# Supplementary Material of "DeepHandMesh: A Weakly-supervised Deep Encoder-Decoder Framework for High-fidelity Hand Mesh Modeling"

Gyeongsik Moon[1], Takaaki Shiratori[2], and Kyoung Mu Lee[1]

[1] ECE & ASRI, Seoul National University, Korea
[2] Facebook Reality Labs
`{mks0601,kyoungmu}@snu.ac.kr`, `tshiratori@fb.com`

In this supplementary material, we present more experimental results that could not be included in the main manuscript due to the lack of space.

## 1 Texture loss

Although the overall shape of the hand mesh is close to the groundtruth, some vertices can move inconsistently across time steps. To prevent this, we employ texture consistency loss, similar to [7,8]. Specifically, we first train the DeepHandMesh and obtain a hand mesh of a neutral pose, which is considered as the easiest pose to estimate. Then, the mesh is used to unwrap neutral pose RGB images from all views of our dataset to a 1024×1024 UV texture $\mathbf{T}^*$ using Poisson reconstruction [1]. We use $\mathbf{T}^*$ as a groundtruth texture and force texture $\mathbf{T}$ of each iteration to be the same with $\mathbf{T}^*$. $\mathbf{T}$ is obtained by unwrapping corresponding RGB images from randomly selected $C_{\text{out}}$ views using mesh output of current iteration in a differentiable way [3,7]. The resolution of $\mathbf{T}$ is 256×256, and we resized $\mathbf{T}^*$ to the same resolution of $\mathbf{T}$. To normalize illumination, we perform normalized cross-correlation (NCC) on each 8×8 patch of $\mathbf{T}$ and $\mathbf{T}^*$ after applying the average blur. The loss function $L_{\text{tex}}$ is defined as follows:

$$L_{\text{tex}} = \delta||(\text{NCC}(\mathbf{T}) - \text{NCC}(\mathbf{T}^*))||_1, \tag{1}$$

where $\delta$ is a binary tensor whose value is one if the corresponding UV coordinate has visible RGB value. This loss function is applied to fine-tune trained DeepHandMesh 15 epochs. The total loss function in fine-tuning stage is $L + L_{\text{tex}}$. During fine-tuning, we used the same learning rate $10^{-4}$, and it is reduced by a factor of 10 at the 10th and 12th epochs. $\lambda_{\text{lap}}$ is set to 1.

Figure 1 shows standard deviation $\sigma$ of each pixel value on the UV space after fine-tuning without $L_{\text{tex}}$ (a) and with (b). (c) shows $\sigma$ difference between (a) and (b), which is defined as (b) subtracted by (a) (*i.e.*, blue colors in (c) indicate $\sigma$ decreased after fine-tuning). As the figures show, $L_{\text{tex}}$ helps to decrease $\sigma$, but not at a noticeable amount. We guess that this is because (a) already shows low standard deviation.

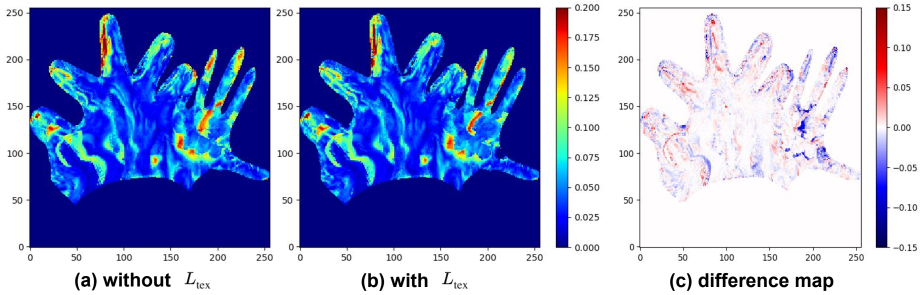


Fig. 1: Visualization of effect of $L_{tex}$.

## 2 Effect of the skeleton corrective

To demonstrate the effectiveness of the identity-dependent skeleton corrective $\varDelta\mathbf{S}_\beta$, we compare 3D joint distance error $\mathrm{P}_{\mathrm{err}}$ in Table 1. The error is defined as a Euclidean distance between $\mathbf{P}$ and $\mathbf{P}^*$, where $*$ indicates groundtruth. The table shows our skeleton corrective refines the skeleton of the initial hand model successfully.

| Settings | $\mathrm{P}_{\mathrm{err}}$ (mm) |
|---|---|
| Without $\varDelta\mathbf{S}_\beta$ | 4.82 |
| **Ours (full)** | **2.38** |

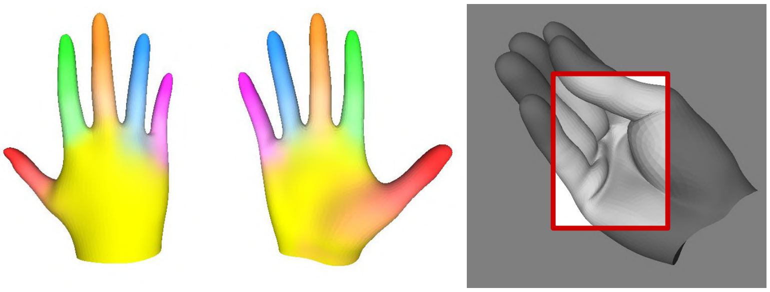
Table 1: 3D joint distance error $\mathrm{P}_{\mathrm{err}}$ (mm) comparison between with and without our identity-dependent skeleton corrective $\varDelta\mathbf{S}_\beta$.

## 3 Skinning weight corrective

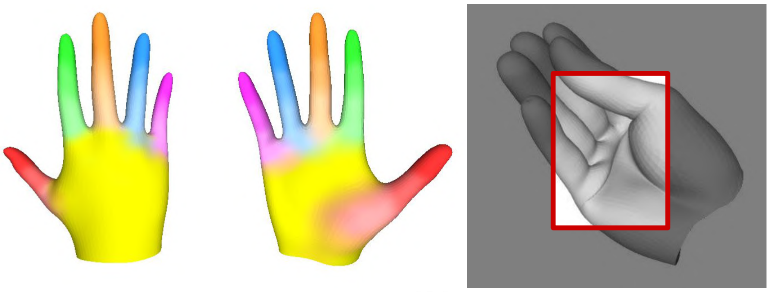
To refine pre-defined skinning weight $\mathbf{W}$, we estimate identity-dependent skinning weight corrective $\varDelta\mathbf{W}_\beta \in \mathbb{R}^{V\times J}$ from the identity vector $\beta$ using two fully-connected layers. The refined skinning weight $\mathbf{W}^* \in \mathbb{R}^{V\times J}$ is obtained as follows:

$$\mathbf{w}^*_{v,j} = \frac{\max(\mathbf{w}_{v,j} + \varDelta\mathbf{w}_{v,j}, 0)}{\sum_{j=1}^{J}\max(\mathbf{w}_{v,j} + \varDelta\mathbf{w}_{v,j}, 0)}, v = 1, \ldots, V, j = 1, \ldots, J,$$

where $\mathbf{w}^*_{v,j}$, $\mathbf{w}_{v,j}$, and $\varDelta\mathbf{w}_{v,j}$ denote refined skinning weight, initial skinning weight, and skinning weight corrective of $j$th joint of $v$th vertex from $\mathbf{W}^*$, $\mathbf{W}$, and $\varDelta\mathbf{W}_\beta$, respectively. We clamp the refined skinning weight to be positive value and normalize it to make the summation 1. To encourage locality like Loper et al. [4], $\varDelta\mathbf{w}_{v,j}$ is estimated only when $\mathbf{w}_{v,j} \neq 0$. Otherwise, it is set to zero.



(a) without $\Delta\mathbf{W}_\beta$



(b) with $\Delta\mathbf{W}_\beta$

Fig. 2: Color-coded skinning weight (left and middle) and deformed hand mesh (right) comparison between without and with skinning weight corrective $\Delta\mathbf{W}_\beta$.

We trained the DeepHandMesh with an additional $\Delta\mathbf{W}_\beta$. Figure 2 shows color-coded skinning weight and deformed hand mesh. As the figure shows, the skinning weight of mainly finger root parts changed, and this change results in different skin deformation around the finger root parts. However, as there is no groundtruth mesh, it is hard to tell which hand mesh is more realistic clearly. We believe this skinning weight corrective $\Delta\mathbf{W}_\beta$ can be helpful when initial skinning weight $\mathbf{W}$ is bad and a bottleneck of better performance.

## 4 Dataset description

We provide detailed descriptions and visualizations of the sequences in our newly constructed dataset. The pre-defined hand poses include various sign languages that are frequently used in daily life and extreme poses where each finger assumes a maximally bent or extended. When capturing the conversational gestures, subjects are instructed with minimal instructions, for example, waving their hands as if telling someone to come over. The hand poses in our dataset are carefully chosen to sample a variety of poses and conversational gestures while being easy to follow by capture participants.

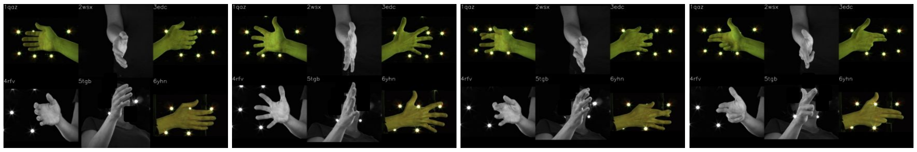
The hand pose estimator was trained on our held-out human annotation dataset, which includes the 3D rotation center coordinates of hand joints from

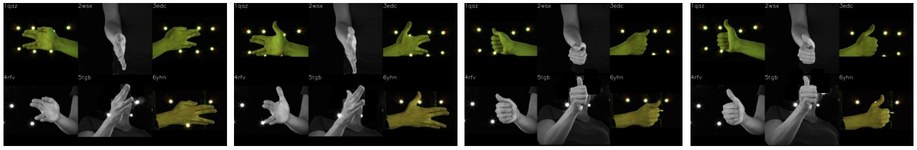
6K frames. The predicted 2D hand joint locations of each view were triangulated with RANSAC to robustly obtain the groundtruth 3D hand joint coordinates. The hand pose estimator used to obtain groundtruth 3D hand joint coordinates achieves *2.78 mm error* on our held-out human-annotated test set, which is quite low. From the 3D reconstruction, we rendered groundtruth depth maps for all camera views.

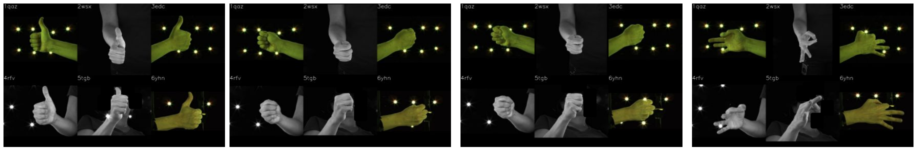
**Training set.** Figure 3, 4, and 5 show the pre-defined hand poses in training set. Belows are detailed descriptions of each sequence.

- neutral relaxed: the neutral hand pose. Hands in front of the chest, fingers do not touch, and palms face the side.
- neutral rigid: the neutral hand pose with maximally extended fingers, muscles tense.
- good luck: hand sign language with crossed index and middle fingers.
- fake gun: hand gesture mimicking the gun.
- star trek: hand gesture popularized by the television series Star Trek.
- star trek extended thumb: “star trek” with extended thumb.
- thumb up relaxed: hand sign language that means “good”, hand muscles relaxed.
- thumb up normal: “thumb up”, hand muscles average tenseness.
- thumb up rigid: “thumb up”, hand muscles very tense.
- thumb tuck normal: similar to fist, but the thumb is hidden by other fingers.
- thumb tuck rigid: “thumb tuck”, hand muscles very tense.
- aokay: hand sign language that means “okay”, where palm faces the side.
- aokay upright: “aokay” where palm faces the front.
- surfer: the SHAKA sign.
- rocker: hand gesture that represents rock and roll, where palm faces the side.
- rocker front: the “rocker” where palm faces the front.
- rocker back: the “rocker” where palm faces the back.
- fist: fist hand pose.
- fist rigid: fist with very tense hand muscles.
- alligator closed: hand gesture mimicking the alligator with a closed mouth.
- one count: hand sign language that represents “one.”
- two count: hand sign language that represents “two.”
- three count: hand sign language that represents “three.”
- four count: hand sign language that represents “four.”
- five count: hand sign language that represents “five.”
- indextip: thumb and index fingertip are touching.
- middletip: thumb and middle fingertip are touching.
- ringtip: thumb and ring fingertip are touching.
- pinkytip: thumb and pinky fingertip are touching.
- palm up: has palm facing up.
- finger spread relaxed: spread all fingers, hand muscles relaxed.
- finger spread normal: spread all fingers, hand muscles average tenseness.
- finger spread rigid: spread all fingers, hand muscles very tense.
- capisce: hand sign language that represents “got it” in Italian.

- claws: hand pose mimicking claws of animals.
- peacock: hand pose mimicking peacock.
- cup: hand pose mimicking a cup.
- shakespeareyorick: hand pose from Yorick from Shakespeare's play Hamlet.
- dinosaur: hand pose mimicking a dinosaur.
- middle finger: hand sign language that has an offensive meaning.


neutral relaxed neutral rigid good luck fake gun


star trek star trek extended thumb thumb up relaxed thumb up normal


thumb up rigid thumb tuck normal thumb tuck rigid aokay

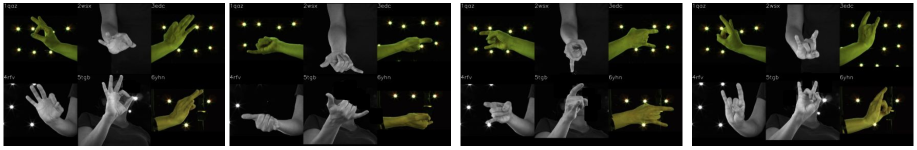
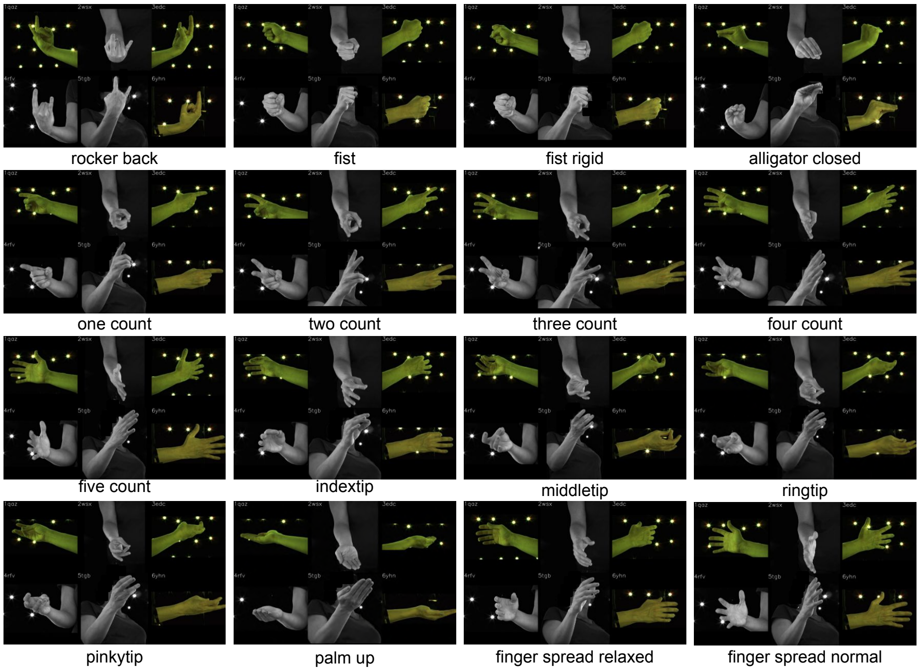

aokay upright surfer rocker rocker front

Fig. 3: Visualization of the sequences in the training set.



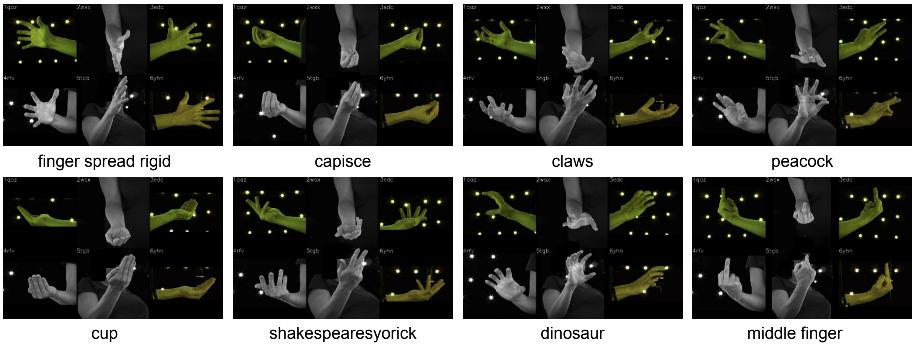
rocker back, fist, fist rigid, alligator closed

one count, two count, three count, four count

five count, indextip, middletip, ringtip

pinkytip, palm up, finger spread relaxed, finger spread normal

Fig. 4: Visualization of the sequences in the training set.



finger spread rigid, capisce, claws, peacock

cup, shakespearesyorick, dinosaur, middle finger

Fig. 5: Visualization of the sequences in the training set.

**Testing set.** Figure 6 shows the conversational gestures in testing set. Belows are detailed descriptions of each sequence.

- five count: count from one to five.
- five countdown: count from five to one.
- fingertip touch: thumb touch each fingertip.
- relaxed wave: wrist relaxed, fingertips facing down and relaxed, wave.
- fist wave: rotate wrist while hand in a fist shape.
- prom wave: wave with fingers together.
- palm down wave: wave hand with the palm facing down.
- index finger wave: hand gesture that represents “no” sign.
- palmer wave: palm down, scoop towards you, like petting an animal.
- snap: snap middle finger and thumb.
- finger wave: palm down, move fingers like playing the piano.
- finger walk: mimicking a walking person by index and middle finger.
- cash money: rub thumb on the index and middle fingertips.
- snap all: snap each finger on the thumb.

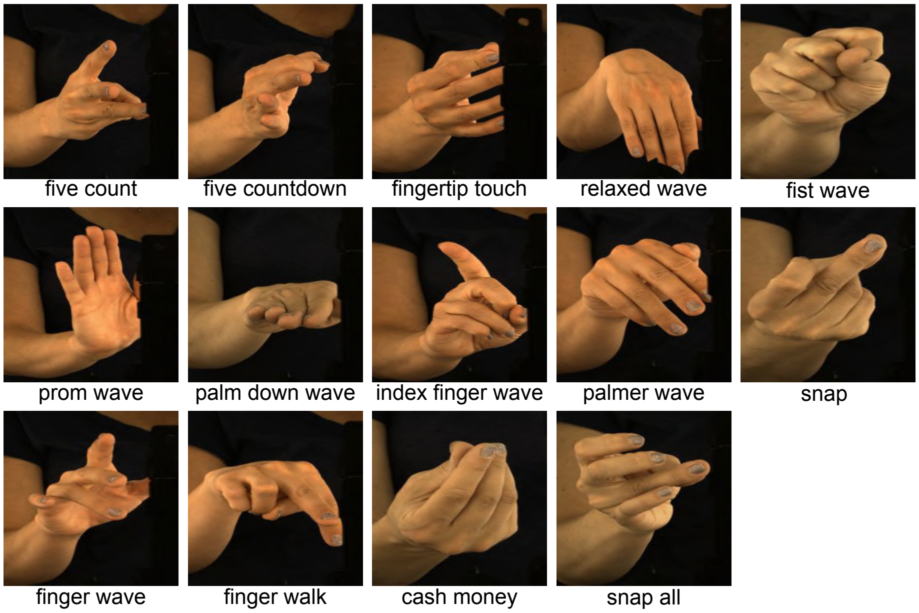


Fig. 6: Visualization of the sequences in the testing set.

Figure 7 shows rendering of our constructed multi-view studio for the data capture.

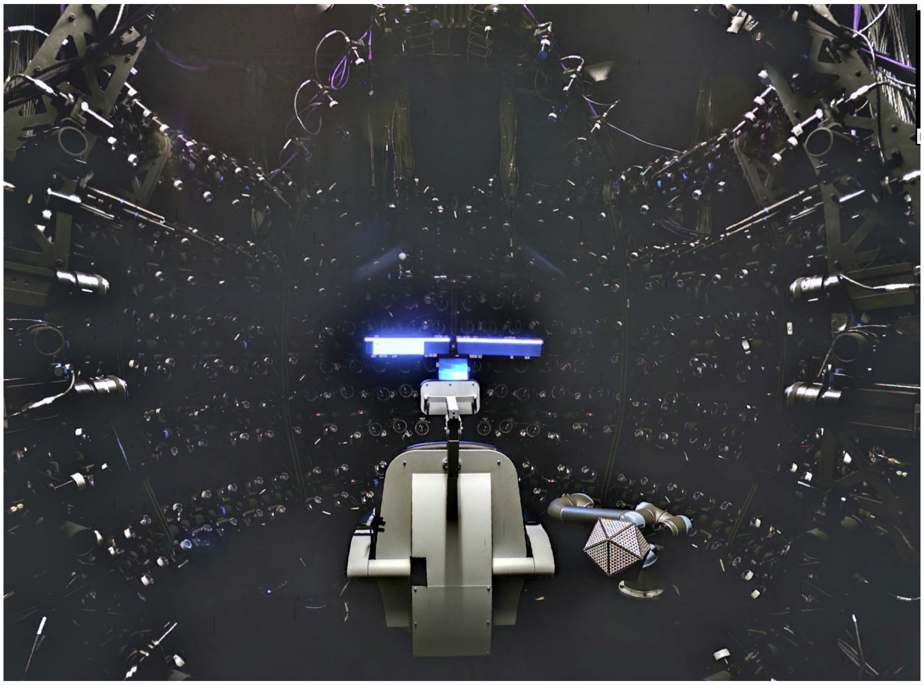


Fig. 7: Rendering of our constructed multi-view studio.

## 5 Comparison with state-of-the-art methods

**Comparison with MANO under the similar mesh resolution.** We train and test our DeepHandMesh with a hand model the resolution of which is similar to that of MANO, and compare its qualitative results with those from MANO in Figure 8. Our low-resolution DeepHandMesh uses a hand mesh model with 792 vertices, while the MANO is based on a hand mesh model with 778 vertices. The figure shows that our DeepHandMesh provides a more realistic hand mesh compared with MANO under the similar resolution of the hand mesh model. Note that our DeepHandMesh is trained in a weakly-supervised way without per-vertex loss function, while MANO is based on fully-supervised training with per-vertex loss. When we train low resolution version of the DeepHandMesh, $L2$ norm regularizers are used for the correctives (*i.e.*, $\Delta\mathbf{S}_\beta$, $\Delta\mathbf{M}_\beta$, and $\Delta\mathbf{M}_\theta$).

**Comparison with MANO on the dataset of MANO.** We tried to train and test our DeepHandMesh on MANO dataset [5]. However, we observed that there are only 50 registrations available for each subject, which are not large enough to train DeepHandMesh. Also, some of the 3D scans include an object grasped by a hand. This makes training our system on the MANO dataset hard because rendered groundtruth depth maps $\mathcal{D}^*$ include those objects. Although we tried to use the registered meshes that do not include the objects for depth map rendering, we noticed that the rendered depth map lost high-frequency information in the original 3D scans because of the low-resolution mesh in MANO, which makes the depth maps hard to be used as groundtruth depth maps $\mathcal{D}^*$.

**Comparison with Kulon et al. [2].** We also tried to compare our DeepHandMesh with Kulon et al. [2]. They use mesh supervision when they train their high-resolution hand model (*i.e*, 7,907 vertices). As there is no mesh groundtruth in our dataset, we trained their model with the same loss functions as ours (*i.e.*, **Pose loss** and **Depth map loss**). We observed that without per-vertex mesh supervision, their model provides severely distorted hand mesh. We could not train our DeepHandMesh on the Panoptic dome dataset [6] that Kulon et al. [2] used because high-quality multi-view depth maps are not available in that dataset. Instead, we compare hand mesh output of the same hand pose from our DeepHandMesh and Kulon et al. [2] trained on our dataset and the Panoptic dome dataset [6], respectively. Although this is not a perfectly fair comparison, we think that this can roughly show how the final outputs of each method are different. Figure 9 shows our DeepHandMesh provides significantly more realistic hand mesh than Kulon et al. [2].

**Comparison on publicly available datasets.** As our DeepHandMesh is a personalized system, it is hard to compare with other hand models (*i.e.*, MANO [5] and Kulon et al. [2]) that support cross-identity on publicly available datasets. Instead, we tried to best to provide comparisons between them on our dataset.

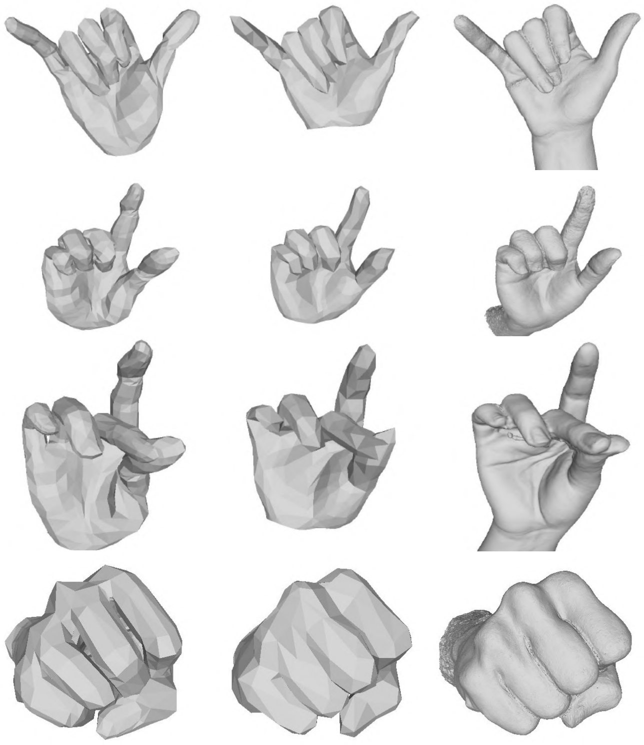


Fig. 8: Estimated hand mesh comparison with MANO [5] and our DeepHandMesh using the hand model of the similar resolution.

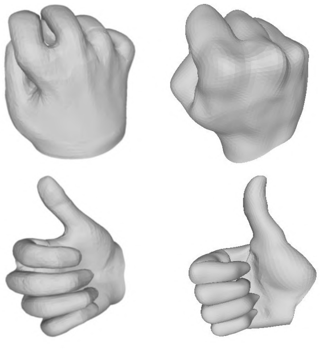


Fig. 9: Estimated hand mesh comparison with Kulon et al. [2] and our DeepHandMesh. The results of Kulon et al. [2] are taken from their paper.

## 6 Qualitative rendered results

We provide rendered result using texture obtained from Section 1 in Figure 10.

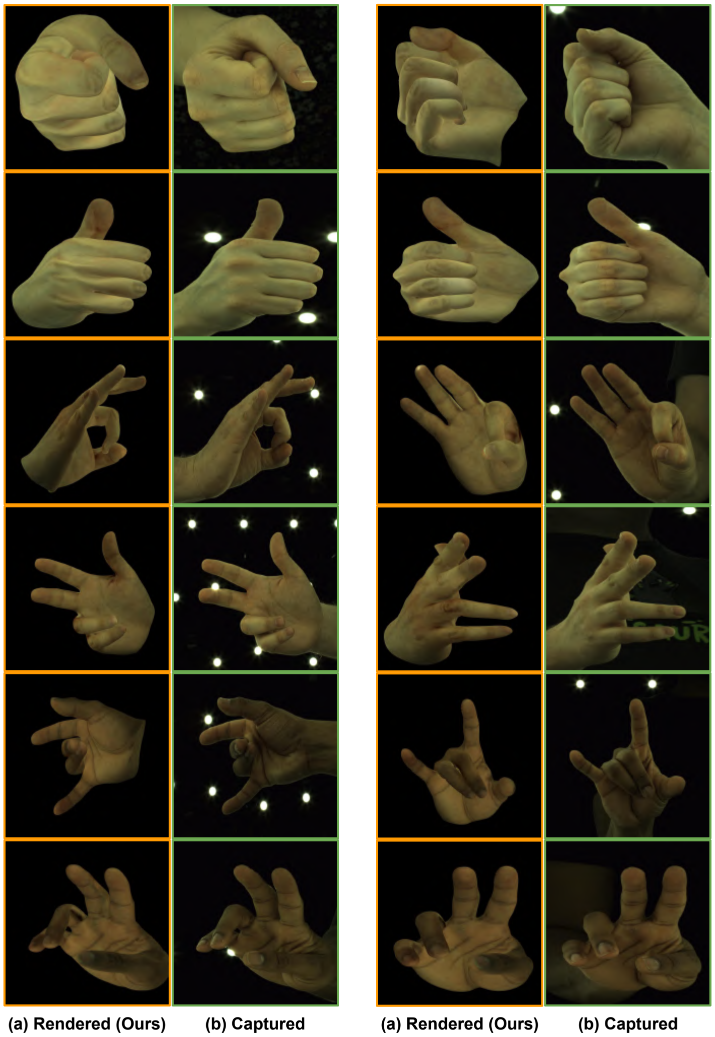


Fig. 10: Comparison between our rendered image and captured image from cameras.

## References

1. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the fourth Eurographics symposium on Geometry processing. vol. 7 (2006)
2. Kulon, D., Wang, H., Güler, R.A., Bronstein, M., Zafeiriou, S.: Single image 3D hand reconstruction with mesh convolutions. BMVC (2019)
3. Lombardi, S., Saragih, J., Simon, T., Sheikh, Y.: Deep appearance models for face rendering. ACM TOG (2018)
4. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM TOG (2015)
5. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM TOG (2017)
6. Simon, T., Joo, H., Matthews, I., Sheikh, Y.: Hand keypoint detection in single images using multiview bootstrapping. In: CVPR (2017)
7. Wei, S.E., Saragih, J., Simon, T., Harley, A.W., Lombardi, S., Perdoch, M., Hypes, A., Wang, D., Badino, H., Sheikh, Y.: VR facial animation via multiview image translation. ACM TOG (2019)
8. Yoon, J.S., Shiratori, T., Yu, S.I., Park, H.S.: Self-supervised adaptation of high-fidelity face models for monocular performance tracking. In: CVPR (2019)