2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) | 979-8-3315-5511-5/26/$31.00 ©2026 IEEE | DOI: 10.1109/WACV61042.2026.00683

Enhancing Monocular 3D Hand Reconstruction with Learned Texture Priors

Giorgos Karvounas[1,3] gkarv@ics.forth.gr
Nikolaos Kyriazis[1] kyriazis@ics.forth.gr
Iason Oikonomidis[1] oikonom@ics.forth.gr
Georgios Pavlakos[2] pavlakos@cs.utexas.edu
Antonis A. Argyros[1,3] argyros@ics.forth.gr

[1]ICS-FORTH, [2]University of Texas at Austin, [3]University of Crete

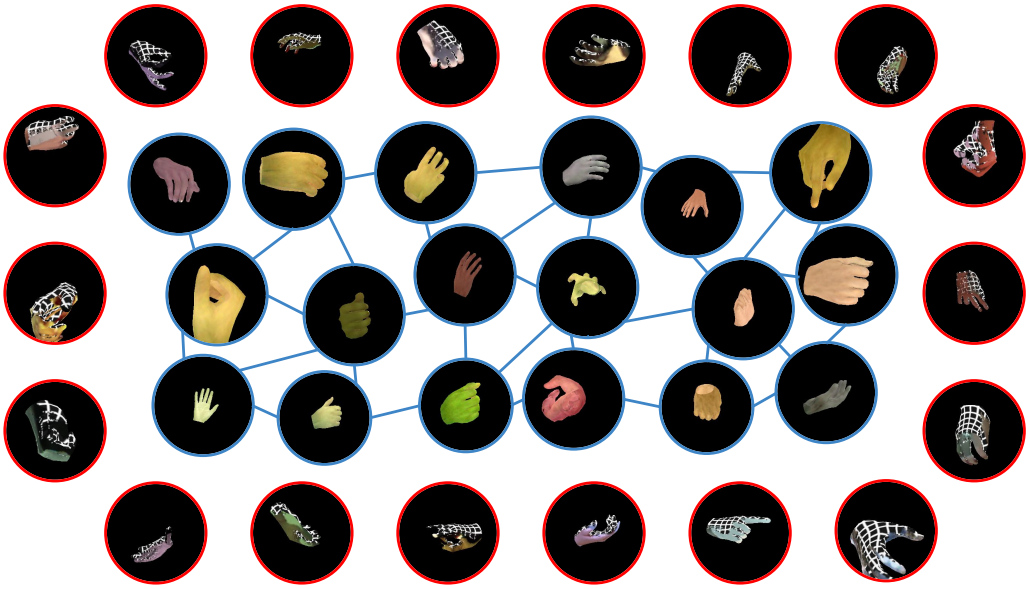

Figure 1. *In-the-wild* hand images provide the richest, but most incomplete and noisy input for building texture priors (samples outlined red, grid regions indicate missing information in the monocular setting). Our method is able to consolidate such data, enabling robust full texture suggestions (samples outlined blue).

Abstract

We revisit the role of texture in monocular 3D hand reconstruction, not as an afterthought for photorealism, but as a dense, spatially grounded cue that can actively support pose and shape estimation. Our observation is simple: even in high-performing models, the overlay between predicted hand geometry and image appearance is often imperfect, suggesting that texture alignment may be an underused supervisory signal. We propose a lightweight texture module that embeds per-pixel observations into UV texture space and enables a novel dense alignment loss between predicted and observed hand appearances. Our approach assumes access to a differentiable rendering pipeline and a model that maps images to 3D hand meshes with known topology, allowing us to back-project a textured hand onto the image and perform pixel-based alignment. The module is self-contained and easily pluggable into existing reconstruction pipelines. To isolate and highlight the value of texture-guided supervision, we augment HaMeR [47], a high-performing yet unadorned transformer architecture for 3D hand pose estimation. The resulting system improves both accuracy and realism, demonstrating the value of appearance-guided alignment in hand reconstruction. Code, models and weights are available at https://gkarv.github.io/hand-texture-module

1. Introduction

Reconstructing 3D hand pose and shape from a single image is a long-standing scientific challenge. Despite its under-constrained nature, accurate monocular estimation is criti-

2642-9381/26/$31.00 ©2026 IEEE
DOI 10.1109/WACV61042.2026.00683

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

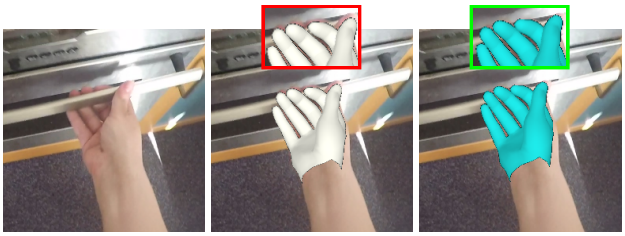

Figure 2. What is otherwise a 3D estimation of high quality, clearly has room for improvement, as indicated by the imperfect match of the prediction and the observation (middle, estimated by HaMeR [47]). This can be remedied (right, our method).

cal across domains. In healthcare, it supports remote physiotherapy and motor assessment. In robotics, it enables teleoperation and human-robot collaboration. In education, it powers skill transfer and simulation. In consumer technology, it drives AR, gaming, and social interaction. While multi-camera setups offer precision, they are impractical at scale. Monocular methods offer portability and accessibility, but only if they achieve sufficient accuracy. This makes the problem both scientifically important and practically impactful.

Our starting point is a simple observation. Even high-performing models often produce hand reconstructions that do not perfectly align with the observed image appearance (see Fig. 2). This discrepancy suggests that current training signals, while effective, do not fully exploit available information. Motivated by this observation, we examine whether adding explicit, spatially grounded supervision based on image appearance can improve accuracy. Instead of treating texture as a byproduct, we use it as a source of alignment feedback. By linking pixels to predicted surfaces, we provide the model with a direct mechanism to correct its estimates.

With the main driving observation being misalignment, we examine a route towards dense alignment at training time. We approach this with a pixel-wise alignment loss between the observed appearance and the predicted hand surface. Computing such a loss requires three components: a 3D hypothesis of the hand (provided by an image-to-3D method), a model of its texture (the focus of our work), and a rendering mechanism (readily available through differentiable rendering libraries, e.g. [50]) to project the textured mesh back onto the image. This setup allows us to transform discrepancies between prediction and observation into a usable training signal, and quantify its impact.

Dense, high-quality hand texture models are not readily available. Existing annotated datasets tend to provide full texture supervision only under multiview or studio conditions, which do not scale and often lack appearance diversity. In contrast, unannotated monocular data is abundant but provides only partial and imperfect observations. To bridge this gap, we construct texture supervision directly from monocular images using predictions from a pretrained 3D hand model. By projecting visible pixels onto the mesh surface and collecting them in UV space, we generate partial, view-specific texture maps without requiring ground truth UVs. These partial maps are then embedded into a learnable texture representation (see Fig. 1). To that end, we employ a transformer architecture that attends directly to sparse UV-RGB observations. This module accepts an arbitrary number of input pixels and produces a complete, embedded texture. The flexibility of this design allows it to operate across varying levels of visibility and density, making it well-suited for learning from in-the-wild images. This approach eliminates the need for dense manual annotation, supports training from uncurated monocular data, and enables scalable learning from partial signals. We integrate this module into an existing monocular reconstruction pipeline by attaching it to a model that maps an input image to a 3D hand mesh (see Fig. 3). During training, the mesh is projected back onto the image, and the visible surface points are used to extract UV-RGB pairs. Although the initial 3D prediction may contain errors, the transformer embeds the resulting sparse observations into a complete texture representation. This enables the model to generate coherent textures from incomplete inputs, handling diverse visibilities and observation densities typical of in-the-wild data.

Our main contributions are:

- We recast texture from "nice-to-have output" to a dense alignment signal: via differentiable rendering we link image pixels to the predicted mesh in UV space and optimize a photometric alignment loss to correct pose/shape.
- A plug-in texture module that takes arbitrary sparse UV-RGB samples and, with pixel-level attention and an upsampling head, synthesizes a coherent full texture embedding robust to visibility and sparsity.
- A scalable bootstrapping procedure that mines UV-RGB correspondences from monocular images using a pretrained reconstructor, enabling self-supervised training without ground-truth textures or multiview capture.
- The module attaches to any renderable image-to-mesh hand pipeline, provides texture-guided supervision during training, and can be frozen or removed at inference yielding measurable accuracy gains with no runtime overhead.

2. Related Work

Hand pose and shape estimation: A wide range of methods have been proposed to estimate hand pose and shape from visual data, often relying on parametric models or voxel-based representations. Approaches such as HAMBA [17], Mesh Graphormer [35] and HandOS [6] leverage transformer architectures and mesh-based representations to recover detailed hand shape and pose. Self-supervised and weakly-supervised learning strategies have

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

also been explored to reduce reliance on annotated data [41, 54]. Earlier methods like [1, 2, 19] focus on regressing 3D hand parameters or directly estimating mesh vertices from single images. Voxel-based techniques, such as HandVoxNet and its improved variant HandVoxNet++ [38, 39], utilize volumetric occupancy grids to represent hand geometry in 3D space. Luan et al. [37] introduced a scalable frequency-domain framework that leverages graph-based feature mapping and frequency decomposition to reconstruct 3D hand meshes with fine-grained details. Collectively, these methods demonstrate ongoing progress in balancing accuracy, generalizability, and data efficiency in hand pose and shape estimation.

Hand texture models: Modeling realistic hand textures is essential for photorealistic rendering and accurate hand appearance reconstruction. HTML [49] introduces a generative framework that synthesizes detailed hand textures from monocular images, enabling high-fidelity appearance modeling. Handy [48] presents a neural parametric hand model that jointly captures geometry and texture in a coherent manner, enhancing realism across diverse viewing conditions. NIMBLE [33] further extends this line of work by integrating a unified model of hand shape, pose, and appearance, offering controllable and expressive hand synthesis. These approaches collectively contribute to the development of comprehensive models that go beyond geometry to faithfully represent hand appearance in diverse scenarios.

Hand reconstruction: Methods based on convolutional neural networks [32, 62] build on top of existing texture models to incorporate photometric loss. Chen et al. [7] propose to regress hand texture via MLP layers by taking a global feature vector as input. However, this approach handles occlusions poorly. Recent approaches have explored generative and reconstruction techniques for modeling human hands in 3D. Diffusion-based models [10, 44], have shown promising results in synthesizing realistic hand poses and appearances through probabilistic generative pipelines. Neural Radiance Fields (NeRFs) [40] have also been adopted for hand modeling, offering view-consistent and detailed reconstructions [12, 13, 22, 61]. These methods leverage volumetric rendering and neural representations to capture complex hand geometries and articulation. However, despite their effectiveness, many of these approaches remain computationally expensive and resource-intensive, limiting their practicality in real-time or resource-constrained environments [8, 30, 31]. This motivates the need for more efficient yet accurate hand modeling techniques.

Hand datasets: Researchers use a wide variety of datasets to train hand pose methods. These datasets provide 3D annotations, captured in both indoor and outdoor settings. Featuring a multi-camera set up, InterHand2.6M [43] focuses on the strong interaction of hands. FreiHAND [63], HO-3D [23] and DexYCB [3] are similarly captured under multi-camera studio settings, with emphasis on hand-object interactions. It is common to augment the training process using also datasets captured in the Panoptic Studio [29, 53, 57]. Additionally, AssemblyHands [45] provides 3D ground truth while participants perform various assembly and disassembly tasks, again in a multi-camera setting. Despite the importance of 3D annotation, 2D annotation still plays a critical role. Therefore, large scale datasets like COCO-WholeBody [28] and Halpe [18] are commonly used [6, 47, 60]. HInt [47] was recently introduced, containing 2D annotated images from 3 egocentric datasets, NEWDAYS [9], VISOR [16] and Ego4D [20].

3. Method

Our goal is to improve monocular 3D hand reconstruction by integrating appearance-based supervision into an existing geometry-driven pipeline. To this end, we augment a baseline model that predicts 3D hand meshes from images (referred to as **BaseNet**) with a texture module that reconstructs UV-space textures from sparse, view-specific observations. This enables us to render a textured hand back into the input image and define a dense photometric loss that guides geometric refinement. The method consists of three components: the base reconstruction model, the texture model, and the associated training procedures. We describe each in turn.

3.1. Hand Texture Model

The texture model transforms sparse and potentially noisy appearance observations into a coherent UV-space representation, enabling dense, image-space supervision that improves the alignment and accuracy of the predicted hand geometry.

The texture model operates on a set of UV-space appearance observations $\mathcal{S} = \{(u_i, c_i)\}_{i=1}^{L}$, where each $u_i \in [0,1]^2$ denotes a UV coordinate and $c_i \in \mathbb{R}^3$ is the corresponding RGB color sampled from the input image. Given this variable-length input, the model predicts a dense UV texture map $\hat{\mathcal{T}} \in \mathbb{R}^{3\times H_T \times W_T}$, aligned with the fixed UV layout of the hand mesh.

In terms of architecture, we are faced with the following challenge: we are required to operate on sparse, pixel-perfect, variable-length input, and on a structured, image-based output. To that end, we employ a transformer-based encoder that attends to pixels, coupled with a convolutional upscaler, to yield the final full hand texture.

Transformer-Based Encoder: Given the sparse observations $\mathcal{S}$, we embed the RGB information via a 1×1 con-

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

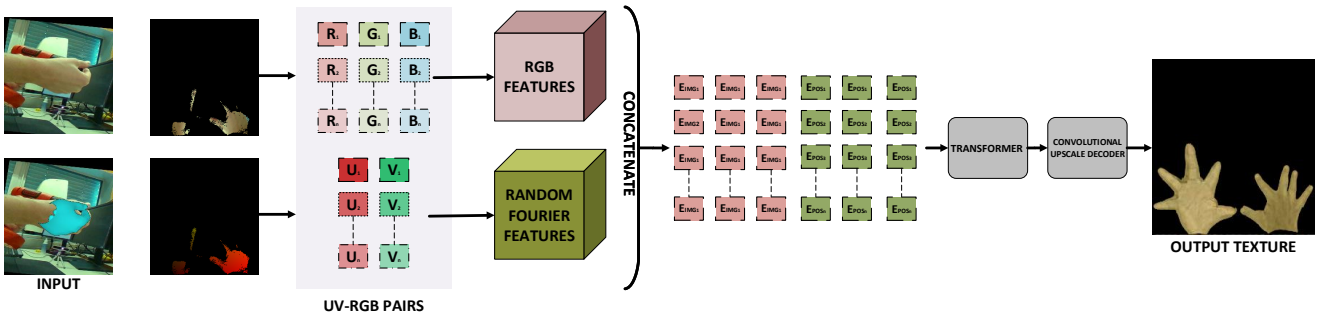

Figure 3. Our texture model consolidates variable-length, sparse, view-specific UV-RGB observations, extracted from a predicted 3D hand mesh and camera (BaseNet), into a complete hand texture. RGB values are embedded via a convolutional layer, and UV coordinates are encoded using Random Fourier Features [55]. These feature pairs are concatenated and processed by a transformer decoder. A convolutional upscaling decoder synthesizes the final, dense texture map.

volutional layer to produce an initial feature map. This is reshaped into a sequence of L tokens of dimension D:

$$\mathbf{E}_{\text{img}} = \text{Conv1D}(S) \in \mathbb{R}^{L \times D}. \tag{1}$$

To inject spatial context, we use a random Fourier encoding [55] $\gamma(\cdot)$ applied to UV coordinates, that corresponds to each pixel, inputs $C \in \mathbb{R}^{2 \times L}$, yielding positional encodings $\mathbf{E}_{\text{pos}} \in \mathbb{R}^{L \times D'}$:

$$\mathbf{E}_{\text{pos}} = \gamma(C). \tag{2}$$

We concatenate these encodings:

$$\mathbf{E} = [\mathbf{E}_{\text{img}} \,\|\, \mathbf{E}_{\text{pos}}] \in \mathbb{R}^{L \times (D+D')}. \tag{3}$$

To aggregate global features, we introduce a set of T learned query tokens $Q \in \mathbb{R}^{T \times (D+D')}$ that attend to $\mathbf{E}$ using a transformer. This is implemented using the TransformerDecoder implementation from PyTorch with $\mathbf{K}$ layers:

$$\mathbf{Z} = \text{Transformer}(Q, \mathbf{E}^\top), \tag{4}$$

where $\mathbf{Z} \in \mathbb{R}^{T \times D''}$ contains the encoded scene information and $\mathbf{E}$ serves as key and value.

Convolutional Upscaling Decoder: We employ a decoder to convert the latent representation $\mathbf{Z}$ into a full-resolution texture. It progressively upsamples the features using pixel shuffling operations [52], which efficiently refine image details while increasing the spatial resolution, ultimately reaching 224×224 pixels.

Each stage refines the feature representation, and the final convolution projects the upscaled features into RGB space, yielding the final hand texture $\hat{T} \in \mathbb{R}^{3 \times 224 \times 224}$ using 5 upscaling layers.

Weakly supervised texture learning: As explained in Sec. 4, the most beneficial fine-tuning strategy involves pre-training the texture model and using it in a fixed inference mode during the subsequent fine-tuning of BaseNet.

We train the texture model in a weakly supervised setting using partial appearance observations extracted from monocular images. For each training image I, we obtain a predicted mesh and camera $(\hat{M}, \hat{c}) = f_{\text{base}}(I)$ and extract a set of UV-RGB samples $\mathcal{S} = \{(u_i, c_i)\}_{i=1}^{L}$ by projecting visible mesh vertices into the image and associating their appearance with UV coordinates. These samples define supervision only over the visible subset of the UV domain. The texture model is trained to reconstruct a full texture $\hat{\mathcal{T}} = T(\mathcal{S})$ that agrees with the observed colors at their corresponding UV positions.

We define a sparse supervision map T^* in UV space by assigning each u_i to its observed color c_i, and a binary mask $\mathcal{M} \in \{0,1\}^{H_T \times W_T}$ indicating which texels are observed. The loss consists of a masked L_1 term and a Fourier-domain consistency loss:

$$\mathcal{L}_{\text{weak}} = \left\| \mathcal{M} \cdot (\hat{\mathcal{T}} - T^*) \right\|_1 + \lambda_{\text{freq}} \cdot \mathcal{L}_{\text{Fourier}}, \tag{5}$$

$$\mathcal{L}_{\text{Fourier}} = \left\| |\mathcal{F}(\hat{\mathcal{T}} \cdot \mathcal{M})| - |\mathcal{F}(T^* \cdot \mathcal{M})| \right\|_1, \tag{6}$$

where $\mathcal{F}(\cdot)$ denotes the discrete Fourier transform applied independently to each channel. This combination encourages both local accuracy in observed regions and global consistency in frequency space, helping the model produce coherent textures even from sparse and incomplete inputs.

The training samples are pseudo-labeled. In part, random samples are drawn directly on the texture space of parametric texture models, *e.g.*, the HTML model [49] in our case. The majority amount to partial textures extracted by back-projecting poses inferred by BaseNet, on real data, including the imperfections.

During training, we randomize the overall color of the texture, to encourage the model not to embed too far away from corner cases, *e.g.* tattoos, gloves, occluders, *etc*.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

3.2. Extending monocular 3D reconstruction

We consider a monocular hand reconstruction system, denoted **BaseNet**, that maps an input image $I \in \mathbb{R}^{3\times H\times W}$ to a 3D hand mesh and camera estimate, written as $(\hat{M}, \hat{c}) = f_{\text{base}}(I)$. We assume that the predicted mesh $\hat{M}$ is defined over a fixed topology with a consistent UV mapping. Using the image, predicted geometry, and camera, we extract UV-RGB correspondences $\mathcal{S} = \mathcal{P}(I, \hat{M}, \hat{c})$ by back-projecting visible surface points into the image. These sparse samples are passed to a texture module, which predicts a dense UV-space texture $\hat{\mathcal{T}} = T(\mathcal{S})$. The textured mesh is rendered using a differentiable renderer, yielding a synthetic image $\hat{I}_{\text{hand}} = \mathcal{R}(\hat{M}, \hat{\mathcal{T}}, \hat{c})$ that is compared to the input appearance. For explanatory clarity, we present the entire supervision pipeline in a simplified and compositional form. The resulting training objective is:

$$\mathcal{L}_{\text{total}} = \underbrace{\mathcal{L}_{\text{base}}}_{\text{BaseNet}} + \lambda_{\text{tex}} \cdot \underbrace{\mathcal{L}_{\text{tex}}(I, \hat{I}_{\text{hand}})}_{\text{Texture path}}. \tag{7}$$

In this work, we instantiate **BaseNet** using HaMeR [47], a high-performing unadorned transformer-based model for monocular 3D hand reconstruction. HaMeR predicts hand pose, shape, and camera parameters, regressing the MANO [51] model from a single image. For differentiable rendering, we use PyTorch3D [50], which provides efficient projection and rasterization of textured meshes. Overall, to get an improved method, we fine-tune BaseNet by also incorporating the aforementioned changes.

3.3. Putting it all together

With our texture model as a sidecar to BaseNet, we perform a fine-tuning task. We re-train a pre-trained BaseNet, by maintaining its backbone fixed and only revisiting the head. Unless otherwise specified (see ablations in Sec. 4), our texture model remains fixed, as well.

To the original BaseNet losses we simply amend our own. Otherwise, in terms of configuration and datasets, we follow the original BaseNet training strategy. During training, BaseNet learns to adjust its head so as to incur less alignment error on the training data. Finally, the texture module consists of 1.1 million parameters, with each training pass taking approximately 70 milliseconds.

4. Results

We evaluate our approach in two parts. First, we demonstrate the positive effect of incorporating our appearance-based supervision into an existing monocular 3D hand reconstruction pipeline. Then, we take a closer look at certain aspects that are rather intrinsic to the texture model itself.

Throughout, we adopt a controlled setup, keeping the backbone frozen and using a lightweight texture module to focus the analysis on the added value of texture-guided signals. The results show consistent improvements across datasets and metrics, suggesting that, even under conservative conditions, appearance supervision offers a meaningful complement to geometry-driven training, establishing the foundation on which more elaborate improvements can be founded (see Sec. 5).

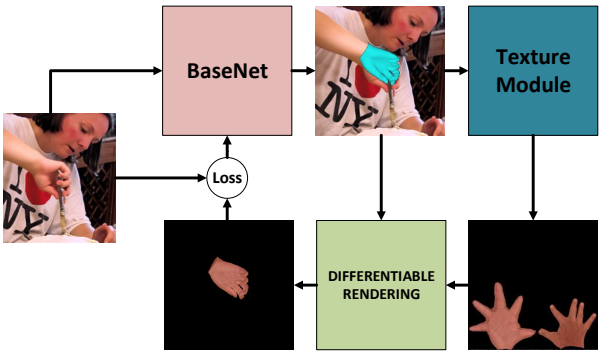

Figure 4. During training, the current 3D estimate from BaseNet is used to extract a noisy partial hand texture from the observation. This is refined through our texture model and is used to render a textured 3D hypothesis. The difference between the observation and the rendering comprises our additional loss term.

Experiments were conducted on a dual-socket AMD EPYC 7513 server with 64 physical cores, 1 TB RAM, and 4× NVIDIA A100-SXM4-80GB GPUs using CUDA 12.4. For this work we only employed one of the GPUs.

4.1. Contrastive evaluation

We instantiate BaseNet with HaMeR [47], and throughout this work, the two terms are used interchangeably. For evaluation, we follow the experimental setup of HaMeR, utilizing the HInt dataset, which consists of 2D-annotated images drawn from three egocentric datasets: NEWDAYS [9], VISOR [14], and Ego4D [20]. Performance is assessed using the PCK metric [59], consistent with HaMeR's protocol.

To ensure a fair comparison and to attribute observed effects solely to our texture module, we deliberately refrain from retraining the HaMeR backbone, which comprises over 600M parameters. Given such high capacity, retraining could overshadow the impact of our comparatively lightweight module, making it difficult to isolate the contribution of texture-based supervision.

4.1.1. Quantitative analysis

To quantify the effect of our texture model, as used through our dense alignment loss, we evaluated the HaMeR validation protocol [47] on the original implementation of HaMeR and our training-time enhancement. The findings in Tab. 1 indicate that our approach consistently outperforms HaMeR in the "All" and "Occluded" keypoint categories across all datasets and thresholds, with the most pronounced improve-

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

Category	Method	NEWDAYS			VISOR			Ego4D		
		@0.05	@0.10	@0.15	@0.05	@0.10	@0.15	@0.05	@0.10	@0.15
All	HaMeR [47]	49.4	79.3	89.8	44.4	77.5	89.7	40.3	72.4	85.2
	Ours	**49.5**	**79.5**	**89.9**	**46.8**	**79.1**	**90.6**	**42.6**	**73.5**	**86.1**
Visible	HaMeR [47]	**62.2**	**89.0**	**95.1**	58.5	88.4	95.0	53.9	84.2	91.8
	Ours	61.9	88.8	95.0	**61.2**	**89.0**	**95.3**	**57.2**	**84.8**	**92.3**
Occluded	HaMeR [47]	28.4	62.4	80.1	26.9	61.8	81.2	24.3	58.7	77.3
	Ours	**29.3**	**63.0**	**80.4**	**29.4**	**64.2**	**82.8**	**26.2**	**60.1**	**78.8**

Table 1. Comparison between HaMeR and our method on the Hint dataset at different thresholds. The proposed approach achieves better performance on this dataset. While HaMeR does outperform the proposed approach on the "visible keypoints" of the NEWDAYS subset, the difference is marginal.

Method	PA-MPJPE ↓	PA-MPVPE ↓	F@5 ↑	F@15 ↑
I2L-MeshNet [42]	7.4	7.6	0.681	0.973
Pose2Mesh [11]	7.7	7.8	0.674	0.969
I2UV-HandNet [4]	6.7	6.9	0.707	0.977
METRO [34]	6.5	6.3	0.731	0.984
Tang *et al.* [56]	6.7	6.7	0.724	0.981
Mesh Graphormer [35]	5.9	6.0	0.764	0.986
MobRecon [5]	**5.7**	5.8	0.784	0.986
AMVUR [27]	6.2	6.1	0.767	0.987
HaMeR [47]	6.0	**5.7**	**0.785**	**0.990**
Ours	6.1	**5.7**	**0.785**	**0.990**

Table 2. The proposed method achieves performance comparable to that of the initial method on the FreiHAND [63] dataset.

ments observed for occluded keypoints. For instance, on VISOR at @0.10, our method achieves 64.2% PCK for occluded keypoints, compared to 61.8% for HaMeR. This trend is consistent across all occluded cases, demonstrating the effectiveness of our texture-guided supervision in scenarios with limited visibility. In the "Visible" category, both methods perform comparably: HaMeR obtains slightly higher scores on the NEWDAYS subset (maximum difference of 0.3), while our method attains the best or equal results on VISOR and Ego4D. Overall, these results validate the benefit of incorporating appearance-based alignment, particularly in the presence of occlusions, while maintaining competitive performance in visible regions.

Additionally, we include comparisons in standardized 3D benchmarks, such as the FreiHAND [63] benchmark (see Tab. 2) and the HO3Dv2 [23, 25] benchmark (see Tab. 3). These results show a comparative performance that is on par with HaMeR, showing no regression on studio-captured, well-annotated, and relatively clean datasets, while at the same time improvement is achieved on egocentric, in-the-wild, or heavily occluded settings.

Similarly, our method also performs on par with another recent state-of-the-art baseline, HAMBA [17] (see Tab. 4), which builds on a modified Mamba [15, 21] architecture

Method	AUC_J ↑	PA-MPJPE ↓	AUC_V ↑	PA-MPVPE ↓	F@5 ↑	F@15 ↑
Liu *et al.* [36]	0.803	9.9	0.810	9.5	0.528	0.956
HandOccNet [46]	0.819	9.1	0.819	8.8	0.564	0.963
I2UV-HandNet [4]	0.804	9.9	0.799	10.1	0.500	0.943
Hampali *et al.* [23]	0.788	10.7	0.790	10.6	0.506	0.942
Hasson *et al.* [26]	0.780	11.0	0.777	11.2	0.464	0.939
ArtiBoost [58]	0.773	11.4	0.782	10.9	0.488	0.944
Pose2Mesh [11]	0.754	12.5	0.749	12.7	0.441	0.909
I2L-MeshNet [42]	0.775	11.2	0.722	13.9	0.409	0.932
METRO [34]	0.792	10.4	0.779	11.1	0.484	0.946
MobRecon[5]	-	9.2	-	9.4	0.538	0.957
Keypoint Trans [24]	0.786	10.8	-	-	-	-
AMVUR [27]	0.835	8.3	0.836	8.2	0.608	0.965
HaMeR	**0.846**	**7.7**	**0.841**	**7.9**	**0.635**	**0.980**
Ours	0.844	7.8	0.840	8.0	0.630	**0.980**

Table 3. **Comparison with the state-of-the-art on the HO3D dataset [23].** We use the HO3Dv2 protocol and report metrics that evaluate accuracy of the estimated 3D joints and 3D mesh. PA-MPVPE and PA-MPJPE numbers are in mm.

with additional components and training strategies. Interestingly, we attain similar performance through a straightforward training-time addition of supervision on an otherwise unadorned transformer-based architecture.

	Method	**New Days**			**VISOR**			**Ego4D**		
		@0.05	@0.10	@0.15	@0.05	@0.10	@0.15	@0.05	@0.10	@0.15
All	HAMBA [17]	48.7	79.2	**90.0**	**47.2**	**80.2**	**91.2**	41.7	72.9	85.5
	Ours	**49.5**	**79.5**	89.9	46.8	79.1	90.6	**42.6**	**73.5**	**86.1**
Vis.	HAMBA [17]	61.2	88.4	94.9	**61.4**	**89.6**	**95.6**	56.0	84.3	91.9
	Ours	**61.9**	**88.8**	**95.0**	61.2	89.0	95.3	**57.2**	**84.8**	**92.3**
Occl.	HAMBA [17]	28.2	62.8	**81.1**	**29.9**	**66.6**	**84.3**	25.2	59.2	77.6
	Ours	**29.3**	**63.0**	80.4	29.4	64.2	82.8	**26.2**	**60.1**	**78.8**

Table 4. Performance comparison between our method and HAMBA [17] on the HInt dataset. Best results are shown in **bold**.

4.1.2. Qualitative analysis

From the results of the quantitative analysis (see Sec. 4.1.1) we draw samples that help understand what the quantitative improvement amounts to in terms of visual improvement and "fit" improvement. As shown in Fig. 5, dense alignment during training, enabled through our texture model, leads to

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

	Variant	New Days			VISOR			Ego4D		
		@0.05	@0.10	@0.15	@0.05	@0.10	@0.15	@0.05	@0.10	@0.15
All	H&M*	45.9	78.1	89.6	39.9	76.8	89.7	37.1	71.4	85.0
	H&M	47.4	79.2	**90.0**	43.4	77.8	90.1	40.7	72.9	85.6
	H	**49.5**	**79.5**	89.9	**46.8**	**79.1**	**90.6**	**42.6**	**73.5**	**86.1**
Visible	H&M*	58.1	87.7	94.9	51.4	87.6	95.2	48.3	82.6	91.7
	H&M	58.7	88.5	**95.1**	56.5	88.4	**95.3**	54.0	83.8	92.1
	H	**61.9**	**88.8**	95.0	**61.2**	**89.0**	**95.3**	**57.2**	**84.8**	**92.3**
Occluded	H&M*	26.2	61.4	80.2	26.1	61.6	81.3	23.4	58.0	77.4
	H&M	29.0	62.7	**81.0**	27.6	62.8	81.6	25.7	59.8	78.1
	H	**29.3**	**63.0**	80.4	**29.4**	**64.2**	**82.8**	**26.2**	**60.1**	**78.8**

Table 5. Performance comparison on the HInt dataset with the head retrained using the texture module. We denote training the Head by H, training the head with the warmed up texture module by H&M. Finally, H&M* denotes training of the head and the texture module from scratch.

# Visible Pixels	L1 ↓	SSIM ↑
200	7.6	0.85
500	6.2	0.89
1000	5.9	0.89
2000	5.4	0.91
ALL	**3.3**	**0.95**

Table 6. Quantitative metrics on the effect the number of visible UV pixels has on texture reconstruction quality.

"fitter" overlays of hypotheses over the corresponding observations. For example, a visually apparent improvement regards the better localization of the fingers.

4.2. Texture model insights

We analyze how different configurations influence the performance of our method or the texture model itself. When it comes to texture quality or fidelity, we employ the SSIM and L1 metrics and measure against known ground truth generated from HTML [49].

Initialize/Freeze? We compare the following variants, w.r.t. fine-tuning HaMeR:

- H: only fine-tune the head of HaMeR (the texture module is warmed up),
- $H\&M$: fine-tune both the head of HaMeR and the warmed up texture module, and,
- $H\&M*$: like $H\&M$, but with a randomly initialized texture model.

The results in Tab. 5 indicate that H is the most effective approach in this context, and the one considered throughout. It is anticipated that with changes and/or in a different setting, the results might differ (see Sec. 5).

Observation density VS "recall": Because each monocular observation offers only partial UV coverage, it is important to understand how the number of observed UV-pixel pairs affects the quality of texture reconstruction. To investigate this, we vary the number of visible UV samples provided to the texture module (see Tab. 6). We find that our method maintains high reconstruction fidelity with as few as 1,000 visible UV pixels. Both L1 loss and SSIM remain stable above this threshold, while reconstruction quality degrades more noticeably below it. This analysis shows that the proposed approach is robust to the sparse and incomplete supervision typical of in-the-wild monocular data, and remains practical in real-world conditions where full surface visibility is rare.

Hyperparameter tuning: We search for the optimal architectural configuration for the texture module by varying the number of transformer layers $\mathbf{K}$ and the embedding dimension D (see Eq. (1)). Table 7 summarizes the impact of these choices, evaluated using L1 loss and SSIM. We find that SSIM remains consistently high (up to 0.95) across several configurations, indicating robust perceptual quality. However, L1 loss shows more variation, with the lowest value (3.35) achieved using 8 layers and an embedding dimension of 6. This suggests that increasing model depth and selecting moderate embedding sizes can yield incremental improvements in pixel-wise accuracy, while overall visual quality remains stable. This is the configuration used throughout.

Navigating Towards Better Textures: Through hands-on experimentation, we quickly identified design choices that consistently yield sharper and more realistic textures. Using pixel shuffle for upsampling, incorporating Fourier-based positional encodings, and adopting a moderately deep transformer backbone emerged as particularly effective. Additionally, combining L1 and Fourier-domain losses proved crucial for producing high-fidelity textures. These choices allow the texture module to operate reliably even with limited supervision, making it suitable for single-view reconstruction tasks in practical scenarios.

# of Layers	Emb. Dim	L1↓	SSIM ↑
6	6	6.25	0.90
8	8	5.45	0.90
4	6	5.12	0.91
4	8	4.36	0.94
6	8	3.77	0.95
8	6	**3.35**	**0.95**

Table 7. Performance metrics for different number of layers and embedding dimensions.

5. Conclusions and future work

We have shown that adding a lightweight, texture-aware supervision module only during training delivers systematic improvements to monocular 3-D hand reconstruction while leaving the runtime architecture unchanged—the only difference at test time is the improved weights. Injecting

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

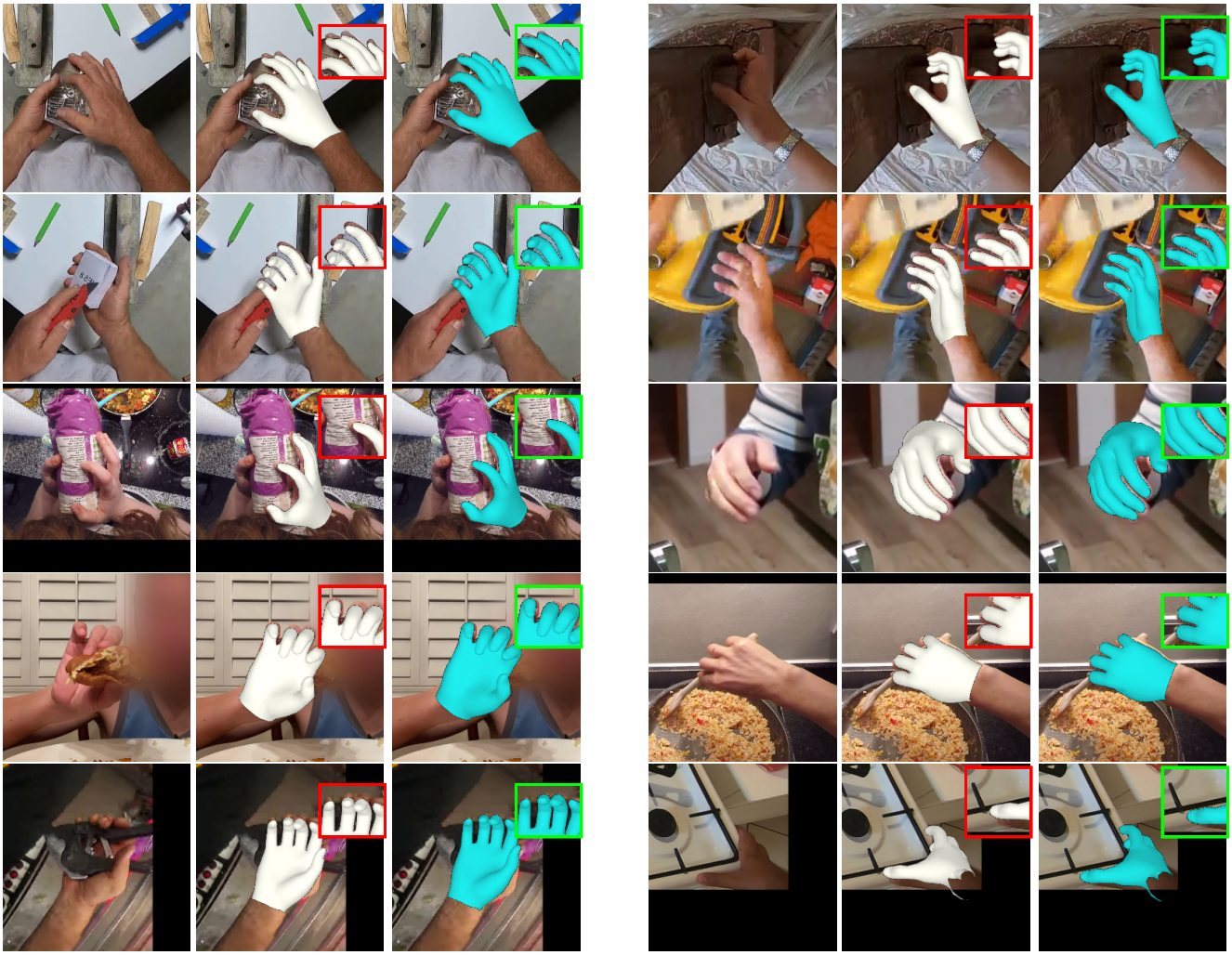

Figure 5. Indicative qualitative results on the HInt dataset. The first column shows the input image. The second column presents the estimated hand prediction from HaMeR. The third column displays the predicted result from our method. Evidently the proposed method, using texture priors, can better estimate the hand shape and pose.

fine-grained appearance cues into optimization increases PCK by up to 2.7% on heavily occluded HInt frames and yields consistent, albeit smaller, gains on FreiHAND and HO-3D. The effect is most pronounced in scenarios where geometry-only cues are ambiguous—severe occlusion and egocentric viewpoints—underscoring the complementary value of texture information.

The proposed module is practical: it requires no extra manual annotation, reconstructs dense UV maps from as few as 1.000 observed texels, adds $<$ 70ms per training iteration, and incurs zero test-time cost. A moderate-depth transformer with Fourier positional encodings and pixel-shuffle up-sampling strikes the best balance between accuracy and efficiency.

Our work provides a complete, modular solution for integrating learned texture priors into monocular 3D hand reconstruction pipelines. The results across multiple benchmarks confirm that the method achieves its core objectives: improving geometric and appearance accuracy in a weakly-supervised setting, without reliance on multi-view data or dense annotations. The framework is robust, scalable, and readily deployable in real-world scenarios.

Still, the tools and insights developed here naturally give rise to a number of promising extensions:

- Apply the texture prior to a broader range of hand reconstruction backbones to further validate its generality.
- Develope more advanced differentiable rendering strategies to improve training stability and convergence, as advocated in [30].
- Explore the texture module as a denoiser for pose estimation errors through targeted weakly-supervised training.
- Investigate closer integration between the texture prior and base network for possible gains in optimization dynamics.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

Acknowledgments

This work was supported by the European Union (EUHE Magician - Grant Agreement 101120731) and the internal FORTH-ICS project "Large-scale, Diverse 3D Modeling of Human Hands"- DiveHands. The authors also gratefully acknowledge the support for this research from the VMware University Research Fund (VMURF). GP acknowledges the support of NSF-2504906 and Gifts from Adobe and Google.

References

[1] Danilo Avola, Luigi Cinque, Alessio Fagioli, Gian Luca Foresti, Adriano Fragomeni, and Daniele Pannone. 3d hand pose and shape estimation from rgb images for keypoint-based hand gesture recognition. *Pattern Recognition*, 129: 108762, 2022. 3

[2] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3d hand shape and pose from images in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10843–10852, 2019. 3

[3] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, et al. Dexycb: A benchmark for capturing hand grasping of objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9044–9053, 2021. 3

[4] Ping Chen, Yujin Chen, Dong Yang, Fangyin Wu, Qin Li, Qingpei Xia, and Yong 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*, pages 12929–12938, 2021. 6

[5] Xingyu Chen, Yufeng Liu, Yajiao Dong, Xiong Zhang, Chongyang Ma, Yanmin Xiong, Yuan Zhang, and Xiaoyan Guo. Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20544–20554, 2022. 6

[6] Xingyu Chen, Zhuheng Song, Xiaoke Jiang, Yaoqing Hu, Junzhi Yu, and Lei Zhang. Handos: 3d hand reconstruction in one stage. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17304–17314, 2025. 2, 3

[7] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10451–10460, 2021. 3

[8] Zhaoxi Chen, Gyeongsik Moon, Kaiwen Guo, Chen Cao, Stanislav Pidhorskyi, Tomas Simon, Rohan Joshi, Yuan Dong, Yichen Xu, Bernardo Pires, et al. Urhand: Universal relightable hands. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 119–129, 2024. 3

[9] Tianyi Cheng, Dandan Shan, Ayda Hassen, Richard Higgins, and David Fouhey. Towards a richer 2d understanding of hands at scale. *Advances in Neural Information Processing Systems*, 36:30453–30465, 2023. 3, 5

[10] Wencan Cheng, Hao Tang, Luc Van Gool, and Jong Hwan Ko. Handdiff: 3d hand pose estimation with diffusion on image-point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2274–2284, 2024. 3

[11] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 769–787. Springer, 2020. 6

[12] Hongsuk Choi, Nikhil Chavan-Dafle, Jiacheng Yuan, Volkan Isler, and Hyunsoo Park. Handnerf: Learning to reconstruct hand-object interaction scene from a single rgb image. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13940–13946. IEEE, 2024. 3

[13] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20533–20543, 2022. 3

[14] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European conference on computer vision (ECCV)*, pages 720–736, 2018. 5

[15] Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning (ICML)*, 2024. 6

[16] Ahmad Darkhalil, Dandan Shan, Bin Zhu, Jian Ma, Amlan Kar, Richard Higgins, Sanja Fidler, David Fouhey, and Dima Damen. Epic-kitchens visor benchmark: Video segmentations and object relations. *Advances in Neural Information Processing Systems*, 35:13745–13758, 2022. 3

[17] Haoye Dong, Aviral Chharia, Wenbo Gou, Francisco Vicente Carrasco, and Fernando D De la Torre. Hamba: Single-view 3d hand reconstruction with graph-guided bi-scanning mamba. *Advances in Neural Information Processing Systems*, 37:2127–2160, 2024. 2, 6

[18] Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, and Cewu Lu. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE transactions on pattern analysis and machine intelligence*, 45(6):7157–7173, 2022. 3

[19] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong 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*, pages 10833–10842, 2019. 3

[20] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d:

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 3, 5

[21] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 6

[22] Zhiyang Guo, Wengang Zhou, Min Wang, Li Li, and Houqiang Li. Handnerf: Neural radiance fields for animatable interacting hands. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21078–21087, 2023. 3

[23] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. Honnotate: A method for 3d annotation of hand and object poses. In *CVPR*, 2020. 3, 6

[24] Shreyas Hampali, Sayan Deb Sarkar, Mahdi Rad, and Vincent 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*, pages 11090–11100, 2022. 6

[25] Shreyas Hampali, Sayan Deb Sarkar, Mahdi Rad, and Vincent Lepetit. Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation. In *CVPR*, 2022. 6

[26] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11807–11816, 2019. 6

[27] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 758–767, 2023. 6

[28] Sheng Jin, Lumin Xu, Jin Xu, Can Wang, Wentao Liu, Chen Qian, Wanli Ouyang, and Ping Luo. Whole-body human pose estimation in the wild. In *European Conference on Computer Vision*, pages 196–214. Springer, 2020. 3

[29] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE international conference on computer vision*, pages 3334–3342, 2015. 3

[30] Giorgos Karvounas, Nikolaos Kyriazis, Iason Oikonomidis, Aggeliki Tsoli, and Antonis A Argyros. Multi-view image-based hand geometry refinement using differentiable monte carlo ray tracing. In *British Machine Vision Conference (BMVC 2021)*, Virtual, UK, 2021. BMVA. 3, 8

[31] Giorgos Karvounas, Nikolaos Kyriazis, Iason Oikonomidis, and Antonis Argyros. Dynamic multiview refinement of 3d hand datasets using differentiable ray tracing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3156–3166, 2023. 3

[32] Minje Kim and Tae-Kyun Kim. Bitt: Bi-directional texture reconstruction of interacting two hands from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10726–10735, 2024. 3

[33] 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 Transactions on Graphics (TOG)*, 41(4):1–16, 2022. 3

[34] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1954–1963, 2021. 6

[35] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12939–12948, 2021. 2, 6

[36] Shaowei Liu, Hanwen Jiang, Jiarui Xu, Sifei Liu, and Xiaolong 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*, pages 14687–14697, 2021. 6

[37] Tianyu Luan, Yuanhao Zhai, Jingjing Meng, Zhong Li, Zhang Chen, Yi Xu, and Junsong Yuan. Scalable high-fidelity 3d hand shape reconstruction via graph-image frequency mapping and graph frequency decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 3

[38] Jameel Malik, Ibrahim Abdelaziz, Ahmed Elhayek, Soshi Shimada, Sk Aziz Ali, Vladislav Golyanik, Christian Theobalt, and Didier Stricker. Handvoxnet: Deep voxel-based network for 3d hand shape and pose estimation from a single depth map. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7113–7122, 2020. 3

[39] Jameel Malik, Soshi Shimada, Ahmed Elhayek, Sk Aziz Ali, Christian Theobalt, Vladislav Golyanik, and Didier Stricker. Handvoxnet++: 3d hand shape and pose estimation using voxel-based neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):8962–8974, 2021. 3

[40] 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. *Communications of the ACM*, 65(1):99–106, 2021. 3

[41] Gyeongsik Moon. Bringing inputs to shared domains for 3d interacting hands recovery in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17028–17037, 2023. 3

[42] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 752–768. Springer, 2020. 6

[43] 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

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.

image. In *European Conference on Computer Vision*, pages 548–564. Springer, 2020. 3

[44] Supreeth Narasimhaswamy, Uttaran Bhattacharya, Xiang Chen, Ishita Dasgupta, Saayan Mitra, and Minh Hoai. Handiffuser: Text-to-image generation with realistic hand appearances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2468–2479, 2024. 3

[45] Takehiko Ohkawa, Kun He, Fadime Sener, Tomas Hodan, Luan Tran, and Cem Keskin. Assemblyhands: Towards egocentric activity understanding via 3d hand pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12999–13008, 2023. 3

[46] JoonKyu Park, Yeonguk Oh, Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Handoccnet: Occlusion-robust 3d hand mesh estimation network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1496–1505, 2022. 6

[47] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3d with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9826–9836, 2024. 1, 2, 3, 5, 6

[48] Rolandos Alexandros Potamias, Stylianos Ploumpis, Stylianos Moschoglou, Vasileios Triantafyllou, and Stefanos Zafeiriou. Handy: Towards a high fidelity 3d hand shape and appearance model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4670–4680, 2023. 3

[49] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 54–71. Springer, 2020. 3, 4, 7

[50] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv preprint arXiv:2007.08501*, 2020. 2, 5

[51] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Trans. Graph.*, 36(6):245:1–245:17, 2017. 5

[52] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016. 4

[53] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017. 3

[54] Adrian Spurr, Aneesh Dahiya, Xi Wang, Xucong Zhang, and Otmar Hilliges. Self-supervised 3d hand pose estimation from monocular rgb via contrastive learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11230–11239, 2021. 3

[55] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020. 4

[56] Xiao Tang, Tianyu Wang, and Chi-Wing Fu. Towards accurate alignment in real-time 3d hand-mesh reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11698–11707, 2021. 6

[57] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10974, 2019. 3

[58] Lixin Yang, Kailin Li, Xinyu Zhan, Jun Lv, Wenqiang Xu, Jiefeng Li, and Cewu 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*, pages 2750–2760, 2022. 6

[59] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2878–2890, 2012. 5

[60] Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4210–4220, 2023. 3

[61] Allan Zhou, Moo Jin Kim, Lirui Wang, Pete Florence, and Chelsea Finn. Nerf in the palm of your hand: Corrective augmentation for robotics via novel-view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17907–17917, 2023. 3

[62] Jiayin Zhu, Zhuoran Zhao, Linlin Yang, and Angela Yao. Hifihr: Enhancing 3d hand reconstruction from a single image via high-fidelity texture. In *DAGM German Conference on Pattern Recognition*, pages 115–130. Springer, 2023. 3

[63] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas 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*, pages 813–822, 2019. 3, 6

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:38:55 UTC from IEEE Xplore. Restrictions apply.