Im2Hands: Learning Attentive Implicit Representation of Interacting Two-Hand Shapes — Supplementary Material

Jihyun Lee[1] Minhyuk Sung[1] Honggyu Choi[1] Tae-Kyun Kim[1,2]
[1] KAIST [2] Imperial College London

In this supplementary document, we first show more qualitative results of our method in Section S.1. We then show the results of our additional ablation study in Section S.2. Finally, we report our implementation details and experimental details in Section S.3 and S.4, respectively.

S.1. Additional Qualitative Results

S.1.1 Video Results

We provide the video results (https://youtu.be/3yNGSRz564A) of our method on image-based two-hand reconstruction in comparison to HALO [7] and IntagHand [9]. This video contains the reconstruction results on InterHand2.6M [11] test image sequences that are used in the main experiments in the paper. For our method and HALO, we use DIGIT [4] to generate keypoint inputs from single images. Please note that our method and the baseline methods [7, 9] are originally proposed for two-hand reconstruction from single images and/or keypoints, thus the shapes were reconstructed from *each frame independently*. One important future research direction would be to extend our model to additionally utilize temporal information for tracking applications.

S.1.2 Ablation Study

In Figure S1, we show the qualitative examples of our ablation study in Table 4 in the main paper. The shown examples are produced from single images, where we use the keypoints predicted by DIGIT [4] as inputs. In the figure, $\mathcal{I}$ produces two-hand shapes that do not look plausible due to the input errors from the *predicted* two-hand keypoints. $\mathcal{K}+\mathcal{I}$ generates more plausible shapes through input keypoint refinement performed by $\mathcal{K}$, however, it still does not properly model hand-to-hand interactions (e.g., finger contacts). Our full model, $\mathcal{K}+\mathcal{I}+\mathcal{R}$, reconstructs the most accurate shapes with higher hand-to-image and hand-to-hand coherency.

S.1.3 Additional Qualitative Comparison

In Figure S2 (*please see the next page*), we also show the additional examples of our qualitative comparison of interacting two-hand reconstruction on InterHand2.6M [11]. Compared to HALO [7] and IntagHand [9], Im2Hands can reconstruct interacting two-hand shapes with **a higher resolution, less penetrations, and better hand-to-image and hand-to-hand alignments**. The shown examples were produced from single image inputs to perform a fair comparison with IntagHand, where our method and HALO again leveraged DIGIT [4] as a keypoint estimator.

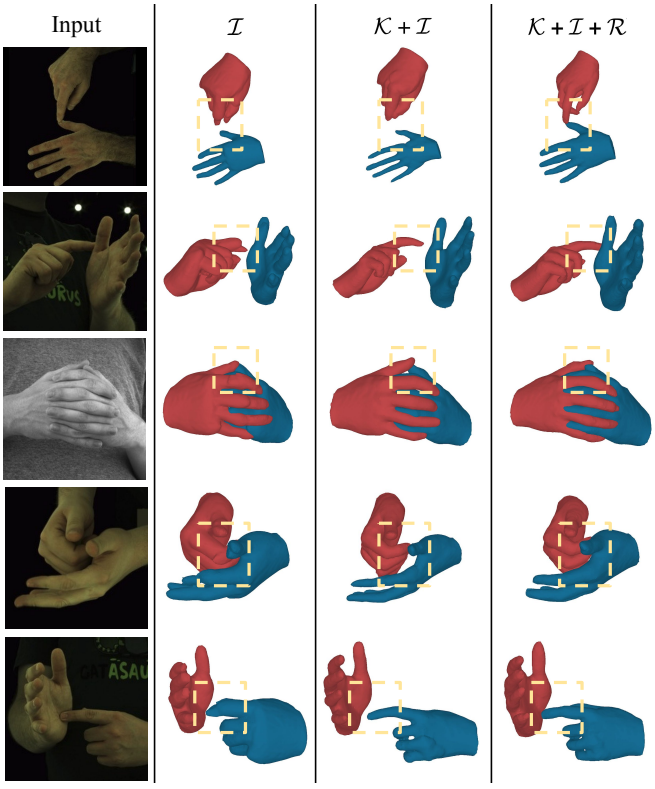

Figure S1. **Qualitative examples of ablation study on InterHand2.6M [11].** $\mathcal{I}$, $\mathcal{R}$ and $\mathcal{K}$ denotes Initial Hand Occupancy Network, Two-Hand Occupancy Refinement Network, and Input Keypoint Refinement Network, respectively.

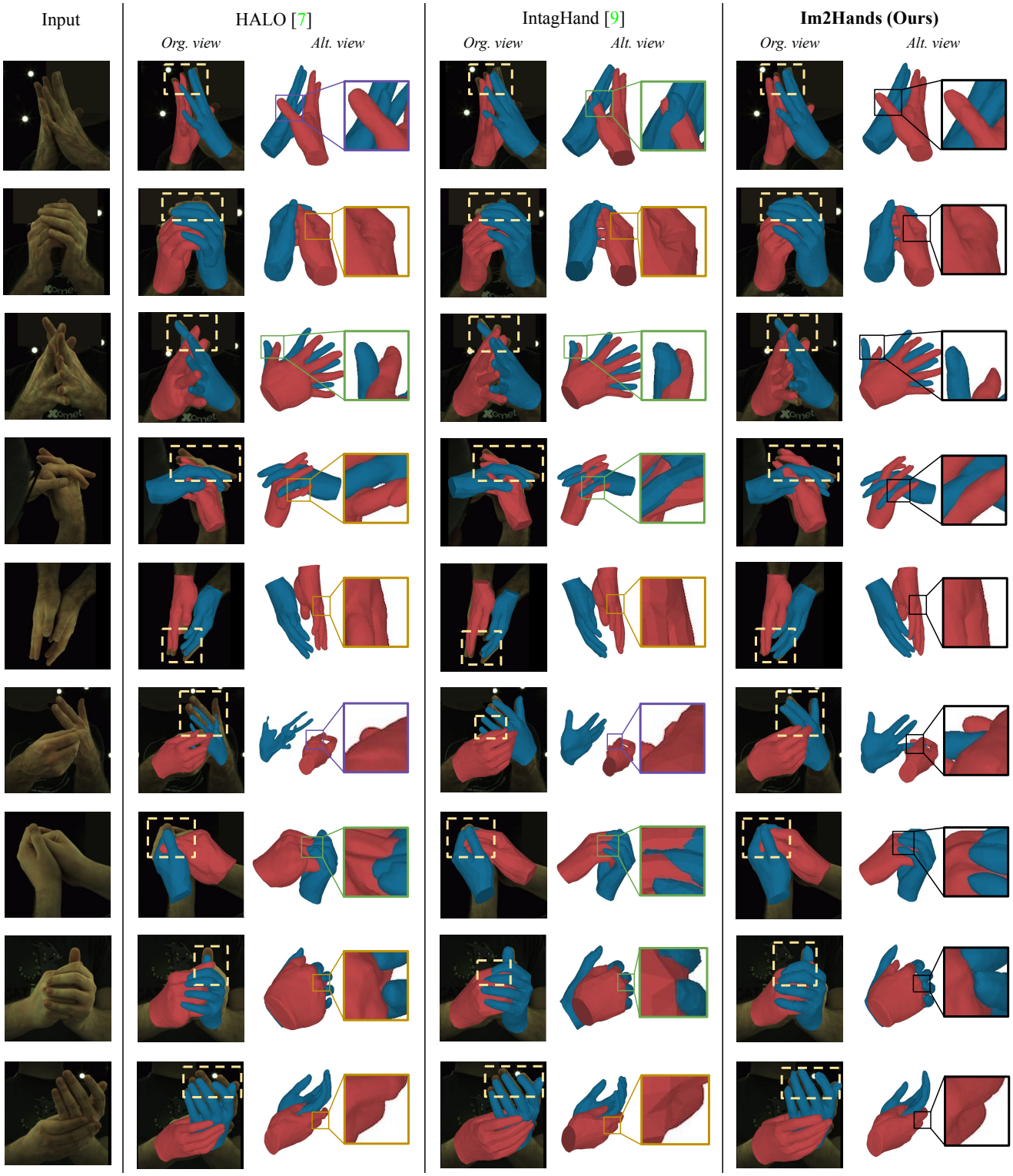

Figure S2. **Additional qualitative examples of *image-based* interacting two-hand reconstruction on InterHand2.6M [11].** We compare the results of our method with HALO [7] and IntagHand [9]. **Green boxes** show penetrations, **brown boxes** show non-smooth shapes, and **purple boxes** show shapes with bad image alignment. Our method produces two-hand shapes with **better hand-to-image and hand-to-hand coherency, less penetrations, and a higher resolution**.

S.2. Additional Ablation Study

We now report the quantitative results of more detailed ablation study. In what follows, we first explain the notations for each of the evaluated variations of Im2Hands.

- $\mathcal{I}$ **− ImageCond** denotes a variation where no image conditioning is used in $\mathcal{I}$, resulting in a model equivalent to HALO [8].
- $\mathcal{I}$ **− QueryImageAtt** denotes a variation where no query-image attention is used in $\mathcal{I}$. Instead, pixel-aligned features (e.g., PIFu [12]) are used to condition our initial occupancy on an input image.
- $\mathcal{I} + \mathcal{R}$ **− InitOccCond** denotes a variation where the initial occupancy probability estimated by $\mathcal{I}$ is not used to condition our two-hand refined occupancy estimation in $\mathcal{R}$.
- $\mathcal{I} + \mathcal{R}$ **− FeatureCloud** denotes a variation where the feature cloud conversion is not performed in $\mathcal{R}$.
- $\mathcal{I} + \mathcal{R}$ **− ContextLatent** denotes a variation where the context latent extraction is not performed in $\mathcal{R}$. Instead, global latent vector of each hand point cloud is used in the refined occupancy estimation.

Detailed Ablation Study With Ground Truth Keypoints. In Table S1, our quantitative results across the variations of Im2Hands are shown. Note that the *ground truth* keypoint inputs are used in these experiments. Our results demonstrate that each of the proposed model components contributes to more accurate two-hand shape estimation, and thus the proposed full model achieves the best performance.

Table S1. **Results of detailed ablation study using the *ground truth* 3D hand keypoints.** Experiments are performed on InterHand2.6M [11] dataset.

Method	IoU (%) ↑	CD (mm) ↓
$\mathcal{I}$ − ImageCond	74.7	2.62
$\mathcal{I}$ - QueryImageAtt	75.8	2.51
$\mathcal{I}$	77.2	2.32
$\mathcal{I}$ + $\mathcal{R}$ - InitOccCond	67.0	3.44
$\mathcal{I}$ + $\mathcal{R}$ - FeatureCloud	77.4	2.32
$\mathcal{I}$ + $\mathcal{R}$ - ContextLatent	77.6	2.31
Im2Hands ($\mathcal{I}$ + $\mathcal{R}$)	**77.8**	**2.30**

Detailed Ablation Study With Predicted Keypoints. In Table S2, we additionally show our results using the keypoints *predicted* by DIGIT [4] to examine the effectiveness of each of our components on more various settings. In the table, our full model is again shown to achieve the best performance. Considering the ablation study results using the ground truth (Table S1) and predicted (Table S2) keypoints together, one interesting observation is that $\mathcal{I}$ contributes more to the performance improvement when using the ground truth keypoints input, while $\mathcal{R}$ contributes more to it when using the *predicted* keypoints input. It reveals that *both* $\mathcal{I}$ and $\mathcal{R}$ are essential to enable robust two-hand shape reconstruction given input keypoints with various degrees of noise.

Table S2. **Results of detailed ablation study using the 3D hand keypoints *predicted* by DIGIT [4].** Experiments are performed on InterHand2.6M [11] dataset.

Method	IoU (%) ↑	CD (mm) ↓
$\mathcal{K}$ + $\mathcal{I}$ − ImageCond	53.0	5.63
$\mathcal{K}$ + $\mathcal{I}$ - QueryImageAtt	53.9	5.47
$\mathcal{K}$ + $\mathcal{I}$	55.4	5.36
$\mathcal{K}$ + $\mathcal{I}$ + $\mathcal{R}$ - InitOccCond	55.1	5.18
$\mathcal{K}$ + $\mathcal{I}$ + $\mathcal{R}$ - FeatureCloud	58.3	4.78
$\mathcal{K}$ + $\mathcal{I}$ + $\mathcal{R}$ - ContextLatent	58.4	4.79
Im2Hands ($\mathcal{K}$ + $\mathcal{I}$ + $\mathcal{R}$)	**59.4**	**4.75**

Different Point Sampling Densities. For our two-hand occupancy refinement, we represent each initial hand shape with 512 farthest points on the surface. In Table S3, we also show the results with different point sampling densities. Our model performance (in IoU) is not affected much by the point density, while the training time is reduced when the number of points is decreased – showing the effectiveness of our method. Our model achieves state-of-the-art results even with the sparse 256 points.

Table S3. **Training time (second per iteration) and IoU with varying point sampling density**. Time is obtained as an average of 1K measurements. For measuring IoU, we used the ground truth keypoint inputs.

# of sampled points	Training time (s)	IoU (%)
256	**1.04**	77.4
512 (Ours)	1.74	**77.8**
1024	3.15	77.6

S.3. Implementation Details

In this section, we report more details of our implementation that could not be included in the main paper due to the space limit. Note that more implementation details are also available through our code[1].

[1] https://github.com/jyunlee/Im2Hands

S.3.1 Network Architecture

Initial Hand Occupancy Network ($\mathcal{I}$). For the query positional embedding module used to compute our query-image attention (PosEnc in Equation 3), we use a shared MLP composed of two fully-connected layers, each of them followed by ReLU activation and dropout with a rate of 0.01. For the image encoder-decoder (ImgEnc in Equation 3), we use a ResNet-50 [6] architecture as an encoder and a CNN composed of four deconvolutional layers as a decoder. For the multi-headed self-attention module (MSA in Equation 3), we extract features of 8×8 image patches using an encoder of Vision Transformer [3] and apply self-attention with two attention heads. The resulting features extracted by query-image attention are concatenated with the features extracted by HALO [7] encoder after the first layer in the part occupancy functions of HALO. For the architecture of HALO encoder and part occupancy functions, we follow the design of HALO. We thus refer the reader to [7] for more details.

Two-Hand Occupancy Refinement Network ($\mathcal{R}$). For iso-surface point extraction, we evaluate the occupancy probabilities at uniformly sampled query points in 3D space and collect the query points that are estimated to be on the surface. We then apply farthest point sampling (FPS) to obtain 512 points to create each of the hand point clouds (i.e., $\mathcal{P}_l$ and $\mathcal{P}_r$). For feature cloud conversion, we use the same image encoder-decoder used in $\mathcal{I}$. For point cloud encoder (PCEnc in Equation 5), we use the same encoder architecture as in AIR-Net [5] except for the input point dimension, which is increased due to our feature cloud conversion procedure. We use a shared PCEnc for both sides of hand feature clouds, but we distinguish each side by concatenating a binary label – $[1, 0]$ for left hand and $[0, 1]$ for right hand – to each of the point features. For our context encoder ($\mathrm{ContextEnc}$ in Equation 6), we concatenate the inputs (z_l, z_r, z_I) and apply an MLP composed of two fully-connected layers, each of them followed by ReLU activation. For our point cloud decoder that estimates the refined occupancy (PCDec in Equation 7), we concatenate the query coordinate x with the initial occupancy probability at x and feed the resulting query vector to the decoder of AIR-Net along with $\mathcal{A}_s$ and z_c. For more details on the architecture of PCEnc and PCDec, please refer to [5].

Input Keypoint Refinement Network ($\mathcal{K}$). For KptEnc, we use (1) an embedding layer to embed the index of each keypoint and (2) an MLP composed of two fully-connected layers to encode the coordinate of each keypoint. We then concatenate the index feature and the coordinate feature for each of the keypoints and set them as node features in a two-hand skeleton graph. We then feed the skeleton graph to a graph convolutional network (GCN) composed of four layers with residual connections. The updated node features are directly used for multi-headed self-attention (MSA) between the patch-wise image features, which are extracted by the same Vision Transformer [3] encoder used in $\mathcal{I}$. The updated node features are then fed to an output keypoint coordinate regressor, which is an MLP composed of two fully-connected layers – each of them followed by ReLU activation and dropout of a rate of 0.01.

S.3.2 Training Details

For $\mathcal{I}$ and $\mathcal{R}$, we train each of the networks for 10 epochs with a batch size of 8. We use an Adam optimizer with an initial learning rate of $1e-4$, betas of $[0.9, 0.999]$, an epsilon of $1e-8$, and a weight decay parameter of $1e-5$. We additionally use a learning rate scheduler to decay the learning rate by 0.2 every 5000 training steps. For the loss function to train $\mathcal{R}$, we use a weighted sum of the proposed loss terms (i.e., occupancy loss and penetration loss), with the weight values set as 1 and 0.001, respectively. Other training details (e.g., training query sampling) are the same as in the original HALO framework (please refer to [7] for more detail). For $\mathcal{K}$, we train the network for 30 epochs with a batch size of 32. We use an Adam optimizer with an initial learning rate of $1e-4$ with a scheduler to decay the learning rate by 0.3 every 5000 training steps.

S.4. Detailed Experimental Setups

S.4.1 Metric Computation

For Im2Hands and HALO [8], we extract the reconstructed meshes by evaluating occupancy probabilities at uniformly sampled query points in 3D space and applying Marching Cubes [10]. We then compute our metrics (i.e., mean Intersection over Union and Chamfer L1-Distance) after mid-joint alignment of each hand. Note that the existing works [9, 14] use Mean Per Vertex Position Error (MPVPE) as an evaluation metric, which assumes one-to-one vertex correspondence between the ground truth and the predicted meshes. As our method does not assume such vertex correspondence, we use mean Intersection over Union and Chamfer L1-Distance as our evaluation metrics as in other implicit function-based reconstruction methods [2, 8].

S.4.2 Setups for Generalizability Test

In this section, we report more details of our setups for the generalizability test (Section 4.4 in the paper). For pre-processing the two-hand frames in RGB2Hands [13] and EgoHands [1] datasets, we compute a coarse foreground mask obtained by thresholding the depth map provided by [1, 13] to mask out the approximate background region. We then directly apply Im2Hands trained only on InterHand2.6M [11] to evaluate its generalization ability to unseen hand shapes and appearances. Other experimental settings are the same as in our main experiments on InterHand2.6M dataset.

References

[1] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *ICCV*, 2015. 4

[2] Boyang Deng, John P Lewis, Timothy Jeruzalski, Gerard Pons-Moll, Geoffrey Hinton, Mohammad Norouzi, and Andrea Tagliasacchi. Nasa neural articulated shape approximation. In *ECCV*, 2020. 4

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 4

[4] Zicong Fan, Adrian Spurr, Muhammed Kocabas, Siyu Tang, Michael J Black, and Otmar Hilliges. Learning to disambiguate strongly interacting hands via probabilistic per-pixel part segmentation. In *3DV*, 2021. 1, 3

[5] Simon Giebenhain and Bastian Goldlücke. Air-nets: An attention-based framework for locally conditioned implicit representations. In *3DV*, 2021. 4

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[7] Korrawe Karunratanakul, Adrian Spurr, Zicong Fan, Otmar Hilliges, and Siyu Tang. A skeleton-driven neural occupancy representation for articulated hands. In *3DV*, 2021. 1, 2, 4

[8] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael J Black, Krikamol Muandet, and Siyu Tang. Grasping field: Learning implicit representations for human grasps. In *3DV*, 2020. 3, 4

[9] Mengcheng Li, Liang An, Hongwen Zhang, Lianpeng Wu, Feng Chen, Tao Yu, and Yebin Liu. Interacting attention graph for single image two-hand reconstruction. In *CVPR*, 2022. 1, 2, 4

[10] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. 1987. 4

[11] 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*, 2020. 1, 2, 3, 4

[12] 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*, 2019. 3

[13] Jiayi Wang, Franziska Mueller, Florian Bernard, Suzanne Sorli, Oleksandr Sotnychenko, Neng Qian, Miguel A Otaduy, Dan Casas, and Christian Theobalt. Rgb2hands: real-time tracking of 3d hand interactions from monocular rgb video. *ACM TOG*, 2020. 4

[14] Baowen Zhang, Yangang Wang, Xiaoming Deng, Yinda Zhang, Ping Tan, Cuixia Ma, and Hongan Wang. Interacting two-hand 3d pose and shape reconstruction from single color image. In *ICCV*, 2021. 4