Model-based 3D Hand Reconstruction via Self-Supervised Learning (Supplementary Material)

Yujin Chen[1*] Zhigang Tu[1†] Di Kang[2] Linchao Bao[2]
Ying Zhang[3] Xuefei Zhe[2] Ruizhi Chen[1] Junsong Yuan[4]
[1]Wuhan University [2]Tencent AI Lab [3]Tencent [4]State University of New York at Buffalo
{yujin.chen, tuzhigang, ruizhi.chen}@whu.edu.cn {di.kang, zhexuefei}@outlook.com
linchaobao@gmail.com yinggzhang@tencent.com jsyuan@buffalo.edu

This is the appendix of the main text. We first provide detailed information about the proposed statistical regularization terms (Section 1), implementation and architecture (Section 2). Then we compare the results in camera coordinates with a fully-supervised setting in Section 3. Finally, more visualization results are presented in Section 5.

1. Statistical Regularization

We introduce three regularization terms, including the texture regularization, the scale regularization, and the skeleton regularization (Section 3.4.1 in the main text), to make the output 3D hands more reasonable. The details are as follows.

Texture Regularization. Since the skin color of hands typically is uniform, we propose a texture regularization term E_C to penalize outlier RGB values, where $f^C(c)$ is used to compute per-vertex color loss:

$$E_C = \frac{con_{sum}}{n} \sum_{i=1}^{n} f^C(c_i) \tag{1}$$

$$f^C(c) = \begin{cases} 0, & \text{if } \bar{c} - 2\sigma_c < c < \bar{c} + 2\sigma_c, \\ \parallel c - \bar{c} \parallel_2^2, & \text{else,} \end{cases} \tag{2}$$

Here, $\bar{c} \in \mathbb{R}^3$ is the average RGB of all vertices and $\sigma_c \in \mathbb{R}^3$ is the standard deviation for three color channels.

Scale Regularization. The scale regularization term is used to constrain the length of the hand bone to provide a reference for the scale uncertainty in this monocular 3D reconstruction task. For each dataset, we define an average bone length $\bar{l}$ of the proximal phalanx of the middle finger (the bone $\overrightarrow{9\underline{1}0}$ that between the 9th joint and the 10th joint in Fig. 3A of the main text). The scale regularization term is defined as $E_s = \parallel l - \bar{l} \parallel^2$ to encourage the length l of the estimated hand model's proximal phalanx of the middle finger to be close to the average length $\bar{l} \in \mathbb{R}^1$. We empirically set $\bar{l} = 2.82cm$.

Skeleton Regularization. The skeleton regularization term is used to penalize invalid hand pose. Instead of using a regularization on pose parameters θ to make the pose to be close to the average pose [1] (we call it the average pose prior), we think that feasible poses at different distances from the average pose should be treated equally without any penalty. We therefore define the feasible range $[min_i, max_i]$ (Table 1) for each rotation angle a_i (as shown in Fig. 3B of the main text) and then penalize those who exceed the threshold:

$$E_J = \frac{1}{15} \sum_{i=1}^{15} f_i^J(a_i) \tag{3}$$

$$f_i^J(a) = \begin{cases} min_i - a, & \text{if } a \le min_i, \\ 0, & \text{if } min_i \le a \le max_i, \\ a - max_i, & \text{if } a \ge max_i, \end{cases} \tag{4}$$

As shown in Fig. 1, we give some samples of using the average pose prior and ours pose prior. When using the average pose prior, the projected output joints may be reasonable but the hand configures in 3D shape is not valid. We think this is because the average pose prior penalizes all poses according to their distance from the average pose, which cannot distinguish valid or invalid hand configure. While we only penalize the invalid hand pose and also determine this penalty term according to the degree of deviation, which results in much better performance.

2. Implementation and Architecture Details

For the weighting factors in Section 3.4 of the main text, we set $w_{3d} = 1$, $w_{2d} = 0.001$, $w_{con} = 0.0002$, $w_{geo} = 0.001$, $w_{photo} = 0.005$, $w_{regu} = 0.01$, $w_{ori} = 100$, $w_{SSIM} = 0.2$, $w_C = 0.5$, $w_s = 10000$, and $w_J = 10$.

*Work done during an internship at Tencent AI Lab.
†Corresponding author: tuzhigang@whu.edu.cn

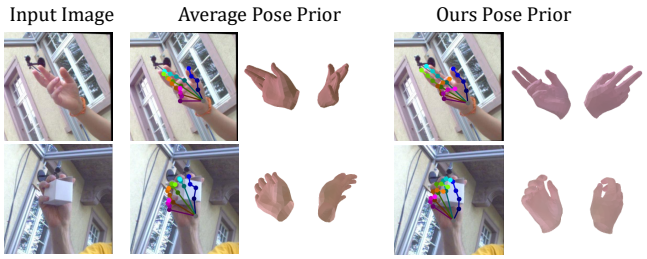

Figure 1: Qualitative comparison of the average pose prior [1] and ours pose prior. We give two samples with the input image, projected output keypoints, and 3D mesh in two viewpoints.

Bone	Azimuth	Pitch	Roll
$\overrightarrow{12}$	(-22.5,33.75)	(-22.5,22.5)	(0,90)
$\overrightarrow{23}$	(-5,5)	(-22.5,22.5)	(-5,5)
$\overrightarrow{34}$	(-5,5)	(-100,20)	(-5,5)
$\overrightarrow{56}$	(-10,10)	(-100,10)	(-5,5)
$\overrightarrow{67}$	(-5,5)	(-100,10)	(-5,5)
$\overrightarrow{78}$	(-5,5)	(-100,10)	(-5,5)
$\overrightarrow{910}$	(-10,10)	(-100,10)	(-5,5)
$\overrightarrow{1011}$	(-5,5)	(-100,10)	(-5,5)
$\overrightarrow{1112}$	(-5,5)	(-100,10)	(-5,5)
$\overrightarrow{1314}$	(-10,10)	(-100,10)	(-5,5)
$\overrightarrow{1415}$	(-5,5)	(-100,10)	(-5,5)
$\overrightarrow{1516}$	(-5,5)	(-100,10)	(-5,5)
$\overrightarrow{1718}$	(-10,20)	(-100,10)	(-20,5)
$\overrightarrow{1819}$	(-5,5)	(-100,10)	(-5,5)
$\overrightarrow{1920}$	(-5,5)	(-100,10)	(-5,5)

Table 1: The minimum and maximum values in *degrees* of the joint angle parameters used in our proposed skeleton regularization term.

For the HO-3D, we use the 2D keypoints information to crop the hand region from the raw image and then resize the cropped image into 224×224 as training samples and rely on the provided 2D bounding box to crop the testing frames. We don't apply any data augmentation for the FreiHAND.

The 3D reconstruction network has an encoder-decoder architecture. The EfficientNet-b0 [2] encodes the input image $I \in \mathbb{R}^{224\times224\times3}$ to a latent feature map $m_h \in \mathbb{R}^{7\times7\times1536}$, where we also take an intermediate feature map $m_l \in \mathbb{R}^{56\times56\times32}$. A vector $v_h \in \mathbb{R}^{1536}$ is got from m_h through max pooling and then passed through a series of fully connected layers f_{base}. Then multiple heads $(f_{pose}, f_{shape}, f_{trans}, f_{rot}, f_{scale})$ are used to estimate pose θ, shape β, translation T, rotation R and scale s (Table 2). We use a series of 2D convolution layers f_{conv} and two heads f_{tex}, f_{light} to encode the higher resolution feature m_l into the hand texture C and scene lighting L (Table 3).

Stage	Operator	Output
f_{base}	Linear(1536,1024),BN,ReLU	1×1024
	Linear(1024,512),BN,ReLU	1×512
f_{pose}	Linear(512,128),ReLU	1×128
	Linear(128,30)	1×30
f_{shape}	Linear(512,128),ReLU	1×128
	Linear(128,10)	1×10
f_{trans}	Linear(512,128),ReLU	1×128
	Linear(128,32)	1×32
	Linear(128,3)	1×3
f_{rot}	Linear(512,128),ReLU	1×128
	Linear(128,32)	1×32
	Linear(128,3)	1×3
f_{scale}	Linear(512,128),ReLU	1×128
	Linear(128,32)	1×32
	Linear(128,1)	1×1

Table 2: The hand regressor architecture. Linear transformation layers are given as Linear (in_size, out_size).

Stage	Operator	Output
f_{conv}	Conv2d(32,48,10,4,1), ReLU	$13 \times 13 \times 48$
	MaxPool(3,2)	$6 \times 6 \times 48$
	Conv2d(48,64,3,1,0), ReLU	$4 \times 4 \times 64$
	MaxPool(2,2)	$2 \times 2 \times 64$
	Flatten	1×256
f_{tex}	Linear(256,64),ReLU	1×64
	Linear(64,2334)	1×2334
f_{light}	Linear(256,64),ReLU	1×64
	Linear(64,11)	1×11

Table 3: The texture and lighting regressor architecture. Convolution parameters are given as Conv2d (in_channels, out_channels, kernel_size, stride, padding). The 2D max pooling is given as MaxPool (kernel_size, stride). The linear transformation layer is given as Linear (in_size, out_size).

Method / Supervision	MANO-CNN		Ours	
	MPJP↓	MPVPE↓	MPJPE↓	MPVPE↓
FSL	8.72	8.84	8.66	8.77
SSL	12.75	12.81	10.57	10.60

Table 4: Comparison of unaligned results of MANO-CNN and *Ours* under fully-supervision (FSL) or self-supervision (SSL) on the FreiHAND testing set.

3. Comparison to using GT 3D as Supervision

In all of our experiments, we do not use the scale information (provided by the FreiHAND testing set) or the depth information (provided by the HO-3D testing set). It is typical to evaluate the 3D hand pose and shape estimation in the hand-centric coordinate (e.g., using Procrustes

Evaluation Matrix \ Keypoint Set	OpenPose	Predicted		Projected	
		w/o 2D-3D	w/ 2D-3D	w/o 2D-3D	w/ 2D-3D
Per Joint	0.807	0.820	**0.828**	0.808	0.825
Per Frame Mean	0.799	0.815	**0.825**	0.805	0.823
Per Frame Max	0.466	0.517	0.545	0.543	**0.577**

Table 5: Comparison of the AUC (higher is better) of 2D keypoint sets used or outputted at the training stage on FreiHAND in different evaluation matrix, where the 2D error value is from 0 to 50 pixel.

alignment), but it is also important to accurately output 3D hands with accurate position and scale. To this end, we show our results in camera coordinates of the proposed self-supervised (SSL) method and the results of a fully-supervised (FSL) scheme. Note that this section reports the results of raw output without using Procrustes alignment. For FSL, we use ground truth 2D keypoint annotations whose keypoint confidences are set to be the same, and a 3D joint loss is additionally used to give real 3D supervision. The 3D joint loss enforces the $k = 21$ of output joints $J = \{j_i \in \mathbb{R}^3 | 1 \leq i \leq k\}$ and ground truth joints $J^{gt} = \{j_i^{gt} \in \mathbb{R}^3 | 1 \leq i \leq k\}$ to be aligned. We add the E_{3dj} to E by a weighting factor $w_{3dj} = 100$.

$$E_{3dj} = \frac{1}{k} \sum_{i=1}^{k} \| j_i - j_i^{gt} \|_2^2 \tag{5}$$

A big advantage of using 3D annotation is that it can help the network to learn the depth value of the output hand. In Table 4, we compare unaligned results of our method with MANO-CNN under the FSL and SSL settings. Both approaches get much better performance under FSL than SSL, and our approach outperforms MANO-CNN under FSL and SSL (a 0.06*cm* decrease in MPJPE under FSL and a 2.18*cm* decrease in MPJPE under SSL). We further find that when degrading the supervision strength from FSL to SSL, our method shows less performance degradation than MANO-CNN, where the MPJPE of our method increases by 22.1% while by 46.2% in MANO-CNN. We think this is because our approach is robust to the self-supervised setting than MANO-CNN.

4. More Comparison of Different 2D Keypoint Sets

In Section 4.4.2 and Fig 4 of the main body, we compared the fraction of "Per Joint" 2D error. Here, we further visualize the fraction of "Per Frame Max" 2D error, i.e., the fraction of frames within maximum 2D MPJPE in *pixel*, in Fig. 2. The AUC (0∼50 pixel) of "Per Joint" error, "Per Frame Mean" error and "Per Frame Max" error in 2D are shown in Table 5.

As shown in Fig. 2 and Table 5, the 2D-3D consistency loss E_{con} improves the AUC of both the 2D branch (from ***Predicted w/o 2D-3D*** to ***Predicted w/ 2D-3D***) and the 3D branch (from ***Projected w/o 2D-3D*** to ***Projected w/ 2D-3D***). All results of the 2D and 3D branches outperform ***OpenPose*** which is used as the keypoint supervision source.

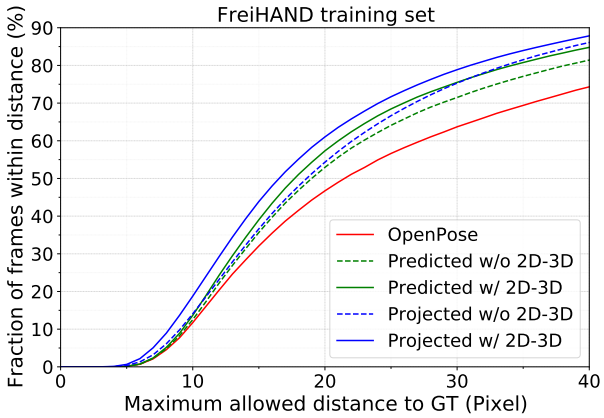

Figure 2: A comparison of 2D keypoint sets used or outputted at the training stage on FreiHAND. The fraction of frames within maximum joint distance is plotted.

As shown in Table 5, in term of "Per Joint" and "Per Frame Mean", the predicted results from the 2D branch are better than the projected results from the 3D branch since the 2D branch is designed for 2D keypoints estimation while the results of the 3D branch are projected from 3D outputs. While in terms of "Per Frame Max", we find that the projected results are better than the predicted ones. We believe this is due to the constraints contained in the MANO model, the projected results eliminate outlier 2D results. Thus, the AUC of "Per Frame Max", which is more sensitive to outliers, is greatly improved from OpenPose to Predicted & Projected due to E_{con} and the powerful regularization characteristic of network training (Section 4.4.2 of the main body).

5. Qualitative Results

We provide more qualitative results of single-view 3D hand reconstruction. Fig. 3 and 4 report the qualitative results of samples from the FreiHAND testing set. Fig. 5 shows the qualitative results of samples from the HO-3D testing set. Note that hands in HO-3D suffer from more

serious occlusion, resulting in more object or background pixels in the masked foreground during the self-supervised texture learning. So the texture estimation is less accurate on the HO-3D.

Since the HO-3D provides video sequences in the testing set, we visualize the results of sequence frames on the HO-3D testing set in Fig. 6. Although we do not use temporal information during the training and only use a single image for inference, our model outputs accurate shape with consistent shape and texture for each video sequence.

References

[1] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Conference on Computer Vision and Pattern Recognition*, 2019. 1, 2

[2] Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, 2019. 2

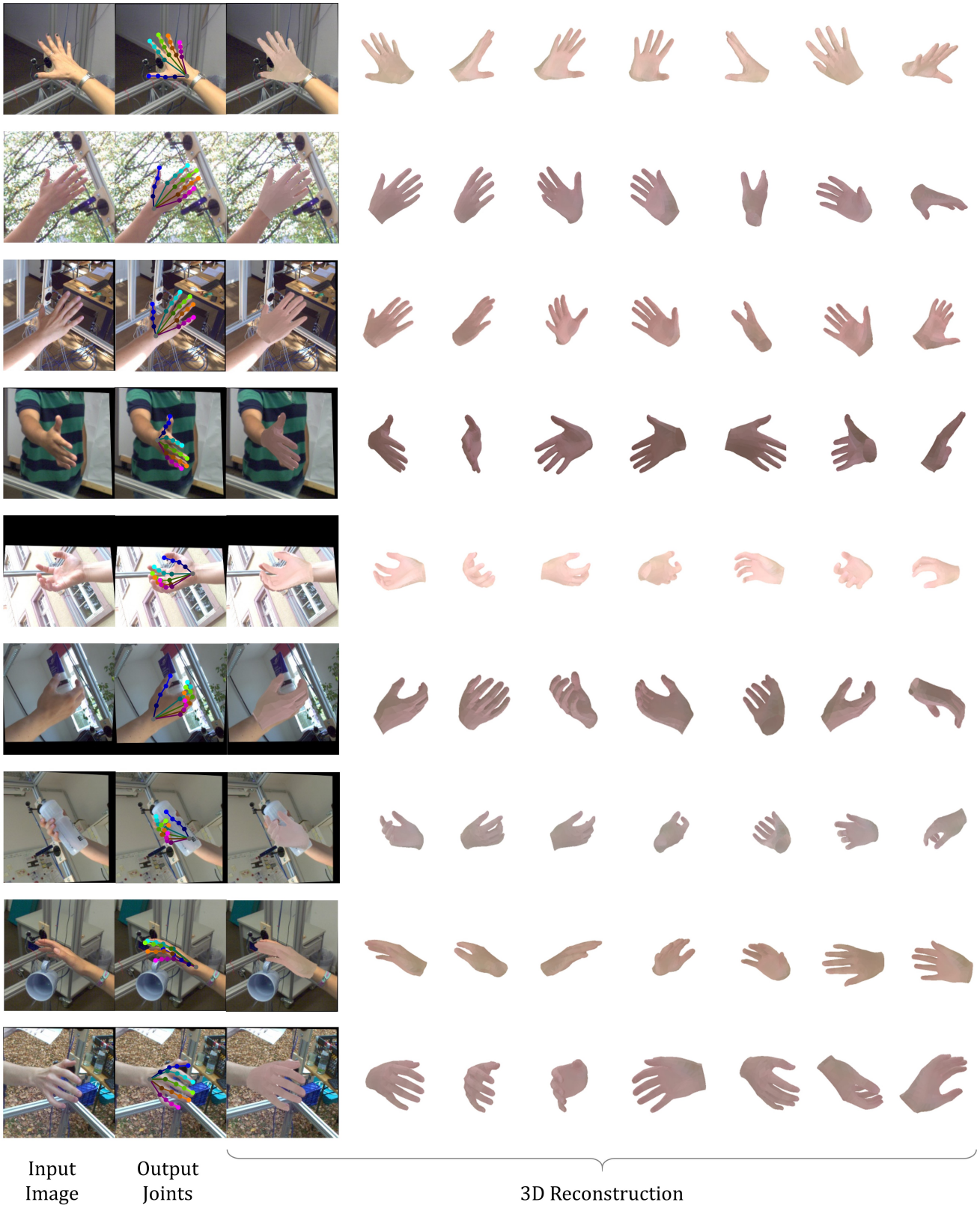

Figure 3: Qualitative visualization of our method on the FreiHAND testing set (Part 1). From left to right: input image, output 3D joints projected to image space, output 3D mesh projected to image space, 3D mesh from different views (4th-10th column).

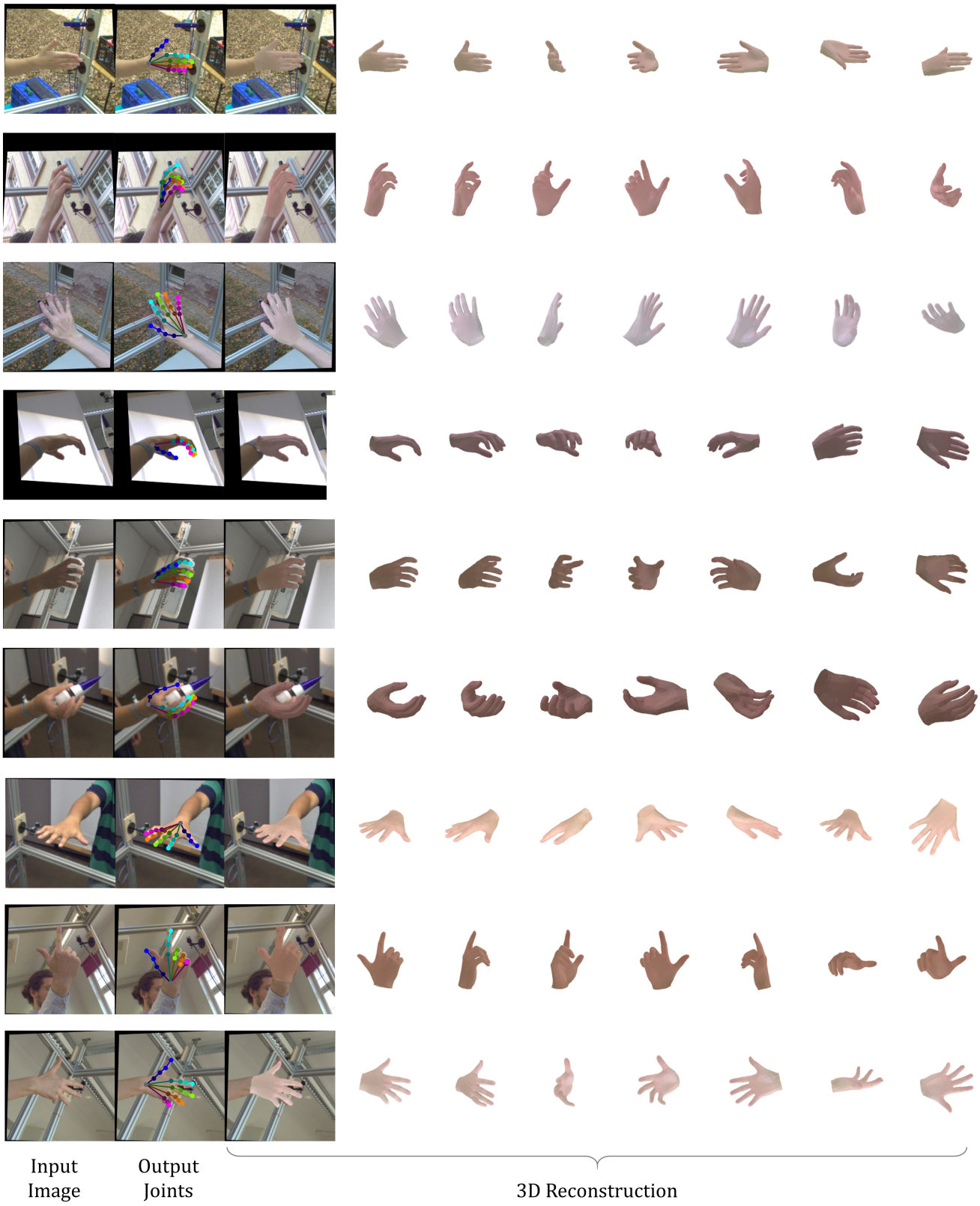

Figure 4: Qualitative visualization of our method on the FreiHAND testing set (Part 2). From left to right: input image, output 3D joints projected to image space, output 3D mesh projected to image space, 3D mesh from different views (4th-10th column).

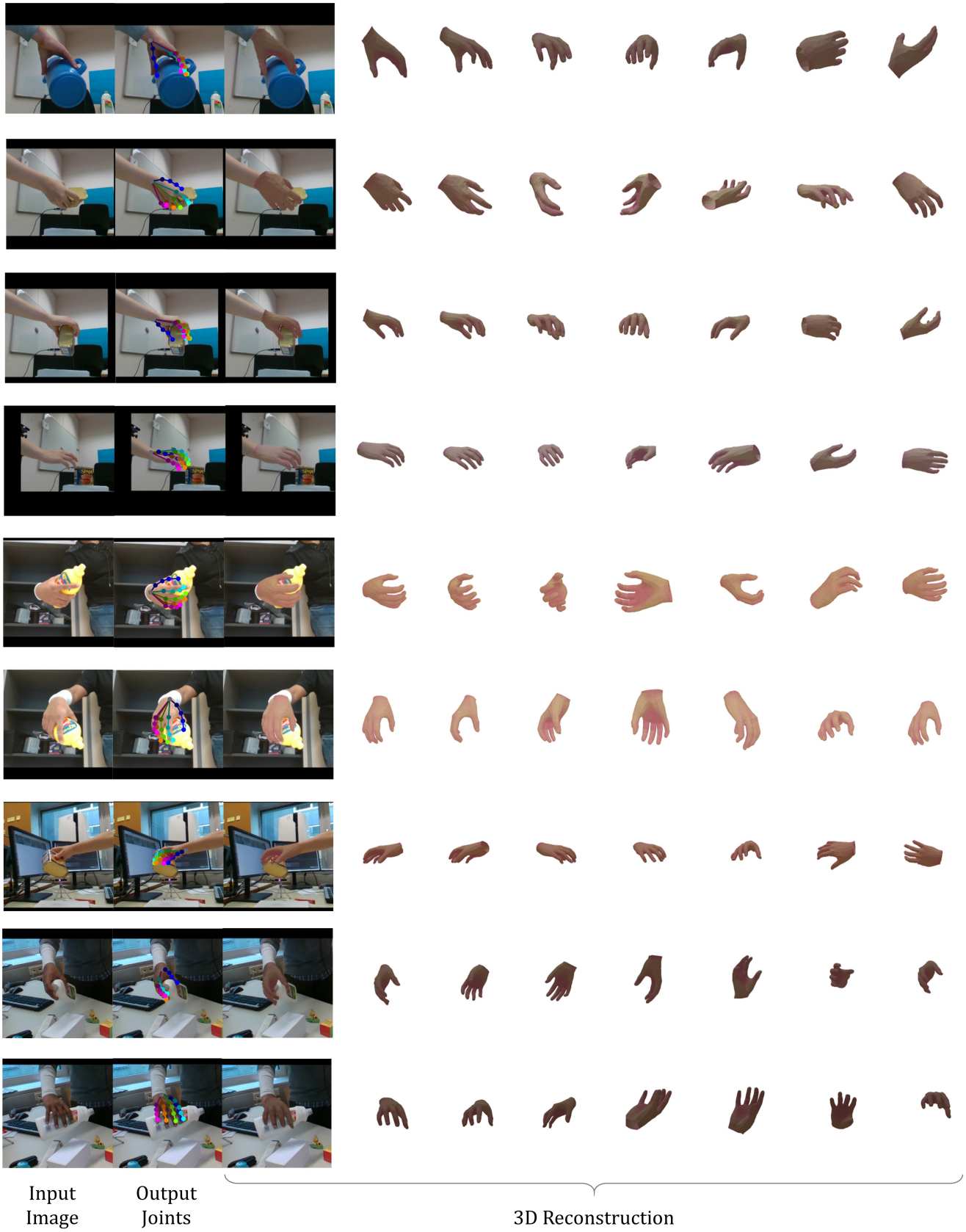

Figure 5: Qualitative visualization of our method on the HO-3D testing set. From left to right: input image, output 3D joints projected to image space, output 3D mesh projected to image space, 3D mesh from different views (4th-10th column).

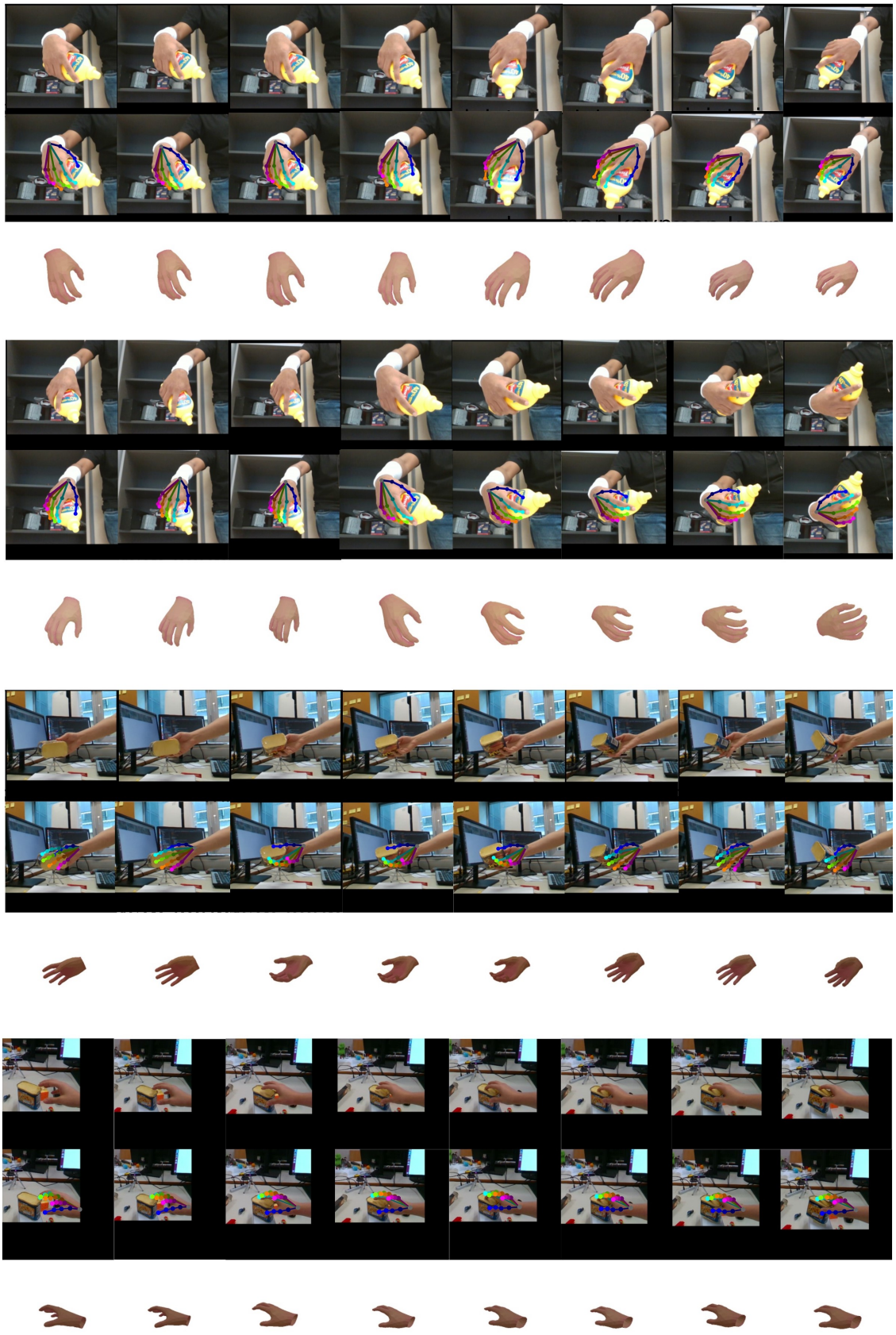

Figure 6: Qualitative visualization of our method on the HO-3D testing set. We give hand reconstruction results for four sequences. Eight samples are shown in each sequence, and we visualize the input image, projected output joints, and output mesh for each sample.