A Probabilistic Attention Model with Occlusion-aware Texture Regression for 3D Hand Reconstruction from a Single RGB Image

Zheheng Jiang[1] Hossein Rahmani[1] Sue Black[2] Bryan M. Williams[1]
[1]Lancaster University [2]St John's College of the University of Oxford
{z.jiang11,h.rahmani,b.williams6}@lancaster.ac.uk, sue.black@sjc.ox.ac.uk

Abstract

Recently, deep learning based approaches have shown promising results in 3D hand reconstruction from a single RGB image. These approaches can be roughly divided into model-based approaches, which are heavily dependent on the model's parameter space, and model-free approaches, which require large numbers of 3D ground truths to reduce depth ambiguity and struggle in weakly-supervised scenarios. To overcome these issues, we propose a novel probabilistic model to achieve the robustness of model-based approaches and reduced dependence on the model's parameter space of model-free approaches. The proposed probabilistic model incorporates a model-based network as a prior-net to estimate the prior probability distribution of joints and vertices. An Attention-based Mesh Vertices Uncertainty Regression (AMVUR) model is proposed to capture dependencies among vertices and the correlation between joints and mesh vertices to improve their feature representation. We further propose a learning based occlusion-aware Hand Texture Regression model to achieve high-fidelity texture reconstruction. We demonstrate the flexibility of the proposed probabilistic model to be trained in both supervised and weakly-supervised scenarios. The experimental results demonstrate our probabilistic model's state-of-the-art accuracy in 3D hand and texture reconstruction from a single image in both training schemes, including in the presence of severe occlusions.

1. Introduction

3D hand shape and texture reconstruction from a single RGB image is a challenging problem that has numerous applications such as human-machine interaction [1, 2], virtual and augmented reality [3–6], and sign language translation [7]. In recent years, there has been significant progress in reconstructing 3D hand pose and shape from a monocular images [8–16]. These approaches can be generally categorized into model-based and model-free approaches. Model-based approaches [9,13–15] utilize a parametric model such as MANO [17] and train a network to regress its parametric representation in terms of shape and pose. Since the parametric model contains priors of human hands, these approaches are robust to environment variations and weakly-supervised training [12]. However, the shape and pose regression is constrained by the parametric model that is learned from the limited hand exemplars [8].

In contrast, model-free approaches [8, 11, 12, 16] regress the coordinates of 3D hand joints and mesh directly instead of using parametric models. Despite the remarkable results they have achieved, there are several limitations. For example, Graph-CNN is used by [8, 11] to model neighborhood vertex-vertex interactions, but such models cannot capture long range dependencies among vertices. Although [12] has addressed this issue by employing self-attention mechanism, it does not distinguish joints and vertices, processing them together in a same self-attention module. Moreover none of these works can support weakly supervised training and often require a large amount of 3D annotations of both joints and vertices to reduce depth ambiguity in monocular 3D reconstruction [18].

Motivated by the above observations, our first goal is to combine the benefits of the model-based and model-free approaches. To this end, we develop a probabilistic method that incorporates the MANO model into a prior-net to estimate the prior probability distribution of joints and vertices instead of using deterministic settings as previous approaches have done. To relax the solution space of the MANO model, an Attention-based Mesh Vertices Uncertainty Regression model (AMVUR) is proposed to estimate the conditioned probability distribution of the joints and vertices. In AMVUR, to improve feature representation of joints and vertices, a cross-attention model is proposed to capture the correlation between 3D positional encoded joints and mesh vertices, followed by a self-attention model for capturing the short/long range dependencies among mesh vertices. With the proposed architecture, the AMVUR model can be jointly trained with the prior-net to achieve superior performance to using them independently. To the

best of our knowledge, our probabilistic attention model is the first approach that learns the probability distribution of hand joints and mesh under a probabilistic model.

The ability to reconstruct 3D hands with high-fidelity texture is helpful for 3D Hand Personalization and improves the performance of hand tracking systems [19–21]. Moreover, Hand texture reconstruction is important for the user experience and bodily self-consciousness in immersive virtual reality systems [3]. We thus propose a learning based occlusion-aware hand texture regression model by introducing an occlusion-aware rasterization and reverse interpolation to achieve high-fidelity hand texture reconstruction.

Our contributions are summarized as follows: **(1)** We introduce an Attention-based Mesh Vertices Uncertainty Regression model (AMVUR) comprising a cross attention module for capturing the correlation between joints and mesh vertices and a self-attention module for capturing the short/long range dependencies among mesh vertices. **(2)** We propose a novel probabilistic attention model to learn the probability distribution of hand joints and mesh vertices, where the MANO parametric model is regarded as a prior-net and jointly trained with AMVUR. **(3)** We propose an Occlusion-aware Hand Texture Regression model to achieve high-fidelity hand texture reconstruction, including in the presence of severe occlusions. **(4)** We demonstrate that our network can be trained in both fully supervised and weakly supervised training schemes, achieving state-of-the-art (SOTA) performance on the three benchmark 3D hand reconstruction datasets: HO3Dv2 [22], HO3Dv3 [23] and FreiHand [24].

2. Related Work

Model-based Methods: Recently, numerous works have been proposed to reconstruct the 3D hand by regressing the shape and pose parameters of a parametric hand model named MANO [17] that is learned from around 1K high-resolution 3D hand scans. Boukhayma et al. [15] regress these parameters along with camera parameters via a deep convolutional encoder which takes a hand image and 2D joint heat-maps extracted from a joint detection network as input. Zhang et al. [25] propose an iterative regression module to fit the camera and model parameters from 2D joint heat-maps. Attention based approaches have also received increasing attention. Liu et al. [26] introduce an attention based contextual reasoning module for modeling hand-object interaction. A most recent approach [14] proposes to inject hand information into occluded regions by using a self-attention mechanism. However, their approach is a 2D spatial attention mechanism that is unable to capture correlation between mesh vertices in 3D space.

By utilizing strong hand priors of MANO, several other approaches [13, 18] have attempted to reconstruct 3D hand shape and pose with weak supervision. Kulon et al. [10] apply Parametric Model Fitting to generate 3D mesh from detected 2D hand keypoints. The fitted mesh is then used as a supervisory signal to train a feed-forward network with a mesh convolutional decoder. Spurr et al. [18] introduce biomechanical constraints to guide the network to predict feasible hand poses with weakly-annotated real-world data. Chen et al. [13] use 2D joints extracted from an off-the-shelf 2D pose estimator as a supervisory signal to train a model-based autoencoder to estimate 3D hand pose and shape. However, similar to the model based approaches, they do not exploit correlation between joints and mesh vertices, yet our proposed AMVUR model addresses this issue and improves the feature representation of joints and vertices.

Model-free Methods: Although hand parametric models such as MANO serve as a strong structural prior to support 3D hand reconstruction, help to handle severe occlusions and help to accommodate weakly-annotated data, approaches that rely on this can easily get stuck in the model's parameter space, resulting in a non-minimal representation problem [8, 11]. To relax this heavy reliance on the parameter space, some approaches directly regress 3D positions of mesh vertices instead of predicting the model's parameters. Among these approaches, Kolotouros et al. [8] and Hongsuk et al. [11] combine an image-based CNN and a GraphCNN to estimate human mesh coordinates directly. Lin et al. [12] argue that GraphCNN can only capture the local interactions between neighboring vertices of the triangle mesh, so they use a self-attention mechanism to capture global interactions between the vertices and joints. Most recently, Hampali et al. [16] first extract joint features by localizing them on CNN feature maps, then take these features and their spatial encodings as the input to a transformer model for 3D hand pose estimation. However, spatial encoding is ambiguous to describe joints' 3D locations, especially for overlapping 3D joints in 2D images. Different from the above approaches, in AMVUR, a cross-attention module is proposed to learn the correlation between joints and mesh vertices, followed by a self-attention module to learn the correlation between different vertices.

Texture Reconstruction: 3D hand texture estimation has wide applications in virtual and augmented reality, but most of the previously mentioned approaches do not address this problem. Qian et al. [27] follow the 3D morphable face models [28] to create a parametric hand texture model using principal component analysis (PCA). However this model requires a hand dataset with 3D textured scans and is established on only 51 hand exemplars. Recently, Chen et al. [13] propose to regress hand texture via MLP layers by taking a global feature vector as input. However, this is limited to generating very coarse hand texture and is unable to recognize occlusions. In contrast, our texture hand model is able to reconstruct high-fidelity hand texture with occlusions.

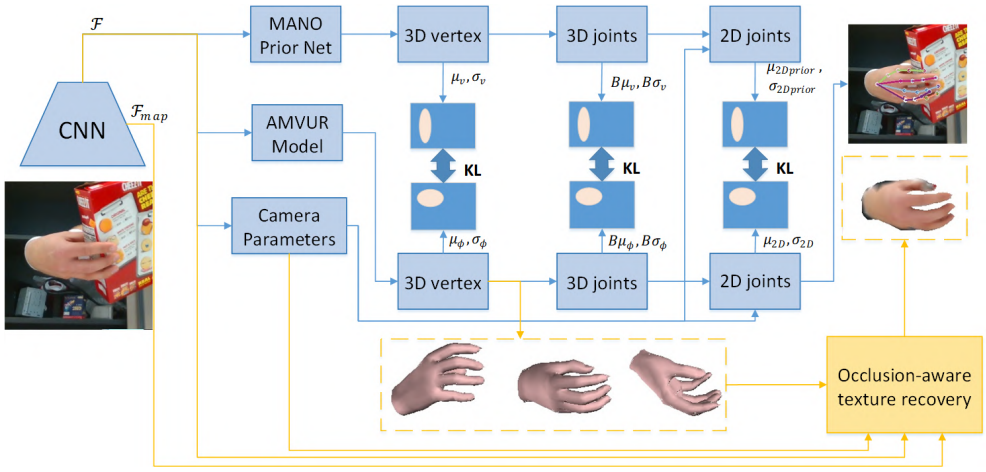

Figure 1. Overview of our proposed method. The blue arrows and yellow arrows denote the flows of hand mesh reconstruction and hand texture regression respectively. We firstly extract a global feature vector $\mathcal{F}$ and a shallow feature map $\mathcal{F}_{map}$ from the backbone CNN. For hand mesh reconstruction, our MANO prior-net and AMVUR model take the global feature vector as input and are jointly trained to estimate the probability distributions of the 3D vertices, 3D joints and 2D joints. During training, the probability distributions estimated by these two models are tied by the KL-divergence. The camera model estimates camera parameters that are used to project the 3D joints and 3D vertices to 2D space. For hand texture regression, an Occlusion-aware texture recovery model is proposed to reconstruct occlusion-aware high-fidelity hand texture by taking the global feature vector, the shallow feature map, estimated camera parameters and estimated 3D vertices as inputs. During inference, the test image is passed through the CNN followed by AMVUR to generate the most likely 3D hand mesh which is then fed to the Occlusion-aware texture recovery model to reconstruct a textured mesh.

3. Proposed Model

An overview of our proposed model is presented in Fig. 1. We first extract a global feature vector $\mathcal{F}$ and a shallow feature map $\mathcal{F}_{map}$ from our backbone Convolutional Neural Network (CNN). Then, we introduce a Bayesian model to describe the relationship between 3D hand joints, 3D hand mesh vertices, 2D hand joints and camera parameters for the regression task by taking $\mathcal{F}$ as input. This is described in Sections 3.1 and 3.2 along with our problem formulation. Our proposed AMVUR uses $\mathcal{F}$ to learn the correlation between joints and mesh vertices, described in Section 3.3. Our proposed Occlusion-aware Hand Texture Regression, which addresses occlusion-aware high-fidelity hand texture reconstruction by utilizing the global feature vector and the shallow feature map, is described in Section 3.4.

3.1. Problem Formulation

Given a 2D image I containing a hand, our goal is to predict the locations of the 2D hand joints $J_{2D} \in \mathbb{R}^{K\times 2}$, 3D hand joints $J_{3D} \in \mathbb{R}^{K\times 3}$, 3D hand mesh vertices $V_{3D} \in \mathbb{R}^{\mathcal{V}\times 3}$ and camera parameters C, where K is the number of joints and $\mathcal{V}$ is the number of mesh vertices. Most MANO based methods [9, 10, 13, 14] firstly propose a deep regression model to fit the MANO pose and shape parameters, from which the final 3D hand mesh is estimated via a MANO layer. This limits the learning ability of the deep neural network for 3D hand mesh regression. In contrast to the above methods, we use the MANO model as a prior-net and integrate it with our proposed Attention-based Mesh Vertices Uncertainty Regression model (AMVUR) for end-to-end training. Let δ and θ donate the model parameters of prior-net and AMVUR respectively, learned from the training dataset $D = \left\{J^i_{3D}, V^i_{3D}, J^i_{2D}, C^i, I^i\right\}_{i=1}^{T}$, where T is the total number of training images. The model parameters are estimated by maximizing the log likelihood function

$$\ln \mathcal{L}\left(\delta\right) = \ln \prod_i P\left(J^i_{2D}, J^i_{3D}, V^i_{3D}, C^i | I^i; \delta\right) \tag{1}$$

where $P\left(J^i_{2D}, J^i_{3D}, V^i_{3D}, C^i | I^i; \delta\right)$ is a prior joint probability distribution estimated using the MANO model. This is maximised as:

$$\begin{aligned} &\underset{\delta,\theta}{\operatorname{argmax}} \ln \mathcal{L}\left(\delta, \theta\right) \\ &= \underset{\delta,\theta}{\operatorname{argmin}} \sum_i \Big(- \ln Q\left(J^i_{2D}, J^i_{3D}, V^i_{3D}, C^i | I^i; \theta\right) \\ &\quad + \ln \frac{Q\left(J^i_{2D}, J^i_{3D}, V^i_{3D}, C^i | I^i; \theta\right)}{P\left(J^i_{2D}, J^i_{3D}, V^i_{3D}, C^i | I^i; \delta\right)} \Big), \end{aligned} \tag{2}$$

where $Q\left(J^i_{2D}, J^i_{3D}, V^i_{3D}, C^i | I^i; \theta\right)$ is an approximate joint probability distribution that is learned by the proposed Attention-based Mesh Vertices Uncertainty Regression model (see Section 3.3 for more details). The dependencies between the variables $J^i_{3D}, V^i_{3D}, J^i_{2D}, C^i$ and I^i are governed by a Bayesian Network represented via the Directed Acyclic Graph (DAG) shown in Figure 2. Bayes' theorem allows $P\left(J^i_{2D}, J^i_{3D}, V^i_{3D}, C^i | I^i; \delta\right)$ to be factorized using the DAG as the product of $P\left(V^i_{3D} | I^i; \delta\right)$, $P\left(C^i | I^i; \delta\right)$, $P\left(J^i_{3D} | V^i_{3D}; \delta\right)$ and $P\left(J^i_{2D} | J^i_{3D}, C^i; \delta\right)$. During training,

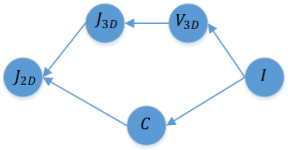

Figure 2. A DAG describes the dependence between 2D joints, 3D joints, 3D mesh vertices, camera parameters and image.

the probability distribution of $Q\left(J_{2D}^i, J_{3D}^i, V_{3D}^i, C^i | I^i; \theta\right)$ and the prior probability distribution generated by the MANO model are encouraged to be close to each other. This allows the probability distribution training of $Q\left(J_{2D}^i, J_{3D}^i, V_{3D}^i, C^i | I^i; \theta\right)$ to be conditioned on the prior distribution. $Q\left(J_{2D}^i, J_{3D}^i, V_{3D}^i, C^i | I^i; \theta\right)$ can be factorized as the product of $Q\left(V_{3D}^i | I^i; \phi\right)$, $Q\left(C^i | I^i; \gamma\right)$, $Q\left(J_{3D}^i | V_{3D}^i; \phi\right)$ and $Q\left(J_{2D}^i | J_{3D}^i, C^i; \phi, \gamma\right)$, where $\phi, \gamma \in \theta$ are trainable parameters in our AMVUR model.

Considering observation noise and model error, we assume the approximate probability distribution of $Q\left(V_{3D} | I; \phi\right)$ and prior probability distribution of $P\left(V_{3D} | I; \delta\right)$ take on Gaussian distributions $\mathcal{N}_\phi\left(\mu_\phi, diag\left(\sigma_\phi\right)\right)$ and $\mathcal{N}\left(\mu_v, diag\left(\sigma_v\right)\right)$, respectively. $\mu_\phi, \sigma_\phi \in \mathbb{R}^{\mathcal{V}\times 3}$ and $\mu_v, \sigma_v \in \mathbb{R}^{\mathcal{V}\times 3}$ are learned from our AMVUR model and the MANO model. Given the above, we can derive our loss function for mesh vertices as:

$$\begin{aligned}\mathcal{L}_{V_{3D}} &= -\ln Q\left(V_{3D}^i | I^i; \phi\right) + \ln \frac{Q\left(V_{3D}^i | I^i; \phi\right)}{P\left(V_{3D}^i | I^i; \delta\right)} \\ &= \sum_m \left[\frac{1}{2}\left(\frac{\bar{V}_{3D}^{i,m} - \mu_\phi^m}{\sigma_\phi^m}\right)^2 + \ln\sqrt{2\pi}\sigma_\phi^m\right] \\ &\quad + \frac{1}{2}\left[\ln\prod_m \frac{\sigma_v^m}{\sigma_\phi^m} - d + \sum_m \frac{\sigma_\phi^m + \left(\mu_v^m - \mu_\phi^m\right)^2}{\sigma_v^m}\right]\end{aligned} \tag{3}$$

where $\bar{V}_{3D}^{i,m}$ denotes the ground truth 3D coordinates of the mesh vertices of the i^{th} image. m is an index of each dimension. The mean μ_ϕ and variance σ_ϕ are learned via two MLP neural networks with $\phi \in \theta$. The mean μ_v of prior net is learned via the MANO model and variance σ_v is supposed to be equal to $\mathbf{1}$. d denotes the dimension of μ_v. The last term of the equation penalizes difference between the approximate distribution Q and the prior distribution P during training. Different from the previously widely used L1/L2 loss, which is less able to capture the data distribution, our loss function allows our model to consider the uncertainty and variability in the hand, which is important for modeling complex and varied 3D meshes. Further, sampling from the distribution during training of our probabilistic model allows the model to explore different variations of the mesh, leading to a more robust and generalizable model.

Since the prior probability distribution of camera parameters is unknown, we assume that $Q\left(C | J_{2D}^i, I^i; \gamma\right)$ and $P\left(C | J_{2D}^i, I^i; \delta\right)$ are subject to Gaussian distributions $\mathcal{N}_\gamma\left(\mu_\gamma, diag\left(\mathbf{1}\right)\right)$ and $\mathcal{N}\left(\bar{C}^i, diag\left(\mathbf{1}\right)\right)$. The loss function for the camera parameters can then be derived as:

$$\mathcal{L}_C = \sum_m \left(\bar{C}^{i,m} - \mu_\gamma^m\right)^2, \tag{4}$$

where $\bar{C}^{i,m}$ denotes the m^{th} index of the ground truth camera parameters of the i^{th} image. The mean μ_γ is learned via a MLP neural network.

To model the dependence between the 3D joints and 3D mesh vertices, we follow the common use in [12–14, 17] to use a pre-defined regression matrix $B \in \mathbb{R}^{K\times\mathcal{V}}$ from the MANO model. Meanwhile, the loss function for the 3D joints can be derived as:

$$\begin{aligned}\mathcal{L}_{J_{3D}} &= \sum_m \left[\frac{1}{2}\left(\frac{\bar{J}_{3D}^{i,m} - \left(B\mu_\phi\right)_m}{\left(B\sigma_\phi\right)_m}\right)^2 + \ln\sqrt{2\pi}\left(B\sigma_\phi\right)_m\right] \\ &\quad + \frac{1}{2}\left[\ln\prod_m \frac{\left(B\sigma_v\right)_m}{\left(B\sigma_\phi\right))_m} - d\right. \\ &\quad \left. + \sum_m \frac{\left(B\sigma_\phi\right)_m + \left(\left(B\mu_v\right)_m - \left(B\mu_\phi\right)_m\right)^2}{\left(B\sigma_v\right)_m}\right],\end{aligned} \tag{5}$$

where $\bar{J}_{3D}^{i,m}$ denotes the ground truth 3D joints.

To model the dependence between the 2D joints, the 3D joints and camera parameters, a weak perspective camera model: $J_{2D} = sJ_{3D}R + T$ is adopted, where s is the scale, $R \in \mathbb{R}^3$ and $T \in \mathbb{R}^3$ denote the camera rotation and translation of camera parameters C, respectively. The camera parameters are in axis–angle representation using radians followed by Rodrigues' rotation formula to obtain the rotation matrix. The loss function for 2D joints is derived as:

$$\begin{aligned}\mathcal{L}_{J_{2D}} &= \sum_m \left(\frac{1}{2}\left(\frac{\bar{J}_{2D}^{i,m} - S_m(\mu_\phi)}{S_m(\sigma_\phi)}\right)^2 + \ln\sqrt{2\pi}S_m(\sigma_\phi)\right) \\ &\quad + \frac{1}{2}\left[\ln\prod_m \frac{S_m(\sigma_v)}{S_m(\sigma_\phi)} - d\right. \\ &\quad \left. + \sum_m \frac{S_m(\sigma_\phi) + \left(S_m(\mu_v) - S_m(\mu_\phi)\right)^2}{S_m(\sigma_v)}\right],\end{aligned} \tag{6}$$

where $S_m(x) = \left(sBxR + T\right)_m$, and $\bar{J}_{2D}^{i,m}$ denotes the ground truth 2D joints of the i^{th} image.

3.2. Weakly Supervised Problem

With our defined Bayesian model, we are able to study the problem of training our model under the more challenging condition of no 3D ground truth information (such as 3D keypoints, 3D mesh vertices and camera parameters) being available for training. To tackle this problem, we deal with the variables J_{3D}, V_{3D} and C as hidden variables. So we

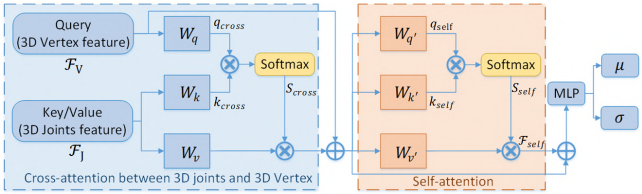

Figure 3. The Attention-based Mesh Vertices Uncertainty Regression (AMVUR) module.

aim to maximise the following with respect to θ:

$$\begin{aligned}\underset{\theta}{\operatorname{argmax}} \ln \mathcal{L}(\theta) &= \underset{\theta}{\operatorname{argmax}} \sum_i \ln P\left(J_{2D}^i | I^i; \theta\right) \\ &= \underset{\theta}{\operatorname{argmax}} \sum_i \ln \iiint P\left(J_{2D}^i, J_{3D}, V_{3D}, C^i | \right. \\ &\left. I^i; \theta\right) \mathrm{d}J_{3D}\mathrm{d}V_{3D}\mathrm{d}C\end{aligned} \tag{7}$$

However, direct marginalization of eq. (7) is intractable. So, a variance inference algorithm is developed to compute the penalized maximum likelihood estimation. The total weakly-supervised loss function of our model is:

$$\begin{aligned}Loss &= E_{V_{3D}\sim\mathcal{N}_\phi, C\sim\mathcal{N}_\gamma, J_{3D}\sim\mathcal{N}_B} \\ &- \ln P\left(J_{2D}^i | J_{3D}, V_{3D}, C, I^i; \delta\right) \\ &+ D_{KL}\left[Q_\phi\left(V_{3D} | J_{2D}^i, I^i; \phi\right) \parallel P\left(V_{3D} | J_{2D}^i, I^i; \delta\right)\right] \\ &+ D_{KL}\left[Q_B\left(J_{3D} | V_{3D}, J_{2D}^i; \phi\right) \parallel P\left(J_{3D} | V_{3D}, J_{2D}^i; \delta\right)\right] \\ &+ D_{KL}\left[Q_\gamma\left(C | J_{2D}^i, I^i; \theta\right) \parallel P\left(C | J_{2D}^i, I^i; \delta\right)\right],\end{aligned} \tag{8}$$

where $D_{KL}[Q \parallel P]$ denotes the Kullback–Leibler divergence, which measures how the approximate probability distribution of Q is different from the prior probability distribution of P. $E_{V_{3D}\sim\mathcal{N}_\phi, C\sim\mathcal{N}_\gamma, J_{3D}\sim\mathcal{N}_B} - \ln P\left(J_{2D}^i | J_{3D}, V_{3D}, C, I^i; \theta\right)$ can be computed using eq. (6) after sampling V_{3D}, J_{3D} and C from probability distributions of $\mathcal{N}_\phi(\mu_\phi, diag(\sigma_\phi))$, $\mathcal{N}_\gamma(\mu_\gamma, diag(\sigma_\gamma))$ and $\mathcal{N}_B(B\mu_\phi, diag(B\sigma_\phi))$. The prior probability distributions of $P\left(J_{3D} | V_{3D}, J_{2D}^i; \delta\right)$ and $P\left(V_{3D} | J_{2D}^i, I^i; \delta\right)$ are learned via prior-net. We adopt the camera model of [13] to estimate the prior probability distributions of $P\left(C | J_{2D}^i, I^i; \delta\right)$, which is assumed to follow the Gaussian distributions $\mathcal{N}_\gamma(\mu_\gamma, diag(\mathbf{1}))$. The detailed derivation can be found in the Supplementary Information.

3.3. Attention-based Mesh Vertices Uncertainty Regression

The vast majority of previous works [9, 10, 13, 14] focus on adopting the MANO parametric model and consider regression of pose and shape parameters. However, the pose and shape regression is heavily constrained by the MANO parametric model that was constructed using limited hand exemplars. To overcome this limitation, we introduce an Attention-based Mesh Vertices Uncertainty Regression model (AMVUR) to relax the heavy reliance on the MANO model's parameter space and establish correlations between joints and meshes. To better guide our proposed AMVUR model during training, our probabilistic model takes the MANO parametric model as a prior-net and the AMVUR model estimates the probability distribution of mesh vertices conditioned on the prior-net. The illustration of AMVUR is shown in Figure 3.

To construct the 3D vertex and 3D joint features, we firstly extract global feature $\mathcal{F}$ from the backbone CNN. Inspired by the positional encoding of [29], we encode positional information by attaching the initial MANO 3D coordinates of joints and mesh vertices to the image-level feature vector $\mathcal{F}$ to obtain the new joint feature matrix $\mathcal{F}_J \in \mathbb{R}^{2051\times K}$ and new vertex feature matrix $\mathcal{F}_V \in \mathbb{R}^{2051\times\mathcal{V}}$. The initial MANO 3D coordinates are obtained by sending zero vectors of pose and shape to the MANO model. Unlike the traditional Transformer method that is only dependant on a self-attention mechanism, we also exploit the correlation between 3D joints and 3D vertices via our cross-attention module. In our cross-attention module, we take 3D vertex features as query and 3D joint features as key and value to model their correlation. With $S_{cross} \in \mathbb{R}^{\mathcal{V}\times K}$ representing the correlation map, we have

$$S_{cross} = softmax\left(\frac{q_{cross}k_{cross}^T}{\sqrt{d_{cross}}}\right) \tag{9}$$

where $q_{cross} = W_q\mathcal{F}_V$ and $k_{cross} = W_k\mathcal{F}_J$ denote the query and key embedding. d_{cross} denotes the feature dimension of the key k_{cross}. The output of the cross-attention module is computed as $\mathcal{F}_{cross} = W_v\mathcal{F}_J S_{cross}$. $W_q, W_k, W_v \in \theta$ are trainable parameters applied for query, key and value embedding. We further add a residual connection between $\mathcal{F}_{cross}$ and the primary feature $\mathcal{F}_V$, which preserves essential information for mesh vertices regression.

In our self-attention module, we extract the query q_{self}, key k_{self} and value v_{self} from $\mathcal{F}_V$ by introducing $W_{q'}, W_{k'}, W_{v'}$. We use softmax to generate the correlation map after matrix multiplication of q_{self} and k_{self}:

$$S_{self} = softmax\left(\frac{q_{self}k_{self}^T}{\sqrt{d_{self}}}\right), \tag{10}$$

where the self-attention module output is computed as $\mathcal{F}_{self} = W_{v'}\mathcal{F}_J S_{cross}$. We add a residual connection between $\mathcal{F}_{self}$ and the output feature $\mathcal{F}_{cross}$ of the previous cross-attention module. Finally two MLP layers are adopted to estimate μ and σ to represent the gaussian probability distribution of 3D coordinates of the mesh vertices.

3.4. Occlusion-aware Hand Texture Regression

The hand texture estimation has recently received more attention due to its significant application in immersive vir-

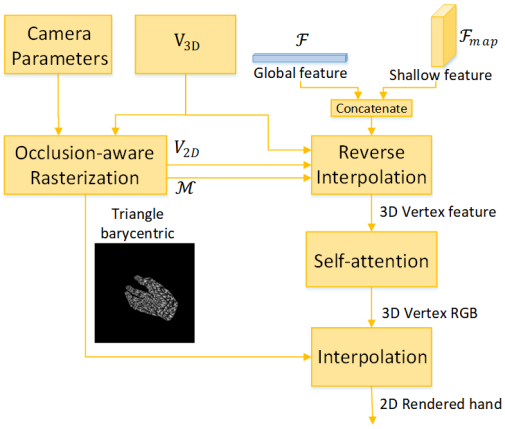

Figure 4. The Occlusion-aware Hand Texture Regression module.

tual reality. However existing hand texture model is unable to generate high-fidelity hand texture and be aware of occlusion. To address above problems, we propose an Occlusion-aware Hand Texture Regression. As shown in Figure 4, we regress per-vertex RGB values to represent hand texture. To achieve this goal, we first leverage a rasterizer to implement the mapping between world coordinates and camera pixel coordinates. In our rasterization, a manifold triangle mesh with vertices predicted by our AMVUR model is first created to represent the hand surface. All triangles are then projected to the 2D space, meanwhile per-pixel auxiliary data including barycentric coordinates and triangle IDs are preserved in the rasterization operation. We retrieve visible triangle IDs and create an binary occlusion mask by looking up the three vertices from each visible triangle. Unlike traditional interpolation in Render, which expands per-vertex data from 3D to pixel space, our reverse interpolation is proposed to construct per-vertex data from pixel to 3D space. The extracted feature on the vertex V_{2D}^m is

$$\mathcal{H}_m = \mathcal{B}\left(V_{2D}^m, \mathcal{F} \| \mathcal{F}_{map} \| V_{3D}^m\right), \tag{11}$$

where $\mathcal{B}(V, X)$ interpolates on the projected 2D point V from tensor X via bilinear interpolation. $\mathcal{F}_{map}$ is a feature map extracted from a shallow layer of the backbone CNN, which preserves rich pixel-level information. $\|$ is the concatenation operation. Afterwards, the 3D Vertex feature $\mathcal{H}_m$ is fed into the self-attention layer described in Eq. 10, followed by a common interpolation for generating the 2D rendered hand image. We adopt the differentiable rasterization and interpolation from [30], allowing our Occlusion-aware Texture Regression model to be trained in an end-to-end manner. Our loss function for training our texture regression model is

$$\mathcal{L}_{tex} = \left\| I_{rend} \odot \mathcal{M} - I \odot \mathcal{M} \right\|_2, \tag{12}$$

where I_{rend} is the output image of our texture regression model, and $\odot$ denotes elementwise multiplication. $\mathcal{M}$ is a 2D binary occlusion Matrix that indicates the hand region on the texture map, which is obtained from our rasterization.

4. Experiments

We evaluated our method on three widely used datasets for 3D hand reconstruction from a single RGB image: HO3Dv2 [22], HO3Dv3 [23] and FreiHAND [24]. We present evaluation of the performance of our method in two scenarios: supervised and weakly-supervised training. Our results on the HO3Dv2 and HO3Dv3 datasets were evaluated anonymously using the online evaluation system.[1][2]. We also present ablation studies to evaluate the importance of each component of the proposed method.

4.1. Datasets

HO3Dv2 [22] is a hand-object interaction dataset which includes significant occlusions. The dataset consists of 77,558 images from 68 sequences, which are split into 66,034 images (from 55 sequences) for training and 11,524 images (from 13 sequences) for testing. Each image contain one of 10 persons manipulating one of 10 objects.

HO3Dv3 [23] is a recently released hand-object interaction dataset with more images and more accurate annotations than HO3Dv2. It contains 103,462 hand-object 3D pose annotated RGB images, which are split into 83,325 training images and 20,137 testing images.

4.2. Metrics and Implementation Details

Evaluation Metrics. To quantitatively evaluate our 3D hand reconstruction, we report average Euclidean distance in millimeters (mm) between the estimated 3D joints/mesh and ground truth (MPJPE/MPVPE), and the area under their percentage of correct keypoint (PCK) curves (AUC_J/AUC_v) for the thresholds between 0mm and 50mm. For the 3D mesh, we also report F-score of vertices at distance thresholds of 5mm and 15mm by F_5 and F_{15}, respectively. Following previous work [9, 10, 13, 14], we report 3D metrics after procrustes alignment.

Implementation details. All experiments are conducted on two NVidia GeForce RTX 3090 Ti GPUs. We use the Adam optimizer [36] to train the network with batch size of 32. For all supervised experiments, we use ResNet50 [37] as our backbone CNN, following [11, 14, 16, 26, 32, 35]. We use EfficientNet-B0 [38] as our backbone in the weakly-supervised setting, following [13]. We extract the shallow $\mathcal{F}_{map}$ and global $\mathcal{F}$ features from the first convolution layer and the last fully connected layer before the classification layer of the backbone model, respectively. The code is available on github: https://github.com/ZhehengJiangLancaster/AMVUR.

[1]HO3Dv2:https://codalab.lisn.upsaclay.fr/competitions/4318?
[2]HO3Dv3:https://codalab.lisn.upsaclay.fr/competitions/4393?

Table 1. Hand reconstruction performance compared with SOTA methods on HO3Dv2 after Procrustes alignment. [31]* develops a synthetic dataset with 1520 poses and 216 viewpoints during training to overcome the long-tailed distribution of hand pose and viewpoint.

Training Scheme	Method	Category	$AUC_J \uparrow$	MPJPE↓	$AUC_V \uparrow$	MPVPE↓	$F_5 \uparrow$	$F_{15} \uparrow$
Supervised	Liu et al. [26]	Model-based	0.803	9.9	0.810	9.5	0.528	0.956
	HandOccNet [14]	Model-based	0.819	9.1	0.819	8.8	0.564	0.963
	I2UV-HandNet [32]	Model-based	0.804	9.9	0.799	10.1	0.500	0.943
	Hampali et al. [22]	Model-based	0.788	10.7	0.790	10.6	0.506	0.942
	Hasson et al. [33]	Model-based	0.780	11.0	0.777	11.2	0.464	0.939
	ArtiBoost [34]	Model-based	0.773	11.4	0.782	10.9	0.488	0.944
	Pose2Mesh [11]	Model-free	0.754	12.5	0.749	12.7	0.441	0.909
	I2L-MeshNet [35]	Model-free	0.775	11.2	0.722	13.9	0.409	0.932
	METRO [12]	Model-free	0.792	10.4	0.779	11.1	0.484	0.946
	Chen et al. [31]*	Model-free	-	9.2	-	9.4	0.538	0.957
	Keypoint Trans [16]	Model-free	0.786	10.8	-	-	-	-
	Ours(prior-net)	Model-based	0.783	10.9	0.77	11.5	0.460	0.936
	Ours(AMVUR)	Model-free	0.814	9.3	0.813	9.4	0.533	0.958
	Ours(final)	Probabilistic	**0.835**	**8.3**	**0.836**	**8.2**	**0.608**	**0.965**
Weakly-Supervised	S^2HAND [13]	Model-based	0.765	-	0.769	-	0.44	0.93
	Ours(prior-net)	Model-based	0.752	12.4	0.760	12.0	0.417	0.925
	Ours(AMVUR)	Model-free	0.778	10.8	0.698	15.1	0.375	0.907
	Ours(final)	Probabilistic	**0.787**	**10.3**	**0.784**	**10.8**	**0.48**	**0.949**

Table 2. Hand reconstruction performance compared with SOTA methods on HO3Dv3 dataset after Procrustes alignment.

Training Scheme	Method	Category	$AUC_J \uparrow$	MPJPE↓	$AUC_V \uparrow$	MPVPE↓	$F_5 \uparrow$	$F_{15} \uparrow$
Supervised	ArtiBoost [34]	Model-based	0.785	10.8	0.792	10.4	0.507	0.946
	Keypoint Trans [16]	Model-free	0.785	10.9	-	-	-	-
	Ours(prior-net)	Model-based	0.780	11.3	0.781	11.0	0.471	0.931
	Ours(AMVUR)	Model-free	0.803	9.8	0.811	9.7	0.528	0.953
	Ours	Probabilistic	**0.826**	**8.7**	**0.834**	**8.3**	**0.593**	**0.964**
Weakly-Supervised	S^2HAND [13]	Model-based	0.769	11.5	0.778	11.1	0.448	0.932
	Ours(prior-net)	Model-based	0.759	12.1	0.763	11.9	0.422	0.921
	Ours(AMVUR)	Model-free	0.778	10.9	0.724	13.6	0.403	0.904
	Ours(final)	Probabilistic	**0.789**	**10.5**	**0.785**	**10.7**	**0.475**	**0.944**

4.3. Comparison with SOTA Methods

We compare our supervised and weakly supervised methods against existing state-of-the-art methods on HO3Dv2 and HO3Dv3 in Tables 1 and 2 respectively after applying Procrustes alignment on their results. We present results on FreiHAND in the Supplementary Material. We conduct experiments on three settings of our proposed model: **(1) Ours(prior-net)**, where we individually train the MANO prior model, **(2) Ours(AMVUR)**, where we individually train the proposed Attention-based Mesh Vertices Uncertainty Regression, and **(3) Ours(final)**, where the MANO prior-net is jointly trained with AMVUR. As shown in Tables 1 and 2, our probabilistic method achieves the best results across all metrics for both the supervised and unsupervised scenarios. In the weakly-supervised setting, our approach not only achieves the best performance compared to the other state-of-the-art weakly-supervised approaches, but also outperforms some of supervised approaches such as [11,35]. It is interesting to see that even though our AMVUR outperforms the other state-of-the-art Model-free approaches [11, 12, 32] in the supervised training scheme, its contribution is lower in the weakly-supervised training scheme due to the increasing solution space of mesh reconstruction. Evaluation before Procrustes alignment is reported in the Supplementary Material. Figure 5 shows that our probabilistic model generates more accurate hand pose and shape than the other state-of-the-art methods on the HO3Dv2 dataset. Despite some hands being severely occluded, our probabilistic model produces better results.

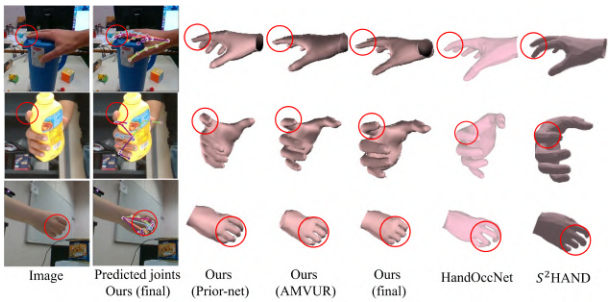

Figure 5. Qualitative comparison of the proposed models and SOTA 3D hand mesh estimation methods HandOccNet [14] and S^2HAND [13] on HO3Dv2.

Table 3. Impacts of loss terms in the supervised training scheme on HO3Dv2 dataset after Procrustes alignment.

Loss terms				MPJPE↓	MPVPE↓	$F_5 \uparrow$	$F_{15} \uparrow$
$\mathcal{L}_{V_{3D}}$	$\mathcal{L}_{J_{3D}}$	$\mathcal{L}_C$	$\mathcal{L}_{J_{2D}}$				
✓				9.7	8.4	0.596	0.960
✓	✓			8.5	8.3	0.605	0.964
✓	✓	✓		8.6	8.3	0.606	0.963
✓	✓	✓	✓	**8.3**	**8.2**	**0.608**	**0.965**

Table 4. Impacts of loss terms in the weakly-supervised training scheme on HO3Dv2 dataset after Procrustes alignment.

Loss terms				MPJPE↓	MPVPE↓	F_5 ↑	F_{15} ↑
$\mathcal{L}_{J_{2D}}$	D_{KL}^{C}	$D_{KL}^{V_{3D}}$	$D_{KL}^{J_{3D}}$				
✓				11.9	11.8	0.434	0.931
✓	✓			11.6	11.5	0.442	0.935
✓	✓	✓		11.1	11.0	0.46	0.947
✓	✓	✓	✓	**10.3**	**10.8**	**0.48**	**0.949**

4.4. Ablation Study

To verify the impact of each proposed component, we conduct extensive ablation experiments on HO3Dv2.

4.4.1 Effect of Each Loss Term

In Table 3, we present the ablation study for the supervised training scheme, where the significant contributions of $\mathcal{L}_{V3D}$ and $\mathcal{L}_{J3D}$ to the 3D mesh reconstruction are seen clearly. It is not surprising to see that $\mathcal{L}_C$ and $\mathcal{L}_{J2D}$ do not give a significant contribution to performance improvement in the supervised training scheme, since their purpose is to help learn the 2D projection and rendering. In contrast to the supervised training scheme, we only use 2D joints annotation to reconstruct the hand mesh the in weakly-supervised training setting. So our baseline in Table 4 only uses $\mathcal{L}_{J2D}$ without any other constraint or prior knowledge. As shown in Table 4, using D_{KL}^{C} , $D_{KL}^{V_{3D}}$ and $D_{KL}^{J_{3D}}$ consistently improves all metrics, demonstrating their benefits. Specifically, $D_{KL}^{V_{3D}}$ and $D_{KL}^{J_{3D}}$ bring significant improvements to the mesh (MPVPE, F_5 and F_{15}) and joint (MPJPE) reconstruction, respectively.

4.4.2 Analysis of AMVUR model.

In terms of regressing the 3D vertex coordinates, a naive approach is to regress vertex coordinates with a series of fully connected layers on the top of our CNN backbone. In experiment A of Table 5, we construct our baseline by replacing AMVUR with fully connected layers to estimate the probabilistic distribution of the vertices. From Table 5, AMVUR clearly outperforms this design, demonstrating the importance of capturing the correlation between joints and mesh vertices during regression. Each major component of our AMVUR, i.e., cross-attention, self-attention and positional encoding, is evaluated in experiments B,C and D, respectively. In B and C, all tokens use different indices to describe their locations following the traditional Transformer. We observe that cross-attention and self-attention are critical for performance improvement. Positional encoding further improves the performance of our approach.

4.4.3 Comparison of Texture Estimation Model

To quantitatively measure the quality of the estimated hand textures, we use the SSIM [39] and PSNR [40] on the hand region. Different from S^2HAND [13], we propose a more intelligent strategy to regress hand texture from the combination of the global and shallow features, which leads to better performance in terms of SSIM and PSNR in Table 6. From Fig. 6, we can see that the results of S^2HAND [13] lack fine details and have larger color differences from the input. In contrast, our approach has better capability in reconstructing high-fidelity hand textures.

Table 5. Analysis of the AMVUR.

Exp.	Setup	MPJPE↓	MPVPE↓	F_5 ↑	F_{15} ↑
A	Baseline	11.4	11.4	0.462	0.932
B	Self-attention	10.5	10.9	0.496	0.948
C	B+Cross-attention	8.9	8.7	0.581	0.958
D	C+Positional	**8.3**	**8.2**	**0.608**	**0.965**

Figure 6. Qualitative comparison of our proposed model and SOTA texture regression model S^2HAND [13] on HO3Dv2.

Table 6. Quantitative comparison of our model and SOTA texture regression model S^2HAND [13] on HO3Dv2.

Method	PSNR↑	SSIM↑
S^2HAND [13]	27.8	0.973
Ours	**41.7**	**0.994**

5. Conclusion

In this paper, we have proposed a novel probabilistic model for 3D hand reconstruction from single RGB images, capable of reconstructing not only the 3D joints and mesh vertices but also the texture of the hand accurately despite the presence of severe occlusions. Our approach includes several novelties, including our AMVUR approach, which relaxes the heavy parameter space reliance of the MANO model, allowing more accurate reconstruction. This is trained with our prior-net and includes an attention mechanism to capture correlation between joints and mesh vertices in 3D space. We demonstrated that our proposed probabilistic model achieves state-of-the-art accuracy in fully-supervised and weakly-supervised training. Moreover, we proposed an occlusion-aware Hand Texture Regression model for accurate texture reconstruction.

Acknowledgments. The work is supported by European Research Council under the European Union's Horizon 2020 research and innovation programme (GA No 787768).

References

[1] N. Conci, P. Ceresato, and F. G. De Natale, "Natural human-machine interface using an interactive virtual blackboard," in *2007 IEEE International Conference on Image Processing*, vol. 5, pp. V–181, IEEE, 2007. 1

[2] R. Yin, D. Wang, S. Zhao, Z. Lou, and G. Shen, "Wearable sensors-enabled human–machine interaction systems: from design to application," *Advanced Functional Materials*, vol. 31, no. 11, p. 2008936, 2021. 1

[3] S. Jung and C. E. Hughes, "Body ownership in virtual reality," in *2016 International Conference on Collaboration Technologies and Systems (CTS)*, pp. 597–600, IEEE, 2016. 1, 2

[4] J. Wang, F. Mueller, F. Bernard, S. Sorli, O. Sotnychenko, N. Qian, M. A. Otaduy, D. Casas, and C. Theobalt, "Rgb2hands: real-time tracking of 3d hand interactions from monocular rgb video," *ACM Transactions on Graphics (ToG)*, vol. 39, no. 6, pp. 1–16, 2020. 1

[5] S. Han, B. Liu, R. Cabezas, C. D. Twigg, P. Zhang, J. Petkau, T.-H. Yu, C.-J. Tai, M. Akbay, Z. Wang, *et al.*, "Megatrack: monochrome egocentric articulated hand-tracking for virtual reality," *ACM Transactions on Graphics (ToG)*, vol. 39, no. 4, pp. 87–1, 2020. 1

[6] F. Mueller, M. Davis, F. Bernard, O. Sotnychenko, M. Verschoor, M. A. Otaduy, D. Casas, and C. Theobalt, "Real-time pose and shape reconstruction of two interacting hands with a single depth camera," *ACM Transactions on Graphics (ToG)*, vol. 38, no. 4, pp. 1–13, 2019. 1

[7] X. Liang, A. Angelopoulou, E. Kapetanios, B. Woll, R. Al Batat, and T. Woolfe, "A multi-modal machine learning approach and toolkit to automate recognition of early stages of dementia among british sign language users," in *European Conference on Computer Vision*, pp. 278–293, Springer, 2020. 1

[8] N. Kolotouros, G. Pavlakos, and K. Daniilidis, "Convolutional mesh regression for single-image human shape reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4501–4510, 2019. 1, 2

[9] L. Ge, Z. Ren, Y. Li, Z. Xue, Y. Wang, J. Cai, and J. Yuan, "3d hand shape and pose estimation from a single rgb image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10833–10842, 2019. 1, 3, 5, 6

[10] D. Kulon, R. A. Guler, I. Kokkinos, M. M. Bronstein, and S. Zafeiriou, "Weakly-supervised mesh-convolutional hand reconstruction in the wild," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4990–5000, 2020. 1, 2, 3, 5, 6

[11] H. Choi, G. Moon, and K. M. Lee, "Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose," in *European Conference on Computer Vision*, pp. 769–787, Springer, 2020. 1, 2, 6, 7

[12] K. Lin, L. Wang, and Z. Liu, "End-to-end human pose and mesh reconstruction with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1954–1963, 2021. 1, 2, 4, 7

[13] Y. Chen, Z. Tu, D. Kang, L. Bao, Y. Zhang, X. Zhe, R. Chen, and J. Yuan, "Model-based 3d hand reconstruction via self-supervised learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10451–10460, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[14] J. Park, Y. Oh, G. Moon, H. Choi, and K. M. Lee, "Handoccnet: Occlusion-robust 3d hand mesh estimation network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1496–1505, 2022. 1, 2, 3, 4, 5, 6, 7

[15] A. Boukhayma, R. d. Bem, and P. H. Torr, "3d hand shape and pose from images in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10843–10852, 2019. 1, 2

[16] S. Hampali, S. D. Sarkar, M. Rad, and V. Lepetit, "Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11090–11100, 2022. 1, 2, 6, 7

[17] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *ACM Transactions on Graphics*, vol. 36, no. 6, 2017. 1, 2, 4

[18] A. Spurr, U. Iqbal, P. Molchanov, O. Hilliges, and J. Kautz, "Weakly supervised 3d hand pose estimation via biomechanical constraints," in *European Conference on Computer Vision*, pp. 211–228, Springer, 2020. 1, 2

[19] M. de La Gorce, N. Paragios, and D. J. Fleet, "Model-based hand tracking with texture, shading and self-occlusions," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2008. 2

[20] A. Tkach, A. Tagliasacchi, E. Remelli, M. Pauly, and A. Fitzgibbon, "Online generative model personalization for hand tracking," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 6, pp. 1–11, 2017. 2

[21] J. Taylor, V. Tankovich, D. Tang, C. Keskin, D. Kim, P. Davidson, A. Kowdle, and S. Izadi, "Articulated distance fields for ultra-fast tracking of hands interacting," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, pp. 1–12, 2017. 2

[22] S. Hampali, M. Rad, M. Oberweger, and V. Lepetit, "Honnotate: A method for 3d annotation of hand and object poses," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3196–3206, 2020. 2, 6, 7

[23] S. Hampali, S. D. Sarkar, and V. Lepetit, "Ho-3d_v3: Improving the accuracy of hand-object annotations of the ho-3d dataset," *arXiv preprint arXiv:2107.00887*, 2021. 2, 6

[24] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. Argus, and T. Brox, "Freihand: A dataset for markerless capture of hand pose and shape from single rgb images," in *Proceedings*

of the IEEE/CVF International Conference on Computer Vision, pp. 813–822, 2019. 2, 6

[25] X. Zhang, Q. Li, H. Mo, W. Zhang, and W. Zheng, "End-to-end hand mesh recovery from a monocular rgb image," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2354–2364, 2019. 2

[26] S. Liu, H. Jiang, J. Xu, S. Liu, and X. Wang, "Semi-supervised 3d hand-object poses estimation with interactions in time," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14687–14697, 2021. 2, 6, 7

[27] N. Qian, J. Wang, F. Mueller, F. Bernard, V. Golyanik, and C. Theobalt, "Html: A parametric hand texture model for 3d hand reconstruction and personalization," in *European Conference on Computer Vision*, pp. 54–71, Springer, 2020. 2

[28] H. Dai, N. Pears, W. A. Smith, and C. Duncan, "A 3d morphable model of craniofacial shape and texture variation," in *Proceedings of the IEEE international conference on computer vision*, pp. 3085–3093, 2017. 2

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017. 5

[30] S. Laine, J. Hellsten, T. Karras, Y. Seol, J. Lehtinen, and T. Aila, "Modular primitives for high-performance differentiable rendering," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, pp. 1–14, 2020. 6

[31] X. Chen, Y. Liu, Y. Dong, X. Zhang, C. Ma, Y. Xiong, Y. Zhang, and X. Guo, "Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20544–20554, 2022. 7

[32] P. Chen, Y. Chen, D. Yang, F. Wu, Q. Li, Q. Xia, and Y. Tan, "I2uv-handnet: Image-to-uv prediction network for accurate and high-fidelity 3d hand mesh modeling," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12929–12938, 2021. 6, 7

[33] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11807–11816, 2019. 7

[34] L. Yang, K. Li, X. Zhan, J. Lv, W. Xu, J. Li, and C. Lu, "Artiboost: Boosting articulated 3d hand-object pose estimation via online exploration and synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2750–2760, 2022. 7

[35] G. Moon and K. M. Lee, "I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image," in *European Conference on Computer Vision*, pp. 752–768, Springer, 2020. 6, 7

[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. 6

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. 6

[38] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*, pp. 6105–6114, PMLR, 2019. 6

[39] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004. 8

[40] A. Hore and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *2010 20th international conference on pattern recognition*, pp. 2366–2369, IEEE, 2010. 8