Intrinsic Hand Avatar: Illumination-aware Hand Appearance and Shape Reconstruction from Monocular RGB Video

Pratik Kalshetti
Indian Institute of Technology Bombay
pmkalshetti@iitb.ac.in

Parag Chaudhuri
Indian Institute of Technology Bombay
paragc@cse.iitb.ac.in

Abstract

Reconstructing a user-specific hand avatar is essential for a personalized experience in augmented and virtual reality systems. Current state-of-the-art avatar reconstruction methods use implicit representations to capture detailed geometry and appearance combined with neural rendering. However, these methods rely on a complicated multi-view setup, do not explicitly handle environment lighting leading to baked-in illumination and self-shadows, and require long hours for training. We present a method to reconstruct a hand avatar from a monocular RGB video of a user's hand in arbitrary hand poses captured under real-world environment lighting. Specifically, our method jointly optimizes shape, appearance, and lighting parameters using a realistic shading model in a differentiable rendering framework incorporating Monte Carlo path tracing. Despite relying on physically-based rendering, our method can complete the reconstruction within minutes. In contrast to existing work, our method disentangles intrinsic properties of the underlying appearance and environment lighting, leading to realistic self-shadows. We compare our method with state-of-the-art hand avatar reconstruction methods and observe that it outperforms them on all commonly used metrics. We also evaluate our method on our captured dataset to emphasize its generalization capability. Finally, we demonstrate applications of our intrinsic hand avatar on novel pose synthesis and relighting. We plan to release our code to aid further research.

1. Introduction

A user-specific digital hand avatar is essential to realize a realistic, immersive experience in virtual reality (VR) and augmented reality (AR) applications. In our context, an avatar is represented by geometry (*e.g.* triangle mesh) and appearance (*e.g.* spatially-varying material). Accurate geometry of the user's hand shape is essential for precise interaction with virtual objects, whereas the realistic appearance of the user's hand texture plays a vital role in providing a sense of embodiment to the user. We focus on the problem of automatically generating such a personalized hand avatar from a monocular video.

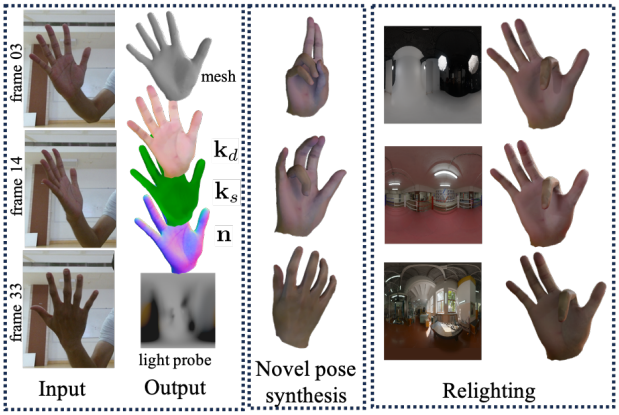

Figure 1. Given a monocular RGB video of a user's hand, we reconstruct the user's hand geometry and appearance, along with environment lighting. Our reconstructed intrinsic hand avatar can be posed and rendered under novel lighting.

Recently, there has been a lot of research in creating digital avatars of humans [7, 13, 15, 20, 21, 45, 48, 59, 67] and faces [16–18, 68]. However, hand avatar reconstruction has unique challenges, like large self-occlusion, contact, and substantial pose variation compared to full-body and face. Further, hands play a much more critical role in an interactive experience, so developing methods tailored for reconstructing hand avatars is essential.

There has been vast research in hand pose estimation [2, 6, 10, 23, 26, 38, 49–51, 55, 56], but relatively less advances towards modeling hand appearance [8, 11, 24, 28, 42]. Traditional approaches to hand appearance modeling learn a PCA basis of textures [28, 42] from a large scan of various hands. However, these approaches cannot generalize to unseen hands. Recent advances in neural rendering have enabled learning digital hand avatars from monocular [15,

18, 20, 21, 24, 59, 67, 68], or multi-view videos [8, 11, 45]. However, these approaches do not disentangle lighting from the intrinsic appearance of the user's hand and thus fail to account for self-occlusion.

We propose a method to reconstruct a hand avatar from a monocular RGB video of a user's hand acquired under real-world environment lighting, within minutes. Our avatar representation comprises a parametric triangle mesh and spatially-varying physically-based materials [30]. Finally, we model lighting using a high dynamic range environment light probe to disentangle the intrinsic appearance of the user's hand and illumination. Our method jointly optimizes a parametric mesh, material, and lighting using a realistic shading model in a differentiable rendering framework. We use a differentiable deferred renderer [19] which computes direct illumination using Monte Carlo path tracing. This enables us to generate realistic shading (c.f. HandAvatar [8]) under more general real-world environments (c.f. HARP [24]).

Our method requires only a short monocular video (50 frames) such that each region on the hand is visible in at least some frames.

We compare our method with state-of-the-art hand avatar reconstruction methods on a publicly available dataset and observe that it outperforms them on all commonly used metrics.

We summarize our primary contributions below

- We propose a method to reconstruct an intrinsic hand avatar from monocular RGB video of the hand in arbitrary poses captured under real-world environment lighting. Our method jointly optimizes hand shape and appearance in a differentiable rendering framework in minutes.
- We extend the widely used MANO hand model by linearly subdividing it and incorporating per-vertex offsets to capture fine-level details.

2. Related Work

2.1. Explicit Models

Over the years, there have been various models to represent the human hand. OpenPose [46], MediaPipe [53] estimate landmark points on the hand from RGB images to represent the pose of the skeleton. Gorce *et al*. [12] introduced the idea of analysis-by-synthesis to estimate the full hand surface represented by a triangle mesh. With the advent of the parametric hand mesh model MANO [43], a large number of methods [6, 10, 69] regress the MANO shape and pose parameters. To allow more degrees of freedom, Kulon *et al*. [26] regressed vertices using mesh convolutions. There is also a large literature on depth-based hand reconstruction [23, 50, 51]. However, all of these above methods focus on the user's hand shape and pose, and do not capture its appearance. Our geometric model leverages a parametric hand mesh model to regularize the optimization and also allows modification to the mesh to capture fine-level details specific to the user.

Inspired by 3D morphable face model (3DMM) [4], HTML [42] proposes a linear appearance model for inferring UV texture, and NIMBLE [28] infers an albedo, specular and normal map. However, these methods are limited by their training set and thus cannot capture unseen hands whose appearance lies outside their dataset. Our method directly infers the intrinsic color of the user's hand from image observations and thus generalizes better than existing PCA-based textures.

2.2. Implicit Models

The recent success of neural implicit representations has led to several methods to reconstruct 3D geometry and appearance from image collections. NeRF [32] and follow-ups [14, 29, 36, 39, 41, 48, 52, 61] use a volumetric representation to calculate the radiance accumulated along a ray by marching through a 5D light field. However, these methods fail to capture accurate geometry because of the ambiguity of volume rendering. Implicit surface reconstruction methods(VolSDF [60], UNISURF [37], NeuS [57]) leverage an occupancy network or signed distance function to model a more accurate geometry. Apart from the computationally expensive ray marching strategy for rendering, these methods cannot handle dynamic scenes. To handle deformable shapes, SNARF [9] proposed forward deformation fields from canonical space to deformed space and uses iterative root finding to find correspondences. SelfRecon [20], InstantAvatar [21] uses this idea to reconstruct human avatars, while IMAvatar [68] constructs face avatars from monocular video. LISA [11], the first method to learn an implicit shape and appearance of hands from multiview images, uses kinematic transformations of the bones to handle deformations. While these implicit surfaces can be converted to meshes for traditional graphics applications, this is suboptimal.

More recently, some methods [15, 18, 67] combine explicitly mesh-based representation with deformation networks and appearance networks for compatibility with standard graphics pipelines. However, none of the above methods handle lighting, which leads to incorrect shadows and baked-in illumination in the albedo. HandAvatar [8] attempts to mitigate this problem by introducing a self-occlusion-aware shading field which comprises an albedo field and an illumination field. However, their network requires a large training dataset per user per environment, and thus inference suffers on out-of-distribution data. HARP [24] avoids any neural networks by using an analysis-by-synthesis approach to model hand shape and

appearance. Specifically, it uses a mesh-based parametric model, a vertex displacement map, a normal map, and albedo together with a shadow-aware differentiable rendering to obtain a personalized hand avatar from monocular video. However, it can only handle a single light source and has limited texture resolution. Our method also uses an analysis-by-synthesis approach to generate a personalized hand avatar from monocular video, but can handle environment lighting and produces a more detailed texture compared to HARP, and generalizes to new shapes and appearances, unlike HandAvatar.

2.3. Material and Lighting Estimation

Reconstructing an object from images is the problem of inverse rendering, which decomposes the underlying intrinsic properties such as geometry, material and lighting. The forward rendering equation [22] (for non-emissive surfaces) computes the outgoing radiance L_o at surface point $\mathbf{x}$ along direction ω_o by integrating the reflected light over hemisphere Ω:

$$L_o(\mathbf{x}, \omega_o) = \int_{\Omega} L_{in}(\mathbf{x}, \omega_i) f(\mathbf{x}, \omega_i, \omega_o)(\omega_i \cdot \mathbf{n}) \mathrm{d}\omega_i \quad (1)$$

where $L_{in}(\mathbf{x}, \omega_i)$ is the incoming radiance along direction ω_i and f denotes the bidirectional reflectance distribution function (BRDF) which describes how much light arriving from direction ω_i at $\mathbf{x}$ is reflected towards ω_o.

Recent methods have successfully estimated lighting and BRDF from image collections. To obtain this intrinsic decomposition, NeRD [5] and PhySG [63] use spherical Gaussians, NeRFactor [65] uses a low-resolution environment map to represent illumination, while Zhang *et al*. [66] uses MLPs to model indirect illumination. However, these methods require long training and inference times because of the multiple MLPs required for representation. In contrast, NVDIFFREC [35] uses deferred shading and the split-sum approximation for direct illumination but does not model self-shadows which are crucial in hands. Further, NVDIFFRECMC [19] evaluates direct lighting integral using Monte Carlo integration and ray tracing to model shadow rays. However, it requires multiview images of a static object. Our method follows the renderer from NVDIFFRECMC to account for self-shadows, however, unlike NVDIFFRECMC, our method is capable of reconstructing from a monocular video of a dynamic hand.

3. Hand Avatar Reconstruction

Given a monocular RGB video of a user's hand performing arbitrary poses under real-world environment lighting, our goal is to reconstruct a personalized hand avatar of the user. Specifically, our hand avatar representation consists of a triangle mesh to capture the user's hand shape (geometry) and spatially-varying materials (stored as 2D textures) to capture the user's hand appearance. We assume that every region on the user's hand is visible in at least one of the frames of the video. We also assume that the hand is illuminated under an unknown environment lighting which we model using a high dynamic range (HDR) environment probe.

Our method jointly optimizes a parametric mesh, material, and lighting using a realistic shading model in a differentiable rendering framework [19] incorporating Monte Carlo path tracing. Our method is summarized in Fig. 2, and each step is described in detail below; Sec. 3.1 describes the parametric mesh used to model the hand geometry, Sec. 3.2 details the material and lighting model while Sec. 3.3 explains our deferred rendering pipeline. Finally, we explain our optimization procedure in Sec. 3.4.

3.1. Geometry

Our parametric mesh model is based on the MANO [43] hand model for geometry and HTML [42] for UV mapping. The MANO model is a parameterized human hand model described by a function of pose $\theta \in \mathbb{R}^{45}$ (capturing local rotation at each of the 15 joints) and shape $\beta \in \mathbb{R}^{10}$ (coefficients for the PCA shape blend shapes) returning $N = 778$ vertices and $F = 1538$ faces. Additionally, to account for the global rigid transformation, we include global orientation and translation in the pose resulting in $\theta \in \mathbb{R}^{45+6}$. However, the original MANO model is too coarse to capture detailed geometry and appearance. Inspired by Alldieck *et al*. [1], we adapt MANO as follows.

Initially, we allow offsets $\mathbf{D} \in \mathbb{R}^{3N}$ from the template $\mathbf{T} \in \mathbb{R}^{3N}$:

$$M(\beta, \theta, \mathbf{D}) = W(T(\beta, \theta, \mathbf{D}), J(\beta), \theta, \mathbf{W}) \quad (2)$$

$$T(\beta, \theta, \mathbf{D}) = \mathbf{T} + B_s(\beta) + B_p(\theta) + \mathbf{D} \quad (3)$$

where W is a linear blend-skinning function using the skinning weights $\mathbf{W}$ to pose the rest-pose skeleton joints $J(\beta)$ and vertices $T(\beta, \theta, \mathbf{D})$ (obtained after pose- $B_p(\theta)$ and shape-dependent $B_s(\beta)$ deformations).

Further, to capture fine-level details, we uniformly subdivide the template mesh by adding new vertices on edge midpoints and correspondingly update the parametric mesh model. Our subdivided mesh has $N = 14652$ vertices and $F = 38450$ faces after performing two iterations of subdivision. Our extended MANO is differentiable with respect to the geometry parameters shape β, pose θ, and offsets $\mathbf{D}$, thus enabling gradient backpropagation.

3.2. Appearance

Material We model the hand material's reflectance using Disney's principled BRDF [30] which is a parametric model

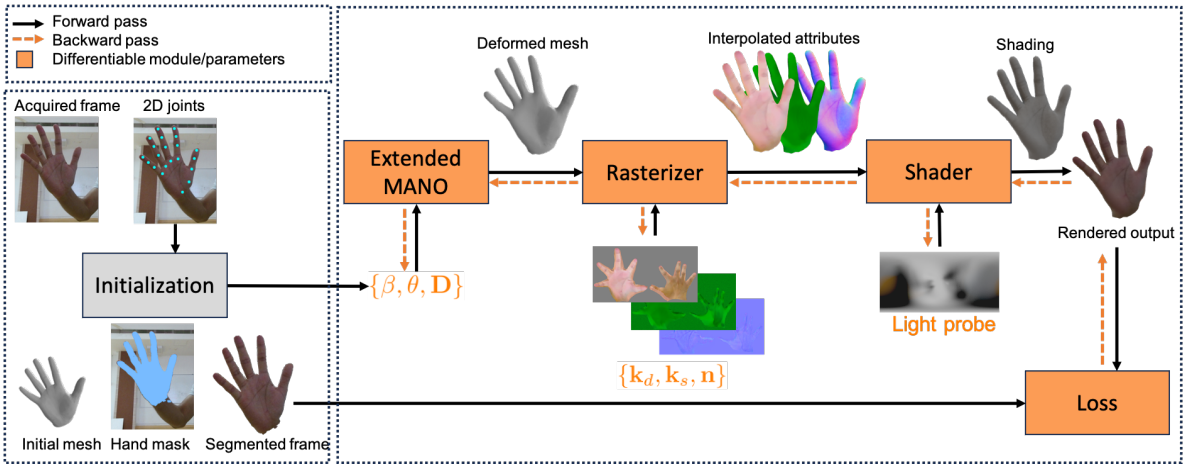

Figure 2. The input to our method is a monocular RGB video of a user's hand. For each frame f, we initialize the pose of the MANO [43] parametric hand model by fitting it to observed 2D joint positions [53] and use the posed mesh to segment the hand region. We extend the original MANO model (β, θ) by subdividing and introducing offsets $\mathbf{D}$ to capture detailed geometry. We use a differentiable deferred rendering pipeline: first, we rasterize the PBR material textures along with surface intersection point and normal to obtain interpolated vertex attributes and, then, use Monte Carlo integration to estimate outgoing radiance in the presence of an environment light. Our method jointly optimizes the geometry $(\{\theta_f\}, \beta, \mathbf{D})$, material $(\mathbf{k}_d, \mathbf{k}_s, \mathbf{n})$, and lighting (modeled as HDR light probe) parameters using an analysis-by-synthesis approach to minimize the difference between the rendered and the input image.

offering a good balance between simplicity and flexibility. Specifically, our BRDF is parameterized by a three-channel diffuse albedo $\mathbf{k}_d$, a roughness channel r, a metalness channel m and a tangent space normal map $\mathbf{n}$. Since hand skin does not resemble a metal, we clamp the metalness value m to be low. The specular highlight color is calculated according to $\mathbf{k}_s = (1-m)\cdot 0.04 + m\cdot\mathbf{k}_d$ [31]. Further, following a standard convention [35], we store these values in a texture $\mathbf{k}_{orm} = (o, r, m)$, where o is left unused. Our physically-based material model enables us to model the realistic hand appearance of various users.

Lighting We model environment lighting using an HDR light probe, which supplies incident radiance from all directions on the sphere. Specifically, we use a 2D floating point texture map parameterized by spherical coordinates to represent our light probe. Our lighting representation enables us to model real-world environment lighting which is typically found in the usage scenarios for AR/VR devices.

3.3. Rendering

We now describe how the geometry, material, and lighting are utilized in a differentiable deferred rendering pipeline. We first rasterize the scene into geometry buffers (G-buffer) that include the surface intersection point and normal, and the interpolated material parameters for each pixel. Given the G-buffer, we now perform the shading pass to determine the outgoing radiance. Specifically, we use Monte Carlo (MC) integration to estimate the outgoing radiance. This requires us to calculate visibility, which is evaluated by tracing a shadow ray in the incoming light direction. This formulation allows us to explicitly handle self-shadows caused due to self-occlusion among fingers.

To achieve fast optimization times, we use low sample counts per pixel and combat the inherent variance of MC integration using differentiable image denoising (SVGF [44]) and multiple importance sampling [54]. Please refer to NVDIFFRECMC [19] for more details on sampling and denoising. We follow previous work in denoising for production rendering [3] and separate lighting into diffuse $\mathbf{c}_d$ and specular $\mathbf{c}_s$ terms. We denoise each term separately, creating denoised buffers $D(\mathbf{c}_d)$ and $D(\mathbf{c}_s)$. The final rendered image $\mathbf{c}$ is obtained (see Fig. 3) as

$$\mathbf{c} = \mathbf{k}_d \cdot D(\mathbf{c}_d) + D(\mathbf{c}_s) \tag{4}$$

Note that we use a small sample count per pixel and denoising only during optimization. During evaluation and for all the images presented in the paper, we use a higher sample count per pixel without denoising.

The rasterizer [27] and shading [19] are differentiable and thus allow gradients to be backpropagated to lighting, material as well as geometry parameters.

3.4. Optimization

Given a video with F frames, we jointly optimize the mesh parameters $\left(\beta, \mathbf{D}, \{\theta_f\}_{f=1}^{F}\right)$, material parameters $(\mathbf{k}_d, \mathbf{k}_s, \mathbf{n})$ and the light probe texture using self-supervision (analysis-by-synthesis) on the monocular RGB video. Our optimization minimizes the sum of data and var-

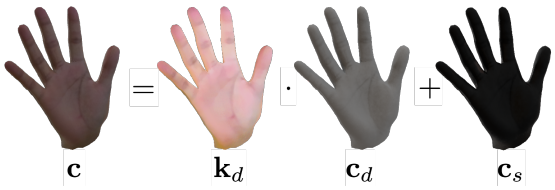

Figure 3. Our final rendered image is obtained from diffuse reflectance or albedo $\mathbf{k}_d$, diffuse lighting $\mathbf{c}_d$, and specular lighting $\mathbf{c}_s$. (no denoising)

ious regularizer terms.

$$E_{total} = E_{data} + \sum_i \omega_i E_i \tag{5}$$

We describe each of these terms below.

Data term Our image space loss computes the L_1 norm of the error between the tone-mapped [35] (since our shading images are HDR) output rendered image $\tilde{\mathbf{c}}$ and the input reference image $\tilde{\mathbf{c}}_{ref}$, applied on tone-mapped versions. We also use a mask loss which computes a L_2 norm of the error between the silhouettes of the rendered $\mathbf{s}$ and input $\mathbf{s}_{ref}$ images. We obtain the input image silhouette $\mathbf{s}_{ref}$ using the mesh obtained during the initialization stage.

$$E_{data} = \|\tilde{\mathbf{c}} - \tilde{\mathbf{c}}_{ref}\|_1 + \|\mathbf{s} - \mathbf{s}_{ref}\|_2 \tag{6}$$

Geometry regularizer We regularize the introduced offsets $\mathbf{D}$ by adding a Laplacian mesh regularizer [47]:

$$E_{lap} = \frac{1}{N}\sum_{i=1}^{N} \|\delta_i - \delta_i'\|^2 \tag{7}$$

where δ_i and δ_i' are the uniformly-weighted differential of the optimized and the initial mesh for a vertex v_i given by $\delta_i = v_i - \frac{1}{|N_i|}\sum_{j \in N_i} v_j$, with N_i being the one-ring neighborhood of vertex v_i.

Material regularizer We regularize material parameters using a smoothness loss similar to NeRFactor [65]. We define the smoothness prior for $\mathbf{k}_d$ as

$$E_{\mathbf{k}_d} = \sum_{x_{surf}} \|\mathbf{k}_d(x_{surf}) - \mathbf{k}_d(x_{surf} + \epsilon)\|^2 \tag{8}$$

where ϵ is a small random displacement vector, $\mathbf{k}_d(x_{surf})$ denotes the value of $\mathbf{k}_d$ at world space position x_{surf} at the primary hit point on the object. We apply similar smoothness priors for specular $E_{\mathbf{k}_s}$ and normal $E_{\mathbf{n}}$ maps, respectively.

Additionally, we use the learned perceptual image patch similarity (LPIPS) [64] loss to encourage perceptual similarity.

$$E_{LPIPS} = \mathcal{L}_{LPIPS}(\mathbf{c}, \mathbf{c}_{ref}) \tag{9}$$

Lighting regularizer We note that the prior smoothness cannot disentangle material parameters and lighting. We use a monochrome image loss between demodulated lighting terms (i.e., before being multiplied by diffuse reflectance $\mathbf{k}_d$) and reference image $\mathbf{c}_{ref}$ [19].

$$E_{light} = \|Y(\mathbf{c}_d + \mathbf{c}_s) - V(\mathbf{c}_{ref}\|^2 \tag{10}$$

where $Y(\mathbf{x}) = (\mathbf{x}_r + \mathbf{x}_g + \mathbf{x}_b)/3$ is a luminance operator, and $V(\mathbf{x}) = \max(\mathbf{x}_r, \mathbf{x}_g, \mathbf{x}_b)$ is the HSV value component.

Initialization Our method requires that the initial mesh roughly aligns with the input image. By adopting a tracking algorithm, we initialize the MANO parameters (β, θ). Specifically, we use an analysis-by-synthesis approach to minimize the L_1 distance of the detected 2D hand joint positions $\{j_i'\}_i^{21}$ (obtained using MediaPipe [53] and the projected 2D joint positions $\{j_i\}_i^{21}$ of the posed MANO mesh.

$$E_{2D}(\theta) = \sum_{i=1}^{21} \|j_i(\theta) - j_i'\|^2 \tag{11}$$

We use this initial mesh to segment the hand region in the input image from the background and forearm. Our data term depends on this input segmentation mask. Finally, the material and lighting parameters are randomly initialized.

Implementation We minimize E_{total} using SGD optimizer for frame-specific parameters $\left(\{\theta_f\}_{f=1}^{F}\right)$ and Adam optimizer [25] for other parameters. Our optimization strategy sequentially updates parameters for each frame, over multiple epochs, *i.e.* in each epoch (*i.e.* over all frames), we perform F update steps. We clamp the material parameters after each iteration to ensure physically valid textures. We use a higher learning rate for lighting parameters and a lower learning rate for material parameters. The values of the weights used in the regularization terms of our energy term mentioned in Tab. 1. We use PyTorch [40] for optimization. Given a monocular video with 100 frames of resolution 1024×1024, our method with 100 epochs can reconstruct a reasonable hand avatar within 10 minutes on an NVIDIA GeForce RTX 3080 Ti.

4. Experimental Results

We evaluate our method on the task of hand avatar reconstruction and compare it with state-of-the-art methods. Further, to highlight the impact of our method, we demonstrate applications in novel pose synthesis and re-lighting.

ω_{lap}	$\omega_{\mathbf{k}_d}$	$\omega_{\mathbf{k}_s}$	$\omega_{\mathbf{n}}$	ω_{LPIPS}	ω_{light}
1000	0.1	0.05	0.025	0.1	0.15

Table 1. Regularization weights.

4.1. Datasets

To compare with state-of-the-art methods on hand avatar reconstruction, we use the InterHand2.6M dataset [34]. This data is captured in a dome with controlled lighting and provides fitted MANO parameters obtained using NeuralAnnot [33]. The images are centered, scaled by a factor of 1.3 around the hand, and cropped to 256×256.

Further, to evaluate the method on various hand shapes and appearances, we capture our dataset using a calibrated (known camera intrinsics) RGB camera. We ask the user to start with a stretched hand pose and rotate their hand to show all regions of their hand (see Fig. 1. Our dataset comprises approximately 1000 frames captured from 7 different users. This dataset is challenging because of its real-world lighting and resembles the actual set-up where AR/VR devices are typically deployed.

4.2. Metrics

We report metrics focusing on the rendered image quality, including PSNR, SSIM [58], and LPIPS. Due to the absence of ground-truth meshes for real data, we cannot explicitly evaluate geometry reconstruction. However, the above metrics implicitly capture geometry reconstruction together with appearance reconstruction.

4.3. Evaluation of hand avatar reconstruction

We compare our method with previous hand avatar reconstruction methods HARP [24] and HandAvatar [8] on the InterHand2.6M dataset. Unlike our physically realistic shading, HARP uses shadow mapping in its differentiable rendering pipeline and only supports a single light source. In Fig. 4, we observe that our method produces sharper texture details and handles shadows more accurately than HARP. We also observe that our method accurately captures fine geometric details as seen in the shading images in Fig. 5 whereas these details are missing in the shading images of HandAvatar. We also see that our method produces correct shadows when compared to the self-occlusion-aware shading field in HandAvatar.

We further evaluate our rendering result with recent monocular methods using volume rendering (HumanNeRF [59]) and surface rendering (SelfRecon [20]) in Fig. 6. Unlike our fast test-time optimization (minutes), these methods require long training times (hours).

We quantitatively evaluate our method and compare it with prior methods in Tab. 2 and Tab. 3. We observe that our method outperforms existing methods on all metrics.

Figure 4. We compare the reconstructed hand obtained from our method with HARP [24], which uses a similar shadow-aware differentiable rendering in an analysis-by-synthesis approach. Unlike the blurred output from HARP, our method accurately captures fine-level appearance details.

Method	PSNR↑	SSIM↑	LPIPS↓
SelfRecon [20]	26.38	0.878	0.142
HumanNeRF [59]	27.64	0.883	0.114
Ours	**28.66**	**0.897**	**0.090**

Table 2. Quantitative evaluation on InterHand2.6M 5fps dataset (*test/Capture0/ROM03_RT_No_Occlusion*).

Method	PSNR↑	SSIM↑	LPIPS↓
HARP [24]	16.157	0.866	0.167
HandAvatar [8]	29.423	0.914	0.088
Ours	**31.179**	**0.936**	**0.061**

Table 3. Quantitative evaluation on InterHand2.6M 30fps dataset (images 500 to 999 of *test/Capture0/ROM03_RT_No_Occlusion/cam400266)*.

Finally, we show the output of our method on our captured dataset in Fig. 7. We observe that our method accurately reconstructs detailed textures and disentangles lighting from albedo. Results on additional users can be found in the supplementary material.

4.4. Generalization

We demonstrate the ability of our method to capture out-of-distribution appearance as opposed to MLP-based HandAvatar in Fig. 9 and Tab. 4. This demonstrates the suitability of our method to be deployed in AR/VR applications.

4.5. Ablation Study

We evaluate the influence of our design choices. We show the effect of introducing offsets in the MANO model and subdivision on capturing detailed geometry in Tab. 5.

Additionally, we replace our explicit mesh-based geometry module with a neural signed distance function (SDF)

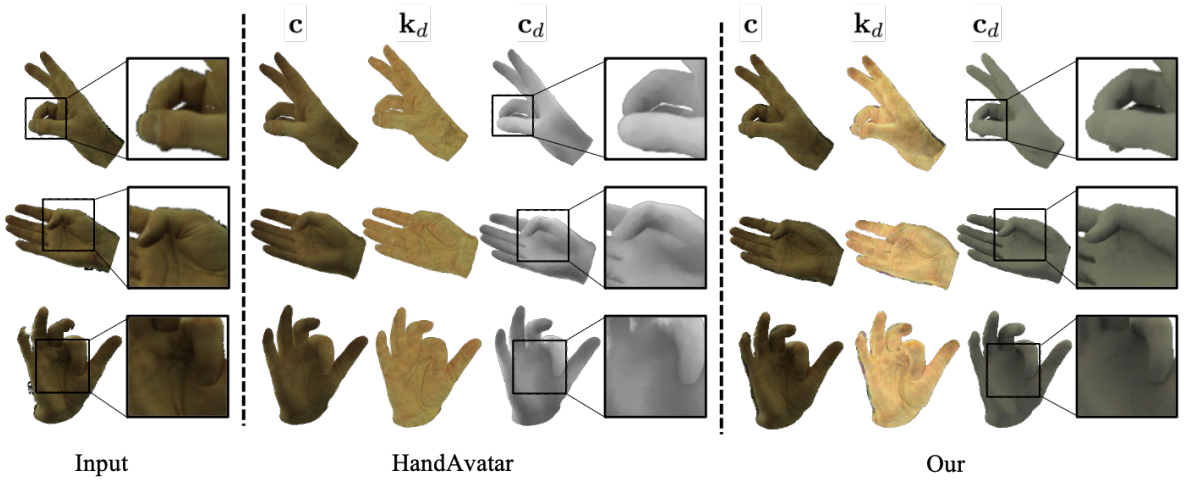

Figure 5. We compare the shadow-aware capability of our method with HandAvatar, which uses a self-occlusion-aware shading field to model self-shadows. Compared to the local-pair occupancy field of HandAvatar, our method explicitly models shadow rays to produce more accurate self-shadows in complicated poses.

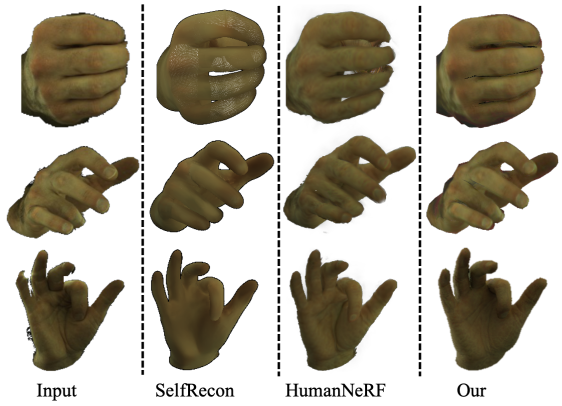

Figure 6. Our method reconstructs high-fidelity texture details compared to the surface rendering method of SelfRecon [20] and correctly captures illumination compared to the volume rendering method of HumanNeRF [59].

Method	PSNR↑	SSIM↑	LPIPS↓
HandAvatar [8]	27.94	0.894	0.091
Ours	**30.20**	**0.926**	**0.060**

Table 4. Quantitative evaluation on InterHand2.6M 5fps dataset (*test/Capture1/ROM03_RT_No_Occlusion/cam400270*) where HandAvatar is trained on *test/Capture0/ROM04_RT_Occlusion*.

integrated with a differentiable marching tetrahedron from NVDIFFRECMC [19]. To follow their assumption of the static object and multi-view set-up, we use 80 images from the InterHand2.6M dataset in a fixed stretched pose (*image13002*) of a user (*capture0*) captured from multiple cameras at a single time instant. In Fig. 8, we can observe that their method fails to capture a human hand from sparse

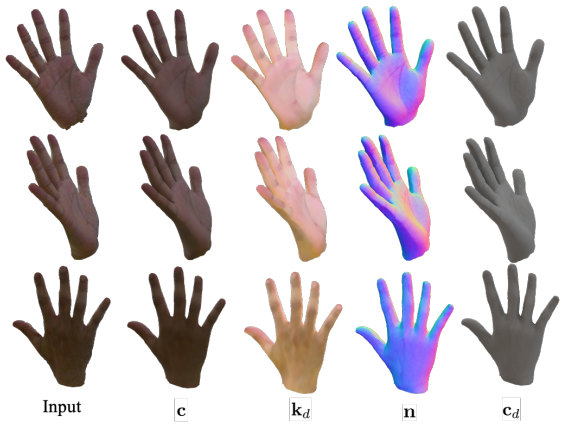

Figure 7. On hands reconstructed from images of our captured dataset (user1), we can see fine geometric details captured on the palm in the normal and shading images.

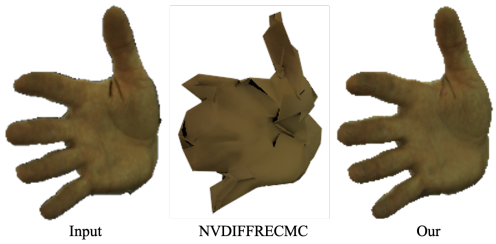

Figure 8. Comparison with the multi-view static 3D object reconstruction method of NVDIFFRECMC [19].

Figure 9. Our method based on test-time-optimization produces the correct appearance of a new user (*capture1* from InterHand2.6M dataset) compared to inferred output from HandAvatar trained on another user's images (*capture0*).

MANO mesh	w/o subdivision	w subdivision
w/o offsets	28.12	28.24
w offsets	28.63	**29.14**

Table 5. Ablation study on the effect of using offsets and subdivision on the original MANO model. We report PSNR on 300 frames from our captured data.

views, whereas our method gracefully reconstructs even in a multi-view set-up.

4.6. Applications

We demonstrate some use cases of our reconstructed hand avatar from our capture dataset (*user1*) with only 40 frames where the user rotates the hand in a stretched pose.

Novel pose synthesis We obtain novel poses from the InterHand2.6M dataset (*test/capture_1/ROM03_RT_No_Occlusion/cam_400270*) and render the reconstructed hand avatar under a new environment light probe obtained from Poly Haven [62]. We observe the consistent appearance and realistic self-shadows as shown in Fig. 1 (middle column).

Re-lighting We demonstrate the interaction of our intrinsic hand avatar with environment lighting. In Fig. 1, we show the same hand avatar placed under different environment lighting and observe consistent rendering.

These applications validate the correctness of our reconstructed intrinsic hand avatar and also emphasize the potential of our method.

5. Discussion

Impact Our method takes an important step in reconstructing an intrinsic hand avatar of a user's hand from a monocular RGB video. Our method addresses one of the main issues of other avatar reconstruction methods which do not disentangle material and lighting, and fail to generalize to new users. Thus, AR/VR applications can leverage our method to reconstruct a user-specific hand avatar capturing the user's hand shape and appearance and providing a more natural and immersive experience.

Limitations Our method is sensitive to a reasonably good initialization of the mesh parameters and the segmentation mask via the data term. Further, our method assumes that the hand in the input video is clearly visible and thus cannot handle occlusion caused by other objects in the scene. If any part of the user's hand is never visible in the video, that region in the reconstructed hand avatar will remain at its initialized value as it never gets updated in the optimization. Although our method can handle dynamic hand poses in the video, it assumes static lighting throughout the video. We clamp the metalness of our material to lower values, which does not allow the modeling of shiny fingernails.

Future work The material model can be improved by incorporating subsurface scattering observed in human skin. Incorporating bulges and wrinkles into the geometry deformation model can greatly improve realism and enable capturing fine-level shape and appearance details. Another interesting direction would be to model fine hair on the back of the hand to improve the realism of the avatar.

Ethical Considerations The recent advances in reconstructing digital avatars raise the concern of nefarious use cases. The foremost danger in reconstructing such a hand avatar from an RGB camera is the use of fingerprints. To protect users' fingerprint privacy, all images in the paper are down-sampled before the acquisition, which also leads to slightly blurry textures.

6. Conclusion

We present a method that generates an intrinsic avatar of the user's hand from a monocular RGB video. Our method jointly optimizes a parametric mesh, physically-based material, and environment lighting using a realistic shading model in a differentiable rendering framework incorporating Monte Carlo path tracing. In contrast to existing methods where illumination is baked into the appearance, our method disentangles the intrinsic properties of the underlying appearance (modeled by *Intrinsic Hand Avatar*) and environment lighting leading to realistic self-shadows. Our method can generate an avatar within minutes and supports arbitrary hand poses captured real-world environment, which enables our method to be used for a variety of applications.

References

[1] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Detailed human avatars from monocular video. In *International Conference on 3D Vision*, pages 98–109, Sep 2018. 3

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

[3] Steve Bako, Thijs Vogels, Brian Mcwilliams, Mark Meyer, Jan NováK, Alex Harvill, Pradeep Sen, Tony Derose, and Fabrice Rousselle. Kernel-predicting convolutional networks for denoising monte carlo renderings. *ACM TOG*, 36(4), 2017. 4

[4] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, page 187–194, USA, 1999. ACM Press/Addison-Wesley Publishing Co. 2

[5] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P.A. Lensch. Nerd: Neural reflectance decomposition from image collections. In *ICCV*, 2021. 3

[6] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3d hand shape and pose from images in the wild. In *CVPR*, pages 10843–10852, 2019. 1, 2

[7] Andrei Burov, Matthias Nießner, and Justus Thies. Dynamic surface function networks for clothed human bodies. In *ICCV*, pages 10754–10764, October 2021. 1

[8] Xingyu Chen, Baoyuan Wang, and Heung-Yeung Shum. Hand avatar: Free-pose hand animation and rendering from monocular video. In *CVPR*, pages 8683–8693, June 2023. 1, 2, 6, 7

[9] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *ICCV*, 2021. 2

[10] Yujin Chen, Zhigang Tu, Di Kang, Linchao Bao, Ying Zhang, Xuefei Zhe, Ruizhi Chen, and Junsong Yuan. Model-based 3d hand reconstruction via self-supervised learning. In *CVPR*, 2021. 1, 2

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

[12] Martin de La Gorce, David J. Fleet, and Nikos Paragios. Model-based 3d hand pose estimation from monocular video. *IEEE TPAMI*, 33(9), 2011. 2

[13] Zijian Dong, Chen Guo, Jie Song, Xu Chen, Andreas Geiger, and Otmar Hilliges. Pina: Learning a personalized implicit neural avatar from a single rgb-d video sequence. In *CVPR*, pages 20470–20480, June 2022. 1

[14] Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia 2022 Conference Papers*, 2022. 2

[15] Yao Feng, Jinlong Yang, Marc Pollefeys, Michael J. Black, and Timo Bolkart. Capturing and animation of body and clothing from monocular video. In *SIGGRAPH Asia 2022 Conference Papers*, SA '22, 2022. 1, 2

[16] Xuan Gao, Chenglai Zhong, Jun Xiang, Yang Hong, Yudong Guo, and Juyong Zhang. Reconstructing personalized semantic facial nerf models from monocular video. *ACM TOG*, 41(6), 2022. 1

[17] Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos P Zafeiriou. Fast-ganfit: Generative adversarial network for high fidelity 3d face reconstruction. *IEEE TPAMI*, 2021. 1

[18] Philip-William Grassal, Malte Prinzler, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. Neural head avatars from monocular rgb videos. In *CVPR*, pages 18653–18664, 2022. 1, 2

[19] Jon Hasselgren, Nikolai Hofmann, and Jacob Munkberg. Shape, light, and material decomposition from images using monte carlo rendering and denoising. In *NeurIPS*, volume 35, pages 22856–22869, 2022. 2, 3, 4, 5, 7

[20] Boyi Jiang, Yang Hong, Hujun Bao, and Juyong Zhang. Selfrecon: Self reconstruction your digital avatar from monocular video. In *CVPR*, 2022. 1, 2, 6, 7

[21] Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Instantavatar: Learning avatars from monocular video in 60 seconds. In *CVPR*, pages 16922–16932, June 2023. 1, 2

[22] James T. Kajiya. The rendering equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques*, page 143–150, 1986. 3

[23] Pratik Kalshetti and Parag Chaudhuri. Local Scale Adaptation to Hand Shape Model for Accurate and Robust Hand Tracking. *Comput. Graph. Forum*, 2022. 1, 2

[24] Korrawe Karunratanakul, Sergey Prokudin, Otmar Hilliges, and Siyu Tang. Harp: Personalized hand reconstruction from a monocular rgb video. In *CVPR*, pages 12802–12813, June 2023. 1, 2, 6

[25] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *ICLR*, 2015. 5

[26] Dominik Kulon, Riza Alp Guler, Iasonas Kokkinos, Michael M. Bronstein, and Stefanos Zafeiriou. Weakly-supervised mesh-convolutional hand reconstruction in the wild. In *CVPR*, 2020. 1, 2

[27] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM TOG*, 39(6), 2020. 4

[28] Yuwei Li, Longwen Zhang, Zesong Qiu, Yingwenqi Jiang, Nianyi Li, Yuexin Ma, Yuyao Zhang, Lan Xu, and Jingyi Yu. Nimble: A non-rigid hand model with bones and muscles. *ACM TOG*, 41(4), jul 2022. 1, 2

[29] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *CVPR*, 2021. 2

[30] Stephen McAuley, Stephen Hill, Naty Hoffman, Yoshiharu Gotanda, Brian Smits, Brent Burley, and Adam Martinez. Practical physically-based shading in film and game production. In *ACM SIGGRAPH 2012 Courses*, 2012. 2, 3

[31] Stephen McAuley, Stephen Hill, Adam Martinez, Ryusuke Villemin, Matt Pettineo, Dimitar Lazarov, David Neubelt, Brian Karis, Christophe Hery, Naty Hoffman, and Hakan Zap Andersson. Physically based shading in theory and practice. In *ACM SIGGRAPH 2013 Courses*, 2013. 4

[32] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 2

[33] Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Neuralannot: Neural annotator for 3d human mesh training sets. In *CVPRW*, 2022. 6

[34] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. Interhand2.6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image. In *ECCV*, 2020. 6

[35] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting Triangular 3D Models, Materials, and Lighting From Images. In *CVPR*, pages 8280–8290, June 2022. 3, 4, 5

[36] Atsuhiro Noguchi, Xiao Sun, Stephen Lin, and Tatsuya Harada. Neural articulated radiance field. In *ICCV*, 2021. 2

[37] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *ICCV*, 2021. 2

[38] Paschalis Panteleris, Iason Oikonomidis, and Antonis Argyros. Using a single rgb frame for real time 3d hand pose estimation in the wild. In *WACV*, 2018. 1

[39] Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. *ICCV*, 2021. 2

[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019. 5

[41] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural Radiance Fields for Dynamic Scenes. In *CVPR*, 2020. 2

[42] Neng Qian, Jiayi Wang, Franziska Mueller, Florian Bernard, Vladislav Golyanik, and Christian Theobalt. Html: A parametric hand texture model for 3d hand reconstruction and personalization. In *ECCV*, page 54–71, Berlin, Heidelberg, 2020. Springer-Verlag. 1, 2, 3

[43] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM TOG*, 36(6):245:1–245:17, 2017. 2, 3, 4

[44] Christoph Schied, Anton Kaplanyan, Chris Wyman, Anjul Patney, Chakravarty R. Alla Chaitanya, John Burgess, Shiqiu Liu, Carsten Dachsbacher, Aaron Lefohn, and Marco Salvi. Spatiotemporal variance-guided filtering: Real-time reconstruction for path-traced global illumination. In *Proceedings of High Performance Graphics*, 2017. 4

[45] Kaiyue Shen, Chen Guo, Manuel Kaufmann, Juan Zarate, Julien Valentin, Jie Song, and Otmar Hilliges. X-avatar: Expressive human avatars. *CVPR*, 2023. 1, 2

[46] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, 2017. 2

[47] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian surface editing. In *Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, page 175–184, 2004. 5

[48] Shih-Yang Su, Frank Yu, Michael Zollhöfer, and Helge Rhodin. A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. In *NeurIPS*, 2021. 1, 2

[49] Andrea Tagliasacchi, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. Robust articulated-icp for real-time hand tracking. In *Comput. Graph. Forum*, volume 34:5, pages 101–114, 2015. 1

[50] Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, Arran Topalian, Erroll Wood, Sameh Khamis, Pushmeet Kohli, Toby Sharp, Shahram Izadi, Richard Banks, Andrew Fitzgibbon, and Jamie Shotton. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. *ACM TOG*, 35, 2016. 1, 2

[51] Anastasia Tkach, Andrea Tagliasacchi, Edoardo Remelli, Mark Pauly, and Andrew Fitzgibbon. Online generative model personalization for hand tracking. *ACM TOG*, 36(6):1–11, 2017. 1, 2

[52] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *ICCV*. IEEE, 2021. 2

[53] Andrey Vakunov, Chuo-Ling Chang, Fan Zhang, George Sung, Matthias Grundmann, and Valentin Bazarevsky. Mediapipe hands: On-device real-time hand tracking. In *Workshop on Computer Vision for AR/VR*, 2020. 2, 4, 5

[54] Eric Veach and Leonidas J. Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, page 419–428, New York, NY, USA, 1995. Association for Computing Machinery. 4

[55] C. Wan, T. Probst, L. Gool, and A. Yao. Dense 3d regression for hand pose estimation. In *CVPR*, 2018. 1

[56] Jiayi Wang, Franziska Mueller, Florian Bernard, Suzanne Sorli, Oleksandr Sotnychenko, Neng Qian, Miguel A. Otaduy, Dan Casas, and Christian Theobalt. RGB2Hands: Real-Time Tracking of 3D Hand Interactions from Monocular RGB Video. *ACM TOG*, 39(6), 2020. 1

[57] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021. 2

[58] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4):600–612, 2004. 6

[59] Chung-Yi Weng, Brian Curless, Pratul P. Srinivasan, Jonathan T. Barron, and Ira Kemelmacher-Shlizerman. HumanNeRF: Free-viewpoint rendering of moving people from monocular video. In *CVPR*, pages 16210–16220, June 2022. 1, 6, 7

[60] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *NeurIPS*, 2021. 2

[61] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. PlenOctrees for real-time rendering of neural radiance fields. In *ICCV*, 2021. 2

[62] Greg Zaal. Poly haven. `https://polyhaven.com/hdris`, 2021. 8

[63] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. PhySG: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In *CVPR*, 2021. 3

[64] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 5

[65] Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul Debevec, William T. Freeman, and Jonathan T. Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM TOG*, 40(6), 2021. 3, 5

[66] Yuanqing Zhang, Jiaming Sun, Xingyi He, Huan Fu, Rongfei Jia, and Xiaowei Zhou. Modeling indirect illumination for inverse rendering. In *CVPR*, 2022. 3

[67] Hao Zhao, Jinsong Zhang, Yu-Kun Lai, Zerong Zheng, Yingdi Xie, Yebin Liu, and Kun Li. High-fidelity human avatars from a single rgb camera. In *CVPR*, 2022. 1, 2

[68] Yufeng Zheng, Victoria Fernández Abrevaya, Marcel C. Bühler, Xu Chen, Michael J. Black, and Otmar Hilliges. I M Avatar: Implicit morphable head avatars from videos. In *CVPR*, 2022. 1, 2

[69] Yuxiao Zhou, Marc Habermann, Weipeng Xu, Ikhsanul Habibie, Christian Theobalt, and Feng Xu. Monocular real-time hand shape and motion capture using multi-modal data. In *CVPR*, 2020. 2