A. Experiment Detail

A.1. Comparison over ICON

The inconsistency between Table. 2 and Fig. 7 arises from the methodology (ICON [8]: model-based vs. Ours: model-free). When evaluating Table. 2, we used the GT SMPL-X skin body model provided by AGORA [4] for ICON, which shares the initial geometry of the GT clothed human dataset (Fig. I (a)) and is beneficial for the distance metrics: P2S and Chamfer. Since ICON receives the signed distance from SMPL-X skin body to query point as input, ICON results using GT SMPL-X show good performance in distance metrics such as P2S and Chamfer (Fig. I (b)). In contrast, our method predicts a depth from a camera origin, and leads to a depth ambiguity (Fig. I (c)). Due to this, although the mesh with good details is inferred, the distance metrics can be worsen than ICON using a strong geometry prior.

Here, we demonstrate another merit on loose cloth cases of our model-free method. In Fig. II and Table. I, our method consistently shows the high-quality reconstructions while ICON suffers from the loss of details in the skirt or hair.

A.2. Comparison with other datasets

We perform an additional experiment to validate the effectiveness of our dataset. Table. II shows the quantitative results on RenderPeople(RP) w.r.t. 3 training scenarios: (1) Ours only, (2) RP only, and (3) Using both of them. As expected, our dataset contributes to the huge improvement in the network training.

<table>
<thead>
<tr>
<th>Training</th>
<th>P2S↓</th>
<th>Chamfer↓</th>
<th>Normal↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>2K2K</td>
<td>1.24</td>
<td>1.35</td>
<td>3.89</td>
</tr>
<tr>
<td>RenderPeople</td>
<td>1.12</td>
<td>0.92</td>
<td>3.64</td>
</tr>
<tr>
<td>2K2K+RenderPeople</td>
<td>0.58</td>
<td>0.63</td>
<td>3.23</td>
</tr>
</tbody>
</table>

Table II. Evaluation on RenderPeople test split.
A.4. Comparison of SOTA normal prediction

We compare our normal prediction method with those of the previous works, PIFuHD [7] and HDNet [1], in Fig. III. While other results show artifacts or blurs, we infer the high-quality normal.

![Comparison of normal prediction](image)

Figure III. Comparison of the normal prediction.

B. Method Detail

B.1. Body part extraction detail

We compute the part-wise transformation matrix $M_i$ with human joints by using the linear least squares (LLS). The goal of this process to transform the input image into a target body part. After that, we crop the transformed image to a predefined size. The similarity transformation matrix is computed based on the target position $(x', y')$ and $i$th part joint position $j_i(x, y) \in J$,

$$
\begin{bmatrix}
  x' \\
  y' \\
  1
\end{bmatrix} =
\begin{bmatrix}
  a & -b & c \\
  b & a & d \\
  0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix}
$$

where $a$, $b$, $c$, and $d$ are an element of the similarity matrix.

To solve the above equation, we need at least two joints because the degree of freedom of the similarity matrix is 4. After arranging the above matrix for elements, the equation for each point is stacked in row as follows,

$$
\begin{bmatrix}
  x_1 & -y_1 & 1 & 0 \\
  y_1 & x_1 & 0 & 1 \\
  \vdots & \vdots & \vdots & \vdots \\
  x_N & -y_N & 1 & 0 \\
  y_N & x_N & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  a \\
  b \\
  c \\
  d
\end{bmatrix} =
\begin{bmatrix}
  x'_1 \\
  y'_1 \\
  \vdots \\
  x'_N \\
  y'_N
\end{bmatrix},
$$

where $N$ is the number of joints that we use to compute the transformation matrix $M$.

For detail, the joints $j_i(x, y) \in J$ and the pre-defined size for cropping are shown in the Table. III. Each joint position is a value that maps the upper left to $[-1, -1]$ and the lower right to $[1, 1]$. The detail process of transforming and cropping the image is as shown in Fig. IV. This process has the advantage of requiring few computation cost.

<table>
<thead>
<tr>
<th># of Keypoint</th>
<th>Keypoint $j_i(x, y)$</th>
<th>Crop size $s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head</td>
<td>2</td>
<td>[-1/12, 1/12, 1/12, 1/12]</td>
</tr>
<tr>
<td>Body</td>
<td>4</td>
<td>(-1/9, -1/6, -1/12, 1/12, 1/12, 1/12)</td>
</tr>
<tr>
<td>Upper arms</td>
<td>2</td>
<td>(0, 1/36, 0, 1/6)</td>
</tr>
<tr>
<td>Lower arms</td>
<td>3</td>
<td>(0, -1/36, 0, 0, 0, 1/18)</td>
</tr>
<tr>
<td>Left lower legs</td>
<td>2</td>
<td>(-3/128, -9/64, -3/128, 9/64)</td>
</tr>
<tr>
<td>Right lower legs</td>
<td>2</td>
<td>(-3/128, 9/64, 3/128, 9/64)</td>
</tr>
<tr>
<td>Feet</td>
<td>4</td>
<td>(0, 1/24, 0, 1/24, 0, -1/24)</td>
</tr>
</tbody>
</table>

Table III. In this work, we set five parts with twelve predefined positions and sizes; Head: two keypoints from left/right ears. Body: four keypoints from left/right shoulders and left/right heaps. Arms: two keypoints from a shoulder and an elbow, three keypoints from an elbow, an wrist, and a finger. Legs: two keypoints from a hip and a knee, two keypoints from a knee and an ankle. Feet: four keypoints from an ankle, a sole, and two toes.

![Body part extraction process](image)

Figure IV. Body part extraction from input image. Through the part-wise transformation matrix $M$ calculated using the input human joint $j(x, y)$, we transform the target part so that it is centered on the image. After that, we crop it to a fixed size $s$ and extract the part image.

B.2. Network description

In this section, we provide detailed descriptions of our network in Tables IV and V where we indicate the input in blue and the output in red. In the INPUT column, element-wise sum denotes ‘+’, and ⊕ denotes channel concatenation operator.

![Network description](image)

Table IV. Details of a high-resolution depth network. We estimate high-resolution front and back depth maps in a three-step cascade structure. In each cascade block, low resolution depth maps are up-sampled to higher resolution depth maps with a corresponding normal feature map.
Table V. Details of dual-encoder AU-Net. The first encoder, indexed by 1 to 9, takes a low-resolution image as input. The second encoder, indexed by 10 to 18, takes predicted low-resolution front and back normal maps as input. Finally, a decoder, indexed by 19 to 39, fuses features of the two encoders and predicts front and back depth maps.

### B.3. Coarse-to-fine vs. Part-wise for normal prediction

![GT](a) \( \overline{C2F} \) (c) Ours

Figure V. Comparison with coarse-to-fine method.

We adopted a coarse-to-fine normal refinement module of DeepHuman [11] after changing the resolution of the input to 2K in Fig. V. The performance was worse (0.890, 47GB in batch-size 1) than ours (0.679, 39GB in batch-size 3) in terms of normal error and memory requirement. The part-wise model is a practical solution for 2K or even larger resolution images because it does not extract global image features for normal map prediction, while the coarse-to-fine approach requires additional convolution layers to enlarge receptive fields for high resolution images.

### B.4. Poisson reconstruction

For fair comparison, we first convert the predicted depth map into the depth point cloud. We then overlap the point-cloud (Black) and the mesh (Gray) generated by a screened Poisson surface construction [2]. Fig. VI shows that they are completely overlapped, which means our high-resolution reconstruction does not cause a blur artifact.

### B.5. Gaussian blending

Fig. VII compares normal maps merged by averaging and Gaussian blending. Averaging approach shows streaking artifacts at the boundary whereas ours does not.

![Image](a) Face (b) Outside (c) Side (d) Inside

Figure VI. Comparison of pointcloud and mesh using the Poisson.

![Image](a) Occlusion (b) Ambiguity of boundary

Figure VII. Comparison of averaging and Gaussian blending.

### B.6. Failure case

Fig. VIII shows some failure cases. 3D recovery using only the front and back view depth maps shows that it is vulnerable to occlusions caused by objects or human body parts. In Fig. VIII (a), the correct shape could not be created because the spatial information between the arm and the torso could not be known only from the front and back depths. In addition, the network has difficulty in distinguishing it from garment texture and creating a clear depth boundary. Fig. VIII (b) shows an unnatural appearance as the boundary between the holding cup and the human body is smooth. We expect that the limitations can be improved by using a skinned human model [3].

![Image](a) Occlusion (b) Ambiguity of boundary

Figure VIII. Failure cases caused by occlusions and boundary ambiguities.
C. Dataset Detail

C.1. Feasibility of rendering at 2K resolution

We took 6,000 × 4,000 resolution images using 80 multi-view DSLR cameras. Sharing whole original images requires 1,850TB. Instead, since each body model in our dataset has 1M vertices and 2M faces, it is enough to be synthesized into 2K images as shown in Fig. IX.

Figure IX. Rendered multi-view images example.

C.2. Multi-view dataset capture

Thanks to the high-quality of our 3D models, we can render images from arbitrary viewpoints, whose example of an original and rendered images at 2K res. in Fig. IX. To acquire synchronized images, we used a VILTROX VL-500RT controller to generate trigger signals. Then, an Arduino board receives the signals and broadcasts them to the connected devices including the strobe lights and the DSLR cameras.

C.3. UV unwrapping

With our dataset, we are able to implement UV unwrapping modules for DensePose and SMPL template models (see Fig. X). Note that the texture map can be noisy if the model wears loose clothes (<5%).

Figure X. UV texture maps for DensePose and SMPL.

C.4. Dataset appearance example

Fig. XI shows appearance, mesh and normal examples of our dataset. Our dataset contains subjects of various genders, ages, objects, poses, and cloths with high-resolution scans.

D. More Results

D.1. In-the-wild qualitative results

Fig. XII, XIII shows the qualitative result made from in-the-wild internet photos. Although there is noise taken with a camera, our result forms a detailed human body shape compared to other methods [6, 7, 10]. Additionally, we reconstruct a sequence of images taken with a DSLR and make it as a video.

D.2. Dataset qualitative results

Fig. XIV, XV shows the qualitative comparison results of RenderPeople [5] and THuman2.0 [9] dataset. Compared to other methods [6, 7, 10], we reconstruct in a more clean and detailed surface when expressing the shape of a face, cloth, etc.
Figure XI. Example appearance, mesh and normal of our dataset. Each subject has high-resolution geometry features.
Figure XII. Qualitative comparison of in-the-wild images with other methods.
Figure XIII. Qualitative comparison of in-the-wild images with other methods.
Figure XIV. Qualitative comparisons of reconstructed results on the RenderPeople dataset.
Figure XV. Qualitative comparisons of reconstructed results on the THuman2.0 dataset.
References


