Parametric Hand Texture Model for 3D Hand Reconstruction and Personalization

Neng Qian, Jiayi Wang, Franziska Mueller, Florian Bernard, Vladislav Golyanik, Christian Theobalt

MPI–I–2020–4-001 June 2020

Authors' Addresses

Neng Qian
Jiayi Wang
Franziska Mueller
Florian Bernard
Vladislav Golyanik
Christian Theobalt

Max-Planck-Institut für Informatik
Campus E 1 4

D-66123 Saarbrücken

Acknowledgements

The authors would like to thank all participants of the recordings for data acquisition. The work was supported by the ERC Consolidator Grant *4DRepLy* (770784).

Abstract

3D hand reconstruction from image data is a widely-studied problem in computer vision and graphics, and has a particularly high relevance for virtual and augmented reality. Although several 3D hand reconstruction approaches leverage hand models as a strong prior to resolve ambiguities and achieve a more robust reconstruction, most existing models account only for the hand shape and poses and do not model the texture. To fill this gap, in this work we present the first parametric texture model of human hands. Our model spans several dimensions of hand appearance variability (*e.g.*, related to gender, ethnicity, or age) and only requires a commodity camera for data acquisition. Experimentally, we demonstrate that our appearance model can be used to tackle a range of challenging problems such as 3D hand reconstruction from a single monocular image. Furthermore, our appearance model can be used to define a neural rendering layer that enables training with a self-supervised photometric loss. We make our model publicly available.

Keywords

hand texture model, appearance modeling, hand tracking, 3D hand reconstruction

Contents

1 Introduction

Hands are one of the most natural ways for humans to interact with their environment. And as interest in virtual and augmented reality (VR/AR) grows, so does the need for reconstructing a user's hands to enable intuitive and immersive interactions with the virtual environment. Ideally, this reconstruction contains accurate hand shape, pose, and appearance. However, it is a challenging task to capture a user's hands from just images due to the complexity of hand interactions and self-occlusion. In recent years, there has been significant progress in hand pose estimation from monocular depth [51, 28, 52, 1, 14, 24, 8] and RGB [55, 44, 7, 53] images. Although most of these works estimate only joint positions, a few recent works attempt to reconstruct the hand geometry as well [25, 6, 2, 54].

Despite these recent advances, there is little work that addresses the reconstruction of hand appearance. However, hand appearance personalization is important for increasing immersion and the sense of "body-ownership" in VR applications [21], and for improved tracking and pose estimation through analysis-by-synthesis approaches. Without a personalized appearance model, existing pose estimation methods must use much coarser hand silhouettes [6, 2, 54] as an approximation of appearance. One approach to obtain a personalized hand texture is to project the tracked geometry to the RGB image and copy the observed color to the texture map [10]. However, only a partial appearance of the observed hand parts can be recovered with this method and tracking errors can lead to unnatural appearances. In addition, without explicit lighting estimation, lighting effects will be baked into the results of these projection-based methods.

To address this gap, we present for the first time a data-driven parametric model of hand appearance (see Fig. 1.1). We captured a large variety of hands and aligned the scans in order to enable principal component analysis (PCA) and build a textured parametric hand model. PCA compresses the variations of natural hand appearances to a low dimensional appearance basis, thus enabling a more robust appearance fitting. Our model can additionally produce

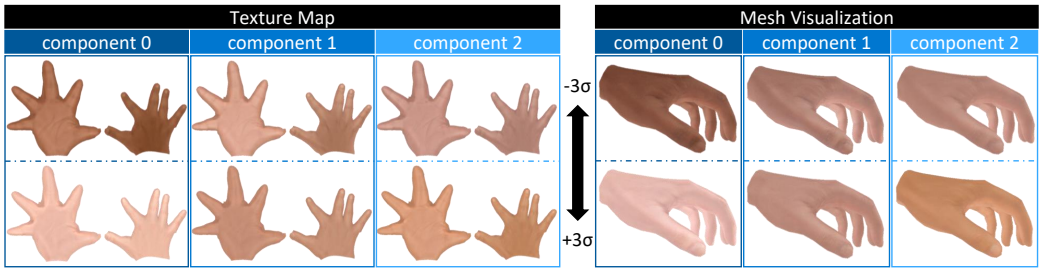

Figure 1.1: We present the first parametric hand texture model. Our model successfully captures appearance variations from different gender, age, and ethnicity.

plausible appearance of the entire hand from fitting to partial observations from a single RGB image. Our main **contributions** can be summarized as follows:

- We introduce a novel parametric model of hand texture that we make publicly available. Our model is based on a dataset of high-resolution hand scans of 51 subjects with variety in gender, age, and ethnicity.
- We register our scans to the popular MANO hand model [37] in order to create a statistical hand appearance model that is also compatible with MANO.
- We demonstrate that our new parametric texture model allows to obtain a personalized 3D hand mesh from a single RGB image of the user's hand.
- We present a proof-of-concept neural network layer which uses the MANO shape and pose model in combination with our proposed texture model in an analysis-by-synthesis fashion. It enables a self-supervised photometric loss, directly comparing the textured rendered hand model to the input image.

2 Related Work

The use of detailed, yet computationally efficient, hand models for hand tracking applications is well studied [29, 33, 39, 46]. Nevertheless, many such methods require time-consuming expert adjustments to personalize the model to a user's hand, making them difficult to deploy to the end-user. Therefore, we focus our review to methods that can automatically generate personalized articulated hand models from images. However, we will see that almost all these methods exclusively consider shape personalization and do not include texture or appearance.

2.1 Modeling Hand Geometry

Two types of personalizable hand models exist in the literature, *i.e.*, heuristic parameterizations that directly move and scale the geometric primitives of the models [10, 45, 49, 35, 50], and data-driven statistical parameterizations that model the covariance of hand geometry [22, 37]. Although heuristic approaches are expressive, infeasible hand-shape configurations can arise when fitting such models to single images due to ambiguities between shape and pose. Thus, existing approaches must perform the personalization offline over a set of depth images [45, 49, 35], or design additional heuristic constraints [50] to resolve these ambiguities. On the other hand, data-driven parameterizations [22] provide a low-dimensional shape representation and natural priors on hand configurations. The recent MANO model [37] additionally provides learned data-driven pose-dependent shape corrections to the geometry to avoid artifacts in posing a hand model through *linear blend skinning* (LBS). This model has been applied in many recent hand pose estimation methods [6, 54, 2, 18, 2] and has been used to annotate hand pose estimation benchmarks [56, 18, 17].

Nonetheless, and despite the popularity of the MANO model of hand geometry, there exists no data-driven parametric texture model for providing

realistic appearance. As such, in this work we present for the first time a hand appearance model that is fully compatible with MANO. Although MANO has a rather low-resolution mesh (778 vertices), our appearance model is defined in texture space so that a much higher texture resolution is available.

2.2 Modeling Appearance

With a few exceptions [10, 11], the previously mentioned works do not model hand texture. The works of de La Gorce *et al.* [10, 11] incorporate heuristic texture personalization for hand-tracking using an analysis-by-synthesis approach. Their approach obtains only a partial estimate of the hand texture using the current pose estimate, and relies on a smoothness prior to transfer color to unobserved parts by a diffusion process on a per-frame basis. Romero *et al.* [37] provide the raw RGB scans used to register the MANO model, but they contain strong lighting effects like shadows and over-exposed regions. Hence, it is not possible to recover accurate appearance from these scans as we show in the supplementary document. Despite the lack of a parametric hand texture model, the benefits of having such a model can be readily seen in face modeling literature. For example, 3D morphable face models (3DMM) [5, 20, 31, 15, 9] provide parametric geometry and appearance models for faces that have been used to drive research in many recent works in diverse applications [12]. For example, these 3DMMs were used within analysis-by-synthesis frameworks for RGB tracking [36, 48], and as unsupervised loss for learning-based methods [47]. Our proposed parametric hand appearance model has the potential to drive similar advances in the hand pose estimation and modeling community.

3 Textured Parametric Hand Model

Our hand texture acquisition pipeline is summarized in Fig. 3.1. First, we record two image sequences observing the palm side and the backside of the hand, respectively. Subsequently, we run rigid structure from motion (SfM) [3, 38] to obtain a 3D reconstruction of the observed hand side (Sec. 3.1). Next, we remove the scene background, and register both (partial) hands scans to the MANO model [37] based on nonlinear optimization. Afterwards, the texture of the partial hand scans is mapped to the registered mesh. We then remove shadows from the textures and stitch them to obtain a complete texture of the hand (Sec. 3.2). The textured parametric model in subsequently generated using PCA (Sec. 3.3).

3.1 Data Acquisition

In total, we captured 51 subjects with varying gender, age, and ethnicity (see Fig. 3.2). To minimize hand motion during scanning, we record the upper and lower hand sides separately, so that the subjects can rest their hand on a flat surface. As such, for each subject we obtain four scans, *i.e.*, back and palm sides for both left and right hands. The scanning takes $\sim$90 seconds for one hand side, so that the total scanning time of $\sim$6 minutes is required per person.

To obtain 3D hand scans, we use SONY's 3DCreator App [43]. The 3D reconstruction pipeline includes three stages, *i.e.*, initial anchor point extraction, simultaneous localization and mapping (SLAM) with sparse points [23], and online dense 3D reconstruction (sculpting) [42]. The output is a textured high-resolution surface mesh (of one hand side as well as the background), which contains $\sim 6.2k$ vertices and $\sim 11k$ triangles in the hand area on average. By design, our hand texture model is built for the right hand.

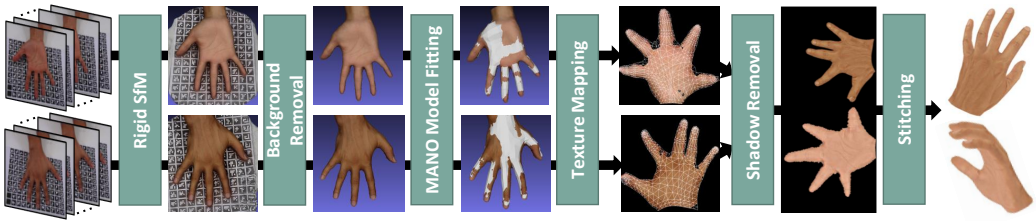

Figure 3.1: **Overview of our hand texture acquisition pipeline.** We run rigid structure from motion (SfM) on a set of input images to obtain a scanned mesh for back and palm side of the hand, respectively. After removing background vertices, we fit the MANO template mesh to extract the texture from the scan. We remove lighting effects and seamlessly stitch the front and back texture, resulting in a complete texture for the captured hand (visualized on the 3D hand mesh from 2 virtual views on the right).

For model creation we mirror the left hand meshes, so that we use a total of 102 "right" hands for modeling. We note that by mirroring we can also use the texture model of "right" hand for the left hand. In the following, we will abstract away this technical detail and describe our texture modeling approach for a single hand.

3.2 Data Canonicalization

To learn the texture variations in a data-driven manner it is crucial that the acquired 3D scans are brought into a common representation. Due to the popularity and the wide use of the MANO model of hand geometry, we decided to build the hand texture in the MANO space. This has the advantage that existing hand reconstruction and tracking frameworks that are based on MANO, such as [27, 6, 18], can be directly extended to also incorporate hand texture. We point out that our texture model can also be used with other models by defining the respective UV mapping. Our data canonicalization comprises several consecutive stages, *i.e.*, *background removal*, *MANO model fitting*, *texture mapping*, *shadow removal*, and *seamless stitching*, which we describe next.

3.2.1 Background Removal

For each hand we have reconstructed two textured meshes, one that shows the hand palm-down on a flat surface, and one that shows the hand palm-up on a flat surface (cf. Sec. 3.1). In both cases, the background, *i.e.*, the flat

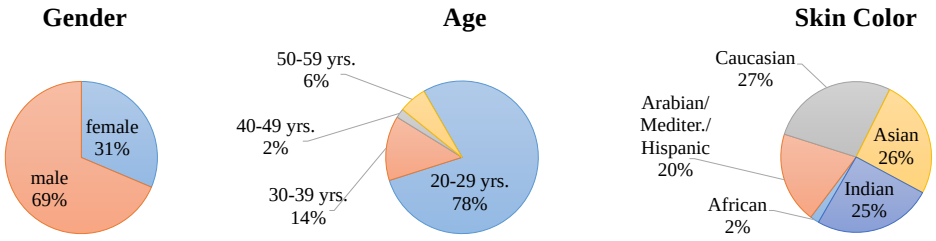

Figure 3.2: Distribution of age, gender, and skin color for our 51 captured subjects. We use the Goldman world classification scale [40] for classifying skin color.

surface that the hand is resting on, is also reconstructed as part of the mesh. Hence, in order to remove the background, we perform a robust plane fitting based on RANSAC [13], where a plane is fitted to the flat background surface. To this end, we sample 100 random configurations of 3 vertices, fit a plane to the sampled points, and then count the number of inliers. Any point that has a distance to the fitted plane that is smaller than the median edge length of the input scanned mesh is considered as inlier. Eventually, the plane that leads to the largest inlier count is considered the background plane. We have empirically found that this approach is robust and able to reliably identify the flat surface in all cases. Eventually, we use a combination of distance-based and color-based thresholding to discard background vertices in the scanned mesh. In particular, we discard a vertex if its distance from the background plane is less than 1cm and the difference between the red and green channel of the vertex color is smaller than 30 (RGB $\in [0, 255]^3$). This yields better preservation of hand vertices that are close to the background plane.

3.2.2 MANO Model Fitting

Subsequently, we fit the MANO hand model to the filtered hand mesh (*i.e.,* the one without the background). To this end, we first obtain the MANO shape and pose parameters based on the hand tracking approach of Mueller *et al.* [27]. The approach uses a Gauss-Newton optimization scheme that makes use of additional information based on trained machine learning predictors (*e.g.,* for correspondence estimation). Since their method was developed for 3D reconstruction and tracking of hands in *depth images*, we render synthetic depth images from our partial hand scan meshes. Note that the approach [27] was partially trained on synthetic depth images and thus we have found that it is able to produce sufficiently good fits of the MANO geometry to our data.

However, since the MANO model is relatively coarse (778 vertices), and

more importantly, it has a limited expressivity of hand shape (it only spans the variations of their training set of 31 subjects), we have found that there are still some misalignments. To also allow for deformations outside the shape space of the MANO model, we hence use a complementary non-rigid refinement. To this end, we use a variant of non-rigid *iterative closet point* (ICP) [4] that optimizes for individual vertex displacements that further refine the template, which in our case is the fitted MANO model. As our objective function, we use 3D point-to-point and point-to-plane distances together with a spatial smoothness regularizer [16]. An accurate alignment is especially important at salient points, like fingertips, to ensure high perceptual quality. Hence, we add prior correspondences for the fingertips and the wrist to the non-rigid ICP fitting. We automatically obtain these correspondences in the input scanned mesh using OpenPose [41]. The influence of the prior correspondences is shown in our evaluation (see Section 5.1).

3.2.3 Texture Mapping

After having obtained an accurate alignment of the hand template, *i.e.*, the fitted MANO model plus non-rigid deformation for refinement, to our textured high-resolution hand scan, we transfer the scan texture to a texture map. To this end, we have manually defined UV coordinates for the MANO model template by unwrapping the mesh to a plane (see texture mapping step in Fig. 3.1). We project each vertex in the high-resolution hand scan to the closest point on the surface of the fitted MANO hand template. Using the barycentric coordinates of this projected point together with the UV coordinates of the template mesh, we transfer the color to the texture map. After performing this procedure for all vertices of our high-resolution hand scan, there can still be some texels (pixels in the texture map) that are not set (we have found that about 6.5% of the hand interior does not have a defined texture). To deal with that, holes are filled based on inpainting with neighboring texels.

3.2.4 Shadow Removal

While we aim to minimize the amount of lighting effects during scanning, the obtained texture map still contains smooth shading (see Fig. 3.3a). Since the shading effects have low frequency, they can be separated and removed using a Laplacian image pyramid. To this end, we first build a Laplacian pyramid with 5 levels from the texture map that we obtained in the previous step. We observe that the deepest level separates the (almost) constant skin color as well as the smooth shading from the texture details that are kept on earlier

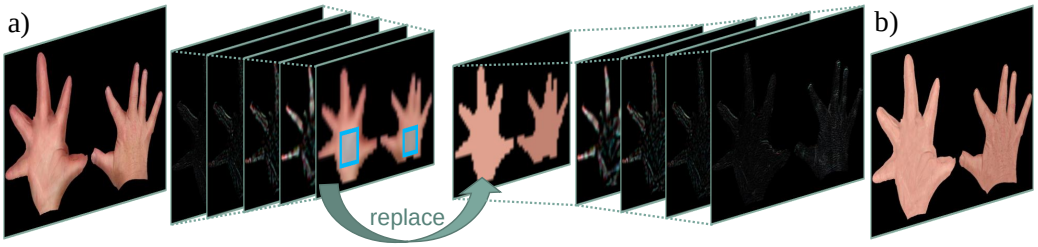

Figure 3.3: **Shadow removal.** (a) Original texture and its Laplacian pyramid decomposition. (b) The shading effects are removed by modifying the deepest level.

levels of the pyramid. We replace this deepest level with a constant skin color for palm and back side, respectively, effectively removing the smooth shading. We obtain this constant skin color by averaging in the well-lit area (see blue rectangles in Fig. 3.3). Note how the texture details from higher levels are preserved in the modified texture map (see Fig. 3.3b).

3.2.5 Seamless Texture Stitching

Since so far this texture mapping is performed both for the palm-up and palm-down facing meshes, we eventually blend both partial texture maps to obtain a complete texture map of the hand. To this end, we use a recent gradient-domain texture stitching approach that directly operates in the texture atlas domain while preserving continuity induced by the 3D mesh topology across atlas chart boundaries [32].

3.3 Texture Model Creation

Let $\{T_i\}_{i=1}^n$ be the collection of 2D texture maps that we obtain after data canonicalization as described in Sec. 3.2. In order to create a parametric texture model we employ PCA. We vectorize each T_i to obtain the vector $t_i \in \mathbb{R}^{618,990}$ that stacks the red, green and blue channels of all hand texels. PCA first computes the data covariance matrix

$$C = \frac{1}{n-1} \sum_{i=1}^{n} (t_i - \bar{t})(t_i - \bar{t}) \,, \tag{3.1}$$

for $\bar{t} = \frac{1}{n} \sum_{i=1}^{n} t_i$ being the average texture. Subsequently, eigenvalue decomposition of $C = \Phi \Lambda \Phi^T$ is used to obtain the principal components Φ and

the diagonal matrix of eigenvalues Λ. With that we obtain the parametric texture model for the parameter vector $\alpha \in \mathbb{R}^k$ as

$$t(\alpha) = \bar{t} + \Phi\,\alpha\,. \tag{3.2}$$

The resulting PCA model has $k = 101$ parameters.

4 Applications

To demonstrate possible use cases of our parametric hand appearance model, we present two applications. First, we consider 3D hand reconstruction and personalization from a single monocular RGB image. Subsequently, we show the usage as a neural network layer enabling a self-supervised photometric loss.

4.1 3D Hand Personalization from a Single Image

Given a single monocular RGB image of a hand, we aim to reconstruct a 3D hand mesh that is personalized to the user's shape and appearance. This application consists of four steps: (1) initialization of shape and pose parameters of the MANO model, (2) non-rigid shape and pose refinement, (3) partial texture extraction, and (4) estimation of appearance parameters of our model.

4.1.1 Shape and Pose Initialization

We use the method of Boukhayma *et al.* [6] to obtain an initial pose and shape estimate of the MANO template mesh from a single RGB image. As discussed before, the MANO shape space is not always expressive enough to perfectly fit the user's hand shape. In addition, the results from the method by Boukhayma *et al.* do not yield sufficiently accurate reprojection of the mesh onto the image plane as shown in Fig. 4.1 (second from the left). Hence, this initial mesh is further refined.

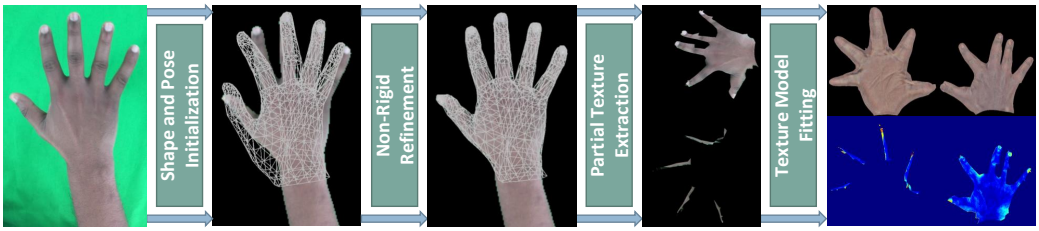

Figure 4.1: **3D hand personalization from a single image.** Starting from a single RGB input image (left), we first initialize the mesh using the method by Boukhayma *et al.* [6]. Next, we refine the fit non-rigidly and extract the partial hand texture. By fitting our parametric texture model, we are able to obtain a complete texture which minimizes the error to the input texture (right).

4.1.2 Non-Rigid Refinement of the Initial Mesh

We non-rigidly refine the initial mesh estimate to better fit the silhouette of the hand in the image. In particular, this is done by optimizing for the 3D displacement of each vertex using ICP constraints on the boundary vertices. We first define the set of boundary vertices of the hand mesh $\bar{\mathcal{V}} \subset \mathcal{V}$, *i.e.*, the set of vertices on the silhouette. Let $\Pi : \mathbb{R}^3 \rightarrow \Omega$ be the camera projection converting from 3D world coordinates to 2D pixel locations. For each boundary vertex $\bar{\boldsymbol{v}}_i$, we first find the closest hand silhouette pixel $\bar{\boldsymbol{p}}_i$ in the image domain Ω as

$$\bar{\boldsymbol{p}}_i = \underset{\boldsymbol{p} \in \Omega}{\arg\min} \; ||\Pi(\bar{\boldsymbol{v}}_i) - \boldsymbol{p}||_2 \quad \text{s.t.} \quad n(\boldsymbol{p})^\top \Pi(n(\bar{\boldsymbol{v}}_i)) > \eta \,. \tag{4.1}$$

Here, $n(\boldsymbol{p})$ is the 2D boundary normal at pixel $\boldsymbol{p}$ (calculated by Sobel filtering), and $\Pi(n(\bar{\boldsymbol{v}}_i))$ is the 2D image-plane projection of the 3D vertex normal at $\bar{\boldsymbol{v}}_i$. The threshold $\eta = 0.8$ discards unsuitable pixels based on normal dissimilarity. We then use this closest hand silhouette pixel $\bar{\boldsymbol{p}}_i$ as correspondence for boundary vertex $\bar{\boldsymbol{v}}_i$ if it is closer than δ (= 4% of the image size)

$$\bar{\boldsymbol{c}}_i = \begin{cases} \bar{\boldsymbol{p}}_i, & \text{if } ||\Pi(\bar{\boldsymbol{v}}_i) - \bar{\boldsymbol{p}}_i||_2 < \delta \\ \emptyset, & \text{otherwise} \end{cases} \,. \tag{4.2}$$

We can then optimize for the refined 3D vertex positions using the computed correspondences in the following objective function:

$$E(\mathcal{V}) = \frac{1}{|\bar{\mathcal{V}}|} \sum_{\bar{\boldsymbol{v}}_i \in \bar{\mathcal{V}}} ||\Pi(\bar{\boldsymbol{v}}_i) - \bar{\boldsymbol{p}}_i||_2^2 + w_{\text{smth}} \sum_{\boldsymbol{v}_j \in \mathcal{V}} \sum_{\boldsymbol{v}_k \in \mathcal{N}_j} \frac{1}{|\mathcal{N}_j|} ||(\boldsymbol{v}_j - \boldsymbol{v}_k) - (\boldsymbol{v}_j^0 - \boldsymbol{v}_k^0)||_2^2 \,, \tag{4.3}$$

where $\mathcal{N}_j$ is the set of neighboring vertices of vertex $\boldsymbol{v}_j$, and $\mathcal{V}^0 = \{\boldsymbol{v}_\bullet^0\}$ are the vertex positions obtained from the previous ICP iteration. We use 20 ICP iterations in total where $\mathcal{V}, \mathcal{V}^0$ are initialized from the shape and pose initialization step as described above.

4.1.3 Partial Texture Extraction

For each fully visible triangle, *i.e.*, when all its 3 vertices are visible, we extract the color from the input image and copy it to the texture map. This yields a partial texture map where usually at most half the texels have a value assigned and all other texels are set to $\emptyset$. We then obtain the vectorized target texture map t^{trgt} (as for model creation in Section 3.3).

4.1.4 Estimation of Appearance Parameters

Subsequently, we find the appearance parameters of our model that best fit the user's hand by solving the least-squares problem with Tikhonov regularization:

$$\underset{\alpha \in \mathbb{R}^k}{\arg\min} \sum_{t_i^{\text{trgt}} \neq \emptyset} (t_i^{\text{trgt}} - t(\alpha)_i)^2 + w_{\text{reg}} ||\alpha||_2^2 \,. \tag{4.4}$$

Note that our proposed parametric appearance model enables us to obtain a complete texture. In contrast to the extracted partial texture, the result is free of lighting effects and artifacts caused by small misalignments of the hand model.

4.2 Self-Supervised Photometric Loss

Several previous works have trained a neural networks to regress joint positions or MANO model parameters from RGB images [26, 55, 53, 7, 44, 56]. The most common loss is the Euclidean distance between the regressed and ground-truth 2D or 3D joint positions. Some works have also explored a silhouette loss between the MANO mesh and the hand region in the input image [6, 2, 54]. Our proposed parametric hand texture model makes it possible to use a *self-supervised* photometric loss, which can complement the mentioned fully supervised losses. With that, when training a network to predict shape and pose with such an approach, we additionally obtain a hand texture estimate. To this end, we introduce a *textured hand model layer*, which we explain now.

4.2.1 Textured Hand Model Layer

Given a pair of MANO shape and pose parameters (β, θ), as well as the texture parameters α, our model layer computes the textured 3D hand mesh $\mathcal{M}(\beta, \theta, \alpha)$. An image of this mesh is then rendered using a scaled orthographic projection. As such, this rendered image can directly be compared to the input image $\mathcal{I}$ using a photometric loss in an analysis-by-synthesis manner. We formulate the photometric loss as

$$\mathcal{L}_{\text{photo}}(\beta, \theta, \alpha) = \frac{1}{|\Gamma|} \sum_{(u,v)\in\Gamma} ||\text{render}(\mathcal{M}(\beta, \theta, \alpha))(u, v) - \mathcal{I}(u, v)||_2 \,, \quad (4.5)$$

where Γ is the set of pixels which the estimated hand mesh projects to. The use of a differentiable renderer makes the photometric loss $\mathcal{L}_{\text{photo}}$ fully differentiable and enables backpropagation for neural network training.

4.2.2 Network Training

We train a residual network with the architecture of ResNet-34 [19] to regress the shape β, pose θ, and texture parameters α from a given input image. In addition to the self-supervised photometric loss, we employ losses on 2D joint positions, 3D joint positions, and L2-regularizers on the magnitude of the shape, pose, and texture parameters. The network is trained in PyTorch [30], using the differentiable renderer provided in PyTorch3D [34]. We assume a single fixed illumination condition for training. We leave the joint estimation of additional lighting and material properties to future work.

5 Experiments

In this section, we evaluate our proposed parametric hand texture model, explore different design choices in our hand texture acquisition pipeline, and present results of our two example applications.

5.1 Texture Model Evaluation

5.1.1 Compactness

Fig. 5.1 (left) shows the compactness of our texture model. The plot describes how much the explained variance in the training dataset increases with the number of used principal components. The first few components already explain a significant amount of variation since they account for more global changes in the texture, *e.g.*, skin tone. However, adding more components continuously increases the explained variance.

5.1.2 Generalization

For evaluating generalization, we use a leave-one-subject-out protocol. We remove the data of one subject, *i.e.*, the two texture samples from left and right hand, and build the texture PCA model from the remaining subjects. Then, we aim to reconstruct the left-out textures using the built model and measure the reconstruction error as the mean absolute distance (MAD) of the vectorized textures. As shown in Fig. 5.1 (middle), our model is able to obtain a low reconstruction error for all subjects.

5.1.3 Specificity

We also evaluate the specificity of our model, which quantifies the similarity between random samples from the model and the training data. To this end, we first randomly sample a texture instance from our model based on a

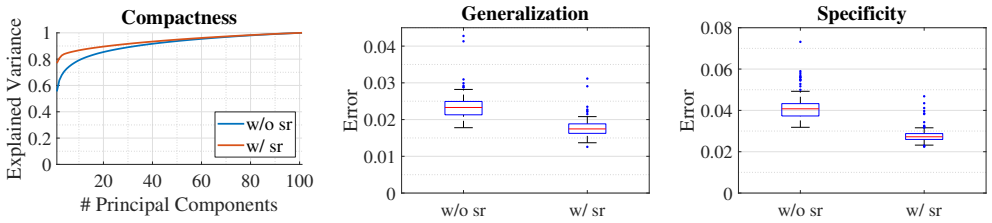

Figure 5.1: Quantitative evaluation of our model in terms of compactness, generalization and specificity. Note that the version with shadow removal ("w/ sr") substantially outperforms the version without shadow removal ("w/o sr").

multivariate standard Normal distribution (due to the Gaussian assumption of PCA). Then, for this random texture we find the nearest texture in our training dataset in terms of the MAD. We repeat this procedure 200 times, and report the statistics of the MAD in the boxplot in Fig. 5.1 (right).

5.1.4 Influence of Shadow Removal

Fig. 5.1 also shows compactness, generalization, and specificity for a version of the texture model that was built without shadow removal ("w/o sr"). It can be seen that the version without shadow removal performs worse compared to the one with shadow removal ("w/ sr") in all metrics. When the lighting effects are not removed, they increase the variance in the training dataset. Hence, more principal components are necessary to explain variation and the reconstruction of unseen test samples has a higher error. In the supplemental material, we also show visually that the principal components for the model without shadow removal have to account for strong lighting variation.

5.1.5 Influence of Prior Correspondences

To ensure a good alignment of the hand template mesh and the scanned mesh, as explained in Section 3.2, for the non-rigid ICP-based refinement step in our model building stage we make use of prior correspondences for the fingertips and the wrist. Fig. 5.2 compares the textures obtained by running the non-rigid ICP fitting with and without prior correspondences. Especially for the thumb, the tip is often not well-aligned, resulting in a missing finger nail in the texture. Using explicit prior correspondences alleviates this issue.

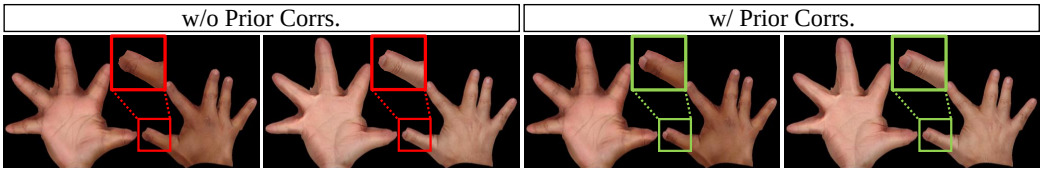

Figure 5.2: Using non-rigid ICP-based refinement with prior correspondences for fingertips and the wrist improves the alignment of the hand template mesh to the scanned mesh, yielding better textures (right). (Textures shown before shadow removal.)

5.2 Application Results: 3D Hand Personalization

Here, we show results for obtaining a personalized 3D hand model from a single RGB image (see Section 4.1). As discussed in Section 4.1, since the output mesh of state-of-the-art regression approaches [6] do not have a low reprojection error, we use a non-rigid refinement step based on silhouettes. To simplify segmentation in our example application, we captured the images of the users in front of a green screen. We note that this could also be replaced by a dedicated hand segmentation method, which we leave for future work. Fig. 5.3 shows hand model fits as well as complete recovered textures from a

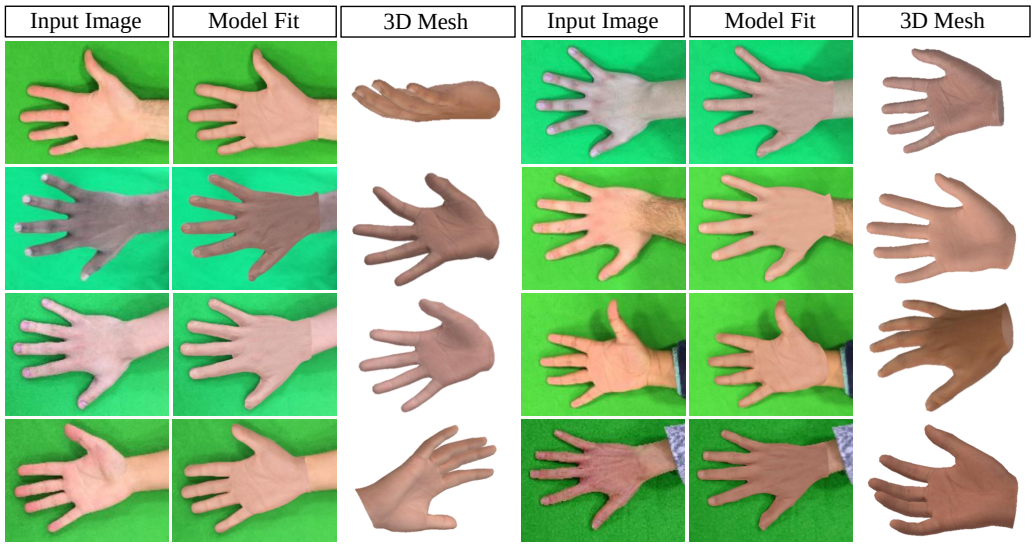

Figure 5.3: Hand personlization from a single RGB input image for different subjects.

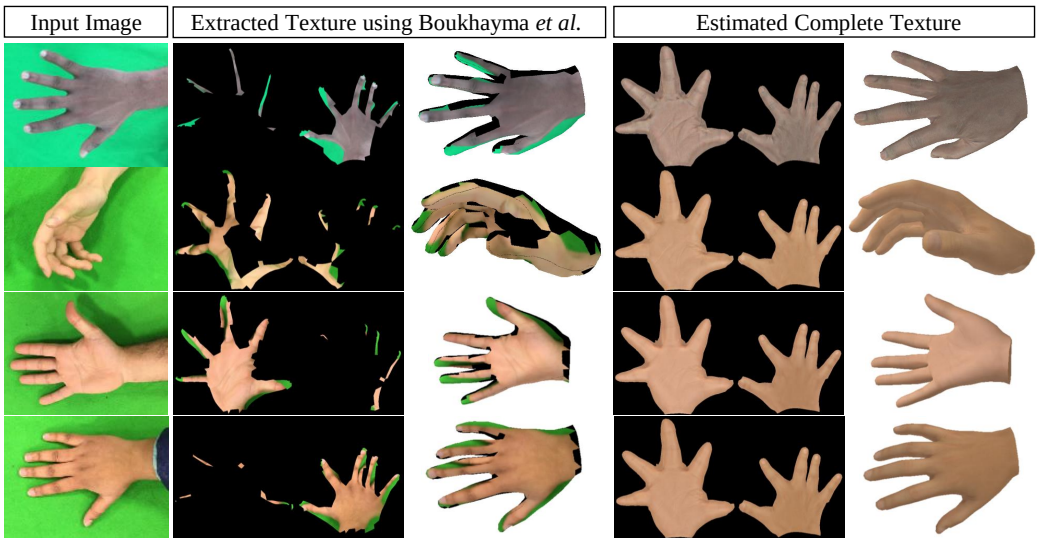

Figure 5.4: Our parametric hand texture model comprises a low-dimensional PCA space. Hence, fitting to noisy or partially corrupted input textures is robust and yields a realistic and complete texture estimate.

single RGB image for several subjects. Since we use a low-dimensional PCA space to model hand texture variation, we can robustly estimate a plausible and complete texture from noisy or partially corrupted input (see Fig. 5.4). In contrast, a texture that is directly obtained by projecting the input image onto a mesh obtained by the method of Boukhayma *et al.* [6] contains large misalignments and a significant amount of background pixels, and thus is severely corrupted.

5.3 Application Results: Photometric Neural Network Loss

Our self-supervised photometric loss (see Section 4.2) enables to not only obtain shape and pose estimates as in previous work, but in addition to also estimate hand appearance. To demonstrate this we train our network on the recently proposed FreiHAND dataset [56]. For details of the experimental setup, please see the supplementary document.

In Fig. 5.5, we show hand model fits predicted by a neural network trained with and without our photometric loss (cf. Sec 4.2). We note that the pose and shape prediction with the photometric loss are quantitatively similar to the predictions without (the mean aligned vertex errors (MAVE) are

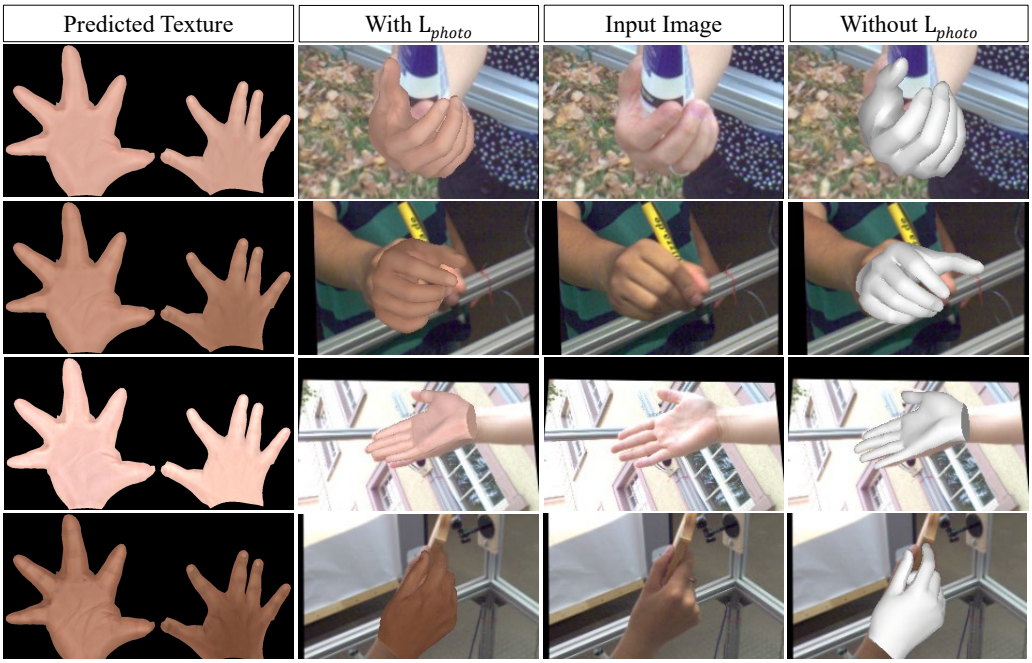

Figure 5.5: We show the predicted pose and texture from a neural network trained using a photometric loss L_{photo} enabled by our parametric hand texture model.

1.10 cm vs 1.14 cm respectively, and mean aligned keypoint errors (MAKE) are 1.11 cm vs 1.14 cm respectively). In addition, these results are comparable to the current state of the art [56] with a MAVE of 1.09 cm and MAKE of 1.10 cm. We stress that our method with the photometric loss additionally infers a high resolution, detailed texture of the full hand, which the other methods do not.

6 Limitations and Discussion

We have presented a parametric hand texture model that can be successfully fitted to various users. Although our model contains detailed texture, the underlying geometry of the MANO mesh is rather coarse (778 vertices). This could be improved by using a mesh with higher-resolution and by extending the MANO shape space with more detailed geometry. Our proposed model was built by using PCA decomposition. In follow-up works, different techniques for building the model could be explored, *e.g.*, learning a suitable subspace with an autoencoder neural network. In our model creation pipeline we explicitly remove lighting effects from the captured textures. When using the hand texture model in our example applications, we did not explicitly model or estimate the scene lighting. Hence, the model might not be able to perfectly explain the input observations. In the future, texture parameters and lighting could be estimated jointly, *e.g.*, for texture personalization or within an analysis-by-synthesis framework.

7 Conclusion

In this work we introduced the first parametric texture model of human hands. The model is based on data that captures 102 hands of people with varying gender, age and ethnicity. For model creation, we carefully designed a data canonicalization pipeline that entails background removal, fitting of the geometric hand model MANO, texture mapping and shadow removal. Moreover, we demonstrate that our model makes two highly relevant applications feasible: 3D hand personalization from a single color image, as well as using differentiable rendering for employing an analysis-by-synthesis approach within a neural learning framework. We make our model publicly available to the research community and thereby expect that in the future this will spawn further works related to the modeling of human hands.

Appendix A Experiment Details for Hand Model Layer

The provided training dataset contains a total of 130,240 images: 32,560 unique images of hands with foreground masks, times 4 methods of background composition. However, 3 of the composition methods attempt to blend the hand into the background, which introduces severe artifacts in the hand appearance (see Fig. A.1).

We only use the unaltered 32,560 unique images for our training data to avoid learning these artifacts for texture estimation. The provided foreground masks were used to perform background augmentation without additional image harmonization or image coloration. We train the ResNet-34 [19] using Adam, with a learning rate of 0.001, and for 200 epochs in all of our experiments.

A full comparison of the pose and shape performance is provided in Table A.1. Following the evaluation procedure of [56], the meshes were aligned using Procrustes alignment as a rigid body transformation. Errors are measured in Euclidean distance (cm) between corresponding vertex points **(Mesh)** or keypoints **(KP)**. Area under the percentage-of-correct-keypoints curve **(AUC)** and F-scores at two different thresholds **(F@5mm and F@15mm)** are additionally provided. Our method achieve slightly better pose and shape performance with the photometric loss **(Proposed)**

Figure A.1: The provided composed Freihand data contain noticeable texture artifacts. These images were not used for training.

Table A.1: Evaluation of our method on the FreiHand dataset [56]. All numbers are from the online leader board. Keypoint (KP) and Mesh errors are measured in cm.

	KP Error	KP AUC	Mesh Error	Mesh AUC	F@5mm	F@15mm
Zimmerman et al. [56]	1.10	0.783	1.09	0.783	0.516	0.934
Boukhayma et al. [6]	3.50	0.351	1.32	0.738	0.427	0.895
Hasson et al. [18]	1.33	0.737	1.33	0.736	0.429	0.907
w/o Photometric	1.14	0.774	1.14	0.774	0.499	0.925
Proposed	1.11	0.781	1.10	0.781	0.508	0.930

than without **(w/o Photometric)**, and it achieves similar accuracy to the current state-of-the-art methods.

Appendix B Principal Components Without Shadow Removal

It is desirable for the parametric texture model to not include lighting effects. Although we aimed to have uniform lighting while acquiring the scans, there are still smooth shading effects, especially at the boundary to the flat background surface. Without the shadow removal step in our pipeline, the lighting effects contribute a large portion of the variation in the dataset. Hence, lighting variations are present in some of the first principal components of the PCA space. Refer to Fig. B.1 and the supplementary video for visualizations.

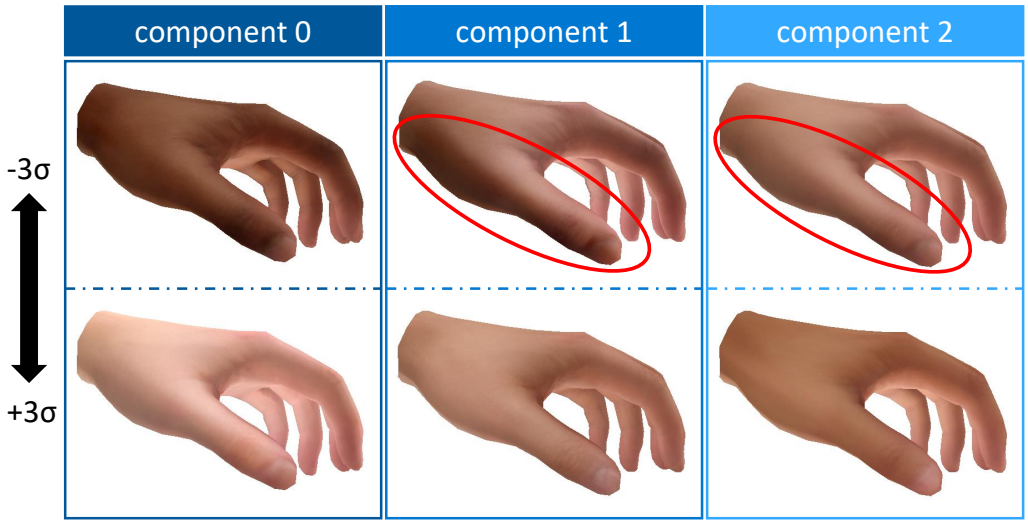

Figure B.1: When we build the texture PCA model without shadow removal, the principal components contain a significant amount of lighting variation.

Appendix C Shadow Removal on Original MANO Scans

The original scans from which the MANO shape and pose model [37] was built do also include vertex colors. However, they contain strong lighting effects like shadows and over-saturated areas. Fig. C.1 shows how our shadow removal procedure fails when applied to textures extracted from the original MANO scans. Note that even if more sophisticated approaches are applied, it is inherently not possible to recover accurate appearance information from completely dark or over-saturated areas. The strong shadows also contain high-frequency components that are difficult to remove while preserving texture details.

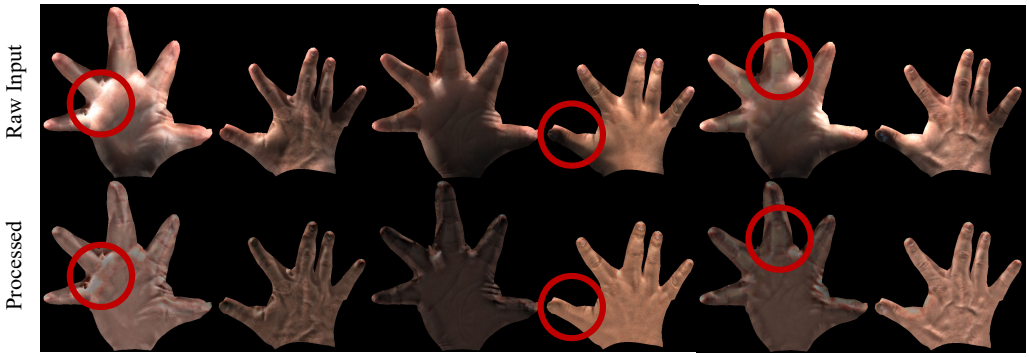

Figure C.1: Results of our shadow removal technique when applied to textures extracted from the original MANO scans. The appearance information cannot be properly recovered due to strong lighting effects (left, middle). The strong shadows also contain high-frequency components that are difficult to remove (right).

Bibliography

[1] S. Baek, K. In Kim, and T.-K. Kim. Augmented skeleton space transfer for depth-based hand pose estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[2] S. Baek, K. I. Kim, and T.-K. Kim. Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[3] C. Bailer, M. Finckh, and H. P. A. Lensch. Scale robust multi view stereo. In *Eurupean Conference for Computer Vision (ECCV)*, 2012.

[4] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 14(2):239–256, 1992.

[5] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, pages 187–194, 1999.

[6] A. Boukhayma, R. d. Bem, and P. H. Torr. 3d hand shape and pose from images in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[7] Y. Cai, L. Ge, J. Cai, and J. Yuan. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *European Conference on Computer Vision (ECCV)*, 2018.

[8] Y. Chen, Z. Tu, L. Ge, D. Zhang, R. Chen, and J. Yuan. So-handnet: Self-organizing network for 3d hand pose estimation with semi-supervised learning. In *International Conference on Computer Vision (ICCV)*, 2019.

[9] H. Dai, N. Pears, W. A. Smith, and C. Duncan. A 3d morphable model of craniofacial shape and texture variation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3085–3093, 2017.

[10] M. de La Gorce, D. J. Fleet, and N. Paragios. Model-based 3d hand pose estimation from monocular video. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33(9):1793–1805, 2011.

[11] M. de La Gorce, N. Paragios, and D. J. Fleet. Model-based hand tracking with texture, shading and self-occlusions. In *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[12] B. Egger, W. A. P. Smith, A. Tewari, S. Wuhrer, M. Zollhöefer, T. Beeler, F. Bernard, T. Bolkart, A. Kortylewski, S. Romdhani, C. Theobalt, V. Blanz, and T. Vetter. 3d morphable face models – past, present and future, 2019.

[13] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[14] L. Ge, Y. Cai, J. Weng, and J. Yuan. Hand pointnet: 3d hand pose estimation using point sets. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] T. Gerig, A. Morel-Forster, C. Blumer, B. Egger, M. Luthi, S. Schönborn, and T. Vetter. Morphable face models-an open framework. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 75–82. IEEE, 2018.

[16] M. Habermann, W. Xu, M. Zollhoefer, G. Pons-Moll, and C. Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions on Graphics (TOG)*, 38(2):1–17, 2019.

[17] S. Hampali, M. Rad, M. Oberweger, and V. Lepetit. Honnotate: A method for 3d annotation of hand and objects poses supplementary material.

[18] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid. Learning joint reconstruction of hands and manipulated objects. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

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

[20] P. Huber, G. Hu, R. Tena, P. Mortazavian, P. Koppen, W. J. Christmas, M. Ratsch, and J. Kittler. A multiresolution 3d morphable face model and fitting framework. In *Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2016.

[21] S. Jung and C. Hughes. Body ownership in virtual reality. In *International Conference on Collaboration Technologies and Systems (CTS)*, pages 597–600, 10 2016.

[22] S. Khamis, J. Taylor, J. Shotton, C. Keskin, S. Izadi, and A. Fitzgibbon. Learning an efficient model of hand shape variation from depth images. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[23] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2007.

[24] S. Li and D. Lee. Point-to-pose voting based hand pose estimation using residual permutation equivariant layer. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[25] J. Malik, A. Elhayek, F. Nunnari, K. Varanasi, K. Tamaddon, A. Héloir, and D. Stricker. Deephps: End-to-end estimation of 3d hand pose and shape by learning from synthetic depth. *International Conference on 3D Vision (3DV)*, 2018.

[26] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt. GANerated Hands for Real-Time 3D Hand Tracking from Monocular RGB. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[27] F. Mueller, M. Davis, F. Bernard, O. Sotnychenko, M. Verschoor, M. A. Otaduy, D. Casas, and C. Theobalt. Real-time pose and shape reconstruction of two interacting hands with a single depth camera. *ACM Transactions on Graphics (TOG)*, 38(4), 2019.

[28] M. Oberweger, P. Wohlhart, and V. Lepetit. Training a feedback loop for hand pose estimation. In *International Conference on Computer Vision (ICCV)*, 2015.

[29] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *British Machine Vision Conference (BMVC)*, 2011.

[30] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.

[31] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3d face model for pose and illumination invariant face recognition. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 296–301. Ieee, 2009.

[32] F. Prada, M. Kazhdan, M. Chuang, and H. Hoppe. Gradient-domain processing within a texture atlas. *ACM Transactions on Graphics (TOG)*, 37(4), 2018.

[33] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun. Realtime and robust hand tracking from depth. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

[34] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari. Pytorch3d. `https://github.com/facebookresearch/pytorch3d`, 2020.

[35] E. Remelli, A. Tkach, A. Tagliasacchi, and M. Pauly. Low-dimensionality calibration through local anisotropic scaling for robust hand model personalization. In *International Conference on Computer Vision (ICCV)*, 2017.

[36] S. Romdhani and T. Vetter. Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 986–993. IEEE, 2005.

[37] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017.

[38] J. L. Schönberger and J.-M. Frahm. Structure-from-Motion Revisited. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[39] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, et al. Accurate, robust, and flexible real-time hand tracking. In *ACM Conference on Human Factors in Computing Systems (CHI)*, 2015.

[40] M. A. Shiffman, S. Mirrafati, S. M. Lam, and C. G. Cueteaux. *Simplified facial rejuvenation*. Springer Science & Business Media, 2007.

[41] T. Simon, H. Joo, I. Matthews, and Y. Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[42] Sony Corporation. 3D Creator App (White Paper). `https://dyshcs8wkvd5y.cloudfront.net/docs/3D-Creator-Whitepaper.pdf`, 2018. Version 3: August 2018.

[43] SONY3DCreator. `https://3d-creator.sonymobile.com/`.

[44] A. Spurr, J. Song, S. Park, and O. Hilliges. Cross-modal deep variational hand pose estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[45] S. Sridhar, A. Oulasvirta, and C. Theobalt. Interactive markerless articulated hand motion tracking using rgb and depth data. In *International Conference on Computer Vision (ICCV)*, 2013.

[46] J. Taylor, V. Tankovich, D. Tang, C. Keskin, D. Kim, P. Davidson, A. Kowdle, and S. Izadi. Articulated distance fields for ultra-fast tracking of hands interacting. *ACM Transactions on Graphics (TOG)*, 36(6), 2017.

[47] A. Tewari, M. Zollhöfer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *International Conference on Computer Vision (ICCV)*, 2017.

[48] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[49] A. Tkach, M. Pauly, and A. Tagliasacchi. Sphere-meshes for real-time hand modeling and tracking. *ACM Transactions on Graphics (ToG)*, 35(6), 2016.

[50] A. Tkach, A. Tagliasacchi, E. Remelli, M. Pauly, and A. Fitzgibbon. Online generative model personalization for hand tracking. *ACM Transactions on Graphics (ToG)*, 36(6), 2017.

[51] J. Tompson, M. Stein, Y. Lecun, and K. Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics*, 33, 2014.

[52] C. Wan, T. Probst, L. Van Gool, and A. Yao. Crossing nets: Combining gans and vaes with a shared latent space for hand pose estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[53] L. Yang, S. Li, D. Lee, and A. Yao. Aligning latent spaces for 3d hand pose estimation. In *International Conference on Computer Vision (ICCV)*, 2019.

[54] X. Zhang, Q. Li, H. Mo, W. Zhang, and W. Zheng. End-to-end hand mesh recovery from a monocular rgb image. In *International Conference on Computer Vision (ICCV)*, 2019.

[55] C. Zimmermann and T. Brox. Learning to Estimate 3D Hand Pose from Single RGB Images. In *International Conference on Computer Vision (ICCV)*, 2017.

[56] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. Argus, and T. Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *International Conference on Computer Vision (ICCV)*, 2019.

Below you find a list of the most recent research reports of the Max Planck Institut for Informatics. Most of them are accessible via WWW using the URL `http://www.mpi-inf.mpg.de/publications/research-reports/`. Paper copies (which are not necessarily free of charge) can be ordered either by regular mail or by e-mail at the address below.

Max Planck Institut für Informatik
– Library and Publications –
Campus E 1 4
66123 Saarbrücken
Germany

E-mail: `library@mpi-inf.mpg.de`

MPI-I-2016-5-002	A. Mishra, K. Berberich	Leveraging Semantic Annotations to Link Wikipedia and News Archives
MPI-I-2016-5-001	D. Gupta, K. Berberich	Diversifying Search Results Using Time
MPI-I-2016-4-002	S. Sridhar, G. Bailly, E. Heydrich, A. Oulasvirta, C. Theobalt	FullHand: Markerless Skeleton-based Tracking for Free-Hand Interaction
MPI-I-2016-4-001	S. Sridhar, F. Müller, M. Zollhöfer, D. Casas, A. Oulasvirta, C. Theobalt	Real-time Joint Tracking of a Hand Manipulating an Object from RGB-D Input
MPI-I-2014-RG1-002	P. Baumgartner, U. Waldmann	Hierarchic Superposition with Weak Abstraction
MPI-I-2014-5-002	A. Anand, I. Mele, S. Bedathur, K. Berberich	Phrase Query Optimization on Inverted Indexes
MPI-I-2014-5-001	M. Dylla, M. Theobald	Learning Tuple Probabilities in Probabilistic Databases
MPI-I-2014-4-002	S. Sridhar, A. Oulasvirta, C. Theobalt	Fast Tracking of Hand and Finger Articulations Using a Single Depth Camera
MPI-I-2014-4-001	K. Kim, J. Tompkin, C. Theobalt	Local High-order Regularization on Data Manifolds
MPI-I-2013-5-002	F. Makari, B. Awerbuch, R. Gemulla, R. Khandekar, J. Mestre, M. Sozio	A Distributed Algorithm for Large-scale Generalized Matching
MPI-I-2013-1-001	C. Huang, S. Ott	New Results for Non-preemptive Speed Scaling
MPI-I-2012-RG1-002	A. Fietzke, E. Kruglov, C. Weidenbach	Automatic Generation of Invariants for Circular Derivations in SUP(LA) 1
MPI-I-2012-RG1-001	M. Suda, C. Weidenbach	Labelled Superposition for PLTL
MPI-I-2012-5-004	F. Alvanaki, S. Michel, A. Stupar	Building and Maintaining Halls of Fame Over a Database
MPI-I-2012-5-003	K. Berberich, S. Bedathur	Computing n–Gram Statistics in MapReduce
MPI-I-2012-5-002	M. Dylla, I. Miliaraki, M. Theobald	Top-k Query Processing in Probabilistic Databases with Non-materialized Views
MPI-I-2012-5-001	P. Miettinen, J. Vreeken	MDL4BMF: Minimum Description Length for Boolean Matrix Factorization
MPI-I-2012-4-001	J. Kerber, M. Wand, M. Bokeloh, H. Seidel	Symmetry Detection in Large Scale City Scans