PALM: A Dataset and Baseline for Learning Multi-Subject Hand Prior

Zicong Fan[1,2,3,†] Edoardo Remelli[1] David Dimond[1] Fadime Sener[1]
Liuhao Ge[1] Bugra Tekin[1] Cem Keskin[1] Shreyas Hampali[1]
[1]Meta Reality Labs [2]ETH Zürich [3]Max Planck Institute for Intelligent Systems, Tübingen

1. PALM Visualizations

Figure 3 shows several images from PALM dataset along with the rendered meshes, MANO registrations and hand masks.

2. Personalization with InterHand2.6M

Figures 4 and 5 show more qualitative results on the InterHand2.6M dataset for the single-image personalization. We show results with rendering in training (input image) environment and novel environment under novel poses. Notice that the avatars from our method are more realistic than the baseline methods.

3. PALM Content

We provide the following in the release of PALM dataset:

- Multi-view RGB images.
- 3D meshes for each pose.
- MANO registrations for each pose.
- 2D and 3D keypoint for each pose.
- Camera calibration for each subject.
- Mask and depth map renderings for each image.

4. Implementation Details

Training requirements: Training the PALM-Net prior takes around 48 hours on 8 H200 GPUs. In particular, at each step, we randomly pick 128 images from the training data and evenly distribute the 128 images to the 8 GPUs for computing the gradients. The personalization takes around 3.5 hours on a single H200 GPU and we randomly sample 8 images to train. To reduce training time, we select top 1848 images from our dataset to train based on the lowest fitting errors. More images can be used to train our prior given more computation resources.

Multi-stage prior training: We use a multi-stage approach to train the PALM-Net prior. First, we pretrain the geometry network of PALM-Net with only SDF derived from a high-resolution MANO mesh using 7 subjects. This is to initialize a hand-like shape of the geometry network to allow more stable convergence before training on image data. Using this pretrained geometry network, we train on image data. In particular, we freeze the geometry network in the first 6000 steps and unfreeze after to avoid the under-trained radiance network affecting the hand geometry. We do random sampling of 3D points along the ray at first and enable importance sampling at step 1000. We segment out the foreground for training our model using segmentation masks and apply random color to the background training images to avoid hand models explaining the background pixels. We enable physically-based rendering after step 12k. We stop the prior training at step 24k. We use a learning rate of 1e-3 for all networks by default. For the shape code, appearance code, pose encoding layer, geometry, we use 1e-4.

Weights	λ_{pbr}	λ_{segm}	λ_{normal}	λ_{eikonal}	λ_{LPIPS}	λ_{LAP}	λ_{latent}
Prior training	0.2	0.1	0.5→0.1	1e-5	0.1	0.1	1e-3
Personalization	0.2	0.1	0	0	0.1	0.1	1e-3

Table 1. **Loss term weights in prior training and personalization**. During prior training, λ_{normal} drops from 0.5 to 0.1 linearly in the first 24k steps.

Recall that the total loss $\mathcal{L}$ for training our multi-subject model is defined as

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{\text{rf}} + \lambda_{\text{pbr}}\mathcal{L}_{\text{pbr}} + \lambda_{\text{segm}}\mathcal{L}_{\text{segm}} + \lambda_{\text{normal}}\mathcal{L}_{\text{normal}} \\ + \lambda_{\text{eikonal}}\mathcal{L}_{\text{eikonal}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}} \quad (1) \\ + \lambda_{\text{LAP}}\mathcal{L}_{\text{LAP}} + \lambda_{\text{latent}}\mathcal{L}_{\text{latent}}. \quad (2)\end{aligned}$$

To encourage smooth hand surface, we apply a Laplacian loss [11] by sampling 3D points around the hand surface and compute their normal values $\{\mathbf{n}^s\}$. We enforce these normal values to be similar to nearby surface points $\{\mathbf{n}^s_\epsilon\}$:

$$\mathcal{L}_{\text{LAP}} = \sum_{s \in \mathcal{S}} (\mathbf{n}^s \cdot \mathbf{n}^s_\epsilon - 1)^2. \tag{3}$$

The weights of the individual loss terms in prior training and in personalization can be found in Table 1.

Personalization training: Given a pretrained prior model and an input image, we perform personalization. In particular, we freeze the all weights except the shape code,

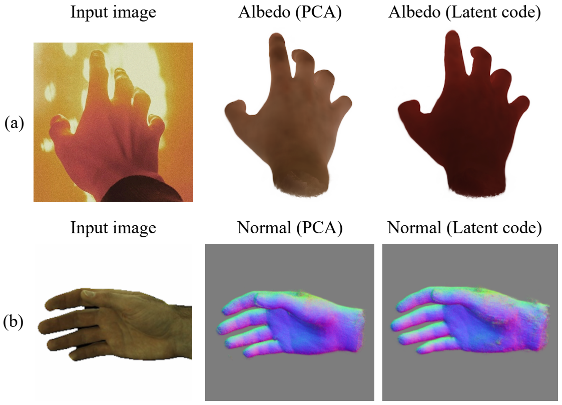

Figure 1. **Effects of PCA space on albedo and geometry and the effect of modelling environment**.

appearance code and the spherical gaussian environment map. We initialize PCA code as zeros. We train for 5200 steps in total, freeze the shape code from step 1500 to avoid overfitting, and enable physically-based rendering at step 2000. We have the same total loss $\mathcal{L}$ for personalization but with different weights (see Table 1). Essentially disabling normal supervision because it is not available in personalization. We also disable eikonal loss as the geometry network is not trained. For in-the-wild images, given an internet hand image, we use HaMeR [9] to obtain MANO hand poses and SAM [4, 5] to obtain hand segmentation using the language prompt "hand.". For InterHand2.6M images, we use the prompt "foreground." for SAM.

Modelling details: Unlike [12], which uses a temporal occupancy grid for empty space skipping [1], maintaining such a temporal grid is infeasible for a large amount of poses. To this end, we simply apply empty space skipping on points that are more than 1.5cm from the MANO mesh. This does not require us to maintain a memory-intensive occupancy grid and to scale to training on thousands of poses.

InterHand2.6M sequences: We use InterHand2.6M [6] to evaluate hand personalization on seen environment and novel poses for the baseline and ours. For each sequence, we uniformly select 20 images for our experiments. The first image is used for personalization and all images are for novel pose evaluation. The sequence names can be found in Table 2.

5. Additional Experiments

Effects of the PCA space: During prior learning, our method disentangle multiple subjects by optimizing on a latent vector for each subject. We empirically found that optimizing on the PCA space (obtained from a PCA decomposition of the subject latent codes) has more reasonable colored albedos and have fewer artifacts on the geometry (see Figure 1).

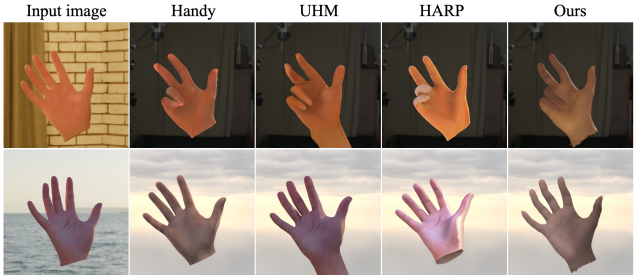

Figure 2. **Personalization results on synthetic dataset**. The first column shows the image used for personalization. The environment map for the training and evaluation images are different.

Synthetic results: Figure 2 shows the reconstructed hand avatars in the novel environments. Our method provides more plausible relighting results compared to the baselines even in very challenging biased lightings.

6. Baselines

We compare with Handy [10], UHM [7], HARP [3] in this paper. However, only HARP models lighting (using a single point light). For the seen environment and novel poses experiment on InterHand2.6M [6], we use the trained lighting to render HARP and use a white environment map to render the others as they do not support rendering with novel lights. For unseen environment experiments, for example, on in-the-wild sequences, we use the given environment map to render all baseline and ours. The rendering is done using the Cycles engine of Blender.

Handy [10]: Handy leverages a prior model trained on 3dMD hand scans from 1,200 subjects. It learns a texture prior using StyleGAN [2] from the texture maps of these scans, and a PCA-based geometry model from the corresponding hand meshes. To obtain the personalized texture, we optimize the latent code of the StyleGAN model to fit the input RGB images. Additionally, for synthetic and InterHand2.6M datasets, we incorporate a 3D keypoint loss based on ground-truth annotations.

HARP [3]: HARP does not use a prior model and performs personalization by optimizing the mesh vertices and texture map based on the MANO hand model. They also optimize a single light source to account for the shadow effects. We use the HARP personalization code for personalization.

UHM [7]: UHM provides a mesh based geometry prior model for the hand and obtains hand texture by copying the RGB image values to the texture map after registration. We use the official code for personalization and additionally include 3D keypoints for optimization.

Sequence Name	Image IDs
c0_ROM03_RT_No_Occlusion_400262	**12966**, 13080, 13194, 13308, 13422, 13536, 13650, 13764, 13878, 13992, 14106, 14220, 14334, 14448, 14562, 14676, 14790, 14904, 15018, 15132
c0_ROM03_RT_No_Occlusion_400451	**12966**, 13080, 13194, 13308, 13422, 13536, 13650, 13764, 13878, 13992, 14106, 14220, 14334, 14448, 14562, 14676, 14790, 14904, 15018, 15132
c0_ROM04_RT_Occlusion_400275	**17746**, 17788, 17830, 17872, 17914, 17956, 17998, 18040, 18082, 18124, 18166, 18208, 18250, 18292, 18334, 18376, 18418, 18460, 18502, 18544
c0_ROM04_RT_Occlusion_400418	**17746**, 17788, 17830, 17872, 17914, 17956, 17998, 18040, 18082, 18124, 18166, 18208, 18250, 18292, 18334, 18376, 18418, 18460, 18502, 18544
c0_ROM05_RT_Wrist_ROM_400270	**19583**, 19607, 19631, 19655, 19679, 19703, 19727, 19751, 19775, 19799, 19823, 19847, 19871, 19895, 19919, 19943, 19967, 19991, 20015, 20039
c0_ROM05_RT_Wrist_ROM_400488	**19583**, 19607, 19631, 19655, 19679, 19703, 19727, 19751, 19775, 19799, 19823, 19847, 19871, 19895, 19919, 19943, 19967, 19991, 20015, 20039
c1_ROM03_RT_No_Occlusion_400456	**21646**, 21760, 21874, 21988, 22102, 22216, 22330, 22444, 22558, 22672, 22786, 22900, 23014, 23128, 23242, 23356, 23470, 23584, 23698, 23812
c1_ROM03_RT_No_Occlusion_400486	**21646**, 21760, 21874, 21988, 22102, 22216, 22330, 22444, 22558, 22672, 22786, 22900, 23014, 23128, 23242, 23356, 23470, 23584, 23698, 23812
c1_ROM04_RT_Occlusion_400266	**17468**, 17510, 17552, 17594, 17636, 17678, 17720, 17762, 17804, 17846, 17888, 17930, 17972, 18014, 18056, 18098, 18140, 18182, 18224, 18266
c1_ROM04_RT_Occlusion_400439	**17468**, 17510, 17552, 17594, 17636, 17678, 17720, 17762, 17804, 17846, 17888, 17930, 17972, 18014, 18056, 18098, 18140, 18182, 18224, 18266
c1_ROM05_RT_Wrist_ROM_400314	**24004**, 24028, 24052, 24076, 24100, 24124, 24148, 24172, 24196, 24220, 24244, 24268, 24292, 24316, 24340, 24364, 24388, 24412, 24436, 24460
c1_ROM05_RT_Wrist_ROM_400469	**24004**, 24028, 24052, 24076, 24100, 24124, 24148, 24172, 24196, 24220, 24244, 24268, 24292, 24316, 24340, 24364, 24388, 24412, 24436, 24460

Table 2. **InterHand2.6M sequences and image IDs in experiments**. “c0” denotes capture 0; The number at the end of each sequence name is the camera ID; The image IDs are the basenames of the images without the postfix. The image ID in bold is used for personalization.

7. Metrics

We use the peak signal-to-noise ratio (PSNR), structural simlariry index (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS) [13] between the rendered images using the personalized avatar, and the ground-truth RGB images to quantify the accuracy of reconstructed hand avatars.

8. Discussion and Future Work

Since we leverage a multi-subject prior for hand personalization, our method does not support hands with tattoo and accessories. Future work can consider capturing subjects with tattoo and accessories to bridge this gap. Our current method is memory intensive due to the use of hash grid [8]. Future work can explore more light-weight representations such as Gaussian Splatting.

References

[1] Xu Chen, Tianjian Jiang, Jie Song, Max Rietmann, Andreas Geiger, Michael J. Black, and Otmar Hilliges. Fast-SNARF: A fast deformer for articulated neural fields. *IEEE TPAMI*, 2023. 2

[2] Tero Karras. A style-based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*, 2019. 2

[3] Korrawe Karunratanakul, Sergey Prokudin, Otmar Hilliges, and Siyu Tang. HARP: Personalized hand reconstruction from a monocular rgb video. In *CVPR*, 2023. 2

[4] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 2

[5] Luca Medeiros. Lang Segment Anything. https://github.com/luca-medeiros/lang-segment-anything, 2023. 2

[6] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. InterHand2.6M: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image. In *ECCV*, pages 548–564, 2020. 2

[7] Gyeongsik Moon, Weipeng Xu, Rohan Joshi, Chenglei Wu, and Takaaki Shiratori. Authentic hand avatar from a phone scan via universal hand model. In *CVPR*, pages 2029–2038, 2024. 2

[8] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM TOG*, 41(4):1–15, 2022. 3

[9] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3D with transformers. In *CVPR*, 2024. 2

[10] Rolandos Alexandros Potamias, Stylianos Ploumpis, Stylianos Moschoglou, Vasileios Triantafyllou, and Stefanos Zafeiriou. Handy: Towards a high fidelity 3d hand shape and appearance model. In *CVPR*, pages 4670–4680, 2023. 2

[11] Radu Alexandru Rosu and Sven Behnke. Permutosdf: Fast multi-view reconstruction with implicit surfaces using permutohedral lattices. In *CVPR*, pages 8466–8475, 2023. 1

[12] Shaofei Wang, Bozidar Antic, Andreas Geiger, and Siyu Tang. IntrinsicAvatar: Physically based inverse rendering of dynamic humans from monocular videos via explicit ray tracing. In *CVPR*, pages 1877–1888, 2024. 2

[13] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 3

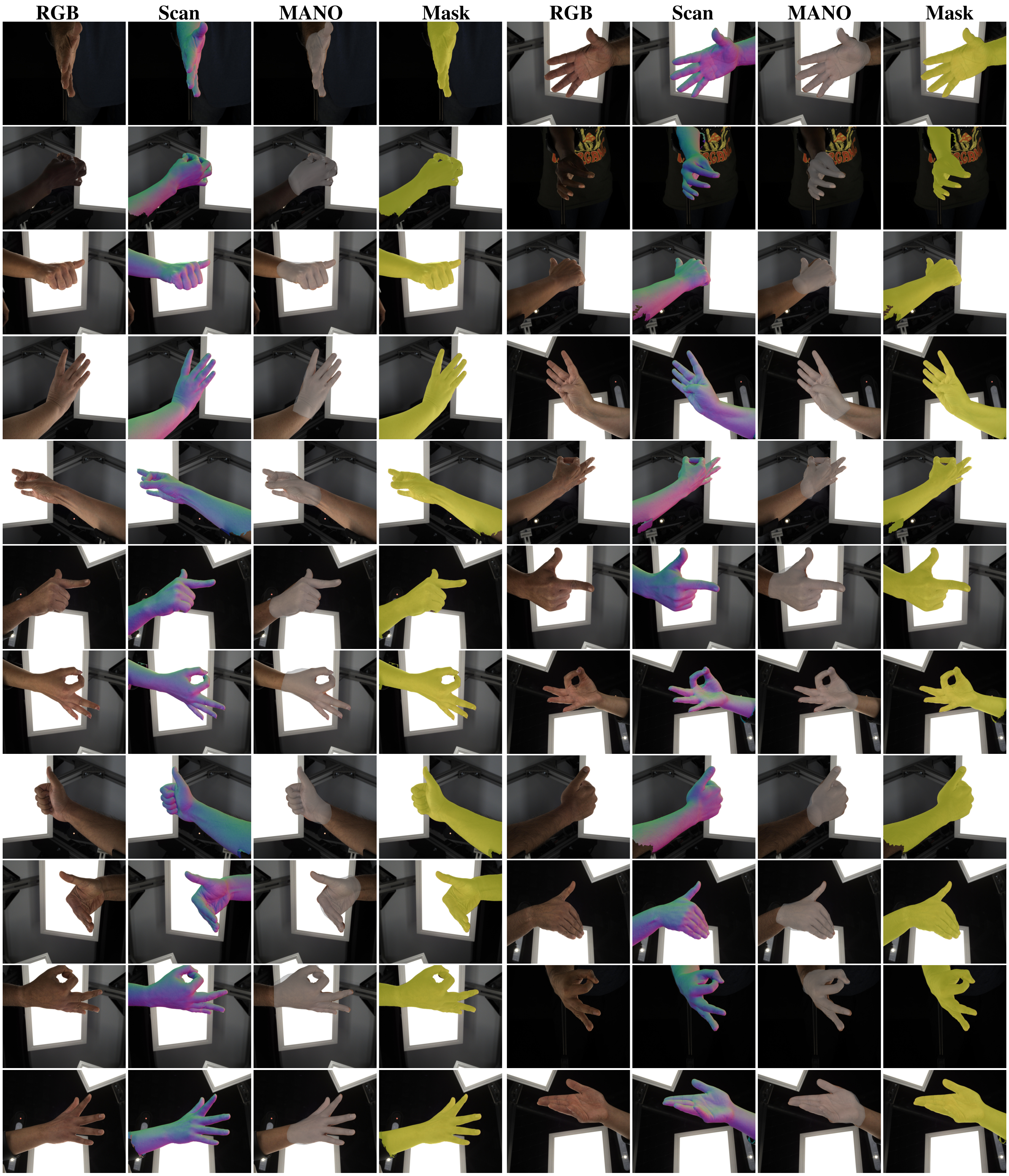

Figure 3. We present several examples from our PALM dataset under different hand shapes and skin tones. The images in this figure show RGB, normal, MANO registrations, and segmentation masks in that order.

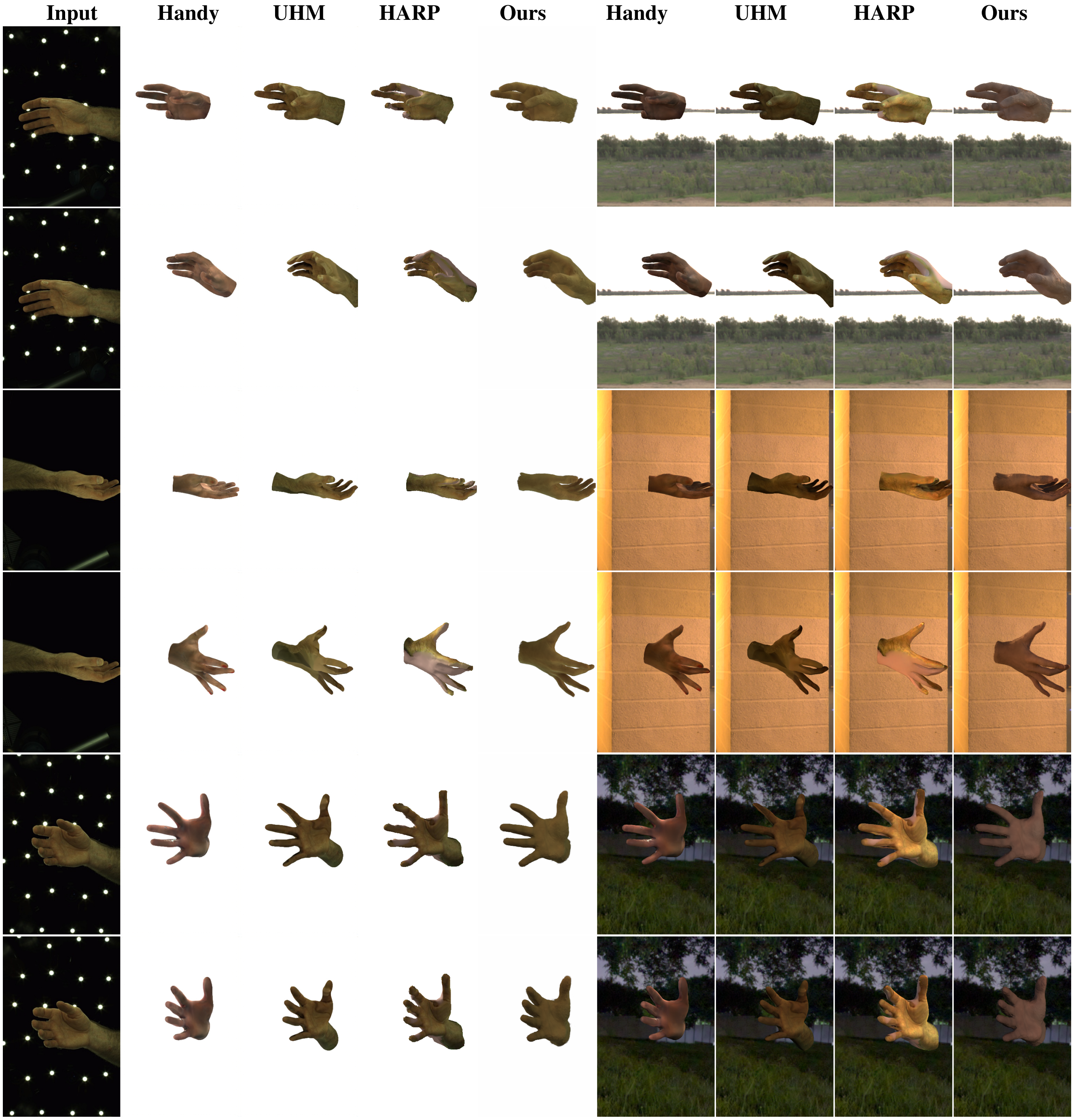

Figure 4. We provide qualitative comparison between the baseline methods and our method for single-image personalization on InterHand2.6M images. The first column shows the image used for personalization. Columns 2-5 represent reconstructed hand avatars rendered with training (input image) environment and novel pose. The last 4 columns represent the reconstructed hand avatars rendered with novel environment and poses.

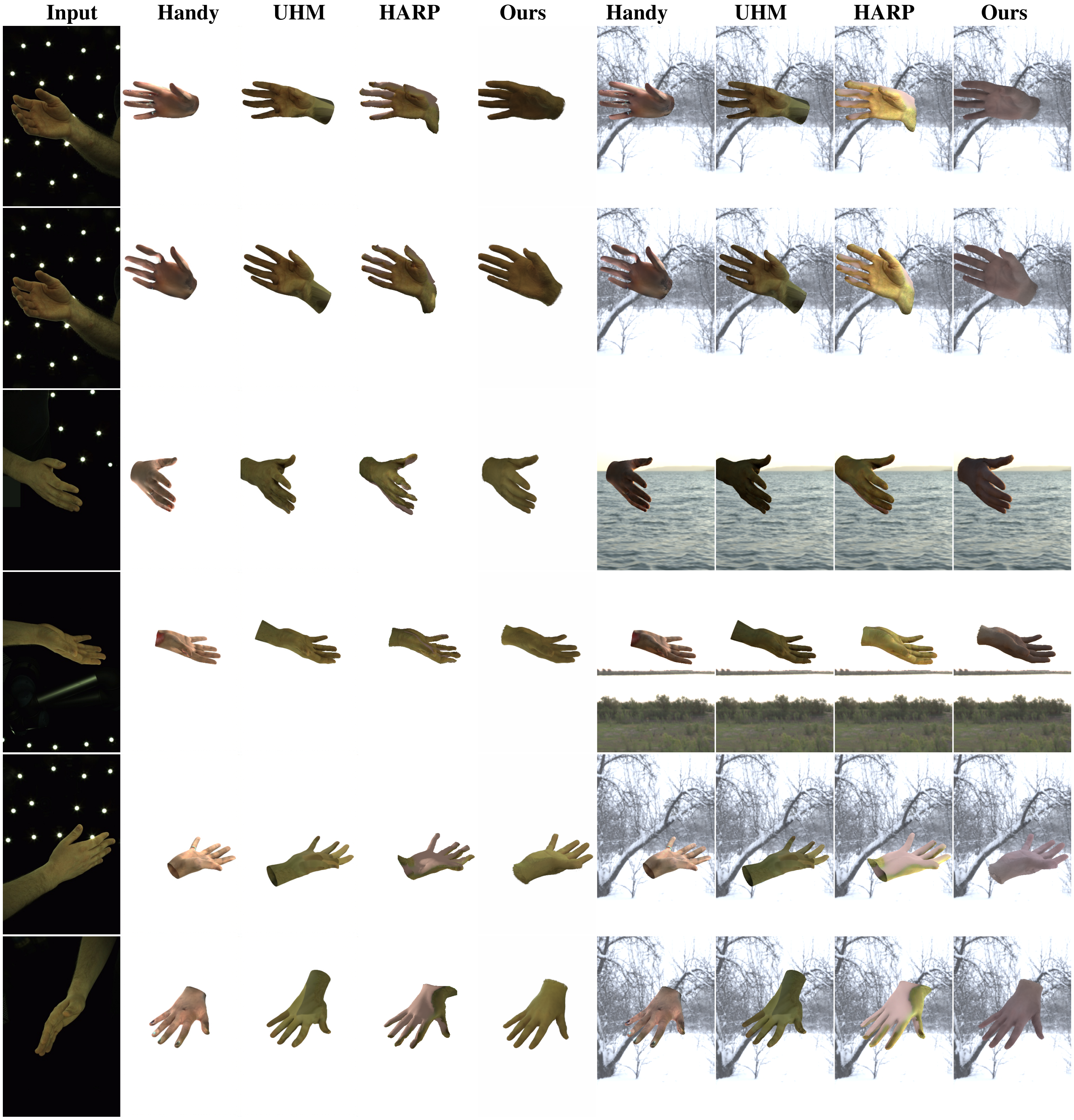

Figure 5. We provide qualitative comparison between the baseline methods and our method for single-image personalization on Interhand2.6M iamges. The first column shows the image used for personalization. Columns 2-5 represent reconstructed hand avatars rendered with training (input image) environment and novel pose. The last 4 columns represent the reconstructed hand avatars rendered with novel environment and poses.