# DeepMultiCap: Performance Capture of Multiple Characters Using Sparse Multiview Cameras Supplementary Material

## 1. Texture Inference

Our method can be easily extended to predict texture of the reconstructed geometry models. Similar to PIFu [4], we consider texture as a 3D RGB vector function defined within a certain band near the surface points. In our implementation, we consider texture inference as the next level of geometry fine level framework, and we feed geometry embeddings concatenated with texture features to a MLP layer:

$$F^C(\boldsymbol{X}) = g^C(T^C(\Phi_m^C(\boldsymbol{x}, \boldsymbol{I}), \Omega^H(\boldsymbol{X}))) \quad (1)$$

where the color level self-attention module $T^C$ merges the multi-view color information from multi-vew image features $\Phi_m^C(\boldsymbol{x}, \boldsymbol{I})$ concatenated with fine level 3D embeddings $\Omega^H$, and $g^C$ predicts the RGB vector conditioned with the meta feature extracted by $T^C$.

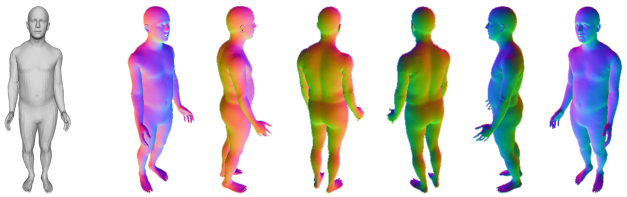
## 2. Implementation Details

**Loss Function** For multi-view inputs, similar to PIFu [4], the loss function of the geometry inference is designed as:

$$\mathcal{L}_m = \frac{1}{N} \sum_{i=0}^{N} |F^T(\boldsymbol{X}) - F^*(\boldsymbol{X})|^2 \quad (2)$$

where $N$ is the number of query points, $F^T$ denotes the predicted occupancy probability and $F^*(\boldsymbol{X}) \in \{0, 1\}$ is the ground truth value where 0 represents the points inside the surface while 1 for outside points. Different from PIFuHD [5], we use L2 loss instead of the Binary Cross Entropy (BCE) loss, since we find the BCE loss can lead to too sharp results, which can be unstable under the real world setting and even cause artifacts. For color inference, we draw the same principle while the difference is that now $F^T$ in Eqn. 2 predict a 3D RGB vector instead of a scalar, and the $F^*$ denotes the ground truth color value.

During training, we first train the coarse network. After that, for memory efficiency, we fix the coarse level and train the fine level. Similarly, the color network is trained with the geometry network fixed.



(a) SMPL (b) Rendered global SMPL normal maps

Figure 1: Examples of rendered SMPL global normal maps, whose three channels are the normal vector under SMPL model coordinate system.

**Network Architecture** Our method builds on a coarse to fine level attention-aware framework combined with SMPL models. To merge geometry information from different observations, inspired by [6], we design two self-attention layers with $n_{head} = 8$ multi-head module and 256 embedding size, where the first feature in the output sequence is used as the meta feature representing the global spatial information.

In the coarse level, following PIFu [4], we design the image encoder based on a 4 stack hourglass architecture [2], which results in 256-D low resolution feature maps with size of $128 \times 128$. A 3D convolution network with two layers and three residual blocks is used to extract 3D semantic features of $128^3$ volumetric representation of SMPL, where the output features have 64 channels with size of $32^3$. The multi-view image features are concatenated with SMPL embeddings and fed to the self-attention module. Finally, a multi-layer perceptron is trained to fit the implicit function, which takes in the merged 320-D feature as well as the 3D coordinate of the query points, and thus the number of neurons is (323, 1024, 512, 256, 128, 1).

In the fine level, a pix2pixHD [7] network is used to predict frontal image normal maps as introduced in PIFuHD. We further design the normal encoder by using 1 stack hourglass and removing the average pooling in the intermediate layers to obtain a high resolution feature map with $512 \times 512$ size. The number of channels in the output normal features is adjusted to 32 for memory efficiency. Therefore, the first layer in the fine level MLP has 355 neurons with the 355-D meta feature as input, which is merged by the fine level

self-attention module. A highly detailed 3D human can be finally reconstructed through the two level framework.

**Experimental Setup** To train the network, we collect 1700 3D human models from Twindom[1] and THuman2.0 [8]. $512 \times 512$ images and normal maps are rendered through perspective cameras from every other degree in yaw axis with random evaluation. The renderer is based on taichi [1], which enables us to perform rendering on headless machines equipped with GPUs.

Given the rendered images, we estimate 3D skeleton for each subject through a 4D association algorithm proposed in [9] and fit SMPL-X [3] with 25 pose keypoints, 70 face keypoints and 21 hand keypoints. The normal of SMPL surface under its canonical model coordinate system is calculate to render global normal maps as introduced in the main paper. Figure 1 provide examples of the global normal maps.

In order to train the network to handle occlusions in real world scenes, we further argument the training data by masking out random regions with various shapes and edges. The contours of blocked parts are formed by adding basic geometric figures including cube, spheroid with corrupted edges. To simulate multi-person scenes, we project other persons to the masks, where non-occlusion to heavy occlusion scenes can be generated. Figure 2 shows examples from our training dataset.

Besides, to help the network aware of visible details and leverage SMPL information for robust reconstruction, we use a sampling method based on the visiblity of points. The input points during training are sampled from Gaussian distribution centered by surface points with standard deviation $\boldsymbol{\sigma}$ as introduced in [4]. We further choose a small standard deviation $\boldsymbol{\sigma}_0$ for visible points to guide the network to learn fine-grained geometry details, while a larger $\boldsymbol{\sigma}_1$ for invisible points to avoid unreasonable predictions, which we find contributes to the improvement of performance under occluded scenes. In practice, we set $\sigma_0$ as 0.02 and $\sigma_1$ as 0.005.

In spirit of maintaining the same setup for training and inference, during reconstruction, we firstly normalize SMPL to a unit cube, and transform the query points to the SMPL model coordinate system to predict the occupancy field. The strategy ensures the consistency between training and testing, preventing the instability of reconstruction brought by the significant difference among real world coordinate space and training virtual environment.

## 3. MultiHuman Dataset

As introduced in the main paper, to better evaluate multi-person performance capture systems like ours, we collect a high resolution 3D human model dataset, MultiHuman, which contains 150 multi-person interacting scenes (including both natural and close interactions). In each scene, the number of person is within the range from 1 to 3, where each consists of about 300,000 triangles with photo-realistic texture. According to the level of occlusions and elements of interactions in the scenes, we divide the dataset into several categories for a detailed evaluation, i.e., single person scenes, occluded single person scenes, two natural interactive person scenes, two close interactive person scenes, and three person scenes. Figure 4 offers more examples of our dataset. To fill in the blank of available multi-person dataset of the community, we will make the dataset public, which will surely benefit the development of future algorithms.

[1] https://web.twindom.com/

(a) (b) (c)

Figure 2: Data argumentation by adding basic geometric figures with corrupted edges (a), and other persons to simulate multi-person scenes (b) and (c).

Figure 3: Texture reconstruction on real world videos. Our method is able to reconstruct multi-perosn texture from multi-view images. More results will be shown in our supplemental video.

## 4. Additional Results

**Ablation Study** Our method achieves the state-of-the-art performance by leveraging a spatial attention module to merge multi-view features and utilizing human pose and shape prior SMPL to compensate for the missing information due to occlusions in multi-person scenes. We further design a SMPL global normal map to help the attention module to identify the view orientation and better capture the details, which is similar to the position encoding method introduced in [6]. In the main paper we evaluate how the attention module and the SMPL information effect the quan-


Figure 4: Examples of MultiHuman dataset. Our dataset consists of high quality 3D human models with photo-realistic texture. According to the occlusion level and number of persons in the scene, we divide the dataset into 5 categories, including single person scenes, occluded single person scenes (by various objects), two natural interactive single person scenes, two close interactive person scenes, and three person scenes (from top to bottom).


Figure 5: Qualitative results of ablation study on MultiHuman dataset. We evaluate the performance of (e) our method and the alternative approaches including (b) ours without SMPL (which is equal to PIFuHD [5] combined with the attention module), (c) ours without the attention module and (d) ours without the designed SMPL global normal maps.

titative accuracy of prediction. Figure 5 offers qualitative examples to illustrate how our method benefits from the design. Without SMPL information (Ours w/o SMPL), reconstruction results can be fragmental due to occlusions in multi-person scenes. SMPL serves as a 3D proxy to help the network generate robust results. However, without the self-attention module to effectively merge information from multi-view observations, the approach can lead to artifacts when reconstructing multi-person, which is clearly demonstrated in the three person scene in Figure 5. Besides, our method without the guidance of SMPL global normal maps (Ours w/o SN) will have less detailed results, indicating that the SMPL normal maps further help the attention module to capture fine grained geometry features and reconstruct high-fidelity 3D humans.

**Texture Inference** We provide examples of the texture reconstruction results on real world scenarios in Figure 3.

Our method is able to reconstruct vivid multi-human texture from multi-view scenes.

For more results, please refer to our supplementary video.

## References


[1] Yuanming Hu, Tzu-Mao Li, Luke Anderson, Jonathan Ragan-Kelley, and Frédo Durand. Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)*, 38(6):201, 2019. 2

[2] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pages 483–499. Springer, 2016. 1

[3] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, pages 10975–10985, 2019. 2

[4] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *ICCV*, October 2019. 1, 2

[5] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *CVPR*, 2020. 1, 4

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 1, 2

[7] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, pages 8798–8807, 2018. 1

[8] 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*, pages 5746–5756, 2021. 2

[9] Yuxiang Zhang, Liang An, Tao Yu, Xiu Li, Kun Li, and Yebin Liu. 4d association graph for realtime multi-person motion capture using multiple video cameras. In *CVPR*, pages 1324–1333, 2020. 2