# BiTT: Bi-directional Texture Reconstruction of Interacting Two Hands from a Single Image

Minje Kim[1] Tae-Kyun Kim[1,2]
[1] KAIST [2] Imperial College London

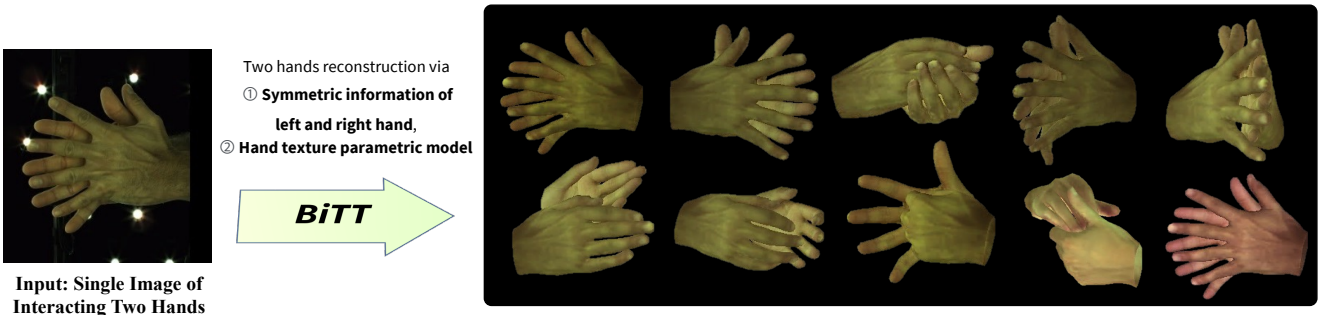


Figure 1. Taking a single image input, our method renders the personalized texture of two hands at novel views, poses, and light conditions.

## Abstract

*Creating personalized hand avatars is important to offer a realistic experience to users on AR / VR platforms. While most prior studies focused on reconstructing 3D hand shapes, some recent work has tackled the reconstruction of hand textures on top of shapes. However, these methods are often limited to capturing pixels on the visible side of a hand, requiring diverse views of the hand in a video or multiple images as input. In this paper, we propose a novel method, BiTT(Bi-directional Texture reconstruction of Two hands), which is the first end-to-end trainable method for relightable, pose-free texture reconstruction of two interacting hands taking only a single RGB image, by three novel components: 1) bi-directional (left ↔ right) texture reconstruction using the texture symmetry of left / right hands, 2) utilizing a texture parametric model for hand texture recovery, and 3) the overall coarse-to-fine stage pipeline for reconstructing personalized texture of two interacting hands. BiTT first estimates the scene light condition and albedo image from an input image, then reconstructs the texture of both hands through the texture parametric model and bi-directional texture reconstructor. In experiments using InterHand2.6M and RGB2Hands datasets, our method significantly outperforms state-of-the-art hand texture reconstruction methods quantitatively and qualitatively. The code is available at https://github.com/yunminjin2/BiTT.*

## 1. Introduction

3D human reconstruction has been studied in various areas. With the increasing usage of human-computer interaction, virtual reality (VR), and augmented reality (AR), reconstruction of human parts, including the full body, face, and hand, has been intensively studied for years. In particular, hand pose estimation, shape, and texture reconstruction are essential tasks for AR/VR interfaces. 3D hand reconstruction is still a challenging task due to the highly varied poses and shapes of hands. Previous works [3, 4, 9, 14, 23, 45] focused on estimating the 3D pose and shape of a single hand. A single hand reconstruction has been recently extended to two interacting hands [18, 19, 21, 49] and hand-object interaction [2, 8, 15, 16, 46] scenarios.

Learning the appearance of objects and humans is currently in active research for realistic reconstruction. NeRF [27] represents objects/scenes by a neural radiance field based on volume rendering. Appearance reconstruction of clothed human bodies [1, 12, 22, 30, 44] and faces [20, 25, 32, 38, 48] has been intensively studied compared to hands. For learning hand appearance, LISA [7] used a radiance field to learn the shape and color appearance from

multi-view images. In the latest works [5, 7, 10, 17, 29] to reconstruct the hand appearance, they take multi-view images or a monocular video as input to learn the texture of mostly single hands [5, 7, 10, 17, 29]. HandAvatar [5] and HARP [17] render relightable hand appearance by estimating the albedo of a single hand.

3D reconstruction from a single image is also another challenging task. Non-visible side of an object should be estimated with given a single image for full 3D reconstruction. Self-Supervised 3D Mesh Reconstruction (SMR) [11] estimates 3D meshes with texture from a single image in a self-supervised manner. Wu *et al.* [43] reconstructed vase artifacts into 3D mesh with environment lighting, shiny material, and texture albedo. Other works [9, 14, 23, 45] also focus on constructing symmetric objects or single-type objects. Human hands, however, exhibit non-symmetric features such as the palm and back. Given the information on one side of the hand, the other side of the hand texture should be estimated to fully reconstruct the hand. S2Hand [6] and AMVUR [13] presented a method to reconstruct both the appearance and geometry of a single hand from a single image. Nevertheless, their appearance is in blurred textures, omitting detail texture appearance.

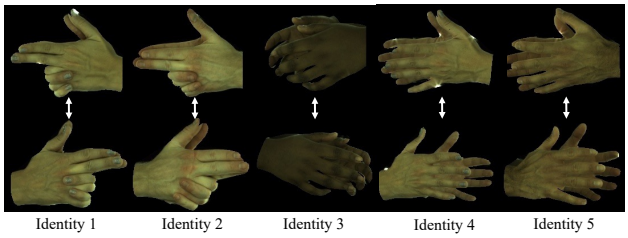


Figure 2. Symmetrical hand textures of different identities are shown, taken from pairs of diametrical camera views of InterHand2.6M [28].

In this paper, we propose a novel approach that exploits texture symmetry of left and right hands through the bi-directional reconstruction of two hand textures from a single image (see Fig. 2). Our method is trained per scene with only one single image, taking visible texture information from both hands and the parametric model of hand texture to reconstruct realistic hand appearances. Given an input image and a mesh of hands, our method predicts lights and albedo image (please refer to Sec. 3.1). In the coarse stage, our model generates full hand textures with estimated vectors using HTML [34], the hand texture parametric model. With the estimated albedo image and coarse stage estimated texture map, the bi-directional texture reconstructor (BTR) yields the UV maps of both hand textures by utilizing the feature maps of a left and right hand. The proposed BiTT method can render both hands with fully controllable light conditions, poses, and camera views. We evaluated the method using the InterHand2.6M [28], and RGB2Hands [40] dataset and achieved high-fidelity appearances compared to state-of-the-art methods. We present qualitative results where the rendered images realistically capture personalized hand textures (e.g. wrinkles, veins, nails). Fig. 1 shows that our method is capable of controlling light conditions, camera views, and hand poses. To the best of our knowledge, this is the first method to reconstruct both hands with textures from a given single image input.

In summary, our main contributions are as follows: 1) we introduce a novel framework BiTT, the first method for rendering two interacting hands from a single image. 2) We propose the bi-directional texture reconstruction, exploiting the texture symmetry of left and right hands. 3) We introduce a way to use the texture parametric model for recovering invisible texture. 4) We demonstrate that our framework is an end-to-end trainable for photorealistic two-hand avatars with controllable poses, views, and light conditions.

## 2. Related Work

In hand appearance modeling, NIMBLE [24] learns factorization of albedo, specular, and normal maps from high-definition hand textures. S2HAND [6] estimates camera poses, colored meshes, and lighting conditions in a simultaneous way, but its rendering quality does not demonstrate detailed (or personalized) hand appearances. AMVUR [13] reconstructs a hand using attention-based mesh vertices and the occlusion-aware texture regression model. However, its per-vertex texture reconstruction is not adaptable to the reconstruction of realistic hand textures due to its low resolution. In addition, the reconstructed hand textures contain background colors since the method does not consider geometric misalignment noise. LISA [7] learns an implicit color field with implicit shape representation but still lacks detailed hand textures despite multi-view inputs.

**Neural-Radiance-Field based Representation.** In recent years, the application of hand representation by the neural radiance field has been studied extensively. HandAvatar [5] has improved the hand appearance with the occupancy field and predicted self-occlusion shadows. However, it requires a large amount of training data and time for a new instance of hand. HandNeRF [10], Livehand [29] extracts hand texture from multi-view image sequence with volume rendering in neural radiance field for hand reconstruction. To develop subject-specific shape and appearance using NeRF-based methods such as HandAvatar, HandNeRF, and LiveHand, a substantial volume of images within a fixed light condition, up to thousands of images for each single sequence, is typically needed for training. Since these methods are integrated into an implicit space, they require additional steps for mesh extraction to further applications. They also face difficulties in controlling illumination, including rendering shadows and editing textures.

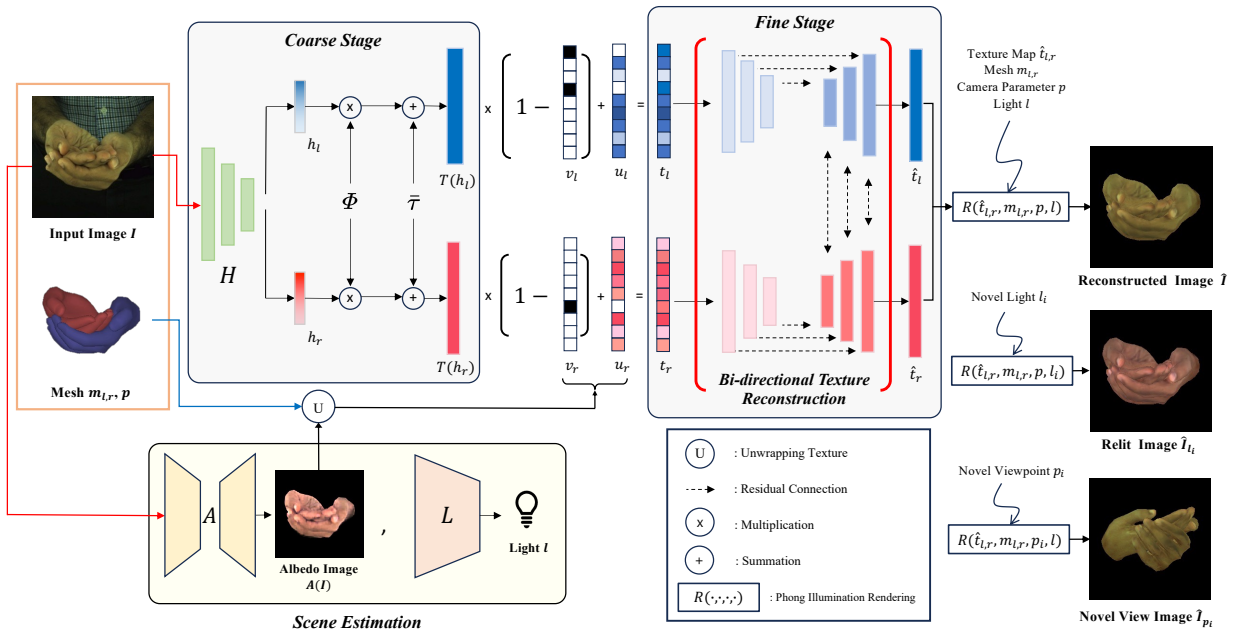


Figure 3. The architecture of BiTT. Our method consists of three steps: (1) scene estimation, (2) coarse stage, and (3) fine stage estimation. The scene estimation understands the scene by predicting the albedo image and lighting conditions with a given input image. Full detailed textures of both hands are reconstructed from the single image input. The hand texture parametric model is adopted in the coarse stage, then the bi-directional texture reconstruction refines the personalized hand textures by the texture symmetry of left-right-hands. Finally, we render both hands with Phong Illumination [33].

**UV Texture Map based Representation.** Unlike the implicit function-based methods, HARP [17] reconstructed a single hand by mesh rendering with a UV texture map. HARP reconstructs a personalized single hand using the UV texture map, normal map, and self-occlusion shadow. HARP renders detailed hand textures including out-of-distribution appearance like tattoos and accessories. However, the HARP method assumes the texture color into a specific value, and as a result, the occluded part of the texture tends to be a fixed value, missing out the detailed texture.

**Positioning BiTT to w.r.t related works.** Two-hand texture rendering based on UV maps has not been explored to the best of our knowledge. In implicit space, HandNeRF [10] reconstructed two hands from a multi-view image sequence. Reconstruction of interacting two hands from a single image is a challenging task, as we have to lift a 2D image to achieve accurate 3D hand representation. Also, two hands encounter a higher rate of occlusion compared to a single hand, thus restoring occluded texture is another challenge. Whereas using multi-views or a video sequence as input helps reconstruct two interacting hands, taking a single image input poses more significant challenges. However, two hands convey more information than a single hand, which enables us to reconstruct both hands realistically within a single image. In this work, we propose a novel method that exploits the two-hand symmetric texture information and employs the hand texture parametric model as prior. The proposed method demonstrates that only a single image is enough to reconstruct a realistic two-hand avatar in contrast to the prior works [5, 10, 17, 29].

## 3. Methods

We propose a texture reconstruction model for two interacting hands with a single image. Fig. 3 shows the architecture of BiTT. Given a single RGB image $I$ of interacting two hands, we reconstruct realistic personalized textures for a two-hand avatar. BiTT is composed of three steps: the scene estimation, the coarse stage based on the parametric model, and the fine stage composed with the bi-directional texture reconstructor.

For reconstructing the full texture of both hands from a single image, our method relies on utilizing the symmetry between the left and right hands. While the disparity between the left and right hand texture is assumed to be not significant (e.g. see Fig. 2), it is beneficial to leverage the symmetrical data to reconstruct detailed textures even on occluded hands. For those pixels not visible on both hands, the texture parametric model is adopted, and its estimation is further refined.

### 3.1. Scene Estimation

Scene estimation involves estimating the environment, including the light and albedo of hands in the input image. To estimate the environment, our light network $L$ estimates the light parameter $l$ which is composed of the ambient light color, diffuse, specular, and direction. Furthermore, we also predict an albedo image through our albedo network $A$, which represents the surface reflectance of objects using the Lambertian surface. Details of the light and albedo networks are found in the supplementary material.

### 3.2. Coarse Stage

Since 2D image lacks the information to reconstruct 3D objects, non-symmetric object reconstruction [1, 6, 26, 31, 37, 39, 44] suffers from blurry textures. Several major works [5, 7, 17, 27, 29] overcome the lack of information by multi-view images or a monocular video as input. Instead of increasing the input information, we augment data through the parametric model. The coarse stage is based on the hand texture parametric model (HTML) [34], which encodes the hand texture through PCA algorithm and is able to create a full texture of hand from a single image.

**Background on HTML [34].** HTML is a hand texture parametric model that can create a full-texture UV map with a given vector. HTML scanned 51 hands and aligned them to a canonical space with MANO [36] model fitting. After the MANO fitting, texture mapping is done with manually defined UV coordinates. Finally, a parametric model $T$ is created by PCA on vectors $\tau_i \in \mathbb{R}^{618990}$ with the collection of 2D texture maps where 618,990 is a total number of 206,330 pixels in texture map with RGB channels. Given the covariance matrix $C \in \mathbb{R}^{618990\times618990} = \frac{1}{n-1}\sum_{i=1}^{n}(\tau_i-\bar{\tau})(\tau_i-\bar{\tau})^\top$, where $\bar{\tau} = \frac{1}{n}\sum_{i=1}^{n}\tau_i$, the principal components $\Phi \in \mathbb{R}^{618990\times101}$ are obtained by singular value decomposition of $C = \Phi\Sigma\Phi^\top$, where $\Sigma \in \mathbb{R}^{101\times101}$ is a diagonal matrix. With the principal components $\Phi$, we get a full texture eigenvalue $T(\alpha) \in \mathbb{R}^{618990}$ for a given parameter vector $\alpha \in \mathbb{R}^{101}$ with $T(\alpha) = \bar{\tau}+\Phi\alpha$. For more details on HTML, please refer to [34].

**Hand Texture Parametric Model in Coarse Stage.** The HTML [34] network $H$ in Fig. 3 is an encoder that estimates both the hand parameter vectors from an input image $I$. Given the left-hand parameter vector $h_l$ and the right-hand parameter vector $h_r$, we can obtain the full texture eigenvalues $T(h_l), T(h_r)$. It can be formally defined as:

$$T(h_i) = \bar{\tau} + \Phi h_i, \text{ where } h_l, h_r = H(I), i = l, r \quad (1)$$

$$\hat{I}_{coarse} = \{R(T(h_i), m_i, p, l)\}_{i=l,r} \quad (2)$$

$I$ is an input image, $R$ is the differentiable renderer based on Phong model [33], and $\hat{I}_{coarse}$ is the reconstructed image from the coarse stage. With the texture vector $T(h_i)$ and meshes $m_i$, $R$ renders two hand meshes with texture on the 2D space at a camera viewpoint $p$ and light condition $l$.

### 3.3. Fine Stage

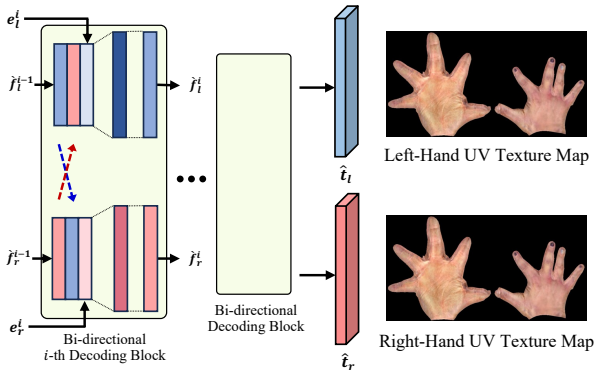


Figure 4. Detailed architecture of the decoding layer in the bi-directional texture reconstruction.

To render more realistic and personalized textures, we use the symmetric features of left/right hands rather than using features independently. We propose a novel bi-directional texture reconstructor (BTR) to efficiently use the symmetric texture features of both hands. BTR reconstructs the full texture from visible pixels in the albedo image $A(I)$. We create a visible UV texture map and texture vector by unwrapping hand textures as we know the mapping between the UV texture map and image pixels, as well as the mapping between the UV texture map and the texture vector. With the visible texture vector for each hand unwrapped from the albedo image $A(I)$, denoted as $u_l, u_r$ for each hand, and $v_l, v_r$ which are visible mask texture vectors for each hand, we synthesize texture vector $t_l, t_r$ defined as:

$$t_i = T(h_i)(1 - v_i) + u_i, \text{ where } i = l, r \quad (3)$$

BiTT, then, generates full both hand textures through the BTR.

In addition, with the visible albedo texture $u_l, u_r$, we render $\hat{I}_{albedo}$ with the estimated light $l$ to use in loss function:

$$\hat{I}_{albedo} = \{R(u_i, m_i, p, l)\}_{i=l,r} \quad (4)$$

**Bi-directional Texture Reconstruction.** Given the synthesized texture vectors $t_l, t_r$, the bi-directional texture reconstructor (BTR) encodes each texture vector, denoted as $e_l, e_r$. After encoding each hand texture vector, the decoder in BTR decodes the embedded feature to create a full texture vector using skip connections. Fig. 4 describes the detailed

description of the BTR decoder. In each bi-directional decoding block, the embedded features of the same hand are concatenated with a skip connection. We also concatenate the other hand texture embedded features in a bi-directional way to use the symmetric information of two hands. Afterward, we obtain the decoded variable from the concatenated feature at the i-th level, denoted as $\grave{f}^i$. The decoded features $\grave{f}_l^i, \grave{f}_r^i$, which means the $i$-th level of the left hand and the right hand, respectively, are defined as:

$$\grave{f}_l^i = \sigma(\mathcal{N}([e_l^i, \grave{f}_l^{i-1}, \grave{f}_r^{i-1}])) \tag{5}$$

$$\grave{f}_r^i = \sigma(\mathcal{N}([e_r^i, \grave{f}_r^{i-1}, \grave{f}_l^{i-1}])) \tag{6}$$

where $\sigma$ denotes a ReLU activation function, $\mathcal{N}$ is a convolution neural network (CNN), and [] denotes the channel concatenation operator.

After the decoding, we obtain the full texture vectors $\hat{t}_l, \hat{t_r} \in \mathbb{R}^{3\times 206330}$ for each hand. Thus, our network is formulated as:

$$\hat{I} = BiTT(I) = \left\{R(\hat{t}_i, m_i, p, l)\right\}_{i=l,r} \tag{7}$$

where $\hat{I}$ is the final rendered image of reconstructed both hands.

### 3.4. Loss Functions

The loss functions we use to train our model are as follows.

**Reconstruction Loss.** As our method aims to represent the realistic appearance of the input image, we included the reconstruction loss $\mathcal{L}_{rec}$ to measure the similarity between the input image $I$ and three distinct rendered images: rendered image from fine stage ($\hat{I}$), rendered image from coarse stage ($\hat{I}_{coarse}$), and the rendered image with albedo visible texture ($\hat{I}_{albedo}$). The reconstruction loss $\mathcal{L}_{rec}$ is defined as:

$$\mathcal{L}_{rec} = \lambda_{rec}\|(I - \hat{I})\|_1 + \lambda_{rec}^{coarse}\|I - \hat{I}_{coarse}\|_1 + \lambda_{rec}^{albedo}\|I - \hat{I}_{albedo}\|_1 \tag{8}$$

**Reconstruction Loss on Non-visible Pixels.** Visible information itself is not enough to accurately recreate the complete texture of hands. Even when symmetrical aspects are taken into account, it is still not sufficient to cover the entire hand texture. Thereby, we resort to the full texture obtained in the coarse stage for those pixels not observed in either of the hands. We measure the L1 loss between the coarse stage estimated hand texture and fine stage estimated hand texture with the invisible map mask. $\mathcal{L}_{nv}$ is defined as:

$$\mathcal{L}_{nv} = \sum_{i=l,r} \|((T(h_i) - \hat{t}_i)(1 - v_i)\|_1 \tag{9}$$

where $T(h_i)$ is the reconstructed hand texture in the coarse stage, $\hat{t}_i$ is the reconstructed hand texture in the fine stage for each hand. The visible mask of the hand texture is denoted as $v_i$ and thus, $1 - v_i$ represents the invisible mask of the texture. These textures are simultaneously refined with those of visible and symmetric texture reconstruction and consistency losses.

**Albedo Consistency Loss.** The albedo image should be obtained independent of lighting conditions. To obtain an accurate albedo image, we augment individually rendered images $\hat{I}_{coarse,\hat{l}_i}$ with different lighting conditions $\hat{l}_i$ and obtain the albedo images through the albedo network $A$. Our new albedo loss function $\mathcal{L}_{alb}$ calculates the L1 loss between albedo images from different lights. Thus, the albedo loss term $\mathcal{L}_{alb}$ is defined as:

$$\mathcal{L}_{alb} = \sum_{i=1}^{n}\sum_{j=i+1}^{n} [\|A(\hat{I}_{coarse,\hat{l}_i}) - A(\hat{I}_{coarse,\hat{l}_j})\|_1]. \tag{10}$$

In our experiment, we rendered three different lights in total: a reconstructed light, a light from a different direction, and a light with a different color to make the albedo network $A$ estimate the albedo image consistently even in the different lighting conditions.

**Symmetric Loss.** For learning the symmetric feature of two hands, we apply a symmetric loss term $\mathcal{L}_{sym}$ which is the L1 loss between the left and right hands. The $\mathcal{L}_{sym}$ is defined as:

$$\mathcal{L}_{sym} = \lambda_{sym}\|(\hat{t}_l - \hat{t}_r)\|_1 \tag{11}$$

**Total Loss.** In summary, our loss function is defined as:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_{nv}\mathcal{L}_{nv} + \lambda_{alb}\mathcal{L}_{alb} + \lambda_{sym}\mathcal{L}_{sym} \tag{12}$$

where $\lambda_{rec}^{coarse} = 0.8$, $\lambda_{rec}^{albedo} = 0.4$, $\lambda_{nv} = 0.2$, $\lambda_{alb} = 0.2$, $\lambda_{sym} = 0.3$ are used to train our model and fixed for all experiments.

## 4. Experiments

### 4.1. Experimental Settings

Our training framework follows a per-scene training like NeRF [27]. Given multiple images of the scene, they learn to generate novel views of the scene. Efforts have been made to reduce the number of required images, such as PixelNeRF [47] which trains NeRF with just one or a few images. In addition, HARP [17], HandAvatar [5], HandNeRF [10], and others [7, 29] are based on per-scene training, requiring dozens of multiple frames of the same hand (scene). Our method, utilizing texture symmetry and a parametric texture model, requires only a single image for training.

Table 1. Quantitative comparisons of BiTT, S2Hand [6], HTML [34] and HARP [17]. Training data are from Interhand2.6M [28] including all identities. For evaluations, novel poses and viewpoints are randomly selected from the same hand identity. In the case when not using GT mesh, we used IntagHand [21] for obtaining meshes.

(a) Using GT mesh in all methods.

| Evaluation | Method | L1↓ | LPIPS↓ | PSNR↑ | MS-SSIM↑ |
|---|---|---|---|---|---|
| Appearance Reconstruction | S2Hand [6] | 0.0206 | 0.1340 | 26.39 | 0.8570 |
| | HTML [34] | 0.0256 | 0.1292 | 24.72 | 0.8152 |
| | HARP [17] | 0.0157 | **0.0696** | 28.11 | 0.9061 |
| | **BiTT(ours)** | **0.0101** | 0.1019 | **30.41** | **0.9349** |
| Novel Poses | S2Hand | 0.0221 | 0.1343 | 25.70 | 0.8507 |
| | HTML | 0.0255 | 0.1291 | 24.49 | 0.8153 |
| | HARP | 0.0239 | 0.1266 | 25.79 | 0.8546 |
| | **BiTT(ours)** | **0.0209** | **0.1261** | **26.54** | **0.8564** |
| Different Views | S2Hand | 0.0217 | 0.1320 | 25.73 | 0.8484 |
| | HTML | 0.0254 | 0.1282 | 24.42 | 0.8133 |
| | HARP | 0.0234 | 0.1189 | 25.97 | 0.8346 |
| | **BiTT(ours)** | **0.0204** | **0.1092** | **27.79** | **0.8843** |

(b) Without using GT mesh in all methods.

| Evaluation | Method | L1↓ | LPIPS↓ | PSNR↑ | MS-SSIM↑ |
|---|---|---|---|---|---|
| Appearance Reconstruction | S2Hand [6] | 0.0264 | 0.1214 | 25.72 | 0.8897 |
| | HTML [34] | 0.0268 | 0.1207 | 24.48 | 0.8545 |
| | HARP [17] | 0.0237 | 0.1047 | 25.17 | 0.8697 |
| | **BiTT(ours)** | **0.0131** | **0.1044** | **28.40** | **0.9093** |
| Novel Poses | S2Hand | 0.0280 | 0.1525 | 23.06 | 0.8092 |
| | HTML | 0.0310 | 0.1299 | 23.46 | 0.8281 |
| | HARP | 0.0256 | 0.1410 | 24.32 | 0.8419 |
| | **BiTT(ours)** | **0.0223** | **0.1228** | **25.12** | **0.8423** |
| Different Views | S2Hand | 0.0244 | 0.1512 | 24.22 | 0.8335 |
| | HTML | 0.0291 | 0.1297 | 24.22 | 0.8375 |
| | HARP | 0.0251 | 0.1367 | 24.49 | 0.8507 |
| | **BiTT(ours)** | **0.0210** | **0.1273** | **26.34** | **0.8674** |

Table 2. Quantitative comparisons between compared methods in RGB2Hands [40] dataset. All training and testing images are randomly selected. As RGB2Hands has no ground truth mesh, we used IntagHand [21] as an off-the-shelf model for two hand mesh reconstruction.

| Evaluation | Method | L1↓ | LPIPS↓ | PSNR↑ | SSIM↑ | MS-SSIM↑ |
|---|---|---|---|---|---|---|
| Appearance Reconstruction | S2Hand [6] | 0.0179 | 0.0601 | 25.72 | 0.9459 | 0.9286 |
| | HTML [34] | 0.0203 | 0.0923 | 24.42 | 0.8927 | 0.9075 |
| | HARP [17] | 0.0155 | **0.0433** | 25.63 | 0.9309 | **0.9344** |
| | **BiTT(ours)** | **0.0148** | 0.0683 | **26.02** | **0.9501** | 0.9323 |
| Novel Poses | S2Hand | 0.0222 | 0.0778 | 24.22 | 0.9326 | 0.8991 |
| | HTML | 0.0233 | 0.0961 | 23.25 | 0.8829 | 0.8900 |
| | HARP | 0.0208 | **0.0758** | 23.88 | 0.9043 | 0.9042 |
| | **BiTT(ours)** | **0.0196** | 0.0774 | **24.54** | **0.9352** | **0.9046** |

Our model is trained in end-to-end manner. For each scene, our training process involves 700 epochs with a learning rate decay by half every 200 steps, starting from an initial rate of 0.001. Each scene training process is completed in less than 7 minutes.

**Datasets.** We mainly use the InterHand2.6M [28] dataset which is composed of interacting two hands for both quantitative and qualitative evaluations. Since HARP [17] and NeRF-based approaches [5, 10, 29] require hundreds of images per scene for training, can only involve a few scenes to experiments. Requiring a single image, our experiments are conducted through a total of 378 scenes (images) utilizing all 26 hand identities of the dataset. After training, we evaluated its performance with different poses and different views of the hands from the same identity. Specifically, we evaluated with 8 distinct poses and 8 varying viewpoints for each scene, resulting in a total of 6,048 testing images.

We also evaluated our model with the RGB2Hands [40] dataset, however, since RGB2Hands does not present multiview images, we evaluated it with different pose images. Across 300 scenes in the RGB2Hands dataset, we trained our method and evaluated it with 40 different poses of images for each scene, thus 12,000 testing images in total.

For each dataset, we present the proportion of visible, invisible, and usable symmetric texture pixels in each hand texture at Tab. 3. Using symmetric information, we can acquire up to 60% texture of each hand; otherwise, we have only about 35% full texture available from the input images. This demonstrates that using the symmetric information between both hands is reasonable for reconstructing two hand textures from a single image.

**Metrics.** For the quantitative comparisons of different appearance models, we use a set of metrics that is often applied to assess the fidelity and quality of rendered images. We use the L1, learned perceptual image patch similarity (LPIPS) [50], the structural similarity metric (SSIM) [41], the multiscale structural similarity metric (MS-SSIM) [42], and the peak signal-to-noise ratio (PSNR).

**Compared Methods.** We compare our method with S2Hand [6] which reconstructs a single hand with per-vertex texture rendering. HARP [17] and HTML [34] are

Table 3. Pixel ratio comparison between left-hand, right-hand visible texture, usable symmetric texture, and invisible texture in our experiment dataset. Colors refer to the Fig. 5 mask label.

| Dataset | InterHand2.6M [28] | RGB2Hands [40] |
|---|---|---|
| Left-hand visible texture | 35.40% | 39.72% |
| Right-hand visible texture | 36.44% | 39.06% |
| Usable symmetric texture | 24.29% | 10.21% |
| Invisible texture | 19.89% | 24.95% |

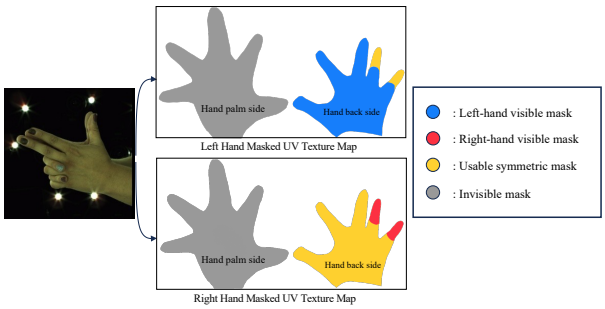


Figure 5. This figure shows the visible, invisible, usable symmetric texture mask on the UV texture map from an image.

the methods that reconstruct a single hand with UV map rendering, and we compare the appearance quality among these methods. We modify S2Hand, HTML, and HARP to two hands for comparison. These methods are extended to estimate each hand texture discretely while learning each hand texture from a rendered image. Implicit function-based methods [5, 7, 10, 29] are not included in the comparison, as their methods are not straightforward to extend for two hands, and their methods require at least hundreds of images per scene for training. For showing robustness on geometric misalignment, we present two result tables; using ground truth mesh in all the compared methods at Tab. 1a and without using ground truth mesh in all methods at Tab. 1b. Where the ground truth mesh is unavailable, such as RGB2Hands data, we initialize hand meshes using the off-the-shelf method, IntagHand [21].

### 4.2. Evaluation on Texture Reconstruction

**Comparing with Prior Arts.** We show the qualitative results in Fig. 6 and the quantitative comparisons in Tab. 1 and Tab. 2. The results show that BiTT significantly outperforms other baselines, especially in reconstructing the invisible hand parts. HTML [34], S2Hand [6] are not able to represent detailed appearances like vessels, wrinkles, and hair, whereas BiTT captures detailed personalized appearances using symmetric information. HARP [17] excels in the visible side appearance (regarding Tab. 1, Tab. 2), but it lacks the capability to reconstruct the invisible sides, merely displaying a uniform color in those areas. Notably, BiTT remains robust to geometric misalignments, maintaining high


Figure 6. Qualitative results of HTML [34], S2Hand [6], HARP [17], and BiTT rendered on novel-pose and viewpoint. The last two rows pertain to the RGB2Hands [40] dataset, while the remaining rows are from the InterHand2.6M [28] dataset.

performance even without using GT as having the advantages of the parametric model. The performance gain on RGB2Hands Tab. 2 is relatively less significant than on InterHand2.6M [28]. This is due to the fact that RGB2Hands exhibits a lower percentage of usable symmetric texture, indicated in Tab. 3. More results of BiTT are shown at Fig. 7.

We qualitatively compare HandNeRF [10] based on the results reported in their paper, since there is no detailed experiment description and the models/codes are not available. As shown in Fig. 8, even if our model is trained from a single image, our model can capture the realistic texture of both hands comparably to HandNeRF. Note that HandNeRF is trained over hundreds of images from 10 views. Furthermore, BiTT is able to render in different illuminations as shown in Fig. 1 and Fig. 7.

### 4.3. Ablation Study

**Symmetric information and Albedo Consistency Loss.** We perform an ablation study of the use of symmetric information in reconstructing hand texture. We replace the

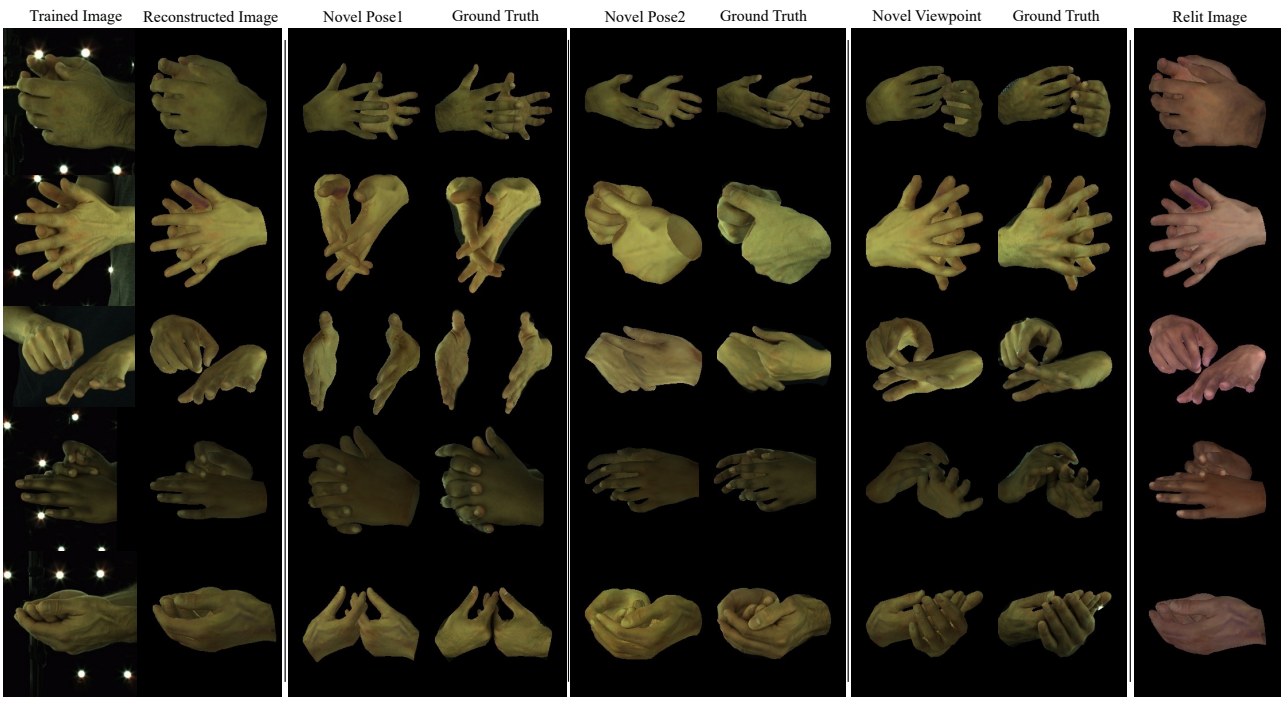


Figure 7. Using only a single image input, our method reconstructs realistic detailed textures for both hands. We present some results having different poses, viewpoints with corresponding ground truth images, and relighted images.

Table 4. Effects of coarse stage estimation, use of symmetric texture information (Sym. Tex.), and albedo consistency loss. Results are the mean value evaluated on novel poses and viewpoints.

| Coarse Stage | Sym. Tex. | $\mathcal{L}_{alb}$ | LPIPS↓ | PSNR↑ | SSIM↑ |
|---|---|---|---|---|---|
| ✓ | | | 0.1329 | 25.36 | 0.8940 |
| ✓ | ✓ | | 0.1230 | 26.21 | 0.9163 |
| ✓ | ✓ | ✓ | **0.1176** | **27.16** | **0.9199** |

bi-directional connection with a uni-directional connection in BTR and $\mathcal{L}_{sym}$ is omitted. As shown in Tab. 4, not using the symmetric information significantly degrades the performance of reconstructing invisible side appearances.

We also perform an ablation study for the albedo consistency loss. In Tab. 4, the albedo consistency loss improves the overall performance of the model by more precisely estimating the albedo image.

## 5. Conclusions

In this work, we presented a novel two-hand texture reconstruction method from a single image called BiTT. First, the bi-directional texture reconstructor is proposed to create the full texture of both hands interactively. Second, we introduce a way to use the texture parametric model for recovering texture. The experimental results demonstrate that our method outperforms existing methods both qualitatively and quantitatively. We believe that our work can present a realistic experience to users by accurately representing their personalized hands in AR/VR applications.

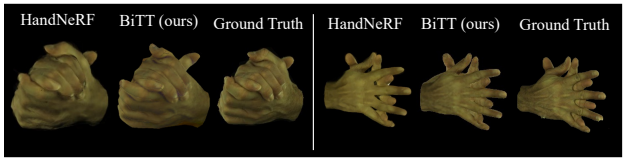


Figure 8. Qualitative results of HandNeRF [10] and BiTT.

**Limitations and future work.** One limitation is that although BiTT is robust to the geometric misalignments through the texture parametric model, there still exist instances where seriously misaligned meshes cause a significant level of noise. Future work could involve the learning method that can refine 3D meshes through detailed texture reconstruction. Another future work can be extending our system to detect and incorporate tattoos or accessories on the invisible side of the hands through generative networks[35], enhancing realism.

**Acknowledgements.** This work was in part supported by NST grant (CRC 21011, MSIT), KOCCA grant (R2022020028, MCST), IITP grant (RS-2023-00228996, MSIT).

## References


[1] Thiemo Alldieck, Mihai Zanfir, and Cristian Sminchisescu. Photorealistic monocular 3d reconstruction of humans wearing clothing. In *CVPR*, 2022. 1, 4

[2] Anil Armagan, Guillermo Garcia-Hernando, Seungryul Baek, Shreyas Hampali, Mahdi Rad, Zhaohui Zhang, Shipeng Xie, MingXiu Chen, Boshen Zhang, Fu Xiong, Yang Xiao, Zhiguo Cao, Junsong Yuan, Pengfei Ren, Weiting Huang, Haifeng Sun, Marek Hrúz, Jakub Kanis, Zdeněk Krňoul, Qingfu Wan, Shile Li, Linlin Yang, Dongheui Lee, Angela Yao, Weiguo Zhou, Sijia Mei, Yunhui Liu, Adrian Spurr, Umar Iqbal, Pavlo Molchanov, Philippe Weinzaepfel, Romain Brégier, Grégory Rogez, Vincent Lepetit, and Tae-Kyun Kim. Measuring generalisation to unseen viewpoints, articulations, shapes and objects for 3d hand pose estimation under hand-object interaction. In *ECCV*, 2020. 1

[3] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In *CVPR*, 2019. 1

[4] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Weakly-supervised domain adaptation via gan and mesh model for estimating 3d hand poses interacting objects. In *CVPR*, 2020. 1

[5] Xingyu Chen, Baoyuan Wang, and Heung-Yeung Shum. Hand avatar: Free-pose hand animation and rendering from monocular video. In *CVPR*, 2023. 2, 3, 4, 5, 6, 7

[6] Yujin Chen, Zhigang Tu, Di Kang, Linchao Bao, Ying Zhang, Xuefei Zhe, Ruizhi Chen, and Junsong Yuan. Model-based 3d hand reconstruction via self-supervised learning. In *CVPR*, 2021. 2, 4, 6, 7

[7] Enric Corona, Tomas Hodan, Minh Vo, Francesc Moreno-Noguer, Chris Sweeney, Richard Newcombe, and Lingni Ma. Lisa: Learning implicit shape and appearance of hands. In *CVPR*, 2022. 1, 2, 4, 5, 7

[8] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *CVPR*, 2018. 1

[9] Shubham Goel, Angjoo Kanazawa, , and Jitendra Malik. Shape and viewpoints without keypoints. In *ECCV*, 2020. 1, 2

[10] Zhiyang Guo, Wengang Zhou, Min Wang, Li Li, and Houqiang Li. Handnerf: Neural radiance fields for animatable interacting hands. In *CVPR*, 2023. 2, 3, 5, 6, 7, 8

[11] Tao Hu, Liwei Wang, Xiaogang Xu, Shu Liu, and Jiaya Jia. Self-supervised 3d mesh reconstruction from single images. In *CVPR*, 2021. 2

[12] Tao Hu, Tao Yu, Zerong Zheng, He Zhang, Yebin Liu, and Matthias Zwicker. Hvtr: Hybrid volumetric-textural rendering for human avatars. In *3DV*, 2022. 1

[13] Zheheng Jiang, Hossein Rahmani, Sue Black, and Bryan M Williams. A probabilistic attention model with occlusion-aware texture regression for 3d hand reconstruction from a single rgb image. In *CVPR*, 2023. 2

[14] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 1, 2

[15] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael Black, Krikamol Muandet, and Tang. Siyu. Grasping field: Learning implicit representations for human grasps. In *3DV*, 2020. 1

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

[17] Korrawe Karunratanakul, Sergey Prokudin, Otmar Hilliges, and Siyu Tang. Harp: Personalized hand reconstruction from a monocular rgb video. In *CVPR*, 2023. 2, 3, 4, 5, 6, 7

[18] Jihyun Lee, Junbong Jang, Donghwan Kim, Minhyuk Sung, and Tae-Kyun Kim. Fourierhandflow: Neural 4d hand representation using fourier query flow. In *NIPS*, 2023. 1

[19] Jihyun Lee, Minhyuk Sung, Honggyu Choi, and Tae-Kyun Kim. Im2hands: Learning attentive implicit representation of interacting two-hand shapes. In *CVPR*, 2023. 1

[20] Biwen Lei, Jianqiang Ren, Mengyang Feng, Miaomiao Cui, and Xuansong Xie. A hierarchical representation network for accurate and detailed face reconstruction from in-the-wild images. In *CVPR*, 2023. 1

[21] 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, 6, 7

[22] Ruilong Li, Julian Tanke, Minh Vo, Michael Zollhofer, Jurgen Gall, Angjoo Kanazawa, and Christoph Lassner. Tava: Template-free animatable volumetric actors. In *ECCV*, 2022. 1

[23] Xueting Li, Sifei Liu, Kihwan Kim, Shalini De Mello, Varun Jampani, Ming-Hsuan Yang, and Jan Kautz. Self-supervised single-view 3d reconstruction via semantic consistency. In *ECCV*, 2020. 1, 2

[24] Yuwei Li, Longwen Zhang, Zesong Qiu, Yingwenqi Jiang, Nianyi Li, Yuexin Ma, Yuyao Zhang, Lan Xu, and Jingyi Yu. Nimble: A non-rigid hand model with bones and muscles. *ACM TOG*, 2022. 2

[25] Zhiyuan Ma, Xiangyu Zhu, Guojun Qi, Zhen Lei, and Lei Zhang. Otavatar: One-shot talking face avatar with controllable tri-plane rendering. In *CVPR*, 2023. 1

[26] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Realfusion: 360 reconstruction of any object from a single image. In *CVPR*, 2023. 4

[27] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1, 4, 5

[28] 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. 2, 6, 7

[29] Akshay Mundra, Mallikarjun B R, Jiayi Wang, Marc Habermann, Christian Theobalt, and Mohamed Elgharib. Livehand: Real-time and photorealistic neural hand rendering. In *ICCV*, 2023. 2, 3, 4, 5, 6, 7

[30] Atsuhiro Noguchi, Xiao Sun, Stephen Lin, and Tatsuya Harada. Neural articulated radiance field. In *ICCV*, 2021. 1

[31] Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. In *ICCV*, 2019. 4

[32] Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *ICCV*, 2021. 1

[33] Bui Tuong Phong. *Illumination for Computer Generated Pictures*. Association for Computing Machinery, 1998. 3, 4

[34] Neng Qian, Jiayi Wang, Franziska Mueller, Florian Bernard, Vladislav Golyanik, and Christian Theobalt. HTML: A Parametric Hand Texture Model for 3D Hand Reconstruction and Personalization. In *ECCV*, 2020. 2, 4, 6, 7

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *arxiv:2112.10752*, 2021. 8

[36] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM TOG*, 2017. 4

[37] 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. 4

[38] Soumyadip Sengupta, Angjoo Kanazawa, Carlos D. Castillo, and David W. Jacobs. Sfsnet: Learning shape, refectance and illuminance of faces in the wild. In *CVPR*, 2018. 1

[39] Qiuhong Shen, Xingyi Yang, and Xinchao Wang. Anything-3d: Towards single-view anything reconstruction in the wild. *arXiv:2304.10261*, 2023. 4

[40] 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. 2, 6, 7

[41] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. In *IEEE TIP*, 2004. 6

[42] Zhou Wang, Eero P Simoncelli, and Alan C Bovic. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 6

[43] Shangzhe Wu, Ameesh Makadia, Jiajun Wu, Noah Snavely, Richard Tucker, and Angjoo Kanazawa. De-rendering the world's revolutionary artefacts. In *CVPR*, 2021. 2

[44] Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J. Black. ICON: Implicit Clothed humans Obtained from Normals. In *CVPR*, 2022. 1, 4

[45] Yufei Ye, Shubham Tulsiani, and Abhinav Gupta. Shelf-supervised mesh prediction in the wild. In *CVPR*, 2021. 1, 2

[46] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What's in your hands? 3d reconstruction of generic objects in hands. In *CVPR*, 2022. 1

[47] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *CVPR*, 2021. 5

[48] Chang Yu, Xiangyu Zhu, Xiaomei Zhang, Zhaoxiang Zhang, and Zhen Lei. Graphics capsule: Learning hierarchical 3d face representations from 2d images. In *CVPR*, 2023. 1

[49] Zhengdi Yu, Shaoli Huang, Fang Chen, Toby P. Breckon, and Jue Wang. Acr: Attention collaboration-based regressor for arbitrary two-hand reconstruction. In *CVPR*, 2023. 1

[50] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 6