Imperial College London
Department of Computing

Advancing 3D Hand Reconstruction and Modelling with Geometric Deep Learning

Dominik Kulon

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing and the Diploma of Imperial College London
November 2020

Copyright Declaration

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-NonCommercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Statement of Originality

I acknowledge that I am the sole author of this thesis, and, except where otherwise indicated, it describes my own research.

Dominik Kulon

London, 23 October 2020

Abstract

We introduce substantial advancements to the field of hand reconstruction and modelling. We observe that the current state-of-the-art methods for hand pose estimation do not generalize well to samples captured in a non-laboratory environment and cannot provide real-time mobile inference. We also identify issues with the existing deformable models of the human hand that have limited shape variability and cannot express texture variations. We effectively address these problems from multiple directions. The first contribution is a weakly-supervised mesh-convolutional neural network capable of reconstructing a 3D hand mesh from a single image in the wild. It outcompetes the existing systems while working in real-time on mobile devices. We make this possible by proposing an automated data collection system and novel framework for non-linear latent representation of 3D hand meshes and image alignment. We also lead a data collection effort to generate high-resolution 3D hand scans with texture on a previously unseen scale in terms of subjects and pose diversity. Our dataset allows us to build the most diverse deformable hand model with shape and texture variations capable of representing children, adults, and the elderly from a broad range of ethnic groups. We show that our systems obtain state-of-the-art performance and enable novel industrial applications.

Acknowledgements

I am grateful to my supervisors, Michael Bronstein and Stefanos Zafeiriou, for their mentorship and support without which my work would have not been possible. I find myself very lucky to have collaborated with advisors who gave me lots of freedom while motivating and inspiring my work. Large organizational effort that Stefanos put to run and fund the projects allowed for any of the described research to take place. Moreover, he has always stayed at the forefront of research suggesting novel applications. I would be ignorant to forget that he directed me towards hand reconstruction research three years ago despite the lack of relevant publications in the field. It turned out to define and progress my career. A scientific community and the environment that Michael has built around him has always been an immense source of inspiration.

My special gratitude goes to Rıza Alp Güler whose research expertise hugely influenced my work and made a difference in my research quality that cannot be easily quantified.

I have been fortunate to work with Iasonas Kokkinos and I want to thank him for constant feedback and advice.

Since I started research at Imperial I had the opportunity to collaborate with many brilliant people: Haoyang Wang, Shunwang Gong, Stylianos Ploumpis, and Yuxiang Zhou. I would also like to express my appreciation to the Ariel AI team with whom I have had a great pleasure to develop cutting-edge augmented reality applications. This team, besides already mentioned people, includes: George Papandreou, Brian Fulkerson, Petros Koutras, Himmy Tam, Efstratios Skordos, Dan Stoddart, Andreas Lazarou, Antonis Kakolyris, Stathis Galanakis, and Mohammad Rami Koujan. I also want to thank Haoyu Lan for all the support during and after my stay in Los Angeles.

Especially, I thank my partner Jasper Yuliat - for everything.

Dedication

To our future selves transformed by the digital technology.

‘The point is not to construct synthetic humanity
but rather to open up a new chapter in the Book of Technology:
one containing systems of any degree of complexity.’

Stanisław Lem. Summa Technologiae.

Contents

Copyright Declaration i

Statement of Originality iii

Abstract iii

Acknowledgements vii

1 Introduction 1

1.1 Contributions 2

1.1.1 3D Hand Reconstruction 2

1.1.2 Hand Modelling 4

1.2 Publications 5

2 Background on Hand Reconstruction 7

2.1 3D Hand Pose Estimation 8

2.2 Hand Mesh Recovery 10

2.3 Body Mesh Recovery 11

2.4 Hand Pose Datasets 13

3 Background on Human Body Part Modelling **19**

3.1 Shape Models 20

3.2 SMPL 21

3.2.1 Linear Blend Skinning 23

3.2.2 Joint Locations 26

3.2.3 Pose-Dependent Deformations 26

3.2.4 Summary 27

3.3 Hand Models 28

3.4 Unified Models 30

3.5 Geometric Deep Learning 31

3.5.1 Graph Laplacian 31

3.5.2 Graph Fourier Transform 32

3.5.3 Graph Filtering 33

3.5.4 Spectral CNN 34

3.5.5 Fast Spectral CNN 34

4 Iterative Model Fitting **35**

4.1 YouTube 3D Hands Dataset 37

4.1.1 3D Shape Representation 38

4.1.2 Parametric Model Fitting 39

4.1.3 Optimization 41

4.1.4 Automated Data Collection 41

4.2 Conclusion 43

5 Mesh Autoencoder **45**

5.1 Model Implementation 46

5.2 Data Acquisition 47

5.3 Evaluation 49

5.4 Conclusion 50

6 3D Hand Reconstruction in the Wild **51**

6.1 Graph Morphable Model 53

6.1.1 Evaluation 55

6.2 Direct Regression 57

6.2.1 Spiral Operator 57

6.2.2 Architecture 59

6.2.3 Training 59

6.2.4 Evaluation 60

6.3 Conclusion 68

7 Large-Scale Hand Scans Collection **69**

7.1 Hand Reconstruction System 72

7.2 Data Collection Protocol 72

7.3 Data Quality 76

7.4 Conclusion 77

8 Statistical Deformable Hand Model **79**

8.1 Bone-Level Skinned Model of the Human Hand (Hand BLSM) 81

8.1.1 Skeleton Transformations 82

8.1.2 Modelling Bone Length Variations 84

8.1.3 Modelling Shape Variations 84

8.1.4 Posing 85

8.2 Training 85

8.2.1 Bone Basis and Bone-Corrective Blendshapes 86

8.2.2 Shape Blendshapes 86

8.3 Registration 87

8.3.1 Pose Estimation 87

8.3.2 Fitting 89

8.4 Evaluation 91

8.5 Conclusion 92

9 Texture Model **93**

9.1 Texture Mapping 94

9.2 Texture Model 95

9.3 Evaluation 97

9.4 Conclusion 98

10 Conclusion **101**

Bibliography **103**

List of Tables

2.1 Characteristics of the datasets for hand pose estimation. Our YouTube 3D Hands is the only in the wild dataset with 3D annotations. Our MeDigital dataset contains highly accurate mesh annotations of 1,360 subjects of a wide variety of ethnic background and age performing complex motions. 16

3.1 A summary of SMPL model parameters. 27

3.2 A summary of SMPL model functions. 27

6.1 GMM system performance. Mesh L1 reconstruction error, inference time, and number of parameters for systems with different types of generators evaluated on Panoptic DomeDb. The scale of the target meshes is smaller than in real world. 55

6.2 3D pose estimation and mesh reconstruction performance on `FreiHAND`. For the first two rows lower is better, for the other rows higher is better. 61

6.3 Pose Error. Performance with different decoders trained on a reduced dataset. 65

6.4 Mesh reconstruction with different decoders on the `YouTube` dataset trained on a reduced dataset. 65

6.5 Comparison of the network performance with iterative fitting on the `MPII` test set in terms of Pose Error. 65

6.6 Pose Error. Performance on different combinations of datasets. 67

7.1 Data comparison. Additional unique features of our dataset (MeDigital Hands) include: presence of jewellery, medical skin and bones conditions, continuous motions instead of discrete gestures. 76

8.1 3D Pose Error. Performance with models trained on datasets with different hand fittings. 91

List of Figures

2.1 A hand model with annotations of twenty-one keypoints. 9

3.1 Early body models by Dariu Gavrilla. 20

3.2 Registrations of the body scans from the CAESAR dataset. Reproduced from [ACCP03]. 22

3.3 SMPL body model. Reproduced from [LMR+15]. 22

3.4 Hand models proposed in publications. 28

4.1 Minimizing the 2D reprojection error leads to anatomically invalid poses in the absence of prior on joint rotation angles. 36

4.2 Iterative model fitting. 37

4.3 Error terms for the parametric model fitting and pose prior explanation. 39

4.4 Samples from the YouTube dataset. 42

5.1 Mesh autoencoder learns a low-dimensional latent representation of 3D hand shapes. 46

5.2 a) Random shapes sampled from our high resolution linear model. Posed examples can be found in Figure 6.2. b) Samples from the MANO model (taken from [RTB17]). 48

5.3 Autoencoder reconstruction results for challenging samples in terms of the pose (left pair) and shape (right pair). 48

5.4 Interpolation in the latent space between two random samples. Please, note that source and target shapes (e.g. finger length) are also different. 48

6.1 Graph Morphable Model regression baseline. The presented system generates a 3D hand model corresponding to the hand in the image. The system consists of two branches trained simultaneously. The network in the top branch is capable of generating 3D hand models from a single image in a wide variety of shapes and poses. The ground truth meshes are aligned to the mean hand model in a canonical frame. The network in the lower branch regresses parameters of a weak perspective projection to align the generated mesh with the image. The gradient from the keypoints loss does not flow through the top branch to prevent the camera regressor from affecting the quality of mesh reconstructions. 53

6.2 Qualitative results of the GMM system. 56

6.3 We propose an approach for end-to-end neural network training with mesh supervision that is obtained through an automated data collection method. We process a large collection of YouTube videos and analyze them with 2D hand keypoint detector followed by parametric model fitting (right side). The fitting results are used as a supervisory signal ('mesh loss') for a feed-forward network with a mesh convolutional decoder tasked with recovering a 3D hand mesh at its output (left side). 57

6.4 a) Spiral selection of neighbours for a center vertex in clockwise order (taken from [BBP+19]). b) The hierarchy of topologies used in our decoder. 58

6.5 Architecture of the spiral decoder. $C(w, l)$: A spiral convolution with w filters and a spiral length l; $U(p)$: upsampling by a factor p; $FC(d)$: a fully connected layer with d output dimension; $R(d_1, d_2)$: vector reshaping into a $d_1 \times d_2$ matrix. 59

6.6 2D PCK and average 2D joint distance (px) to ground-truth for `MPII` 62

6.7 3D PCK and average 3D joint distance (mm) to ground-truth for `RHD`. 62

6.8 Qualitative mesh reconstruction results shown on `YouTube` (left half) and `MPII` (right half). 63

6.9 Qualitative performance comparison with state-of-the-art for in the wild hand reconstruction. Our system is two times more accurate; the difference is especially visible on blurry images, hands wearing gloves, saturated images, and hands interacting with objects. 64

6.10 Our system, denoted as Ariel, is both faster and more accurate than existing approaches. 66

6.11 The system shows impressive results on egocentric images enabling potential applications for mobile and wearable devices. 67

6.12 A London-based startup Ariel AI developed augmented reality mobile applications that incorporate the proposed system. On the left, the thesis author records himself on an entry level iPhone (SE) running a neural network capable of processing images at 250 FPS. On the right, a user tests the application for hand reconstruction that allows applying different visual filters and drawing with an index finger. . . 68

7.1 Data collection project in Science Museum in London. 70

7.2 Hand scanner. 71

7.3 Participant experience. 72

7.4 Camera views and renders. 73

7.5 Consent form. 74

7.6 Selection of gestures from hand scanning protocols. 75

7.7 Age and ethnic diversity of participants. 76

7.8 Texture details. 77

7.9 Hand renders. 78

8.1 Overview of the Bone-Level Skinned Model (BLSM) reproduced from [WGK+20]. The top branch shows skeleton transformations while the lower branch describes the process of the template synthesis. Bone scale parameters are shared between branches. Both streams are combined to pose the updated template with linear blend skinning (LBS). 81

8.2 Samples of bone-dependent shape variations that model surface deformations due to bone lengths. 83

8.3 Samples of subject-specific shape variations that model surface deformations for fixed bone lengths. 83

8.4 Rest pose registrations. 88

8.5 Minimum and maximum hand shapes modelled by the first component of bone-dependent shape variations. Our hand model is capable of representing children and adults. 88

9.1 UV maps for the body and hand templates. 95

9.2 Examples of transferred textures. 96

9.3 Random samples from the texture model visualized on the initial template. 96

9.4 Evaluation of the texture model. 97

9.5 Pairs of the noisy test samples (top row) and their reconstructions (bottom row). 99

Chapter 1

Introduction

Mobile devices have made augmented reality (AR) applications widely accessible. Most recent smartphones are capable of running neural networks that a few years ago required professional workstations and they feature LiDAR technology that allows mapping the user's environment. The progress has been largely driven by entertainment and retail applications. Among FAAMG companies dominating the American market by their capitalization, all of them are actively involved in research on augmented reality technology. Snap reports that 170 million of Snapchat users engage in trying various visual filters everyday that allow, among many use cases, transforming in real-time images of the human body and the environment as well as overlaying digital objects onto the physical world. Health care is also benefiting from the related developments. CAE has been creating virtual environments for training neurosurgeons and pre-operation practice. Doctors at Imperial College Healthcare have been experimenting with mixed reality to support surgeons in operating theatre during treatment or to care for patients in coronavirus wards.

The interaction with digital objects rendered on a mobile phone is performed via a touch screen. Head-mounted devices such as Oculus or HoloLens headsets enable more intuitive control by estimating and tracking locations of hand joints because

of the availability of stereo depth cameras or specialized sensors. Even then, the accuracy and range of recognizable poses is limited and the hand shape is not personalized. The performance is critically lower on smartphones which cannot perform inference on large neural networks and usually process single RGB images as input. In fact, until recently it has not been possible to perform robust hand pose estimation in non-laboratory environments, not even on specialized GPUs. The change came with a paper that is part of this thesis.

Our motivation stems from the assumption that AR and related technology (virtual/mixed reality) will progress and become more popular causing a growing interest in seamless human-computer interaction, motion tracking, virtual telepresence, and immersive gaming. We want to advance hand reconstruction technology to the level that it is highly accurate, fast on mobile devices, and capable of modelling wide range of shapes including personalization to a user's hand.

1.1 Contributions

In this thesis, we propose novel approaches for hand reconstruction from a single image and statistical deformable modelling of the human hand with texture. We also introduce large-scale datasets for both tasks. The proposed systems are showed to generalize well to challenging samples and have been deployed for industrial applications.

1.1.1 3D Hand Reconstruction

The first challenge lies in developing systems that can predict a 3D hand mesh given an image of a cropped hand region and find a correspondence between 3D points and image pixels. Usually, this task has been approached by predicting 2D landmarks

and lifting them into a 3D pose or finding parameters of the hand model. The existing systems perform well on laboratory benchmarks but cannot generalize to the distribution of images captured in unconstrained environments. They often predict only a sparse set of keypoints instead of a dense point cloud and usually they are not capable of operating in real-time. We discuss this further in **Chapter 2** that presents background on hand reconstruction including the existing methods for 3D hand joint estimation, hand mesh reconstruction, similarities with research on the full body mesh reconstruction, and available datasets. As many of the hand reconstruction methods rely on predicting parameters of statistical deformable hand models, we explain how to synthesize a 3D mesh in **Chapter 3** that includes background on body part modelling and discusses approaches for learning the shape representation of 3D meshes.

We notice that there are no datasets that we could use for training 3D hand reconstruction systems that work in non-laboratory environments. Therefore, in **Chapter 4** we develop an automated approach to generate training data from unannotated images for 3D hand reconstruction and pose estimation. The method is applied to generate the first dataset of 3D hand meshes aligned with in the wild images comprising hundreds of subjects performing wide variety of gestures.

Alternatively to the existing methods, we explore whether it is possible to use graph convolutional neural networks to learn statistical deformable models of the human hand. We achieve this in **Chapter 5** by training a mesh autoencoder end-to-end from a collection of meshes in an unsupervised way and we show gains in terms of reconstruction accuracy and speed compared to the existing methods.

Motivated by the quality of the generated dataset and the positive outcome of the experiments on mesh autoencoding, in **Chapter 6** we propose a novel framework for hand reconstruction based on the idea of encoding images into a latent non-linear representation of 3D hand shapes. The proposed system largely outperforms existing

approaches for hand pose estimation and mesh reconstruction from a single image. Notably, it attains 50% lower pose estimation error compared to the currently best model on the in-the-wild scenario. Moreover, it supports fast inference at 250 FPS on mobile devices.

1.1.2 Hand Modelling

We have been using hand models to generate training data for mesh reconstruction systems by fitting the model into 2D landmarks. Therefore, the ability of the network to represent various subject-specific shapes and articulations is bounded by the expressiveness of a deformable hand model. Moreover, shapes predicted by the network are solely correlated with hand joint locations due to the data collection strategy and thus are not pixel-aligned with the hand silhouette. By collecting a large-scale dataset of 3D hand scans we could potentially improve accuracy of the hand reconstruction system by building a model that is able to represent a more diverse set of population and by using 3D scans as a source of dense supervision which could lead to a better alignment. Unfortunately, the existing statistical deformable models of the human hand cannot represent diverse shapes in terms of age and ethnicity. This is not surprising given that they are learned from a small number of subjects and scans. The existing models also do not include texture variations that beside computer graphics applications could be incorporated into a self-supervised hand reconstruction system based on a photometric loss.

Motivated by the need to improve expressiveness of deformable hand models, we lead a large-scale data collection effort to generate high-resolution 3D hand scans that is presented in **Chapter 7**. We produce 533,687 scans captured from 1,360 subjects effectively obtaining the most diverse, by two orders of magnitude, collection of hand shapes in terms of age and ethnicity with high-resolution textures. Moreover, the scans were generated from images captured from multiple cameras with known

parameters resulting in a collection of 2,668,435 images with 3D mesh annotations.

To enable statistical modelling of shape and texture variations, we process the collected hand scans to a common representation. In **Chapter 8**, we build the most diverse deformable hand model learned from our dataset of scans. We are the first to decouple bone-length and identity-specific shape variations for hand modelling improving parameterization for augmented reality applications. The model is fully integrated with standard graphics packages. We also present a statistical texture model capable of representing children, adults, and the elderly from a broad range of ethnic groups with an unprecedented quality in **Chapter 9**.

1.2 Publications

Part of the thesis is based on the following publications:

- Single Image 3D Hand Reconstruction with Mesh Convolutions. Dominik Kulon, Haoyang Wang, Riza Alp Güler, Michael Bronstein, Stefanos Zafeiriou. **British Machine Vision Conference (BMVC)**. 2019.

- Weakly-Supervised Mesh-Convolutional Hand Reconstruction in the Wild. Dominik Kulon, Riza Alp Güler, Iasonas Kokkinos, Michael Bronstein, Stefanos Zafeiriou. **CVPR (Oral, Best Paper Award Nomination)**. 2020.

  This work was also presented in the Fourth Workshop on Computer Vision for AR/VR (CV4ARVR).

Contributions regarding the dataset of 3D hand scans and statistical deformable modelling of the human hand and texture described in Chapters 7, 8, 9 have not yet been under review due to our interest in industrial applications.

Chapter 2

Background on Hand Reconstruction

Hand pose estimation is a task of estimating locations of a sparse set of keypoints usually corresponding to fingertips and joints in the hand skeleton tree (Figure 2.1). Hand reconstruction is a more general task that aims at estimating a relatively dense point cloud of the hand in the image that describes both the pose and subject-specific shape of the target object. We categorize a method as belonging to the latter category if it allows generation of different subject-specific shapes.

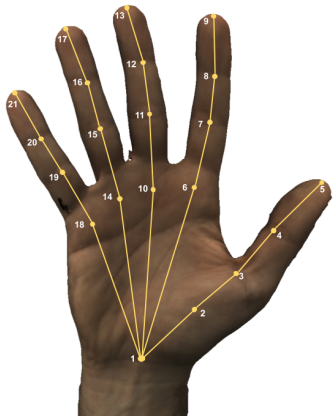
Hand reconstruction technology can facilitate a broad array of applications in human-computer interaction, augmented reality, virtual telepresence, digital characters animation, motion tracking, analyzing interactions between multiple individuals, or sign language recognition. We focus our attention on the hand reconstruction task using only RGB images as input. This is significantly more challenging than estimation when a depth sensor is also available [YGS+18, SRY+15, GLYT16, OL17, WSK15]. The use of RGB input instead of, for example, a depth map allows us to perform the analysis on large sources of visual data such as YouTube and develop systems that work with video cameras enabling low-cost industrial applications for virtual or augmented reality.

Hand reconstruction is an extremely complex problem due to variations of possible hand shapes, articulations, environmental conditions, data capture devices, depth ambiguity, and difficulties with annotating data. A hand has twenty-seven degrees of freedom and therefore it is costly to generate the training data that sufficiently covers possible configurations. Moreover, fingers are similar to each other, self-occlusions of joints are common, and occlusions can be caused by grasped objects. Input data can also have low resolution, severe distortions, motion blur, and be captured from various perspective (e.g. egocentric view from smartglasses, close view from selfie cameras). The position of the hand with respect to the camera is unknown preventing robust scale approximation. Data annotation poses another difficulties because we need to obtain 3D annotations for a 2D image which is an open research problem. Given an image captured from a single view with no prior knowledge on the space and no temporal information, the problem becomes ill-posed as multiple 3D articulations could be projected to match 2D annotations.

2.1 3D Hand Pose Estimation

Research in the area of hand pose estimation from RGB input has started gaining popularity only recently. Multiview bootstrapping [SJMS17] is an iterative improvement technique where the initial annotations of a hand from multiple views, obtained by executing a 2D keypoint detector, are triangulated in 3D and reprojected for training in the next iteration if they meet the quality criteria. This system is part of the OpenPose framework [WRKS16] for 2D pose annotation.

Numerous approaches have been proposed to regress a 3D pose from estimated 2D keypoints [ZB17, POA18, LDS+19, CGCY18]. Dibra et al. [DMB+18] incorporate depth regularization to supervise 3D pose estimation represented by angles of rotation. In contrast, variational approaches learn a shared latent space by an alternated



1	wrist	12	middle dip
2	thumb metacarpophalangeal joint (mcp)	13	middle tip
3	thumb proximal interphalangeal joint (pip)	14	ring mcp
4	thumb distal interphalangeal joint (dip)	15	ring pip
5	thumb tip	16	ring dip
6	index mcp	17	ring tip
7	index pip	18	little mcp
8	index dip	19	little pip
9	index tip	20	little dip
10	middle mcp	21	little tip
11	middle pip		

Figure 2.1: A hand model with annotations of twenty-one joints. A kinematic structure is shown as links connecting the joints.

training between variational autoencoders [SSPH18], disentanglement of the latent representation [YY19], or the alignment of latent spaces [YLLY19]. This allows for sampling outputs in any of the learned modalities. Gao et al. [GWF+19] address the problem of estimating the pose of hands interacting with objects in a variational framework. Generative networks can also be used to transfer synthetic images into a space of real-world images [MBS+18] by modifying a cycle adversarial loss [ZPIE17] to include a geometric consistency term that ensures pose-preservation during an image translation.

Tekin et al. [TBP19] estimate hand and object poses with a single-shot network. Baek et al. [BKK20] address the problem of estimating poses of hands interacting with objects by generating images of hands without the objects as an intermediate step. Cai et al. [CGL+19] proposed to build a spatio-temporal graph and use graph convolutions in the spectral domain to estimate a 3D pose.

Recently, Zhang et al. [ZCL+20] incorporated a teacher-student framework to supervise a compact network for 3D hand pose estimation. Wang and Chen [WC20] attempt to distill knowledge from the depth-based pose estimation network to a network that takes as input an RGB image. Moon et al. [MYW+20] estimate 3D poses of interacting hands. The DOPE [WBC+20] method combines body, face, and hand

experts to distill the training data for a whole body network. MEgATrack method [HLC+20] processes four monochrome images captured with cameras mounted on a virtual reality headset to estimate a 3D pose. Gesture recognition network can be combined with a pose estimation network to boost the performance of each task [YLL+20]. Fan et al. [FLW20] propose a framework to switch between lightweight and heavyweight models for computational efficiency.

2.2 Hand Mesh Recovery

MANO [RTB17] is a hand model parameterized by angles of rotations specified for each joint in the kinematic tree and blend weights of a linear function that models the shape of the person. Different articulations are obtained by applying linear blend skinning which interpolates rotations matrices assigned to the joints to transform a vertex according to the angles. Additionally, pose-dependent corrective offsets are learned to address the loss of volume caused by the skinning method. Recently, numerous works have been proposed to find a 3D pose by regressing MANO and camera parameters.

Boukhayma et al. [BBT19] regress these parameters from an image and heatmaps obtained from OpenPose. The method is evaluated on images in the wild but the error is relatively high. The works by Zhang et al. [ZLM+19] and Baek et al. [BKK19] introduce methods that iteratively regress model parameters from the heatmaps. Hasson et al. [HVT+19a] predict MANO parameters and reconstruct objects the hands interact with.

Ge et al. [GRL+19] use graph convolutions in the spectral domain to recover a hand mesh. The system includes a heatmap regression module and, in case of real-world images, is supervised with 2D keypoints, depth maps, and meshes approximated with a pretrained model from ground truth heat maps.

Yang et al. [YCLK20] build a recurrent network trained on synthetically generated sequences of hand motions and then fine-tune it to a single-frame real-world images. Moon and Lee [ML20] obtain excellent results on a laboratory hand reconstruction benchmark by predicting 1D heatmaps of each vertex in the output mesh. Yang et al. [YLX+20] regress MANO parameters in a multi-stage pipeline with intermediate supervision that includes silhouette, depthmap, 2D heatmaps, and 3D joints error terms.

2.3 Body Mesh Recovery

Approaches based on regressing MANO parameters described in the previous subsection originate from works on body mesh recovery. Numerous methods have been proposed to recover a subject-specific body mesh from an RGB image by finding parameters of the SMPL model. In SMPLify, Bogo et al. [BKL+16] estimate the 2D body joint locations and use these estimates to iteratively fit the SMPL model. The space of plausible shapes is constrained by adding prior and interpenetration error terms to the objective function. The shape parameters are only derived from the joint locations and thus the shape of the predicted model is close to the average.

The issue of shape estimation inaccuracy is addressed in other works by introducing a silhouette error term. Lassner et al. [LRK+17] extend the SMPLify method by penalizing the bi-directional distance from points of the projected model to the ground-truth silhouette. This approach provides a better estimate of the body shape than its predecessor but it again requires numerically solving a complex optimization problem.

The Human Mesh Recovery framework [KBJM18] reconstructs the body mesh in an end-to-end manner by regressing the model parameters and minimizing the reprojection loss of the joint locations given by the SMPL model and the ground-truth 2D

image annotations. The authors introduce an adversary to discriminate body parameters that constrains the pose space. Tan et al. [TBC17] train a decoder from the SMPL parameters to a body silhouette and afterwards they train an encoder-decoder network that outputs a silhouette given an RGB image. Pavlakos et al. [PZZD18] train a neural network that consists of modules responsible for predicting the mask, landmarks, and model parameters. Varol et al. [VCR+18] combine segmentation, 2D pose, and 3D pose predictions to infer the volumetric representation of the body into which they fit the SMPL model. Neural Body Fitting [OLPM+18] is an end-to-end system that integrates the SMPL model within a CNN. The network is a sequence of layers responsible for computing the segmentation, model parameters, and joint positions. Finally, the joints are projected onto the image to compute the loss. Weng at al. [WCKS19] show that it is possible to create an animated character from a single photo.

All models described above have none or very limited variance of hand poses and face expressions. The authors of the Total Capture system [JSS18] build a unified body model that is iteratively fit into the point cloud obtained from a multi-camera setup. The system produces impressive results but is limited to the laboratory setting. The follow-up work [XJS19] enables predicting the model parameters in the wild by fitting the model into predicted joint confidence and orientation maps. SMPLify-X [PCG+19] improves on SMPLify by additionally fitting articulated hands and face expressions, using a VAE pose prior, and changing the collision penalizer. SPIN [KPBD19] combines network-based parameters regression for robust initialization with iterative fitting to 2D joints. The authors of ExPose [CPB+20] propose a method to regress parameters of a statistical body model by combining predictions from part specific networks that process body, hand, and face image crops.

Alternatively to methods based on regression of body model parameters, Kolotouros et al [KPD19] propose a system to encode an image and concatenate its features

with coordinates of a template mesh in a canonical pose. The resulting graph is passed through a sequence of spectral graph convolutions to recover the body model corresponding to the image.

2.4 Hand Pose Datasets

Deep learning research is largely driven by the availability of large scale datasets. Therefore, it is of crucial importance to discuss the existing resources. The summary of collected information can be found in Table 2.1. We also take depth image datasets into consideration because they can be used as an additional source of supervision for RGB-based neural networks.

Small-Scale Datasets Datasets for hand pose estimation started being published relatively recently. Even then, their small size and lack of subjects diversity restricted potential applications to laboratory environments. The ASTAR [XC13] dataset that was released in 2013 contains 870 samples with the 320×240 depth map resolution captured from 30 subjects performing 29 gestures including poses from the Chinese Number Counting system and the American Sign Language alphabet. Twenty joints are annotated using the ShapeHand dataglove. The use of a dataglove does not allow obtaining natural RGB images of hands. The ICVL [TCTK14] dataset that is annotated leveraging 3D skeletal tracking followed by manual refinements contains samples from 10 subjects. Annotations in the NYU dataset [TSLP14] are obtained automatically by fitting a 3D hand model. It features higher resolution of depth maps then its predecessors (640×480). The provided RGB images are projections of the point clouds from a 3D sensor resulting in low-quality 2D renders. Authors of MSRA14 [QSW+14a] and MSRA15 dataset [SWL+15] recorded various poses using Intel's Creative Interactive Camera. Multiview Hand Pose dataset [GDOEC19] has

been annotated using a Leap Motion Controller. A relatively small dataset called SHREC 2017 [DSWV+17] contains 2,800 samples for gesture recognition tasks generated with the Intel Real Sense Depth camera.

Fingertip Annotations The Dexter dataset [SOT13] for hand motion tracking contains fingertip annotations of 2,137 samples collected from a single subject. HandNet [WSK15] includes more than 200,000 sample where fingertips are annotated using magnetic sensors. The Dexter+Object [SMZ+16] dataset contains manual annotations of hands interacting with a cuboid-shaped object and positions of the cuboid corners. EgoDexter [MMS+17] explores object interactions from an egocentric view.

BigHand2.2M The BigHand2.2M benchmark [YYS+17] with more than 2.2 million samples covers wide range of hand poses performed by 10 subjects. The annotations of 21 joints were made automatically by attaching six 6D magnetic sensors on the hand and applying inverse kinematics on a hand model. 290,000 of the samples are captured from the egocentric perspective.

Egocentric Real Data Egocentric perspective is common for images captured by mobile or wearable devices. Robust hand pose estimation in such scenarios could allow analysing manual tasks and enable novel ways of interacting with objects in augmented and virtual reality. UCI-EGO [RKSI+14] consists of four motion sequences of object interactions such as typing, grasping, and opening a door. The dataset of first-person hand action sequences [GHYBK18] includes 45 categories manipulating 25 different objects. Apart from hand pose annotations, authors provide 6-dimensional object pose ground truth with their 3D mesh models for 4 objects.

Synthetic Data The Rendered Hand Pose Dataset (RHD) consists of 41,258 training and 2,728 testing samples of rendered characters [ZB17]. It has been com-

monly used for RGB-based hand pose estimation due to its large size and challenging viewpoints. SyntheticHand [MEC+17], contains samples generated from a rigged 3D model of a human adult male hand. SynthHands [MMS+17] is another large synthetic dataset for egocentric hand pose research with depth and colour images, 3D annotations, and camera calibration parameters. GANerated Hands Dataset [MBS+18] is a large synthetic dataset translated into the space of real images via a cycle-consistent generative adversarial network. The images are captured from the egocentric view and have a small resolution of 256×256 pixels. A hand model that incorporates a shape displacement function is used to produce the SynHand5M [MEN+18] dataset.

RGB Laboratory Benchmarks Stereo Hand Pose Tracking Benchmark (STB) contains 15,000 training images of a single subject performing random poses and counting gestures with 5 different backgrounds and a test set of 3,000 images with the same background [ZJC+17]. The dataset has been solved in recent publications [GRL+19] with a high error tolerance due to inaccurate ground truth annotations. Panoptic DomeDb [SJMS17] provides 3D annotations of the whole human body including 21 joints of hands and face landmarks. It contains multiple sequences of videos captured from around 31 synchronized HD cameras and 479 VGA cameras which parameters are provided. Monocular Total Capture (MTC) is a human motion dataset containing body, hands, and face annotations of 40 subjects [XJS19]. The dataset was collected in a laboratory environment posing a risk of domain overfitting; however, due to its diverse and mostly accurate annotations it can be used to learn a robust prior on 3D hand shapes. FreiHAND [ZCY+19] is a recently released dataset with 130,240 training images [ZCY+19]. It is the only dataset that contains 3D mesh annotations with backgrounds artificially blended in place of the green screen. The test set consisting of 3,960 samples was collected without the green screen in a controlled outdoor and office environment. It contains difficult

Title	Year	Depth resolution	RGB resolution	No. samples	No. subjects	No. joints
ASTAR [XC13]	2013	320x240	-	870	30	20
Dexter 1 [SOT13]	2013	320x240	320x240	2,137	1	5
ICVL [TCTK14]	2014	320x240	-	17,604	10	16
MSRA14 [QSW+14a]	2014	320x240	-	2,400	6	21
NYU [TSLP14]	2014	640x480	?	81,009	2	36
UCI-EGO [RKSI+14]	2014	320x240	640x480	400	2	26
HandNet [WSK15]	2015	320x240	-	212,928	10	6
MSRA15 [SWL+15]	2015	320x240	-	76,375	9	21
Dexter+Object [SMZ+16]	2016	320x240	640x480	3,145	2	5
STB [ZJC+17]	2016	640x480	640x480	18,000	1	21
Tzionas et al. [TBS+16a]	2016	640x480	640x480	36,000	?	?
BigHand2.2M [YYS+17]	2017	640x480	-	∼2,200,000	10	21
EgoDexter [MMS+17]	2017	640x480	640x480	1,485	4	5
FPHA [GHYBK18]	2017	640x480	1920x1080	105,459	6	21
HandDB Rend. [SJMS17]	2017	-	mixed	14,261	synthetic	21 (2D)
MHP [GDOEC19]	2017	-	640x480	20,500	9	20
MPII+NZSL [SJMS17]	2017	-	mixed	2,758	large	21 (2D)
RHD [ZB17]	2017	320x320	320x320	43,986	synthetic	21
SHREC 2017 [DSWV+17]	2017	640x480	-	2,800	28	22
SyntheticHand [MEC+17]	2017	512x424	-	∼700,000	synthetic	20
SynthHands [MMS+17]	2017	640x480	640x480	∼220,000	synthetic	21
GANerated Hands [MBS+18]	2018	-	256x256	∼330,000	synthetic	21
HandDB Panoptic [SJMS17]	2018	-	1920x1080	14,817	multiple	21 (2D)
Monocular Total Capture [XJS19]	2018	-	1920x1080	111,000	40	21
Panoptic DomeDb [SJMS17]	2018	-	1920x1080	videos	multiple	21
SynHand5M [MBS+18]	2018	320x240	-	∼5,000,000	synthetic	mesh
Dense Hands [MDB+19]	2019	640x480	-	80,000	5	21
FreiHAND [ZCY+19]	2019	-	224x224	134,200	32	mesh
ObMan [HVT+19b]	2019	?	?	∼150,000	synthetic	mesh
OneHand10K [YWL19]	2019	-	mixed	11,703	large	21 (2D)
SynHandEgo [MENS19]	2019	320x240	-	∼1,000,000	synthetic	mesh
ContactPose [BTT+20]	2020	?	?	∼2,900,000	50	21
InterHand2.6M [MYW+20]	2020	-	512x334	∼2,600,000	27	21
YouTube 3D Hands (Ours)	2020	-	1920x1080	50,175	large	mesh
MeDigital Hands (Ours)	2020	-	2448x2048	2,668,435	1,360	mesh

Table 2.1: Characteristics of the datasets for hand pose estimation. Our YouTube 3D Hands is the only in the wild dataset with 3D annotations. Our MeDigital dataset contains highly accurate mesh annotations of 1,360 subjects of a wide variety of ethnic background and age performing complex motions.

poses with object interactions and varying lighting. The test set annotations are not available and evaluation is performed by anonymously submitting predictions to the online competition.

Hand Interactions The problem of hand pose estimation becomes more complex in the presence of hand-hand interactions due to hand shape similarity and severe occlusions. Tzionas et al. [TBS+16a] capture 29 sequences of interacting hands including object manipulation. Mueller et al. [MDB+19] propose a synthetic dataset where virtual models are posed by tracking hands of 5 subjects. InterHand2.6M [MYW+20] is the largest dataset focused on annotating interactions between two hands that was captured in a laboratory studio. During the last year, the research community focused more attention on the task of modelling hand-object interactions from RGB images because the models trained on standard hand pose estimation benchmarks do not generalize to such scenarios which might be of high importance for modelling robotic grasping. ObMan Dataset [HVT+19b] consists of 150K synthetic RGB and depth images of hands grasping 2.7K everyday objects. ContactPose [BTT+20] introduces a large dataset of over 2.9M samples that includes 2,306 grasps of 25 household objects performed by 50 participants.

In the Wild Images Despite the wealth of possible applications, there are no datasets for monocular 3D hand reconstruction from RGB images in a non-laboratory environment. The MPII+NZSL (MPII) dataset [SJMS17] contains in the wild images collected by manually annotating two publicly available datasets with 2D landmarks: the MPII Human Pose dataset [APGS14] showing every-day human activities and a set of images from the New Zealand Sign Language Exercises [MMAP]. It includes blurry, low-resolution, occluded, and interacting with objects hand images what makes the dataset particularly challenging. Training and test sets count 1,912 and 846 samples. OneHand10K [YWL19] contains 2D annotations of over 10K

in-the-wild images.

Our Datasets In the thesis, we introduce the largest and most diverse dataset for hand pose estimation and 3D hand mesh reconstruction in the wild (YouTube 3D Hands). We have also collected 533,687 high resolution scans from 1,360 subjects including images from multiple cameras (MeDigital Hands). We discuss these datasets in details in Chapters 4, 7, 8. For completeness we summarize them in Table 2.1.

Chapter 3

Background on Human Body Part Modelling

Generative body models allow synthesizing 3D meshes of the human body given input parameters. Beside obvious applications in computer animation and human body understanding, they have potential to power cutting-edge technology for augmented reality, holographic telepresence, gaming, motion capture, fashion retail, and neural image synthesis. We are specifically interested in statistical deformable modelling of hands that can cover variations of shape and texture across the whole population and constrain the space of valid poses. Face, body, and hand models are usually constructed separately due to different data acquisition procedures and level of deformations. Nevertheless, they share common techniques that we discuss in this background chapter. We also present methods for performing convolutions on meshes that enable novel approaches for representation learning of 3D shapes.

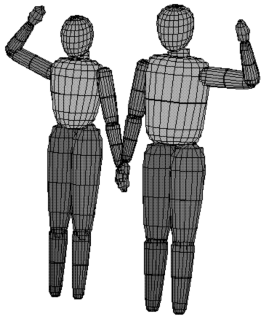


Figure 3.1: Early body models by Dariu Gavrilla [Gav98].

3.1 Shape Models

Early body models rely on geometric primitives. Dariu Gavrilla, in his PhD thesis [Gav98], introduced male and female body models approximated by coarse surface - ELLEN and DARIU (Figure 3.1) - to recover a human body from an image for the tasks of motion tracking and recognition. Subsequent works tried to better represent the articulated figures with finer topological structures but the shape space was constrained by the modelling skills of the computer graphics designer and linear blending of transforms [KM96, PF01].

The important work that enabled statistical modelling of shape of the target object came from Blanz and Vetter [BV99]. They constructed the first morphable model [BV99, BAP+17, BRZ+16, LGJV17] by performing dimensionality reduction on a set of meshes representing human faces with a fixed topological structure (number of vertices and triangulation). Their shape model is a PCA-based implementation that produces new shape instances from a combination of linear basis of shapes. Specifically, a novel shape instance $\boldsymbol{T}_S(\boldsymbol{\beta})$ is obtained from

$$\boldsymbol{T}_S(\boldsymbol{\beta}) = \bar{\boldsymbol{T}} + B_S(\boldsymbol{\beta}) \tag{3.1}$$

where $\bar{\boldsymbol{T}} \in \mathbb{R}^{3N}$ is the mean shape of the model with N vertices and $B_S(\boldsymbol{\beta})$ is a blend

shape function

$$B_S(\boldsymbol{\beta}) = \sum_{d=1}^{|\boldsymbol{\beta}|} \beta_d \boldsymbol{S}_d. \tag{3.2}$$

$\boldsymbol{S} = [\boldsymbol{S}_1, ..., \boldsymbol{S}_D] \in \mathbb{R}^{3N \times D}$ is the orthonormal basis matrix of the shape eigenvectors and $\boldsymbol{\beta} = [\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_D]^T$ is the shape vector that contains shape parameters (blend weights). The number of principal components D defines the degree of freedom of the model.

The first shape model of the human body was created by Allen et al. [ACCP03]. They constructed a PCA model from 250 registered body scans in an A-pose coming from the CAESAR dataset (Figure 3.2). Registration is the process of finding a spatial transformation that aligns two sets of vertices. This allows us to parameterize a collection of meshes to a form where each mesh has the same number of vertices and topology by aligning a source surface (e.g. body template) to the target (e.g. body scan). Registered meshes are said to be in dense correspondence with one another because semantically corresponding points are mapped onto each other.

The pose changes are obtained by applying linear blend skinning which interpolates rotations matrices assigned to the joints to transform a vertex. This technique, widely used in computer graphics, causes the loss of volume of the surface close to the joints. The follow-up work [ACPH06] addresses this issue by adding a corrective offset. Newer models have been developed to improve modelling of pose and shape variations and include dynamic soft-tissue deformations [ASK+05, HSS+09, CLZ13, HLRB12, PMRMB15].

3.2 SMPL

We discuss the Skinned Multi-Person Linear model (SMPL, Figure 3.3) [LMR+15] in details because it has become a standard method for 3D human body understanding

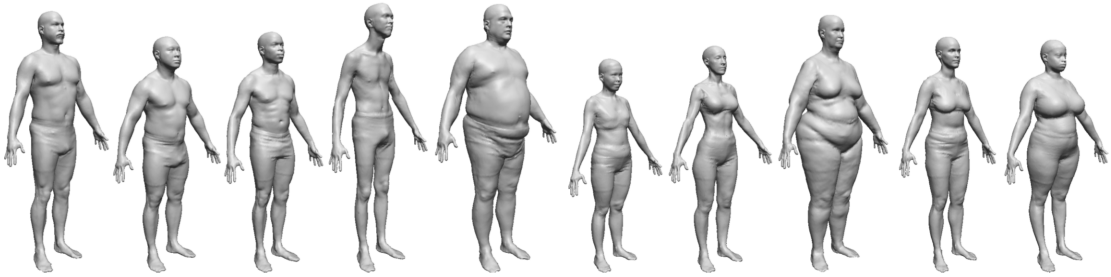


Figure 3.2: Registrations of the body scans from the CAESAR dataset. Reproduced from [ACCP03].

research. To learn shape deformations, the authors use 1,786 high-quality 3D scans of different subjects performing a wide range of poses. The scans are registered with an artist created mesh that has 6,890 vertices.

The model with N vertices and K joints is defined by the function $M(\boldsymbol{\beta}, \boldsymbol{\theta})$ that maps shape and pose parameters ($\boldsymbol{\beta} \in \mathbb{R}^{|\boldsymbol{\beta}|}$ and $\boldsymbol{\theta} \in \mathbb{R}^{K\times 3}$ respectively) into an instance of the model represented by a mesh:

$$M(\boldsymbol{\beta}, \boldsymbol{\theta}; \phi) : \mathbb{R}^{|\boldsymbol{\beta}|\times|\boldsymbol{\theta}|} \rightarrow \mathbb{R}^{3N} \tag{3.3}$$

where $|\boldsymbol{\beta}|$ depends on the training procedure and ϕ is a set of learned model parameters. In subsequent sections, we define each parameter and derive a differentiable formula for $M(\boldsymbol{\beta}, \boldsymbol{\theta}; \phi)$.

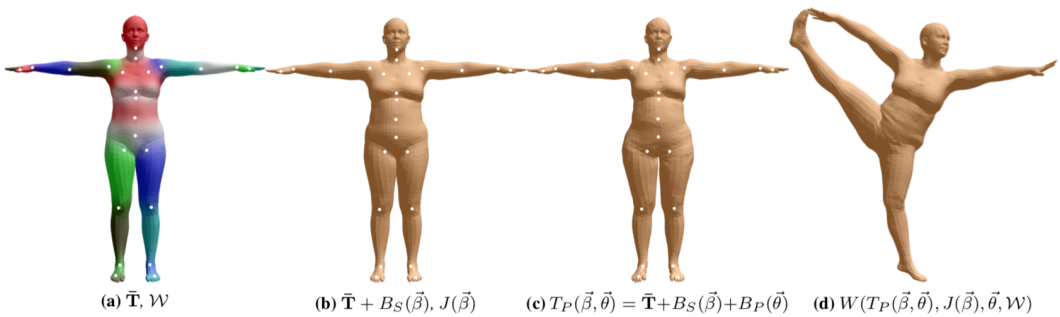


Figure 3.3: SMPL body model. Reproduced from [LMR+15].

3.2.1 Linear Blend Skinning

The objective of this section is to obtain a parameterized control over the pose of the template with realistic deformations. Defining pose deformations for each vertex is impractical. It is more efficient to specify a function that deforms the template with respect to the underlying skeleton. Skinning is the process of controlling deformations of a given mesh in a reference posture (e.g. T-pose for the human body) using a set of deformation primitives such as rigid transformations associated with bones of the underlying skeleton. Deformation can be parameterized by the angles between joints in the kinematic structure.

Approach The general approach for skinning can be presented as follows:

1. Define joints which are local coordinate systems: $\boldsymbol{J} \in \mathbb{R}^{K\times 3}$.
2. Define a set of pose parameters usually represented by three Euler angles per joint: $\boldsymbol{\theta} \in \mathbb{R}^{K\times 3}$.
3. Define a function to represent the word transformation of a joint k: $G_k^{'}(\boldsymbol{\theta}, \boldsymbol{J}) \in \mathbb{R}^{4\times 4}$
4. Assign vertices into the object space of each joint. Each vertex can be assigned to multiple joints.
5. Define a skinning function that transforms vertices according to the joint transformations and weights defined for each vertex assignment.

Representing Rotations The kinematic structure is a tree which connects joints into a hierarchical structure needed to represent rotations. The pose is defined by $\boldsymbol{\theta} = [\boldsymbol{\omega}_1^T, ..., \boldsymbol{\omega}_K^T]^T$ where $\boldsymbol{\omega}_j \in \mathbb{R}^3$ denotes the axis-angle representation of the relative rotation of part j with respect to its parent in the kinematic tree. The axis

angle ω_j for every joint j is transformed to a local 3×3 rotation matrix $\boldsymbol{R}_j$ using the Rodrigues' rotation formula:

$$\boldsymbol{R}_j = \boldsymbol{I} + \hat{\bar{\boldsymbol{\omega}}}_j \sin ||\boldsymbol{\omega}_j|| + \hat{\bar{\boldsymbol{\omega}}}_j \cos ||\boldsymbol{\omega}_j|| \tag{3.4}$$

where $\bar{\boldsymbol{\omega}}_j = \frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}_j||}$ is the unit norm axis of rotation, $\hat{\bar{\boldsymbol{\omega}}}_j$ the skew-symmetric matrix of the 3-vector $\bar{\boldsymbol{\omega}}_j$, and $\boldsymbol{I} \in \mathbb{R}^{3x3}$ is an identity matrix. The skew-symmetric matrix is a square matrix which transpose equals its negative, i.e. $\hat{\bar{\boldsymbol{\omega}}}_j^T = -\hat{\bar{\boldsymbol{\omega}}}_j$. It has the following form:

$$\hat{\bar{\boldsymbol{\omega}}}_j = \begin{bmatrix} 0 & -\bar{\omega}_{j3} & \bar{\omega}_{j2} \\ \bar{\omega}_{j3} & 0 & -\bar{\omega}_{j1} \\ -\bar{\omega}_{j2} & \bar{\omega}_{j1} & 0 \end{bmatrix}. \tag{3.5}$$

If we apply the Rodrigues' formula to our skeleton with K joints, we obtain the rotation $\boldsymbol{R}_1$ for the root frame with respect to the world and relative joint coordinate frames $\boldsymbol{R}_2, \boldsymbol{R}_3, ..., \boldsymbol{R}_K$. The world rotation of a joint j depends on rotations of all its parents in the kinematic structure: $\boldsymbol{R}_1 \boldsymbol{R}_2 ... \boldsymbol{R}_{parent(j)}$.

Skinning Functions We denote the ordered set of joint ancestors of a joint k in the kinematic structure as $A(k)$. The world transformation of a joint k is given by:

$$\boldsymbol{G}_k(\boldsymbol{\theta}, J) = \prod_{j \in A(k)} \begin{bmatrix} R_{j11} & R_{j12} & R_{j13} & j_{j1} \\ R_{j21} & R_{j22} & R_{j23} & j_{j2} \\ R_{j31} & R_{j32} & R_{j33} & j_{j3} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

where $\boldsymbol{j}_j$ is a 3-element vector in $\boldsymbol{J}$.

In order to find the joint's position in the world coordinate system, we need to apply an inverse of $\boldsymbol{G}_k$ for the reference pose $\boldsymbol{\theta}^*$: $\boldsymbol{G}_k(\boldsymbol{\theta}^*, \boldsymbol{J})^{-1} \boldsymbol{J}_k$.

Now, we are able to find the new joint's position $\boldsymbol{J}'_k$:

$$\boldsymbol{G}'_k(\theta, \boldsymbol{J}) = \boldsymbol{G}_k(\boldsymbol{\theta}, \boldsymbol{J})\boldsymbol{G}_k(\boldsymbol{\theta}^*, \boldsymbol{J})^{-1}, \tag{3.7}$$

$$\begin{bmatrix} \boldsymbol{J}'_k \\ 1 \end{bmatrix} = \boldsymbol{G}'_k(\boldsymbol{\theta}, \boldsymbol{J}) \begin{bmatrix} \boldsymbol{J}_k \\ 1 \end{bmatrix} \tag{3.8}$$

This gives as transformation for the underlying skeleton. However, we also want to compute new positions for template vertices. The following subsections describe two skinning functions.

Rigid Skinning In rigid skinning, each vertex is associated with a single joint and transformed into the world space by the appropriate joint matrix. Therefore, every vertex $\bar{\boldsymbol{t}}_i \in \mathbb{R}^3$ in the rest pose $\bar{\boldsymbol{T}} \in \mathbb{R}^{3N}$ assigned to a joint k is transformed into $\bar{\boldsymbol{t}}_i{}'$ according to:

$$\begin{bmatrix} \bar{\boldsymbol{t}}_i{}' \\ 1 \end{bmatrix} = \boldsymbol{G}'_k(\theta, \boldsymbol{J}) \begin{bmatrix} \bar{\boldsymbol{t}}_i \\ 1 \end{bmatrix} \tag{3.9}$$

This approach leads to discontinuities near joints and causes a separation of segments that should be continuous.

Linear Blend Skinning We can remove the discontinuity by linearly blending vertices near the joint. This method is wildly used in the industry due to its computational efficiency. In linear blend skinning, each joint influencing deformation of the mesh is given a scalar weight and the weighted sum gives the vertex position in the new pose:

$$\begin{bmatrix} \bar{\boldsymbol{t}}_i{}' \\ 1 \end{bmatrix} = \sum_{k=1}^{K} (w_{i,k}\boldsymbol{G}'_k(\boldsymbol{\theta}, \boldsymbol{J})) \begin{bmatrix} \bar{\boldsymbol{t}}_i \\ 1 \end{bmatrix} \tag{3.10}$$

where $\boldsymbol{W} \in \mathbb{R}^{NxK}$ are blend weights which elements $w_{i,k}$ represent how much the rotation of part k influences the vertex i. To support efficient computation, $\boldsymbol{W}$ should be sparse with usually at most 4 parts influencing a vertex and each row of $\boldsymbol{W}$ should sum to 1 (e.g. this allows defining the last column of $\boldsymbol{W}$ implicitly and compute it at run-time). We denote the skinning function that transforms each vertex of the template according to Equation 3.10 in the following way:

$$W(\boldsymbol{T}, \boldsymbol{J}, \boldsymbol{\theta}, \boldsymbol{W}) : \mathbb{R}^{3N \times |\boldsymbol{J}| \times |\boldsymbol{\theta}| \times |\boldsymbol{W}|} \rightarrow \mathbb{R}^{3N}. \tag{3.11}$$

3.2.2 Joint Locations

The joints are defined as the function of shape parameters $\boldsymbol{\beta}$ and represented by their 3D locations in the rest pose:

$$J(\boldsymbol{\beta}; \mathcal{J}, \bar{\boldsymbol{T}}, \boldsymbol{S}) = \mathcal{J}(\bar{\boldsymbol{T}} + B_S(\boldsymbol{\beta}; \boldsymbol{S})) \tag{3.12}$$

The matrix $\mathcal{J}$ regresses from vertex locations to joint locations. It can be initialized by segmenting the model into K rings surrounding anatomical joints. This segmentation is used to compute the joint locations by taking the average of the ring segments.

3.2.3 Pose-Dependent Deformations

We denote as $R : \mathbb{R}^{|\boldsymbol{\theta}|} \rightarrow \mathbb{R}^{9K}$ a function that maps a pose vector $\boldsymbol{\theta}$ to a vector of concatenated rotation matrices (Equation 3.4). The vertex displacement $B_P(\boldsymbol{\theta}; \boldsymbol{P})$ from the mean template is computed as follows:

$$B_P(\boldsymbol{\theta}; \boldsymbol{P}) = \sum_{n=1}^{9K} (R_n(\boldsymbol{\theta}) - R_n(\boldsymbol{\theta}^*)) \boldsymbol{P}_n \tag{3.13}$$

Notation	Dimensionality	Description
$\boldsymbol{\beta}$	$\mathbb{R}^D$	A vector of shape parameters.
$\boldsymbol{\theta}$	$\mathbb{R}^{K\times 3}$	A matrix of pose parameters.
$\bar{\boldsymbol{T}}$	$\mathbb{R}^{3N}$	The mean of the shape model.
$\boldsymbol{W}$	$\mathbb{R}^{N\times K}$	Blend weights of the skinning function.
$\boldsymbol{S}=[\boldsymbol{S}_1,...,\boldsymbol{S}_D]$	$\mathbb{R}^{3N\times D}$	A matrix of shape displacements.
$\boldsymbol{P}=[\boldsymbol{P}_1,...,\boldsymbol{P}_{9K}]$	$\mathbb{R}^{3N\times 9K}$	A matrix of pose displacements.
$\boldsymbol{J}$	$\mathbb{R}^{K\times 3}$	Joints in the rest pose template.
$\mathcal{J}$	$\mathbb{R}^{K\times N}$	A matrix that transforms rest vertices into rest joints.

Table 3.1: A summary of SMPL model parameters.

Notation	Dimensionality	Description
$B_S(\boldsymbol{\beta})$	$\mathbb{R}^D \rightarrow \mathbb{R}^{3N}$	A blend shape function.
$B_P(\boldsymbol{\theta})$	$\mathbb{R}^{\lvert\boldsymbol{\theta}\rvert} \rightarrow \mathbb{R}^{3N}$	A pose displacement function.
$W(\bar{\boldsymbol{T}}, J, \boldsymbol{\theta}, \boldsymbol{W})$	$\mathbb{R}^{3N\times\lvert\boldsymbol{J}\rvert\times\lvert\boldsymbol{\theta}\rvert\times\lvert W\rvert} \rightarrow \mathbb{R}^{3N}$	A linear blend skinning function.
$J(\boldsymbol{\beta})$	$\mathbb{R}^D \rightarrow \mathbb{R}^{3K}$	A joint regressor to compute K joints.

Table 3.2: A summary of SMPL model functions.

where the blend shapes $\boldsymbol{P}_n \in \mathbb{R}^{3N}$ are displacement vectors learned to minimize the distance between the model and registrations. $R_n(\boldsymbol{\theta})$ denotes the n^{th} element of $R(\boldsymbol{\theta})$.

3.2.4 Summary

The model $M(\boldsymbol{\beta}, \boldsymbol{\theta}; \phi) : \mathbb{R}^{\lvert\boldsymbol{\beta}\rvert x\lvert\boldsymbol{\theta}\rvert} \rightarrow \mathbb{R}^{3N}$ with N vertices, K joints, and D shape components is defined as:

$$M(\boldsymbol{\beta}, \boldsymbol{\theta}) = W(T_P(\boldsymbol{\beta}, \boldsymbol{\theta}), J(\boldsymbol{\beta}), \boldsymbol{\theta}, \boldsymbol{W}) \tag{3.14}$$

The template is modified in an additive way before skinning:

$$T_P(\boldsymbol{\beta}, \boldsymbol{\theta}) = \bar{\boldsymbol{T}} + B_S(\boldsymbol{\beta}) + B_P(\boldsymbol{\theta}) \tag{3.15}$$

Table 3.1 and Table 3.2 summarize model parameters and functions described in this chapter.

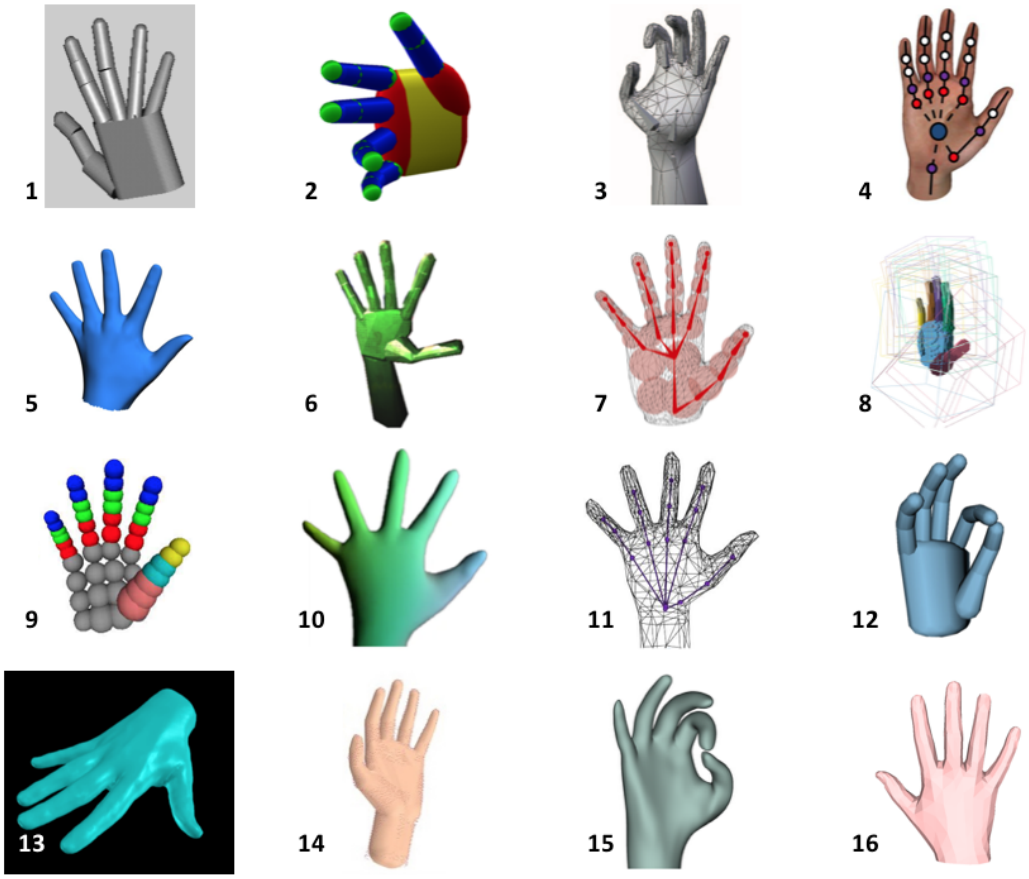


Figure 3.4: Hand models (sorted by a year): 1 - Rehg et Kanade [RK94], 2 - Oikonomidis et al. [IOA11], 3 - De La Gorce et al. [dLGFP11], 4 - LibHand [Š11], 5 - Ballan et al. [BTG+12], 6 - Melax et al. [MKO13], 7 - Sridhar et al. [SOT13], 8 - Schmidt et al. [SNF14], 9 - Qian et al. [QSW+14b], 10 - Sharp et al. [SKR+15], 11 - Khamis et al. [KTS+15], 12 - Tagliasacchi et al. [TST+15], 13 - Tzionas et al. [TBS+16b], 14 - Tkach et al. [TPT16], 15 - Taylor et al. [TBC+16], 16 - Romero et al. (MANO) [RTB17]. Figures reproduced from the cited papers.

3.3 Hand Models

The progress of hand modelling is aligned with developments in statistical deformable modelling of the human body. We group hand models into three categories depending on the shape representation. We visualize selected models in Figure 3.4.

Geometric Approximations Okinomidis et al. [IOA11, OKA11, OKA12, OLA14] model the hand by approximating its shape with 3D geometric primitives: *"The palm*

is modeled as an elliptic cylinder and two ellipsoids for caps. Each finger consists of three cones and four spheres, except for the thumb which consists of an ellipsoid, two cones and three spheres." The parametrization of hand kinematics is crafted manually and constraints are based on anatomical studies. Qian et al. [QSW+14b] approximate the polygonal mesh model used by Okinomidis with 48 spheres to provide a simple geometric representation that speeds up computations and supports real-time hand tracking. Melax et al. [MKO13] propose a representation based on the union of convex rigid bodies. The geometric primitives are also considered in models produced by Rehg and Kanade [RK94], in the work based on real-time registration [TST+15], or to obtain a surface composed of the convex-hull of the spheres [TPT16]. In DART [SNF14], rigid geometric parts are voxelized and the surface is represented via a signed distance function. The volumetric extent of the hand can also be represented as a mixture of weighted 3D Gaussian functions [SMOT15, SOT13].

Template Skinning A linear blend skinning (LBS) model [LCF00] consisting of a triangular mesh with an underlying kinematic skeleton and a set of skinning weights is applied for motion capture of hands interacting with objects [BTG+12, TBS+16b]. The achieved results [BTG+12] are robust but require manual rigging and a multi-camera capture setup. Taylor at al. [TBC+16] also use LBS but they represent the smooth surface based on approximate Loop subdivision. To fit a hand into monocular images, De La Gorce et al. [dLGFP11] model the hand with a triangulated surface posed using skeleton subspace deformation. Scaling parameters are defined for each bone to accommodate different hand shapes. Recently, a Phong surface model [SCY+20] has been proposed to enable fast hand tracking on a head-mounted holographic computer (HoloLens 2) with a low computational budget.

Personalized Shape Modeling Taylor et al. [TSR+14] address a problem of acquiring a user-specific model from depth sequences. The model is based on subdivision surfaces that allows simultaneous optimization over correspondences and model parameters using nonlinear minimization. Khamis et al. [KTS+15] learn a deformable hand surface model from a set of noisy partial depth scans of 50 people performing various poses. A hand shape is parameterized as a linear combination of the basis matrices and the subdivision surface model is posed with LBS. MANO (*hand Model with Articulated and Non-ridig defOrmations*) [RTB17], discussed in the previous chapter on hand reconstruction background, is similar to SMPL and represents subject-specific shapes with a learned linear function and pose-dependent corrective blend shapes. DeepHandMesh [MSL20] learns a personalized model in a weakly-supervised manner from multi-view depth maps and estimated 3D joint coordinates.

3.4 Unified Models

SMPL+H [RTB17] extends SMPL by learning hand shape variations from full body scans that have low resolution for hands and transferring pose-dependent offsets from MANO. Frank [JSS18] stitches a publicly available deformable face model [CWZ+13] and an artist-created rigged hand template to SMPL. SMPL-X [PCG+19] attaches the FLAME head model [LBB+17] to SMPL+H and fits the full model to scans to avoid stitching artifacts that are evident in Frank. Recently, researchers proposed to replace PCA-based models and learn non-linear shape variations by training variational autoencoders implemented as fully-connected layers [XBZ+20]. Their unified model is learned from full body scans and close-up scans of heads and hands.

3.5 Geometric Deep Learning

In the following subsections, we discuss seminal works on graph convolutions, that constitute building blocks of a mesh autoencoder that we train in Chapter 5 where filtering is performed in the spectral domain. Our interest lies in applying deep learning to learn invariant shape features in a triangular mesh [BBL+17]. To this end, spectral approaches [BZSL14, DBV16, KW17, LMBB18] express convolutions in the frequency domain. Spatial convolutions define local charting [MBBV15, BMRB16, MBM+17, LDCK18] that allows addressing the issue of orientation ambiguity by imposing a canonical direction or angular max pooling. Moreover, the fixed topology of a triangular mesh can be leveraged to design pooling functions based on mesh coarsening methods. Alternatively, convolution operators can be defined with an attention mechanism to weight the neighbour selection [VBV18, VCC+18].

3.5.1 Graph Laplacian

We are interested in analyzing signals [SNF+13] defined on an undirected graph $G = (V, \mathcal{E}, W)$ where V is a set of vertices, $\mathcal{E}$ a set of edges, and W a weighted adjacency matrix. A signal $f : V \to \mathbb{R}$ can be represented as a vector $\boldsymbol{f} \in \mathbb{R}^N$, where the i^{th} component of the vector $\boldsymbol{f}$ represents the function value at the i^{th} vertex in V.

Non-normalized graph Laplacian is defined as

$$\Delta = D - W \tag{3.16}$$

where $D = diag(\sum_j W_{ij})$ is the degree matrix. For any signal $\boldsymbol{f} \in R^N$ the graph Laplacian satisfies

$$(\Delta f)(i) = \sum_{j \in N_i} W_{ij}(f(i) - f(j)) \tag{3.17}$$

where $N_i = \{j \mid (i,\ j) \in E\}$ it the neighbourhood of a vertex i.

Graph Laplacians are symmetric and positive semi-definite. Therefore they have a complete set of orthogonal eigenvectors $\mathbf{\Phi} = (\phi_1, ..., \phi_n)$ where $n = |V|$ with real, non-negative eigenvalues $\lambda_1, ..., \lambda_n$ satisfying

$$\Delta\phi_i = \lambda_i\phi_i. \tag{3.18}$$

Due to the above properties, Δ admits an eigendecomposition

$$\Delta = \phi\Lambda\phi^T \tag{3.19}$$

where $\Lambda = (\lambda_1, ..., \lambda_N)$.

The Dirichlet energy of f is the measure of the smoothness defined as

$$||\boldsymbol{f}||_G^2 = \frac{1}{2}\sum_{ij}^{n} w_{ij}(f_i - f_j)^2 = \boldsymbol{f}^T\Delta\boldsymbol{f} \tag{3.20}$$

3.5.2 Graph Fourier Transform

Graph Fourier Transform is the expansion of $\boldsymbol{f}$ in terms of the eigenvectors of the graph Laplacian

$$\hat{f}(\lambda_l) =< \boldsymbol{f}, \phi_l >= \sum_{i=1}^{N} f(i)\phi_l^*(i), \tag{3.21}$$

which can be written in the matrix form:

$$\hat{\boldsymbol{f}} = \boldsymbol{\phi}^T\boldsymbol{f}. \tag{3.22}$$

From the previous equation, we can see that the inverse graph Fourier transform is

given by

$$f(i) = \sum_{l=1}^{N} \hat{f}(\lambda_l)\phi_l(i) \tag{3.23}$$

or equivalently

$$\boldsymbol{f} = \boldsymbol{\phi}\hat{\boldsymbol{f}}. \tag{3.24}$$

3.5.3 Graph Filtering

Graph filtering is the process of representing an input signal $\boldsymbol{f}_{in}$ in terms of the eigenvectors of the graph Laplacian and amplifying or attenuating the contributions of some of the eigenvectors with a filter $\hat{\boldsymbol{h}}$ to obtain $\boldsymbol{f}_{out}$:

$$f_{out}(\lambda_l) = \hat{f}_{in}(\lambda_l)\hat{h}(\lambda_l). \tag{3.25}$$

If we take an inverse graph Fourier transform, we obtain

$$\hat{f}_{out}(i) = \sum_{l=1}^{N} \hat{f}_{in}(\lambda_l)\hat{h}(\lambda_l)\phi_l(i). \tag{3.26}$$

Small spatial support of the convolutional filter reduces the risk of overfitting. From the Parseval's identity we observe that smoothness in the frequency domain corresponds to localization in space:

$$\int_{-\infty}^{+\infty} |x|^{2k} |f(x)|^2 \, dx = \int_{-\infty}^{+\infty} \left| \frac{\partial^k \hat{f}(\omega)}{\partial \omega^k} \right|^2 dw. \tag{3.27}$$

3.5.4 Spectral CNN

Bruna et al. [BZSL14] expressed convolutions in the spectral domain

$$(f \star h) = \hat{h}(\Delta) f \tag{3.28}$$

where the filter is parameterized using a smooth spectral transfer function:

$$\hat{h}(\Delta) = \Phi \hat{h}(\Lambda) \Phi^T \tag{3.29}$$

$$\hat{h}(\Lambda) = diag(\hat{h}(\lambda_1), ..., \hat{h}(\lambda_N)) \tag{3.30}$$

This method has multiple disadvantages. The filters are basis-dependent and thus do not generalize across domains. The method requires $O(n)$ parameters per layers and the computation of forward and inverse Fourier transforms has $O(n^2)$ complexity. Moreover, there is no guarantee of spatial localization of filters.

3.5.5 Fast Spectral CNN

Defferrard et al. [DBV16] parameterized the filter with a Chebyshev expansion of order r such that

$$h(\tilde{\Delta}) = \sum_{j=0}^{r} \alpha_j T_j(\tilde{\Delta}) \tag{3.31}$$

where $T_k(\lambda) = 2\lambda T_{k-1}(\lambda) - T_{k-2}(\lambda)$ with $T_0 = 1$ and $T_1 = \lambda$. $\tilde{\Delta} = 2\lambda_n^{-1}\Delta - \mathrm{I}$ denotes rescaling of the Laplacian eigenvalues from the interval $[0,\ \lambda_n]$ to $[-1, 1]$. This approach has a constant number of parameters per layer and filters are guaranteed to have r-hops support. There is also no explicit computation of the eigenvectors resulting in $O(nr)$ computational complexity. Noteworthy, the Graph Convolutional Network (GCN) [KW17] that has gained large popularity simplifies this method by considering first-order polynomials.

Chapter 4

Iterative Model Fitting

The existing hand pose estimation and mesh reconstruction systems significantly underperform in non-laboratory environments. Given prevalence of deep learning approaches, collecting the data from a distribution of non-laboratory images could enable in the wild applications. However, annotating 2D images captured by hand-held devices with 3D keypoints poses extreme challenges. One could try to leverage a multi-camera setup; however, this is costly and has limited scalability to the extent that it cannot be applied to everyday scenarios. There has been a growing interest in building synthetic datasets. While there are some promising results, the difference in performance of networks trained on datasets that come from a similar distribution to the test samples is substantial compared to networks trained on rendered samples. Moreover, training difficulty increases as the network is more prone to overfitting and the sheer process of constructing synthetic datasets can be cumbersome. For example, until recently there was no statistical texture model of the human hand. Therefore, sampling realistic skin patterns was an unresolved research problem. In fact, we solve this problem in Chapter 9 but this is preceded by a large-scale data collection effort. Another avenue of research is in aligning latent spaces of datasets sampled from different distributions. These methods are conceptually very interest-



Figure 4.1: Minimizing the 2D reprojection error leads to anatomically invalid poses in the absence of prior on joint rotation angles.

ing but have not proven state-of-the-art performance.

We propose an alternative solution to gather dense 3D mesh annotations for images captured in a natural environment. We observe that 2D keypoint detectors perform well because data collection can be performed by human contractors who annotate keypoints manually. Given a robust 2D keypoint detector, we propose a system to align a 3D hand model with 2D annotations. The task is ill-posed because multiple poses can be projected onto the image such that sparse 2D keypoints are aligned. The unconstrained approach usually leads to anatomically invalid poses (e.g. fingers bent in the opposite direction) as show in Figure 4.1.

To address the pose ambiguity issues, we propose a novel prior on hand joint rotations. Furthermore, we introduce a fast and accurate method for dataset generation from unlabeled images, that relies on fitting a hand model to 2D keypoint detections while respecting a prior on rotation angles. Our data generation algorithm processes a sequence of YouTube videos, generating a curated dataset of images with 3D hand meshes. As a result, we introduce the first dataset with 3D hand annotations of 50,175 hand meshes aligned with in the wild images comprising hundreds of subjects performing a wide variety of tasks. The quantitative evaluation is performed in Chapter 6 where we also show that systems trained with the novel dataset significantly outperform existing approaches in laboratory and in the wild scenarios. Both qualitative and quantitative results show cutting-edge results.

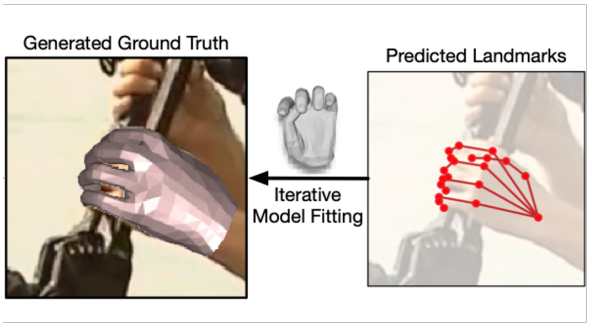


Figure 4.2: Iterative model fitting.

To summarise, our contributions are as follows:

- We introduce an automated approach to generate training data from unannotated images for 3D hand reconstruction and pose estimation. The proposed method is scalable and does not require large computational resources.
- We release the dataset of 50,000 meshes aligned with frames selected from over a hundred YouTube videos.

4.1 YouTube 3D Hands Dataset

Despite the wealth of possible applications, there are no datasets for monocular 3D hand reconstruction from RGB images in-the-wild. The only existing collection of images of hands captured in a non-laboratory environment [SJMS17] contains less than 3000 samples with a manual annotation of 2D points.

In order to train a neural network for the 3D reconstruction of hands across different domains, we have built a system for fast and automated dataset generation from YouTube videos, providing us with a diverse set of challenging hand images. Instead of annotating them manually, we use a weakly-supervised approach that first detects keypoints using OpenPose, and then lifts them into 3D shapes by iteratively fitting a 3D deformable model. While the accuracy of the trained model could be bounded by

the performance of OpenPose, we show in Chapter 6 that it results in a state-of-the-art 3D hand reconstruction and pose estimation system when evaluated on external datasets with manual annotations.

4.1.1 3D Shape Representation

Our approach relies on fitting the MANO model in tandem with a prior on angles of rotation. MANO predicts $N = 778$ vertices on the hand surface and $K = 16$ joints through a differentiable function $M(\beta, \theta)$ that maps shape $\beta \in \mathbb{R}^{|\beta|}$ and pose $\theta \in \mathbb{R}^{K\times 3}$ parameters into an instance of the model represented by a mesh:

$$M(\beta, \theta, \boldsymbol{T}_\delta, s; \phi) : \mathbb{R}^{|\beta|\times|\theta|\times|\boldsymbol{T}_\delta|\times|s|} \rightarrow \mathbb{R}^{N\times 3} \tag{4.1}$$

where $|\beta|$ depends on the training procedure and ϕ is a set of learned model parameters that we omit in a further discussion. Moreover, we have camera parameters s for scaling the model and $\boldsymbol{T}_\delta \in \mathbb{R}^3$ to translate it. Global orientation is modeled by the first row of θ.

Rather than model the joint angles as free variables, which can lead to impossible estimates during optimization, we constrain them to lie in the convex hull of some pre-computed cluster centers. For joint i we use $C = 64$ Euler-angle clusters $P_i^1, ..., P_i^C$ obtained via k-means [GK19] and express any angle as follows:

$$\theta_i = P(w)_i = \frac{\sum_{c=1}^{C} \exp(w^c) P_i^c}{\sum_{c=1}^{C} \exp(w^c)}. \tag{4.2}$$

While constraining the final angle estimate to take plausible values, at the same time this expression allows us to optimize over unconstrained variables w^c. We represent all constrained angles in terms of a parameter matrix $w \in \mathbb{R}^{K\times C}$, while allowing global orientation w_0 to be unrestricted.

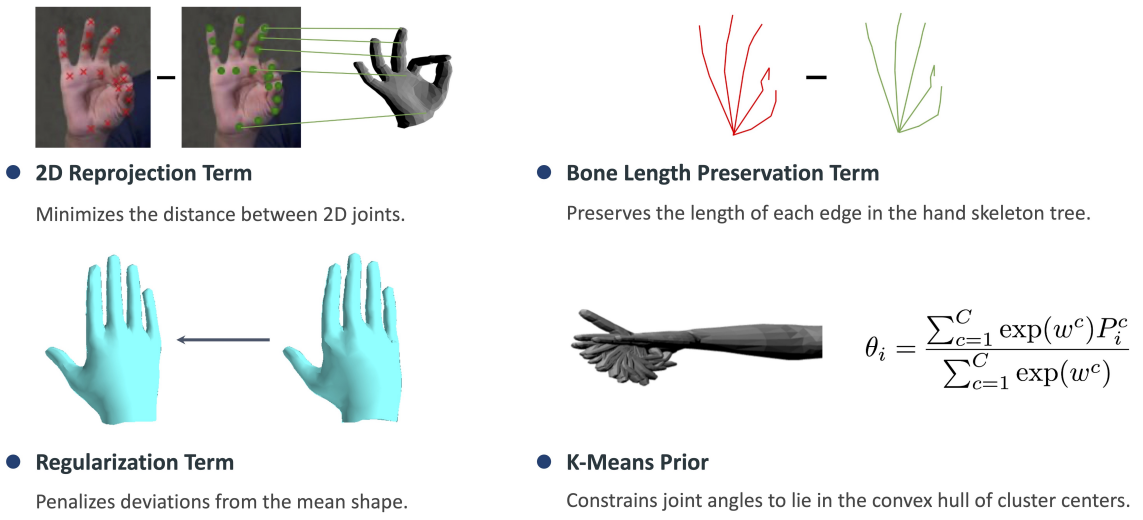


Figure 4.3: Error terms for the parametric model fitting and pose prior explanation.

We have observed by training various models and building a graphical interface for manipulating latent variables that this simple approach performs particularly well in our scenario. Specifically, it requires only a small dataset of angles, unlike VAE priors; restricts pose space to plausible poses, unlike PCA priors that are characterized by unrealistic interpolation which can lead to intersecting fingers; and allows fitting unseen poses. However, it does not model pairwise dependencies, which we leave to future work.

4.1.2 Parametric Model Fitting

Our supervision comes in the form of 2D landmarks extracted by OpenPose. We define a fitting procedure that tries to find a matching 3D pose from the MANO mesh through a sparse matrix $\mathcal{J} \in \mathbb{R}^{N \times (K+F)}$ that regresses from model vertices to $K = 16$ joint and $F = 5$ fingertip positions, delivering the hand pose $J \in \mathbb{R}^{(K+F) \times 3}$:

$$J(\beta, w, \boldsymbol{T}_\delta, s) = \mathcal{J}^T M(\beta, P(w), \boldsymbol{T}_\delta, s). \tag{4.3}$$

We fit the model to 2D annotations by minimizing the following objective:

$$\{\beta^*, w^*, \boldsymbol{T}_\delta^*, s^*\} = \arg\min_{\beta, w, \boldsymbol{T}_\delta, s} (E_{2D} + E_{bone} + E_{reg}), \tag{4.4}$$

consisting of a 2D reprojection term E_{2D}, a bone length preservation cost E_{bone}, and a regularization term E_{reg}. We visualize the error terms in Figure 4.3.

In particular, the joint error term minimizes the distance between 2D joints

$$E_{2D}(\beta, w, \boldsymbol{T}_\delta, s) = ||\Lambda_{2D}(\Pi_K(J(\beta, P(w), \boldsymbol{T}_\delta, s)) - Y)||^2 \tag{4.5}$$

where Y are 2D detector predictions and Π_K is the intrinsic camera projection to 2D, initialized as in the approach of Pavlakos et al. [PCG+19]. Λ_{2D} is an experimentally chosen mask that amplifies the influence of fingertips by 1.7 and wrist by 2.5 and reduces influence of the metacarpophalangeal (MCP) joints (base of each finger) that are often inaccurately annotated by 0.7.

The bone loss E_{bone} ensures that the length of each edge in the hand skeleton tree $\mathcal{E}$ is preserved, i.e.

$$E_{bone}(\beta, w, \boldsymbol{T}_\delta, s) = \sum_{(i,j)\in\mathcal{E}} \mid ||J_{2D_j} - J_{2D_i}|| - ||Y_j - Y_i|| \mid \tag{4.6}$$

where $J_{2D_i} = \Pi_K(J(\beta, w, \boldsymbol{T}_\delta, s))_i$.

The regularization term $E_{reg}(\beta, \theta) = \lambda_\theta||\theta||^2 + \lambda_\beta||\beta||^2$ penalizes deviations from the mean pose to ensure realistic deformations. The hyperparameters $\lambda_\theta = 0.1$ and $\lambda_\beta = 1000$ were chosen experimentally.

4.1.3 Optimization

We use the Adam optimizer with different learning rates for camera, pose, and shape parameters (10^{-2}, 10^{-2}, 10^{-5} respectively) and small learning rate decay (multiplicative factor of 0.95) after each 500 iterations. We start with 1,500 iterations optimizing over camera parameters and global orientation where the joints set is reduced to a wrist and MCP joints excluding thumb. Afterwards, we perform 2,500 iterations over all parameters. We fit 4,000 samples per batch on GeForce RTX 2080 Ti which takes on average 10 min.

4.1.4 Automated Data Collection

The data collection system iterates a list of YouTube links, downloads a video, extracts frames, runs OpenPose, fits MANO to each frame, and selects a small subset of filtered samples. The depth of the projected mesh is proportional to the ratio of standard deviations between X coordinates of the projected mesh and its world position.

Filtering of plausible samples is performed by thresholding total OpenPose confidence score, per-joint confidence score, and the mean squared error between projected MANO joints and OpenPose predictions normalized by the distance from the camera.

To create our `YouTube` dataset, we process 102 videos for the training set and randomly select at most 500 samples per video that meet the threshold conditions. Most of the samples cover sign language conversations performed by people of wide variety of nationality and ethnicity. Some videos include, for example, a hundred people signing a common phrase to a webcam all over the world. The validation and test sets cover 7 videos with an empty intersection of subjects with the training set. We



Figure 4.4: Samples from the YouTube dataset.

selected test videos to be diverse and challenging including conversations captured outdoor, dance poses, American, Australian, and Taiwanese sign languages. Additionally, we run our system on the COCO dataset [LMB+14] and extract 7,048 hand images for training. The combined training set contains 54,173 samples and validation and test sets count 1,525 images each. Samples from the dataset are visualized in Figure 4.4.

4.2 Conclusion

We have introduced the largest and most diverse dataset of 3D hand meshes aligned with in the wild images. In order to collect the images, we proposed a fully automated procedure that extracts frames from YouTube videos and fits a parametric hand model into 2D keypoint detections. The dataset can enable hand reconstruction systems that work in non-laboratory environments which can lead to novel augmented and virtual reality applications or sign language recognition systems.

Chapter 5

Mesh Autoencoder

The existing body and hand models have limited ability to represent shape variations. Changes in the subject-specific hand shape are usually modeled with linear bases and surface changes due to skeleton rotations are obtained by blending rotation matrices assigned to the joints. Some methods propose to learn additional offset functions to address artifacts of the skinning methods or learn soft-tissue deformations. The composition of different functions leads to over-parameterization that slows down inference time and increases training difficulty.

Promising results came from Ranjan et al. [RBSB18] who proposed to use spectral convolutions to find a low-dimensional non-linear representation of the human face in an unsupervised manner. The network requires 75% less parameters than a PCA-based morphable model and obtains competitive reconstruction results. Cheng et al. [CBZ+19] define this architecture in an adversarial setting obtaining more detailed reconstructions and Zhou et al. [ZDKZ19] include additional features in the graph structure to predict the face shape with texture.

Inspired by this approach, we propose to train a mesh autoencoder on highly articulated data of human hands (Figure 5.1). Our experiments show that we are able

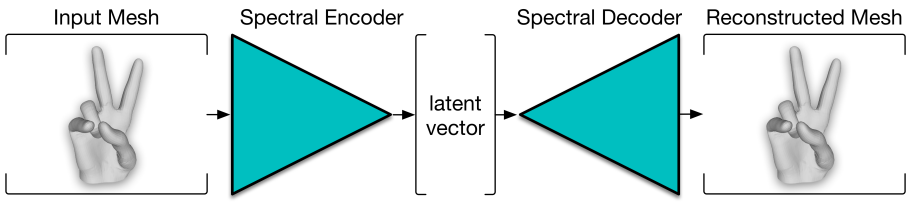


Figure 5.1: Mesh autoencoder learns a low-dimensional latent representation of 3D hand shapes.

to learn a latent non-linear representation of hand meshes in an end-to-end manner. In the proposed framework, pose deformations are learned directly by a neural network and therefore the model does not suffer from computational drawbacks and training inefficiency of skinning methods with corrective offsets. It also has a significantly smaller number of parameters than the initial model and constrains the space of valid poses.

5.1 Model Implementation

To find a non-linear representation of 3D hand shapes, we follow design choices of CoMA [RBSB18] which is a mesh autoencoder with four layers of fast spectral convolutions followed by sampling.

We start with input mesh with 7,907 vertices followed by sequence of convolutions (16, 32, 32, 48 filters) and downsampling - that we discuss later - after each convolutional layer (4, 4, 2, 2 graph reduction factors). Afterwards, we apply a fully connected layer to obtain a latent vector with 64 parameters. The decoder is symmetric. We choose leaky ReLU for the activation function based on experimental evaluation and we use a smaller filter with the Chebyshev polynomial of order $r = 3$.

The training data comes from a MANO-like model with 7,907 vertices. We train the network for 6 hours on a single GeForce RTX 2080 Ti with a batch size 64 and

learning rate 0.001. In addition to the L1 reconstruction loss, we impose an L2 penalty on the latent vector weighted by 5e-7 and we use L2 regularization scaled by 5e-5. The loss function is minimized with AdamW opimizer with a decay factor 10e-6. Our deformable model of the human hand is the decoder with $Z = 64$ latent parameters.

Sampling We create a hierarchy of meshes with the number of vertices reduced by the factor of 2 at each stage of the decoder. We contract vertex pairs based on quadric error metrics [GH97] that allow the coarse vertices to be a subset of the denser mesh. We project each collapsed node into the closest triangle in the downsampled mesh [RBSB18] and use the barycentric coordinates of the projected vertex to define interpolation weights for the upsampling matrix.

More specifically, we project a vertex $v_q \in V$ discarded during contraction into the closest triangle $v_i, v_j, v_k \in V_d$ of the downsampled mesh V_d obtaining $\tilde{v}_p$. We compute its barycentric coordinates w_1, w_2, w_3 such that $\tilde{v}_p = w_1 v_i + w_2 v_j + w_3 v_k$ and $w_1 + w_2 + w_3 = 1$. The upsampling matrix $Q_u \in \mathbb{R}^{m \times n}$, for which $V_u = Q_u V_d$ and $m > n$ hold, is formed by setting $Q_u(q, i) = w_i$, $Q_u(q, j) = w_j$, $Q_u(q, k) = w_k$, and $Q_u(q, l) = 0$ for $l \notin \{i, j, k\}$.

5.2 Data Acquisition

There are no large collection of meshes that can be used to train our system. Therefore, we build a new high resolution hand model following MANO with ten times more vertices and removed scale variations from PCA linear bases. Afterwards, we also compute a distribution of valid poses from registrations of around a thousand scans from the MANO dataset to sample realistic meshes for training the autoencoder



Figure 5.2: a) Random shapes sampled from our high resolution linear model. Posed examples can be found in Figure 6.2. b) Samples from the MANO model (taken from [RTB17]).

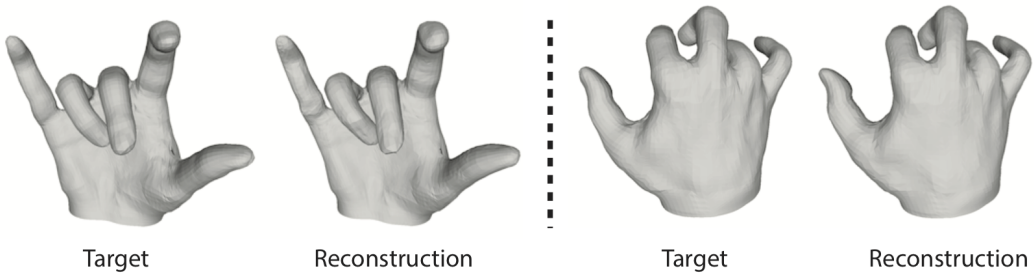


Figure 5.3: Autoencoder reconstruction results for challenging samples in terms of the pose (left pair) and shape (right pair).

Hand Model. The number of vertices in MANO (778) is not sufficient to realistically model shape deformations of the human hand. A potential increase in the number of vertices introduces difficulties with scan registrations and significantly increases optimization time for the tasks of parameter training and pose fitting. Moreover, the shape model in MANO was created by applying a dimensionality reduction to rigidly aligned scans. This results in a first principal component that models scale deformations. This is undesirable for our model fitting procedure that applies a similarity alignment at the beginning of optimization. To address these issues, we

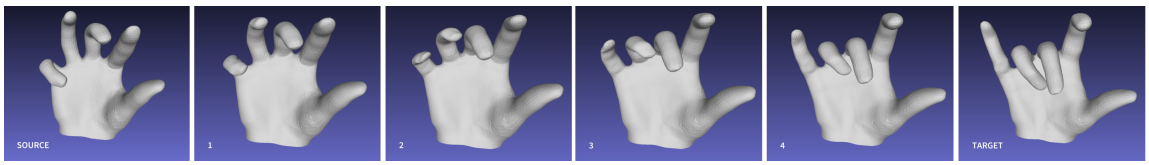


Figure 5.4: Interpolation in the latent space between two random samples. Please, note that source and target shapes (e.g. finger length) are also different.

introduce a hand model with 7,907 vertices which is more than SMPL and MANO combined. We register the reference template to a set of scans with selected keypoints. Then the shape components are computed by applying PCA to the set of Procrustes aligned registrations. The size of the model is controlled by a scale multiplier. Figure 5.2 shows that we are able to model fine-details of the hand surface.

We create a PCA shape model from $D = 36$ registrations of the rest pose $\{\boldsymbol{T}^i\}_{i=1}^{D}$ to obtain the set of parameters $\{\bar{\boldsymbol{T}}, \boldsymbol{S}\}$. Random blend shapes are shown in Figure 5.2. We analyse the cumulative ratio between the variance captured by the active components and the total amount of variance. It turns out that 99% of the variance is explained by 24 principal components and 85% by 6 components.

Pose Sampler. We use samples obtained from our high-resolution linear model to train the proposed mesh autoencoder. The shape coefficients are sampled from a standard normal distribution. In order to sample plausible and diverse hand poses, we resort to the distribution of joint angles in the MANO database. For each 15 joint angles in the human hand kinematic tree, we compute euler-angle clusters via K-means [GK19]. During synthesis of training samples, we randomly select euler-angle cluster centers for each joint, effectively sampling shapes from the manifold of plausible shapes. In our experiments, we have used 64 rotation centers.

5.3 Evaluation

The quantitative evaluation is performed on the task of reconstructing 3D hands from images in Chapter 6. Qualitatively, we show that the autoencoder produces visually indistinguishable reconstructions from the input (Figure 5.3) and has a meaningful latent representation providing smooth transitions in the latent space (Figure 5.4).

5.4 Conclusion

We have introduced the first hand model learned end-to-end from a collection of meshes in an unsupervised way. The model is capable of generating realistic posed hand shapes with 7,907 vertices. We achieve this by training a mesh autoencoder with convolutions performed in the frequency domain. The decoder has only four layers resulting in low computational requirements. Interpolation in the latent space produces meaningful transitions enabling fast optimization for the problem of model fitting by regressing latent parameters as we show in Chapter 6.

Chapter 6

3D Hand Reconstruction in the Wild

Monocular 3D hand reconstruction can facilitate a broad array of applications in human-computer interaction, augmented reality, virtual telepresence, or sign language recognition. Despite a wealth of potential applications, the current state-of-the-art methods do not generalize well to samples captured in a non-laboratory environment and cannot provide real-time mobile inference.

In this chapter, we present solutions that enable on-device reconstruction of a dense hand mesh in the wild that have been already adopted by the industry based on our publications. Even though our method can deliver a dense hand mesh, we outperform systems that try to solve a more narrow task of estimating coordinates of a sparse set of keypoints usually corresponding to hand joints and fingertips. The standard approach is to regress heatmaps of 2D landmarks and find a matching 3D pose [ZB17, POA18, MBS+18]. More recent works, that also focus on full-blown mesh reconstruction, rely on fitting deformable models to estimated landmarks, effectively exploiting a prior distribution in 3D shape space to perform 3D reconstruction [BBT19, ZLM+19, BKK19, HVT+19a]. Such models are usually parameterized by angles of rotation and weights of a linear deformation function [RTB17]. We propose an alternative solution for 3D deformable alignment and reconstruction of

articulated shapes by exploiting the intrinsic structure of the 3D hand meshes using graph convolutions based on our findings from Chapter 5.

Our first key contribution is a graph convolutional mesh reconstruction method from a single image. Furthermore, we propose a weakly-supervised training method for 3D mesh reconstruction trained primarily on a large-scale dataset of hand action images gathered from YouTube videos that we introduced in Chapter 4. Beyond the data-centric advancements, we also obtain a substantial improvement by using a spatial mesh convolution method that is based on neighbourhood ordering. We train the final end-to-end system from scratch by using an objective function that merely enforces the reconstruction of a hand mesh that is aligned with the image. Finally, we show how the proposed system outperforms all existing approaches as evaluated on public benchmarks and enables novel industrial applications.

Our system attains 50% lower hand pose estimation error compared to the currently best model on the in-the-wild scenario, which is our main focus. It also outperforms previous methods on hand pose estimation in controlled environments, without tuning or overfitting on a particular dataset.

To summarise, our contributions are as follows:

- We propose a novel framework for hand reconstruction based on the idea of encoding images into a latent non-linear representation of 3D hand shapes.
- We propose a simple loss function for mesh reconstruction that allows training neural networks with no intermediate supervision.
- We demonstrate that spatial mesh convolutions outperform spectral methods and SMPL-based [LMR+15, RTB17] models for hand reconstruction.
- We largely outperform the current state-of-the-art on pose estimation and mesh

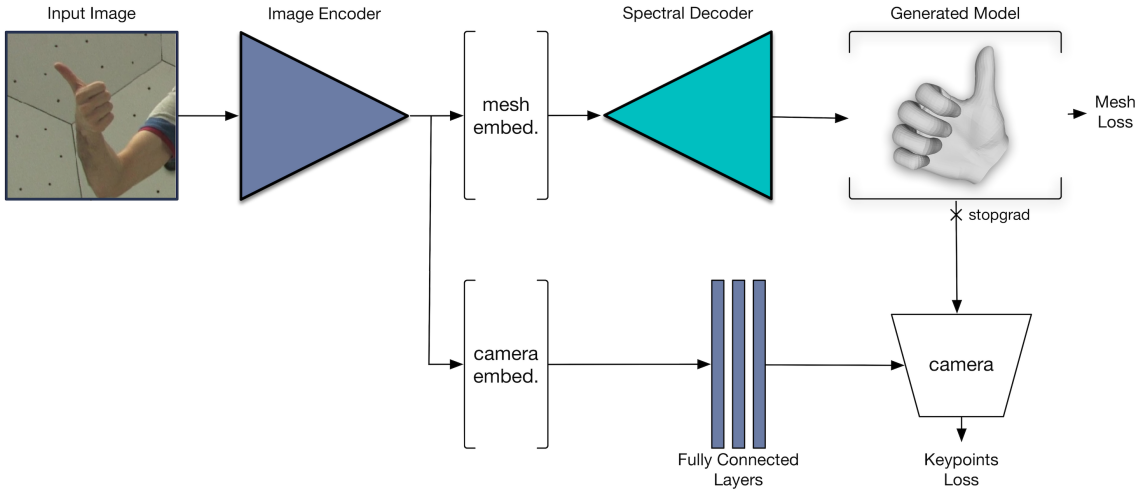


Figure 6.1: Graph Morphable Model regression baseline. The presented system generates a 3D hand model corresponding to the hand in the image. The system consists of two branches trained simultaneously. The network in the top branch is capable of generating 3D hand models from a single image in a wide variety of shapes and poses. The ground truth meshes are aligned to the mean hand model in a canonical frame. The network in the lower branch regresses parameters of a weak perspective projection to align the generated mesh with the image. The gradient from the keypoints loss does not flow through the top branch to prevent the camera regressor from affecting the quality of mesh reconstructions.

reconstruction tasks, while using a simple and single-shot encoder-decoder architecture.

- We show that the system works in real time on mobile devices at 250 FPS while maintaining a superior accuracy.

6.1 Graph Morphable Model

In Chapter 5, we built a mesh autoencoder and showed that a generator can act as a statistical deformable model, capable of modelling different shapes and poses, that we refer to as a Graph Morphable Model (GMM) to emphasise that it is obtained by a dimensionality reduction using graph neural networks. Based on these findings, we propose a method to regress latent vectors that control deformations of this model

from hand image crops which is shown in Figure 6.1.

Network Architecture The single image mesh generator consists of the image encoder E_{image}, for which we use the DenseNet-121 network [HLVDMW17], pretrained on the ImageNet classification task. The outputs of the E_{image} are a mesh embedding that is passed to the pre-trained Graph Morphable Model $\mathcal{D}_{GMM}$, and a camera embedding fed into fully connected layers to estimate parameters of a weak perspective camera $\mathcal{D}_{camera}$. The hand joints, obtained by taking the average of the surrounding ring vertices in the generated mesh, are projected onto the image plane to compute the loss based on 2D keypoint annotations. More specifically, we minimize the loss:

$$\begin{aligned}\mathcal{L} = &\sum_i |\hat{\mathcal{Y}}_i - \mathcal{D}_{GMM}(E_{image}(X_i))|_1 \\ &+ \lambda_{kpts} \sum_i |\hat{\mathcal{J}}_i - \mathcal{D}_{camera}(E_{image}(X_i))|_1 \\ &+ \lambda_{embed} \sum_i ||E_{image}(X_i)_{1:Z}||_2\end{aligned} \tag{6.1}$$

for ground truth meshes $\hat{\mathcal{Y}}_i$, keypoint annotations $\hat{\mathcal{J}}_i$, and input images X_i. We set hyperparameters λ_{kpts} to 0.01 and λ_{embed} to 5e-5. We also add L2 regularization weighted by 1e-5.

During training, we freeze the weights of the GMM and train the image encoder and camera regressor simultaneously for 130 epochs with the same hardware setting and optimizer as the autoencoder but we set learning rate to 1e-4. The network is able to reproduce hand pose early in the training while longer optimization reduces noise around fingertips for extreme poses. We zero the gradient that flows from the camera regression module through $\mathcal{D}_{GMM}$ as we find it to provide better reconstruction results.

	Spectral, fixed	Spectral, fine-tuned	MANO-like
Reconstruction error [mm]	2.33	**2.30**	2.56
Inference time (generator) [ms]	-	**3.04**	4.64
Inference time (generator) [fps]	-	**329**	216
Number of params. (generator)	**393,080**	**393,080**	2,498,612

Table 6.1: GMM system performance. Mesh L1 reconstruction error, inference time, and number of parameters for systems with different types of generators evaluated on Panoptic DomeDb. The scale of the target meshes is smaller than in real world.

6.1.1 Evaluation

When we first proposed the GMM system, datasets with 3D hand mesh annotations did not exist. We need the dataset where meshes are aligned in canonical coordinates. To collect the dataset, we fit the high resolution parametric hand model (Chapter 5) to 3D keypoints annotations from the Panoptic dome dataset [SJMS17]. This dataset provides 3D annotations of the whole human body including 21 hand joints. It contains multiple sequences of videos with a large number of subjects captured from multiple views in a laboratory setting. We use a subset of around 30 synchronized HD cameras and sample best annotations from a single camera in terms of visibility after each 100th frame. Our preprocessed dataset contains 40360 training, 3000 validation, and 3000 test samples with 3D keypoint annotations projected to the image coordinates.

We implement a baseline method that replaces a spectral decoder from Figure 6.1 with a TensorFlow implementation of our high resolution MANO model (MANO-like). We also fine-tune the network (Spectral, fine-tuned) by training our network (Spectral, fixed) for 170 more iterations without freezing the decoder's weights. Table 6.1 shows that we are able to obtain a lower reconstruction error despite the fact that the MANO model was used to generate the ground truth data in the first place. The table also presents that the spectral decoder is six times smaller in terms of number of parameters, obtains lower inference time, and operates in

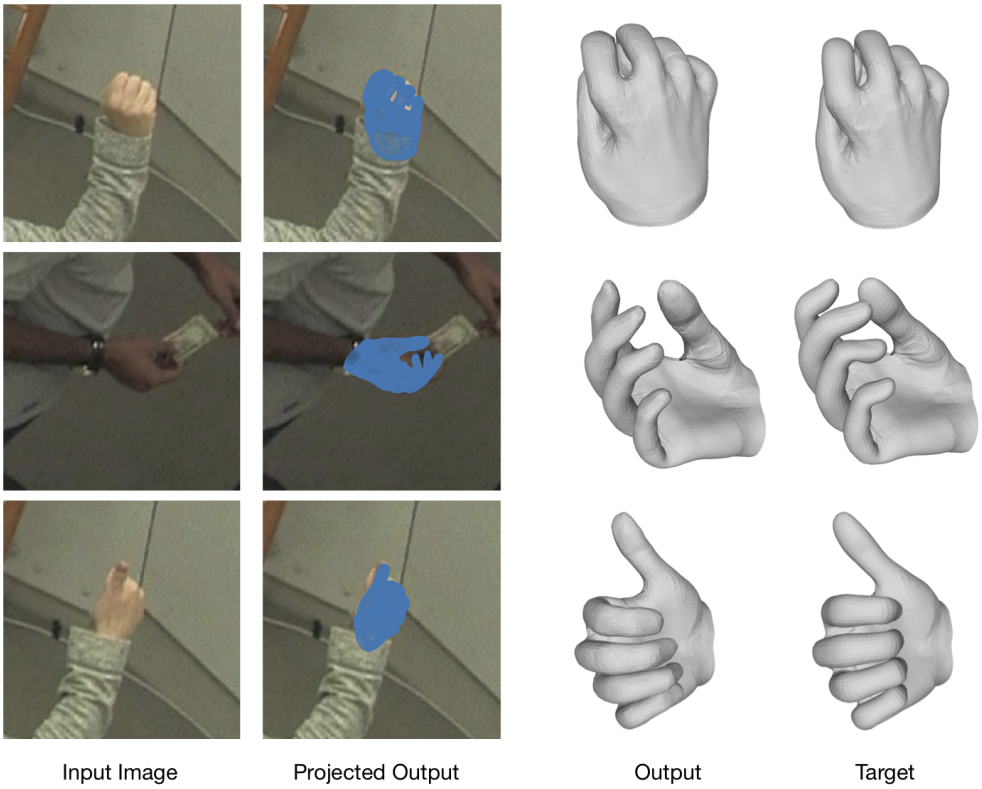


Figure 6.2: Qualitative results of the GMM system.

real-time. The DenseNet-121 image encoder runs at 53 FPS, therefore, the whole system maintains real-time performance. The visualization of the results of the mesh recovery system is presented in Figure 6.2.

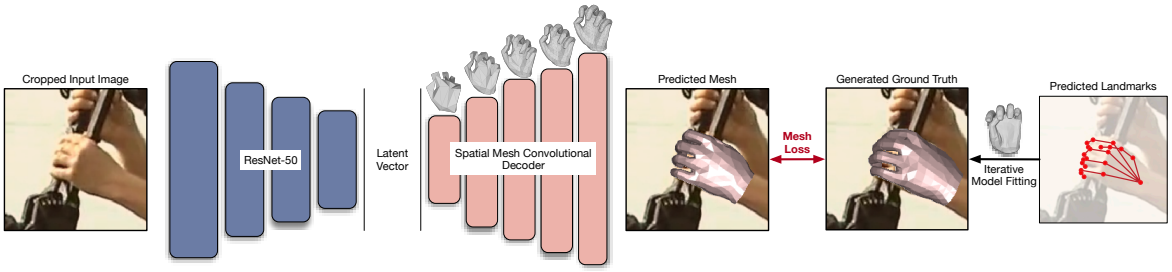


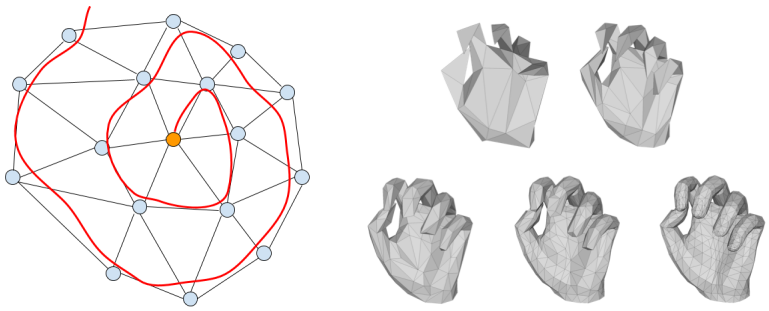
Figure 6.3: We propose an approach for end-to-end neural network training with mesh supervision that is obtained through an automated data collection method. We process a large collection of YouTube videos and analyze them with 2D hand keypoint detector followed by parametric model fitting (right side). The fitting results are used as a supervisory signal ('mesh loss') for a feed-forward network with a mesh convolutional decoder tasked with recovering a 3D hand mesh at its output (left side).

6.2 Direct Regression

We introduce further advances to the mesh regression system presented in the previous section that dramatically improves its accuracy and robustness. We propose a simple encoder-decoder system, as demonstrated in Figure 6.3, that directly reconstructs the mesh in image coordinates and works in non-laboratory environments. Spectral methods that we used for mesh convolutions are isotropic due to rotation invariance of the graph Laplacian which might prevent the network from modelling fine-details of the mesh surface. To address this limitation, we use a spatial convolutional mesh decoder $\mathcal{D}_{mesh}$, that imposes canonical ordering of neighbours to localize the kernels which we experimentally have shown to be superior with respect to alternative decoding strategies.

6.2.1 Spiral Operator

In our decoder we use the spiral patch operator for constructing spatial neighborhoods. Lim et al. [LDCK18] define a spiral selection of neighbours for a center vertex



a) b)

Figure 6.4: a) Spiral selection of neighbours for a center vertex in clockwise order (taken from [BBP+19]). b) The hierarchy of topologies used in our decoder.

v by imposing the order on elements of the k-disk obtained in the following way:

$$\begin{aligned} 0\text{-ring}(v) &= \{v\}, \\ (k+1)\text{-ring}(v) &= N(k\text{-ring}(v)) \setminus k\text{-disk}(v), \\ k\text{-disk}(v) &= \cup_{i=0\ldots k}\, i\text{-ring}(v), \end{aligned}$$

where $N(V)$ is the set of all vertices adjacent to any vertex in the set V. The spiral patch operator [BBP+19] is then the ordered sequence $S(v)$

$$\mathrm{S}(v) = (v, 1\text{-ring}(v), \ldots, k\text{-ring}(v)).$$

As orientation, we follow the clockwise direction from each vertex and randomly initialize the first neighbour of v. An example spatial configuration is demonstrated in Figure 6.4. The spiral convolution of features $f(S(v))$ with the kernel g_ℓ can then be defined by

$$(f * g)_v = \sum_{\ell=1}^{L} g_\ell \, f\big(S_\ell(v)\big), \tag{6.2}$$

where the spiral length L is fixed for each vertex in the same layer of a neural network.

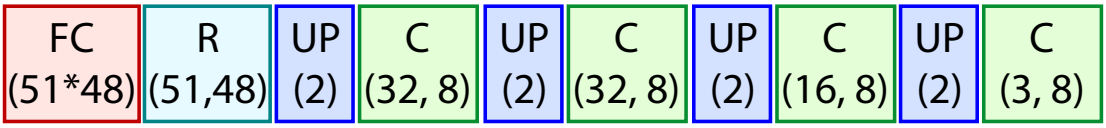


Figure 6.5: Architecture of the spiral decoder. $C(w, l)$: A spiral convolution with w filters and a spiral length l; $U(p)$: upsampling by a factor p; $FC(d)$: a fully connected layer with d output dimension; $R(d_1, d_2)$: vector reshaping into a $d_1 \times d_2$ matrix.

6.2.2 Architecture

Given an image crop X centered around the hand, we embed it into a latent vector $Z = E_{image}(X)$ with 64 parameters. The decoder, $\mathcal{D}$, takes the embedding as input and produces a mesh $\mathcal{Y} = \mathcal{D}_{mesh}(Z)$. We use a standard ResNet-50 network as the encoder, E_{image}. The architecture of the spiral decoder is detailed in Figure 6.5. If the spiral sequence is shorter than required due to erroneous triangulation or mesh border, we pad it with a node initially centered at 0. We only consider k-disks for $k = 2$. We choose leaky ReLU for the activation function based on experimental evaluation. The resulting topology hierarchy obtained using the sampling strategy is shown in Figure 6.4. The number of vertices at each layer $n \in \{51, 100, 197, 392, 778\}$.

6.2.3 Training

The loss function consists of the L1 vertex reconstruction term and edge length preservation component

$$\begin{aligned} \mathcal{L} &= \lambda_{vertex} |\hat{\mathcal{Y}} - \mathcal{Y}|_1 \\ &+ \lambda_{edge} \sum_{(u,v) \in \mathcal{E}_{mesh}} | \, ||\hat{\mathcal{Y}}_v - \hat{\mathcal{Y}}_u|| - ||\mathcal{Y}_v - \mathcal{Y}_u|| \, | \end{aligned} \tag{6.3}$$

for ground truth meshes $\hat{\mathcal{Y}}$ and set of edges $\mathcal{E}_{mesh}$. Hyperparameters are set to $\lambda_{vertex} = 0.01$ and $\lambda_{edge} = 0.01$. There is no explicit pose estimation loss because we found it to have no effect. The joint coordinates are obtained during evaluation

from a mesh as in Equation 4.3.

The network is trained with the Adam optimizer with learning rate 10^{-4} for 150 epochs. Learning rate decay with factor 0.1 occurs after 90-th and 120-th epoch. The images are normalized with the mean and standard deviation computed from the ImageNet training set and output meshes are normalized based on statistics computed from the subset of our training data. We augment the data with random image crops and transformations. The data augmentation is necessary for generalization to real world examples where the input images cannot be cropped based on the ground truth annotations. We train the network for 2 days on a single GeForce RTX 2080 Ti with a batch size 32 and crop size of 192×192.

6.2.4 Evaluation

Mesh reconstruction methods have started gaining popularity only recently and therefore there are no well-established benchmarks for hand recovery. To show the robustness of our method, we evaluate it on popular hand pose estimation datasets. Moreover, we show the hand reconstruction performance on the `FreiHAND` dataset and conduct a self-comparison study of different mesh decoders on our `YouTube` dataset. Lastly, we show that both major contributions, neural network architecture and data annotation system, result in the state-of-the-art performance in the absence of one or other.

Datasets For each of the datasets, beside `FreiHAND` that provides vertex annotations (Section 2.4), we fit the MANO model following the optimization approach described in Chapter 4. In case of 3D annotations, we only change the intrinsic camera to the identity function in Equation 4.5 and then we project the output mesh based on dataset camera parameters. The depth is reconstructed as before.

	Ours	MANO Fit [ZCY+19]	MANO CNN [ZCY+19]	Boukh. et al. [BBT19]	Hasson et al. [HVT+19a]	Mean Shape [ZCY+19]
Pose Error [mm] ↓	**0.84**	1.37	1.1	3.5	1.33	1.71
Mesh Error [mm] ↓	**0.86**	1.37	1.09	1.32	1.33	1.64
Pose AUC ↑	**0.834**	0.73	0.783	0.351	0.737	0.662
Mesh AUC ↑	**0.83**	0.729	0.783	0.738	0.736	0.674
F@5 [mm] ↑	**0.614**	0.439	0.516	0.427	0.429	0.336
F@15 [mm] ↑	**0.966**	0.892	0.934	0.894	0.907	0.837

Table 6.2: 3D pose estimation and mesh reconstruction performance on FreiHAND. For the first two rows lower is better, for the other rows higher is better.

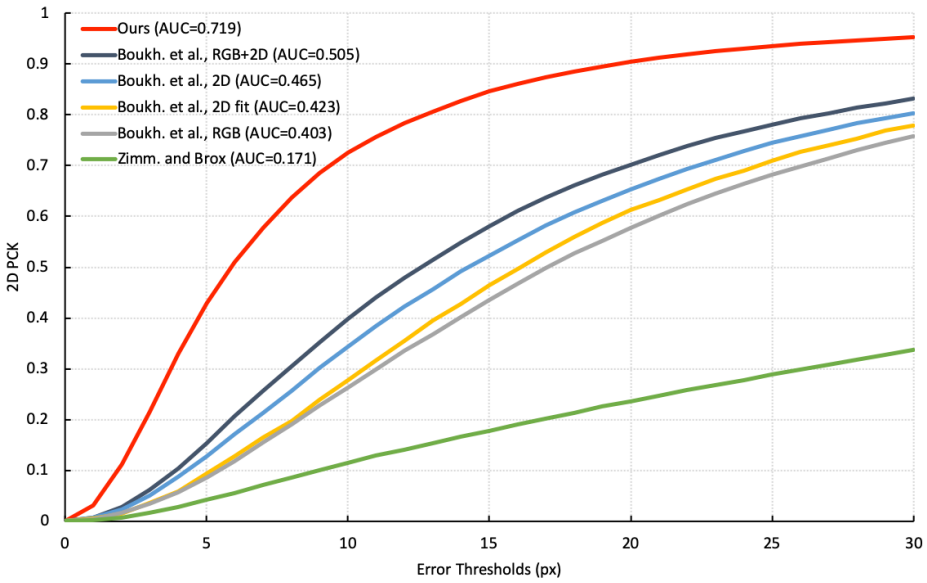
Metrics To evaluate hand pose estimation and mesh reconstruction performance, we measure the average euclidean distance (Pose/Mesh Error), the percentage of correct points (2D/3D PCK) for different thresholds, and the Area Under Curve (AUC) for PCK. For 3D benchmarks, we measure the error after rigid alignment. Before computing a 2D error, we orthographically project the estimated 3D pose. For self-comparison study, we report the mean absolute error (MAE) in addition to Mesh Error.

Hand Pose Estimation Figure 6.6 shows evaluation results on the MPII dataset. Our system significantly outperforms other approaches, halving the Pose Error of the leading MANO-based method and improving AUC by 0.21 points.

Figure 6.7 shows 3D pose estimation results on RHD. This is a very popular benchmark on which we also outcompete existing methods by a significant margin.

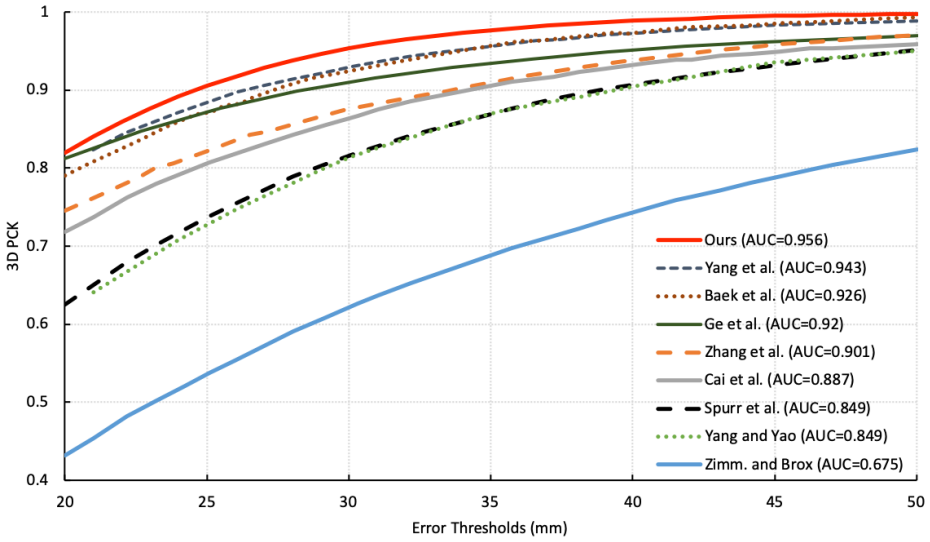
The proposed system also achieves a superior performance on the FreiHAND dataset as proved in Table 6.2. Notably, most of the competing methods in this benchmark rely on MANO parameters regression.

Mesh Reconstruction We evaluate the quality of mesh reconstructions on the FreiHAND dataset (Table 6.2). In addition to the standard metrics, we report the F-score at a given threshold d ($F@d$) which is the harmonic mean of precision and recall [KPZK17]. Qualitative results on the in the wild datasets can be found in Figure 6.8 and the comparison with a previous state-of-the-art is shown in Figure 6.9.



	Ours	Boukh. et al. RGB+2D [BBT19]	Boukh. et al. RGB [BBT19]	Boukh. et al. 2D Fit [BBT19]	Boukh. et al. RGB [BBT19]	Z&B [ZB17]
Pose Error [px]	**9.27**	18.95	22.36	20.65	23.04	59.4

Figure 6.6: 2D PCK and average 2D joint distance (px) to ground-truth for `MPII`



	Ours	Yang et al.[YLLY19]	Spurr et al. [SSPH18]	Y&Y [YY19]	Z&B [ZB17]
Pose Error [mm]	**10.92**	13.14	19.73	19.95	30.42

Figure 6.7: 3D PCK and average 3D joint distance (mm) to ground-truth for `RHD`.



Figure 6.8: Qualitative mesh reconstruction results shown on `YouTube` (left half) and `MPII` (right half).

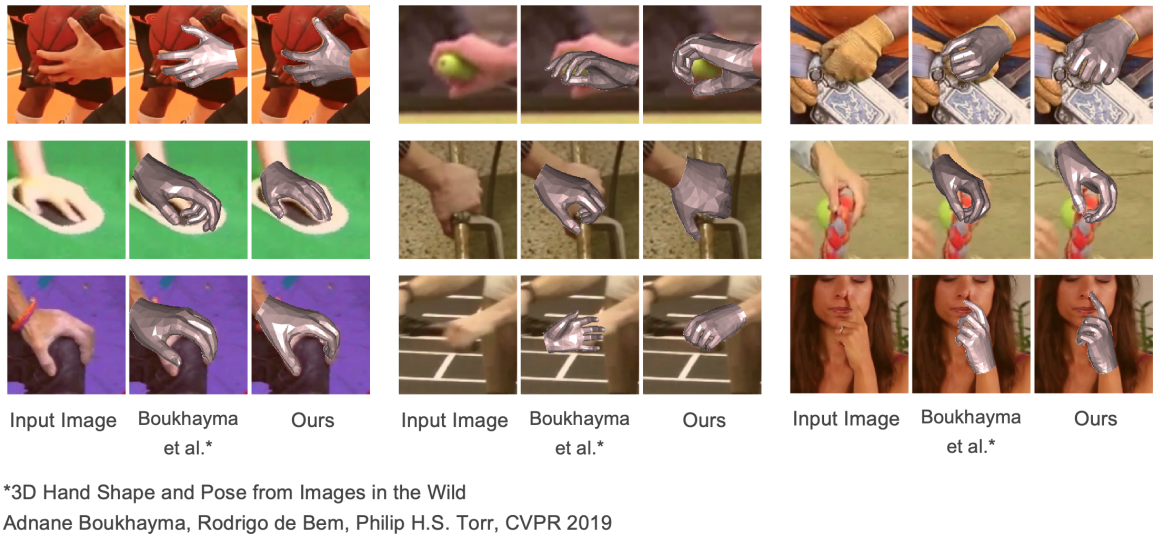


Figure 6.9: Qualitative performance comparison with state-of-the-art for in the wild hand reconstruction. Our system is two times more accurate; the difference is especially visible on blurry images, hands wearing gloves, saturated images, and hands interacting with objects.

Self-Comparison Table 6.3 and Table 6.4 show the pose estimation and mesh reconstruction error on different types of mesh decoders. To perform broad hyperparameter search that ensures fair comparison of the baseline methods, we use a reduced training dataset, corresponding to row 6 in Table 6.6, to speed-up optimization.

`Spiral` refers to the proposed approach, `Spectral` follows the GMM architecture but with no camera estimation branch. `Spiral GMM` is the spiral implementation of GMM which incorporates decoder pretraining that is fixed afterwards and a camera estimation branch. `Spiral GMM, tuned` is a fine-tuned version obtained by resuming training of all parameters of the `Spiral GMM`. Finally, `MANO` is similar to Boukhayma et al. [BBT19] but with the loss function on mesh vertices instead of joints. We observe that our edge length preservation loss term stabilizes `MANO` shape parameters regression that in prior works was addressed by imposing a large L2 penalty [BBT19] to mitigate divergence.

We find that a spatial method outperforms a spectral approach. Moreover, end-

Decoder Type	RHD [mm]	MPII [px]	YouTube [px]
MANO	12.086	11.448	14.958
Spectral	11.638	9.858	13.612
Spiral GMM	11.138	10.074	11.999
Spiral GMM, tune	11.121	10.117	11.799
Spiral	**11.052**	**9.434**	**10.698**

Table 6.3: Pose Error. Performance with different decoders trained on a reduced dataset.

Decoder Type	MAE [px]	Mesh Error [px]
MANO	13.023	27.485
Spectral	9.058	19.317
Spiral GMM	8.462	18.206
Spiral GMM, tune	8.380	18.057
Spiral	**7.827**	**17.096**

Table 6.4: Mesh reconstruction with different decoders on the `YouTube` dataset trained on a reduced dataset.

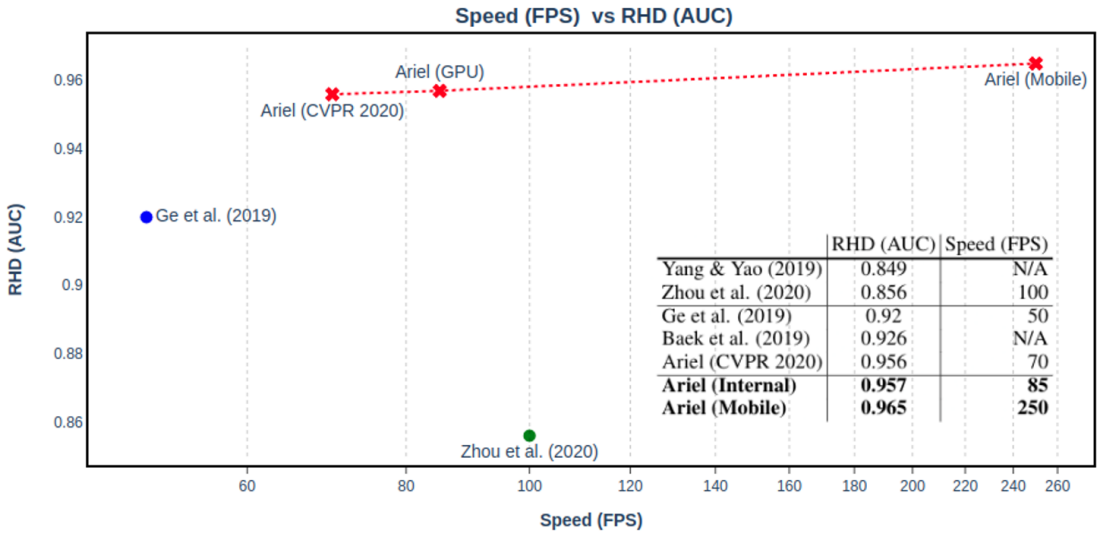
to-end training with loss on meshes in the image coordinate system is better than pretraining in the canonical frame and estimating camera parameters.

Comparison With Iterative Fitting Our system is supervised with meshes iteratively aligned with keypoint predictions. Naturally the question arises whether the network performance is bounded by the performance of iterative fitting. We compare both approaches on the only existing manually annotated in the wild dataset. Additionally, we select 390 out of 846 samples that were filtered according to sanity checks described in Chapter 4.

Based on Table 6.5 we observe that our system performs slightly worse to iterative fitting on a filtered dataset. This is expected as these are samples on which the

Model	MPII [px]	MPII Filtered [px]
Iterative Fitting	10.295	**5.111**
Network	**9.265**	5.555

Table 6.5: Comparison of the network performance with iterative fitting on the `MPII` test set in terms of Pose Error.



	RHD (AUC)	Speed (FPS)
Yang & Yao (2019)	0.849	N/A
Zhou et al. (2020)	0.856	100
Ge et al. (2019)	0.92	50
Baek et al. (2019)	0.926	N/A
Ariel (CVPR 2020)	0.956	70
Ariel (Internal)	**0.957**	**85**
Ariel (Mobile)	**0.965**	**250**

Figure 6.10: Our system, denoted as Ariel, is both faster and more accurate than existing approaches.

keypoint estimator has very high confidence. Importantly, the network performs much better when the whole dataset is considered.

Inference Time The network is also characterized by the strikingly faster inference time. Specifically, the reported timing (Chapter 4) for iterative fitting of 4000 samples / 10 minutes is with a batch size of 4000. A single sample takes 110 seconds (GPU) or 70 seconds (CPU) for fitting to 2D annotations, exluding keypoint predictions. By contrast, our network inference time is 85 FPS (GeForce RTX 2080 Ti).

Most importantly, after optimization for faster inference on mobile devices, our method facilitates real-time applications on mobile devices at 250 FPS (Apple A12/A13 Bionic) while maintaining superior performance (Figure 6.10).

Dataset and Method Ablation In Table 6.6, we evaluate the system trained on different combinations of datasets. By comparing rows 1-3, we observe that training a mesh reconstruction system on our dataset leads to better results than training on

ID	MPII	RHD	MTC	FreiHAND	YouTube	RHD [mm]	MPII [px]	YouTube [px]
1	X					15.398	14.930	35.032
2				X		13.258	16.558	23.409
3					X	12.448	10.572	11.052
4	X	X				11.205	11.547	18.539
5	X	X	X			11.202	10.965	16.214
6	X	X	X		X	11.052	9.434	10.698
7	X	X	X	X		10.926	9.672	12.637
8	X	X	X	X	X	10.922	9.292	10.536

Table 6.6: Pose Error. Performance on different combinations of datasets.

other outdoor datasets. Rows 5-8, show that the best results are obtained when our dataset is included during training, even when adding it to the union of all other datasets. We also observe state-of-the-art performance on `RHD` and `MPII` without recent datasets such as `YouTube`, `FreiHAND`, `MTC` proving the efficiency of the proposed neural network (Row 4). Similarly, Table 6.3 shows that the MANO-based method trained with the novel dataset and mesh loss obtains better performance than prior MANO-based implementations indicating the importance of the contributed data collection system.



Figure 6.11: The system shows impressive results on egocentric images enabling potential applications for mobile and wearable devices.



Figure 6.12: A London-based startup Ariel AI developed augmented reality mobile applications that incorporate the proposed system. On the left, the thesis author records himself on an entry level iPhone (SE) running a neural network capable of processing images at 250 FPS. On the right, a user tests the application for hand reconstruction that allows applying different visual filters and drawing with an index finger.

6.3 Conclusion

We have introduced a system for generating a subject-specific hand model from a single image. We have shown that a simple encoder-decoder architecture can obtain superior performance on mesh reconstruction and pose estimation tasks given an appropriately defined objective function. Furthermore, we have proposed an approach for weak supervision that allows adapting the system to non-laboratory domains and additionally improves results on common benchmarks.

The system works in real time on mobile devices enabling wide range of industrial applications and generalizes well to egocentric images supporting wearable devices (Figure 6.11). A British company Ariel AI has incorporated this system to offer cutting-edge augmented reality applications (Figure 6.12).

We believe that our findings justify looking further into mesh generative models for human modeling, as well as other weakly- and self-supervised methods that alleviate the need of acquiring 3D ground truth for humans in unconstrained environments.

Chapter 7

Large-Scale Hand Scans Collection

Three-dimensional scans of the human body enabled groundbreaking applications in human body understanding and modelling. Statistical deformable models constructed from such datasets became a method of choice for human pose estimation systems that work by regressing their parameters from images. These models also found applications in constructing training datasets for neural networks. In Chapter 6 we showed how a deformable model of the human hand can be applied as a source of supervision for a neural network tasked at recovering a 3D hand from a single image captured in a non-laboratory setting. While our method obtains a cutting edge performance and works on mobile devices, the reconstruction results are not pixel accurate because the existing deformable models are unable to capture a wide variety of shape and pose deformations. Moreover, the fitting approach that we presented in Chapter 4 relies on minimizing a 2D reprojection error to a sparse set of keypoints. This could be improved if we were able to provide dense annotations or synthesise a skin texture.

Until recently, there was only a single dataset of 3D scans available that consists of 2,018 scans of 31 subjects [RTB17]. The number of subjects is too small to represent the whole population of human beings. Moreover, the pose space is limited and the



Figure 7.1: Data collection project in Science Museum in London.

poses are discrete. The lack of temporal sequences makes it impossible to synthesize the human motion or learn detailed soft-tissue deformations. Another problem is the noise in texture reconstruction which does not allow constructing statistical models of the human skin that - beside applications in computer graphics - could enable self-supervised pose estimation and synthetic training data generation. Two new datasets that were introduced this year include scans of only 10 and 48 subjects (Table 7.1).

In this chapter we describe an enormous effort to collect the dataset of high-resolution 3D hand motion sequences of people visiting Science Museum in London, which is also a popular tourist destination, in the period of two months in December 2019 - January 2020. This is part of a larger project that also involved collection of the human body reconstructions in a separate scanner; we omit that part in a further discussion because it was performed by other researchers.

Our focus is to address the major limitations of the existing statistical deformable models - that we identified based on body, face, and hand related research - and thus we want to achieve the following goals:

- Collect diverse collection of hand shapes in terms of age and ethnicity to rep-

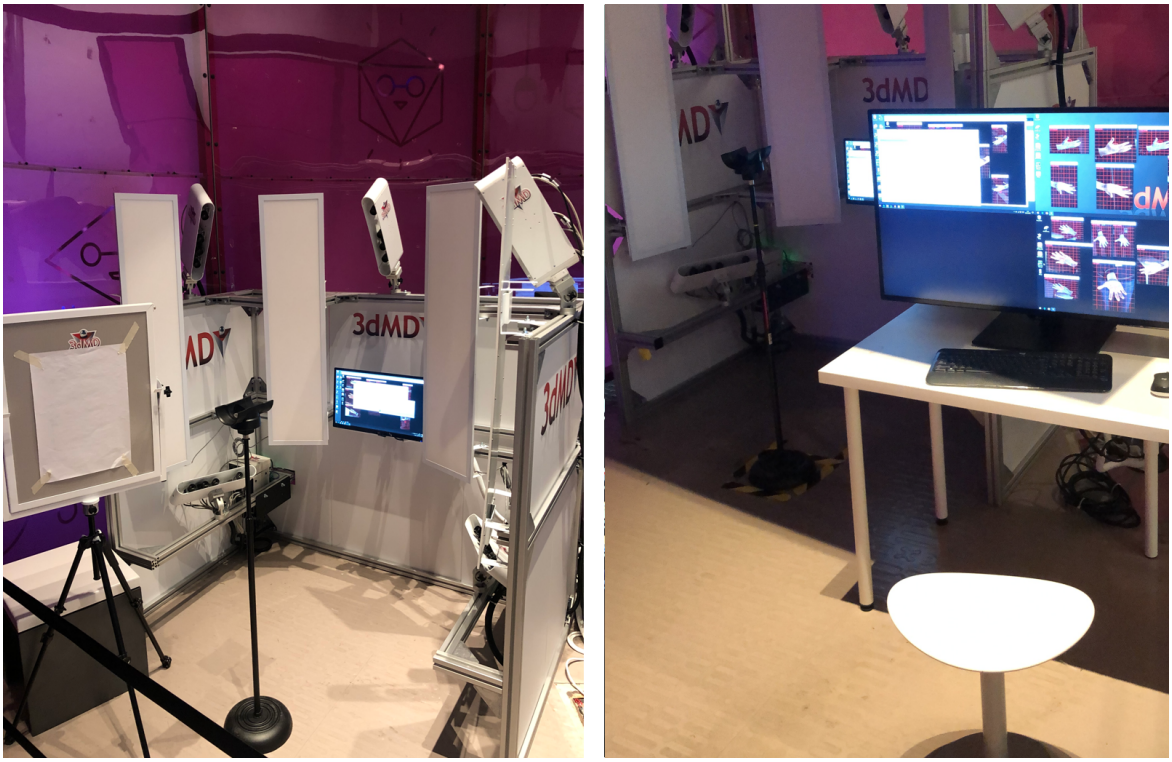


Figure 7.2: Hand scanner.

resent everyone in the population.

- Reconstruct high-resolution hand textures to enable novel applications that were not possible due to lack of such datasets.
- Collect temporal sequences to support human motion synthesis methods.
- Capture diverse poses to build a robust prior on joint-angle rotations.
- Cover poses from sign language alphabets and common types of grasps.
- Capture poses with visible soft-tissue deformations to enable modelling of the pose-dependent surface details.

The result is a dataset of 533,687 scans captured at 30 FPS from 1,360 subjects.

We describe the hand reconstruction system, the data collection procedure, the choice of hand motions, the quality and diversity of the dataset, and compare it



Figure 7.3: Participant experience.

with existing datasets. In Chapter 8 we discus how we processed this dataset and created a deformable hand model. Moreover, in Chapter 9 we describe how we transferred the textures from collected scans and learned texture variations.

7.1 Hand Reconstruction System

We collect the data using the 3dMDhand System (3dMD LLC, Atlanta) presented in Figure 7.2. A subject places a wrist on the hand stand and performs a selected protocol of articulated motions. The system is composed of five camera units where each unit is composed of: a color camera (resolution 2448 x 2048), two grayscale cameras (1624 x 1236), and two speckle pattern projectors. A typical scan of an adult hand consists of 35K vertices (Figure 7.4).

7.2 Data Collection Protocol

Prior to the start of experiment, each participant is given an information sheet and a consent form to sign (Figure 7.3). Participants also complete a questionnaire. Children have they forms filled by a parent. The experiment is compliant with General

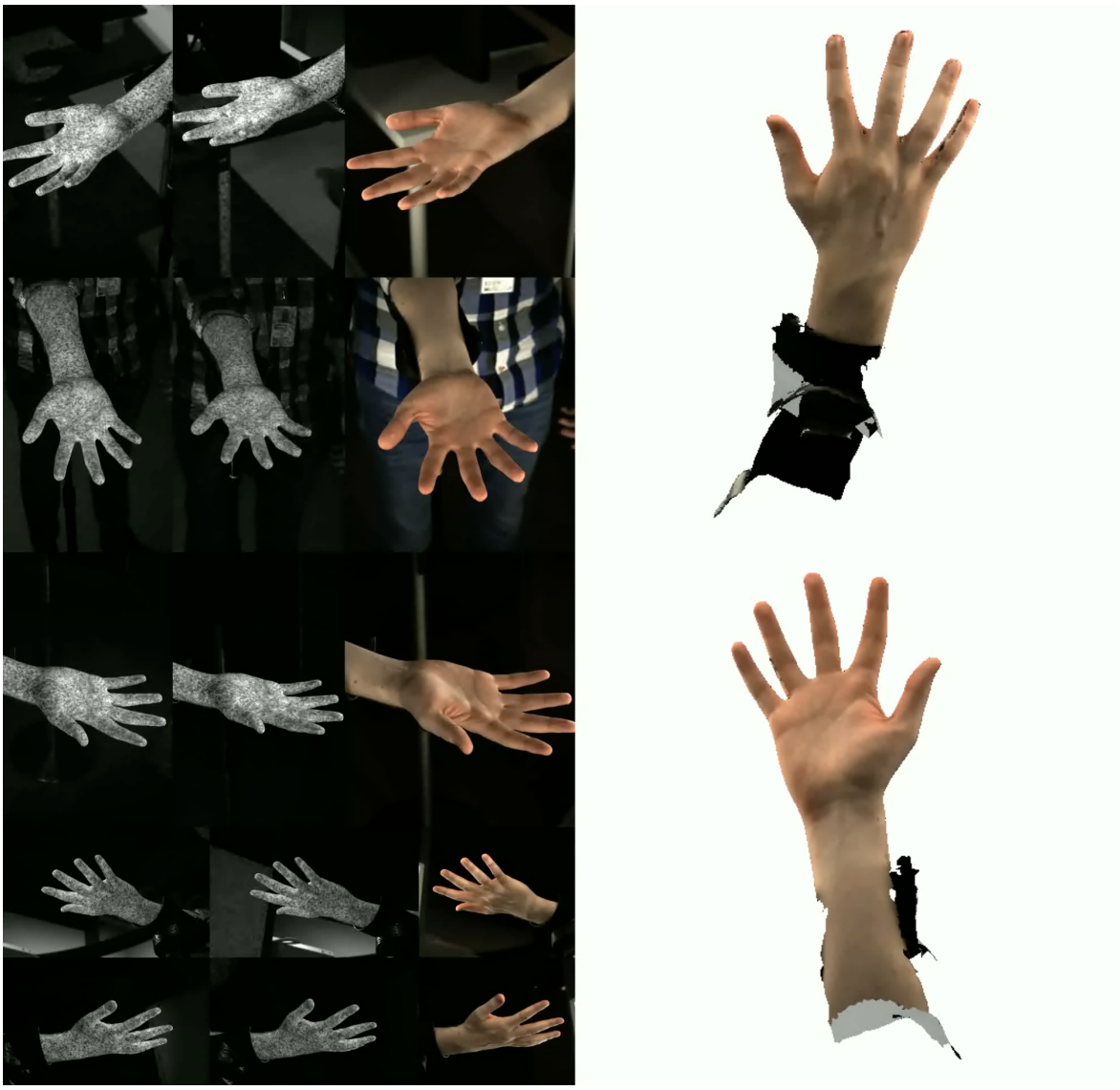


Figure 7.4: Camera views and renders.

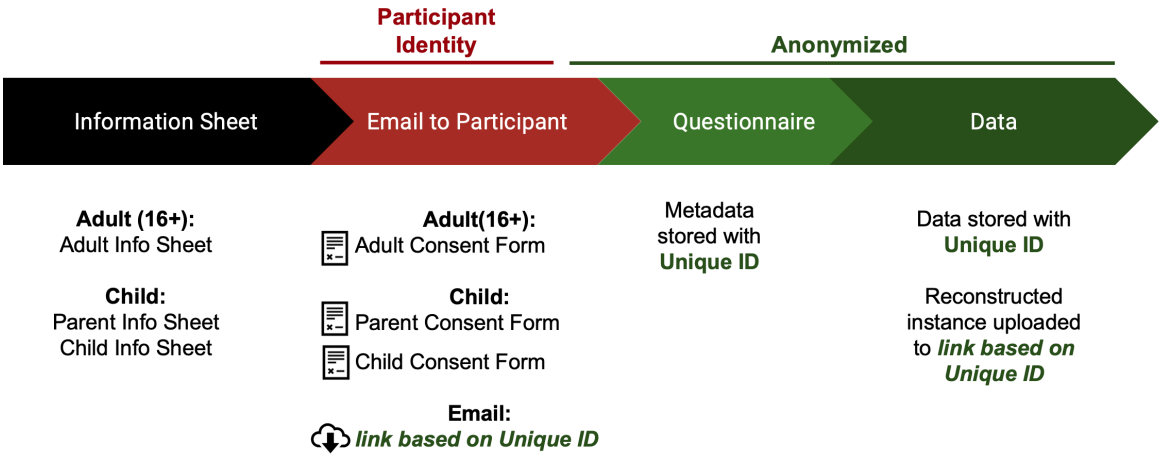


Figure 7.5: Consent form.

Data Protection Regulations (GDPR) and documents have been approved by Imperial College Ethics Committee and Contracts and Personal Information Protection Officer. All data is anoymized and stored under a unique identification number (Figure 7.5).

The questionnaire is designed to collect the following metadata: age, gender, height, weight, injuries/operations, ethnicity (White, Black / Black British, Mixed Heritage, Asian / Asian British, Other [with the option to specify]). Hand scans collection is the second stage after the full-body capture so some of the questions are more relevant to that part of the experiment.

After completing the questionnaire, a subject takes part in a body recording and when this is finished they proceed to the hand scanning experiment. We have designed 10 protocols that are on average 10 seconds long to capture a wide variety of poses. A participant is asked to perform one of the protocols that we assigned for a given day. A person operating the scanner stands next to the subject demonstrating the protocol during recording. Protocols are designed to cover joint angle constraints, common gestures, different types of grasps, sign language alphabet, and challenging poses on which existing pose estimation systems fail (Figure 7.6). The system captures motions at 30 FPS so the transitions between gestures are also recorded.



Figure 7.6: Selection of gestures from hand scanning protocols.

Features	MeDigital Hands	MANO	Hand 3D Studio [ZWXW20]	GHUM [XBZ+20]
No. Subjects	1,360	31	10	48
No. Scans	533,687	2,018	1,540	?

Table 7.1: Data comparison. Additional unique features of our dataset (MeDigital Hands) include: presence of jewellery, medical skin and bones conditions, continuous motions instead of discrete gestures.

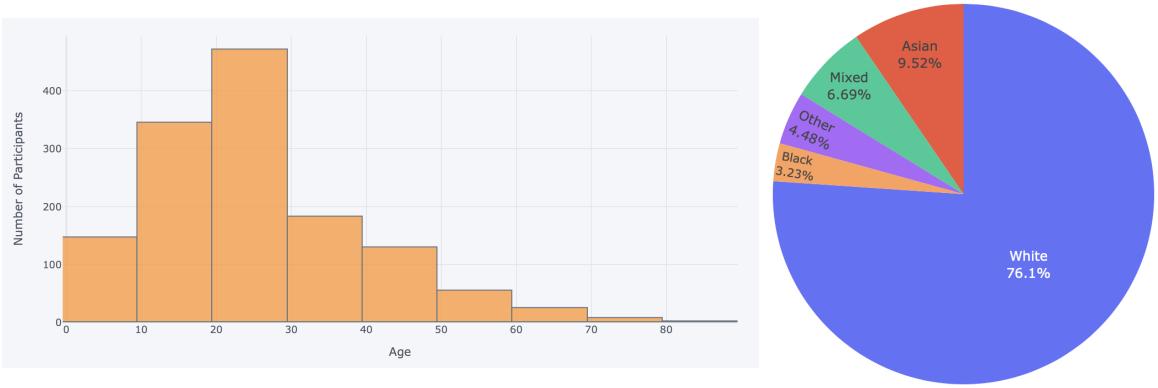


Figure 7.7: Age and ethnic diversity of participants.

Upon completion, the subject observes a surface reconstruction visualizations for one of the frames and watches the output 3D scan. The generated scan is sent to an email address provided during a sign up.

7.3 Data Quality

We collected 533,687 scans from 1,360 subjects (53% male, and 47% female) in the age 3 - 81 years old (Figure 7.7). The second largest existing dataset is captured from 48 subjects (Table 7.1). In terms of ethnic diversity, the dataset includes 325 subjects representing Black, Asian and Minority Ethnic (BAME) groups (Figure 7.7).

Additionally our dataset is characterized by unique features such as high-quality textures (Figure 7.8), presence of jewellery, medical skin and bones conditions, continuous motions instead of discrete gestures. Examples of reconstructions are shown in Figure 7.9.

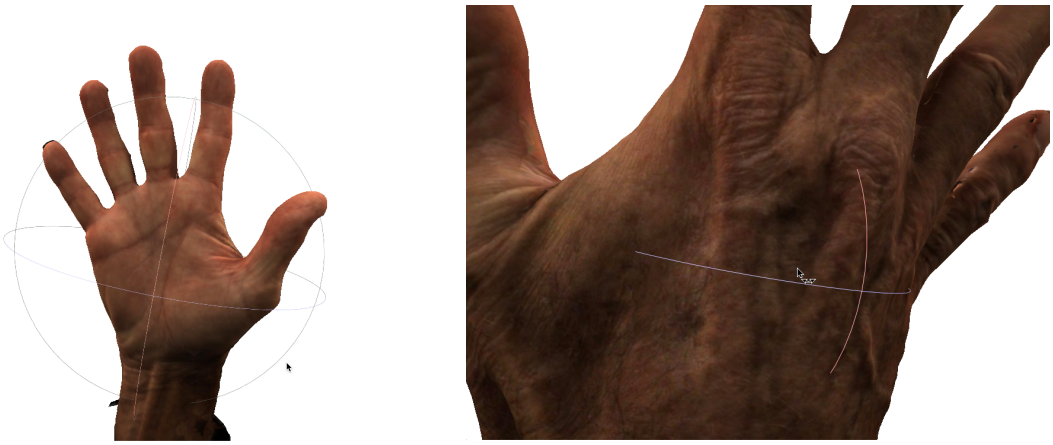


Figure 7.8: Texture details.

7.4 Conclusion

We have collected the largest, by two orders of magnitude, and most diverse dataset of 3D hand scans. Such dataset has potential for advancing research on 3D hand reconstruction and enabling novel applications. Broad age group and ethnic diversity allows building statistical deformable models that represent the significant portion of the human population. High resolution textures can enable self-supervised training of pose estimation systems and generation of realistic synthetic data.

In the remaining chapters we describe the process arriving at a state-of-the-art statistical hand model with texture variations learned from the presented dataset.



Figure 7.9: Hand renders.

Chapter 8

Statistical Deformable Hand Model

The MANO model has become a method of choice in research on shape modelling and pose estimation of the human hand. Researchers have been proposing alternative models but they were merely iterative improvements to the standard formulation of shape modelling based on dimensionality reduction followed by linear blend skinning. Recently, a novel definition has been introduced for the human body modelling that disentangles bone length variations from other subject-specific shape deformations [WGK+20]. This is a crucial improvement over the existing frameworks for articulated shape modelling enabling higher control of template synthesis. Human body in SMPL, on which MANO is based, is generated from a combination of linear bases. Afterwards, the skeleton joints are estimated by regression from the generated mesh.

In contrast, the novel Bone-Level Skinned Model (BLSM) [WGK+20] disentangles bone length variability from other traits such as, for example, dietary habits or physical activity. The synthesis process starts with modelling bone length mesh variability followed by an identity shape update. This process results in more compact and expressive representation of the human body. Disentanglement of bone lengths and identity surface variations improves parameterization for the augmented reality

applications and the model formulation is fully integrated with standard graphics packages like Unity, Blender, and Maya. The BLSM does not include hand shape variability or pose prior on hand joints.

Inspired by this method, we learn an analogous model from the Science Museum data that we introduced in Chapter 7. The task goes far beyond an easy adaptation of the existing research method. Hand scans introduce challenges that cannot be solved in a plug and play manner. The first issue is related to the cost of data annotation. In order to register meshes, we need to fit a hand template into a scan what requires knowledge of at least the joint locations. Manual annotation of over 500,000 scans would be infeasible. We solve this problem by running a state-of-the-art hand pose estimator - that we proposed in Chapter 6 - on images of scans rendered from different views followed by triangulation. We find that the hand pose estimator underperforms if a hand texture is noisy due to self-occlusions. We retrain the system by adding data augmentation to simulate the noise. Furthermore, poses with self-occlusions lead to incomplete hand reconstructions what creates difficulties in registering the hand. The hand surface has also a boundary around the wrist area that can lead to distortions. To solve both issues, we design a custom registration method with strong regularization. The training process is bootstrapped by applying the initially trained model for a subsequent data fitting and retraining.

We evaluate the model by reproducing the YouTube dataset from Chapter 4 with our newly formulated system. We show that the new model leads to lower fitting error and the network from Chapter 6 trained on a novel dataset performs significantly better on the task of 3D hand reconstruction from a single image.

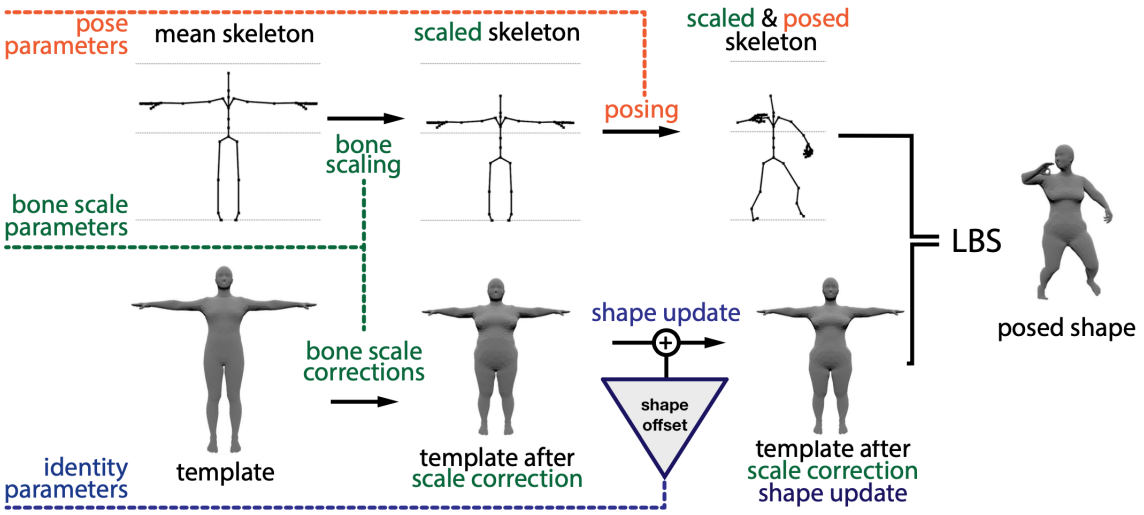


Figure 8.1: Overview of the Bone-Level Skinned Model (BLSM) reproduced from [WGK+20]. The top branch shows skeleton transformations while the lower branch describes the process of the template synthesis. Bone scale parameters are shared between branches. Both streams are combined to pose the updated template with linear blend skinning (LBS).

8.1 Bone-Level Skinned Model of the Human Hand (Hand BLSM)

The BLSM operates in two branches that perform bone transformations and surface update. The outputs are combined to articulate an updated template with linear blend skinning as shown in Figure 8.1. We extract the hand vertices and triangulation from the template to learn hand shape variations that are not trained in the original model.

In general, the Hand BLSM generates a mesh with N vertices using a differentiable function that maps shape parameters into an instance of the model:

$$M(\boldsymbol{c}_b, \boldsymbol{\beta}, \boldsymbol{\theta}) : \mathbb{R}^{|\boldsymbol{c}_b| \times |\boldsymbol{\beta}| \times |\boldsymbol{\theta}|} \rightarrow \mathbb{R}^{N \times 3}, \tag{8.1}$$

$$M(\boldsymbol{c}_b, \boldsymbol{\beta}, \boldsymbol{\theta}) = W(T(\boldsymbol{c}_b, \boldsymbol{\beta}), J(\boldsymbol{c}_b, \boldsymbol{\theta})), \tag{8.2}$$

where the hand skeleton $J(\boldsymbol{c}_b, \boldsymbol{\theta})$ is obtained through scaling coefficients $\boldsymbol{c}_b$ and joint angles of rotation $\boldsymbol{\theta}$. A linear blend skinning function $W()$ controls articulations. The point cloud $\boldsymbol{T} = T(\boldsymbol{c}_b, \boldsymbol{\beta})$ that represents the person template in a rest pose is obtained in an additive way from an average hand type $\bar{\boldsymbol{T}}$, shape correction term due to bone scaling artifacts $T_{BS}(\boldsymbol{c}_b)$, and an identity shape update $T_{ID}(\boldsymbol{\beta})$:

$$T(\boldsymbol{c}_b, \boldsymbol{\beta}) = \bar{\boldsymbol{T}} + T_{BS}(\boldsymbol{c}_b) + T_{ID}(\boldsymbol{\beta}). \tag{8.3}$$

The updates are controlled by bone scale parameters $\boldsymbol{c}_b$ and subject-specific shape parameters $\boldsymbol{\beta}$.

We describe each of the components in details in the following subsections.

8.1.1 Skeleton Transformations

Hand skeleton is modelled as a kinematic tree which nodes represent joints and edges form the bones. We are given a skeleton in its rest pose - also called a bind pose - where a single bone is expressed by a template rotation matrix $\boldsymbol{R}^t$ and translation vector $\boldsymbol{O}^t$ that indicate the transformations between coordinate systems at the two bone joints. The bone transformation $\boldsymbol{B}$ with respect to the bind pose is controlled by a rotation matrix $\boldsymbol{R}$ and a scaling factor s that allows an explicit control over the limb size:

$$\boldsymbol{B} = \underbrace{\begin{bmatrix} s\boldsymbol{I} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{R} & 0 \\ 0 & 1 \end{bmatrix}}_{scaling\ and\ articulation} \underbrace{\begin{bmatrix} \boldsymbol{R}^t & \boldsymbol{O}^t \\ 0 & 1 \end{bmatrix}}_{bind\ pose}. \tag{8.4}$$

The global transformation $\boldsymbol{B}_j$ of the joint j depends on local transformations $\boldsymbol{B}^i$ of the joint and all its parents in the kinematic structure: $\boldsymbol{B}_j = \boldsymbol{B}^1 ... \boldsymbol{B}^{parent(j)} \boldsymbol{B}^j$ where $\boldsymbol{B}_1$ corresponds to the wrist transformation.

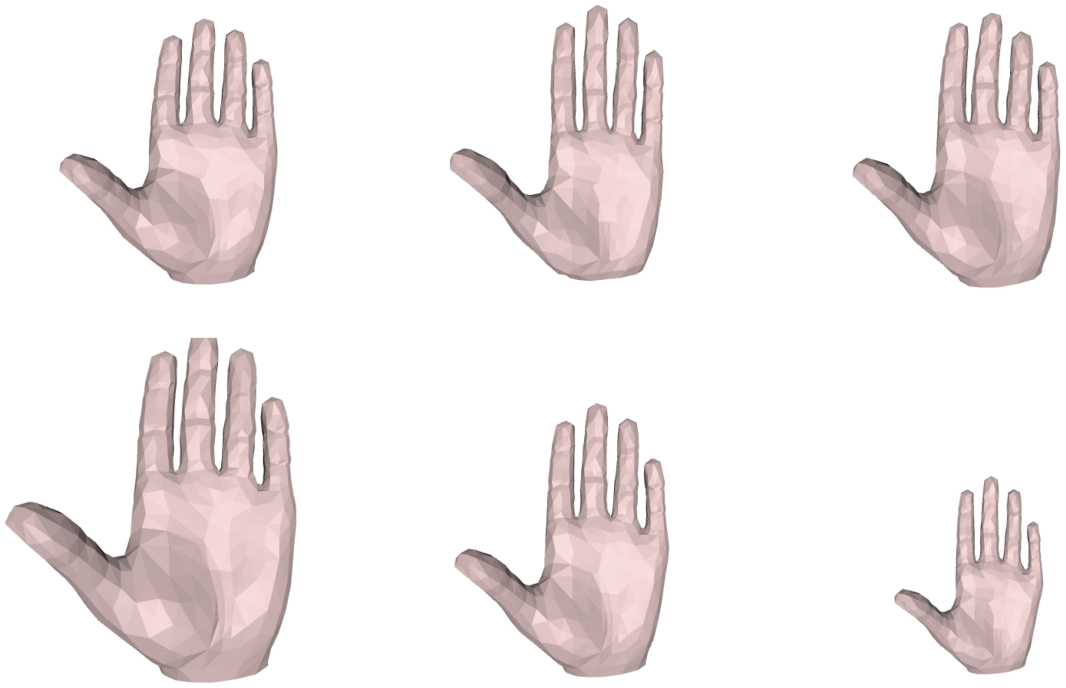


Figure 8.2: Samples of bone-dependent shape variations that model surface deformations due to bone lengths.

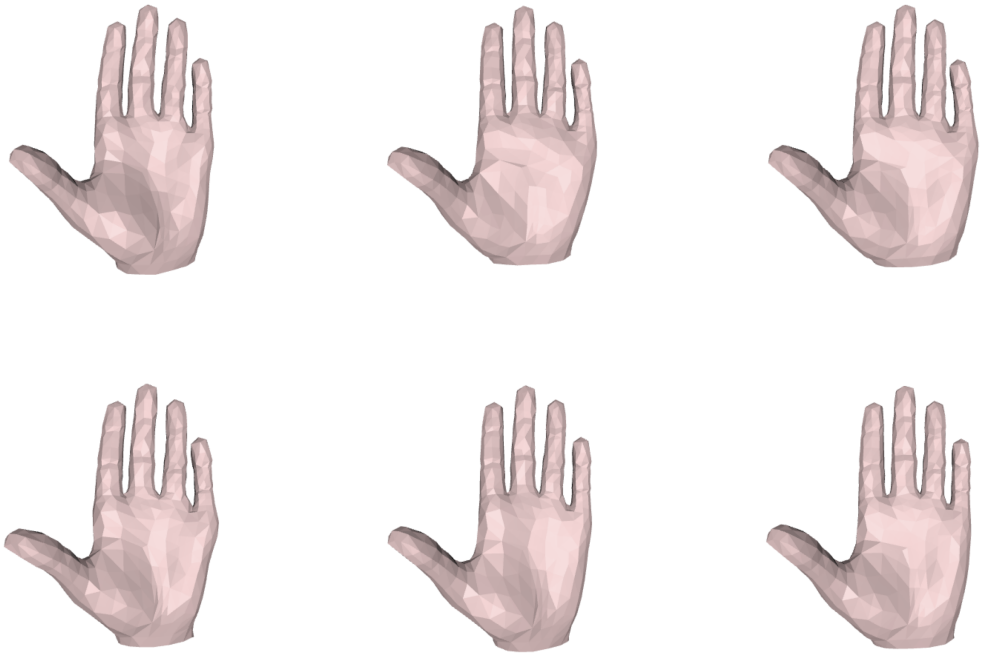


Figure 8.3: Samples of subject-specific shape variations that model surface deformations for fixed bone lengths.

8.1.2 Modelling Bone Length Variations

Individual bone scales are modeled as

$$\boldsymbol{b} = \bar{\boldsymbol{b}} + \boldsymbol{c}_b \boldsymbol{P}_b \tag{8.5}$$

where $\boldsymbol{c}_b$ are the bone scaling coefficients, $\boldsymbol{P}_b$ is the bone-scaling matrix obtained from performing PCA on bone lengths, and $\bar{\boldsymbol{b}}$ is the mean bone scale.

This equation gives us individual bone scales; however, in Equation 8.4 the global transformation is given by a kinematic chain recursion. We can transform the predictions of our PCA model to be compatible with Equation 8.4:

$$s_i = \begin{cases} \boldsymbol{b}_i / \boldsymbol{b}_{i-1}, & i > 0 \\ 1, & i = 0 \end{cases} \tag{8.6}$$

8.1.3 Modelling Shape Variations

Template synthesis is composed of two shape models. The first one models bone-dependent surface variations that account for a substantial part of hand shape variability that can correlate to person's age or gender. We represent these variations as a linear update to the mean template

$$\boldsymbol{T}_{BS}(\boldsymbol{c}_b) = \boldsymbol{c}_b \boldsymbol{P}_{bc} \tag{8.7}$$

where $\boldsymbol{P}_{bc}$ is the matrix of bone-corrective blendshapes. Similarly, the remainder of subject-specific shape variability is modeled as

$$\boldsymbol{T}_{ID}(\boldsymbol{\beta}) = \boldsymbol{\beta} \boldsymbol{P}_s. \tag{8.8}$$

where $\boldsymbol{\beta}$ are the shape coefficients and $\boldsymbol{P}_s$ is the matrix of shape components.

8.1.4 Posing

Given a synthesized template $T(\boldsymbol{c}_b, \boldsymbol{\beta})$ in a rest pose, we pose it with linear blend skinning (LBS) that we discussed in more details in Chapter 3.2.1. The bind pose of the skeleton is described by the matrices $\hat{\boldsymbol{B}}_j$, where the 3D mesh vertices take their canonical values $\boldsymbol{t}_i \in \boldsymbol{T}$, while the target pose is described by $\boldsymbol{B}_j$.

In LBS, each vertex has an associated scalar weight for each joint influencing the vertex deformation. The vertex is then transformed into the world space by linearly blending the bone transformation matrices:

$$\hat{\boldsymbol{t}}_i = \sum_j w_{ij} \boldsymbol{B}_j \hat{\boldsymbol{B}}_j{}^{-1} \boldsymbol{t}_i. \tag{8.9}$$

In the special case when $\boldsymbol{B} = \hat{\boldsymbol{B}}$, we recover the template mesh. Otherwise, $\bar{\boldsymbol{B}}_j{}^{-1}$ maps $\boldsymbol{t}_i$ to the local coordinate system of the bind bone j and then transporting to the target bone in world space using $\boldsymbol{B}_j$.

To constrain the space of valid poses we use a k-means prior introduced in Equation 4.2 and retrained for the new template.

8.2 Training

Parameters of the Hand BLSM are learned from the Science Museum data that we gathered and described in Chapter 7. It is the largest, by two orders of magnitude, and most diverse dataset of human hand scans. We take the first frame from each sequence that corresponds to the rest pose to learn statistical models of shape deformations. We assume that each scan $\boldsymbol{S}$ has a registration $\hat{\boldsymbol{S}}$ that has the same topology

(number of vertices, their order, and triangulation) as the Hand BLSM template. We describe the registration process in details in the next section.

8.2.1 Bone Basis and Bone-Corrective Blendshapes

A linear basis $\boldsymbol{P}_b$ for bone scales in Equation 8.5 is obtained by performing PCA on bone lengths estimated from all registrations $\hat{\boldsymbol{S}}$. We observe that the first eigenvector captures 97.85% of bone length variability and five eigenvectors capture 99.80%.

For each scan, we re-estimate the pose $\boldsymbol{\theta}$ and bone scale coefficients $\boldsymbol{c}_b$ by solving

$$\boldsymbol{\theta}^*, \boldsymbol{c}_b^* = argmin_{\boldsymbol{\theta}, \boldsymbol{c}_b} ||W(\boldsymbol{V}_T, J(\boldsymbol{c}_b, \boldsymbol{\theta})) - \hat{\boldsymbol{S}}||^2 \tag{8.10}$$

where $\boldsymbol{V}_T$ is the initial hand template created by a computer graphics artist.

The bone-corrective basis $\boldsymbol{P}_{bc}$ that update the mesh surface and the mean shape $\hat{\boldsymbol{T}}$ are found by solving

$$\boldsymbol{P}_{bc}^*, \hat{\boldsymbol{T}}^* = argmin_{\boldsymbol{P}_{bc}, \hat{\boldsymbol{T}}} ||W(\hat{\boldsymbol{T}} + \boldsymbol{c}_b \boldsymbol{P}_{bc}, J(\boldsymbol{c}_b, \boldsymbol{\theta})) - \hat{\boldsymbol{S}}||^2. \tag{8.11}$$

Samples of bone-dependent shape variations are shown in Figure 8.2 and extreme bone lengths for the first component are visualized in Figure 8.5.

8.2.2 Shape Blendshapes

The remaining part of shape variability that is not covered by bone-dependent blend shapes can be attributed to identity-specific deformations. We model these residuals as vertex displacements $\boldsymbol{T}_{ID}$ such that

$$W(\hat{\boldsymbol{T}} + \boldsymbol{c}_b \boldsymbol{P}_{bc} + \boldsymbol{T}_{ID}, J(\boldsymbol{c}_b, \boldsymbol{\theta})) = \hat{\boldsymbol{S}}. \tag{8.12}$$

Afterwards, we perform PCA on the set of vertex displacements $\{\boldsymbol{T}_{ID}^{i}\}$ to learn the shape basis $\boldsymbol{P}_{s}$. The first 20 components account for 71.33% of the variability. Samples of identity-specific shape variations are shown in Figure 8.3

8.3 Registration

In order to train our model from 1,360 subjects, we need to register the initial template to the surface of the scans. We process the hand reconstructions in pipeline that consists of multiple stages: rendering a scan from different camera views, estimating a 2D pose for each render, triangulating the results to obtain a 3D pose, fitting the hand model to the estimated joints followed by surface registration. Experiments presented in the thesis require only a rest pose registration for each subject; however, we process all the data for future applications.

8.3.1 Pose Estimation

We start by rendering a hand scan from 30 views rotated uniformly along the axis that is approximately aligned with a vector connecting a wrist and the base of a middle finger. For each render, we want to estimate a 2D pose. Our neural network (Chapter 6) requires an input image that is cropped around the palm. However, hand scans vary in the center location and size of the reconstructed hand so we cannot assume that the bounding box is fixed across sequences or frames.

The existing hand detectors do not perform well on our dataset and manual annotations would be too costly to perform. We find a solution that relies on running a body segmentation network [LXWY19] and estimating a bounding box around the predicted hand pixels. The estimated hand location is often very noisy, nevertheless, we pass the cropped image through our hand reconstruction neural network. After-



Figure 8.4: Rest pose registrations.

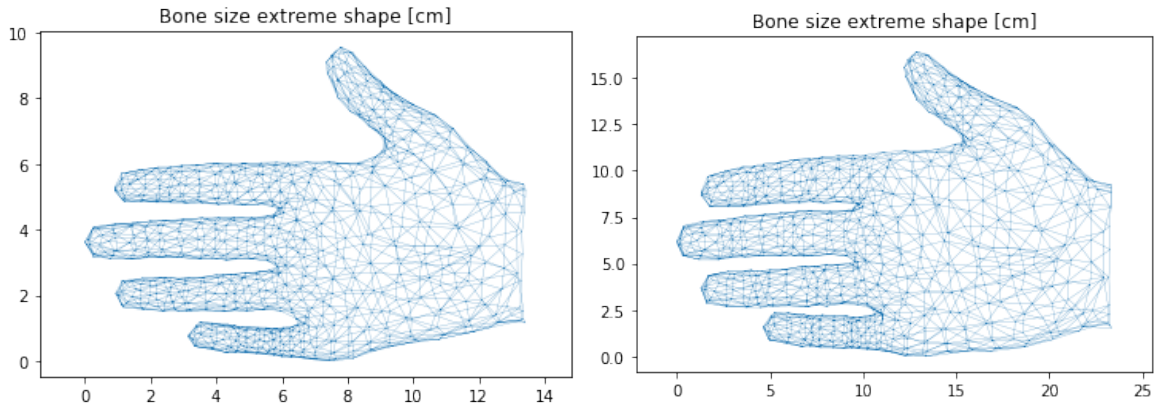


Figure 8.5: Minimum and maximum hand shapes modelled by the first component of bone-dependent shape variations. Our hand model is capable of representing children and adults.

wards, we crop the image around the predicted hand mesh location and we correct the initial prediction. For subsequent frames of the recording of the same subject, we do not re-estimate the location. Instead, we use temporal smoothing based on taking a weighted average of the current bounding box prediction and the previous one which is experimentally shown to be very robust to hand movements. In order to smooth neural network predictions, we average the regressed latent vectors in a similar manner before passing the latent variable to the mesh decoder.

Our neural network predicts a dense 3D hand mesh aligned with the image. We take X, Y coordinates of the selected landmarks on the mesh surface. A 3D pose is triangulated from the predicted 2D keypoints by a gradient descent based minimization of the reprojection error. Afterwards, we keep half of the camera views with the lowest reprojection error and re-triangulate the pose.

8.3.2 Fitting

Model Optimization The estimated 3D pose is associated with anatomical landmarks $\boldsymbol{L}$ that we use for model alignment and fitting to the scan. More specifically, we solve the following optimization problem with gradient descent on the pose parameters, bone scales, and camera parameters ϕ (global orientation and translation):

$$\boldsymbol{\theta}^*, \boldsymbol{s}^*, \phi^* = argmin_{\boldsymbol{\theta},\boldsymbol{s},\phi}(L_{kpts} + \lambda_{dist} L_{dist} + \lambda_{reg\_scale} L_{reg\_scale} + \lambda_{reg\_pose} L_{reg\_pose}) \quad (8.13)$$

where λ are experimentally chosen hyperparameters. The first term penalizes the distance between the landmarks and the corresponding points selected from the template:

$$L_{kpts} = ||\boldsymbol{A}\mathcal{Y} - \boldsymbol{L}||^2 \quad (8.14)$$

where $\mathcal{Y} = W(\boldsymbol{V}_T, J(\boldsymbol{s}, \boldsymbol{\theta}))$ and $\boldsymbol{A}$ is a vertex selector on the template surface. Estimated landmarks are often very noisy due to hand pose estimation error, therefore, we add a Chamfer loss that minimizes a bi-directional distance between point clouds of the hand template and scan:

$$L_{dist}(\mathcal{Y}, \boldsymbol{S}) = \sum_{\boldsymbol{y} \in \mathcal{Y}} min_{\boldsymbol{x} \in \boldsymbol{S}} ||\boldsymbol{y} - \boldsymbol{x}||_2^2 + \sum_{\boldsymbol{x} \in \boldsymbol{S}} min_{\boldsymbol{y} \in \mathcal{Y}} ||\boldsymbol{y} - \boldsymbol{x}||_2^2. \tag{8.15}$$

The optimization is regularized by penalizing deviations from the base template: $L_{reg\_scale}(\boldsymbol{s}) = ||\boldsymbol{s}||^2$ and $L_{reg\_pose}(\boldsymbol{\theta}) = ||\boldsymbol{\theta}||^2$.

Surface Deformations After obtaining initial alignment of the hand template to the scan surface by optimizing pose, shape, and camera parameters of the hand model, we perform a registration by finding an offset $\boldsymbol{T}_{reg}$. We solve the following optimization problem with error terms that ensure smooth deformations:

$$\boldsymbol{T}^*_{reg} = argmin_{\boldsymbol{T}_{reg}}(L_{reg} + L_{edge\_length} + L_{edge} + \lambda_\Delta L_\Delta). \tag{8.16}$$

The L_{reg} minimizes the Chamfer distance between the hand template updated with an offset and the scan:

$$L_{reg} = L_{dist}(\hat{\mathcal{Y}}, \boldsymbol{S}), \tag{8.17}$$

$$\hat{\mathcal{Y}} = W(\boldsymbol{T}(\theta^*, \boldsymbol{s}^*) + \boldsymbol{T}_{reg}, J(\boldsymbol{s}^*, \boldsymbol{\theta}^*)). \tag{8.18}$$

Deformations are constrained by a penalty to preserve initial template edge lengths

$$\mathcal{L}_{edge\_length} = \sum_{(u,v) \in \mathcal{E}_{mesh}} \lambda^{uv}_{edge\_length} \,||\, ||\hat{\mathcal{Y}}_v - \hat{\mathcal{Y}}_u|| - ||\mathcal{Y}_v - \mathcal{Y}_u||\,||^2 \tag{8.19}$$

and a loss term on the edge locations

$$\mathcal{L}_{edge} = \sum_{(u,v) \in \mathcal{E}_{mesh}} \lambda^{uv}_{edge} \,|\, (\hat{\mathcal{Y}}_v - \hat{\mathcal{Y}}_u) - (\mathcal{Y}_v - \mathcal{Y}_u)\,|. \tag{8.20}$$

Hyperparameters $\lambda^{uv}_{edge\_length}$ and λ^{uv}_{edge} are defined per edge because we apply higher regularization to wrist edges due to the surface boundary. We also apply laplacian smoothing defined as $\mathcal{L}_\Delta = \mathbf{\Delta}\hat{\mathcal{Y}}$ where $\mathbf{\Delta}$ is a discrete Laplacian (Chapter 3.5.1).

Examples of registration results are shown in Figure 8.4.

8.4 Evaluation

Fitted hand model	RHD [mm]	YouTube [px]
MANO	10.873	10.494
BLSM	**10.379**	**9.639**

Table 8.1: 3D Pose Error. Performance with models trained on datasets with different hand fittings.

To evaluate our hand model, we compare it against MANO by retraining the hand reconstruction network (Chapter 6) with the dataset that we generate in a similar manner as in Chapter 4 but with our new model. The evaluation method is motivated by the main objective of the thesis - advancing 3D hand reconstruction and pose estimation from a single image. The evaluation results, presented in Table 8.1, clearly indicate improvement from replacing the type of a deformable model in a weakly-supervised neural network.

8.5 Conclusion

We have created the most diverse statistical hand model learned from 1,360 subjects from a wide age range (3 - 81 years old), broad ethnic groups, and a balanced male/female ratio. Its novel formulation based on disentanglement of bone-length shape variations and subject-specific deformations brings advantages to computer graphics and augmented reality applications. Beyond direct applications, the new model can be used as a source of supervision for training hand reconstruction neural networks and we have proved that it indeed leads to a significant improvement, outcompeting our previous state-of-the-art system.

Chapter 9

Texture Model

Texture can provide a strong supervision signal for hand reconstruction systems. So far we have been aligning a parametric hand model with a sparse set of keypoints corresponding to the hand joints that were either manually annotated or estimated by a neural network (Chapter 4). Therefore, the shape variations in the training data were limited to the bone lengths. The fittings were then used as training data for a hand reconstruction neural network (Chapter 6). We achieved excellent results in 3D pose estimation enabling accurate real-time mobile applications. However, the identity-specific hand shape reconstruction quality was bounded by the training data. One could potentially improve shape estimation by annotating silhouettes but this would not work for articulated poses due to a high level of self-occlusions. Less ambiguous information could come from modelling texture variations and incorporating a photometric loss to provide a pixel-accurate alignment via a self-supervised network.

In parallel to our work, Qian et al. [QWM+20] proposed the first hand texture model that is learned from 51 subjects in the age group 20-59 years old dominated by male samples. The authors use a mobile application to reconstruct a partial scan of one side of the hand resting on a table with approximately 6.2K vertices and 11K trian-

gles. Registration process relies then on fitting a parametric model into scans for both sides of the hand. The authors admit that the hand pose estimation network that guides the initial alignment is highly inaccurate. The system is a reliable proof of concept but due to its low-scale dataset and compounding errors from a registration procedure it might not generalize to wide variety of skin tones, marks, medical conditions, tattoos, and jewellery.

In contrast, our dataset introduced in Chapter 7 includes 1,360 subjects 3-81 years old with a balanced male/female ratio and a much higher number of samples in each ethnic group (Figure 7.7). Moreover, our hand scans come from a professional multi-camera system that generates high-resolution meshes of the full surface at 30 FPS with approximately 32K vertices and 65k triangles. Leveraging our cutting edge hand reconstruction system (Chapter 6) and parametric hand model (Chapter 8), we build the most diverse statistical texture model capable of representing children, adults, and the elderly from a broad range of ethnic groups. We believe that our contributions will enable novel industrial applications for virtual and augmented reality and improve the existing hand reconstruction systems.

9.1 Texture Mapping

Given a template $\boldsymbol{T}$ with vertices $\boldsymbol{t} \in \boldsymbol{T}$, a UV space is a two dimensional space $\boldsymbol{U} \subset \mathbb{R}^2$ that satisfies a correspondence property. This means that there exists a bijective map from $\boldsymbol{T}$ to $\boldsymbol{U}$ for all $\boldsymbol{u} \in \boldsymbol{U}$:

$$f(\boldsymbol{t}) \rightarrow \boldsymbol{u} \quad f^{-1}(\boldsymbol{u}) \rightarrow \boldsymbol{t} \tag{9.1}$$

The UV map is found by unwrapping the mesh to a plane. The body template from which we extracted hand vertices already contains the body UV map. Based on the

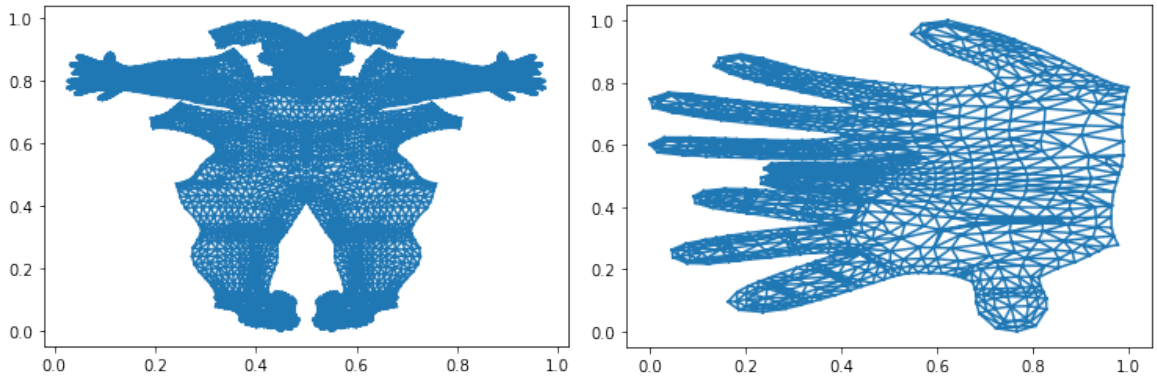


Figure 9.1: UV maps for the body and hand templates.

correspondence between hand mesh faces and body mesh faces, we extract the UV map for the hand (Figure 9.1). Afterwards, we normalize the UV map such that horizontal and vertical coordinates for the vertex $\boldsymbol{t}$ are represented by accordingly $\boldsymbol{u}^h \in [0, 1]$ and $\boldsymbol{u}^v \in [0, 1]$.

UV textures generated by the hand scanner are not in correspondence and contain background pixels. The scan texture is constructed by concatenating 5 colour images that can be seen in Figure 7.4; each of the 5 images has a resolution $2,048 \times 2,448$ or $2,448 \times 2,048$. We do not show the full resolution images due to privacy concerns.

Given registered hand scans in a rest pose (Chapter 8), we are able to transfer textures to the hand template UV space. For each point on the scan, we find the closest point in barycentric coordinates on the registered template and its UV map coordinate. Afterwards, we transfer the color to the template UV map (Figure 9.2).

9.2 Texture Model

We synthesize new textures from a combination of linear basis computed by performing PCA on a collection of UV maps. Specifically, a new texture $\boldsymbol{M}(\boldsymbol{\alpha})$ is obtained

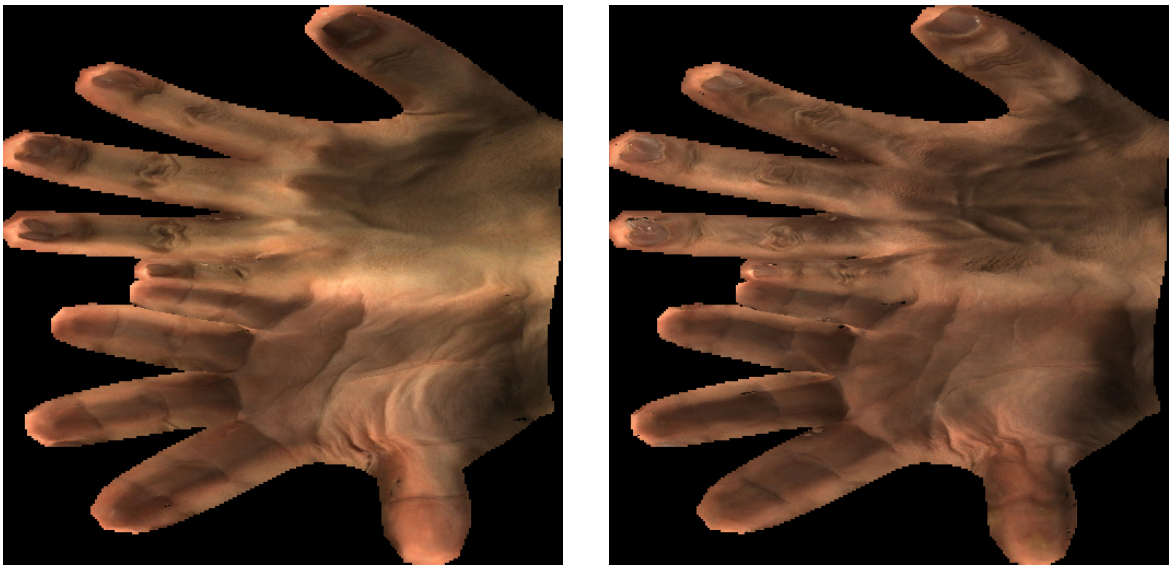


Figure 9.2: Examples of transferred textures.

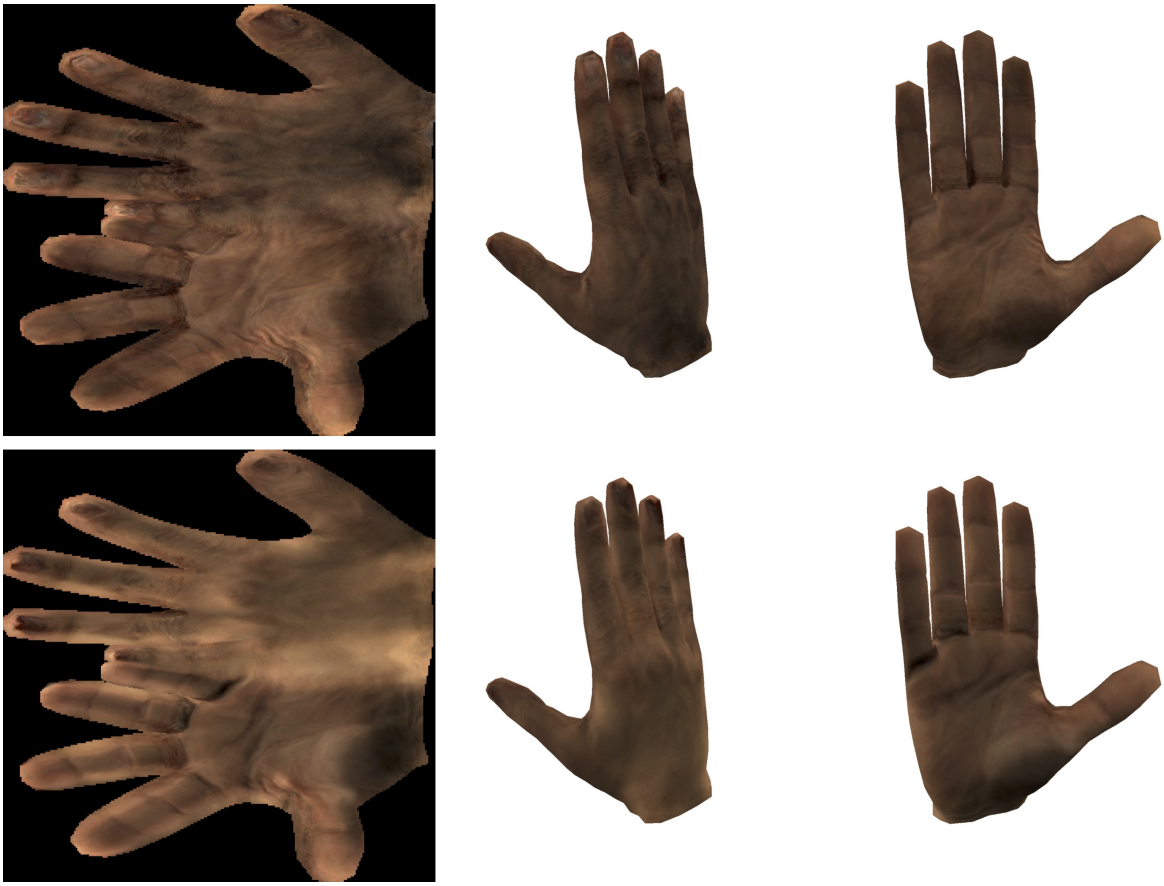


Figure 9.3: Random samples from the texture model visualized on the initial template.

from

$$\boldsymbol{M}(\boldsymbol{\alpha}) = \bar{\boldsymbol{M}} + \sum_{d=1}^{|\boldsymbol{\alpha}|} \alpha_d \boldsymbol{C}_d \tag{9.2}$$

where $\bar{\boldsymbol{M}} \in \mathbb{R}^{3P}$ is the mean texture of the vectorized UV maps with $P = 492,216$ RGB pixels inside the hand boundary region and $\boldsymbol{C} = [\boldsymbol{C}_1, ..., \boldsymbol{C}_D] \in \mathbb{R}^{3P \times D}$ is the orthonormal basis matrix of the texture eigenvectors. Parameters α_d control blending of the bases.

9.3 Evaluation

For the experiments, we create a model from a subset of the dataset counting 844 samples with unique subjects that do not contain significant noise. We also reject samples that include jewellery or tattoos due to their high non-linearities. We could reduce noise by averaging textures from different frames and doing post-processing but we leave this for future work. We follow the evaluation protocol, with minor changes given larger dataset size, for hand texture PCA models established by Qian et al. [QWM+20] that considers three aspects: compactness, specificity, and generalization.

Compactness is the measure of the amount of variance in the training set explained by the model in terms of the number of principal components.

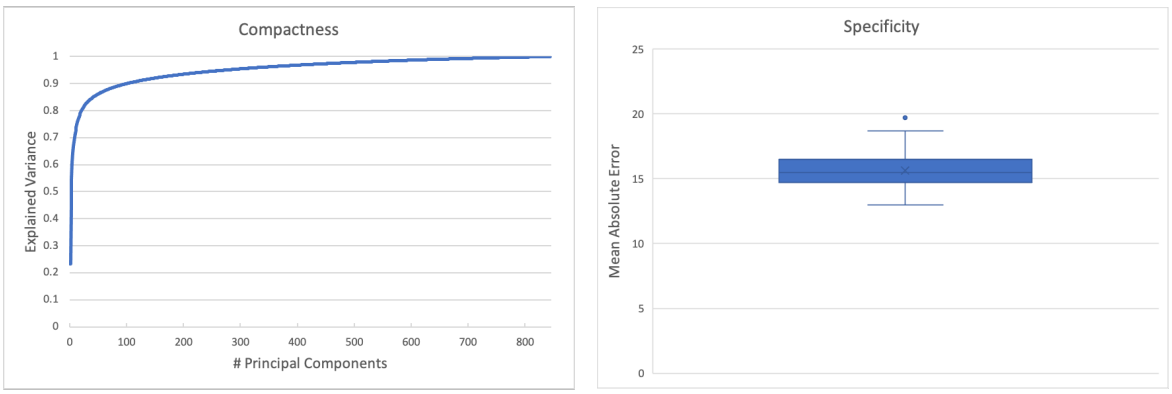


Figure 9.4: Evaluation of the texture model.

Specificity measures the similarity between samples drawn from the model based on a multivariate Gaussian distribution and the training data. We draw 200 samples and find the nearest texture in the training dataset in terms of the mean absolute error. The evaluation results are reported in Figure 9.4.

We also evaluate how the model generalizes to test samples. We consider a challenging task of reconstructing noisy textures and textures with subjects wearing a jewellery that we removed from the training dataset. The test set counts 114 samples and the error is just 7.10 of the RGB intensity on the vectorized hand pixels. We can also observe in Figure 9.5 that the model is robust to the noise.

9.4 Conclusion

We introduced the most diverse hand texture model with the highest resolution of skin details. We showed that the model is capable of generating realistic textures and generalizes well to unseen noisy samples. The model due to its unprecedented quality can enable highly realistic computer graphics animations and find use in training pixel-accurate hand reconstruction networks in a self-supervised manner.

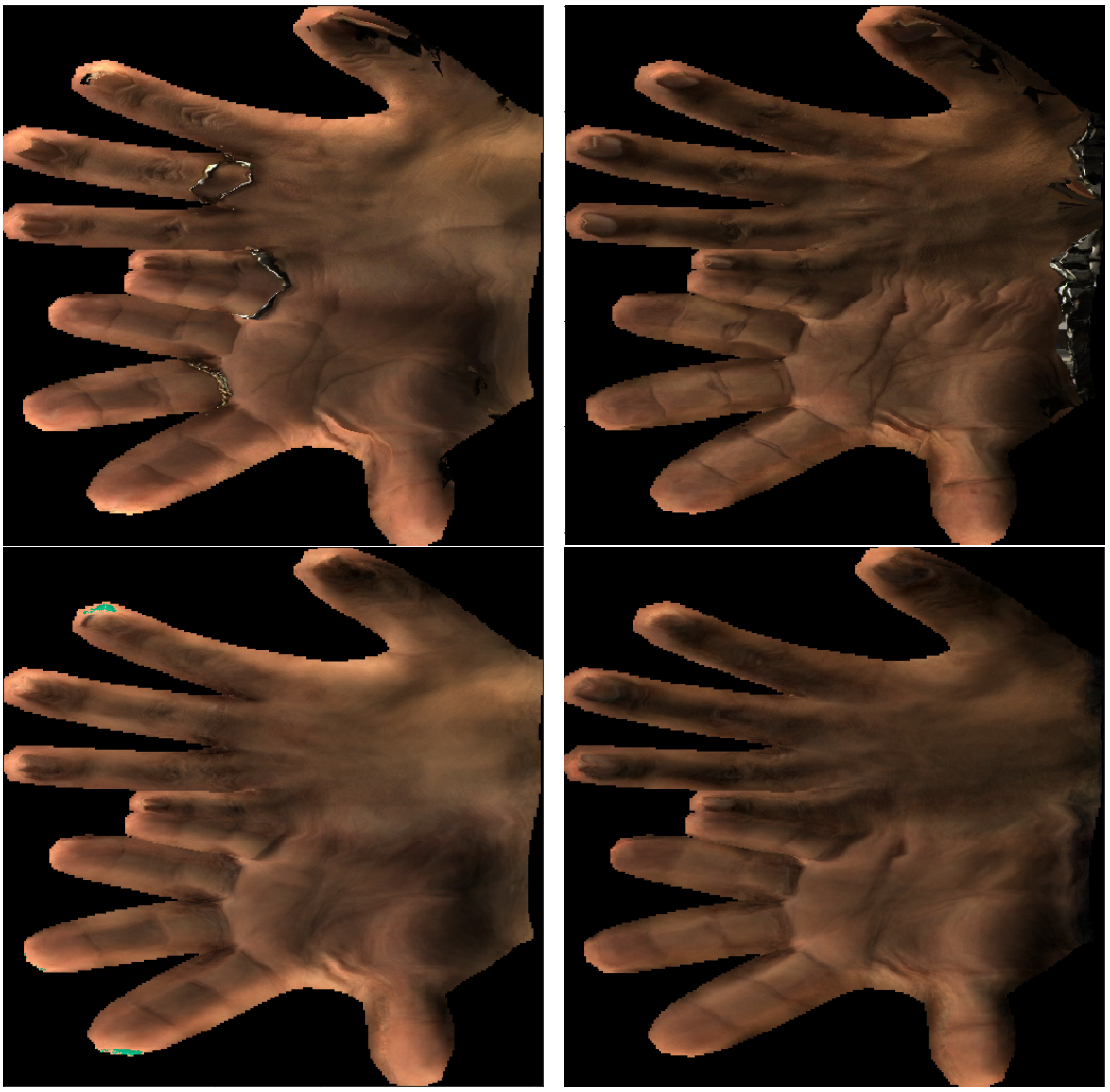


Figure 9.5: Pairs of the noisy test samples (top row) and their reconstructions (bottom row).

Chapter 10

Conclusion

Summary of Thesis Achievements We have advanced the field of hand reconstruction and modelling by addressing its major weaknesses: performance in non-laboratory environments, inference time, data availability, shape diversity and bone length disentanglement, and texture synthesis.

We have showed that neural networks are powerful methods for representing highly-articulated non-linear hand deformations in a compact mesh-convolutional decoder while improving speed and accuracy (Chapter 5). We have addressed the problem of collecting dense 3D hand mesh annotations for RGB images by building an automated system for parametric model fitting and we made the first in the wild dataset for 3D hand pose estimation and mesh reconstruction publicly available (Chapter 4). Furthermore, we have proposed a novel weakly-supervised neural network for hand mesh reconstruction (Chapter 6). The system obtains unprecedented performance on the in the wild scenario while working at 250 FPS on mobile devices.

We have also collected the largest and most diverse dataset of 3D hand scans in Science Museum in London from 1,360 subjects in the age range 3 - 81 years old (Chapter 7). Afterwards, we processed over half a million scans to a common rep-

resentation. This allowed us to perform statistical analysis of shape and texture variations. Specifically, we have built a deformable model of the human hand that disentangles bone length variability from other identity-specific traits (Chapter 8). We have showed that incorporating this model to our weakly-supervised hand reconstruction network improves its state-of-the-art performance even further. We devoted the last chapter to the problem of texture synthesis (Chapter 9). We took advantage of our large-scale hand scans dataset to learn high-resolution skin texture variations which has not been previously possible.

Applications The thesis argues that robust systems for hand mesh reconstruction are crucial for development of augmented/mixed/virtual reality technology by enabling intuitive control and interaction with the digital objects. Such systems also find applications in motion tracking, learning from demonstration / behavioural cloning, virtual telepresence, sign language recognition, healthcare, realistic prosthetic design, and immersive gaming. Our hand reconstruction and iterative model fitting approaches can be adapted to the research on full body capture by changing the underlying mesh representation. The proposed hand reconstruction network and statistical deformable models have been successfully incorporated into Ariel AI products offering novel augmented reality experiences.

Future Work There are still plenty of opportunities to move the field forward by extending our contributions. The current hand reconstruction systems from a single image are accurate in localizing joints and approximating hand surface. However, they are not pixel-accurate because of the data collection approach that relies on iterative model fitting. We could fix this by incorporating a photometric loss to provide an alignment of the hand texture with image pixels via a self-supervised network which is now possible with our statistical texture model. Alternatively, we can take advantage of the collected dataset of 3D scans or the created deformable

model to generate a synthetic dataset with dense annotations. It turns out that networks trained on hand pose datasets do not generalize well to hand-object or hand-hand interactions due to a high level of occlusions and self-similarity between hands. Collecting sufficiently large dataset of manipulating objects might be infeasible and it might be necessary to develop systems that specifically model spatial relations between interacting objects. Another interesting research avenue arises from the observation that deformable hand models are usually posed with skinning methods that cause artifacts and require corrective functions. It would be interesting to explore disentanglment methods and obtain parameterized control over pose-deformations directly from the data. Finally, our contributions open new opportunities that were not possible due to poor hand pose estimation performance in non-laboratory environments such as analyzing hand gestures or training reinforcement learning systems from large repositories of videos.

Bibliography

[ACCP03] Brett Allen, Brian Curless, Brian Curless, and Zoran Popović. The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Trans. Graph.*, 22(3):587–594, July 2003.

[ACPH06] Brett Allen, Brian Curless, Zoran Popović, and Aaron Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 147–156, Aire-la-Ville, Switzerland, 2006. Eurographics Association.

[APGS14] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3686–3693, June 2014.

[ASK+05] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM transactions on graphics (TOG)*, volume 24, pages 408–416. ACM, 2005.

[BAP+17] James Booth, Epameinondas Antonakos, Stylianos Ploumpis, George Trigeorgis, Yannis Panagakis, and Stefanos Zafeiriou. 3d face mor-

phable models" in-the-wild". In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5464–5473. IEEE, 2017.

[BBL+17] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[BBP+19] Giorgos Bouritsas, Sergiy Bokhnyak, Stylianos Ploumpis, Michael Bronstein, and Stefanos Zafeiriou. Neural 3d morphable models: Spiral convolutional networks for 3d shape representation learning and generation. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019.

[BBT19] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3d hand shape and pose from images in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10843–10852, 2019.

[BKK19] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[BKK20] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Weakly-supervised domain adaptation via gan and mesh model for estimating 3d hand poses interacting objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[BKL+16] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In

Computer Vision – ECCV 2016, Lecture Notes in Computer Science. Springer International Publishing, October 2016.

[BMRB16] Davide Boscaini, Jonathan Masci, Emanuele Rodolà, and Michael Bronstein. Learning shape correspondence with anisotropic convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 3189–3197, 2016.

[BRZ+16] James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David Dunaway. A 3d morphable model learnt from 10,000 faces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5543–5552, 2016.

[BTG+12] Luca Ballan, Aparna Taneja, Jürgen Gall, Luc Van Gool, and Marc Pollefeys. Motion capture of hands in action using discriminative salient points. In *European Conference on Computer Vision (ECCV)*, pages 640–653, Firenze, October 2012.

[BTT+20] Samarth Brahmbhatt, Chengcheng Tang, Christopher D. Twigg, Charles C. Kemp, and James Hays. ContactPose: A dataset of grasps with object contact and hand pose. In *The European Conference on Computer Vision (ECCV)*, August 2020.

[BV99] 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*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999.

[BZSL14] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *2nd International Conference on Learning Representations, ICLR 2014,*

Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings, 2014.

[CBZ+19] Shiyang Cheng, Michael Bronstein, Yuxiang Zhou, Irene Kotsia, Maja Pantic, and Stefanos Zafeiriou. Meshgan: Non-linear 3d morphable models of faces. *CoRR*, abs/1903.10384, 2019.

[CGCY18] Yujun Cai, Liuhao Ge, Jianfei Cai, and Junsong Yuan. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *ECCV*, 2018.

[CGL+19] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[CLZ13] Yinpeng Chen, Zicheng Liu, and Zhengyou Zhang. Tensor-based human body modeling. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '13, pages 105–112, Washington, DC, USA, 2013. IEEE Computer Society.

[CPB+20] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. Monocular expressive body regression through body-driven attention. In *European Conference on Computer Vision (ECCV)*, 2020.

[CWZ+13] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, 2013.

[DBV16] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in neural information processing systems*, pages 3844–3852, 2016.

[dLGFP11] 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*, 33(9):1793–1805, Sept 2011.

[DMB+18] Endri Dibra, Silvan Melchior, Ali Balkis, Thomas Wolf, Cengiz Oztireli, and Markus Gross. Monocular rgb hand pose inference from unsupervised refinable nets. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[DSWV+17] Quentin De Smedt, Hazem Wannous, Jean-Philippe Vandeborre, Joris Guerry, Bertrand Le Saux, and David Filliat. SHREC'17 Track: 3D Hand Gesture Recognition Using a Depth and Skeletal Dataset. In I. Pratikakis, F. Dupont, and M. Ovsjanikov, editors, *10th Eurographics Workshop on 3D Object Retrieval*, Lyon, France, April 2017.

[FLW20] Zhipeng Fan, Jun Liu, and Yao Wang. Adaptive computationally efficient network for monocular 3d hand pose estimation. In *ECCV*, 2020.

[Gav98] Dariu Mihai Gavrila. *Vision-Based 3-D Tracking of Humans in Action*. PhD thesis, University of Maryland at College Park, USA, 1998.

[GDOEC19] Francisco Gomez-Donoso, Sergio Orts-Escolano, and Miguel Cazorla. Large-scale multiview 3d hand pose dataset. *Image and Vision Computing*, 81:25–33, 2019.

[GH97] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on*

Computer graphics and interactive techniques, pages 209–216. ACM Press/Addison-Wesley Publishing Co., 1997.

[GHYBK18] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 409–419, 2018.

[GK19] Riza Alp Guler and Iasonas Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[GLYT16] Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3593–3601, 2016.

[GRL+19] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3d hand shape and pose estimation from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10833–10842, 2019.

[GWF+19] Yafei Gao, Yida Wang, Pietro Falco, Nassir Navab, and Federico Tombari. Variational object-aware 3d hand pose from a single rgb image. *IEEE Robotics and Automation Letters*, PP:1–1, 07 2019.

[HLC+20] Shangchen Han, Beibei Liu, Randi Cabezas, Christopher D. Twigg, Peizhao Zhang, Jeff Petkau, Tsz-Ho Yu, Chun-Jung Tai, Muzaffer Akbay, Zheng Wang, Asaf Nitzan, Gang Dong, Yuting Ye, Lingling Tao, Chengde Wan, and Robert Wang. Megatrack: Monochrome egocentric articulated hand-tracking for virtual reality. *ACM Trans. Graph.*, 39(4), July 2020.

[HLRB12] David A Hirshberg, Matthew Loper, Eric Rachlin, and Michael J Black. Coregistration: Simultaneous alignment and modeling of articulated 3d shape. In *European Conference on Computer Vision*, pages 242–255. Springer, 2012.

[HLVDMW17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[HSS+09] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel. A statistical model of human pose and body shape. *Computer Graphics Forum*, 28(2):337–346, 2009.

[HVT+19a] Yana Hasson, Gül Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019.

[HVT+19b] Yana Hasson, Gül Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019.

[IOA11] Nikolaos Kyriazis Iason Oikonomidis and Antonis Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *Proceedings of the British Machine Vision Conference*, pages 101.1–101.11. BMVA Press, 2011. http://dx.doi.org/10.5244/C.25.101.

[JSS18] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8320–8329, 2018.

[KBJM18] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018.

[KM96] Ioannis A Kakadiaris and Dimitris Metaxas. Model-based estimation of 3d human motion with occlusion based on active multi-viewpoint selection. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 81–87. IEEE, 1996.

[KPBD19] Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *International Conference on Computer Vision*, October 2019.

[KPD19] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[KPZK17] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Trans. Graph.*, 36(4):78:1–78:13, July 2017.

[KTS+15] Sameh Khamis, Jonathan Taylor, Jamie Shotton, Cem Keskin, Shahram Izadi, and Andrew Fitzgibbon. Learning an efficient model of hand shape variation from depth images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2540–2548, 2015.

[KW17] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[LBB+17] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017.

[LCF00] J. P. Lewis, Matt Cordner, and Nickson Fong. Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 165–172, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[LDCK18] Isaak Lim, Alexander Dielen, Marcel Campen, and Leif Kobbelt. A simple approach to intrinsic correspondence learning on unstructured 3d meshes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[LDS+19] J. Liu, H. Ding, A. Shahroudy, L. Duan, X. Jiang, G. Wang, and A. Kot Chichung. Feature boosting network for 3d pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019.

[LGJV17] Marcel Lüthi, Thomas Gerig, Christoph Jud, and Thomas Vetter. Gaussian process morphable models. *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[LMB+14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft

coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[LMBB18] Ron Levie, Federico Monti, Xavier Bresson, and Michael M Bronstein. Cayleynets: Graph convolutional neural networks with complex rational spectral filters. *IEEE Transactions on Signal Processing*, 67(1):97–109, 2018.

[LMR+15] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

[LRK+17] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black, and Peter V Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6050–6059, 2017.

[LXWY19] Peike Li, Yunqiu Xu, Yunchao Wei, and Yi Yang. Self-correction for human parsing. *arXiv preprint arXiv:1910.09777*, 2019.

[MBBV15] Jonathan Masci, Davide Boscaini, Michael Bronstein, and Pierre Vandergheynst. Geodesic convolutional neural networks on riemannian manifolds. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 37–45, 2015.

[MBM+17] Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5115–5124, 2017.

[MBS+18] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–59, 2018.

[MDB+19] Franziska Mueller, Micah Davis, Florian Bernard, Oleksandr Sotnychenko, Mickeal Verschoor, Miguel A. Otaduy, Dan Casas, and Christian Theobalt. Real-time Pose and Shape Reconstruction of Two Interacting Hands With a Single Depth Camera. *ACM Transactions on Graphics (TOG)*, 38(4), 2019.

[MEC+17] Meysam Madadi, Sergio Escalera, Alex Carruesco, Carlos Andujar, Xavier Baro, and Jordi Gonzalez. Occlusion aware hand pose recovery from sequences of depth images. *IEEE International Conference on Automatic Face and Gesture Recognition*, 2017.

[MEN+18] Jameel Malik, Ahmed Elhayek, Fabrizio Nunnari, Kiran Varanasi, Kiarash Tamaddon, Alexis Heloir, and Didier Stricker. Deephps: End-to-end estimation of 3d hand pose and shape by learning from synthetic depth. In *2018 International Conference on 3D Vision (3DV)*, pages 110–119. IEEE, 2018.

[MENS19] Jameel Malik, Ahmed Elhayek, Fabrizio Nunnari, and Didier Stricker. Simple and effective deep hand shape and pose regression from a single depth image. *Computers & Graphics*, 85:85–91, 2019.

[MKO13] Stan Melax, Leonid Keselman, and Sterling Orsten. Dynamics based 3d skeletal hand tracking. In *Proceedings of Graphics Interface 2013*, GI '13, pages 63–70, Toronto, Ont., Canada, Canada, 2013. Canadian Information Processing Society.

[ML20] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *European Conference on Computer Vision (ECCV)*, 2020.

[MMAP] R. McKee, D. McKee, D. Alexander, and E. Paillat. Nz sign language exercises. Deaf Studies Department of Victoria University of Wellington.

[MMS+17] Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of International Conference on Computer Vision (ICCV)*, October 2017.

[MSL20] Gyeongsik Moon, Takaaki Shiratori, and Kyoung Mu Lee. Deephandmesh: A weakly-supervised deep encoder-decoder framework for high-fidelity hand mesh modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2020.

[MYW+20] 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 *European Conference on Computer Vision (ECCV)*, 2020.

[OKA11] Iasonas Oikonomidis, Nikolaos Kyriazis, and Antonis A. Argyros. Markerless and efficient 26-dof hand pose recovery. In *Proceedings of the 10th Asian Conference on Computer Vision - Volume Part III*, ACCV'10, pages 744–757, Berlin, Heidelberg, 2011. Springer-Verlag.

[OKA12] Iasonas Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Tracking the articulated motion of two strongly interacting hands.

In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1862–1869. IEEE, 2012.

[OL17] Markus Oberweger and Vincent Lepetit. Deepprior++: Improving fast and accurate 3d hand pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 585–594, 2017.

[OLA14] Iason Oikonomidis, Manolis IA Lourakis, and Antonis A Argyros. Evolutionary quasi-random search for hand articulations tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3422–3429, 2014.

[OLPM+18] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *2018 International Conference on 3D Vision (3DV)*, pages 484–494. IEEE, 2018.

[PCG+19] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[PF01] Ralf Plänkers and Pascal Fua. Tracking and modeling people in video sequences. *Computer Vision and Image Understanding*, 81:285–302, 03 2001.

[PMRMB15] Gerard Pons-Moll, Javier Romero, Naureen Mahmood, and Michael Black. Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 34:120:1–120:14, 08 2015.

[POA18] Paschalis Panteleris, Iason Oikonomidis, and Antonis Argyros. Using a single rgb frame for real time 3d hand pose estimation in the wild. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 436–445. IEEE, 2018.

[PZZD18] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 459–468, 2018.

[QSW+14a] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun. Realtime and robust hand tracking from depth. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1106–1113, June 2014.

[QSW+14b] Chen Qian, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun. Realtime and robust hand tracking from depth. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1106–1113, 2014.

[QWM+20] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2020.

[RBSB18] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. Generating 3d faces using convolutional mesh autoencoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 704–720, 2018.

[RK94] James M. Rehg and Takeo Kanade. Visual tracking of high dof articulated structures: An application to human hand tracking. In *Proceed-*

ings of the Third European Conference-Volume II on Computer Vision - Volume II, ECCV '94, pages 35–46, London, UK, UK, 1994. Springer-Verlag.

[RKSI+14] Grégory Rogez, Maryam Khademi, JS Supančič III, Jose Maria Martinez Montiel, and Deva Ramanan. 3d hand pose detection in egocentric rgb-d images. In *European Conference on Computer Vision*, pages 356–371. Springer, 2014.

[RTB17] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):245:1–245:17, November 2017. (*) Two first authors contributed equally.

[SCY+20] Jingjing Shen, Thomas J Cashman, Qi Ye, Tim Hutton, Toby Sharp, Federica Bogo, Andrew William Fitzgibbon, and Jamie Shotton. The phong surface: Efficient 3d model fitting using lifted optimization. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2020.

[SJMS17] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017.

[SKR+15] Toby Sharp, Cem Keskin, Duncan P. Robertson, Jonathan Taylor, Jamie Shotton, David Kim, Christoph Rhemann, Ido Leichter, Alon Vinnikov, Yichen Wei, Daniel Freedman, Pushmeet Kohli, Eyal Krupka, Andrew W. Fitzgibbon, and Shahram Izadi. Accurate, robust, and flexible real-time hand tracking. In *CHI*, pages 3633–3642. ACM, 2015.

[SMOT15] Srinath Sridhar, Franziska Mueller, Antti Oulasvirta, and Christian Theobalt. Fast and robust hand tracking using detection-guided optimization. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[SMZ+16] Srinath Sridhar, Franziska Mueller, Michael Zollhoefer, Dan Casas, Antti Oulasvirta, and Christian Theobalt. Real-time joint tracking of a hand manipulating an object from rgb-d input. In *Proceedings of European Conference on Computer Vision (ECCV)*, October 2016.

[SNF+13] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

[SNF14] Tanner Schmidt, Richard A. Newcombe, and Dieter Fox. DART: dense articulated real-time tracking. In *Robotics: Science and Systems*, 2014.

[SOT13] Srinath Sridhar, Antti Oulasvirta, and Christian Theobalt. Interactive markerless articulated hand motion tracking using rgb and depth data. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[SRY+15] James S Supancic, Grégory Rogez, Yi Yang, Jamie Shotton, and Deva Ramanan. Depth-based hand pose estimation: data, methods, and challenges. In *Proceedings of the IEEE international conference on computer vision*, pages 1868–1876, 2015.

[SSPH18] Adrian Spurr, Jie Song, Seonwook Park, and Otmar Hilliges. Cross-modal deep variational hand pose estimation. In *CVPR*, 2018.

[SWL+15] X. Sun, Y. Wei, Shuang Liang, X. Tang, and J. Sun. Cascaded hand pose regression. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 824–832, June 2015.

[TBC+16] Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Toby Sharp, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, Arran Topalian, Erroll Wood, Sameh Khamis, Pushmeet Kohli, Shahram Izadi, Richard Banks, Andrew Fitzgibbon, and Jamie Shotton. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. *ACM Trans. Graph.*, 35(4):143:1–143:12, July 2016.

[TBC17] Vince Tan, Ignas Budvytis, and Roberto Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. *BMVC*, 2017.

[TBP19] Bugra Tekin, Federica Bogo, and Marc Pollefeys. H+ o: Unified egocentric recognition of 3d hand-object poses and interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4511–4520, 2019.

[TBS+16a] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient points and physics simulation. *International Journal of Computer Vision*, 118(2):172–193, 2016.

[TBS+16b] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient points and physics simulation. *International Journal of Computer Vision (IJCV)*, 2016.

[TCTK14] Danhang Tang, Hyung Jin Chang, Alykhan Tejani, and Tae-Kyun Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, pages 3786–3793, Washington, DC, USA, 2014. IEEE Computer Society.

[TPT16] Anastasia Tkach, Mark Pauly, and Andrea Tagliasacchi. Sphere-meshes for real-time hand modeling and tracking. *ACM Trans. Graph.*, 35:222:1–222:11, 2016.

[TSLP14] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Trans. Graph.*, 33(5):169:1–169:10, September 2014.

[TSR+14] Jonathan Taylor, Richard Stebbing, Varun Ramakrishna, Cem Keskin, Jamie Shotton, Shahram Izadi, Aaron Hertzmann, and Andrew Fitzgibbon. User-specific hand modeling from monocular depth sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 644–651, 2014.

[TST+15] Andrea Tagliasacchi, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. Robust articulated-icp for real-time hand tracking. In *Proceedings of the Eurographics Symposium on Geometry Processing*, SGP '15, pages 101–114, Aire-la-Ville, Switzerland, Switzerland, 2015. Eurographics Association.

[VBV18] Nitika Verma, Edmond Boyer, and Jakob Verbeek. Feastnet: Feature-steered graph convolutions for 3d shape analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2598–2606, 2018.

[VCC+18] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. *International Conference on Learning Representations*, 2018.

[VCR+18] Gul Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 20–36, 2018.

[Š11] Marin Šarić. Libhand: A library for hand articulation, 2011. Version 0.9.

[WBC+20] Philippe Weinzaepfel, Romain Brégier, Hadrien Combaluzier, Vincent Leroy, and Grégory Rogez. Dope: Distillation of part experts for whole-body 3d pose estimation in the wild. In *ECCV*, 2020.

[WC20] Kewen Wang and Xilin Chen. Pmd-net: Privileged modality distillation network for 3d hand pose estimation from a single rgb image. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2020.

[WCKS19] Chung-Yi Weng, Brian Curless, and Ira Kemelmacher-Shlizerman. Photo wake-up: 3d character animation from a single photo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5908–5917, 2019.

[WGK+20] Haoyang Wang, Riza Alp Guler, Iasonas Kokkinos, George Papandreou, and Stefanos Zafeiriou. Blsm: A bone-level skinned model of the human mesh. In *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020.

[WRKS16] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[WSK15] Aaron Wetzler, Ron Slossberg, and Ron Kimmel. Rule of thumb: Deep derotation for improved fingertip detection. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2015.

[XBZ+20] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020.

[XC13] Chi Xu and Li Cheng. Efficient hand pose estimation from a single depth image. In *International Conference on Computer Vision (ICCV)*, 2013.

[XJS19] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10965–10974, 2019.

[YCLK20] John Yang, Hyung Jin Chang, Seungeui Lee, and Nojun Kwak. Seqhand:rgb-sequence-based 3d hand pose and shape estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020.

[YGS+18] Shanxin Yuan, Guillermo Garcia-Hernando, Björn Stenger, Gyeongsik Moon, Ju Yong Chang, Kyoung Mu Lee, Pavlo Molchanov, Jan Kautz, Sina Honari, Liuhao Ge, Junsong Yuan, Xinghao Chen, Guijin Wang, Fan Yang, Kai Akiyama, Yang Wu, Qingfu Wan, Meysam Madadi, Sergio Escalera, Shile Li, Dongheui Lee, Iason Oikonomidis, Antonis A. Argyros, and Tae-Kyun Kim. Depth-based 3d hand pose estimation: From current achievements to future goals. In *Proceedings of the IEEE*

Conference on Computer Vision and Pattern Recognition, pages 2636–2645, 2018.

[YLL+20] Siyuan Yang, Jun Liu, Shijian Lu, Meng Hwa Er, and Alex C. Kot. Collaborative learning of gesture recognition and 3d hand pose estimation with multi-order feature analysis. In *ECCV*, 2020.

[YLLY19] Linlin Yang, Shile Li, Dongheui Lee, and Angela Yao. Aligning latent spaces for 3d hand pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[YLX+20] Lixin Yang, Jiasen Li, Wenqiang Xu, Yiqun Diao, and Cewu Lu. Bihand: Recovering hand mesh with multi-stage bisected hourglass networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2020.

[YWL19] Cong Peng Yangang Wang and Yebin Liu. Mask-pose cascaded cnn for 2d hand pose estimation from single color images. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(11):3258 – 3268, November 2019.

[YY19] Linlin Yang and Angela Yao. Disentangling latent hands for image synthesis and pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9877–9886, 2019.

[YYS+17] Shanxin Yuan, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim. Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4866–4874, 2017.

[ZB17] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[ZCL+20] Yumeng Zhang, Li Chen, Yufeng Liu, Wen Zheng, and JunHai Yong. Explicit knowledge distillation for 3d hand pose estimation from monocular rgb. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2020.

[ZCY+19] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[ZDKZ19] Yuxiang Zhou, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Dense 3d face decoding over 2500fps: Joint texture & shape convolutional mesh decoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1097–1106, 2019.

[ZJC+17] J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu, and Q. Yang. A hand pose tracking benchmark from stereo matching. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 982–986, Sep. 2017.

[ZLM+19] Xiong Zhang, Qiang Li, Hong Mo, Wenbo Zhang, and Wen Zheng. End-to-end hand mesh recovery from a monocular rgb image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2354–2364, 2019.

[ZPIE17] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[ZWXW20] Z. Zhao, T. Wang, S. Xia, and Y. Wang. Hand-3d-studio: A new multi-view system for 3d hand reconstruction. In *ICASSP 2020 - 2020 IEEE*

International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2478–2482, 2020.