. As an input, it takes the Cartesian coordinates in canonical space which are encoded using a positional encoding with 4 frequency components, and the 32-dimensional feature queried from the generated tri-plane. The decoder for predicting SDF of the human module has the same architecture as the decoder for predicting the occupancy, except that it has no activations for the last layer. The decoder for predicting the occupancy of the composition module has the same architecture as the decoders for predicting the occupancy of other modules. However, instead of taking in the feature from the generated tri-plane as an input, it takes in the intermediate latent feature vectors before the last layer of the decoders for predicting the occupancy of the human module and object module, which are 229-dimensional each.

Our deformation networks $D = (W, N)$ follow the architecture of the deformer of gDNA [3]. The skinning network W is a multi-layer perceptron having the intermediate neuron size of (128, 128, 128, 128, 24) with nonlinear activations of softplus with $\beta = 100$, except for the last layer that uses softmax in order to get normalized skinning weights. As an input, it takes the Cartesian coordinates in canonical space and the latent code $\mathbf{z} \in \mathbb{R}^{64}$ of the training sample. The warping network N is also a multi-layer perceptron having the intermediate neuron size of (128, 128, 128, 128, 3) with nonlinear activations of softplus. As an input, it takes the Cartesian coordinates in canonical space and the SMPL shape parameter $\beta \in \mathbb{R}^{10}$ of the training sample. The input Cartesian coordinates are passed to the last layer for the network to learn residual displacements.

1.2. Training Procedure

Our training consists of three stages. First, we train $\mathcal{M}_{th}$ and $\mathbf{z}_{th}$ with $\mathbf{S}_{th}$ with losses following [3, 4] and additional losses to train the SDF network. The total loss $\mathcal{L}_{M_{th}}$ is as follows:

$$\mathcal{L}_{M_{th}} = \mathcal{L}_{th} + \lambda_{bone}\mathcal{L}_{bone} + \lambda_{joint}\mathcal{L}_{joint} + \lambda_{warp}\mathcal{L}_{warp} + \lambda_{reg\_th}\mathcal{L}_{reg\_th} + \mathcal{L}_{sdf} + \mathcal{L}_{nml} + \mathcal{L}_{igr} + \mathcal{L}_{bbox}, \tag{1}$$

where $\lambda_{warp} = 10$ and $\lambda_{reg\_th} = 10^{-3}$. We set $\lambda_{bone} = 1$ and $\lambda_{joint} = 10$ only for the first epoch and 0 afterwards.

For the second stage, we train $\mathcal{M}_{sh}$ and $\mathbf{z}_{sh}$ with $\mathbf{S}_{sh}$ with the total loss $\mathcal{L}_{M_{sh}}$ being,

$$\mathcal{L}_{M_{sh}} = \mathcal{L}_{sh} + \lambda_{reg\_sh}\mathcal{L}_{reg\_sh}, \tag{2}$$

where $\lambda_{reg\_sh} = 10^{-3}$. As described in the main paper, since we initialize D_{sh} with the pre-trained D_{th}, additional guidance losses as in the first stage are not required. Note that since it is not our primary objective to model the detailed surface of the source human, we don't utilize the hybrid modeling of occupancy and SDF for $\mathcal{M}_{sh}$.

For the last stage, we train $\mathcal{M}_o$, $\mathcal{M}_{comp}$, $\mathbf{z}_{sh}$, and $\mathbf{z}_o$ with the pre-trained $\mathcal{M}_{th}$, $\mathcal{M}_{sh}$ and $\mathbf{z}_{th}$ frozen. As described in

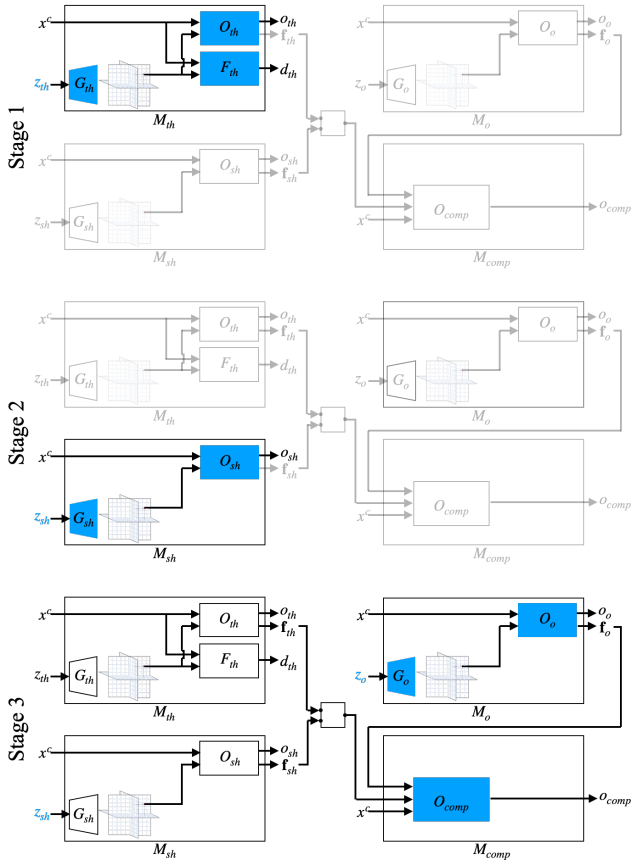

Figure 1. **Training Procedures.** Networks and latent codes that are optimized in each stage are highlighted.

the main paper, $\mathbf{z}_{sh}$ for the last stage are re-initialized as the mean of $\mathbf{z}_{sh}$ after the second stage. The total loss $\mathcal{L}$ is as follows:

$$\mathcal{L} = \mathcal{L}_{comp} + \mathcal{L}_o + \lambda_{fit}\mathcal{L}_{fit} + \lambda_{reg\_sh}\mathcal{L}_{reg\_sh} + \lambda_{reg\_o}\mathcal{L}_{reg\_o}, \quad (3)$$

where $\lambda_{fit} = 0.2$, $\lambda_{reg\_sh} = 50$, and $\lambda_{reg\_sh} = 10^{-3}$.

We train each stage with the Adam optimizer with a learning rate of 0.001 without decay. All stages are trained for 300 epochs. The training procedure for each stage is shown in Fig. 1.

1.3. Inference

We generate the composited canonical shapes of general people with objects by random sampling $\mathbf{z}_{th}$ and $\mathbf{z}_o$ from the Gaussian distribution fitted to each set of latent codes. We then extract meshes using o_{comp} with a resolution of 256^3. We finally repose the output mesh using the SMPL pose parameter with the learned skinning fields.

Method	Pred-to-Scan↓	Scan-to-Pred↓
gDNA	0.0184	0.0154
gDNA(w/ SDF loss terms)	**0.0171**	**0.0145**

Table 1. Fitting accuracy comparison between the original gDNA and gDNA trained with additional SDF loss terms.

2. Data

2.1. Acquisition

We collect 3D scans of the source human with and without objects using a system with synchronized and calibrated 8 Azure Kinects. We capture data 5FPS with the resolution of 2048×1536 for the RGB cameras, and 1024×1024 for the depth cameras. We perform image-based calibration using COLMAP [8] and adjust the optimized camera extrinsics to real-world scale based on the corresponding depth maps. We apply KinectFusion [7] with the code from the repository [1] to fuse the captured depth maps with the voxel resolution of 1.5mm. We reconstruct watertight meshes from the fused output using screened-poisson surface reconstruction [6] of depth 9. In order to obtain SMPL parameters for each captured scan, we use the multi-view extension of SMPLify [1] with the code from the repository [2]. For each scan, we render images from 18 viewpoints and detect 2D keypoints using OpenPose [2], and apply the multi-view extension of SMPLify to estimate SMPL parameters for each scan.

2.2. Data Statistics

We use 180 samples for $\mathbf{S}_{sh}$ and 342 samples for $\mathbf{S}_{sh+o}$. For $\mathbf{S}_{sh+o}$, we consider 4 categories of objects: 5 backpacks (77 samples in total), 6 outwear (94 samples), 8 scarves (89 samples), and 6 hats (82 samples). For running the quantitative evaluation focused on backpacks, we use another set with 300 samples of the source human with 5 backpacks, denoted as $\mathbf{S}_{sh+bp}$. To build a testing set for FID computation, we further capture 343 samples of 3 different unseen identities who wear unseen backpacks, denoted as $\mathbf{S}_{unseen+bp}$. We also use 526 samples of THuman2.0 [10] for $\mathbf{S}_{th}$.

3. Discussion

3.1. Geometry Modeling with SDF

Our primary goal is to enhance the quality of the 3D geometry rather than the rendering qualities, particularly targeting the tasks such as fitting scans of humans with objects or removing objects from scans.

As mentioned in the main paper, we model detailed geometry by jointly predicting SDF together with the occupancy

[1] https://github.com/andyzeng/tsdf-fusion-python
[2] https://github.com/ZhengZerong/MultiviewSMPLifyX

Figure 2. **Qualitative Comparison on Introducing SDF Network in the Human Module.** Top row: Random generated outputs when trained with occupancy only. Bottom row: Random generated outputs when trained with the hybrid modeling of occupancy and SDF. Additionally predicting the SDF improves the details of random generated outputs.

fields. We find that directly replacing the occupancy with the SDF leads to failures in canonicalization. Among the set of correspondences resulting from multiple initials for the root finding algorithm, previous work that uses occupancy representation [3, 4] determines the final correspondence by choosing the point with the highest estimated occupancy. However, in the case of the SDF representation, we empirically find out that choosing the point by only utilizing the estimated SDF leads to poor canonicalization. Moreover, using a single initial by linearly combining the skinning weights of the nearest neighbor on the fitted SMPL mesh and the inverse bone transformations as in [5, 9] also leads to incorrect canonicalization. Hence, we utilize a hybrid modeling of occupancy and SDF by leveraging the advantage of each representation. While directly supervising SDF on the surface normals, we select final correspondences and train the deformation networks using occupancy. For stable training, it is crucial to disable the backpropagation of gradients from the SDF head to the deformation networks and let only the occupancy head supervise them.

As presented in the main paper, utilizing both occupancy and SDF results in better 3D geometry reconstruction. We further verify the benefit of the additional SDF loss terms for the fitting task by fitting unseen scans without objects using the original gDNA and gDNA trained with additional SDF loss terms. As demonstrated in Tab. 1, gDNA trained with additional SDF loss terms reports better fitting accuracy than the original. We also present random generated samples trained with each method in Fig. 2.

(a) gDNA | (b) (a) with SDF loss terms | (c) (a) with adversarial learning

Figure 3. **Qualitative Comparison between hybrid modeling of occupancy and SDF and Normal Adversarial Loss.** (a): Output of original gDNA. (b): Output of gDNA trained with additional SDF loss terms. (c): Output of gDNA with detailed normals via adversarial learning.

(a) gDNA trained without objects | (b) composition output | (c) (b) with adversarial learning

Figure 4. **Qualitative Comparison on Introducing Normal Adversarial Loss to Composition Output.**(a): Output of the human module. (b): Output of the composition module. (c): Output of the composition module with detailed normals via adversarial learning. Normal adversarial learning is detrimental in the current setup.

3.2. Normal Adversarial Loss

gDNA [3] models high-frequency details of their outputs via 2D adversarial learning of normals in canonical space. We show the qualitative comparison between the outputs with hybrid modeling of occupancy and SDF and the outputs with normal adversarial loss in Fig. 3 (b) and (c). We also apply the adversarial learning to our composition output as shown in Fig. 4 (b) and (c). Although normal adversarial learning is able to model high-frequency details to some extent, it fails to model details consistently and undermines the overall quality in the current setup, especially in the facial areas, possibly due to the quality of the training data.

Figure 5. **Example Images of the First User Study.** Subjects are asked to choose the sample with a more authentic shape between top and bottom.

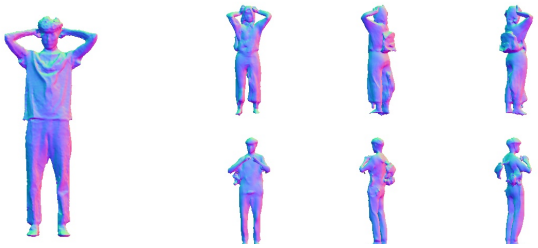

Figure 6. **Example Images of the Second User Study.** Subjects are asked to choose the sample that does not resemble the shape of the source human shown on the left, between top and bottom.

4. Quantitative Evaluation Details

4.1. FID Computation

We compute FID score using the code from the repository of [3]. For the test set, we render 2D normal maps in resolution 256^2 of 343 samples in $\mathbf{S}_{unseen+bp}$ from 18 viewpoints, resulting in 6174 images. For each method, we generate 200 samples in random body sizes and poses of the $\mathbf{S}_{unseen+bp}$ and similarly render 2D normal maps in resolution 256^2 from 18 viewpoints, resulting in 3600 images.

4.2. User Preference Study

We perform two user preference studies (A/B test) via CloudResearch Connect. The first study aims to validate the generation quality of our method over all baselines, and the second study aims to validate the generation diversity of our method over *'gDNA (w/ object)'*.

For the first user study, we show a sample generated with our method along with another sample generated with one of the baseline methods in random order. For each sample, we render 2D normal maps in resolution 256^2 from 3 viewpoints. We ask 50 subjects to answer 5 A/B pairs per baseline by choosing the preferred sample with a more authentic shape. An example of a question is presented in Fig. 5.

For the second user study, we only compare our method with the baseline, *'gDNA (w/ object)'*, with a different protocol. In this study, we similarly render the normal maps from 3 viewpoints from each method and additionally show an image of the source human along with the A/B pairs. Then, we request the observers to choose the sample that looks more different from the source human. The test is intended to see whether the methods can produce diverse human identities with objects, sufficiently different from the source human's appearance. An example of a question is presented in Fig. 6. Similar to the first study, we ask 50 subjects to answer 5 A/B pairs by choosing the sample that better satisfies the question.

4.3. Fitting Comparison

For fitting our model to unseen scans with objects, we follow the fitting process of gDNA [3]. During fitting, we optimize the latent code for the human part, $\mathbf{z}_h$, the latent code for the object part, $\mathbf{z}_o$, and the SMPL shape parameter β with other network frozen. We use $\mathcal{M}_{th}$ for the human module. We initialize $\mathbf{z}_h$ and $\mathbf{z}_o$ each with 8 randomly sampled codes from the Gaussian distribution fitted to each set of latent codes. β is initialized with the obtained SMPL shape parameter during our data acquisition process. The loss $\mathcal{L}_{fitting}$ used for fitting raw scans is as follows:

$$\mathcal{L}_{fitting} = \mathcal{L}_{comp} + \lambda_{reg\_h}\mathcal{L}_{reg\_h} + \lambda_{reg\_o}\mathcal{L}_{reg\_o} \quad (4)$$

$$\mathcal{L}_{comp} = BCE(o_{comp}, o_{unseen}) \quad (5)$$

$$\mathcal{L}_{reg\_h} = \|\mathbf{z}_h\| \quad (6)$$

$$\mathcal{L}_{reg\_o} = \|\mathbf{z}_o\|, \quad (7)$$

where $\lambda_{reg\_h} = 50$ and $\lambda_{reg\_o} = 50$. We optimize for 500 iterations using the Adam optimizer with a learning rate of 0.01 without any weight decay or learning rate decay. Of 8 fitted outputs, the one with the minimum bi-directional Chamfer distance to the target scan is chosen as the final output.

5. Addtional Qualitative Results

Please refer to the supplementary video for additional qualitative results on individual control of the human and object modules, latent code interpolation, composition of multiple objects, and animated results. The video is available at https://taeksuu.github.io/ncho.

[3] https://github.com/mseitzer/pytorch-fid

References

[1] 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. 2

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 2

[3] Xu Chen, Tianjian Jiang, Jie Song, Jinlong Yang, Michael J Black, Andreas Geiger, and Otmar Hilliges. gdna: Towards generative detailed neural avatars. In *CVPR*, 2022. 1, 3, 4

[4] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *ICCV*, 2021. 1, 3

[5] Boyi Jiang, Yang Hong, Hujun Bao, and Juyong Zhang. Selfrecon: Self reconstruction your digital avatar from monocular video. In *CVPR*, 2022. 3

[6] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *TOG*, 2013. 2

[7] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, 2011. 2

[8] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 2

[9] Shaofei Wang, Katja Schwarz, Andreas Geiger, and Siyu Tang. Arah: Animatable volume rendering of articulated human sdfs. In *ECCV*, 2022. 3

[10] Tao Yu, Zerong Zheng, Kaiwen Guo, Pengpeng Liu, Qionghai Dai, and Yebin Liu. Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In *CVPR*, 2021. 2