3D HAND RECONSTRUCTION FROM MONOCULAR CAMERA WITH MODEL-BASED PRIORS

JIAYI WANG

Dissertation zur Erlangung des Grades des

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

Saarbrücken, 2022

Date of Colloquium:	May 3, 2023
Dean of the Faculty:	Prof. Dr. Jürgen Steimle
Chair of the Committee:	Prof. Dr. Philipp Slusalleks
Reviewers:	Prof. Dr. Christian Theobalt
	Prof. Dr. Dan Casas
	Prof. Dr. Jürgen Steimle
Academic Assistant:	Dr. Diogo Luvizon

To my wife Mareike, for her love, patience, and understanding.

ABSTRACT

As virtual and augmented reality (VR/AR) technology gains popularity, facilitating intuitive digital interactions in 3D is of crucial importance. Tools such as VR controllers exist, but such devices support only a limited range of interactions, mapped onto complex sequences of button presses that can be intimidating to learn. In contrast, users already have an instinctive understanding of manual interactions in the real world, which is readily transferable to the virtual world. This makes hands the ideal mode of interaction for down-stream applications such as robotic teleoperation, sign-language translation, and computer-aided design.

Existing hand-tracking systems come with several inconvenient limitations. Wearable solutions such as gloves and markers unnaturally limit the range of articulation. Multi-camera systems are not trivial to calibrate and have specialized hardware requirements which make them cumbersome to use. Given these drawbacks, recent research tends to focus on monocular inputs, as these do not constrain articulation and suitable devices are pervasive in everyday life.

3D reconstruction in this setting is severely under-constrained, however, due to occlusions and depth ambiguities. The majority of state-of-the-art works rely on a learning framework to resolve these ambiguities statistically; as a result they have several limitations in common. For example, they require a vast amount of annotated 3D data that is labor intensive to obtain and prone to systematic error. Additionally, traits that are hard to quantify with annotations - the details of individual hand appearance - are difficult to reconstruct in such a framework. Existing methods also make the simplifying assumption that only a single hand is present in the scene. Two-hand interactions introduce additional challenges, however, in the form of inter-hand occlusion, left-right confusion, and collision constraints, that single hand methods cannot address.

To tackle the aforementioned shortcomings of previous methods, this thesis advances the state-of-the-art through the novel use of model-based priors to incorporate hand-specific knowledge. In particular, this thesis presents a training method that reduces the amount of annotations required and is robust to systemic biases; it presents the first tracking method that addresses the challenging two-hand-interaction scenario using monocular RGB video, and also the first probabilistic method to model image ambiguity for two-hand interactions. Additionally, this thesis also contributes the first parametric hand texture model with example applications in hand personalization.

ZUSAMMENFASSUNG

Virtual- und Augmented-Reality-Technologien (VR/AR) gewinnen rapide an Beliebtheit und Einfluss, und so ist die Erleichterung intuitiver digitaler Interaktionen in 3D von wachsender Bedeutung. Zwar gibt es Tools wie VR-Controller, doch solche Geräte unterstützen nur ein begrenztes Spektrum an Interaktionen, oftmals abgebildet auf komplexe Sequenzen von Tastendrücken, deren Erlernen einschüchternd sein kann. Im Gegensatz dazu haben Nutzer bereits ein instinktives Verständnis für manuelle Interaktionen in der realen Welt, das sich leicht auf die virtuelle Welt übertragen lässt. Dies macht Hände zum idealen Werkzeug der Interaktion für nachgelagerte Anwendungen wie robotergestützte Teleoperation, Übersetzung von Gebärdensprache und computergestütztes Design.

Existierende Hand-Tracking Systeme leiden unter mehreren unbequemen Einschränkungen. Tragbare Lösungen wie Handschuhe und aufgesetzte Marker schränken den Bewegungsspielraum auf unnatürliche Weise ein. Systeme mit mehreren Kameras erfordern genaue Kalibrierung und haben spezielle Hardwareanforderungen, die ihre Anwendung umständlich gestalten. Angesichts dieser Nachteile konzentriert sich die neuere Forschung tendenziell auf monokularen Input, da so Bewegungsabläufe nicht gestört werden und geeignete Geräte im Alltag allgegenwärtig sind.

Die 3D-Rekonstruktion in diesem Kontext stößt jedoch aufgrund von Okklusionen und Tiefenmehrdeutigkeiten schnell an ihre Grenzen. Die Mehrheit der Arbeiten auf dem neuesten Stand der Technik setzt hierbei auf ein ML-Framework, um diese Mehrdeutigkeiten statistisch aufzulösen; infolgedessen haben all diese mehrere Einschränkungen gemein. Beispielsweise benötigen sie eine große Menge annotierter 3D-Daten, deren Beschaffung arbeitsintensiv und anfällig für systematische Fehler ist. Darüber hinaus sind Merkmale, die mit Anmerkungen nur schwer zu quantifizieren sind – die Details des individuellen Erscheinungsbildes – in einem solchen Rahmen schwer zu rekonstruieren. Bestehende Verfahren gehen auch vereinfachend davon aus, dass nur eine einzige Hand in der Szene vorhanden ist. Zweihand-Interaktionen bringen jedoch zusätzliche Herausforderungen in Form von Okklusion der Hände untereinander, Links-Rechts-Verwirrung und Kollisionsbeschränkungen mit sich, die Einhand-Methoden nicht bewältigen können.

Um die oben genannten Mängel früherer Methoden anzugehen, bringt diese Arbeit den Stand der Technik durch die neuartige Verwendung modellbasierter Priors voran, um Hand-spezifisches Wissen zu integrieren. Insbesondere stellt diese Arbeit eine Trainingsmethode vor, die die Menge der erforderlichen Annotationen reduziert und robust gegenüber systemischen Verzerrungen ist; es wird die erste Tracking-Methode vorgestellt, die das herausfordernde Zweihand-Interaktionsszenario mit monokularem RGB-Video angeht, und auch die erste probabilistische Methode zur Modellierung der Bildmehrdeutigkeit für Zweihand-Interaktionen. Darüber hinaus trägt diese Arbeit auch das erste parametrische Handtexturmodell mit Beispielanwendungen in der Hand-Personalisierung bei.

ACKNOWLEDGMENTS

One of the most rewarding aspects of my PhD is having had the privilege to work with and get to know so many amazing people. It is hard to adequately acknowledge all their contributions, but I will try my best here.

I would like to thank my supervisor Christian Theobalt for giving me the opportunity to pursue this degree in the first place. Your guidance and support has nurtured my growth as a researcher, and the group you have built provided the environment to explore and develop ideas together. On that note, I want to thank all members of the Visual Computing and Artificial Intelligence (VCAI) and the Computer Graphics Department, past and present. You have not only inspired me with all your works, but also helped me celebrate the good times and get past the not-so-good. Our tradition of after-lunch coffee break is one of the things I will miss most, moving forward. My sincerest thanks (and apologies) to administrative staff members, Sabine Budde and Ellen Fries, and Information Services and Technology (IST), for handling my confused requests, and for then showing me how to accomplish what I actually wanted. I would like to thank postdocs Florian Bernard and Dan Casas for lending me your time and expertise. I have the greatest appreciation for the discussion and advice you have given me over the years. A big shout-out to all of my dedicated collaborators: Diogo Luvizon, Adam Kortylewski, Suzanne Sorli, Oleksandr Sotnychenko, Neng Qian, and Miguel Otaduy. A special thank you to Franziska Müller for being my closest collaborator and kind office mate. I am deeply grateful for the knowledge you shared and the friendship you have offered. Many thanks to Jürgen Steimle, and Dan Casas for taking the time to be part of my thesis committee, and also to my proofreaders: Mallikarjun B R, Gereon Fox, Franziska Müller, Ikhsanul Habibie, Soshi Shimada, Edith Tretschk, and Viktor Rudnev.

Finally, I thank my family and friends for your love and encouragement, not only during my PhD but throughout my life.

Thank you all for making this dissertation possible.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 MOTIVATION

Interacting with the digital world has become a natural part of daily life, and with the rising popularity of virtual reality (VR) and augmented reality (AR) technology, digital interactions are on the cusp of being brought into the 3D space. Intuitive interaction in this novel space, however, is still an open challenge. Conventional interfaces like keyboard and mouse are unsuitable since they rely on metaphors that are constrained to text or 2D. In real world environments, instinctive 3D interactions already exist through manual manipulation. Therefore, bringing the users' hands into the digital space can extend human's innate mastery of natural surroundings into this new, digital frontier. This transfer would be critical for enabling diverse applications such as VR/AR training, teleconference, robotic telepresence, and computer-aided design.

In order to democratize these emerging technologies, hand tracking methods should have low hardware requirements and be easy to set up. Solutions like data or motion capture gloves require expensive specialized hardware that is hard to personalize. They also inhibit natural articulation and interaction due to the bulkiness of the sensors. Static multi-view systems remove these physical constraints, but involve complex calibration for set-up. They additionally limit the usable area to a predefined capture volume, constraining their use case to controlled in-door environments. Recent head-mounted multi-view systems are more flexible due to their portability. However, they still require specialized hardware to reliably capture, transmit, and process high volumes of data.

In comparison, monocular systems are cheap, flexible, and ubiquitous; this make them the ideal input modality for hand reconstruction. Still, many technical challenges arise from this simplified setup. Single camera systems already suffer from depth ambiguities, and this is made worse by the severe occlusion encountered during hand articulation and interaction. Hands also possess high degrees of freedom, while having self-similar geometry and texture. Together, these traits make 3D reconstruction of hands underconstrained and error prone.

Many recent approaches demonstrated promising results by using neural networks and tackling this ambiguous problem statistically (Boukhayma et al., 2019; Tompson et al., 2014; Zimmermann and Brox, 2017). These

methods learn a mapping between image and 3D pose by directly penalizing the pose error over a training dataset. Despite their successes, still many limitations remain that have not been addressed. These approaches cannot, for example, reconstruct finer details that are difficult to quantify, such as hand texture. At the same time, they require large-scale datasets that are either labor intensive to annotate manually, or require automated annotation that is prone to systematic errors and biases. In the latter case, training with biased data would cause these methods to reproduce the errors at inference time.

Most recent methods also only tackle reconstructing a single hand in free space or with a rigid object. Yet naturally people communicate and interact with *both* hands in every day life. Generalizing these approaches to two-hand interactions is not trivial, due to more complex articulation, left-right confusion, inter-hand occlusions, and collision resolution. Highly ambiguous inputs also often occur in these scenarios and multiple plausible reconstructions exist with wildly different poses. Existing methods cannot quantify this ambiguity, and instead deterministically output a single reconstruction that may be far from the ground truth.

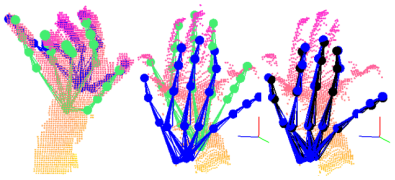
The goal of this thesis is to overcome these limitations through the use of hand-specific priors, and experimentally show how new priors and different methods of incorporating them pushes forward the state-of-the-art. To this end, the thesis presents novel hand capture methods with reduced annotation requirements, while correcting annotation biases. It also presents the first method to track two hand interactions from RGB video in real time, as well as a way to estimate pose distribution of plausible interactions from a single ambiguous RGB image. Finally the thesis presents the first hand appearance prior and demonstrates its advantages in hand texture personalization tasks.

1.2 OVERVIEW

This thesis examines different ways in which prior knowledge about the hands can be incorporated into neural networks.

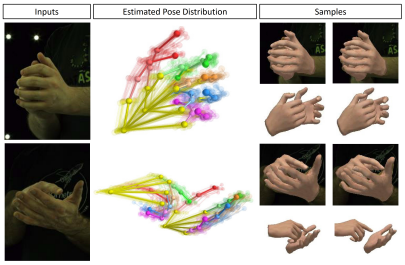
In the setting of single hand pose estimation using a monocular depth camera, the thesis shows how a volumetric model of the hand can be used as a self-supervised loss with the help of differentiable image-formation in chapter 4. This prior knowledge of hand geometry allows a method to make use of the depth map itself as the supervision signal, which reduces the need for difficult-to-acquire 3D annotations and helps overcome annotation bias to which existing methods overfit.

The thesis then tackles the task of two-hand tracking using monocular RGB input. This setting broadens the applicability of the methods, since

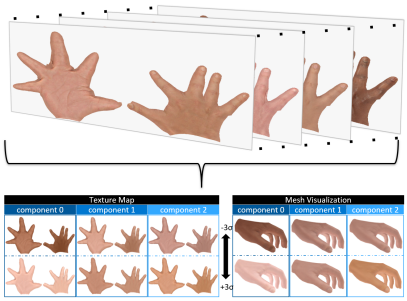


(a) Chapter 4: Self-supervision enables better predictions (green) than biased annotations (blue) and existing methods (black).

(b) Chapter 5: Real-time reconstruction of two interacting hands from monocular RGB video.



(c) Chapter 6: Estimating distributions of plausible two hand reconstructions that are consistent with ambiguous RGB input.



(d) Chapter 7: Parametric texture model as a prior for reconstructing hand appearance for model personalization.

Figure 1.1: The methods of this thesis use various hand priors (geometric, ambiguity, appearance) to improve upon the state of the art.

RGB cameras are more ubiquitous and less sensitive to ambient infrared light. Tracking both hands also allows for more natural communication and interaction for the user. Additional challenges such as depth ambiguity and larger appearance variation need to be accounted for when using such RGB input, however, and left-right disambiguation and inter-hand occlusions need to be addressed with two hands in the scene.

These challenges make the self-supervision approach presented in chapter 4 difficult in particular. In chapter 5, this thesis shows instead how prior knowledge can be incorporated in a second stage as a model-fitting step, as long as a neural network is trained to regress carefully designed image-level features. These features efficiently encode hand segmentation, dense correspondence, 2D keypoints, hand articulation, and root offset, and their use enables the development of the first real-time RGB-based two-hand tracker.

Although the strategy of integrating geometric hand priors allows the method to arrive at a *plausible* solution, the *actual* pose may be very different when the input exhibits extreme ambiguity due to, for example, whole-hand occlusions. In chapter 6, this thesis introduce a method which seeks to quantify this ambiguity and explore the limits of what

information is available in a single RGB image, by explicitly modeling the range of possible poses as a statistical prior. Rather than a single pose, the method estimates a pose distribution supervised by 2D image consistency losses and by a novel regularization term for encouraging 3D diversity of the pose samples. To train and evaluate such a method, the thesis additionally contributes the first dataset with multiple plausible annotations called MultiHands. As will be shown, this approach outperforms existing methods in capturing pose variability, and the output distribution can also be exploited for downstream tasks such as viewpoint selection in a multi-view setup.

Finally, this thesis contributes the first parametric hand texture model in chapter 7 which enables the reconstruction of hand appearance. This model serves as a hand appearance prior, and its application as a loss for hand personalization was demonstrated for both model fitting and learning. This can be further used to generate synthetic data with realistic appearance, which was used in chapter 5.

1.3 STRUCTURE

The ideas of this thesis will be presented in 8 chapters:

- Chapter 1 motivates the topic of hand reconstruction, and gives an overview of the specific challenges this thesis will tackle. It lays out the structure of the thesis and highlights the main technical contributions.
- Chapter 2 summarizes the previous state-of-the-art and places the contributions of this work in context.
- Chapter 3 discusses the different hand modeling techniques, which serve as foundational background needed to understand the thesis.
- Chapters 4, 5, and 6 introduce novel methods that integrate hand specific priors into a learning-based system for 3D hand reconstruction, and provide extensive experimental results to demonstrate the advantages of the proposed model integration.
- Chapter 7 presents the first parametric hand appearance model, and demonstrates its application for hand personalization.
- Chapter 8 summarizes the contributions of this thesis and discusses what future challenges still need to be addressed.

1.4 CONTRIBUTIONS

The main contributions of this thesis are summarized in the following.

The contributions of Chapter 4 (published as Wang et al. (2020b)) are:

- A new self-supervised approach that incorporates a model-based analysis-by-synthesis loss to reduce annotation requirements for 3D hand pose estimation from depth maps.
- Experimental evaluation demonstrating how the proposed loss can overcome biases in annotation, and produce predictions that fit the depth maps better than the "ground truth".

The contributions of Chapter 5 (published as Wang et al. (2020a)) are:

- The first real-time tracking system to reconstruct two interacting hand meshes from monocular RGB video.
- The method uses a novel scalable representation of dense 3D geometry that enables a model fitting formulation for reconstruction. This allows the method to combine advantages of both model-based and learning-based approaches.

The contributions of Chapter 6 (published as Wang et al. (2022)) are:

- The first method to estimate the pose distribution from a single RGB image of interacting hands for modeling pose ambiguity.
- A new benchmark dataset MultiHands designed for evaluating predicted pose distributions by providing multiple plausible annotations per image. A new evaluation protocol and metrics are proposed to make use of this dataset for quantitative evaluation of output distributions.

The contributions of Chapter 7 (published as Qian et al. (2020)) are:

- The first parametric hand texture model that can be used to render realistic RGB images of hands.
- Experiments showing applications of the model for hand personalization in both a learning and model-fitting context.

1.5 PUBLICATIONS

The works presented in this thesis are also published in the following publications:

- Jiayi Wang et al. (2020b). "Generative Model-Based Loss to the Rescue: A Method to Overcome Annotation Errors for Depth-Based Hand Pose Estimation." In: *Automatic Face and Gesture Recognition (FG)*. IEEE, pp. 93–100
- Jiayi Wang et al. (Dec. 2020a). "RGB2Hands: Real-Time Tracking of 3D Hand Interactions from Monocular RGB Video." In: *ACM Transactions on Graphics (TOG)* 39.6
- Jiayi Wang et al. (2022). "HandFlow: Quantifying View-Dependent 3D Ambiguity in Two-Hand Reconstruction with Normalizing Flow." In: *Vision, Modeling and Visualization (VMV) Best Paper Honorable Mention*
- Neng Qian et al. (2020). "HTML: A Parametric Hand Texture Model for 3D Hand Reconstruction and Personalization." In: *European Conference on Computer Vision (ECCV)*. Springer

Additional contributions were made to the following publications but are not included as part of this thesis:

- Tarun Yenamandra et al. (2019). "Convex Optimisation for Inverse Kinematics." In: *International Conference on 3D Vision (3DV)*. IEEE, pp. 318–327
- Viktor Rudnev et al. (2021). "EventHands: Real-Time Neural 3D Hand Pose Estimation from an Event Stream." In: *International Conference on Computer Vision (ICCV)*

2 RELATED WORK

Hand reconstruction from a monocular image is an active research area due to its myriad applications. Although many methods exists to tackle the various challenges of the task, the problem is commonly formulated as a search for parameters that best fit the observations given a digital representation of the hand. This review of existing work will first categorize the types of prior information that can be formulated using different hand representations in the literature. It will then discuss how existing methods approach the search in parameter space. Lastly, this review will present an overview of the common simplifying assumptions made about hand interactions, as well as how recent works address the additional challenges that arise from relaxing these assumptions.

2.1 HAND MODELING FOR REAL-TIME RECONSTRUCTION

Due to the importance of real-time performance in many applications, hand representations should enable efficient evaluation of its parameters' fitness to given observations. At the same time, it should be easy to impose additional constraints for narrowing down the parameter space in order to reduce search time. As such, the process of designing a hand model is about finding the best trade-off between reconstruction quality and efficiency, and about how to limit search space without excluding better fitting solutions. This section reviews how existing representations make this trade-off in order to better incorporate additional knowledge of hand articulation, geometry, appearance, and ambiguities.

2.1.1 *Articulation Modeling*

Articulation can be seen as a configuration of joints to describe the deformation of hands as rigid parts. As joint configurations are limited by the hand anatomy, Rehg and Kanade (1994) applied a kinematic model, a chain of rigid transformations with reduced degrees of freedom to incorporate these mechanical constraints. Lin et al. (2000) and Wu et al. (2001) incorporate additional limits on rotation angle range for each individual joint to prevent hyperextension. These rotation parameterization and joint articulation limits formed the foundational model upon which many later models are built.

More recently, methods apply data-driven techniques to formulate additional constraints that capture inter-correlations of rotations between different degrees of freedom. Spurr et al. (2020) models the space of plausible co-articluations as a convex hull, while Romero et al. (2017) construct a statistical model of pose likelihood using principle component analysis (PCA).

2.1.2 *Geometric Modeling*

Although hand articulation provides a coarse view of hand deformations, it does not encode the volumetric extent of the hand. To capture this information, methods often extend the articulation model with additional structure for geometry modeling.

To ensure that such models are tractable, one approach is to heuristically simplify the hand into a small number of volumetric primitives (Oikonomidis et al., 2011; Qian et al., 2014; Sridhar et al., 2013; Tagliasacchi et al., 2015; Tkach et al., 2016). Although the surface extent of these models is fast to evaluate, the geometry they represent is coarse and discretized. Personalization of these models is also over-parameterized, which often leads to implausible geometric configurations.

Another approach is to use polygonal primitives to capture the hand surface in more detail by using a mesh representation. La Gorce et al. (2011) capture surface deformation as a function of the underlying articulation using linear blend skinning (LBS). Blend shape models additionally parameterize surface variations between individuals by interpolating between registered meshes (Khamis et al., 2015) or between their deformations from a template mesh (Romero et al., 2017). This was later extended with PCA shape space and pose-dependent shape correction (Romero et al., 2017), and filled out with anatomically correct bones and muscles (Li et al., 2021, 2022b) to increase the level of detail captured by the model.

The discrete mesh surface can also be made smooth using Loop subdivision surfaces (Khamis et al., 2015; Taylor et al., 2016), Phong surfaces (Shen et al., 2020), and articulated signed distance fields (Taylor et al., 2017). These smooth parameterizations enable efficient approximations of the optimization problem needed to be solved for tracking, resulting in real-time performance even without GPU acceleration.

2.1.3 *Appearance Modeling*

Modeling the hand texture is important for increasing immersion and the sense of "body-ownership" in VR applications, and it can also aid

in tracking in analysis-by-synthesis approaches (see 2.2.1). When hand texture is needed, most existing methods use a texture map or per-vertex coloring where each color value can vary independently (Chen et al., 2021; La Gorce et al., 2011, 2008). This approach provides partial estimates of the observable hand texture when given a pose estimate and an image. However, this over-parameterization results in unrealistic hand textures when pose errors cause background color to be assigned to the hand. It also leaves out details in the unobserved part of the hand. To address these issues, this thesis contributes the first parametric hand appearance model *HTML* in Chapter 7. It better constrains the appearance by providing a low dimensional PCA space for reconstruction.

2.1.4 *Ambiguity Modeling*

Given monocular image input, many ambiguities arise from projective geometry and occlusions in the scene. Although the abovementioned modeling priors can eliminate implausible pose, geometry, or appearance to narrow down the solution space, multiple *plausible* reconstructions could still exist. Few methods explicitly model this ambiguity, and instead deterministically reconstruct a single solution. However, doing so could incorporate undesired bias during reconstruction and does not reflect the inherent uncertainty in the input. For single hand reconstruction from depth map input, two works exist to address this issue: Ye and Kim (2018) use a hierarchy of Gaussian mixtures to model the distribution of joint locations, and Tkach et al. (2017) use a Gaussian distribution in the model parameter space to model both shape and pose ambiguity. Unlike existing works, this thesis tackles the much more ambiguous task of two-hand reconstruction from RGB input in Chapter 6, where plausible variations are too complex to model with a simple Gaussian.

2.2 HAND TRACKING APPROACHES

Given a hand representation and its corresponding parameter space, the approaches to searching for the optimal reconstruction can be broadly categorized into *generative*, *discriminative*, or *hybrid*.

2.2.1 *Generative Approaches*

Generative approaches start with a proposed vector of hand parameters as initialization, and then use analysis-by-synthesis techniques to iteratively update the parameter to fit the image better. The quality of the fit is

defined as an energy that measures how well the hand model matches pixel-wise or heuristic image features. By using optimization algorithms to minimize this energy, generative methods attempt to recover the correct model parameters (La Gorce et al., 2011, 2008; Oikonomidis et al., 2011; Qian et al., 2014; Sridhar et al., 2013; Tagliasacchi et al., 2015; Tkach et al., 2016). However, due to the self-similarity of the hand structures, the energy derived from heuristic image features has many local minima and is thus sensitive to initialization. On the other hand, generative approaches can easily enforce prior information by incorporating losses and constraints during optimization. They also do not require hard-to-acquire training data and annotations, which eliminates dataset bias as a source of error for generalization.

2.2.2 *Discriminative Approaches*

In contrast to generative approaches, purely discriminative methods attempt to infer the correct parameters from the image. These approaches typically make use of learning-based algorithms to automatically discover powerful high-level image features that can be used to map images to hand parameters directly. To train these algorithms, the correct mapping function is optimized over a large scale image dataset with annotations, usually in the form of 3D joint locations.

Many works explored different learning algorithms, architectures, data representations, and training procedures to best make use of the available data (a non-exhaustive list of such methods includes: Ge et al., 2016, 2017; Keskin et al., 2012; Oberweger et al., 2017; Oberweger et al., 2015; Spurr et al., 2021; Tompson et al., 2014; Wu et al., 2018; Xu and Cheng, 2013; Yang and Yao, 2019; Zhao et al., 2020; Zimmermann and Brox, 2017). However, accurate 3D annotations are difficult to obtain manually and automated annotations are limited to multi-view set-ups with a constrained environment. Given this scarcity of data, it is difficult for purely discriminative methods to generalize, as they tend to overfit to the many biases in the dataset. Due to the lack of explicit priors on hand geometry, reconstruction failures also may no longer resemble hands.

2.2.3 *Hybrid Approaches*

Given the complementary advantages and disadvantages of the two previous approaches, it is natural to investigate how to combine them into a single hybrid approach. One way is to use a discriminative approach to extract interpretable image features, such as 2D finger tip and joint positions (Pavlakos et al., 2019; Shen et al., 2020; Taylor et al., 2016; Ye

et al., 2016), part segmentation (Sridhar et al., 2015), or dense correspondence (Mueller et al., 2019), and to use these to formulate an energy in an optimization framework. Another is to use model parameters as a constrained output space, and make use of the prior terms in the optimization energy as losses to train the image-to-pose mapping function (Boukhayma et al., 2019; Malik et al., 2017; Zhou et al., 2016; Zimmermann et al., 2019). Of these methods, some additionally incorporate an efficient image formation model that enables self-supervision on image input to reduce requirements on training data (Chen et al., 2021; Dibra et al., 2017; Wan et al., 2019).

The methods presented in this thesis further advance this space of hybrid designs to demonstrate their application for overcoming annotation bias (Chapter 4), appearance reconstruction (Chapter 7), tracking two-hand interactions from RGB videos (Chapter 5), and estimating pose ambiguity (Chapter 6).

2.3 HAND INTERACTION

Accounting for the presence of objects during tracking is important for many applications that focus on interactions. However, the majority of the works discussed so far investigate the reconstruction of a single hand in free air. In contrast, capturing hand interactions with arbitrary objects is more challenging due to the immense range of variety in both object appearance and geometry. To simplify the problem, existing methods either reconstruct only the hand pose (Armagan et al., 2020; Hasson et al., 2019; Mueller et al., 2018, 2017; Rogez et al., 2015), or restrict the object to a set of predefined model class (Doosti et al., 2020; Grady et al., 2021; Hampali et al., 2020; Hasson et al., 2020; Karunratanakul et al., 2020; Kyriazis and Argyros, 2014; Liu et al., 2021; Sridhar et al., 2016; Tekin et al., 2019; Tzionas et al., 2016).

Methods from this thesis tackle the even more challenging scenario where the object is known to be another hand. Compared to the object classes addressed using aforementioned methods, another hand is more complex since it can articulate and both hands have similar image features. While some methods (Han et al., 2018; Oikonomidis et al., 2012; Simon et al., 2017) leverage multi-view input to better constrain the problem, most recent works use a single depth map (Kyriazis and Argyros, 2014; Mueller et al., 2019; Taylor et al., 2016, 2017) to reduce the need for calibration and setup. To simplify the input requirements further, this thesis presents (in Chapter 5) the first methods to reconstruct two interacting hands from only monocular RGB video by using a hybrid approach.

Concurrently, Moon et al., 2020 developed a discriminative method trained using the first real dataset with large scale 3D annotations. Later discriminative methods (Fan et al., 2021; Kim et al., 2021) take into account part visibility or segmentation to account for challenging occlusions. To enable surface reconstruction, other methods formulated the output space as a parametric model (Zhang et al., 2021) or as mesh vertices (Li et al., 2022a). Both also introduce attention modules to learn none-local relationships within the image. These methods have in common that they deterministically reconstruct the hands. However, during interactions, the image input often exhibits heavy occlusions between hands or ambiguous semantics that allow for a wide range of plausible reconstructions. In Chapter 6, this thesis presents the first method to explicitly model this ambiguity with a probabilistic approach, and contributes the first benchmark dataset designed to evaluate this task.

3 BACKGROUND

This chapter provides foundational concepts used to model hands in this thesis. First, principal component analysis is presented in Section 3.1, and will be used as a widely applicable technique used to restrict the search space of a model. The kinematic skeleton model is then introduced in Section 3.2, which is used to describe the hierarchical nature of hand articulation. This articulation model is then extended with a volume model (Section 3.3), a surface model (Section 3.4), and an appearance model (Section 3.5) to increase the fidelity of the hand representation. Finally, normalizing flow is introduced as a tractable method to parameterize arbitrary distribution (Section 3.6); this is later used to model pose ambiguity.

3.1 PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is a commonly used technique for dimensionality reduction. Given a data representation that is over-parameterized, PCA can be used to discover a subspace that can still accurately explain the observed variations in the dataset.

When given a set of n data points $\{v_i\}_{i=1}^{n}$ with each d-dimensional point $v_i \in \mathbb{R}^d$ representing, for example, the parameters of a hand, PCA first computes the data covariance matrix

$$C = \frac{1}{n-1} \sum_{i=1}^{n} (v_i - \bar{v}) (v_i - \bar{v})^\top , \tag{3.1}$$

where $\bar{v} = \frac{1}{n} \sum_{i=1}^{n} v_i$ is the average hand parameter.

Subsequently, eigenvalue decomposition of $C = \Phi \Lambda \Phi^T$ is used to obtain the principal components $\Phi \in \mathbb{R}^{d \times d}$ and the diagonal matrix of eigenvalues $\Lambda \in \mathbb{R}^{d \times d}$. To reduce the dimensionality of data to $k < d$, a subset of principal components with k largest eigenvalues $\Phi' \in \mathbb{R}^{d \times k}$ is considered. The reduced parameter space can be expressed as $\alpha \in \mathbb{R}^k$ and the corresponding hand parameter v can be recovered using

$$v(\alpha) = \bar{v} + \Phi' \, \Lambda'^{\frac{1}{2}} \, \alpha \, , \tag{3.2}$$

where $\Lambda' \in \mathbb{R}^{k \times k}$ is the matrix of eigenvalues corresponding to Φ'.

This technique is used to create a bone length subspace in Chapter 4, a hand texture subspace in Chapter 7, and a hand pose and shape subspace in Section 3.4.

3.2 KINEMATIC SKELETONS

The hand is a complex anatomical structure with bones, joints, tendons and muscles all interacting to articulate the fingers. Commonly, geometric models of the hand consider only the bones and joints (See Figure 3.1), while the impact of other internal structures is abstracted as articulation constraints. To simplify things further, each finger is described with three joints (DIP, PIP, MCP), while the thumb contains the CMC rather than the PIP joint. The bottom-most joint of each finger is then connected to the radiocarpal joint (wrist), which removes the carpal bones from consideration. The resulting model is composed of 16 joints and 20 bones.

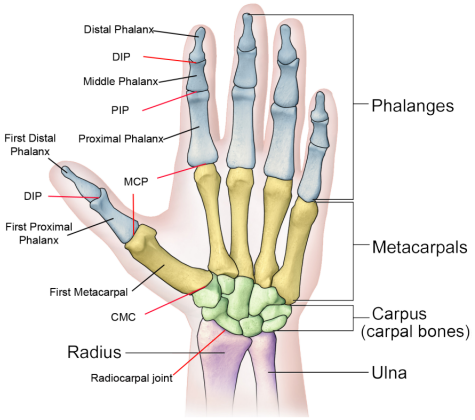


Figure 3.1: An anatomical model of the the hand. Black lines labels the bone and red lines labels the joints. (Figure taken from Wheatland et al., 2015)

A kinematic skeleton parameterizes the locations of joints by representing them as a hierarchy of rigid transformations $T \in \mathrm{SE}(3)$ organized in a tree structure. Here each node corresponds to a joint, and an edge exist between two nodes whose joints are connected by a bone. The root of the tree encodes the global transformation of the root joint, which is often chosen to be the wrist. From there, each ith child node in the tree contains the transformation T_i that would map a point in the child local coordinate to the parent local coordinate. By convention, the joint is positioned at the origin of its own local coordinate. Thus the joint location of the ith joint in the parent's local coordinate J_i^p can be calculated using

$$J_i^p = T_i \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \tag{3.3}$$

Here, the translation in T_i can be interpreted as the local bone vector in the canonical pose, and the rotation as the local articulation. The global position J_i^g can then be obtained by iteratively applying a chain of parent transformations

$$J_i^g = \left(\prod_{j \in \alpha_i} T_j^l \right) \cdot T_i^l \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \tag{3.4}$$

where α_m denotes the path from the m-th node to the root. Note that the impact of local articulation is efficiently passed down the kinematic chain when calculating global position.

Since the range of articulation at each joint is limited by tendon and muscle configuration, the transformation can be further constrained. Romero et al. (2017) applied PCA on a captured pose dataset to reduce the degrees of freedom by using a lower dimensional subspace, used in Chapter 5.

Another approach to reduce degrees of freedom is by limiting articulation along 1 or 2 rotation axes (Rehg and Kanade, 1994). This explicitly models the abduction/adduction and flexion/extension capabilities of each joint (See Figure 3.2). For each rotation axis, a valid range of angles can further be defined to model mechanical limits (Lin et al., 2000).

These joint limit constraints are used in Chapter 4. As the fingertip location is often of interest, it can be included in the kinematic skeleton model by treating it as a joint with 0 degrees of freedom for articulation.

Although a kinematic skeleton captures bone articulation, it can not represent the hand surface or volume. In Section 3.3 and 3.4, extensions are presented which address these shortcomings.

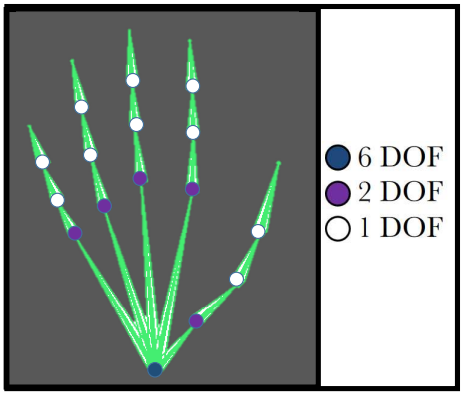


Figure 3.2: A model of hand joints and their degrees of freedom.

3.3 SUM-OF-GAUSSIANS MODEL

The Sum-of-Gaussians model (SoG) was introduced by Stoll et al. (2011) to approximate, among other things, the volumetric extent of the hand (Sridhar et al., 2013). The model uses 3D Gaussians as primitives, which efficiently approximate the space the hand occupies by considering spheres of 1 standard deviation radius. These primitives are then attached rigidly to the bones, so that they can articulation with the joints (See Figure 3.3).

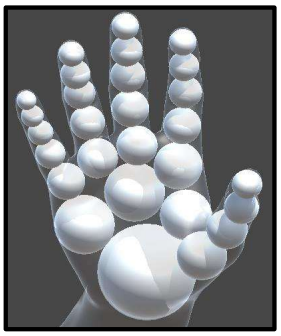


Figure 3.3: A volumetric model of the hand

The choice of Gaussian as primitives serves three advantages: First, the projection of 3D Gaussians to the image plane can be found analytically. This provides an efficient image formation model for an SoG hand. Second, there is also an analytical solution to the overlap between two Gaussians, which enables tractable computation of similarity or dissimilarity losses. Lastly, when considering the similarity between two SoG models, the infinite extent of the Gaussians implicitly approximates a soft closest-point correspondence assignment. This provides differentiable correspondence that is simultaneously optimized along with the similarity loss.

To see how these three properties are derived, consider the 3D model isotropic Gaussian $g_{\mu_h,\sigma_h}(x)$ parameterized by the mean μ_h and the standard deviation σ_h. Given an intrinsic camera matrix K, the projected 2D Gaussian is $g_{\mu_p,\sigma_p}(x) = \Pi_K(g_{\mu_h,\sigma_h}(x))$, where

$$\mu_p = \frac{K \cdot \mu_h}{[\mu_h]_z}, \tag{3.5}$$

$$\sigma_p = \frac{\sigma_h f}{[\mu_h]_z}. \tag{3.6}$$

Here, f is the focal length of K, and $[\mu_h]_z$ is the z component of μ_h.

The overlap $S_{h,k}$ between two d-dimensional isotropic Gaussians $g_{\mu_h,\sigma_h}(x)$ and $g_{\mu_k,\sigma_k}(x)$ can be found using

$$\begin{aligned} S_{h,k} &= \int_{\mathbb{R}^d} g_{\mu_h,\sigma_h}(x) \cdot g_{\mu_k,\sigma_k}(x)\, dx \\ &= \frac{\sqrt{(2\pi)^d(\sigma_h^2\sigma_k^2)^d}}{\sqrt{(\sigma_h^2+\sigma_k^2)^d}} \exp\left(-\frac{||\boldsymbol{\mu}_h - \boldsymbol{\mu}_k||_2^2}{2(\sigma_h^2+\sigma_k^2)}\right). \end{aligned} \tag{3.7}$$

This analytical solution is differentiable with respect to the mean and can be minimized to avoid collision by moving apart the two Gaussian

centers. When a 2D Gaussian image representation is available, its overlap with the projected 2D model Gaussian can be maximized to improve how well the model fits the image.

When considering the overlap between two SoG models, the total overlap S_{total} can be found using

$$\begin{aligned} S_{\text{total}} &= \sum_{h=1}^{N_h} \sum_{k=1}^{N_k} S_{h,k} \\ &= \sum_{h=1}^{N_h} \sum_{k=1}^{N_k} \frac{\sqrt{(2\pi)^d (\sigma_h^2 \sigma_k^2)^d}}{\sqrt{(\sigma_h^2 + \sigma_k^2)^d}} \exp\left(- \frac{||\boldsymbol{\mu}_h - \boldsymbol{\mu}_k||_2^2}{2(\sigma_h^2 + \sigma_k^2)} \right). \end{aligned} \tag{3.8}$$

The exponential term $w_{h,k} = \exp(-\frac{||\boldsymbol{\mu}_h - \boldsymbol{\mu}_k||_2^2}{2(\sigma_h^2 + \sigma_k^2)})$ can be seen as a distance base weighing between a pair of Gaussians. As it assigns higher weights to closer pairs than farther ones, it can be interpreted as a soft differentiable approximation to closest-point correspondence.

In Chapter 4, the SoG hand model is used to formulate a self-supervised loss to train a pose estimator using depth map input. In Chapter 5, it is used in combination with a mesh model as an efficient collision proxy.

3.4 HAND MESH

A mesh representation approximates a surface by discretizing it into as a collection of polygonal primitives, usually triangles (see Figure 3.4). It is usually stored as a graph, which note the vertex position and connectivity. Compared to a SoG model, a mesh can represent detailed surface at the cost of using more primitives.

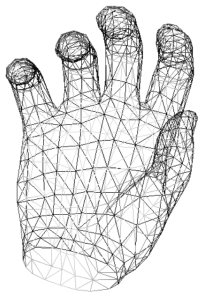


Figure 3.4: A mesh model of the hand

In order to use hand meshes for reconstruction, the surface variations due to articulation and hand shapes must be parameterized. A low dimensional blend-shape space can be introduced to constrain the deformation so that only natural variations in hand shape is modeled (Khamis et al., 2015; Romero et al., 2017). Given a template mesh $\mathbf{T}$, a blend shape basis is defined as a set of per-vertex deformations $\mathbf{S}_n$ from the template, which is often created using a dataset of 3D scans with varying hand shapes. Additional deformations is then represented as a linear combination of blend shape basis. This repa-

rameterization of hand shape $\tilde{\mathbf{T}}$ in terms of the linear weights β can be expressed as

$$\tilde{\mathbf{T}}(\beta) = \mathbf{T} + \sum_{n=1}^{|\beta|} \beta_n \mathbf{S}_n \,. \tag{3.9}$$

Romero et al. (2017) introduced a pose blend shape function to further account for deformations of the mesh as a result of articulation. The final hand template is then defined as

$$\hat{\mathbf{T}}(\beta, \theta) = \tilde{\mathbf{T}}(\beta) + \sum_{n=1}^{k} \alpha_n(\theta) \mathbf{P}_n \,, \tag{3.10}$$

where $\mathbf{P}_n$ is the set of pose blend shape basis, k is the size of this set, θ encodes the articulation of the joints and $\alpha_n(\theta)$ are pose-dependent weights (see Romero et al. (2017) for details of how $\alpha_n(\theta)$ is defined). To articulate the template, linear blend skinning (LBS) is performed (Lewis et al., 2000), i.e.

$$\mathcal{X}(\beta, \theta) = LBS(\hat{\mathbf{T}}(\beta, \theta), J(\beta), \theta, \mathbf{W}) \,, \tag{3.11}$$

where $\mathcal{X}$ is the vertex positions of the articulated mesh, $J(\cdot)$ computes the 3D position of the hand joints in the template, and $\mathbf{W}$ is a matrix of rigging weights used by the skinning function.

One popular parametric mesh model is MANO (Romero et al., 2017). MANO built its blend shape basis by using PCA on 1,000 scans of 30 different subjects in a variety of poses. The result is a mesh with 778 vertices, 1,538 triangular faces, and 16 joints that can be deformed using shape parameters $\beta \in \mathbb{R}^{10}$ and the pose parameters $\theta \in \mathbb{R}^{45}$.

Chapter 7 extends MANO with an appearance model to enable simultaneous reconstruction of hand pose and appearance. MANO is also used in Chapter 5 and 6 for hand surface parameterization.

3.5 HAND TEXTURE

In order to represent also the visual appearance of the hand and not just the geometry, the mesh model needs to be extended with coloration. Although it is possible to define per-vertex color on the mesh itself and use barycentric interpolation to propagate the color to the rest of the faces, the texture detail will be limited by the coarseness of the mesh.

Instead, a highly detailed 2D UV image $\Omega : w \rightarrow c$ can be defined to represent a flattened surface of the hand (See Figure 3.5). Here, $w = (u, v)$ is the 2D pixel coordinate of the UV image, and $c \in [0, 1]^3$ is the color value at the pixel. Then a texture mapping function $F : v \rightarrow w$ can be defined by assigning every mesh vertex v to a UV coordinate w.

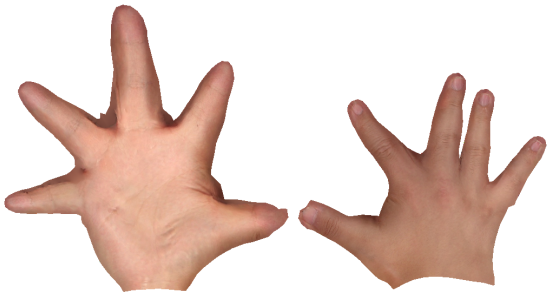


Figure 3.5: A example UV texture image.

Therefore, the vertex color can be found by compositing the two functions $c(v) = \Omega(F(v))$. To find the color of a point $p \in \mathcal{R}^3$ on a triangle face defined by vertices v_1, v_2, v_3, the barycentric coordinate of the point $b(p) = (\lambda_1, \lambda_2)$ can be used to interpolate the UV coordinates of the corresponding vertices, i.e.

$$c(p) = \Omega(\lambda_1 F(v_1) + \lambda_2 F(v_2) + (1 - \lambda_1 - \lambda_2) F(v_3)) \,. \tag{3.12}$$

Since the interpolation happens in the UV coordinate rather than in color space, the high frequency details in the UV image are preserved through the mapping.

Although texture mapping can create detailed appearances even on a coarse hand mesh, the UV image itself is over-parameterized and a represent arbitrary images. Chapter 7 presents a low dimensional texture model to limit the UV image to statistically likely hand textures. This is used to extend MANO for hand appearance personalization tasks.

3.6 NORMALIZING FLOW

Due to the many ambiguities present in monocular 3D reconstruction, many solutions in the parameter space of the model could explain the image equally well. It is useful to model this space of plausible solutions as a distribution, so that the likelihood of a reconstruction can be queried. To find the best matching distribution, a search space must be defined by finding a way to parameterize potentially complicated distributions; one such parameterization is available through the use of a normalizing flow network.

A normalizing flow describes a target distribution $p_Y(\mathbf{y})$ as sequential invertible transformations $f : \mathbb{R}^d \to \mathbb{R}^d$ of a simple probability density $p_Z(\mathbf{z})$, i. e.

$$p_Y(\mathbf{y}) = p_Z(f^{-1}(\mathbf{y})) \left| \det \frac{\partial f^{-1}(\mathbf{y})}{\partial \mathbf{y}} \right| . \tag{3.13}$$

This can be rewritten by using $\mathbf{z} = f^{-1}(\mathbf{y})$ and the inverse function theorem as

$$p_Y(\mathbf{y}) = p_Z(\mathbf{z}) \left| \det \frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} \right|^{-1} . \tag{3.14}$$

Typically, a multivariate normal distribution $\mathcal{N}(\mathbf{0}, I)$ is used as the base distribution p_Z.

When carefully choosing the building blocks, a neural network can represent a transformation f that fulfills all the assumptions made. In this way, the target distribution is parameterized as learned weights in a normalizing flow network. Compared to other parameterizations (such as GANs and VAEs), normalizing flow provides a tractable way to both sample from the distribution, and estimate the probability of a given sample. This allows for an efficient way to compute loss both on the sampled pose space, and in terms of the sample likelihood. For a more detailed overview, refer to Kobyzev et al., 2020.

In order to model varying distributions when given different ambiguous observations, a conditioning input can be incorporated into this framework. Normalizing flow can be extended to conditional normalizing flow (Winkler et al., 2019) by using transformations $f_{\mathbf{x}} : \mathbb{R}^d \to \mathbb{R}^d$ that are parameterized by $\mathbf{x}$, so that for $f(\cdot; \mathbf{x}) := f_{\mathbf{x}}(\cdot)$ the following results are derived:

$$p_{Y|X}(\mathbf{y}|\mathbf{x}) = p_{Z|X}(f_{\mathbf{x}}^{-1}(\mathbf{y})|\mathbf{x}) \left| \det \frac{\partial f_{\mathbf{x}}^{-1}(\mathbf{y})}{\partial \mathbf{y}} \right| \tag{3.15}$$

$$= p_{Z|X}(\mathbf{z}|\mathbf{x}) \left| \det \frac{\partial f_{\mathbf{x}}(\mathbf{z})}{\partial \mathbf{z}} \right|^{-1} . \tag{3.16}$$

This reduces the search for a best fitting distribution to the image to a search for the best conditioning input instead.

This parameterization is used in Chapter 6 to model the pose ambiguity in hand interaction reconstruction from a monocular RGB image.

4

MODEL-BASED SELF-SUPERVISION FOR HAND POSE ESTIMATION

This chapter presents a model-based generative loss for training hand pose estimators on depth images (published as Wang et al. (2020b)). The additional loss allows training of a hand pose estimator that accurately infers the entire set of 21 hand keypoints while only using supervision for 6 easy-to-annotate keypoints (fingertips and wrist). It is shown that the partially-supervised method achieves results that are better than those of fully-supervised methods like Malik et al. (2017). Moreover, it is demonstrated for the first time that such an approach can mitigate the effects of erroneous annotations, i.e. "ground truth" with notable measurement error, during training. As a result, the predictions during inference are able to explain the inputs better than the given "ground truths" (see Figure 4.1).

4.1 INTRODUCTION

Accurate hand-pose estimation from monocular depth images is vital for enabling fine-grained control in human-computer interaction in VR and AR settings (Soliman et al., 2018). However, this is a challenging task due to, e.g. , complex poses, self-similarities, and self-occlusions.

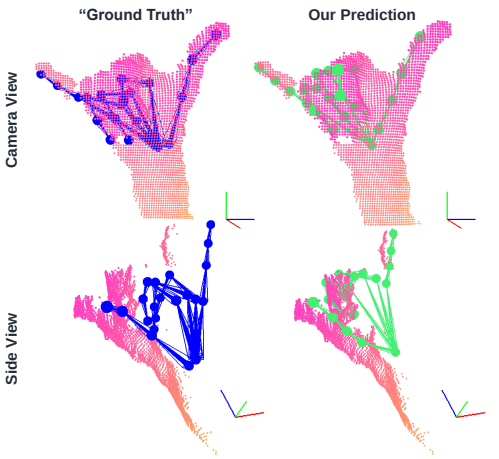


Figure 4.1: The proposed method compensates for erroneous "ground truths" (Blue), resulting in better predictions (Green).

Many existing methods address these challenges with powerful learning-based tools. Such methods dominate the benchmarks on large public datasets such as NYU (Tompson et al., 2014), and Hands in the Million Challenge (HIM) (Yuan et al., 2017). Most of these approaches are trained in a fully supervised manner to predict keypoint positions in 3D. However, the current lack of large-scale training datasets that are both

accurate and diverse causes such methods to overfit. This makes it difficult to generalize well to new settings, or even across benchmarks (Yuan et al., 2017). Retraining these methods on different data requires the full set of 21 keypoint annotations, which are tedious to obtain. More importantly, the annotation process is prone to errors, either due to systematic biases during measurement, or due to human errors. Additionally, methods that learn a direct mapping from depth image to keypoints often ignore the inherent geometry of the hands, such as constant bone lengths or joint angle limits. As such, albeit their general good performance, these methods often produce bio-mechanically implausible poses (Wöhlke et al., 2018).

An alternative to learning-based approaches are model-based hand tracking methods, such as Sridhar et al. (2015), Taylor et al. (2017), and Tkach et al. (2017), among others. These methods use generative hand models to recover the pose that best explains the image through an analysis-by-synthesis strategy. While not suffering from anatomical inconsistencies, and generalizing better to yet-unseen scenarios, they require good initialization of the model parameters in order to minimize the non-convex energy function.

This chapter addresses the shortcomings of both approaches with a generative model-based loss embedded into a learning-based method. Based on a volumetric Gaussian hand model, this loss incorporates additional annotation-free self-supervision from the depth image. When combined with anatomical priors, this supervision can take the place of joint annotations for resolving both hand pose and bone length ambiguities. In total, this approach reduces the number of required annotations from 21 to 6, a 71% decrease. At the same time, the learning-based framework enables accurate and efficient inference without requiring initialization. This effectively combines the main advantages of the two popular categories.

Most existing methods that utilize model-based losses (Malik et al., 2017; Wöhlke et al., 2018; Zhou et al., 2016) do not explain the input images in a generative manner. As such, they still require the full set of 21 annotated keypoints. Additionally, due to the reliance on the annotations as the only source of supervision, these methods can overfit to errors and biases in the annotations. This chapter demonstrate that the proposed method can overcome such errors through the use of proposed additional generative loss (see Figure 4.1).

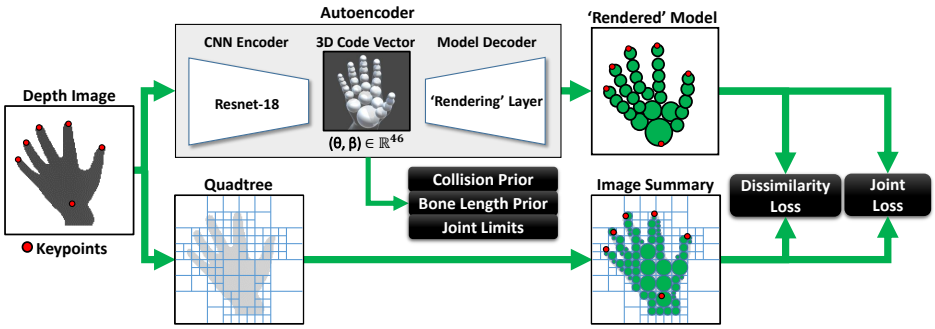


Figure 4.2: **Framework Overview.** The method regresses a code vector representing the parameters of a volumetric Gaussian hand model. It is supervised with a dissimilarity loss, which compares the model to a Gaussian image representation constructed using quadtree, and a joint loss defined on a subset of keypoints. Additional bone lengths and pose prior are used to regularize the encoding.

The main contributions are as follows:

- Compared to classical fully supervised methods, the proposed generative loss significantly reduces the amount of annotations need to accurately infer the full hand pose.
- Despite ambiguities resulting from the reduced annotations, the proposed method can simultaneously infer pose and bone lengths.
- The chapter provide a new dataset, HANDID, which includes fingertips and wrist annotations for 7 users to address the lack of hand shape variations in existing datasets.
- Most importantly, it is demonstrated for the first time that a method trained with a self-supervised loss can produce hand pose that better fits the input than the "ground truth" annotations.

4.2 METHOD

The main idea of the approach is to explain a depth image of a hand based on a generative hand model, cf. Figure. 4.2. Given a depth image as input, the method uses a CNN-based encoder to obtain a low-dimensional embedding of the depth image. The decoder is build upon a parametric hand model that produces a volumetric representation of the hand from the semantically meaningful code vector. By using a suitable representation of the input depth image, the overlap between a "rendering" of the hand representation and the input depth can be used to train the encoder. To be more specific, the volume of the hand are modeled with a collection of 3D Gaussians rigidly attached to a kinematic hand skeleton model. The corresponding image space representation of the hand can be obtained by

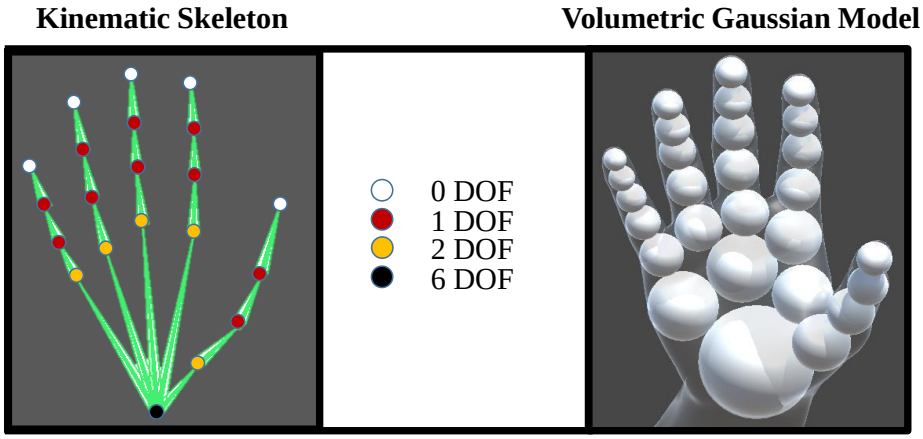


Figure 4.3: **Left:** The skeleton comprises of 20 bones and 15 articulating joints with varying degrees of freedom (DOF). In total, there are 26 joint parameters, and 20 bone length parameters. **Right**: The volumetric Gaussian model.

projecting the Gaussians using the camera intrinsics. Moreover, the depth image can also be reduced to a Gaussian representation that summarizes regions of homogeneous depth obtained with quadtree-decomposition. The similarity between the rendered model and the image can then be described as the depth-weighted overlap of all pairs of model and image Gaussians, and can serve as generative model-based loss during network training. Additional prior losses are added to avoid inter-penetrations of hand parts, violations of joint limits, and unphysiological combinations of bone lengths. Lastly, supervision for a small subset of keypoints is provided as a way to mitigate the multiple minima present in the non-convex energy.

4.2.1 *Hand Model*

Kinematic Skeleton: The shape of the kinematic skeleton is parameterized in terms of bone lengths, and pose as articulation angles with respect to the predefined rotation axes. It comprises of 20 bones with lengths $b \in \mathbb{R}^{20}$ and 26 degrees of freedom (DOF) $\theta \in \mathbb{R}^{26}$ (20 DOF for articulation and 6 DOF for global rotation and translation), see Figure. 4.3.

To ensure that the predicted bone length vector is plausible, b is parameterized by an affine model constructed using 20 PCA basis vectors, i.e.

$$b = b_{\text{avg}} + M_{\text{pca}}\beta \quad . \tag{4.1}$$

Here, $b_{\text{avg}} \in \mathbb{R}^{20}$ is the average bone length vector and $M_{\text{pca}} \in \mathbb{R}^{20\times 20}$ are the linear PCA basis vectors of the bone length variations scaled by the square root of their eigenvalues. By scaling the basis vectors, β

follows an isotropic standard normal distribution, and deviations along each basis are penalized inversely to how much natural variation exists in that direction. Both b_{avg} and M_{pca} are obtained from bone length vectors computed from 10,000 hand meshes sampled from the linear PCA parameters of the MANO model (Romero et al., 2017).

The pose parameter vector θ controls the angles of articulation with respect to the joint axes in the forward kinematics chain, as well as the global translation and rotation of the entire hand, where the latter is parameterized using Euler angles. Given the bone length parameters β and pose θ, the N_j joint positions can be obtained by applying forward kinematics $F(\theta, \beta) \in \mathbb{R}^{N_j \times 3}$.

Volumetric Gaussian Model: Similar to Sridhar et al. (2015) and Stoll et al. (2011), the hand volume is modeled with a mixture of N_m 3D Gaussians, i.e.

$$G_{3D}(x) = \sum_{h=1}^{N_m} g_{\mu_h(\theta,\beta),\sigma_h}(x)\,, \tag{4.2}$$

where g is an isotropic Gaussian with mean $\mu_h(\theta, \beta)$ and standard deviation σ_h. Each Gaussian is attached to a bone on the kinematic skeleton and articulates with that bone.

4.2.2 *Depth Image Representation*

The depth image is represented by a collection of 2D image Gaussian and depth value pairs $\{(g_{\mu_i,\sigma_i}(x), z_i)\}_{i=1}^{N_i}$. Each Gaussian and depth value pair summarizes a roughly homogeneous region with a single depth. To obtain these regions, quadtree clustering is used to recursively divide the image into sub-quadrants until the depth difference within each region is below a threshold c ($c = 20$mm is used for all experiments). The Gaussian $g_{\mu_i,\sigma_i}(x)$, is chosen so that μ_i is the center and σ_i is half the side length of the region. The associated depth value z_i is then the average depth value of the quadrant.

4.2.3 *Model-based Decoder*

To measure the quality of the predicted hand pose and bone length parameters for a given input depth image, a decoder layer is incorporated to "render" the 3D model representation to a 2.5D representation similar to the image representation. The camera-facing surface of the h-th 3D Gaussian is approximated by a projected 2D Gaussian $g_{\mu_p,\sigma_p}(x) = \Pi_K(g_{\mu_h,\sigma_h}(x))$ using the intrinsic camera matrix K and an associated depth value z_p.

4.2.4 *Loss Layer*

For training the network, the loss is decomposed into an unsupervised dissimilarity term E_{dissim} for measuring the discrepancy between depth image and hand model, $E_{\text{collision}}$ to prevent self intersection, E_{bone} for regularizing the bone length parameters β, E_{lim} for regularizing the joint angles θ, and a supervised E_{joint} term for explaining the provided joint locations. The relative importance of each term is balanced with scaling factors λ and the values can be found in Appendix A.1. With that, the total energy reads

$$\begin{aligned} E(\theta,\beta) =& \lambda_{\text{dissim}} E_{\text{dissim}}(\theta,\beta) + \lambda_{\text{collision}} E_{\text{collision}}(\theta,\beta) + \\ & \lambda_{\text{bone}} E_{\text{bone}}(\beta) + \lambda_{\text{lim}} E_{\text{lim}}(\theta) + \lambda_{\text{joint}} E_{\text{joint}}(\theta,\beta)\,. \end{aligned} \tag{4.3}$$

In the following sections, the individual energy terms are described.

4.2.4.1 *Dissimilarity Measure*

To measure the overall similarity between two given (2D Gaussian, depth) tuples, the similarity $S_{i,p}$ between the two Gaussians are weighted by their distance in depth values $\Delta(i,p)$. The pairwise similarity between image Gaussian g_{μ_i,σ_i} and projected model Gaussian g_{μ_p,σ_p} is defined using the integral over the product of the two functions. Since the model Gaussian directly depends on the hand pose vector θ and bone length vector β, $S_{i,p}$ is given by

$$S_{i,p}(\theta,\beta) = \int_{\mathbb{R}^2} g_{\mu_i,\sigma_i}(x)\, g_{\mu_p(\theta,\beta),\sigma_p}(x)\, dx\,. \tag{4.4}$$

To incorporate depth information, $S_{i,p}(\theta,\beta)$ is weighted by the depth difference

$$\Delta(i,p) = \begin{cases} 0, & \text{if } |z_i - z_p| \geq 2\sigma_h \\ 1 - \frac{|z_i - z_p|}{2\sigma_h}, & \text{if } |z_i - z_p| < 2\sigma_h \end{cases}, \tag{4.5}$$

where σ_h is the standard deviation of the unprojected Gaussian g_{μ_h,σ_h} associated with g_{μ_p,σ_p}. This forces the model to not only match the area of the hand in the depth image, but also the observed depth values.

The overall similarity S_{sim} is defined as the sum over all possible pairings between the model and the image Gaussians, and is given by

$$S_{\text{sim}} = \frac{\sum_{i=1}^{N_i} \sum_{p=1}^{N_m} \Delta(i,p) S_{i,p}}{\sum_{i=1}^{N_i} \sum_{k=1}^{N_i} S_{i,k}}\,, \tag{4.6}$$

where the denominator is the self-similarity of the image Gaussians used for normalization. The dissimilarity loss is then defined as $E_{\text{dissim}} = -S_{\text{sim}}$ so it can be used in a loss minimization learning framework.

4.2.4.2 *Collision Prior*

To ensure that the surface represented by the 1σ isosurface of the 3D Gaussians does not (self-)interpenetrate, a repulsive term based on the 3D overlap of the model Gaussians is used. Overloading the notation for the Gaussian overlap $S_{i,j}$ (cf. Equation (4.4)) to denote the similarity between two different model Gaussian components, the loss can be analogously defined

$$E_{\text{collision}} = \sum_{j=1}^{N_m} \sum_{k=j+1}^{N_m} S_{j,k}\,, \tag{4.7}$$

so that Gaussians do not overlap in 3D.

4.2.4.3 *Bone Length Prior*

To keep the bone lengths β plausible, the loss

$$E_{\text{bone}} = ||\beta||_2^2\,, \tag{4.8}$$

can be imposed to penalize the deviation of the predicted bone length parameters from the mean parameter. With that, this term helps to keep the predictions in the high probability region of the normal distribution used in the PCA prior.

4.2.4.4 *Joint Limits*

To keep joint articulations within mechanically and anatomically plausible limits, a joint limit penalty is imposed using

$$E_{\text{lim}} = \sum_{\theta_j \in \theta} \begin{cases} 0, & \text{if } \theta_j^l \leq \theta_j \leq \theta_j^h \\ (\theta_j^l - \theta_j)^2, & \text{if } \theta_j < \theta_j^l \\ (\theta_j - \theta_j^h)^2, & \text{if } \theta_j > \theta_j^h \end{cases}\,, \tag{4.9}$$

where θ_j^l and θ_j^h are the lower and upper limits of θ_j, which are defined based on anatomical studies of the hand (Serra, 2011).

4.2.4.5 *Joint Location Supervision*

An additional supervision loss E_{joint} on a small subset of joint positions $J_1, \ldots, J_{N_s}$ can be applied in order to help the optimizer converge to a good minimum in the overall generative loss function. A combination of 2D and 3D joint location supervisions are used depending on availability. If for a given joint with index j a full 3D supervision is provided, the distance Φ_j between the annotation $J_j \in \mathbb{R}^3$ and the model joint F_j is given by their ℓ_2 distance. If only 2D supervision is provided, Φ_j is the closest ℓ_2 distance between F_j and the ray $\bar{J}_j$ to which the annotation is projected using the camera intrinsics. Hence, Φ_j is defined as

$$\Phi_j = \begin{cases} ||F_j - \langle F_j, \bar{J}_j \rangle \bar{J}_j||_2 \,, & \text{if } J_j \in \mathbb{R}^2 \\ ||F_j - J_j||_2 \,, & \text{if } J_j \in \mathbb{R}^3 \end{cases} , \tag{4.10}$$

where $F_j = F(\theta, \beta)_j$ is the j-th joint obtained from applying forward kinematics with the model parameters.

Due to inaccuracies in the annotation, the ground truth may conflict with the observed image. Hence, the joint loss is modified to account for annotation uncertainty by introducing a "slack" radius $s \in \mathbb{R}_+$ that models the expected uncertainty in millimeters. All predictions within this radius of the ground truth will not be penalized. This allows the encoder to be more robust to erroneous annotations. Together, the joint loss for the subset of N_s joints E_{joint} is defined as

$$E_{\text{joint}} = \sum_{j=1}^{N_s} \begin{cases} 0, & \text{if } \Phi_j \leq s \\ (\Phi_j - s)^2, & \text{if } \Phi_j > s \end{cases} . \tag{4.11}$$

4.3 EXPERIMENTS

Here, the impact of the generative model-based loss is evaluated in terms of pose accuracy and bone length consistency when trained with a reduced set of keypoints. Additionally, the predictions of the proposed method and the erroneous "ground truth" are shown on existing dataset to demonstrate the regularizing effect of the proposed loss against annotation errors.

4.3.1 *Architecture and Training*

The Caffe framework (Jia et al., 2014) is used for implementation of the networks and losses. The Resnet-18 architecture (He et al., 2016) pretrained on ImageNet is used as the encoder to the proposed method. For

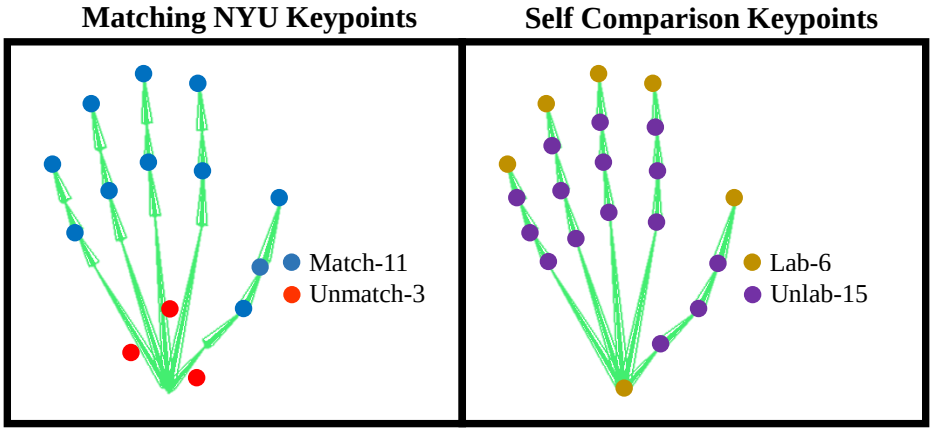


Figure 4.4: **Left:** For comparisons against the state of the art, the proposed model is evaluated on a subset of NYU keypoints (**Match-11**) due to mismatches in skeleton. **Right:** For self-comparison, evaluation was performed on 21 keypoints (**All-21**), 6 of which have supervision (**Lab-6**), and 15 without supervision (**Unlab-15**).

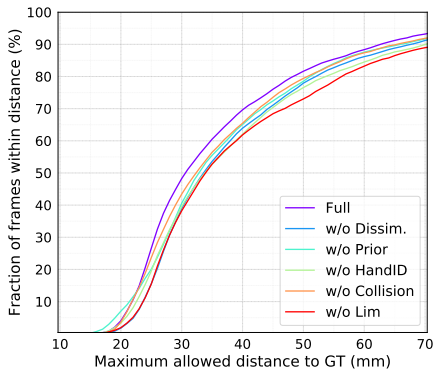
optimization, Adam (Kingma and Ba., 2015) is used with a learning rate of 10^{-5} and a batch size of 16. During training, a forward-backward pass with batch size 16 takes 89ms. A forward pass at inference time takes only 5ms.

4.3.2 *Datasets*

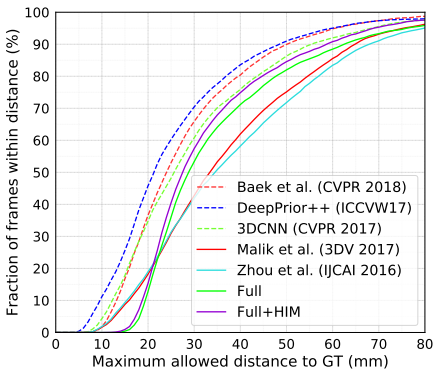
The proposed method is evaluated on two common benchmarks, the NYU dataset (Tompson et al., 2014) and the Hands in the Million Challenge dataset (HIM) (Yuan et al., 2018). The new HANDID dataset is additionally introduced for training to address the lack of hand shape variation in the NYU training data.

NYU Dataset: The NYU dataset is collected using Microsoft Kinect sensors. It contains 72,757 depth images from a single subject in the training set, and 8,252 depth images from two subjects in the test set.

HandID Dataset: Since the NYU training data only contains a single subject, additional training data with more hand shape variations is introduced to help with shape generalization. The HANDID dataset consists of 3,601 frames (640 x 480) from 7 subjects captured with the Intel SR300 sensor. A total of 6 pixels that correspond to the fingertips and wrist are annotated per frame, each with its own occlusion label (See Appendix A.2 for details). During training, a batch contains examples from both HANDID and the NYU dataset with a mixing ratio of 1 : 3.



(a) **Ablation Study:** All components of the method need to work together to resolve ambiguities from the reduced keypoint supervision (all keypoints (**All-21**) evaluated).



(b) **Comparison to SoTA:** The proposed method (**Full**) outperforms competing hybrid methods. By incorporating the HIM dataset(**Full+HIM**), the results are further improved

Figure 4.5: Quantitative evaluation on the NYU dataset.

To emphasize that it is significantly easier to annotate just the fingertips and wrist keypoints, the 5 annotators were asked to label all 21 keypoints for a subset of 10 depth images. It was observed that the additional keypoints take longer to annotate (each joint annotation takes 1.4 times longer) and are less consistent across annotators (with average distance to mean of 10.4 pixels vs 7.3 pixels). In total, the full annotation of 21 joints for 10 images requires 21.2 minutes, while the fingertip and wrist only needs 4.7 minutes.

Hands in the Million Challenge (HIM) Dataset: The proposed method is also evaluated on the HIM dataset (Yuan et al., 2018), where a systematic error in the "ground truth" annotations was discovered. Although the 2D projection of the keypoints into the image plane looks plausible, the 3D keypoint locations do not match the anatomical locations of hand joints (see Figure 4.6).

To quantitatively evaluate this, the minimum-distance-to-point-cloud (MDPC) per joint was used to approximately quantify how well the joint predictions agree with the observed depth image. The NYU annotations and the erroneous HIM annotations have median MDPCs of 9.10mm (avg 10.99mm) and 21.54mm (avg 23.98mm), respectively. By assuming that the physical joint is located roughly at the center of the finger, the HIM annotations would imply an implausible finger thickness of ≈43mm, while the NYU annotations estimates a more reasonable thickness of ≈18mm. This could be caused by a systematic pose-dependent error in corresponding the 3D magnetic sensor positions to the depth camera co-

ordinate. Using the generative model-based loss, the proposed method's predictions are significantly more consistent with the observed depth images. The detailed experiment is presented in Section 4.3.4.

Pre-processing: Similar to established procedures (Baek et al., 2018), the hand is first localized by using the ground truth joint locations and crop the image to a fixed-size cube with 300mm side length. Once localized, the image is re-cropped using the same cube, but centered at the average depth. The cropped image is then normalized to a resolution of 128 x 128 with a scaled depth range between $[-1, 1]$. During training, in-image-plane translation and rotation augmentations, as well as depth augmentations, are applied.

Model Mismatch: Due to a difference in joint definition, only 11 (**Match-11**) of the 14 commonly evaluated NYU keypoints have a joint roughly corresponding to the volumetric model (Figure. 4.4). Therefore, only this subset is evaluated when comparing to other methods. To better demonstrate that the method can infer the positions of unsupervised keypoints, self comparison is done on an expanded set of 21 NYU keypoints (**All-21**) which roughly correspond to anatomical joints of the kinematic skeleton (Figure 4.4, right). The results are further broken down for the 6 supervised (**Lab-6**) and the 15 unsupervised keypoints (**Unlab-15**).

4.3.3 *Ablation Studies*

An ablation study was performed on the NYU dataset. Two metrics are used to evaluate the results.

Keypoint Accuracy: All components are necessary as removing them from the full method (**Full**) reduces accuracy. See Table 4.1 for the average per-joint error in mm, and Figure 4.5a for the percentage of correct frames curve.

Method	Unlab-15	Lab-6	All-21
Full	**16.13**	**20.72**	**17.45**
w.o. E_{dissim}	19.06	21.47	19.75
w.o. E_{bone}	18.53	22.03	19.53
w.o. $E_{\text{collision}}$	16.80	22.20	18.34
w.o. E_{lim}	18.72	22.24	19.73
w.o. HANDID	17.01	23.20	18.78

Table 4.1: **Ablation study** on the NYU dataset (see Figure 4.4). All errors are reported in mm.

Bone Lengths: For bone length evaluation, the ground-truth and the predicted bone lengths are not directly comparable due to the mismatch in model definitions (cf. Figure 4.4, left). Instead, the 20 bone lengths of the hand are treated as a 20-dimensional vector and k-means clustering with $k = 2$ is used to identify the two subjects in the test set of the NYU

dataset. In Table 4.2, the F1 scores (defined as $\frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$) of the two clusters is shown. k-means is meaningful for this task as clustering bone lengths of the annotations (**Ground Truth**) results in perfect F1 scores for both subjects. Note that poses with high self-occlusion have very little information to help disambiguate hand shapes. Thus, one cannot expect methods that perform per-frame estimation to attain a perfect F1 score from the given input.

Discussion: Given the reduced supervision, it is ambiguous whether the loss is minimized by deforming the bone lengths or updating the hand pose. Consequently, the method without bone length prior can arbitrarily distort the bone lengths as long as the fingertips are correctly estimated (see w.o. E_{bone}, Table 4.1). This results in a significant drop in accuracy for keypoints without direct supervision (**Unlab-15**). Correspondingly, k-means clustering fails to find consistent clusters for the two subjects.

Method	S1	S2
Ground Truth	1.00	1.00
Full+HIM	**0.70**	**0.80**
Full	0.63	0.70
w.o. E_{dissim}	0.57	0.59
w.o. E_{bone}	0.52	0.42
w.o. $E_{\text{collision}}$	0.62	0.68
w.o. E_{lim}	0.6	0.42
w.o. HANDID	0.55	0.54

Table 4.2: F1 score of k-means clustering of bone lengths vectors for the two subjects in the test set.

However, the bone length prior alone is not enough to resolve the ambiguity in hand shape. A similar drop in accuracy on unsupervised keypoints (**Unlab-15**) occurs when the dissimilarity loss is removed (see w.o. E_{dissim}, Table 4.1). This is because statistically plausible bone lengths can still vary wildly to accommodate the fingertip annotations, without being constrained to explain the image. Pose priors in the form of joint limits (w.o. E_{lim}) and collision prior (w.o. $E_{\text{collision}}$) additionally constrain the articulations, which improve the keypoint accuracy (Table 4.1).

Due to the NYU training data containing only one hand shape, the method cannot learn to discriminate between hand shapes of different users, leading to F1 scores that are close to random (see w.o. HANDID, Table 4.2). Hence, for the unseen hand shape in the test set this leads to greatly reduced accuracy on supervised keypoints (**Lab-6**). This can be accounted for if hand shape variations are present in the training data. The result of this can be seen when the HandID dataset is used (Full) and the accuracy is further improved when HIM is (Full+HIM).

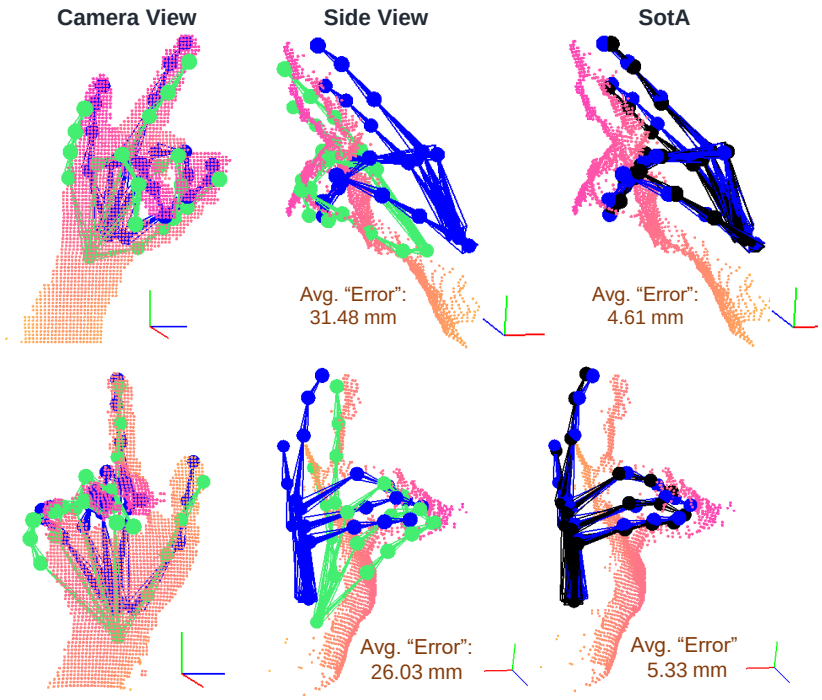


Figure 4.6: **Annotation Errors in HIM:** Although both the "ground truth" (Blue) and the predictions of the proposed method (Green) seem consistent in the camera view. The side views show the "ground truth" is erroneous. State-of-the-art (SotA) method (Wu et al., 2018) (black) over fits to the systematic error.

4.3.4 *Comparison to the State of the Art (SotA)*

Although state-of-the-art methods obtain mean per-joint errors lower than 10mm on the HIM dataset (Ge et al., 2017; Wu et al., 2018), it should be emphasized that this is against the erroneous "ground truth". The proposed method is trained using a "slack" radius of 25 mm to account for this, which results in better fitting pose predictions than even the "ground truth" (see Figure 4.6 and Figure 4.7 for more qualitative evaluation).

Method	Match-11
Full+HIM	**17.73**
Full	18.50
Full+HIM (w.o. E_{dissim})	20.01
Zhou et al., 2016	19.21
Malik et al., 2017	18.35
Baek et al., 2018	14.71
Oberweger et al., 2017	13.10
Ge et al., 2017	15.09

Table 4.3: **Comparison to SotA:** regression-based methods (bottom) do not enforce kinematic consistency while others (top, middle) do.

For a more fair quantitative evaluation, the minimum-distance-to-point-cloud (MDPC) is used instead to approximate how well the predictions fit the input. On the HIM test set comprising of 95,540 images, the proposed method achieves median MD-

PCs of 11.74mm (avg 13.87mm), while Wu et al., 2018 achieves 21.97mm (avg 24.16mm). The predictions of the proposed method better match the NYU annotations with median MDPCs of 9.10 mm (avg 10.99 mm). This suggests that the proposed method produce better fitting predictions while most state-of-the-art methods learn to replicate the errors in the training data. This ability to prevent overfitting is verified in the qualitative evaluations in Figure 4.6 and Figure 4.7 where the propose method can be seen to have more 3D consistent predictions in novel views.

On the NYU dataset (see Table 4.3 and Fig. 4.5b), the proposed method outperforms the other kinematic model-based methods of Malik et al. (2017) and Zhou et al. (2016) while requiring less keypoint annotations. Although methods that directly predict 3D joint positions perform better (Baek et al., 2018; Ge et al., 2017; Oberweger et al., 2017), it should be emphasized that these methods without a model-based generative loss are liable to learning the annotation errors as shown.

In the self-supervised setting, although Dibra et al. (2017) does not provide their predictions on the subset of **Match-11** keypoints, their method performs similarly to Zhou et al. (2016) which the proposed method greatly outperforms. Compared to Wan et al. (2019)'s method with single view training, the proposed method achieved similar performance. While their methods do not require any annotation, the proposed method additionally solves the more ambiguous and harder problem of adapting to the hand shapes of the user during test time. The method of Wan et al. (2019) can only fit to the average hand shape of the training data or to preset bone lengths.

4.3.5 *Adaptation to a New Domain*

Despite the aforementioned annotation errors, the HIM dataset contains a variety of views, poses, and hand shapes that could be useful. To show the effect of the annotation bias and how the dissimilarity loss overcomes it, the proposed method is trained only on the HIM data and then tested on the NYU data. In Figure 4.8, it can be seen that the dissimilarity loss significantly improve generalization performance.

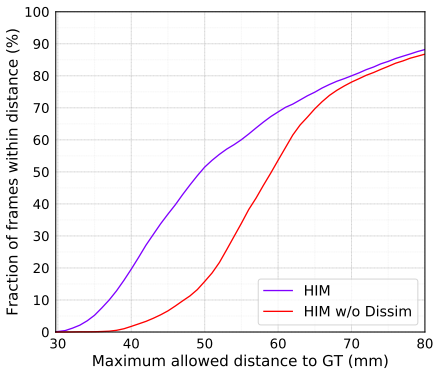


Figure 4.8: **Cross Benchmark Test:** The performance on the NYU dataset after training *only on the HIM dataset*.

Therefore by incorporating this data into training by mixing the

NYU, HIM, and HandID datasets in a single batch with a ratio of 3:3:2, the proposed method can still benefit from all available data. In Figure 4.5b and Table 4.3, it is shown that the dissimilarity loss is still vital to make use of data with erroneous annotations.

4.4 LIMITATIONS & DISCUSSION

Although the proposed method outperforms other kinematic model-based methods, even with less annotations, there is still a gap to recent learning-based methods that regress 3D joint positions. However, these methods

- are not explicitly penalized for producing anatomically implausible shapes due to the lack of an underlying kinematic hand model, and
- are prone to overfit to errors in the training annotations, as well as to errors in the annotation collection method.

Additionally, for poses with heavy self-occlusions, the monocular depth data is not sufficient to resolve ambiguities with the reduced annotation set. Extra supervision, such as from temporal consistency, or from multi-view constraints (as done in Wan et al., 2019), is needed to estimate the pose and shape in these cases.

4.5 CONCLUSION

This chapter has shown that a generative model-based loss can reduce the amount of supervision needed to learn both the pose and shape of hands. This greatly reduces the amount of annotations needed to adapt a method to data obtained in a new domain. Furthermore, it was shown that the generative model-based loss helps to regularize against annotation errors, for example on the HIM dataset, while existing methods overfit to these errors. This demonstrates the importance of ensuring that the model predictions explain not only the annotations but also the image itself.

Visualization of Predictions on the Biased HIM dataset

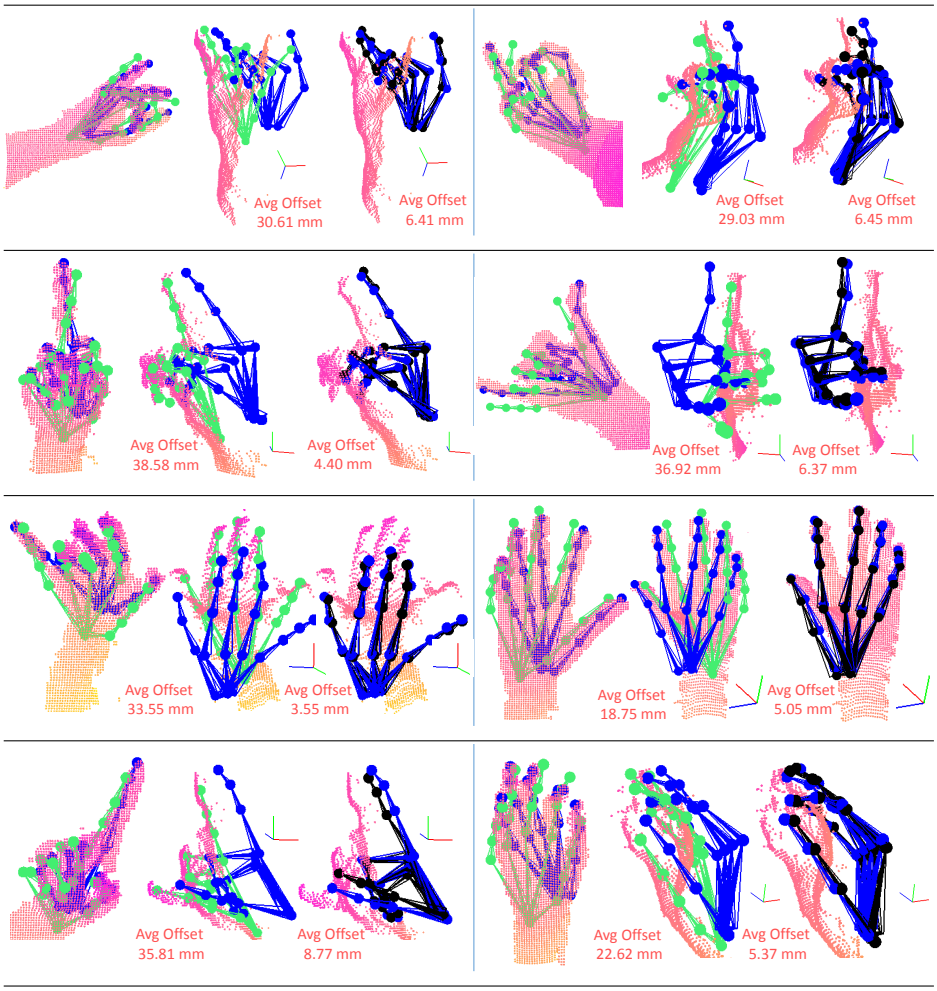


Figure 4.7: Predictions of proposed method are in **green**, the "ground truth" annotations are in **blue**, and State-of-the-Art (SotA) (Wu et al., 2018) predictions are in **black**. Each cell shows the camera view (**left**), a novel view (**middle**), and the same novel view with SotA predictions (**right**). Note that these results are randomly selected and are representative of the whole HIM dataset.

5 LIVE RECONSTRUCTION OF HAND INTERACTIONS FROM MONOCULAR RGB VIDEO

The previous chapter introduced a new approach to train hand pose estimators from monocular depth camera input by using a model-based self-supervised loss. This chapter (published as Wang et al., 2020a) relaxes the single hand assumption by considering hand-hand interactions and moves beyond depth input to the more ubiquitous but more challenging RGB input. As a result, it presents the first real-time method for motion capture of both skeletal pose and 3D surface geometry of hands from a single RGB camera that explicitly considers close interactions. In order to address the inherent ambiguities in RGB data, this chapter proposes a novel multi-task CNN that regresses multiple complementary pieces of information, including segmentation, dense matchings to a 3D hand model, and 2D keypoint positions, together with newly proposed intra-hand relative depth and inter-hand distance maps. These predictions are subsequently used in a generative model fitting framework to aggregate the aforementioned image evidence in order to predict the final pose and shape parameters of 3D hand models for both hands.

The individual components of the proposed RGB two-hand tracking pipeline are experimentally verified through an extensive ablation study. Moreover, the approach is demonstrated to achieve previously unseen two-hand tracking performance from RGB, and quantitatively and qualitatively outperforms existing RGB-based methods that were not explicitly designed for two-hand interactions. Furthermore, the proposed method even performs on-par with depth-based real-time methods that have less ambiguous input data.

5.1 INTRODUCTION

Marker-less 3D hand motion capture is a challenging and important problem. With the abundance of smart and mobile devices, interaction paradigms with computers are changing rapidly and moving farther away from the traditional desktop setting. With the recent progress on virtual and augmented reality (VR/AR), hand pose estimation has gained further attention as direct, natural, and immersive way to interact. The numerous opportunities for application also include robotics, activity recognition, or sign language recognition and translation. Hence, hand

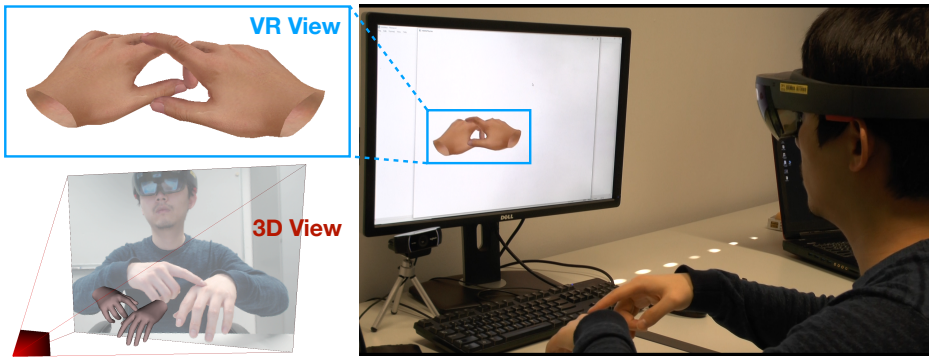


Figure 5.1: The proposed RGB2Hands approach tracks and reconstructs the 3D pose and shape of two interacting hands in real time using a single RGB camera (right). This recovers the global 3D pose and shape (bottom left), which can be used to visualize interacting hands in VR (upper left), among other applications.

pose estimation has been an actively researched topic for years. Depending on the application, several properties are desirable for the method, *e.g.*, marker-less capture, real time performance, capabilities for tracking two interacting hands, automatically adapting to the users' hand shape, or the use of a single RGB camera. However, due to a range of challenges, such as frequent occlusion, depth-scale ambiguity, and self-similarity of hand parts, achieving all of these properties is a difficult task.

To ease the problem, many previous works on 3D hand pose estimation use special depth cameras to provide partial 3D information. Nevertheless, many of them focused on tracking a single isolated hand (Yuan et al., 2018), with only a few exceptions that are able to handle object interactions (Sridhar et al., 2016; Tzionas et al., 2016) or interactions with a second hand (Mueller et al., 2019; Taylor et al., 2016, 2017). In recent years, the research focus has shifted towards methods that use a single RGB camera since these sensors are ubiquitous (Mueller et al., 2018; Zimmermann and Brox, 2017). Despite tremendous progress, to date there is no method explicitly designed for and capable of reconstructing close two-hand interactions from single RGB input. However, humans naturally use both of their hands for interaction with real and virtual surroundings, and for gesturing and communication. Therefore, many applications require hand pose estimation of both hands in close interaction simultaneously.

To this end, this chapter presents the first method for marker-less capture of 3D hand motion and shape from monocular RGB input that successfully handles two closely interacting hands (see Figure 5.1). This real-time approach automatically adapts to the user's hand shape, and reliably captures collision-resolved poses also under difficult occlusions.

Since color images carry no explicit 3D information, the method also have to cope with scale and depth ambiguities. A proper handling of these ambiguities, which are inherent to monocular RGB data, is particularly important in the two-hand case, since mismatches in per-hand depth estimates would lead to incorrectly captured interactions in 3D. Hence, the target setting with a monocular RGB camera is significantly more challenging compared to previous works that make use of depth data, such as Mueller et al. (2019) and Tzionas et al. (2016). To achieve this goal, and thus overcome the challenges and ambiguities of monocular RGB data, a novel multi-task CNN is proposed to regress multiple variables simultaneously. This includes per-pixel left/right hand segmentation masks, dense vertex matchings to a parametric hand model, intra-hand relative depth maps, inter-hand distance, as well as occlusion-robust 2D keypoint positions. These regression targets are designed to explicitly consider the challenges of monocular two-hand reconstruction like strong occlusions and ambiguous relative 3D placement of the hands. The resulting predictions are used in a generative model fitting framework to robustly estimate for both hands the pose and shape parameters of a 3D hand model (see Figure 5.2).

For training the multi-task network, both real and synthetic data are combined from different sources to bridge the domain gap. Since none of the publicly available datasets are sufficient for this purpose, additional dataset comprising both real and synthetic images are created. To obtain real data with (possibly noisy) annotations, the depth-based CNN from Mueller et al. (2019) and an RGB-D sensor is used. To obtain perfectly annotated synthetic data, a simulation system is used that captures physically correct two-hand interactions with diverse hand shapes and appearances. It is shown experimentally that the proposed mixed-data training set, in conjunction with the multi-task CNN, is crucial for successful optimization of the hand model parameters on monocular RGB images. The extensive evaluation, in both 2D and 3D, is enabled by a new benchmark dataset RGB2HANDS that contains significantly stronger hand interactions compared to previous benchmarks.

In summary, this chapter propose the first monocular-RGB-based method for 3D motion capture of two strongly interacting hands, which simultaneously estimates hand pose and shape, while running in real time. The technical contributions in order to achieve this include:

- A *generative model fitting formulation* that is specifically tailored towards fitting parametric 3D hand models of two interacting hands to an RGB image, while taking inherent depth ambiguities and occlusions into account. To this end, information is extracted from

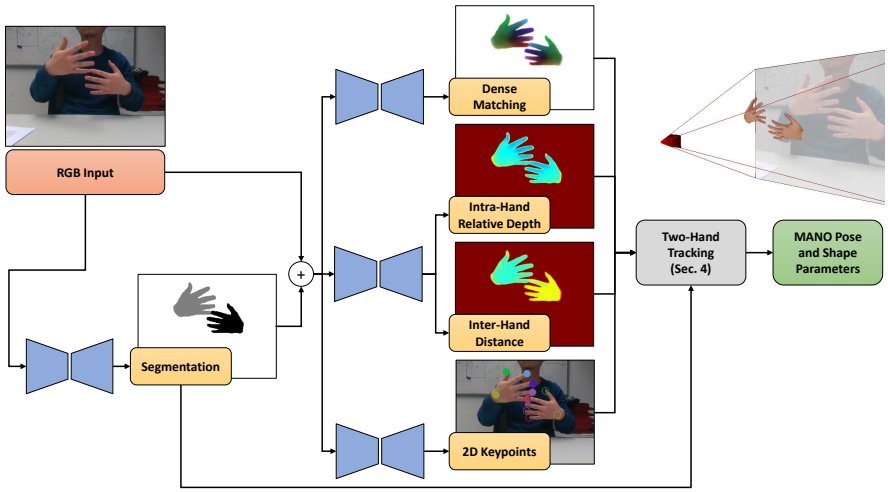


Figure 5.2: Illustration of the proposed RGB2Hands approach. The RGB image is processed by neural predictors that estimate segmentation, dense matching, intra-hand relative depth, inter-hand distances, as well as 2D keypoints. This is then used within a two-hand tracking energy minimization framework. The output are pose and shape of the 3D MANO model (Romero et al., 2017) of both hands, which directly give rise to a bimanual 3D reconstruction.

the input image based on a machine learning pipeline, which is then used as fitting target.

- An *alternative image-based representation of 3D geometry information* is proposed, namely intra-hand relative depth, and inter-hand distance, which can be extracted directly from RGB images using the *multi-task CNN* and is scalable to dense hand surfaces. In combination with 2D keypoints, and an image-to-hand-model matching prediction, this allows the parametric model to be effectively fitted.
- To train these machine learning predictors, synthetic data is used to complement a real dataset that has possibly noisy annotations. For the former, a *physically-correct synthetic data generation framework* is introduced, which is able to account for interacting hands with varying hand identities, both in terms of shape and appearance.
- For performance evaluation, a new benchmark dataset RGB2Hands is introduced. It consists of real two-hand image sequences that comes with manual keypoint annotations of position and occlusion state. Synchronously recorded depth data enables 3D evaluation.

5.2 OVERVIEW

An overview of the approach is presented in Figure 5.2. Given a monocular RGB image that depicts a two-hand interaction scenario, the goal is to recover the global 3D pose and surface geometry by fitting a parametric hand model to both hands in the input image, as described in Section 5.3. Such a model-fitting task requires information extracted from the input image to be used as a fitting target, which however represents a major challenge when using only RGB data. Previous methods that rely on depth data (Mueller et al., 2019; Taylor et al., 2017) are implicitly provided with a much richer input (*i.e.*, global depth), which is the fundamental ingredient for an accurate 3D pose and shape fit. Global depth estimation from a single RGB image, on the other hand, is ill posed.

Note that, in particular in the two-hand case, inconsistent depth estimates per hand would lead to incorrectly captured interactions in 3D. Thus, the method and the scene representation need to be able to handle these ambiguities well. Therefore, in Section 5.4, an alternative representation of dense 3D geometry information is proposed, tailored for a two-hand scenario, which is amenable to be directly extracted from RGB images based on a machine learning pipeline. This is in contrast to existing representations which are limited to sparse (*i.e.*, per-hand and/or per-joint) information and cannot be extended to dense geometry in a scalable way, such as joint heatmaps (Mueller et al., 2018; Zimmermann and Brox, 2017) or part orientation fields (Xiang et al., 2019). To this end, inter-hand distance and intra-hand depth maps are regressed instead, in combination with robust 2D keypoints. This design choice explicitly provides sufficient information to resolve depth ambiguities in the model-fitting step. Furthermore, dense per-pixel surface matchings to the parametric hand model is also regressed directly from input images. This step is designed to be robust against the significant skin tone and illumination variability in RGB images.

Finally, this chapter describes the training data that is used to train the machine learning components in Section 5.5, where a novel methodology is introduced to generate photorealistic and physically accurate synthetic data of sequences with interacting hand motions. To this end, a motion capture-driven physics-based simulation is employed to generate physically-correct sequences of hands with varying identities (skin tone and shape).

5.3 TWO-HAND TRACKING FRAMEWORK

The proposed hand representation builds on the parametric surface hand model MANO proposed by Romero et al. (2017), which is summarized below. Subsequently, the proposed model-based fitting framework will be derived.

5.3.1 *Parametric Pose and Shape Model*

MANO was built from more than 1,000 scans of 30 subjects performing a large variety of poses, and consequently the model is capable of reproducing hand shape variability and surface deformations of articulated hands with high detail. Specifically, for a single hand, MANO outputs a set of 3D vertex positions $\mathcal{X}$ of an articulated 3D hand mesh, i. e.

$$\mathcal{X}(\boldsymbol{\beta}, \boldsymbol{\theta}) = W(T(\boldsymbol{\beta}, \boldsymbol{\theta}), J(\boldsymbol{\beta}), \mathbf{W}) \,, \tag{5.1}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{10}$ and $\boldsymbol{\theta} \in \mathbb{R}^{51}$ are the shape and pose parameters with the latter consisting of 45 articulation parameters and 6 global rotation and translation parameters. $T(\cdot)$ is a parametric hand template in rest pose with pose-dependent corrections to reduce skinning artifacts, $J(\cdot)$ computes the 3D position of the hand joints, and $\mathbf{W}$ is a matrix of rigging weights used by the skinning function W (based on linear blend skinning). See Romero et al., 2017 and Section 3.4 for further details.

As this method targets two-hand scenario, two sets of shape and pose parameters $(\boldsymbol{\beta}_h, \boldsymbol{\theta}_h), h \in \{\text{left}, \text{right}\}$, are used for the left and right hand respectively. To simplify the notation, the parameters of both hands are stacked as $\boldsymbol{\beta} = (\boldsymbol{\beta}_{\text{left}}, \boldsymbol{\beta}_{\text{right}}) \in \mathbb{R}^{20}$ and $\boldsymbol{\theta} = (\boldsymbol{\theta}_{\text{left}}, \boldsymbol{\theta}_{\text{right}}) \in \mathbb{R}^{102}$, and define the unique set of vertices $\mathcal{X} = (\mathcal{X}_{\text{left}}, \mathcal{X}_{\text{right}})$, where the dependence of $\mathcal{X}$ on $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ may be omitted for brevity.

5.3.2 *Overview of Model-Based Fitting Formulation*

In order to track two interacting hands in an image sequence, the parametric MANO model is used within an energy minimization framework. To this end the fitting energy $f(\boldsymbol{\beta}, \boldsymbol{\theta})$ is introduced as

$$f(\boldsymbol{\beta}, \boldsymbol{\theta}) = \Phi(\boldsymbol{\beta}, \boldsymbol{\theta}) + \Omega(\boldsymbol{\beta}, \boldsymbol{\theta}) \,, \tag{5.2}$$

where $\Phi(\cdot)$ is the image fitting term that accounts for fitting the model to the observed RGB image, and $\Omega(\cdot)$ is the regularizer that has the purpose of obtaining a plausible and well-behaved tracking result. By minimizing the fitting energy f, the pose and shape parameters $\boldsymbol{\theta} \in \mathbb{R}^{102}, \boldsymbol{\beta} \in \mathbb{R}^{20}$ (of both hands) are jointly estimated for each frame of the image sequence.

5.3.3 *Image-fitting Term*

Due to the 2D nature of RGB images and the so-resulting depth ambiguities, as well as the additional level of difficulty caused by interactions between the left and right hand, the proposed novel image-fitting term Φ is designed carefully in order to allow for a reliable fit of the parametric hand model. In particular it uses specific information that the multi-task CNN (see Section 5.4) extracts from 2D images that enables the estimation of correct and coherent 3D pose of both hands in interaction, and minimizes the risk of implausible interaction capture due to ambiguous 3D pose estimates of each individual hand. The proposed method combines five components, where the follow terms are used

1. the dense 2D fitting term Φ_{dense},
2. the silhouette term Φ_{sil},
3. the 2D keypoint term Φ_{key},
4. the intra-hand relative depth term Φ_{intra}, and
5. the inter-hand distance term Φ_{inter}.

It should be emphasized that existing methods that are capable of tracking *two hands in interaction* avoid 3D pose ambiguities by heavily relying on depth-based input data that is used in their image-fitting term, which, however complicates the hardware setup. In contrast, the proposed energy terms Φ_{dense}, Φ_{intra}, Φ_{inter} are designed to compensate for the lack of available depth information and enable 3D consistent two-hand reconstructions by using a strong neural prior that extracts suitable information from RGB images only.

With that, the complete image fitting term that accounts for the model-to-image fitting reads

$$\Phi(\boldsymbol{\beta}, \boldsymbol{\theta}) = \Phi_{\text{dense}} + \Phi_{\text{sil}} + \Phi_{\text{key}} + \Phi_{\text{intra}} + \Phi_{\text{inter}}\,, \tag{5.3}$$

where the explicit dependence on $(\boldsymbol{\beta}, \boldsymbol{\theta})$ of the terms have been omitted for the sake of readability. Term weights are provided in Appendix A.5.

The camera intrinsics are assumed to be known and $\Pi : \mathbb{R}^3 \to \Gamma$ defines the projection from camera space onto the image plane. When this is not available, plausible intrinsics can be provided to obtain results accurate up to a scale.

One crucial part for defining the image fitting term is the *dense matching map* $\psi : \mathcal{X} \to \Gamma$, which predicts for each vertex $\boldsymbol{x} \in \mathcal{X}$ the corresponding pixel position $(u, v) \in \Gamma$ in the input image. For the time being ψ is assumed to be known, and later in Section 5.4 how this is obtained will be explained. In the following, when the vertices vertices in the set $\mathcal{X}$ is summed over, only those vertices that are visible are considered, where a vertex $\boldsymbol{x}$ is considered to be visible whenever $\psi(\boldsymbol{x}) \neq \emptyset$.

The individual components will now be explained in depth.

Dense 2D Fitting: Since an RGB image does not contain explicit 3D information, the actual depth of a model vertex is unknown. Hence, the 2D image-plane distance between a projected visible vertex $\Pi(\boldsymbol{x})$ and its corresponding pixel $\psi(\boldsymbol{x})$ is penalized. the dense 2D fitting term is defined as

$$\Phi_{\text{dense}}(\boldsymbol{\beta}, \boldsymbol{\theta}) = \lambda_{\text{d}} \sum_{\boldsymbol{x} \in \mathcal{X}} \|\Pi(\boldsymbol{x}) - \psi(\boldsymbol{x})\|_2^2 , \tag{5.4}$$

where λ_{d} is the relative weight of this term.

Silhouettes: Since the dense matching map might not be perfectly precise for neighboring vertices and pixels, an occlusion-aware silhouette term is introduced to improve the projection error of the estimated hand models in the input image. Similar to previous work (Habermann et al., 2019), a set of *boundary vertices* $\mathcal{X}_b$ is defined and their distance from the silhouette edges in the input image is penalized. The set of boundary vertices is determined based on the current pose and shape estimate in every iteration of the optimization. All hand model vertices that lie close to model-to-background edges in the projected view are chosen.

To efficiently represent the distance to the silhouette edges without explicit correspondences, a Euclidean distance transform representation is used. Since the method need to distinguish the right and left hand, two distance transform images DT_{right} and DT_{left} are created, one for each hand respectively. To this end, the predicted segmentation mask $\mathcal{S}$ (see Section 5.4.1) is used to extract silhouette edges per hand.

Since the method specifically target close two-hand interactions, the segmentation mask does not only contain silhouette edges but also occlusion boundaries (*i.e.*, hand-hand boundaries). Without proper handling, vertices that are occluded by the other hand would be drawn towards the occlusion boundary, which in turn would encourage shrinking of the occluded hand. Thus, the distance transform image for each hand is set to 0 at all pixels that are predicted to belong to the other hand (see Figure 5.3). With that, boundary vertices that project onto the other hand in the input image are not pulled towards the occlusion boundary, which would produce an undesirable distortion effect, leading to a grasping pose, everytime a hand is occluded. Mathematically, this occlusion-aware silhouette term is formulated as

$$\Phi_{\text{sil}}(\boldsymbol{\beta}, \boldsymbol{\theta}) = \lambda_{\text{sil}} \sum_{\boldsymbol{x}_b \in \mathcal{X}_b} \left(DT_{h(\boldsymbol{x}_b)}(\Pi(\boldsymbol{x}_b)) \right)^2 , \tag{5.5}$$

where $h(\boldsymbol{x}_b)$ gives the handedness of boundary vertex $\boldsymbol{x}_b$. Note that an additional normal-based weight is used for each summand as introduced by Habermann et al. (2019). Please refer to this paper for further details.

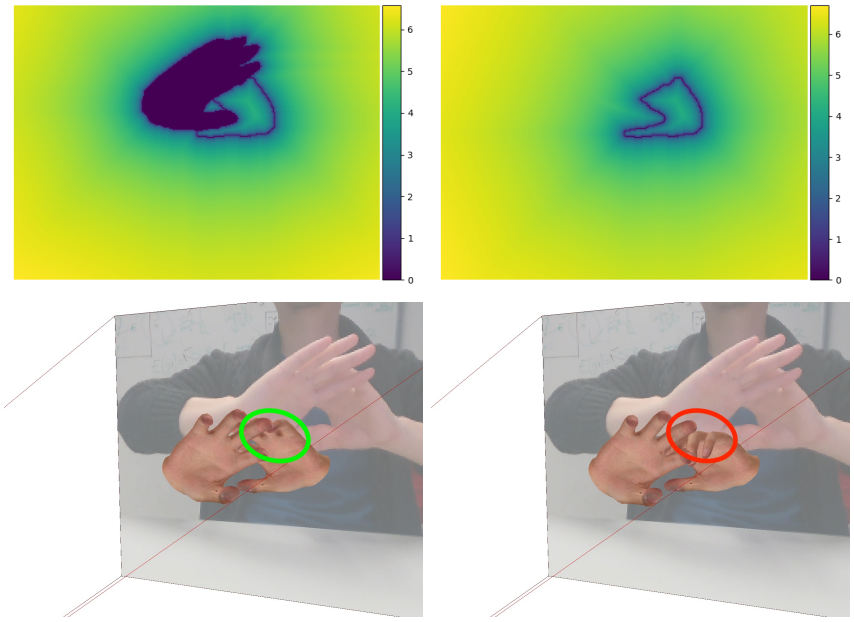


Figure 5.3: Visualization of $log(DT_{\text{left}} + 1)$ with (top left) and without (top right) occlusion handling. The reconstructed hand without occlusion handling (bottom right) incorrectly articulates to explain an occlusion boundary, while the proposed method (bottom left) correctly handles the occlusion.

2D keypoints: Since the dense 2D fitting only constrains visible parts of the hand model, an occlusion-robust 2D keypoint term is added. This term penalizes the discrepancy between corresponding keypoint predictions on the RGB image and the hand model projected to the image plane. The keypoint detection is designed to also be available under occlusion, increasing the robustness to strong occlusions that frequently occur in the two-hand scenario. For each hand the center of the wrist and the 5 fingertip positions are used as keypoints, leading to a total number of 12 keypoints across both hands. $\boldsymbol{x}_j \in \mathbb{R}^3$ are used to denote the 3D position of the j-th keypoint of the hand model. Similarly, $\mathcal{Q}_{\text{key}}(j) \in \Gamma$ denote the pixel position of the j-th keypoint in the image, which is obtained based on the keypoint predictor $\mathcal{Q}_{\text{key}}$ that will be defined in Sec. 5.4. Let $\mathcal{J}$ be the set of *detected* keypoints, which may have less than 12 elements whenever some keypoints do not meet the confidence threshold (see Section 5.4.1). With that, the 2D keypoint term reads

$$\Phi_{\text{key}}(\boldsymbol{\beta}, \boldsymbol{\theta}) = \lambda_{\text{key}} \sum_{j \in \mathcal{J}} \|\Pi(\boldsymbol{x}_j) - \mathcal{Q}_{\text{key}}(j)\|_2^2 . \tag{5.6}$$

Intra-hand relative depth: In order to address depth ambiguities within estimated 3D pose and shape of each individual hand (cf. the Bas-Relief

Ambiguity (Belhumeur et al., 1999)), the *intra-hand relative depth term* is introduced to penalize the differences between per-hand root-relative depth values of the 3D hand model and per-hand relative depth predictions obtained from the RGB image. To this end, the estimated distance along the camera direction (which will be referred to as *z-direction*) from the hand root joint in the model is compared to an analogous output of a machine learning predictor (Section 5.4) that serves as relative depth prior conditioned on the RGB image. Let the function $\text{root}(\boldsymbol{x})$ compute the 3D position of the root joint of the hand to which the vertex $\boldsymbol{x}$ belongs to, and let $(\cdot)_z$ denote the extraction of the z-component of a 3D vector. Moreover, $\mathcal{Q}_{\text{intra}}(u, v)$ denotes the relative depth that was predicted by a neural network in the image at the pixel (u, v). With that, the intra-hand relative depth term Φ_{intra} is defined as

$$\Phi_{\text{intra}}(\boldsymbol{\beta}, \boldsymbol{\theta}) = \lambda_{\text{intra}} \sum_{x \in \mathcal{X}} \left(\mathcal{Q}_{\text{intra}}(\psi(\boldsymbol{x})) - (\boldsymbol{x}_z - \text{root}(\boldsymbol{x})_z)\right)^2 . \tag{5.7}$$

Inter-hand distance: In addition to the intra-hand relative depth, the *inter-hand distance* is also taken into account, where the estimated distance between the root of both hands is compared to the output of a trained learning system predicting the same conditioned on the RGB image. Note that this term is crucial to obtain correct relative placement of the two hands in 3D from monocular RGB data. Let $\text{root}_h, h \in \{\text{left}, \text{right}\}$ be the 3D position of the root joint of a hand and let q_{inter} denote the relative distance of the left hand from the right hand as predicted by a neural network. With that, the inter-hand distance term is defined as

$$\Phi_{\text{inter}}(\boldsymbol{\beta}, \boldsymbol{\theta}) = \lambda_{\text{inter}} \left((\text{root}_{\text{left}})_z - (\text{root}_{\text{right}})_z - q_{\text{inter}}\right)^2 . \tag{5.8}$$

5.3.4 *Hand Model and Tracking Regularization*

In order to enable a plausible and realistic tracking, a regularizer $\Omega(\boldsymbol{\beta}, \boldsymbol{\theta})$ that combines different terms is defined to account for an appropriate regularization of the parametric hand model:

$$\Omega(\boldsymbol{\beta}, \boldsymbol{\theta}) = \Omega_0(\boldsymbol{\beta}, \boldsymbol{\theta}) + \Omega_{\text{overlap}}(\boldsymbol{\beta}, \boldsymbol{\theta}) + \Omega_{\text{scale}}(\boldsymbol{\beta}) . \tag{5.9}$$

Below, *structural regularizers* Ω_0 and Ω_{overlap}, which are well-established terms in hand tracking and reconstruction settings, are summarized. For a detailed description, please refer to previous works, such as Mueller et al. (2019), Romero et al. (2017), and Tagliasacchi et al. (2015). Subsequently, the new (optional) *hand scale prior* Ω_{scale} is introduced, which is designed to address the scale ambiguity that arises specifically when performing

3D reconstruction in monocular RGB data. If this prior is provided, the method is able to obtain *metric* 3D pose and shape reconstruction results.

Structural Regularization: Tikhonov regularization is imposed upon the shape parameter $\boldsymbol{\beta}$, which accounts for it following a multivariate standard normal distribution. Similarly, a thresholded version thereof is used for the pose paraemter $\boldsymbol{\theta}$, so that poses close to the mean pose are not penalized. Furthermore, a temporal regularization is imposed that penalizes the difference between the parameters at the current and the previous frame. Moreover, in order to ensure that the shapes of the left and right hand are similar, the discrepancies between the hand shapes are penalized. These structural regularizers are written in terms of the squared ℓ_2-norm summarily as

$$\Omega_0(\boldsymbol{\beta}, \boldsymbol{\theta}) = \left\| \begin{bmatrix} \lambda_\beta \boldsymbol{\beta} \\ \lambda_\theta \, \mathbf{1}_{>t_\theta}(\boldsymbol{\theta}) \\ \lambda_\tau(\boldsymbol{\beta}' - \boldsymbol{\beta}) \\ \lambda_\tau(\boldsymbol{\theta}' - \boldsymbol{\theta}) \\ \lambda_{\text{sym}}(\boldsymbol{\beta}_{\text{left}} - \boldsymbol{\beta}_{\text{right}}) \end{bmatrix} \right\|_2^2, \tag{5.10}$$

where $\mathbf{1}_{>t_\theta}(\boldsymbol{\theta})$ is a function yielding $\boldsymbol{\theta}$ if $||\boldsymbol{\theta}||_2 > t_\theta$, and $\mathbf{0}$ otherwise. The variables $\boldsymbol{\beta}'$ and $\boldsymbol{\theta}'$ denote the shape and pose parameters from the previous frame, and $\lambda_\bullet$ are the respective weights.

For avoiding collisions between the two hands, as well as within each hand, mesh overlaps as approximated with 3D Gaussians that are attached to the parametric hand model are penalized. The position and size of the Gaussians change according to the shape and pose parameters $(\boldsymbol{\beta}, \boldsymbol{\theta})$ (Mueller et al., 2019). For $\mathcal{N}_i(\boldsymbol{z}|\boldsymbol{\beta}, \boldsymbol{\theta})$ denoting the i-th 3D Gaussian evaluated at the position $\boldsymbol{z} \in \mathbb{R}^3$, the overlap between all pairs (i, j) of Gaussians is computed as

$$\Omega_{\text{overlap}}(\boldsymbol{\beta}, \boldsymbol{\theta}) = \lambda_{\mathcal{N}} \sum_{i,j} \Big(\int_{\mathbb{R}^3} \mathcal{N}_i(\boldsymbol{z}|\boldsymbol{\beta}, \boldsymbol{\theta}) \cdot \mathcal{N}_j(\boldsymbol{z}|\boldsymbol{\beta}, \boldsymbol{\theta}) \, d\boldsymbol{z} \Big)^2 . \tag{5.11}$$

Hand Scale Prior: Since reconstruction from monocular RGB data is inherently ambiguous up to a single scalar factor, the option is given to provide a single metric measurement of the user's hand in order to produce metric results. This measurement is chosen to be the length of the palm, defined as the distance between the middle finger metacarpophalangeal joint (MCP) and the wrist. If the user does not provide this measurement, the palm length is assumed to be given by the mean shape

of the MANO model, i.e., for $\boldsymbol{\beta} = \mathbf{0}$. The hand scale prior is formulated to penalize deviations from the pre-defined palm length α as

$$\Omega_{\text{scale}}(\boldsymbol{\beta}) = \lambda_s \sum_{h \in \{\text{left,right}\}} (\text{palmlength}\,(\boldsymbol{\beta}_h) - \alpha)^2 \quad , \tag{5.12}$$

where the function palmlength$(\cdot)$ computes the length of the palm of the hand model given a set of shape parameters.

5.3.5 *Numerical Optimization*

For the numerical optimization of the fitting energy f in Equation 5.2 a Levenberg-Marquardt (LM) approach is used. The main idea here is to iteratively update the parameters $\boldsymbol{\nu} := (\boldsymbol{\beta}, \boldsymbol{\theta})$ using the Jacobian matrix $\boldsymbol{J}_f$ of f as

$$\boldsymbol{\nu} = \boldsymbol{\nu}^{\text{old}} - (\boldsymbol{J}_f^T \boldsymbol{J}_f + \mu \mathbf{I})^{-1} \boldsymbol{J}_f^T \boldsymbol{f}(\boldsymbol{\nu}^{\text{old}})\,, \tag{5.13}$$

where $\boldsymbol{f}$ is the vector-valued function that stacks all the individual (quadratic) residuals of f, and μ is the LM damping factor. Based on empirical evidence, the LM method is generally known for rapidly decreasing the objective function with very few iterations. Hence, and in order to maintain real-time performance, in addition to efficiently evaluating the Jacobian on the GPU, the iterative optimization is terminated after 10 iterations.

5.4 DENSE MATCHING AND DEPTH REGRESSION

In order to obtain the predictions that were described in the previous section, including predictions for segmentation, dense matching, intra-hand depth, inter-hand distance and 2D keypoints, the RGB input image were feed to a fully-convolutional neural network. This enables the method to work on entire images without requiring a potentially error-prone bounding box estimation for each hand. Since the network is trained using a large training corpus, it successfully learns priors to handle the inherent ambiguities in monocular RGB data. In the following, the network is described, including the outputs, losses, and architecture, in more detail.

5.4.1 *Network Outputs*

The proposed network architecture comprises two stages. In the first stage the network performs per-pixel segmentation into left hand, right

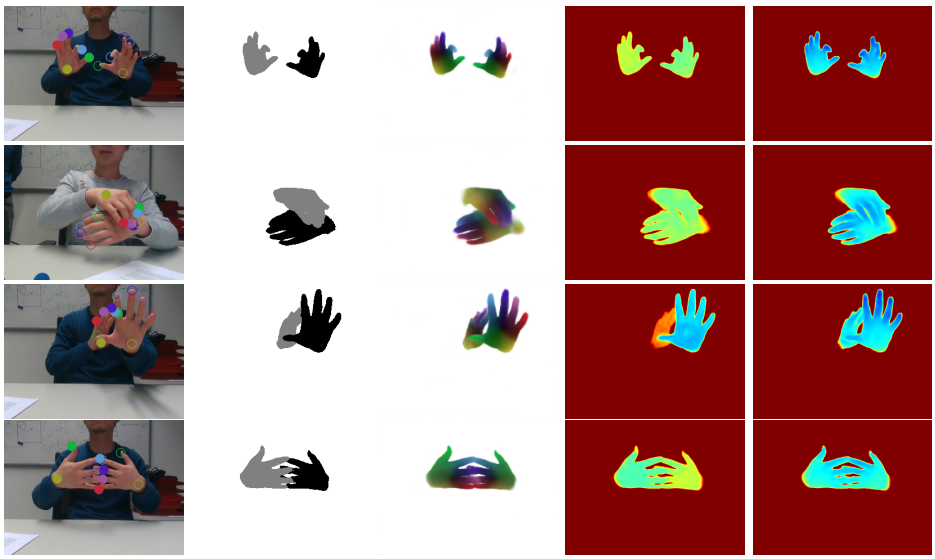


Figure 5.4: Visualization of network outputs. Left to right: 2D keypoints, segmentation, dense matching map, inter-hand distance, intra-hand relative depth.

hand, and background pixels. Then, the architecture branch into multiple subnetworks to regress dense matching, 2D keypoints, intra-hand relative depth, and inter-hand distance (the latter two using a shared multi-task subnetwork). The input for the second stage are both the original RGB input image, as well as the segmentation masks predicted in the first stage. Figure 5.4 shows all outputs predicted from test images.

Segmentation: Let the image have height h and width w. Given only the RGB input image, the first-stage segmentation network predicts probability maps $\mathcal{S}' \in [0,1]^{h \times w \times 3}$ for three classes LEFT, RIGHT, and BG. The probability maps are converted to a segmentation mask $\mathcal{S} \in \{\text{LEFT}, \text{RIGHT}, \text{BG}\}^{h \times w}$ by assigning each pixel the most probable class.

Dense Matching: The dense matching subnetwork regresses a dense matching image $\mathcal{M} \in \mathbb{R}^{h \times w \times k}$, where k is the number of features. Each pixel $\gamma = (u, v) \in \Gamma$ contains the feature vector $\mathcal{M}(\gamma) \in \mathbb{R}^k$ that uniquely determines the surface point of the 3D hand model which is visible at this pixel. The mapping from the feature vector to the 3D model surface is called *dense matching encoding* (see Figure 5.5). Note that the dense matching encoding is the same for the left and right hand, where the

Figure 5.5: Dense matching encoding of MANO model, front and back.

segmentation mask $\mathcal{S}$ is used for disambiguation. The same encoding as Mueller et al., 2019 is used to embed the hand surface to a 3D feature space for the dense matching map. This is done using the method of Bronstein et al., 2006 to approximately preserve geodesic distances in the feature space. The feature space is then mapped to an HSV color space cylinder which results in each finger being assigned a different hue. The extended feature vector at vertex $\boldsymbol{x}$ is denoted as $\eta'(\boldsymbol{x}) \in \mathbb{R}^{k+1}$ and define $\eta'(\boldsymbol{x}) = [\eta(\boldsymbol{x}), s(\boldsymbol{x})]$, where $\eta : \mathcal{X} \to \mathbb{R}^k$ is the original dense matching encoding defined on the mesh. The scalar $s(\boldsymbol{x})$ yields a different value $\sigma(\text{RIGHT}) = 0.5$ or $\sigma(\text{LEFT}) = 0.0$ that encodes which hand $\boldsymbol{x}$ belongs to. The *matching distance* between 3D hand model vertices $\boldsymbol{x}$ and pixels γ in the image can then be measured as

$$\Delta_{\mathcal{M},\mathcal{S}}(\gamma, \boldsymbol{x}) = || \, [\mathcal{M}(\gamma), \sigma(\mathcal{S}(\gamma))] - \eta'(\boldsymbol{x}) \, ||_2 \, . \tag{5.14}$$

The dense matching map $\psi : \mathcal{X} \to \Gamma$ needed to establish correspondence between model vertices and the RGB image can be formulated as

$$\psi'(\boldsymbol{x}) = \arg\min_{\gamma \in \Gamma} \Delta_{\mathcal{M},\mathcal{S}}(\gamma, \boldsymbol{x}) \tag{5.15}$$

$$\psi(\boldsymbol{x}) = \begin{cases} \psi'(\boldsymbol{x}) \, , & \text{if } \Delta_{\mathcal{M},\mathcal{S}}(\psi'(\boldsymbol{x}), \boldsymbol{x}) < t_c \\ \emptyset \, , & \text{otherwise} \end{cases} . \tag{5.16}$$

If the minimum distance of vertex $\boldsymbol{x}$ to all pixels is larger than the threshold t_c, this vertex is likely not visible and be set to $\psi(\boldsymbol{x}) = \emptyset$. The calculation of the dense matching map ψ is efficiently implemented using parallel reduction in CUDA. The dense matching encoding $\eta(\cdot)$ is defined analogously to the approach by Mueller et al. (2019) with $k = 3$.

Intra-Hand Relative Depth: The network further learns to predict an intra-hand relative depth map $\mathcal{D}_{\text{intra}} \in \mathbb{R}^{h \times w}$. For each hand pixel, it contains the estimated depth difference of this hand point to the root of the respective hand. Note that $\mathcal{D}_{\text{intra}}$ is scale-normalized due to the inherent ambiguity in RGB images. This is multiplied with the palm length α to obtain the *metric* relative depth map $\mathcal{Q}_{\text{intra}}$, which is used for 3D model fitting (cf. Equation 5.7).

Inter-Hand Distance: The multi-task CNN also learns to estimate the distance in depth between the two hands. Instead of predicting a single scalar, the distance is regressed as an image $\mathcal{D}_{\text{inter}} \in \mathbb{R}^{h \times w}$. This allows the method to use a fully-convolutional network and thereby enables feature sharing with the intra-hand depth prediction task. Every pixel in $\mathcal{D}_{\text{inter}}$ that belongs to a hand contains the distance of its root joint from

the other hand's root (in the case for only a single hand being visible, a constant value is assigned to all pixels). Note that each pixel in the output can thus be seen as member of an ensemble. Analogous to the intra-hand relative depth, the inter-hand distance is normalized with the size of the hand for training. The ensemble is summarized with one relative distance value d_h per hand by calculating the median over all pixels that are predicted to belong to the respective hand based on the segmentation mask $\mathcal{S}$, i.e.

$$d_h = \underset{\gamma \in \Gamma, \mathcal{S}(\gamma)=h}{\text{median}} \mathcal{D}_{\text{inter}}(\gamma)\,. \tag{5.17}$$

The robust relative distance is set to $d_{\text{inter}} = \text{mean}(d_{\text{left}}, -d_{\text{right}})$. When the two hands are close, d_{left} and d_{right} can be degenerate and have the same sign. In this case, d_{inter} is set to 0. For the model fitting, the *metric* absolute distance is defined as $q_{\text{inter}} := \alpha \cdot d_{\text{inter}}$ (cf. Equation 5.8).

2D Keypoints: Let $\mathcal{J}_{\text{total}}$ be the set of all 12 keypoints, namely the fingertips and wrist of each of the two hands. The 2D keypoint estimation is formulated as heatmap regression task. The network outputs heatmaps $\mathcal{H} \in \mathbb{R}^{h \times w \times |\mathcal{J}_{\text{total}}|}$, a one-channel image for each of the keypoints. Each ground-truth heatmap contains a Gaussian with radius $0.07 \cdot r_c$, where r_c is the edge length of the larger edge of a tight hand crop, scaled to have maximum value 1, centered at the 2D keypoint position. Note that the ground truth is also provided for occluded keypoints which enables the network at test time to predict keypoint locations under strong occlusions which are common for two-hand interactions. The maximum location of each predicted heatmap is extracted using

$$\gamma_j^{\max} = \arg\max_{\gamma \in \Gamma} \mathcal{H}(\gamma, j)\,. \tag{5.18}$$

A threshold t_h is used to filter out low-confidence estimates and obtain the 2D keypoint location as

$$\mathcal{Q}_{\text{key}}(j) = \begin{cases} \gamma_j^{\max}, & \text{if } \mathcal{H}(\gamma_j^{\max}, j) > t_h \\ \varnothing, & \text{otherwise} \end{cases}\,. \tag{5.19}$$

5.4.2 *Network Architecture and Training*

The proposed network consists of several subnetworks as shown in Figure 5.2. Each subnetwork is a U-Net (Ronneberger et al., 2015) with 4 layers for down-sampling and 4 layers for up-sampling, resulting in a bottleneck resolution of $\frac{h}{16} \times \frac{w}{16}$. Skip connections are used between

layers of the same resolution in the down- and up-sampling stream to better preserve local information. Instance normalization is employed instead of batch normalization at every layer as proposed by Ulyanov et al. (2016).

The softmax cross-entropy loss is used for the segmentation prediction and ℓ_2-losses for all other outputs. For real data, a loss mask is used to disable the losses for holes in the annotations, which are present due to the projection between the depth and color channel. Appendix A.3 describes the annotation transfer from the depth to the color image. The whole network is trained end-to-end for 400k iterations using Adam with a learning rate of 0.001 and a beta of 0.9. Data augmentations is performed on-the-fly to further increase the diversity of the training set (see Appendix A.4).

5.5 TRAINING DATA

For training the regressor in a supervised manner, for a given RGB image containing two potentially interacting hands, a ground-truth relative depth map $\mathcal{D}^{\text{GT}}_{\text{intra}}$, the relative inter-hand distance map $\mathcal{D}^{\text{GT}}_{\text{inter}}$, a dense matching image $\mathcal{M}^{\text{GT}}$, and 2D joint position heatmaps $\mathcal{H}^{\text{GT}}$ are ideally required. Existing datasets like the *Rendered Hands Dataset (RHD)* (Zimmermann and Brox, 2017) or *Panoptic* (Joo et al., 2017) only provide a subset of the required annotations (see Table 5.1) and, in particular, do not have dense matching annotations. The former does also not show realistic and physically plausible close two-hand interactions, an important requirement for the target setting. The recent FreiHand dataset (Zimmermann et al., 2019) provides crops of single hands with annotated MANO fits, sometimes even with objects, but no two-hand frames. Generating synthetic interacting hands images from these would require compositing and would lead to unrealistic interaction. Therefore, since manual annotation of the labels required is impossible, a new set of strategies is proposed to obtain annotations for both real and synthetic images. The existing datasets *RHD* and *Panoptic* are added to the proposed real and synthetic datasets to increase data diversity and hence improve generalization. Table 5.1 presents a summary of the different datasets used for training, and gives details about the ground-truth annotations available in each of them. In the following, the procedure for creating the proposed synthetic and real dataset is described. Furthermore, in Section 5.6.2 an ablation study is presented that demonstrates how the proposed real data (with noisy annotations) helps bridge the real-synthetic domain gap, and the perfectly annotated synthetic data mitigates influence of noise.

	Segmentation	Dense Corrs.	Intra-Hand	Inter-Hand	2D Keypoints
Synthetic Data	✓	✓	✓	✓	✓
Real Data	✓	✓	✓	✓	✓
RHD	✓	✗	✓	✓	✓
Panoptic	✗	✗	✗	✗	✓

Table 5.1: Available annotations in existing hand tracking datasets and the proposed datasets.

Real Data: The state-of-the-art depth-based two hand tracker of Mueller et al. (2019) is leveraged to track sequences of two hands in interaction with an RGB-D sensor that captures synchronized color and depth images. These sequences are recorded in front of a green screen to enable background augmentation as post-processing. Mueller's approach outputs MANO per-frame shape $\boldsymbol{\beta}$ and pose $\boldsymbol{\theta}$ parameters, which, in combination with the extrinsic parameters of the RGB and depth sensors of the camera, is used to reproject the surface of the tracked hand to the RGB image. For details on the reprojection see Appendix A.3. Subsequently, the relative depth maps $\mathcal{D}^{\text{GT}}_{\text{intra}}$, inter-hand relative distance maps $\mathcal{D}^{\text{GT}}_{\text{inter}}$, and dense matching images $\mathcal{M}^{\text{GT}}$ can be compute for the real RGB image. Additionally, the 2D keypoint positions from Joo et al., 2017 are used to construct heatmaps $\mathcal{H}^{\text{GT}}$ for supervision. Since tracking a single hand is usually more robust and accurate than tracking two interacting hands, single-hand sequences are also included in the proposed dataset. Depth-based composition is then used to obtain images depicting two hands, see Appendix A.4. Note that bad tracking results and 2D keypoint predictions are leaned manually by visual inspection to ensure reasonable quality in the real data annotations.

Synthetic Data: The above-described approach to annotate real data is not perfect. In some poses the depth-based tracker may exhibit tracking errors. Also, the RGB-D camera has separate depth and RGB optics which are apart by a small baseline. The resulting parallax leads to some occlusion-disocclusion-related holes in the annotations when reprojecting them from the depth channel to the color channel. This makes the real data not sufficiently accurate and unable to produce annotations for highly-challenging poses. To address this issue, synthetically generating images with corresponding annotations are created to complement the real dataset. To this end, and similar in spirit to Zhao et al. (2013) and Mueller et al. (2019), a motion capture-driven physics-based simulation is employed to generate physically-correct hand sequences (*e.g.*, without self-collisions, with accurate inter-hand contact, and with a soft-skin layer) where two hands realistically interact in a large variety of poses. To increase the realism and variety of simulated hand sequences, and in contrast to existing approaches that use a hand template of fixed shape and appearance in the simulation framework, the surface-based

parametric model of MANO is extended to a volumetric representation that is subsequently fed into the simulation (Verschoor et al., 2018). This allows the synthesis of complex hand motions driven by a motion capture sequence, including 2D keypoint positions and heatmaps $\mathcal{H}^{\text{GT}}$, dense correspondence images $\mathcal{C}^{\text{GT}}$, relative depth maps $\mathcal{D}^{\text{GT}}_{\text{intra}}$, and relative inter-hand distance maps $\mathcal{D}^{\text{GT}}_{\text{inter}}$, with varying subject identities. Therefore, data with varying hand shapes can be generated.

Additionally, the MANO model is further extended with photorealistic appearances by a standard texture mapping approach. Hand textures were generated by the HTML appearance model to be presented in chapter 7. The model captures data from users with varying ethnicity, gender and age. In practice, 10 different hand textures are generated. The ability to render physically plausible two-hand interactions for various hand shapes and appearances enables the proposed approach to generalize better to real world scene diversity.

5.6 EXPERIMENTS

In this section the proposed RGB two-hand tracking approach is experimentally evaluated in order to demonstrate its merits. First, the dataset and metrics used in the evaluation is introduced. Subsequently, an ablation study is conducted that evidences the importance of the individual components. Afterwards, the proposed method is compared quantitatively and qualitatively to other related works. Moreover, additional qualitative two-hand tracking results are presented.

5.6.1 *Datasets and Metrics*

Although the dataset by Tzionas et al. (2016) is commonly used to evaluate two-hand tracking methods, it is not well-suited for evaluating two-hand tracking methods with *strong interactions*. This is because in their dataset only very few frames actually exhibit close two-hand interactions. For a more comprehensive evaluation of challenging interaction settings, this chapter introduce a new benchmark dataset, RGB2HANDS, which exhibits stronger interactions and more overlap between the left and right hand. It is illustrated in Figure 5.6 that RGB2HANDS contains more frames with stronger hand-hand interactions compared to the dataset by Tzionas et al. (2016), which is measured in terms of the overlap of the bounding box from the left and right hand.

In the following, details of both dataset are presented as well as the evaluation metrics.

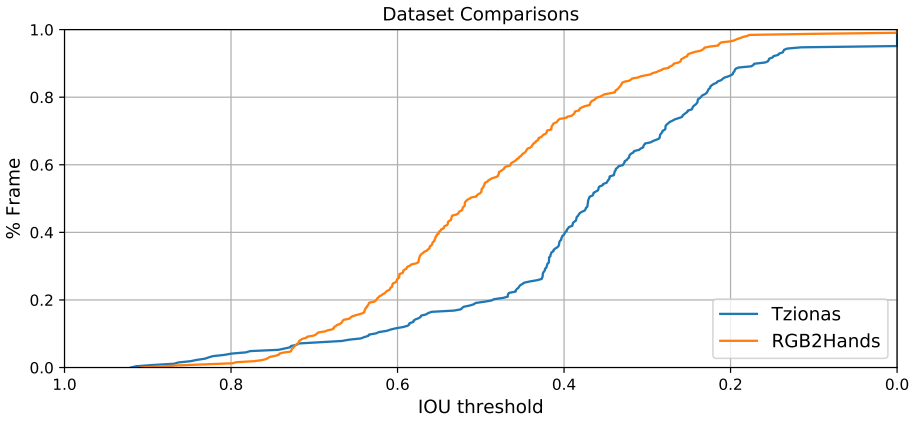


Figure 5.6: The two-hand TZIONAS dataset has significantly fewer frames with strongly interacting and overlapping hands compared to the proposed RGB2HANDS dataset. The plot shows the percentage of frames (y-axis) where the overlap (in terms of the *intersection over union*, IOU) of the left and right hand bounding box is greater than a certain threshold (x-axis).

Tzionas Dataset: The TZIONAS dataset contains 7 two-hand sequences with a total number of 1,307 RGB-D frames. 2D annotations on the depth image are provided every 5th frame for the 14 interior joints of each hand when visible. The camera calibration can be used to obtain 3D annotations by backprojection.

RGB2Hands Dataset: The proposed dataset RGB2HANDS has a total of 1,724 frames which are divided into 4 sequences, where each sequence contains between 316 and 572 frames. To enable 3D evaluation, synchronized depth data are recorded. Using the depth camera calibration, 3D annotations can be obtained for the visible keypoints by backprojection. For quantitative comparisons, out of the 4 sequences, at least every 5th frame was annotated starting from the beginning of the interaction, resulting in a total of 319 annotated frames. The annotation was performed manually, where annotators were asked to identify the 14 interior joints of each hand as done for previous datasets Tzionas et al., 2016. If the location of an occluded joint could be inferred with high confidence, annotators marked this location while also flagging the occlusion to signify that depth cannot be recovered for 3D evaluation. If no reliable guess was possible, this joint was not annotated. Note that this is an advantage over the TZIONAS dataset where only visible joints are annotated.

Metrics: For quantitative comparisons in 2D and 3D, two metrics were used to compare the errors between the annotated ground-truth keypoints and corresponding estimates obtained. First, the mean per-keypoint error

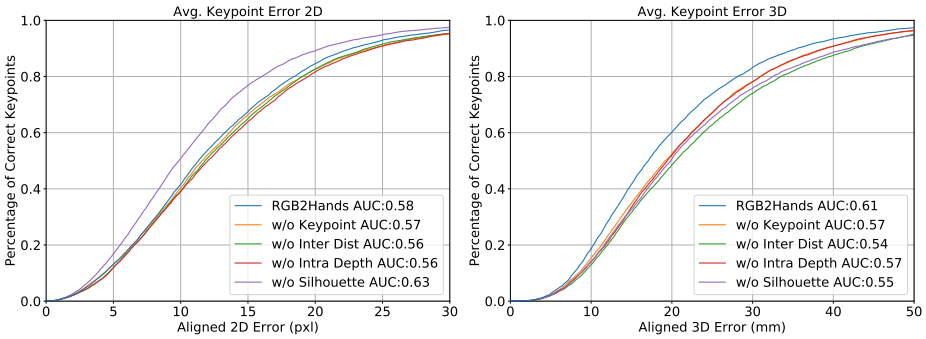


Figure 5.7: Energy term ablation study on the RGB2Hands dataset.

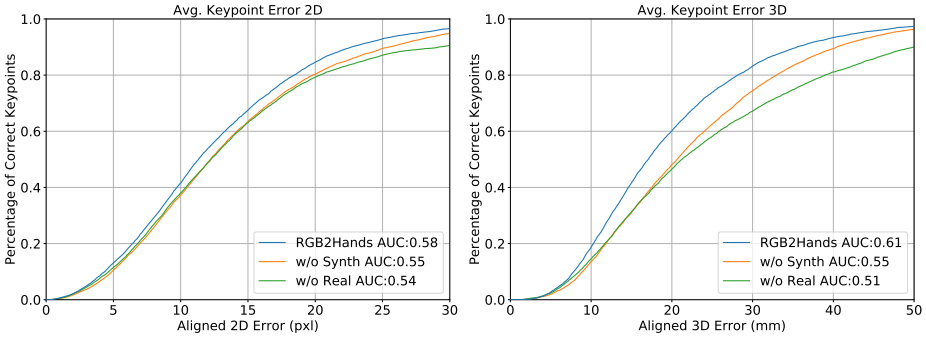


Figure 5.8: Training data ablation study on the RGB2Hands dataset.

in pixels for 2D or in millimeters for 3D are used. Second, to enable a more fine-grained analysis, the *Percentage of Correct Keypoints (PCK)* metric in 2D and 3D are employed. A keypoint estimate is counted as correct if its distance from the ground truth is less than t_{PCK}. By varying the threshold t_{PCK} on the horizontal axis, and showing the respective value on the vertical axis, a PCK curve is plotted.

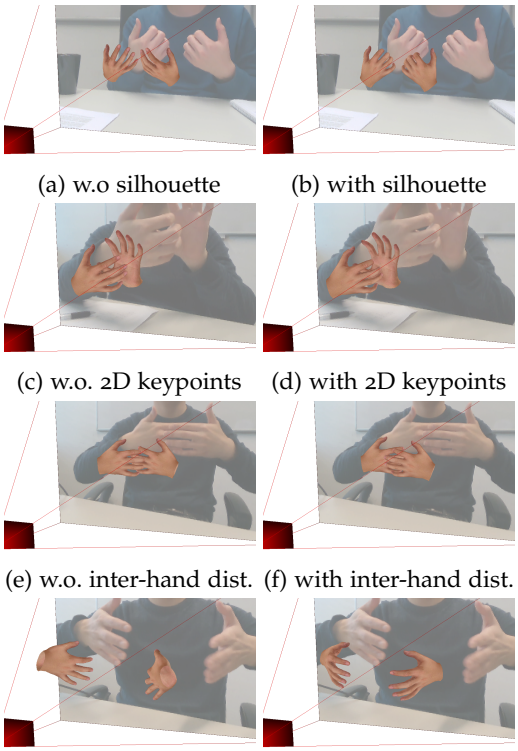
To address the inherent depth-scale ambiguity of RGB images in the 3D evaluation, the estimated keypoints were aligned to the ground truth using Procrustes analysis without rotation. Note that the alignment is performed for both hands jointly, i.e. a single translation and scale value is estimated for both. Hence, the aligned 3D error still captures the quality of the relative hand placement in 3D.

5.6.2 *Ablation Study*

For the ablation experiments, different settings were considered when evaluating the results on the proposed RGB2HANDS dataset. To be more specific, ablations were done to evaluate the effects of

(i) the individual terms in the fitting energy f in Equation 5.2,
(ii) the importance of using the real and the synthetic dataset.

Fitting Energy Terms: In Figure 5.7 the PCK curves across all sequences are shown when leaving out one of the terms in the proposed fitting function, compared to using the whole function in Equation 5.2. All of the terms improve the 3D error. It is notable that the silhouette term does this at the cost of 2D keypoint error. This could be due to the fact that the energy function without silhouette term has local minima with accurate 2D keypoints, but inaccurate 3D pose, which the silhouette term helps to escape from. In Figure 5.9, additional qualitative results of this ablation study are presented. To this end, tracking results with and without individual terms of the optimization problem are shown.



(a) w.o silhouette (b) with silhouette
(c) w.o. 2D keypoints (d) with 2D keypoints
(e) w.o. inter-hand dist. (f) with inter-hand dist.
(g) w.o. intra-hand dist. (h) with intra-hand dist.

Figure 5.9: Energy term ablation study.

Importance of the Proposed Datasets: Additionally, the behavior of the proposed hand tracker is analyzed when training the prediction networks either without the real dataset, or without the synthetic dataset, respectively, see Figure 5.8. When not using the real dataset, or when omitting the synthetic dataset, the PCK curves drop substantially (see green and orange lines), compared to using both datasets (blue line).

Method	Tzionas Dataset		RGB2Hands Dataset			Properties	
	2D Error (pixels)	Missed Frames	2D Error (pixels)	3D Error (mm)	Missed Frames	Output	Runtime (ms/frame)
Tzionas et al. (2016)	5.04	0	-	-	-	global 3D	4960
Mueller et al. (2019)	10.80	0	-	-	-	global 3D	33
Boukhayma et al. (2019)	12.91	13	19.31	27.47	20	weak-persp. 3D	(516) + 16
OpenPose	9.68	13	13.32	-	20	2D keypoints	516
Ours	13.31	0	13.43	20.02	0	global 3D (up to scale)	47

Table 5.2: The proposed method is compared with depth-based and RGB-based hand pose estimation methods on the Tzionas and the RGB2Hands datasets.

5.6.3 *Comparison to Other Methods*

In this section, evaluation is performed comparing the proposed method to existing depth-based as well as RGB-based methods on the RGB2Hands and the Tzionas dataset. Specifically, for depth-based methods, comparisons to Tzionas et al. (2016) and Mueller et al. (2019) are shown. For RGB-based methods, since there is no hand tracking system that was explicitly designed for such input modality and for the scenario of two closely interacting hands, comparisons to the single-hand method by Boukhayma et al. (2019) are shown. For a fair comparison, their procedure of cropping the image around the hand based on OpenPose keypoint predictions (Cao et al., 2019; Simon et al., 2017) is followed. This approach is applied for each hand independently, horizontally flipping the left hand images since their method was designed for right hands only. Although OpenPose does not respect a valid 3D hand geometry, and merely obtains 2D keypoint positions, for the sake of completeness comparisons to the plain OpenPose predictions is conducted.

Comparison on Tzionas Dataset: In Table 5.2 quantitative comparisons to Tzionas et al. (2016), Mueller et al. (2019), Boukhayma et al. (2019), and OpenPose is shown. Although in terms of mean error the proposed method performs worse than the depth-based method by Tzionas et al. (2016), it should be emphasized that theirs is an offline method that is about 100 times slower. However, the results from the proposed method is close to the depth-based real-time method by Mueller et al. (2019), despite the fact that they use much richer input data that contains 3D information. In comparison to the RGB-only method by Boukhayma et al. (2019), in terms of mean error the proposed method achieve results that are on par, while being significantly slower and thereby not real-time capable. In contrast to all other methods, the RGB-based OpenPose is trained to regress 2D keypoint locations which exactly matches the evaluated metric and hence yields a better result. However, it should be

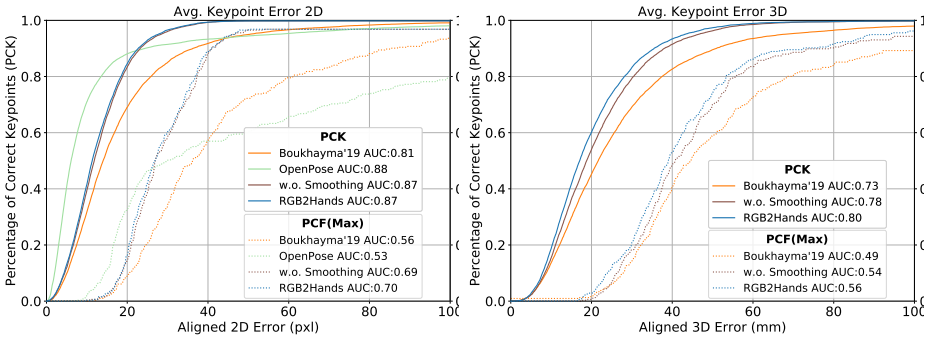


Figure 5.10: Quantitative comparison of the proposed method to Boukhayma et al. (2019) and OpenPose on the RGB2HANDS dataset.

pointed out that such 2D predictions generally do not represent plausible poses, which is also highlighted in the subsequent comparison using the new RGB2HANDS dataset. Contrary to the proposed full-frame method, the other two RGB-only methods require bounding boxes to obtain a hand crop. Consequently, there are 13 frames in the dataset for which no estimates are available due to missing bounding box detection. The proposed method also outputs global 3D pose and shape (up to a scale factor) and runs much faster compared to the other RGB-only methods.

As shown in Figure. 5.6, the TZIONAS dataset does not contain many frames with strongly interacting and overlapping hands. This is the main reason why the evaluated crop-based single-hand RGB methods succeed on this dataset. The advantages of our method become more apparent when compared on more challenging interaction scenarios, which will be presented next.

Comparison on RGB2Hands Dataset: The RGB2HANDS dataset is created to enable evaluation of more challenging hand interactions than previously seen in other datasets. Figure 5.10 shows quantitative results demonstrating that the proposed method (blue line) leads to substantially better PCK curves than the method by Boukhayma et al. (2019) (orange line). Although OpenPose appears to produce good results in terms of the percentage of correct individual keypoints (Figure 5.10, solid line), its percentage of correct frames (PCF), where a frame is considered correct if the maximum keypoint error is under a threshold, is substantially lower compared to others (Figure 5.10, dotted line). This confirms that OpenPose is often accurate for some of the keypoints in a frame while producing large errors for harder (e.g., occluded) keypoints in the same frame. This in turn is a strong indicator that the predicted 2D hand keypoints do not constitute a plausible hand pose due to the missing 3D

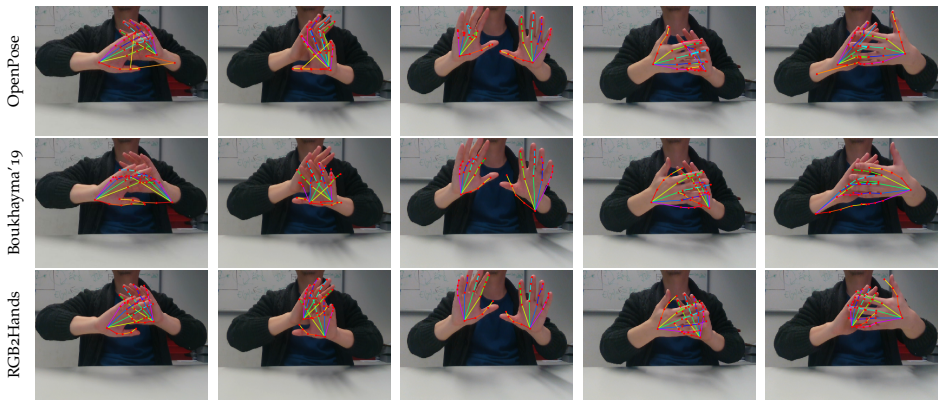


Figure 5.11: Qualitative comparison of the proposed RGB2Hands to Boukhayma et al. (2019) and OpenPose.

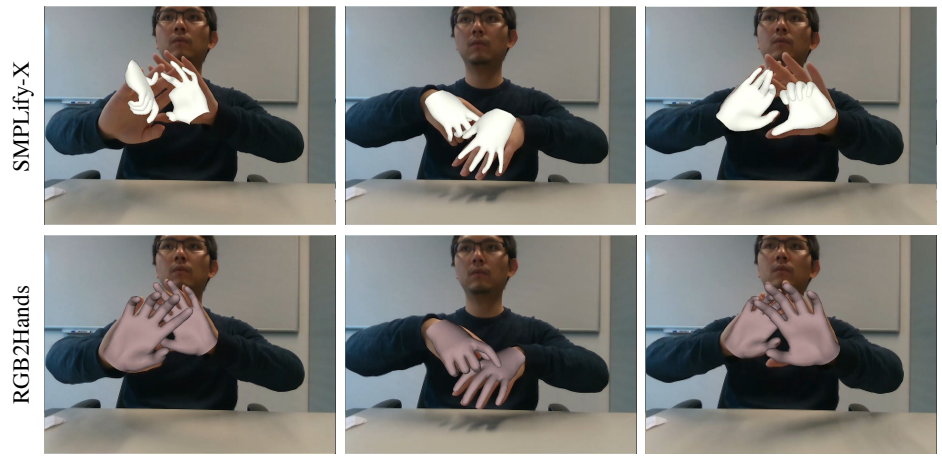


Figure 5.12: Qualitative comparison to SMPLify-X (Pavlakos et al., 2019).

model constraint. This can also be seen in Figure 5.11, where qualitative results are shown. In addition, OpenPose and hence also the method by Boukhayma et al. (2019) fail to detect the hands completely in several frames (see Table 5.2). Lastly, since competing methods do not perform temporal filtering, the proposed method is also evaluated without temporal smoothing ("w.o. Smoothing" in Figure 5.10). It can be seen that the proposed method still outperforms the competitors.

This evaluation on the new RGB2Hands dataset validates the need for methods that are specifically tailored to handle two strongly interacting hands. Running single-hand methods on crops of the two hands individually cannot jointly reason about the two hands, which is crucial for effectively dealing with close interactions.

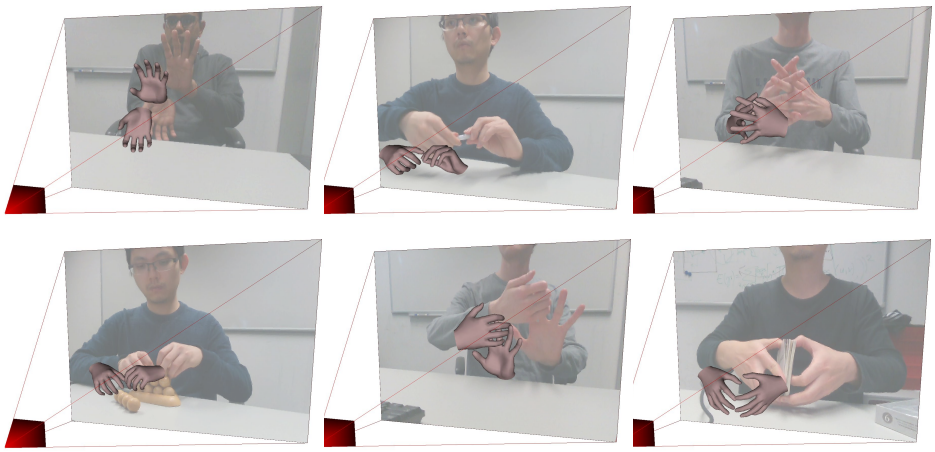


Figure 5.14: Additional results of the proposed RGB2Hands method.

QUALITATIVE COMPARISON TO SMPLIFY-X Qualitative comparisons to SMPLify-X (Pavlakos et al., 2019), which fits a full human body model to monocular RGB images, are shown in Figure 5.12, . Such methods rely on the estimated body pose to detect the hand and to regularize the hand orientation. As such, the proposed method is more stable when the body is not fully visible. SMPLify-X also does not explicitly address overlapping or interacting hands and hence also fails when the hand detection and orientation are correctly estimated.

5.6.4 *Additional Qualitative Results*

Next, the global 3D tracking of two interacting hands in various involved settings are shown. The purpose of this section is to demonstrate the generality and the wide scope of hand tracking scenarios and non-trivial two-hand interactions that the proposed method is capable of handling in real time. Results are also shown on single hand scenes to emphasize that the proposed formulation does not require both hands to be present. Visualizations can be found in Figure 5.1, Figure 5.13, and Figure 5.14.

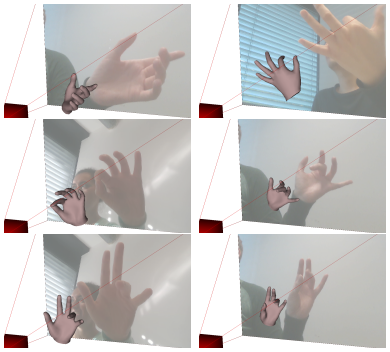


Figure 5.13: Results on single hand scenes.

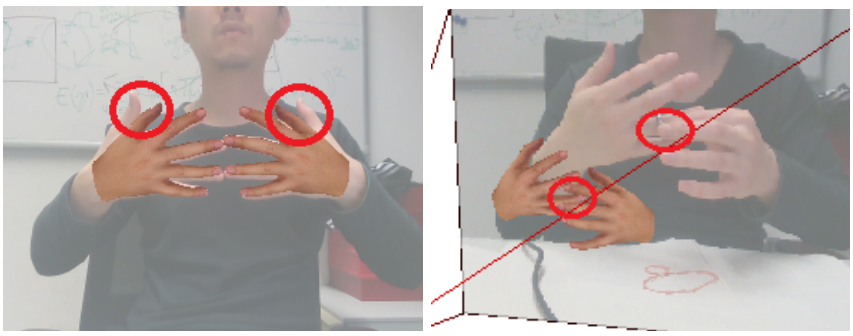


Figure 5.15: Example Failure Cases

5.7 DISCUSSION & FUTURE WORK

Overall this chapter has presented compelling 3D tracking and reconstruction results on challenging sequences of two interacting hands. One important property of this approach is that it directly works on the full input image, rather than explicitly localizing a hand first, and then using a cropped image for further processing. This is in contrast to existing single-hand methods, both RGB-based and depth-based, which could in principle also be applied to the tracking of two hands (by localizing and processing each hand individually). However, these methods oftentimes fail in the case of heavy hand-hand interactions, since in this case it is not possible to obtain a reliable crop, or the visibility of parts of the other hand lead to errors due to severe self-similarities.

Despite the overall good performance of the proposed method, particularly for close hand-hand interaction settings, there are also some downsides that can be addressed in the future. Currently, the proposed method may not always be able to correctly track very fast hand motions, since in this case motion blur may lead to unreliable predictions of the neural network. One potential way to address this is to also include data with simulated motion blur, so that the neural network is able to deal with such cases. Moreover, it is difficult to find a good trade-off between the MANO pose prior and the other energy terms, so that one has to sacrifice either pose variability or pose plausibility. This is most noticeable for thumb articulations (Figure 5.15, left). This could for example be addressed by equipping the MANO model with a kinematic skeleton, and then enforcing explicit joint limit constraints while still using the pose space to capture correlations in joint articulations. Due to inherent depth ambiguity, the proposed method may also have difficulties reconstructing interactions where high precision in relative hand positioning is required; e.g. slotting a ring onto a finger (see Figure 5.15, right). For such tasks, additional cues from a depth sensor or a stereo camera might

be requires. It would be interesting as well to explore the explicit use of the temporal dimension, so that for example hand shape information can be integrated over time, in a similar spirit to bundle adjustment in multi-view reconstruction. Moreover, temporal neural network architectures can be used to obtain temporally smoother predictions and thus further improve temporal tracking consistency. Another open point is optimizing for person-specific hand textures based on a parametric hand texture space.

5.8 CONCLUSION

This chapter has presented the first approach that is specifically tailored towards tracking and reconstruction of two hands in interaction in global 3D from only RGB video. A major challenge in this setting are depth ambiguities, which is addressed here by combining two strong priors, one in form of a parametric 3D hand model, and the other one in form of a multi-task neural network predictor that is trained based on a large body of real and synthetic training data. For training, existing datasets are combined with two new proposed datasets that created specifically for this task. The first one is a real dataset for which (potentially noisy) annotations were obtained based on RGB-D frames. It is complemented by a new synthetic dataset that models physically correct hand interactions while taking hand variability in terms of shape and appearance into account. Moreover, a new benchmark dataset, RGB2HANDS, is introduced which contains annotated sequences showing significantly stronger interactions between two hands in comparison to previous benchmarks. The proposed approach is shown to outperform previous RGB-only methods in complex hand-hand interaction settings, both quantitatively and qualitatively, and even performs on par with a state-of-the-art depth-based real-time approach.

6 MODELING HAND INTERACTION UNCERTAINTY OF MONOCULAR INPUT

While the previous chapter proposed the first method to reconstruct two interacting hands from monocular RGB input, the prediction is a single plausible reconstruction given the ambiguities that stem from projective geometry and heavy occlusions. The fact that many other valid reconstructions exists that fit the image evidence equally well is not reflected in this point estimate. This chapter (published as Wang et al., 2022) propose to address this issue by explicitly modeling the distribution of plausible reconstructions in a conditional normalizing flow framework. This allows the posterior distribution to be directly supervised through a novel determinant magnitude regularization, which is key to varied 3D hand pose samples that project well into the input image. It is also demonstrated that metrics commonly used to assess reconstruction quality are insufficient to evaluate pose predictions under such severe ambiguity. To address this, this chapter propose the first dataset with multiple plausible annotations per image, called MultiHands. The additional annotations enable evaluation of the estimated distributions using the maximum mean discrepancy metric. Through this, the quality of the proposed probabilistic reconstruction is demonstrated and it is shown that explicit ambiguity modeling is better-suited for this challenging problem.

6.1 INTRODUCTION

Reconstructing two interacting hands in 3D is an actively researched topic, as it enables applications in various areas of vision and graphics, including augmented and virtual reality, robotics, or sign language translation. While earlier methods leverage multi-camera setups (Ballan et al., 2012; Sridhar et al., 2013) or depth sensors (Mueller et al., 2019; Taylor et al., 2017), recent works focus on using monocular RGB cameras to enable potential applications in mobile or wearable settings.

However, hand pose estimation from monocular RGB images is a very challenging problem. Hand interactions lead to severe occlusions; and monocular color images exhibit an inherent depth and scale ambiguity. Existing methods (Moon et al., 2020; Wang et al., 2020a; Zhang et al., 2021) aim to *deterministically* estimate the relative depth between the two hands directly. However, this is prone to error in heavily occluded situations

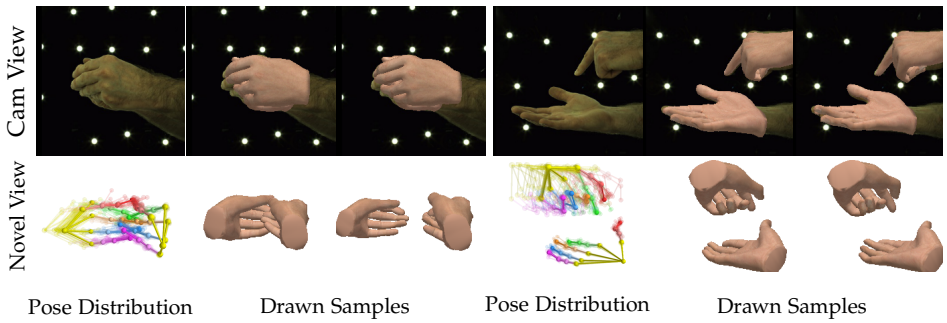


Figure 6.1: Given a single RGB image of two-hand interaction, the proposed method predicts a probability distribution of plausible 3D hand poses that could explain the image. Two different samples projected into the image (top) and into a new viewpoint (bottom) are shown. The proposed probabilistic approach captures the inherent ambiguity of monocular two-hand interaction images.

due to the ill-posed nature of the problem. For example, a small error in hand scale or depth can cause a significant difference in touch points and hence semantics of the interaction. As a result, most methods evaluate the pose of each hand independently using the root-relative pose error which discards important information regarding the the positioning of the hands with respect to each other.

Given these extreme challenges, this chapter proposes to instead explicitly model the ambiguities (see Figure 6.1). Inspired by previous work on reconstruction of human body and face (Kolotouros et al., 2021; Kortylewski et al., 2018; Schönborn et al., 2017; Wehrbein et al., 2021), the proposed approach aims to predict a distribution over likely two-hand poses. To this end, normalizing flow (Rezende and Mohamed, 2015) is adopted as a way to parameterize the posterior distribution that enables not only fast sampling but also differentiable likelihood estimates. This allows the formulation of a novel loss to supervise the shape of the distribution. The proposed regularization term encourages diversity in distribution without sacrificing image consistency, which is key to modeling the severe ambiguities in the target setting.

It is quantitatively demonstrated that the sampled reconstructions capture the range of plausible articulations better than existing state-of-the-art methods. This is facilitated by a new proposed dataset, MultiHands, the first to provide multiple plausible annotations per image for measuring the accuracy of distribution predictions.

In summary, the main contributions of this chapter are:

- A method for reconstructing two-hand interactions that can generate diverse 3D poses which match the observed image.

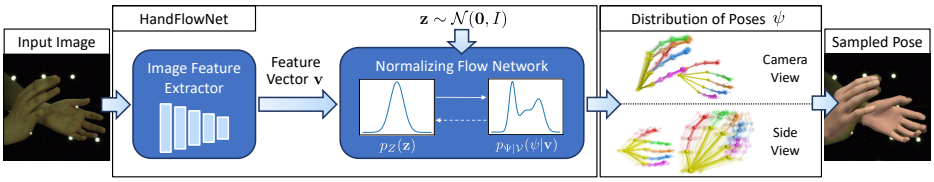


Figure 6.2: The proposed *HandFlowNet* first extracts an image feature vector $\mathbf{v}$ from 2D cues in the input image. The feature vector is then used as conditioning input to a normalizing flow network to output a distribution of 3D hand poses that plausibly explain the monocular input.

- A new regularization term for training conditional normalizing flow to encourage diversity of samples.
- The first dataset to account for pose ambiguity by providing multiple pose annotations.

Finally, it is demonstrated that the estimated pose distribution can be leveraged for disambiguating view-point selection, a downstream application not possible with deterministic approaches.

6.2 METHOD

The goal of the method is to estimate a distribution of 3D hand poses that are plausible to explain a given monocular color image. To address this, *HandFlowNet* is proposed. The method first extracts a feature encoding from the input image, which is then used to generate the desired output pose distribution from a normalizing flow network (Section 6.2.2). The estimated 3D hand poses are parameterized using the MANO hand model (Section 6.2.1).

6.2.1 *Hand Model*

The MANO hand model (Romero et al., 2017) is used to represent the hand surface with additional parameters used for the rigid transformation. The parameterization of a single hand is first described, which is then readily expanded to two hands.

Given 15 joint rotations $R \in \mathbb{R}^{15\times3\times3}$ represented as stacked rotation matrices and shape parameters $\beta \in \mathbb{R}^{10}$, the MANO model computes both the hand surface as a mesh and 3D hand keypoint positions. In order to place the hand correctly relative to the camera, the global rotation parameters $r \in \mathbb{R}^{3\times3}$, the hand root position in image coordinates $t \in \mathbb{R}^2$, and the perspective scale factor $s \in \mathbb{R}$ are additionally estimated. This enables the recovery of the global pose when the focal length is known

at inference (Boukhayma et al., 2019). The combined global and joint rotations $\{r, R\}$ are parameterized using the 6 DOF representation $\theta \in \mathbb{R}^{16\times3\times2}$ as proposed in Zhou et al. (2019)

Therefore, the full set of parameters for a single hand is defined as $\psi = \{\theta, \beta, t, s\} \in \Psi$, where Ψ denotes the parameter space, and the full set of parameters for both hands is defined as $\psi_{\text{both}} = [\psi_{\text{right}}, \psi_{\text{left}}]$. In the following, ψ_{both} will be referred to simply as ψ.

6.2.2 *HandFlowNet*

Given a monocular input image, *HandFlowNet* regresses a distribution of 3D hand poses corresponding to plausible hand poses that could be observed in the image (see Figure 6.2). *HandFlowNet* can be divided into two parts, an image feature extractor and a conditional normalizing flow network that produces a 3D pose distribution and is conditioned on the extracted image feature vector.

Image Feature Extractor: The image feature extractor summarizes the visible, unambiguous features that the sampled poses should reconstruct. ResNet-50 (He et al., 2016) is used as the backbone architecture. From an input image with resolution 224×224, the 2048-dimensional feature vector $\mathbf{v} \in \mathcal{V}$ is extracted from the average pooling of the last residual block, and is used as the conditional vector for the next step.

Normalizing Flow Network: To predict a range of plausible poses, a way to parameterize a pose distribution $p_Y(\mathbf{y})$ must be chosen.

Normalizing flow (Rezende and Mohamed, 2015) does this by learning an invertible transformation $f : \mathbb{R}^d \to \mathbb{R}^d$ of a simple distribution $p_Z(\mathbf{z})$, i. e.

$$p_Y(\mathbf{y}) = p_Z(\mathbf{z}) \left| \det \frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} \right|^{-1} . \tag{6.1}$$

where $\mathbf{y} = f(\mathbf{z})$. This invertible parameterization allows for both differentiable sampling and likelihood estimation. As a result, losses can be applied on each sample to improve reconstruction quality, while supervising the entire distribution using negative log likelihood loss and multiple annotations (as will be discussed in Section 6.2.3). Since a distribution over the space of 3D hand poses Ψ is to be estimated given an image feature vector $\mathbf{v}$, the conditional distribution $p_{\Psi|\mathcal{V}}(\psi|\mathbf{v})$ is of interest. To enable this, normalizing flow can be extended to conditional normalizing flow (Winkler et al., 2019) by using transformations $f_{\mathbf{v}} : \mathbb{R}^d \to \mathbb{R}^d$ parameterized by $\mathbf{v}$, so that

$$p_{\Psi|\mathcal{V}}(\psi|\mathbf{v}) = p_{Z|\mathcal{V}}(\mathbf{z}|\mathbf{v}) \left| \det \frac{\partial f_{\mathbf{v}}(\mathbf{z})}{\partial \mathbf{z}} \right|^{-1} . \tag{6.2}$$

For implementation, the conditional GLOW architecture is used for $f_{\mathbf{v}}$ which has been successfully used in previous work (Kolotouros et al., 2021) due to its quick sampling and probability estimation. For a more detailed overview, please refer to Kobyzev et al. (2020).

By setting $p_{Z|\mathcal{V}} = p_Z \sim \mathcal{N}(\mathbf{0}, I)$, the mode of the distribution $p_{\Psi|\mathcal{V}}(\psi|\mathbf{v})$ can be obtained as $f_{\mathbf{v}}(\mathbf{0})$. This design is chosen to provide easy access to the mode sample for use in the losses.

6.2.3 *Training Losses*

In the following, the losses used for training are detailed. The entire loss is given by

$$\mathcal{L} = \mathcal{L}_{\text{nll}} + \mathcal{L}_{\text{DetMag}} + \mathcal{L}_{\psi} + \mathcal{L}_{\mathcal{J}_{3\text{D}}} + \mathcal{L}_{\mathcal{J}_{2\text{D}}} + \mathcal{L}_{\theta}\,. \tag{6.3}$$

Here, $\mathcal{L}_{\text{nll}}$ and $\mathcal{L}_{\text{DetMag}}$ are used to supervise the likelihood of the annotations, and $\mathcal{L}_{\psi}$, $\mathcal{L}_{\mathcal{J}_{3\text{D}}}$, $\mathcal{L}_{\mathcal{J}_{2\text{D}}}$, and $\mathcal{L}_{\theta}$ are used to supervise the quality of the sampled reconstructions. For network training parameters and loss weights, please refer to Appendix A.7.

Maximum Likelihood Estimation: Given images and their 3D annotation, the probability of the pose annotation ψ^* should be maximized. Hence, the negative log likelihood (NLL) loss is minimized

$$\begin{aligned}\mathcal{L}_{\text{nll}} &= -\ln p_{\Psi|\mathcal{V}}(\psi^*|\mathbf{v}) \\ &= -\ln p_Z(f_{\mathbf{v}}^{-1}(\psi^*)) \left|\det \frac{\partial f_{\mathbf{v}}(\mathbf{z})}{\partial \mathbf{z}}\right|^{-1}\,.\end{aligned} \tag{6.4}$$

When multiple annotations $\{\psi_0^*, ..., \psi_n^*\}$ are available, the NLL loss is minimized over all annotated poses.

Enhancing Pose Variety: It is observed that training the network using just the term $\mathcal{L}_{\text{nll}}$ quickly collapses the variety in the output pose distribution. To explain this, note that $\mathcal{L}_{\text{nll}}$ maximizes $\left|\det \frac{\partial f_{\mathbf{v}}(\mathbf{z})}{\partial \mathbf{z}}\right|^{-1}$, which describes the compression factor between the two spaces for density conservation. Therefore, the network can trivially optimize the conditional distribution by concentrating the density in the pose space, leading to the collapse in $p_{\Psi|\mathcal{V}}(\psi|\mathbf{v})$. To prevent this, the following regularization term is added

$$\mathcal{L}_{\text{DetMag}} = -\ln \left|\det \frac{\partial f_{\mathbf{v}}(\mathbf{z})}{\partial \mathbf{z}}\right|\,. \tag{6.5}$$

Since this term aims at increasing variation in the output distribution of the normalizing flow network only, its gradient is not backpropagate into

the image feature extractor. Otherwise, the extraction network might be hindered in learning pose-relevant features.

Mode Supervision: While $\mathcal{L}_{\text{nll}}$ encourages the probability of the pose annotations to be maximized, the mode sample $f_{\mathbf{v}}(\mathbf{0})$ should also be a valid reconstruction. To accomplish this, the following loss is used

$$\mathcal{L}_{\psi} = ||f_{\mathbf{v}}(\mathbf{0}) - \psi^*||_2^2 \,. \tag{6.6}$$

Note that $\mathcal{L}_{\psi}$ is complementary to $\mathcal{L}_{\text{nll}}$ and both together form a two-sided loss that ensures plausible pose predictions. When multiple annotations exist, one fixed pose is randomly chosen as the mode sample.

Although data with MANO parameter annotation exists, the amount is limited compared to the amount of data with joint position annotations. To make use of all available data, an additional 3D joint position loss on all N_J joints is imposed

$$\mathcal{L}_{\mathcal{J}_{\text{3D}}} = \sum_{i=1}^{N_J} ||\mathcal{J}(\psi)_i - P_i^{\text{3D}}||_2^2 \,, \tag{6.7}$$

where $\mathcal{J}$ is a function defined by the hand model that calculates the 3D joint positions given pose parameters ψ, and P^{3D} are the joint annotations.

2D Consistency: *HandFlowNet* aims to provide a distribution of poses that all correspond to the same input image. Hence, the 2D position of visible joints should be the same for the mode and the samples of the distribution, and should thus match the annotation. The loss

$$\mathcal{L}_{\mathcal{J}_{\text{2D}}} = \sum_{i=1}^{N_J} \eta_i \, ||\Pi(\mathcal{J}(\psi)_i) - P_i^{\text{2D}}||_2^2 \,, \tag{6.8}$$

is employed, where Π is the known camera projection, P^{2D} are the 2D joint position annotations, and $\eta_i = 1$ if the joint i is visible and 0 otherwise. These visibility scores are computed from the meshes of MANO pose annotations. $\mathcal{L}_{\mathcal{J}_{\text{2D}}}$ is calculated on the mode of the distribution $f_{\mathbf{v}}(\mathbf{0})$ and on two samples from the estimated distribution $p_{\Psi|\mathcal{V}}(\psi|\mathbf{v})$.

Rotation Regularization: As explained in Section 6.2.1, the continuous 6-dimensional representation for 3D rotations proposed by Zhou et al. (2019) is used. The representation is not unique, i. e., there are multiple $A \in \mathbb{R}^{3\times 2}$ that represent the same 3D rotation $R \in \text{SO}(3)$. To encourage consistent output, a regularizer is added (Kolotouros et al., 2021) that constrains all rotations in their 6-dimensional representation A to be orthonormal

$$\mathcal{L}_{\theta} = \sum_{A\in\theta} ||A^{\top}A - I||_F^2 \,. \tag{6.9}$$

6.3 CREATING ADDITIONAL ANNOTATIONS

While there exists a single *ground-truth* pose, i.e. the one that forms a given image, recovering this exact pose from an image is ambiguous since there are multiple *plausible pose annotations*. Since the goal is to model this ambiguity with a distribution, the single ground truth found in most datasets is not sufficient for evaluating the predictions and more annotations are needed.

The following section describes how additional annotations are obtained from a provided MANO ground truth.

Plausible Pose Annotations: Given the ground-truth pose parameters ψ_{gt} and a camera projection Π, an annotation ψ_{annot} is plausible if the hand joints fit the observed image and the overall articulation is anatomically possible. To ensure this, the following criteria are used:

- The 2D locations of visible joints should be within a pixel threshold of the ground-truth locations.
- Occluded joints in the original pose should remain occluded.
- The pose should be anatomically likely as measured using the pose PCA space of the MANO model (Romero et al., 2017). A likelihood threshold is used to eliminate extreme articulations.
- No collision between hands. Collisions are detected using Gaussian proxies (Mueller et al., 2019) attached to the MANO model. Collision occurs when the one-standard-deviation spheres of the Gaussian proxies intersect each other.

Annotation Generation: Starting from the ground-truth pose parameters ψ_{gt}, the hand pose parameters are perturbed to generate hand pose proposals. These proposals are checked for plausibility as defined in the above criteria, and implausible pose annotations are rejected. The accepted plausible pose annotations will now serve as new starting poses for the next iteration. This perturbation and plausibility checking is repeated for a fixed number of iterations to obtain the final plausible pose annotations. For additional implementation details, please refer to Appendix A.8.

6.4 EXPERIMENTAL RESULTS

The proposed method is evaluated on existing datasets (Section 6.4.1), and the limitations of commonly used metrics in dealing with ambiguity

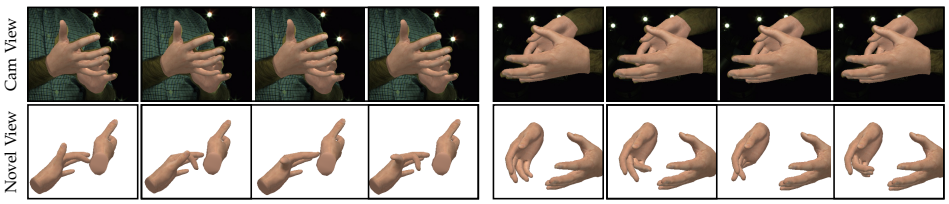


Figure 6.3: The proposed MultiHands dataset captures the ambiguities of monocular input with diverse 3D reconstructions that fit the input images.

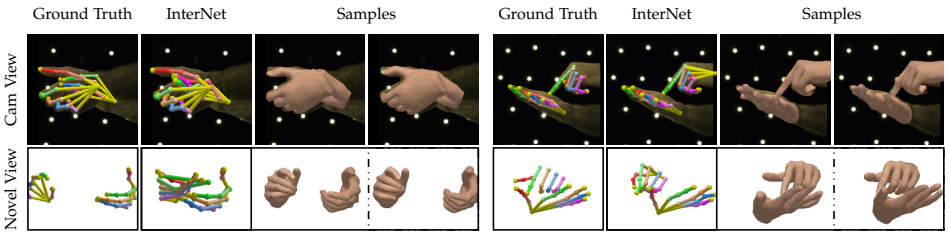


Figure 6.4: The predicted samples are consistent in the camera view (top row), while the 3D diversity can be seen in the novel view (bottom row). This diversity allows for samples that are close to the single InterHand2.6M ground truth, while the deterministic InterNet fails due to ambiguity or heavy occlusions.

are discussed (Section 6.4.2 and 6.4.3). To deal with this ambiguity, an alternative metric is proposed (Section 6.4.3) for evaluation (Section 6.4.7, 6.4.6). Finally, an application beyond pose estimation is shown to demonstrate the advantages of distribution estimation (Section 6.4.8).

6.4.1 *Datasets*

Here each dataset along with practical considerations to be taken into account to run the experiments are described.

InterHand2.6M Dataset: 673,514 training frames labeled as interacting hands from the dataset of Moon et al. (2020) are used for training. Notice that the terms $\mathcal{L}_{\text{nll}}$, $\mathcal{L}_{\psi}$ from Equation 6.4, 6.6, respectively, require MANO parameter annotations. These losses are applied to the subset of 394,599 frames where these are available.

Following the method of InterNet (Moon et al., 2020), RootNet (Moon et al., 2019) is used for hand detection. A 334×334 crop centered around the provided bounding box is used for the image feature extractor.

MultiHands Dataset: Using the method described in Section 6.3, InterHand2.6M is extended with 100 additional annotations for each of the 281,369 test and 394,599 training images with MANO annotations. Since

the losses $\mathcal{L}_{\text{nll}}$ and $\mathcal{L}_{\text{DetMag}}$ can use multiple annotations, MultiHands is also for training. See Figure 6.3 and Figure A.2 for example annotations.

Tzionas Dataset: To demonstrate that the learned 3D pose distribution generalizes to other settings, qualitative results are shown on the Tzionas Dataset (Tzionas et al., 2016). This dataset has seven sequences captured in an office environment with only 2D annotations.

Following Moon et al. (2020), 90% of the annotated 2D frames are used for training and results are shown on the remaining 10%.

6.4.2 *Pose Alignment*

Three different alignments are used to evaluate the mean per-joint position error (MPJPE) in mm. All equations can be found in Appendix A.9. Root-Relative (RR) MPJPE captures the errors in articulation, where each hand is individually root-aligned. Right-Root-Relative (RRR) MPJPE measures the accuracy of the two hands together, where both hands are aligned to just the root of the right hand. Global MPJPE captures the accuracy of the global pose estimate, *without any alignment*.

Although the RR metric is most commonly reported in the literature, it evaluates the two hands independently by ignoring the relative hand placements. Since this placement is vital for most applications, the analysis show and focus on the RRR and Global metrics.

6.4.3 *Problem with Traditional Metrics*

When the observed image is ambiguous, the choice of the target pose can greatly impact the MPJPE even though equally valid alternatives exist. To quantify this effect on InterHand2.6M, InterNet (Moon et al., 2020) and Fan et al. (2021) predictions are evaluated against the closest and farthest annotation in MultiHands (Table 6.1).

Method	Global ↓	RRR ↓	RR* ↓
InterNet (min)	67.2	24.5	22.6
InterNet (max)	103.6	42.2	24.6
Fan et al. (min)	65.7	27.1	20.5
Fan et al. (max)	102.1	45.9	22.5

Table 6.1: MPJPE in mm of deterministic methods. For RR*, the MPJPE is reported for occluded joints.

For the challenging Global and RRR metrics, the choice of plausible annotation accounts for a difference of 36mm and 18mm on average. Even when each hand is evaluated independently with the RR metric, the occluded joints differ by 2mm on average.

This sensitivity to the choice of annotation makes MPJPE unsuitable for the highly ambiguous task of monocular two-hand reconstruction.

Instead, a metric that measures the distances between two pose distributions would better reflect the quality of the predictions.

6.4.4 *Maximum Mean Discrepancy (MMD)*

How well the estimated distribution matches the annotation distribution can be measured using the MMD (Gretton et al., 2012).

The empirical MMD can be estimated given sampled predictions, multiple annotations, and the selection of a kernel function. 100 samples and annotations were used, and Gaussian kernels were chosen for evaluation. All reported MMD are averaged over different kernel distance scales (see Appendix A.9.2 for equations).

6.4.5 *Comparison to the State of the Art*

Competing methods: The widely applied probabilistic baselines Monte Carlo dropout (MC-dropout) (Gal and Ghahramani, 2016), aleatoric uncertainty (Gaussian) (Kendall and Gal, 2017), and Variational Auto Encoder (VAE) (Kingma and Welling, 2014) are implemented for comparisons. The implementation details can be found in Appendix A.10. As reference, comparisons against deterministic methods, Fan et al. (2021) and Moon et al. (2020), are also made by treating the estimates as a Dirac delta distribution. Given each method, 100 pose samples are drawn to find the MMD to ground-truth samples. MMD is computed for RR, RRR, and Global alignment to better understand the sources of ambiguity.

Results: Overall, the proposed method produces estimates that best match the ground-truth distribution (Table 6.2). This is especially notable for the challenging Global and RRR MMD metric, which demonstrates the benefits of the proposed formulation under ambiguity. State-of-the-art deterministic methods fail to account for ground truth variability (Figure 6.4). As a result, they have one of the worst MMD. See Appendix A.12 for more visualizations.

Method	Global ↓	RRR ↓	RR ↓
HandFlowNet	**0.50**	**0.42**	**0.44**
VAE	0.61	0.47	0.48
Gaussian	0.82	0.51	0.46
MCDropout	0.91	0.60	0.51
InterNet	1.12	0.59	0.56
Fan et al.	1.12	0.63	0.50

Table 6.2: Evaluation of predicted distributions in MMD.

For reference, a comprehensive evaluation of the proposed method using the single provided InterHand2.6M annotation can be found in Appendix A.11. There, it is shown that the best sample of the proposed

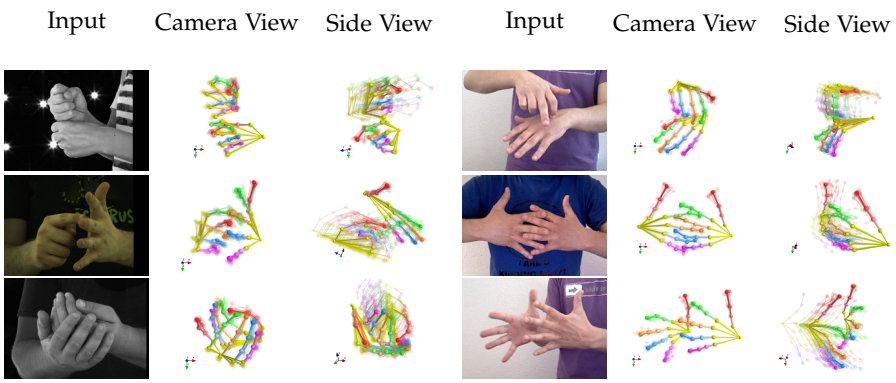


Figure 6.5: Here 30 pose samples are shown superimposed as semi-transparent skeletons. Samples are aligned to the root joint of one hand and the mode of the distribution is made opaque for ease of visualization. Examples are from the InterHand2.6M dataset (left) and the Tzionas dataset (right), where learned 3D ambiguity modeling are transferred using only 2D annotations.

method out-performs the state-of-the-art methods while still remaining competitive as a single pose estimator.

6.4.6 *Ablation Study*

It is shown in Table 6.3 that every loss helps to make the predicted samples match the ground-truth distribution. In particular, the proposed determinant magnitude regularization $\mathcal{L}_{\text{DetMag}}$ is vital for increasing the diversity of 3D samples. The mean standard deviation of the joint positions is improved from 18 to 31 mm while lowering the MMD. Lastly, it is observed that using multiple annotations from MultiHands in the $\mathcal{L}_{\text{nll}}$ and $\mathcal{L}_{\text{DetMag}}$ terms further improves the MMD, which demonstrate the advantage of the differentiable likelihood estimation in the normalizing flow formulation.

Method	Global ↓	RRR ↓	RR ↓
HandFlowNet	**0.50**	**0.42**	**0.44**
w.o. MultiHands	0.53	0.44	0.46
w.o. $\mathcal{L}_{\text{DetMag}}$	0.72	0.49	0.46
w.o. $\mathcal{L}_{\mathcal{J}_{3D}}$	0.65	0.62	0.52
w.o. $\mathcal{L}_{\mathcal{J}_{2D}}$	0.74	0.74	0.46
w.o. $\mathcal{L}_{\psi}$	0.55	0.42	0.45
w.o. $\mathcal{L}_{\theta}$	0.61	0.46	0.45

Table 6.3: Ablation study on dataset and loss terms. All results are in MMD.

6.4.7 *More Qualitative Results*

In Figure 6.5 and Appendix A.12, qualitative results are shown to demonstrate the diversity and accuracy of the learned pose distribution. Specifically, pose samples are shown visualized as superimposed transparent kinematic skeletons. Note that pose variations well reflect the expected monocular ambiguity, and occlusions further increase variability. Hence, the standard deviation of the samples can serve as an indicator for the ambiguity in the input image and thus uncertainty in the pose prediction.

6.4.8 *Application: View Selection*

By using the sample standard deviations to estimate pose ambiguity, camera views that provide the most information for a given motion sequence can be identified. Such information can be useful, for example, in a multi-view capture setup where uninformative cameras can be removed to reduce the hardware and data bandwidth requirements. This is demonstrated on the InterHand2.6M test set with over 100 images in the sequence. The data consist of the 7 sequences in Capture0-1 with interacting hands, each with 140 camera views.

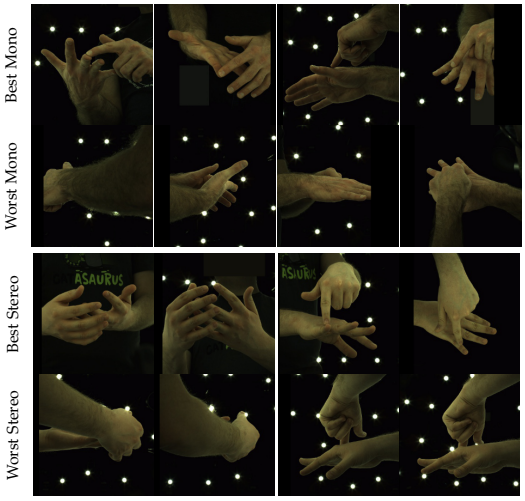


Figure 6.6: Each column shows the **Best Mono** and **Worst Mono** views selected with regards to ambiguity; Likewise, the **Best Stereo** and **Worst stereo** pairs are shown.

The view quality is evaluated using regret (Berry and Fristedt, 1985) in MPJPE: the difference between the MPJPE on the selected view and the lowest MPJPE. The best and worst views selected by the proposed method have a regret of 3.1 and 15.9 mm respectively, while the average regret over the cameras is 10.7 mm. This shows that the proposed method is able to eliminate cameras with ambiguous views where the monocular pose estimator is not expected to perform well, while keeping cameras views where the estimator is likely to succeed.

This view selection task can be extended to stereo camera pairs by combining two monocular pose distributions. By assuming conditional independence, the pose samples from each view can be approximated

with normal distributions and thus can be combined by taking their product. See Figure 6.6 for a qualitative evaluation of the selected views.

6.5 LIMITATIONS AND FUTURE WORK

Although promising results are demonstrated, there are some limitations that could be addressed in future work.

Currently, physically implausible intersections are not penalized in the reconstructions. As demonstrated in related work (Hasson et al., 2019; Wang et al., 2020a), an explicit loss to prevent these intersections could be used to improve the results.

Although promising generalization results are shown on the Tzionas dataset, the method does not tackle completely unconstrained in-the-wild images. This can be solved in the future with more data, especially 2D annotations for in-the-wild data.

While the experiments verified the need for probabilistic pose estimates in ambiguous scenarios, many applications can only make use of a single pose prediction. Future work could investigate ways to integrate additional observations (e. g., temporal information, multi-view images, depth images, task-based priors) to disambiguate the output distribution for a given down-stream task.

6.6 CONCLUSION

This chapter have presents the first two-hand reconstruction approach to explicitly model the inherent ambiguities that arise from using a single monocular input image. Given this challenging setting, the method produces a distribution of plausible reconstructions, from which diverse 3D pose samples can be drawn that all explain the observed image evidence. Additionally, existing evaluation schemes for the performance of hand pose estimation methods are shown to be problematic as they assume a single correct pose even though multiple solutions are equally valid. Along with the proposed dataset with multiple annotations and the distribution metric, the work presented in this chapter demonstrates the need for probabilistic approaches and provides a way to evaluate them.

7 PARAMETRIC HAND TEXTURE MODEL

While the previous chapters focus on leveraging model-based priors to resolve ambiguities in order to achieve more robust *geometric* reconstruction, they offer no mechanism to model the hand *appearance*. To address this, this chapter presents HTML, the first parametric texture model of human hands (published as Qian et al., 2020. The proposed model spans several dimensions of hand appearance variability (e. g. related to gender, ethnicity, or age) and only requires a commodity camera for data acquisition. It is experimentally demonstrated that the proposed appearance model can be used to tackle a range of challenging problems such as 3D hand reconstruction from a single monocular image. Furthermore, the proposed appearance model can be used to define a neural rendering layer that enables training with a self-supervised photometric loss.

7.1 INTRODUCTION

Hands are one of the most natural ways for humans to interact with their environment. As interest in virtual and augmented reality grows, so does the need for reconstructing a user's hands to enable intuitive and immersive interactions with the virtual environment. Ideally, this reconstruction contains accurate hand shape, pose, and appearance. However, it is a challenging task to capture a user's hands from just images due to the complexity of hand interactions and self-occlusion. In recent years, there has been significant progress in hand pose estimation from monocular depth (Baek et al., 2018; Oberweger et al., 2015; Tompson et al., 2014; Wan et al., 2017) and RGB (Mueller et al., 2018; Yang and Yao, 2019; Zimmermann and Brox, 2017) images. Although most of these works estimate only joint positions, a few recent works attempt to reconstruct the hand geometry as well (Boukhayma et al., 2019; Malik et al., 2020; Zhang et al., 2019).

Despite these recent advances, there is little work that addresses the reconstruction of hand appearance. However, hand appearance personalization is important for increasing immersion and the sense of "body-ownership" in VR applications (Jung and Hughes, 2016), and for improved tracking and pose estimation through analysis-by-synthesis approaches. Without a personalized appearance model, existing pose estimation methods must use much coarser hand silhouettes (Boukhayma

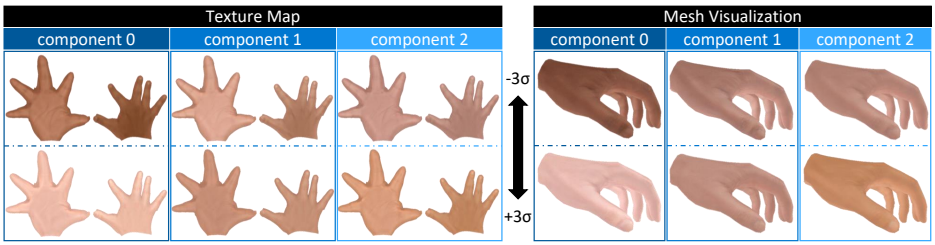


Figure 7.1: HTML is the first parametric hand texture model. The model successfully captures appearance variations from different gender, age, and ethnicity.

et al., 2019; Zhang et al., 2019) as an approximation of appearance. One approach to obtain a personalized hand texture is to project the tracked geometry to the RGB image and copy the observed color to the texture map (La Gorce et al., 2011). However, only a partial appearance of the observed hand parts can be recovered with this method and tracking errors can lead to unnatural appearances. In addition, without explicit lighting estimation, lighting effects will be baked into the results of these projection-based methods.

To address this gap, this chapter present *HTML*, the first data-driven parametric **H**and **T**exture **M**ode**L** (see Figure 7.1). A large variety of hands were captured and the scans aligned in order to enable principal component analysis (PCA) and build a textured parametric hand model. PCA compresses the variations of natural hand appearances to a low dimensional appearance basis, thus enabling a more robust appearance fitting. The proposed model can additionally produce plausible appearance of the entire hand from fitting to partial observations from a single RGB image. The main contributions can be summarized as follows:

- The chapter introduce a novel parametric model of hand texture, HTML, that is made publicly available. The model is based on a dataset of high-resolution hand scans of 51 subjects with variety in gender, age, and ethnicity.
- The scans are registered to the popular MANO hand model (Romero et al., 2017) in order to create a statistical hand appearance model that is also compatible with it.
- The new parametric texture model is demonstrated to enable personalization of 3D hand mesh from a single RGB image of the user's hand in an optimization approach.
- A proof-of-concept neural network layer is presented that combines the MANO model with the proposed texture model in an analysis-by-synthesis fashion. This enables a self-supervised photometric

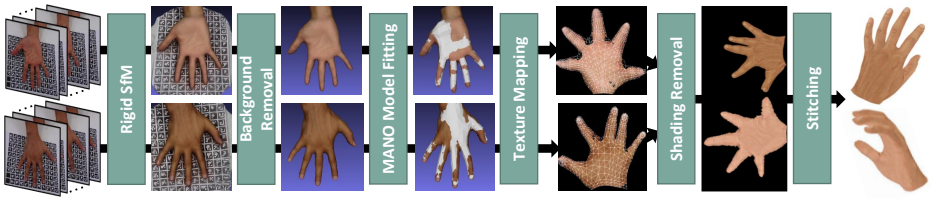


Figure 7.2: **Overview of the hand texture acquisition pipeline.** Rigid structure from motion (SfM) is ran on a set of input images to obtain a scanned mesh for palm and backside of the hand, respectively. After removing background vertices, the MANO template mesh is fitted to extract the texture from the scan. The lighting effects are removed and the front and back textures are seamlessly stitched together, resulting in a complete texture for the captured hand (visualized on the 3D hand mesh from 2 virtual views on the right).

loss that is used to train a method that jointly recovers the hand pose, shape and appearance.

7.2 TEXTURED PARAMETRIC HAND MODEL

The hand texture acquisition pipeline used is summarized in Figure 7.2. First, two image sequences are recorded observing the palm side and the back side of the hand, respectively. Subsequently, rigid structure from motion (SfM) (Bailer et al., 2012; Schönberger and Frahm, 2016) is used to obtain a 3D reconstruction of the observed hand side (Section 7.2.1). Next, the scene background is removed, and both (partial) hand scans are registered to the MANO model Romero et al., 2017 based on nonlinear optimization. Afterwards, the texture of the partial hand scans is mapped to the registered mesh. The shading effects are then removed from the textures and the two partial scans are stitched to obtain a complete hand texture (Section 7.2.2). The parametric texture model in subsequently generated using PCA (Section 7.2.3).

7.2.1 *Data Acquisition*

In total, data from 51 subjects with varying gender, age, and ethnicity are captured (see Figure 7.3). To minimize hand motion during scanning, the palm side and backside of the hand are recorded separately, so that the subjects can rest their hand on a flat surface. As such, for each subject four scans are obtained, i. e. back and palm sides for both left and right hands. The scanning takes ∼90 seconds for one hand side, so that the total scanning time of ∼6 minutes is required per person.

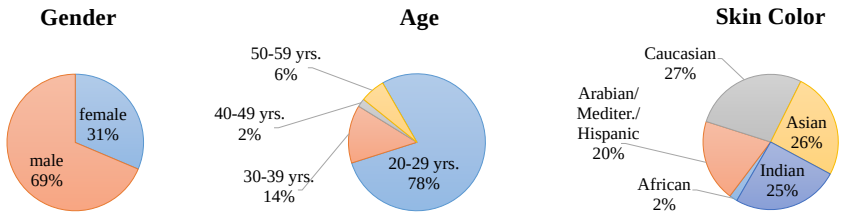


Figure 7.3: Distribution of age, gender, and skin color for the 51 captured subjects. The Goldman world classification scale (Shiffman et al., 2007) is used for classifying skin color.

To obtain 3D hand scans, SONY's 3DCreator App is used. The 3D reconstruction pipeline includes three stages, i. e. initial anchor point extraction, simultaneous localization and mapping (SLAM) with sparse points (Klein and Murray, 2007), and online dense 3D reconstruction (sculpting) (Sony Corporation, 2018). The output is a textured high-resolution surface mesh (of one hand side as well as the background), which contains $\sim 6.2k$ vertices and $\sim 11k$ triangles in the hand area on average. By design, the proposed hand texture model is built for the right hand. For model creation, the left hand meshes are mirrored, so that a total of 102 "right" hands are used for modeling. Note that by mirroring, the texture model of "right" hand can also be used for the left hand. In the following, this technical detail will be abstracted away and the texture modeling approach is described for a single hand.

7.2.2 *Data Canonicalization*

To learn the texture variations in a data-driven manner it is crucial that the acquired 3D scans are brought into a common representation. Due to the popularity and the wide use of the MANO model of hand geometry, the hand texture is built in MANO space. This has the advantage that existing hand reconstruction and tracking frameworks that are based on MANO, such as Boukhayma et al. (2019), Hasson et al. (2019), and Mueller et al. (2019), can be directly extended to also incorporate hand texture. It should be noted that the proposed texture model can also be used with other hand meshes by defining the respective UV mapping. The data canonicalization process comprises of several consecutive stages, i. e. *background removal*, *MANO model fitting*, *texture mapping*, *shading removal*, and *seamless stitching*, which are described next.

Background Removal: For each hand two textured meshes are reconstructed, one that shows the hand palm-down on a flat surface, and one that shows the hand palm-up on a flat surface (cf. Section. 7.2.1). In both

cases, the background, i. e. the flat surface that the hand is resting on, is also reconstructed as part of the mesh. Hence, in order to remove the background, a robust plane fitting based on RANSAC (Fischler and Bolles, 1981) is performed, where a plane is fitted to the flat background surface. To this end, 100 random configurations of three vertices are sampled so that a plane can be fitted. Any point that has a distance to the fitted plane that is smaller than the median edge length of the scanned mesh is considered as an inlier, and their number is then counted. Eventually, the plane that leads to the largest inlier count is considered the background plane. This approach is found to be robust empirically and is able to reliably identify the flat surface in all cases. Eventually, a combination of distance-based and color-based thresholding is used to discard background vertices in the scanned mesh. In particular, a vertex is discarded if its distance from the background plane is less than 1cm and the difference between the red and green channel of the vertex color is smaller than 30 (RGB $\in [0, 255]^3$). This yields better preservation of hand vertices that are close to the background plane.

MANO Model Fitting: Subsequently, the MANO hand model is fitted to each of the front and back filtered hand scan mesh (*i.e.,* the one without background). To this end, the MANO shape and pose parameters are fitted based on the hand tracking approach of Mueller et al. (2019). The approach uses a Gauss-Newton optimization scheme that makes use of additional information based on trained machine learning predictors (*e.g.,* for correspondence estimation). Since their method was developed for 3D reconstruction and tracking of hands in *depth images*, synthetic depth images are rendered from the partial hand scan meshes. Note that the approach of Mueller et al. (2019) was partially trained on synthetic depth images and thus it is able to produce sufficiently good fits of the MANO geometry to the rendered data.

However, since the MANO model is relatively coarse (778 vertices), and more importantly, it has a limited expressivity of hand shape (it only spans the variations of their training set of 31 subjects), some misalignments still remained. To also allow for deformations outside the shape space of the MANO model, a complementary non-rigid refinement of the previously fitted MANO mesh is used to the hand scan. To this end, a variant of non-rigid *iterative closet points* (ICP) (Besl and McKay, 1992) is used that optimizes for individual vertex displacements that further refine the template, which in this case is the fitted MANO model. As the objective function, 3D point-to-point and point-to-plane distances are used together with a spatial smoothness regularizer (Habermann et al., 2019). An accurate alignment is especially important at salient points, like fingertips, to ensure high perceptual quality. Hence, prior correspon-

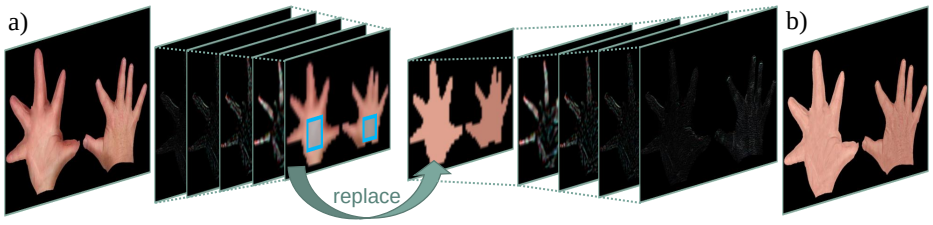


Figure 7.4: **Shading removal.** (a) Original texture and its Laplacian pyramid decomposition. (b) Shading is removed by modifying the deepest level.

dences for the fingertips and the wrist are added to the non-rigid ICP fitting. These correspondences are automatically obtained in the input scanned mesh using OpenPose (Simon et al., 2017). The influence of the prior correspondences is shown in the evaluation (see Section 7.4.1).

Texture Mapping: After having obtained an accurate alignment of the hand template, i. e. the fitted MANO model plus non-rigid deformation for refinement, to the textured high-resolution hand scan, the scan texture is transferred to a texture map. To this end, UV coordinates are manually defined for the MANO model template by unwrapping the mesh to a plane (see texture mapping step in Figure 7.2). Each vertex is projected in the high-resolution hand scan to the closest point on the surface of the fitted MANO hand template. Using the barycentric coordinates of this projected point together with the UV coordinates of the template mesh, the color is transferred to the texture map. After performing this procedure for all vertices of the high-resolution hand scan, there can still be some texels (pixels in the texture map) that are not set (this is found to be about 6.5% of the hand interior). To deal with that, holes are filled based on inpainting with neighboring texels.

Shading Removal: The scans have only low-frequency shading since the environment lighting is carefully controlled during scanning. Thus, the assumptions of having a mostly Lambertian surface and no casted shadows can be made. Since the smooth shading effects have low frequency (see Figure 7.4a), they can be separated and removed using a frequency-based method like the Laplacian image pyramid. To this end, a Laplacian pyramid is built with five levels from the texture map that was obtained in the previous step. It was observed that the deepest level separates the (almost) constant skin color as well as the smooth shading from the texture details that are kept on earlier levels of the pyramid. This deepest level is replaced with a constant skin color for palm and back side, respectively, effectively removing the smooth shading. This constant skin color is obtained by averaging in the well-lit area (see blue

rectangles in Figure 7.4). Note how the texture details from higher levels are preserved in the modified texture map (see Figure 7.4b).

Seamless Texture Stitching: Since so far this texture mapping is performed both for the palm-up and palm-down facing meshes, the partial texture maps are blended in this step to obtain a complete texture map of the hand. To this end, a recent gradient-domain texture stitching approach is used that directly operates in the texture atlas domain while preserving continuity induced by the 3D mesh topology across atlas chart boundaries (Prada et al., 2018).

7.2.3 *Texture Model Creation*

Let $\{T_i\}_{i=1}^{n}$ be the collection of 2D texture maps that is obtained after data canonicalization as described in Section 7.2.2. In order to create a parametric texture model PCA is employed. Each T_i is vectorized to obtain $t_i \in \mathbb{R}^{618,990}$ that stacks the red, green and blue channels of all hand texels. PCA first computes the data covariance matrix

$$C = \frac{1}{n-1} \sum_{i=1}^{n} (t_i - \bar{t}) \, (t_i - \bar{t})^{\top} , \tag{7.1}$$

for $\bar{t} = \frac{1}{n} \sum_{i=1}^{n} t_i$ being the average texture. Subsequently, eigenvalue decomposition of $C = \Phi \Lambda \Phi^{T}$ is used to obtain the principal components Φ and the diagonal matrix of eigenvalues Λ. With that the parametric texture model is obtained for the parameter vector $\alpha \in \mathbb{R}^{k}, k = 101$ as

$$t(\alpha) = \bar{t} + \Phi \, \alpha \, . \tag{7.2}$$

7.3 APPLICATIONS

To demonstrate possible use cases of the proposed parametric hand appearance model, two applications are presented. The model is first used in an optimization framework for 3D hand personalization from single monocular RGB image. Subsequently, the model is used as a neural network layer to enable a self-supervised photometric loss.

7.3.1 *3D Hand Personalization from a Single Image*

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

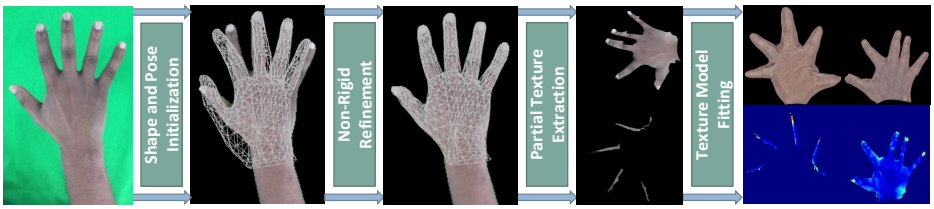


Figure 7.5: **3D hand personalization from a single image.** Starting from a single RGB input image (left), the mesh is first initialized using the method by Boukhayma et al. (2019). Next, the fit is refined non-rigidly and the partial hand texture is extracted. By fitting the proposed parametric texture model to the input texture, a complete texture is obtained (right).

(3) partial texture extraction, and (4) estimation of appearance parameters of the model.

Shape and Pose Initialization: The method of Boukhayma et al. (2019) is used to obtain an initial pose and shape estimate of the MANO template mesh from a single RGB image. As discussed before, the MANO shape space is not always expressive enough to perfectly fit the user's hand shape. In addition, the results from the method by Boukhayma et al. (2019) do not yield sufficiently accurate reprojection of the mesh onto the image plane as shown in Figure 7.5 (second from the left). Hence, this initial mesh is further refined.

Non-Rigid Refinement of the Initial Mesh: The initial mesh estimate is non-rigidly refined to better fit the hand silhouette in the image. Therefore, the 3D displacement of each vertex is optimized using ICP constraints on the boundary vertices. The set of boundary vertices of the hand mesh is defined as $\bar{\mathcal{V}} \subset \mathcal{V}$, i. e. the set of vertices on the silhouette. Let $\Pi : \mathbb{R}^3 \rightarrow \Omega$ be the camera projection converting from 3D world coordinates to 2D pixel locations. For each boundary vertex $\bar{\boldsymbol{v}}_i$, the closest hand silhouette pixel $\bar{\boldsymbol{p}}_i$ in the image domain Ω is found with

$$\bar{\boldsymbol{p}}_i = \underset{\boldsymbol{p} \in \Omega}{\operatorname{argmin}} \; ||\Pi(\bar{\boldsymbol{v}}_i) - \boldsymbol{p}||_2 \quad \text{s.t.} \quad n(\boldsymbol{p})^\top \Pi(n(\bar{\boldsymbol{v}}_i)) > \eta \,. \tag{7.3}$$

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

$$\bar{\boldsymbol{c}}_i = \begin{cases} \bar{\boldsymbol{p}}_i, & \text{if } ||\Pi(\bar{\boldsymbol{v}}_i) - \bar{\boldsymbol{p}}_i||_2 < \delta \\ \emptyset, & \text{otherwise} \end{cases} \,. \tag{7.4}$$

The refined 3D vertex positions is then found using the computed correspondences in the following objective function:

$$E(\mathcal{V}) = \frac{1}{|\bar{\mathcal{V}}|} \sum_{\bar{\boldsymbol{v}}_i \in \bar{\mathcal{V}}} ||\Pi(\bar{\boldsymbol{v}}_i) - \bar{\boldsymbol{p}}_i||_2^2 + \sum_{\boldsymbol{v}_j \in \mathcal{V}} \sum_{\boldsymbol{v}_k \in \mathcal{N}_j} \frac{1}{|\mathcal{N}_j|} ||(\boldsymbol{v}_j - \boldsymbol{v}_k) - (\boldsymbol{v}_j^0 - \boldsymbol{v}_k^0)||_2^2, \tag{7.5}$$

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

Partial Texture Extraction: For each fully visible triangle, i. e. when all its 3 vertices are visible, the color is first extracted from the input image and is then copied to the texture map. This yields a partial texture map where usually at most half the texels have a value assigned and all other texels are set to $\emptyset$. The vectorized target texture map t^{trgt} is then obtained with the same procedure as in model creation (see Section 7.2.3).

Estimation of Appearance Parameters: Subsequently, the appearance parameters of the model that best fit the user's hand are found by solving the least-squares problem with Tikhonov regularization:

$$\underset{\alpha \in \mathbb{R}^k}{\operatorname{argmin}} \sum_{t_i^{\text{trgt}} \neq \emptyset} (t_i^{\text{trgt}} - t(\alpha)_i)^2 + w_{\text{reg}} ||\alpha||_2^2 \,. \tag{7.6}$$

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

7.3.2 *Self-Supervised Photometric Loss*

Previous works have trained neural networks to regress joint positions or MANO model parameters from RGB images (Hasson et al., 2019; Zimmermann and Brox, 2017). The most common loss is the Euclidean distance between the regressed and ground truth joint positions. Some works have also explored a silhouette loss between the mesh and the hand region in the image (Boukhayma et al., 2019). The proposed HTML enables the use of a *self-supervised* photometric loss, which complements the existing fully supervised losses. With that, when training a network to predict shape and pose with such an approach, a hand texture estimate is additionally obtained. To this end, a *textured hand model layer* is introduced, which is explained now.

Textured Hand Model Layer: Given a pair of MANO shape and pose parameters (β, θ), as well as the texture parameters α, the proposed model

layer computes the textured 3D hand mesh $\mathcal{M}(\beta, \theta, \alpha)$. An image of this mesh is then rendered using a scaled orthographic projection. As such, this rendered image can directly be compared to the input image $\mathcal{I}$ using a photometric loss in an analysis-by-synthesis manner. The photometric loss is formulated as

$$\mathcal{L}_{\text{photo}}(\beta, \theta, \alpha) = \frac{1}{|\Gamma|} \sum_{(u,v)\in\Gamma} ||\text{render}(\mathcal{M}(\beta, \theta, \alpha))(u, v) - \mathcal{I}(u, v)||_2 \,, \quad (7.7)$$

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

Network Training: A residual network is trained with the architecture of ResNet-34 (He et al., 2016) to regress the shape β, pose θ, and texture parameters α from a given input image. In addition to the self-supervised photometric loss, losses are applied on 2D joint positions, 3D joint positions, and L2-regularizers are applied on the magnitude of the shape, pose, and texture parameters. The network is trained in PyTorch (Paszke et al., 2019b), using the differentiable renderer provided in PyTorch3D (Ravi et al., 2020). For training, illumination from a single fixed point source is assumed. The joint estimation of additional lighting and material properties is left for future work.

7.4 EXPERIMENTS

In this section, the proposed parametric hand texture model is evaluated to explore the effects of different design choices in the texture acquisition pipeline, and to present results of two example applications.

7.4.1 *Texture Model Evaluation*

Compactness: Figure 7.6 (left) shows the compactness of the proposed texture model. The plot describes how much the explained variance in the training dataset increases with the number of used principal components. The first few components already explain a significant amount of variation since they account for more global changes in the texture, e. g. skin tone. However, adding more components continuously increases the explained variance.

Generalization: For evaluating generalization, a leave-one-subject-out protocol is used. The data of one subject is removed, i. e. the two texture samples from left and right hand, and the PCA model is rebuilt. Then, the left-out textures are reconstructed using the built model and the

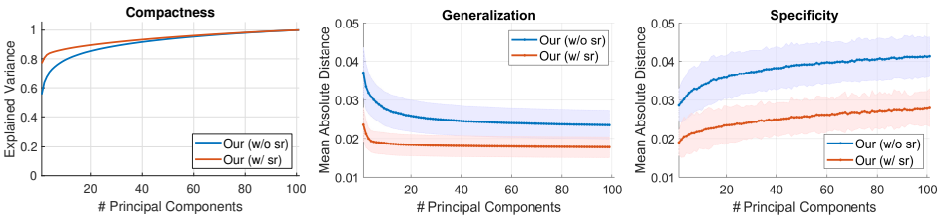


Figure 7.6: Evaluation of compactness, generalization and specificity. Using shading removal ("w/ sr") is substantially better than not using it ("w/o sr").

reconstruction error is measured as the mean absolute distance (MAD) between the vectorized textures. As shown in Figure 7.6 (middle), the reconstruction error decreases monotonically for an increasing number of components for both of the two models.

Specificity: The specificity is also reported, which quantifies the similarity between random samples from the model and the training data. To this end, a texture instance is first sampled from the proposed model using a multivariate standard Normal distribution. Then, the nearest texture in the training dataset is found in terms of the MAD. This procedure is repeated 200 times, and the statistics of the MAD is reported in Figure 7.6.

Influence of Shading Removal: Figure 7.6 also shows compactness, generalization, and specificity for a version of the texture model that was built without shading removal ("w/o sr"). It can be seen that the version without shading removal performs worse compared to the one with shading removal ("w/ sr") in all metrics. When the lighting effects are not removed, they increase the variance in the training dataset. Hence, more principal components are necessary to explain variation and the reconstruction of unseen test samples has a higher error. In the Appendix A.14, it is shown visually that the principal components for the model without shading removal have to account for strong lighting variation.

Influence of Prior Correspondences: To ensure a good alignment of the hand template mesh and the scanned mesh, as explained in Section 7.2.2,

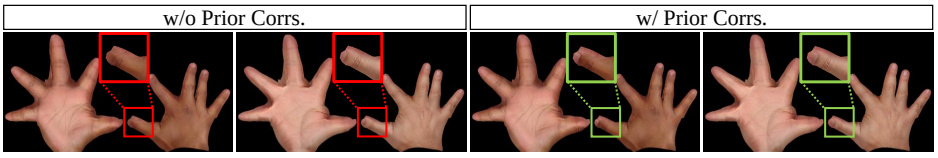


Figure 7.7: Using non-rigid ICP-based refinement with prior correspondences for fingertips and the wrist improves the alignment of the hand template mesh to the scanned mesh, yielding better textures (right). (Textures shown before shading removal.)

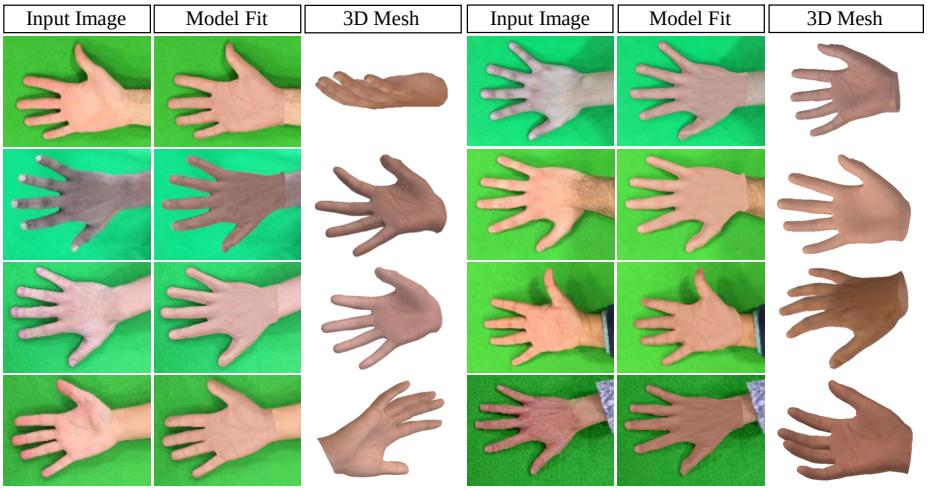


Figure 7.8: Hand personlization from a single RGB image for different subjects.

for the non-rigid ICP-based refinement step in the the model building stage the prior correspondences for the fingertips and the wrist is used. Figure 7.7 compares the textures obtained by running the non-rigid ICP fitting with and without them. Especially for the thumb, the tip is often not well-aligned, resulting in a missing finger nail in the texture. Using explicit prior correspondences alleviates this issue.

7.4.2 *Application Results: 3D Hand Personalization*

Here, the results for obtaining a personalized 3D hand model from a single RGB image are shown (see Section 7.3.1). As previously discussed, since the output meshes of state-of-the-art regression approaches (Boukhayma et al., 2019) do not have a low reprojection error, non-rigid refinement based on silhouettes is used. To simplify segmentation for the example application, the images of the users are captured in front of a green screen. In future work, this could be replaced by a dedicated hand segmentation method. Figure 7.8 shows hand model fits and complete recovered textures from a single RGB image for several subjects. Since a low-dimensional PCA space is used to model hand texture variation, a plausible and complete texture is robustly estimated from noisy or partially corrupted input (see Figure 7.9). In contrast, a texture that is directly obtained by projecting the input image onto a mesh obtained by the method of Boukhayma et al. (2019) contains large misalignments and a significant amount of background pixels, and thus is severely corrupted.

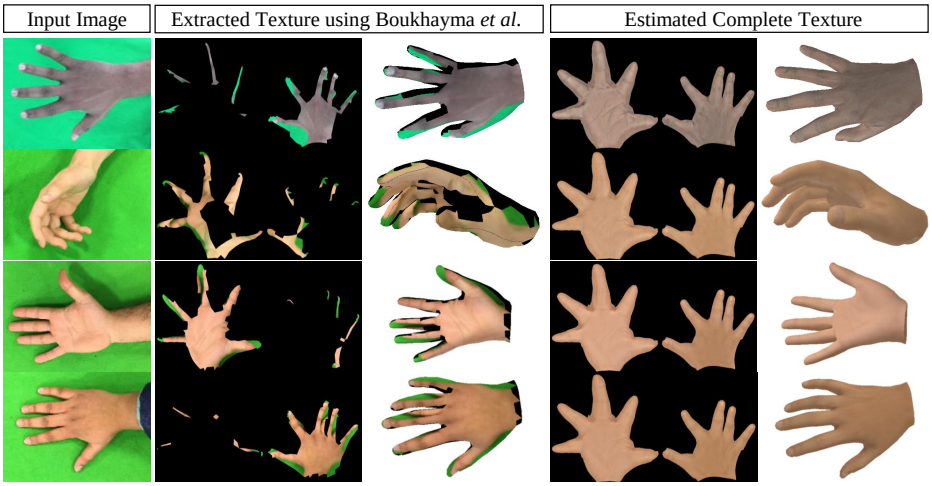


Figure 7.9: Fitting to noisy or corrupted input textures is robust and yields a realistic and complete texture estimate due to the low-dimensional PCA space of the proposed model.

7.4.3 *Application Results: Photometric Neural Network Loss*

The proposed self-supervised photometric loss (see Section 7.3.2) enables to not only obtain shape and pose estimates as in previous work, but in addition to also estimate hand appearance. To demonstrate this a network is trained on the recently proposed FreiHAND dataset (Zimmermann et al., 2019). For details of the experimental setup, please see Appendix A.13. In Figure 7.10, hand model fits predicted by a neural network trained is shown with and without the proposed photometric loss (cf. Section 7.3.2). It should be noted that that the pose and shape prediction with the photometric loss are quantitatively similar to the predictions without (the mean aligned vertex errors (MAVE) are 1.10 cm vs 1.14 cm respectively, and mean aligned keypoint errors (MAKE) are 1.11 cm vs 1.14 cm respectively). In addition, these results are comparable to the current state of the art (Zimmermann et al., 2019) with a MAVE of 1.09 cm and MAKE of 1.10 cm. It should be stressed that the proposed method with the photometric loss additionally infers a high resolution, detailed texture of the full hand, which the other methods do not.

7.5 LIMITATIONS AND DISCUSSION

The experiments have shown that HTML can be used to recover personalized 3D hand shape and appearance. Although the proposed model provides detailed texture, the underlying geometry of the MANO mesh is

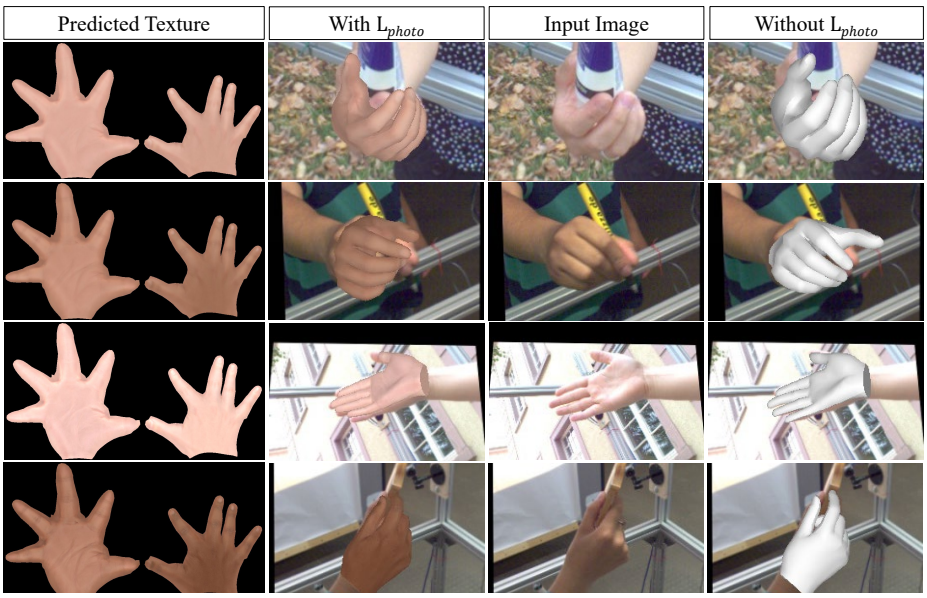


Figure 7.10: Examples of pose and texture predictions from a neural network trained using the proposed photometric loss L_{photo} enabled by the parametric hand texture model.

coarse (778 vertices). This could be improved by using a higher-resolution mesh and extending the MANO shape space with more detailed geometry. Non-linear models, *e.g.,* an autoencoder neural network, can be explored for capturing variations that a linear PCA model cannot. As hand appearance varies during articulation, modeling pose-dependent texture changes can increase the realism. This would need a more complicated capture and registration setup and a significantly larger dataset to capture the whole pose space and diverse users. In terms of applications, estimating lighting in addition to or jointly with the texture parameters can better reconstruct input observations. Correctly modeling lighting for hands, where shadow casting often occurs, is a challenge that would need to be addressed. Other applications of the proposed model, such as exploring how self-supervision can alleviate the need for annotations or improve estimation of pose, can be directions for future research.

7.6 CONCLUSION

This chapter introduced HTML — the first parametric texture model of hands. The model is based on data that captures 102 hands of people with varying gender, age and ethnicity. For model creation, a data canonicalization pipeline is presented that entails background removal, geometric model fitting, texture mapping, and shading removal. More-

over, the experiments demonstrated that the proposed model enables two highly relevant applications: 3D hand personalization from a single RGB image, and learning texture estimation using a self-supervised loss.

8 CONCLUSION

This thesis demonstrated a variety of strategies to integrate model-based priors for 3D hand reconstruction from monocular input.

Starting with the setting of single hand reconstruction using input from depth cameras, Chapter 4 made use of a volumetric prior based on a Sum-of-Gaussians hand model for self-supervision during training. This was shown to reduce annotation requirements and to increase robustness to data biases. Then in the next two chapters, the thesis tackled the even more challenging task of reconstructing two hands during interaction using only monocular RGB input. Chapter 5 developed the first method to deal with this severely under-constrained problem. By relying on a model-fitting step for incorporating additional pose, temporal, and physics-based priors to disambiguate the task, the method reconstructs two physically correct hand meshes that match the image-level hand features extracted by a neural network. The training was enabled by using a novel mix of physically consistent synthetic data and noisy real data bootstrapped from depth-based reconstructions. Chapter 6 further addressed the challenging ambiguities by explicitly modeling the distribution of all image-consistent physically plausible poses. As the first work to systematically quantify pose ambiguity in this setting, it was shown that established metrics defined on point estimates do not reflect the quality of reconstruction, and a distribution metric, supported by a novel dataset with multiple annotations per image, was proposed instead.

Finally, in addition to the pose and geometry reconstruction seen thus far, Chapter 7 presented the first parametric hand appearance model in the literature. This re-parameterized space is shown to regularize hand texture for appearance reconstruction in learning and model-fitting frameworks. This enabled the recovery of detailed texture of the full hand from partial, low resolution, RGB input for hand appearance personalization.

8.1 INSIGHTS

Beyond the main contributions presented in each chapter, several broader insights are drawn from the common themes present in the challenges and how they are tackled.

Resolving Ambiguity of Different Types. As 3D reconstruction from monocular input is severely under-constrained, all chapters proposed ways to deal with the missing information that arises from various ambiguities. The main assumption used for disambiguation is that the object of interest is a hand. The ambiguities that can be constrained by this assumption are considered *explainable ambiguities*, and the remaining unconstrained ambiguities are considered *residual ambiguities*.

As demonstrated in the thesis, model-based priors, such as the Sum-of-Gaussians, MANO, and HTML hand models and their associated statistical regularizers, can narrow down the explainable ambiguities to achieve more accurate reconstruction despite the missing information. However, the residual ambiguities remaining impose a limit on the reconstruction quality. For example, Chapter 6 has shown that the relative position between two hands and the articulation of the occluded hand can drastically vary while projecting to form the same image. In absence of additional assumptions such as user intent, temporal information, or additional view points, disambiguation is fundamentally impossible. In such a case, the recoverable information can be summarized by finding the entire range of valid solutions.

It is important to distinguish between these two ambiguities during modeling, as the type determines whether it should be minimized (explainable) or quantified (residual). If the residual ambiguity is too high for a given application, additional assumptions or observations need to be incorporated into the problem.

Learn to Not Only Answer but to Explain. Most existing learning-based hand pose estimation methods are trained only to reproduce the paired annotation when given an image. As discussed in Chapter 4, this leads to the network learning a direct image-to-pose mapping that overfits to annotation biases.

Instead of simply penalizing the wrong answers, overfitting to such bias can be prevented by ensuring the reconstructions explain the inputs. The self-supervised loss in Chapter 4 imposes this constraint as a loss, which regularizes against annotation bias. Another approach, used in Chapter 5, is to learn to regress image-level features instead. This can be seen as forcing the network to present explicit reasoning, which is then robustly integrated along with additional heuristics and priors to infer a plausible pose through model-fitting. Similarly, Chapter 6 emphasized that direct pose regression is an ambiguous one-to-many mapping that cannot be supervised or evaluated with a single given annotation as answer. Instead, a probabilistic network should be used where samples from the estimated distribution are supervised to explain the input. The resulting estimates can be quantitatively evaluated using the suggested

maximum mean discrepancy metric. Together, these allow for a more complete characterization of the pose information available in the image.

In summary, this thesis demonstrated that it is often not enough to supervise a network by just providing it the answer in ambiguous scenarios. The innovative strategies used in the chapters to explain the input image instead allow the developed methods to be more robust to annotation bias and network prediction errors, while also enable it to quantify image ambiguity.

8.2 OUTLOOK

This thesis presented several novel methods that advanced the state of the art by addressing the extreme ambiguities in monocular 3D reconstruction of hands. However, many challenges still need to be addressed before vision-based hand reconstruction can be relied upon as a natural interface of human-computer interactions. Some of these exciting open research questions are discussed in more details below.

Intent-based Prior. Given the large amount of residual ambiguity remaining in monocular 3D reconstruction, as shown in in Chapter 6, new assumptions must be introduced to resolve it. One interesting method is to use the context of user intent to restrict the reconstruction to a smaller application-dependent pose space. For example, if the AR/VR simulation equips the user's palm with interactive buttons, the intent to press the buttons can be exploited by imposing additional contact constraints to reduce depth ambiguity. Such intent driven priors could be extended to sign language capturing or other communicative gesturing where the application limits the poses of interest. This interplay between a well-defined downstream task and the 3D reconstruction could enable more accurate results at vital points of interaction, and default back to plausible reconstructions when minor discrepancies would not be noticed.

In such a scenario, temporal information can also be better exploited for learning-based modeling. As the hand is no longer tasked to make arbitrary motion or to transition between unrelated peak poses as in current datasets (e.g Moon et al. (2020)), stronger temporal correlation would exist in underlying data which can then be exploited for disambiguation.

Two-Hand Interaction with Object. Extending the two-hand interaction scenario addressed by this thesis, jointly reconstructing two hands while they interact with an unknown object would have additional applications. Although there exists one work (Kwon et al., 2021) tackling a similar problem, it assumes a known object mesh rather than reconstructing both jointly. Not only is there more occlusion and appearance variations for

hand reconstruction in this challenging setting, reconstruction of general object geometry is under-constrained since the object class is not known in advance. This is further aggravated when the object can also deform non-rigidly.

However, tackling the reconstruction of both object and hands jointly has its advantages. Unlike hands, objects under interaction are usually free-standing so that its forces can be analyzed. The sum of forces must be able to counteract gravity if the configuration is assumed to be stable. Analogous to the single hand-object interaction work by Hasson et al. (2019), this stability assumption can reduce depth ambiguity by necessitating contact points at fixed locations. The challenge with two hands then is to model the inter-dependencies between the contact points of both hands in terms of stability criteria as more stable configurations exist.

Similarly, how a user holds an object and how it reacts to being manipulated provides cues about its physical attributes, e.g. its weight and Yong's modulus, which in turn provides hints about forces exerted by the hand. Recovering these physical quantities from images alone would be difficult, but similar efforts using physics-based constraints to jointly estimate physical constants and human pose shows promising results in recovering plausible solutions (Bieler et al., 2019; Dabral et al., 2021). Correctly capturing not just geometry, but also the forces involved and material properties of the object would be vital for interacting with AR/VR applications driven by physics simulations.

A

APPENDIX

A.1 ENERGY TERM WEIGHTS (CHAPTER 4)

To find the weights λ for use as a leaning objective, a implementation of the losses as a traditional model-based tracker on single images was used as a proxy. The weights are tuned so that the different terms evaluates to similar values in this setting, and it was observed that these weights work for training as well. The following weights were used for all experiments: $\lambda_{\text{dissim}} = \lambda_{\text{collision}} = 0.6$, $\lambda_{\text{bone}} = 10^{-4}$, $\lambda_{\text{lim}} = 0.5$, $\lambda_{\text{joint}} = 8 \cdot 10^{-6}$.

A.2 DETAILS ON HANDID DATASET (CHAPTER 4)

The proposed HANDID Dataset contains a total of 3,601 frames from 7 subjects that were acquired with the Intel SR300 sensor. The subjects were instructed to perform simple abduction, adduction, and flexion gestures while the camera recorded in a third-person view. The annotators were told to select 6 pixels for each depth image that correspond to the fingertip and wrist keypoints. In case the keypoints are occluded, the annotators were asked to estimate plausible 2D locations based on previous and future frames in the image sequence and to mark them as occluded. With that, for visible keypoints 3D annotations were obtained using the depth values of the pixels, and for occluded keypoints 2D annotations were obtained. See Figure A.1 for examples.

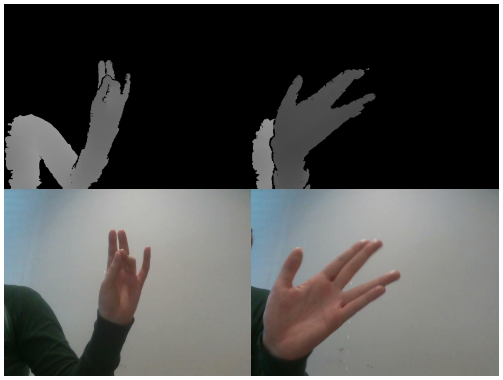


Figure A.1: **HandID Dataset:** Two examples of the depth images captured for the HANDID dataset are shown. The colored images are included for better visualization.

A.3 ANNOTATION TRANSFER FROM DEPTH TO RGB (CHAPTER 5)

Here, the procedure to transfer annotations from the depth image to the RGB image, where both were acquired with a calibrated RGB-D

camera (cf. Section 5.5) are described. To this end, based on camera calibration parameters of the depth and RGB cameras, 3D points are first recovered from the depth image and then reprojected onto the RGB image. However, holes exist in the reprojected image due to an offset between the sensors (baseline), and due to object boundary artifacts of the structured-light depth sensor. To prevent the neural network from learning this, a foreground mask is employed in the respective loss. The final foreground mask is based on computing the intersection of two masks; a foreground mask obtained from the depth image, and a foreground mask that is obtained from the RGB image. For obtaining the former, the depth image is first thresholded, and subsequently processed by a morphological closing operation to correct for boundary artifacts. The foreground mask of the RGB image is obtained by color thresholding, which is straightforward due to the green-screen setup. By using such a mask, only those regions that can be trusted are used for training the multi-task CNN, while regions with potentially missing annotations are not penalized.

A.4 DATA AUGMENTATION (CHAPTER 5)

The real dataset contains sequences of single-hand tracking scenarios with a green-screen to allow background augmentation. Additionally, a color-based segmentation is performed to separate hand and body. For compositing two-hand scenes, two masked hand images from the same person are chosen at random, and flipped accordingly to obtain a pair of left and right hands. The hands are then independently translated from their initial location. Whenever the two hands are overlapping, the depth channel is taken into account to ensure a plausible occlusion when the hands project onto the same location in the image.

A.5 ENERGY TERM WEIGHTS (CHAPTER 5)

In the following, the weights of energy terms used in Equation 5.3 are provided: $\lambda_{\text{d}} = 0.003$, $\lambda_{\text{sil}} = 0.0045$, $\lambda_{\text{key}} = 0.005$, $\lambda_{\text{intra}} = 0.3$, $\lambda_{\text{inter}} = 0.1$, $\lambda_{\beta} = 0.025$, $\lambda_{\theta} = 0.0375$, $\lambda_{\tau} = 0.3$, $\lambda_{\text{sym}} = 0.5$, $\lambda_{\mathcal{N}} = 4.6 \cdot 10^5$, $\lambda_s = 10^3$, $t_{\theta} = 0.1$, $t_r = 2.3$, $t_c = 0.04$, $t_h = 0.7$, $\mu = 1$.

A.6 AUTOMATIC ERROR RECOVERY (CHAPTER 5)

Due to the severe ambiguities in monocular RGB data, like the concave-convex ambiguity, the optimization of the parameters $\boldsymbol{\nu}$ does not always

yield a correct result. Instead, fingers might get bent in the wrong direction to fit the projection in the input. These poses are difficult to escape using a local optimizer. Hence these unnatural poses are detected by observing the magnitude of $\boldsymbol{\theta}_h, h \in \{\text{left, right}\}$. If this magnitude exceeds the threshold t_r, the pose parameters are reset to the closest pose that is still within the threshold.

A.7 HANDFLOW TRAINING SETTINGS (CHAPTER 6)

The PyTorch framework (v1.9) (Paszke et al., 2019a) is used for implementation. The ResNet-50 (He et al., 2016) backbone used is pre-trained on the InterHand2.6M dataset using the weights from Moon et al. (2020). This ensures that the feature vector to the subsequent normalizing flow network contains relevant features for 3D hand pose estimation when starting to train the complete *HandFlowNet*.

AdamW, a version of Adam (Kingma and Ba, 2015) with Decoupled Weight Decay Regularization (Loshchilov and Hutter, 2019), is used for optimization. Default parameters are used except for a learning rate of 10^{-4} and a weight decay of 10^{-4}.

For loss weights, the following are used: $\lambda_{\mathcal{J}_{3D}} = 10^2$, $\lambda_{\mathcal{J}_{2D}} = 10^{-1}$, $\lambda_{\text{DetMag}} = 10^{-3}$, $\lambda_{\psi} = 1.25 \times 10^{-3}$, $\lambda_{\text{nll}} = 10^{-3}$, $\lambda_{\theta} = 10^{-1}$.

A.8 MULTIHANDS DATASET (CHAPTER 6)

In this section, additional details of the algorithm used to generate plausible annotations for the MultiHands dataset is provided (see Algorithm 1 for an overview.).

Overall, the method perturb the ground-truth pose and checks for the four plausibility criteria to generate new annotation. However, doing so naively (e.g. adding Gaussian noise to the pose space annotations) would result mostly in samples that do not fit the plausibility criteria. The proposed method uses several heuristics to speed up the discovery of plausible poses, which are explained in the following sections.

A.8.1 *Translation Sampling:*

First, translation perturbations to the initial ground truth provided in InterHands2.6M are sampled. The goal is to constrain the range of plausible translation samples so that visible joints are *image consistent*, and the resulting pose is *collision free*.

Data: Initial MANO Pose ψ_0
Result: Additional annotations ψ_i

// Ensure: sampled $t \leftarrow \{t_{right}, t_{left}\}$
// 1. $\Delta P^{2D}(\psi_i(t)) < T$
// 2. $\neg$collision$(\psi_i(t))$

$t \leftarrow$ sample_translation(ψ_0);
$\psi_i \leftarrow$ update_translation(ψ_0, t);

for *N iterations* **do**

  // Get finger with occluded joints:
  // $\forall P_j^{3D} \in \{P^{3D}\}$
  // child(P_j^{3D}) are occluded

  $\{P^{3D}\} \leftarrow$ select_finger(ψ_i);
  $\psi_i' \leftarrow \psi_i$

  for P_j^{3D} *in* $\{P^{3D}\}$ **do**

    // Ensure: sampled Q^{3D}
    // 1. bone(Q^{3D}) = bone(P_j^{3D})
    // 2. Q^{3D} is occluded

    $Q^{3D} \leftarrow$ sample_joint(P_j^{3D});
    $\psi_i' =$ update_pose(ψ_i', Q^{3D});
    $\{P^{3D}\} =$ update_child$(\{P^{3D}\}, \psi_i')$;

  end

  // likelihood from pose PCA
  if *is_plausible*(ψ_i') **then**
    $\psi_i \leftarrow \psi_i'$
  end

end

Algorithm 1: Pseudocode for sampling additional annotations

For this, binary search was used to find the ***image consistent range*** of each hand; the range of depth offsets that limits 2D position change to under 3.5 pixels for visible joints.

The ***collision free range*** is then calculated. This is the range of valid left hand depth translations that avoid collision with the right hand. Collision is detected using sphere proxies obtained from the volumetric Gaussian approximation of the MANO model (Mueller et al., 2019; Wang et al., 2020a).

Given these ranges, the final translation change is obtained by first sampling a global depth offset from the overlapping *image consistent ranges* of both hands. The left hand is then offset from the right by sampling from the overlap between the *collision free range* and *image consistent range* of the left hand.

A.8.2 *Articulation Sampling:*

To find a plausible articulation, note that only occluded joints can change their position and the resulting position must also project on to an occluded pixel. Thus occluded joints are iteratively selected and new occluded positions proposed to cut down on the search space. As articulations are propagated down a kinematic chain, all joints on the same finger are considered together.

Select_finger(ψ_i): For each iteration, a finger to perturb is selected for the pose ψ_i. A finger can be selected if there exists an occluded joint whose child joints are all occluded.

For the selected finger, its joint locations $\{P_j^{3\mathrm{D}}\}$ are updated from the base to the tip starting with the first occluded joint.

Sample_joint$(P_j^{\mathbf{3D}})$: Given the current joint location, a new 3D position needs to be found that both preserves the bone length and results in occlusion. To maintain the bone length, points on a 3D sphere centered at the parent joint with radius equal to the bone length are defined as *bone length consistent points*.

Then points that would allow the joint to become visible are eliminate. This is done by first projecting the sphere on to the image plane to obtain a set of pixels that lie within the projection, and then checking the occlusion status of each pixel based on a depth rendering. The occluded pixels locations are unprojected to form rays, and the intersections between the pixel rays and the sphere becomes the *preliminary joint proposals* $Q^{3\mathrm{D}}$.

Update_pose$(\psi_i', Q^{\mathbf{3D}})$: For each *preliminary joint proposal*, the rotation needed to transform the current joint from the original position to the

Figure A.2: More visualizations of the MultiHands dataset. Note the diversity of 3D poses that can be seen in the novel view.

proposed location is calculated. This rotation update is used to update the current pose parameter ψ_i'.

Update_child$(\{P^{3\mathbf{D}}\}, \psi_i')$: The remaining finger joints $\{P_j^{3\mathrm{D}}\}$ are updated using the pose parameter ψ_i'.

is_plausible(ψ_i'): After all child joints in a finger have been updated, the resulting current pose parameter ψ_i' is checked for anatomically plausibility. This is done by converting the pose rotation parameters to the MANO pose PCA parameters, which enables likelihood estimation under Gaussian assumptions. Hand proposals with low log likelihood (less than -60) are rejected.

To generate a single accepted new plausible annotation, pose perturbation is run for 100 iterations. For MultiHands, 100 new plausible annotations were obtained per image. See Figure A.2 for examples of annotations.

A.9 EVALUATION METRICS (CHAPTER 6)

In this section, additional details of the metrics used to evaluate the proposed method are provided.

A.9.1 *Pose Alignments*

Considering $\hat{P}^{3D} = \mathcal{J}(\psi)$ as the 3D joint positions calculated from the estimated hand parameters ψ, the **Global MPJPE (Global)** metric is defined as

$$\text{MPJPE}_{\text{Global}} = \frac{1}{N}\sum_{i=1}^{N} ||\hat{P}_i^{3D} - P_i^{3D}||_2 \,, \tag{A.1}$$

where N is the total number of annotated joints and P^{3D} are the ground-truth 3D joint positions. Note that this metric is computed *without any alignment*.

For *right root alignment*, the joints of both hands are aligned to the right hand root joint before computing the error. Let

$$\mathcal{R}_{\text{r}}(P_i^{3D}) = P_i^{3D} - P_{\text{right\_root}}^{3D} \,, \tag{A.2}$$

be the function that calculates the joint position relative to the right hand root. Then the **Right-Root-Relative MPJPE (RRR)** metric is defined by

$$\text{MPJPE}_{\text{RRR}} = \frac{1}{N}\sum_{i=1}^{N} ||\mathcal{R}_{\text{r}}(\hat{P}_i^{3D}) - \mathcal{R}_{\text{r}}(P_i^{3D})||_2 \,. \tag{A.3}$$

The error for each hand is also evaluated individually. For this, the joint position relative to the corresponding hand root joint is represented by

$$\mathcal{R}(P_i^{3D}) = P_i^{3D} - \text{root}(P_i^{3D}) \,, \tag{A.4}$$

where $\text{root}(\cdot)$ is a function the returns the right/left root joint position if P_i^{3D} belongs to the right/left hand. Then, the **Root-Relative MPJPE (RR)** is defined by:

$$\text{MPJPE}_{\text{RR}} = \frac{1}{N}\sum_{i=1}^{N} ||\mathcal{R}(\hat{P}_i^{3D}) - \mathcal{R}(P_i^{3D})||_2 \,. \tag{A.5}$$

A.9.2 *Maximum Mean Discrepancy (MMD)*

Given the set of all joint positions of predicted pose samples $\hat{\mathcal{P}}^{3D} = \{\hat{P}_i^{3D}\}_{i=1}^{n}$, and the set of ground truth joint positions $\mathcal{P}^{3D} = \{P_i^{3D}\}_{i=1}^{m}$, the Maximum Mean Discrepancy (MMD) with kernel κ can be estimated with:

$$\begin{aligned}\mathrm{MMD}^2(\hat{\mathcal{P}}^{3D}, \mathcal{P}^{3D}) = &\frac{1}{n(n-1)} \sum_{i \neq j}^{n} \kappa\left(\hat{\mathcal{P}}_i^{3D}, \hat{\mathcal{P}}_j^{3D}\right) \\ &+ \frac{1}{m(m-1)} \sum_{i \neq j}^{m} \kappa\left(\mathcal{P}_i^{3D}, \mathcal{P}_j^{3D}\right) \\ &- \frac{2}{nm} \sum_{j=1}^{m} \sum_{i=1}^{n} \kappa\left(\hat{\mathcal{P}}_i^{3D}, \mathcal{P}_j^{3D}\right)\end{aligned} \tag{A.6}$$

A Gaussian kernel is used for κ and the resulting MMD is averaged across Gaussians kernels with standard deviations ranging between $[1\text{–}100mm]$ and sampled at $1mm$ intervals.

MMD^2_{RR} and MMD^2_{RRR}, are defined analogously to $\mathcal{R}(P_i^{3D})$, $\mathcal{R}(\hat{P}_i^{3D})$ or with $\mathcal{R}_{\mathrm{r}}(P_i^{3D})$, $\mathcal{R}_{\mathrm{r}}(\hat{P}_i^{3D})$ respectively.

A.10 RESULTS: BASELINE DETAILS (CHAPTER 6)

For implementing the baselines, HandFlowNet components are reused as much as possible to ensure fair comparison in terms of network capacity. When using the normalizing flow network as a feed forward network in the baseline, $\mathbf{z} = \mathbf{0}$ was used.

To implement MC-dropout, the existing dropout layers were used (with dropout probability of 0.5) in normalizing flow network during inference time to obtain samples. For the Gaussian baseline, the pose distribution was modeled as Gaussian aleatoric uncertainty $\mathcal{N}(\mu, \Sigma)$ inspired by Kendall and Gal (2017). The normalizing flow network is trained to estimate μ and Σ directly from the extracted image feature $\mathbf{v}$. To implement the VAE baseline, a fully connected layer was added to the image feature extractor to act as the encoder for the image. The normalizing flow network then acts as the decoder to recover the hand pose. Empirically, it was found that latent code size of 256 and KL divergence weight of 4×10^{-4} work best as hyper-parameters.

For the MC-dropout and VAE baselines, $\mathcal{L}_{\mathrm{nll}}$ and $\mathcal{L}_{\mathrm{DetMag}}$ cannot be used in their formulation and are thus omitted. Otherwise, all loss terms are used during training.

A.11 RESULTS: DETERMINISTIC COMPARISONS (CHAPTER 6)

In Section 6.4.3, it is shown that the commonly used MPJPE on a single annotation is not suitable for capturing the uncertainty present in the highly ambiguous task of monocular two-hands reconstruction. However,

Method	Global MPJPE ↓	RRR MPJPE ↓	RR MPJPE ↓
HandFlow	**22.8**	**21.0**	**16.1**
HandFlow (Mode)	51.4	30.7	18.2
InterNet	83.3	29.1	19.4
Fan et al.	81.7	32.1	17.2
InterShape	-	33.7	18.7

Table A.1: The proposed method produces samples that are on-par or better than the state-of-the-art methods. All results are in mm.

a comparison to the current state-of-the-art two-hand pose estimation methods is still performed for reference.

The proposed method is compared to InterNet (Moon et al., 2020), InterShape (Zhang et al., 2021), and Fan et al. (2021). Notice that InterNet and Fan et al. both require an additional network to explicitly estimate the global hand position, and InterShape only estimates relative hand position. In contrast, the proposed method directly outputs global hand position. Additionally, InterShape requires the ground-truth bone lengths to scale their results while the proposed method does not.

Evaluation of Samples. Table A.1 show the comparison on InterHand2.6M using MPJPE in mm. To evaluate whether the predicted distribution well captures the ground truth, the established convention (Wehrbein et al., 2021; Ye and Kim, 2018) to sample 100 poses is followed and the values of the *best sample* according to each metric is reported.

The metrics on just the mode sample is additionally reported to provide a baseline of the proposed method as a traditional deterministic pose estimator. It can be seen that *HandFlowNet* produces samples that are significantly closer to the ground truth, while still being competitive even as a single pose estimator. As such, the proposed method better captures the recoverable 3D information from the input.

A.12 MORE QUALITATIVE RESULTS (CHAPTER 6)

In Figure A.3, more renderings of the individual samples of the predicted distribution are shown. In Figure A.4, the skeleton visualization is used to show the spread of the predicted distribution.

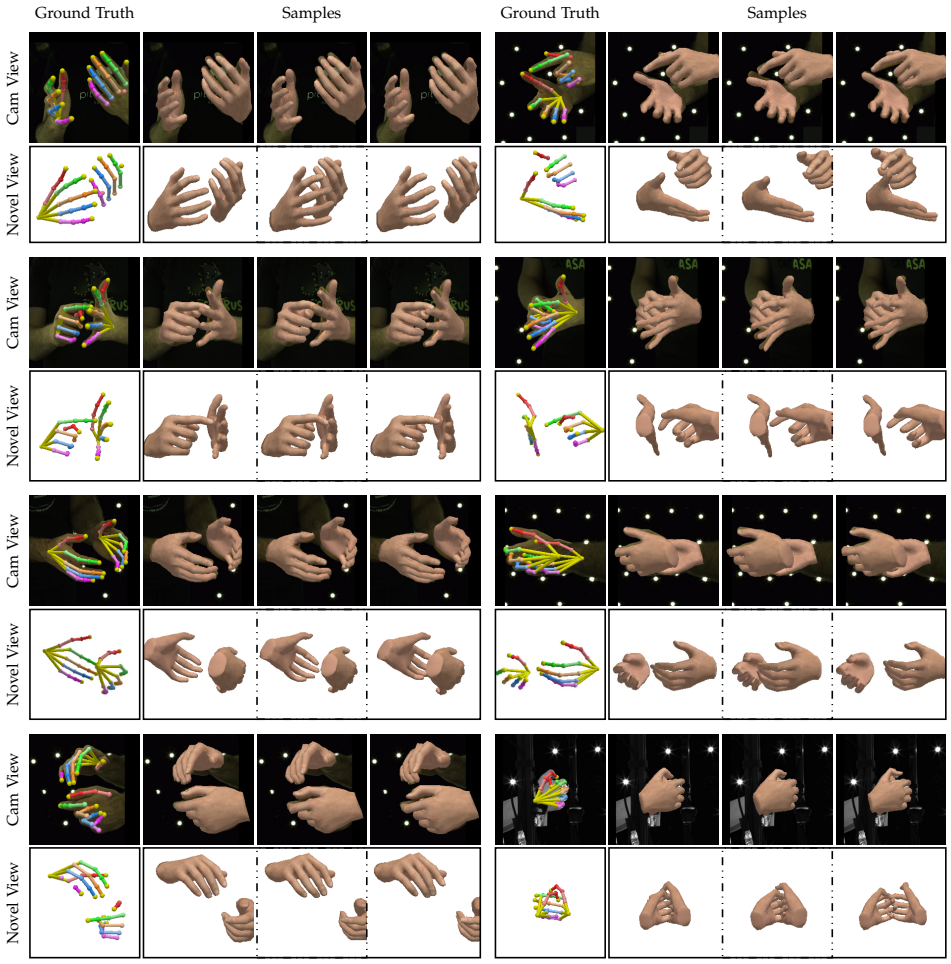


Figure A.3: More individual mesh samples are shown from the camera view and from a novel view. Note that not all joints are annotated in the ground truth. This shows up as missing segments in the skeleton.

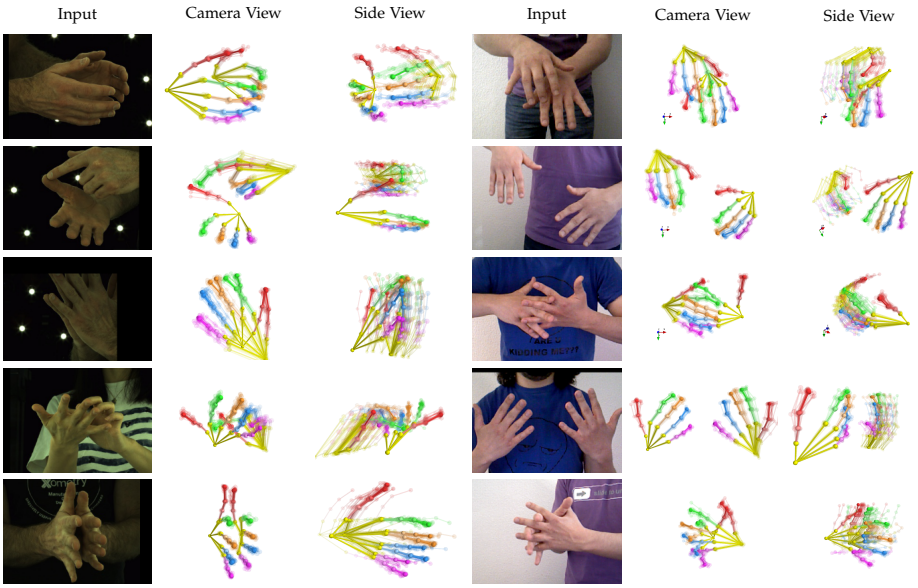


Figure A.4: 30 samples from the estimated distribution are shown, rendered as semi-transparent skeletons, superimposed on a single image. These samples are aligned to the root joint of one hand and the mode of the distribution is made opaque for ease of visualization. Results are shown from both the InterHand2.6M (left) and the Tzionas datasets (right).

	KP Error	KP AUC	Mesh Error	Mesh AUC	F@5mm	F@15mm
Zimmermann et al. (2019)	1.10	0.783	1.09	0.783	0.516	0.934
Boukhayma et al. (2019)	3.50	0.351	1.32	0.738	0.427	0.895
Hasson et al. (2019)	1.33	0.737	1.33	0.736	0.429	0.907
w/o Photometric	1.14	0.774	1.14	0.774	0.499	0.925
Proposed	1.11	0.781	1.10	0.781	0.508	0.930

Table A.2: Evaluation of the proposed method on the FreiHand dataset (Zimmermann et al., 2019). Keypoint (KP) and Mesh errors are measured in cm.

A.13 HTML EXPERIMENTAL DETAILS (CHAPTER 7)

The photometric loss experiments are conducted on the FreiHAND dataset. The provided training dataset contains a total of 130,240 images: 32,560 unique images of hands with foreground masks, times four methods of background composition. However, three of the composition methods attempt to blend the hand into the background, which introduces severe artifacts in the hand appearance (see Figure. A.5).

Only unaltered 32,560 unique images are used for training to avoid learning these artifacts for texture estimation. The provided foreground masks were used to perform background augmentation without additional image harmonization or image coloration. ResNet-34 (He et al., 2016) is trained using Adam, with a learning rate of 0.001, and for 200 epochs in all of the experiments.

A full comparison of the pose and shape performance is provided in Table A.2. Following the evaluation procedure of Zimmermann et al. (2019), the meshes were aligned using Procrustes alignment as a rigid body transformation. Errors are measured in Euclidean distance (cm) between corresponding vertex points **(Mesh)** or keypoints **(KP)**. Area under the percentage-of-correct-keypoints curve **(AUC)** and F-scores at two thresholds **(F@5mm and F@15mm)** are additionally provided. The proposed method achieves slightly better pose and shape performance with the photometric loss **(Proposed)** than without **(w/o Photometric)**, and it achieves similar accuracy to the current state-of-the-art methods.

Figure A.5: The provided composed FreiHand data contains noticeable texture artifacts. These images were not used for training.

A.14 IMPACT OF SHADING REMOVAL (CHAPTER 7)

It is desirable for the parametric texture model to not include lighting effects. Although lighting conditions are made as uniform as possible while acquiring the scans, there are still smooth shading effects, especially at the boundary to the flat background surface. Without the shading removal step in the pipeline, the lighting effects contribute a large portion of the variation in the dataset. Hence, lighting variations are present in some of the first principal components of the PCA space. Refer to Figure A.6.

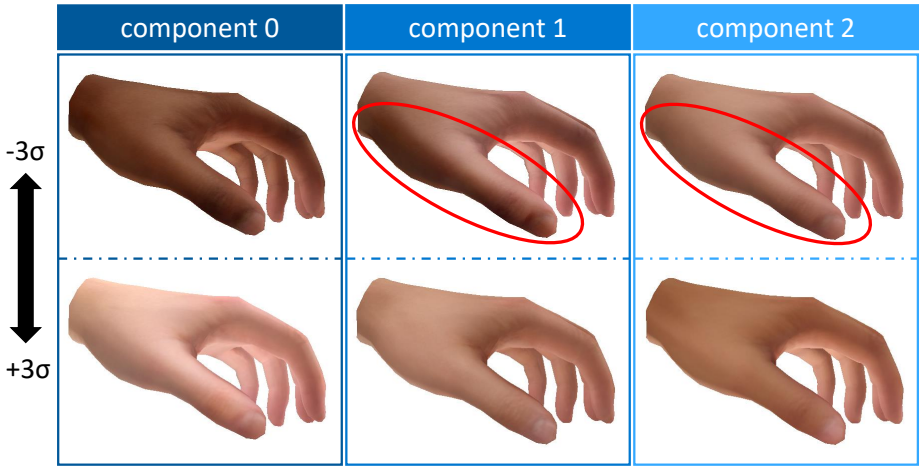


Figure A.6: When the texture PCA model is built without shading removal, the principal components contain a significant amount of lighting variation.

BIBLIOGRAPHY

Armagan, Anil, Guillermo Garcia-Hernando, Seungryul Baek, Shreyas Hampali, Mahdi Rad, Zhaohui Zhang, Shipeng Xie, MingXiu Chen, Boshen Zhang, Fu Xiong, Yang Xiao, Zhiguo Cao, Junsong Yuan, Pengfei Ren, Weiting Huang, Haifeng Sun, Marek Hrúz, Jakub Kanis, Zdeněk Krňoul, Qingfu Wan, Shile Li, Linlin Yang, Dongheui Lee, Angela Yao, Weiguo Zhou, Sijia Mei, Yunhui Liu, Adrian Spurr, Umar Iqbal, Pavlo Molchanov, Philippe Weinzaepfel, Romain Brégier, Gregory Rogez, Vincent Lepetit, and Tae-Kyun Kim (2020). "Measuring Generalisation to Unseen Viewpoints, Articulations, Shapes and Objects for 3D Hand Pose Estimation under Hand-Object Interaction." In: *European Conference on Computer Vision (ECCV)*. Glasgow, Scotland.

Baek, Seungryul, Kwang In Kim, and Tae-Kyun Kim (2018). "Augmented Skeleton Space Transfer for Depth-Based Hand Pose Estimation." In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 8330–8339.

Bailer, Christian, Manuel Finckh, and Hendrik P. A. Lensch (2012). "Scale Robust Multi View Stereo." In: *European Conference for Computer Vision (ECCV)*.

Ballan, Luca, Aparna Taneja, Jürgen Gall, Luc Van Gool, and Marc Pollefeys (2012). "Motion capture of hands in action using discriminative salient points." In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 640–653.

Belhumeur, Peter N, David J Kriegman, and Alan L Yuille (1999). "The bas-relief ambiguity." In: *International Journal of Computer Vision (IJCV)* 35.1, pp. 33–44.

Berry, Donald A and Bert Fristedt (1985). "Bandit problems: sequential allocation of experiments (Monographs on statistics and applied probability)." In: *London: Chapman and Hall* 5.71-87, pp. 7–7.

Besl, Paul J. and Neil D. McKay (1992). "A method for registration of 3-D shapes." In: *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 14.2, pp. 239–256.

Bieler, Didier, Semih Gunel, Pascal Fua, and Helge Rhodin (2019). "Gravity as a reference for estimating a person's height from video." In: *International Conference on Computer Vision (ICCV)*.

Boukhayma, Adnane, Rodrigo de Bem, and Philip H.S. Torr (2019). "3D Hand Shape and Pose From Images in the Wild." In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 10843–10852.

Bronstein, Michael M, Alexander M Bronstein, Ron Kimmel, and Irad Yavneh (2006). "Multigrid multidimensional scaling." In: *Numerical linear algebra with applications* 13.2-3, pp. 149–171.

Cao, Z., G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh (2019). "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields." In: *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Chen, Yujin, Zhigang Tu, Di Kang, Linchao Bao, Ying Zhang, Xuefei Zhe, Ruizhi Chen, and Junsong Yuan (2021). "Model-based 3d hand reconstruction via self-supervised learning." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 10451–10460.

Dabral, Rishabh, Soshi Shimada, Arjun Jain, Christian Theobalt, and Vladislav Golyanik (2021). "Gravity-aware monocular 3d human-object reconstruction." In: *International Conference on Computer Vision (ICCV)*.

Dibra, Endri, Thomas Wolf, Cengiz Oztireli, and Markus Gross (2017). "How to refine 3d hand pose estimation from unlabelled depth data?" In: *2017 International Conference on 3D Vision (3DV)*. IEEE, pp. 135–144.

Doosti, Bardia, Shujon Naha, Majid Mirbagheri, and David J Crandall (2020). "Hope-net: A graph-based model for hand-object pose estimation." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 6608–6617.

Fan, Zicong, Adrian Spurr, Muhammed Kocabas, Siyu Tang, Michael Black, and Otmar Hilliges (2021). "Learning to Disambiguate Strongly Interacting Hands via Probabilistic Per-pixel Part Segmentation." In: *International Conference on 3D Vision (3DV)*.

Fischler, Martin A and Robert C Bolles (1981). "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography." In: *Communications of the ACM* 24.6, pp. 381–395.

Gal, Yarin and Zoubin Ghahramani (2016). "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." In: *International Conference on Machine Learning (ICML)*. PMLR, pp. 1050–1059.

Ge, Liuhao, Hui Liang, Junsong Yuan, and Daniel Thalmann (2016). "Robust 3D Hand Pose Estimation in Single Depth Images: from Single-View CNN to Multi-View CNNs." In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Ge, Liuhao et al. (2017). "3D Convolutional Neural Networks for Efficient and Robust Hand Pose Estimation from Single Depth Images." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 5679–5688.

Grady, Patrick, Chengcheng Tang, Christopher D Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C Kemp (2021). "ContactOpt: Op-

timizing Contact to Improve Grasps." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 1471–1481.

Gretton, Arthur, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola (2012). "A Kernel Two-Sample Test." In: *Journal of Machine Learning Research* 13.25, pp. 723–773.

Habermann, Marc, Weipeng Xu, Michael Zollhöfer, Gerard Pons-Moll, and Christian Theobalt (Mar. 2019). "LiveCap: Real-Time Human Performance Capture From Monocular Video." In: *ACM Trans. Graph.* 38.2, 14:1–14:17. ISSN: 0730-0301.

Hampali, Shreyas, Mahdi Rad, Markus Oberweger, and Vincent Lepetit (2020). "Honnotate: A method for 3d annotation of hand and object poses." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 3196–3206.

Han, Shangchen, Beibei Liu, Robert Wang, Yuting Ye, Christopher D Twigg, and Kenrick Kin (2018). "Online optical marker-based hand tracking with deep labels." In: *ACM Transactions on Graphics (TOG)* 37.4, p. 166.

Hasson, Yana, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid (June 2020). "Leveraging Photometric Consistency Over Time for Sparsely Supervised Hand-Object Reconstruction." In: *Computer Vision and Pattern Recognition (CVPR)*.

Hasson, Yana, Gül Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid (2019). "Learning joint reconstruction of hands and manipulated objects." In: *Computer Vision and Pattern Recognition (CVPR)*.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep residual learning for image recognition." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.

Jia, Yangqing, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell (2014). "Caffe: Convolutional architecture for fast feature embedding." In: *International Conference on Multimedia*, pp. 675–678.

Joo, Hanbyul, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh (2017). "Panoptic Studio: A Massively Multiview System for Social Interaction Capture." In: *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Jung, Sungchul and Charles Hughes (Oct. 2016). "Body Ownership in Virtual Reality." In: *Collaboration Technologies and Systems (CTS)*, pp. 597–600.

Karunratanakul, Korrawe, Jinlong Yang, Yan Zhang, Michael J Black, Krikamol Muandet, and Siyu Tang (2020). "Grasping field: Learning

implicit representations for human grasps." In: *International Conference on 3D Vision (3DV)*. IEEE, pp. 333–344.

Kendall, Alex and Yarin Gal (2017). "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 30.

Keskin, Cem, Furkan Kıraç, Yunus Emre Kara, and Lale Akarun (2012). "Hand Pose Estimation and Hand Shape Classification using Multi-Layered Randomized Decision Forests." In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 852–863.

Khamis, Sameh, Jonathan Taylor, Jamie Shotton, Cem Keskin, Shahram Izadi, and Andrew Fitzgibbon (2015). "Learning an Efficient Model of Hand Shape Variation From Depth Images." In: *Computer Vision and Pattern Recognition (CVPR)*.

Kim, Dong Uk, Kwang In Kim, and Seungryul Baek (Oct. 2021). "End-to-End Detection and Pose Estimation of Two Interacting Hands." In: *International Conference on Computer Vision (ICCV)*, pp. 11189–11198.

Kingma, D. and J. Ba. (2015). "Adam: A method for stochastic optimization." In: *International Conference on Learning Representations (ICLR)*.

Kingma, Diederik P and Jimmy Ba (2015). "Adam: A method for stochastic optimization." In.

Kingma, Diederik P. and Max Welling (2014). "Auto-Encoding Variational Bayes." In: *International Conference on Learning Representations (ICLR)*. Ed. by Yoshua Bengio and Yann LeCun.

Klein, Georg and David Murray (2007). "Parallel Tracking and Mapping for Small AR Workspaces." In: *International Symposium on Mixed and Augmented Reality (ISMAR)*.

Kobyzev, Ivan, Simon JD Prince, and Marcus A Brubaker (2020). "Normalizing flows: An introduction and review of current methods." In: *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 43.11, pp. 3964–3979.

Kolotouros, Nikos, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis (Oct. 2021). "Probabilistic Modeling for Human Mesh Recovery." In: *ICCV*, pp. 11605–11614.

Kortylewski, Adam, Mario Wieser, Andreas Morel-Forster, Aleksander Wieczorek, Sonali Parbhoo, Volker Roth, and Thomas Vetter (2018). "Informed MCMC with Bayesian neural networks for facial image analysis." In: *Bayesian Deep Learning Workshop (NeurIPS)*.

Kwon, Taein, Bugra Tekin, Jan Stühmer, Federica Bogo, and Marc Pollefeys (2021). "H2o: Two hands manipulating objects for first person interaction recognition." In: *International Conference on Computer Vision (ICCV)*.

Kyriazis, Nikolaos and Antonis Argyros (2014). "Scalable 3d tracking of multiple interacting objects." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 3430–3437.

La Gorce, Martin de, David J Fleet, and Nikos Paragios (2011). "Model-based 3d hand pose estimation from monocular video." In: *Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 33.9, pp. 1793–1805.

La Gorce, Martin de, Nikos Paragios, and David J Fleet (2008). "Model-based hand tracking with texture, shading and self-occlusions." In: *Computer Vision and Pattern Recognition (CVPR)*.

Lewis, John P, Matt Cordner, and Nickson Fong (2000). "Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation." In: *Computer Graphics and Interactive Techniques*. ACM, pp. 165–172.

Li, Mengcheng, Liang An, Hongwen Zhang, Lianpeng Wu, Feng Chen, Tao Yu, and Yebin Liu (June 2022a). "Interacting Attention Graph for Single Image Two-Hand Reconstruction." In: *Computer Vision and Pattern Recognition (CVPR)*.

Li, Yuwei, Minye Wu, Yuyao Zhang, Lan Xu, and Jingyi Yu (Aug. 2021). "PIANO: A Parametric Hand Bone Model from Magnetic Resonance Imaging." In: *International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 816–822.

Li, Yuwei, Longwen Zhang, Zesong Qiu, Yingwenqi Jiang, Yuyao Zhang, Nianyi Li, Yuexin Ma, Lan Xu, and Jingyi Yu (2022b). *NIMBLE: A Non-rigid Hand Model with Bones and Muscles*. arXiv: 2202.04533 [cs.CV].

Lin, John, Ying Wu, and T.S. Huang (2000). "Modeling the constraints of human hand motion." In: *Proceedings Workshop on Human Motion*, pp. 121–126.

Liu, Shaowei, Hanwen Jiang, Jiarui Xu, Sifei Liu, and Xiaolong Wang (2021). "Semi-supervised 3d hand-object poses estimation with interactions in time." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 14687–14697.

Loshchilov, Ilya and Frank Hutter (2019). "Decoupled Weight Decay Regularization." In: *International Conference on Learning Representations (ICLR)*.

Malik, Jameel, Ibrahim Abdelaziz, Ahmed Elhayek, Soshi Shimada, Sk Aziz Ali, Vladislav Golyanik, Christian Theobalt, and Didier Stricker (2020). "HandVoxNet: Deep Voxel-Based Network for 3D Hand Shape and Pose Estimation from a Single Depth Map." In: *Computer Vision and Pattern Recognition (CVPR)*.

Malik, Jameel et al. (Oct. 2017). "Simultaneous Hand Pose and Skeleton Bone-Lengths Estimation from a Single Depth Image." In: *International Conference on 3D Vision (3DV)*, pp. 557–565.

Moon, Gyeongsik, Juyong Chang, and Kyoung Mu Lee (2019). "Camera Distance-aware Top-down Approach for 3D Multi-person Pose Estimation from a Single RGB Image." In: *International Conference on Computer Vision (ICCV)*.

Moon, Gyeongsik, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee (2020). "InterHand2.6M: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image." In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 548–564.

Mueller, Franziska, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt (June 2018). "GANerated Hands for Real-Time 3D Hand Tracking from Monocular RGB." In: *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Mueller, Franziska, Micah Davis, Florian Bernard, Oleksandr Sotnychenko, Mickeal Verschoor, Miguel A Otaduy, Dan Casas, and Christian Theobalt (2019). "Real-time Pose and Shape Reconstruction of Two Interacting Hands With a Single Depth Camera." In: *ACM Transactions on Graphics (TOG)* 38.4, p. 49.

Mueller, Franziska, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt (2017). "Real-time Hand Tracking under Occlusion from an Egocentric RGB-D Sensor." In: *International Conference on Computer Vision (ICCV)*.

Oberweger, M. et al. (2017). "DeepPrior++: Improving Fast and Accurate 3D Hand Pose Estimation." In: *International Conference on Computer Vision Workshop (ICCVW)*, pp. 585–594.

Oberweger, Markus, Paul Wohlhart, and Vincent Lepetit (2015). "Training a Feedback Loop for Hand Pose Estimation." In: *International Conference on Computer Vision (ICCV)*. IEEE, pp. 3316–3324.

Oikonomidis, Iason, Nikolaos Kyriazis, and Antonis A Argyros (2011). "Efficient Model-Based 3D Tracking of Hand Articulations using Kinect." In: *British Machine Vision Conference (BMVC)*. Vol. 1. 2.

Oikonomidis, Iasonas, Nikolaos Kyriazis, and Antonis A Argyros (2012). "Tracking the articulated motion of two strongly interacting hands." In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 1862–1869.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala (2019a). "PyTorch: An Imperative Style, High-Performance Deep Learning Library." In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., pp. 8024–8035.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. (2019b). "PyTorch: An imperative style, high-performance deep learning library." In: *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 8024–8035.

Pavlakos, Georgios, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black (2019). "Expressive Body Capture: 3D Hands, Face, and Body from a Single Image." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 10975–10985.

Prada, Fabián, Misha Kazhdan, Ming Chuang, and Hugues Hoppe (2018). "Gradient-domain processing within a texture atlas." In: *ACM Transactions on Graphics (TOG)* 37.4.

Qian, Chen, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun (2014). "Realtime and Robust Hand Tracking from Depth." In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 1106–1113.

Qian, Neng, Jiayi Wang, Franziska Mueller, Florian Bernard, Vladislav Golyanik, and Christian Theobalt (2020). "HTML: A Parametric Hand Texture Model for 3D Hand Reconstruction and Personalization." In: *European Conference on Computer Vision (ECCV)*. Springer.

Ravi, Nikhila, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari (2020). *PyTorch3D*. https://github.com/facebookresearch/pytorch3d.

Rehg, James M and Takeo Kanade (1994). "Visual tracking of high dof articulated structures: an application to human hand tracking." In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 35–46.

Rezende, Danilo and Shakir Mohamed (2015). "Variational inference with normalizing flows." In: *International Conference on Machine Learning (ICML)*, pp. 1530–1538.

Rogez, Grégory, James S Supančič, and Deva Ramanan (2015). "First-person pose recognition using egocentric workspaces." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 4325–4333.

Romero, Javier, Dimitrios Tzionas, and Michael J. Black (Nov. 2017). "Embodied Hands: Modeling and Capturing Hands and Bodies Together." In: *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*. 245:1–245:17 36.6.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-net: Convolutional networks for biomedical image segmentation." In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, pp. 234–241.

Rudnev, Viktor, Vladislav Golyanik, Jiayi Wang, Hans-Peter Seidel, Franziska Mueller, Mohamed Elgharib, and Christian Theobalt (2021).

"EventHands: Real-Time Neural 3D Hand Pose Estimation from an Event Stream." In: *International Conference on Computer Vision (ICCV)*.

Schönberger, Johannes Lutz and Jan-Michael Frahm (2016). "Structure-from-Motion Revisited." In: *Computer Vision and Pattern Recognition (CVPR)*.

Schönborn, Sandro, Bernhard Egger, Andreas Morel-Forster, and Thomas Vetter (2017). "Markov chain monte carlo for automated face image analysis." In: *International Journal of Computer Vision (IJCV)* 123.2, pp. 160–183.

Serra, E Simo (2011). "Kinematic model of the hand using computer vision." PhD thesis. Institut de Robotica i Informatica Industrial.

Shen, Jingjing, Thomas J Cashman, Qi Ye, Tim Hutton, Toby Sharp, Federica Bogo, Andrew Fitzgibbon, and Jamie Shotton (2020). "The phong surface: Efficient 3d model fitting using lifted optimization." In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 687–703.

Shiffman, Melvin A, Sid Mirrafati, Samuel M Lam, and Chelso G Cueteaux (2007). *Simplified facial rejuvenation*. Springer Science & Business Media.

Simon, Tomas, Hanbyul Joo, Iain Matthews, and Yaser Sheikh (2017). "Hand Keypoint Detection in Single Images using Multiview Bootstrapping." In: *Computer Vision and Pattern Recognition (CVPR)*.

Soliman, Mohamed, Franziska Mueller, Lena Hegemann, Joan Sol Roo, Christian Theobalt, and Jürgen Steimle (2018). "Fingerinput: Capturing expressive single-hand thumb-to-finger microgestures." In: *ACM International Conference on Interactive Surfaces and Spaces (ICISS)*, pp. 177–187.

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

Spurr, Adrian, Aneesh Dahiya, Xi Wang, Xucong Zhang, and Otmar Hilliges (2021). "Self-Supervised 3D Hand Pose Estimation from monocular RGB via Contrastive Learning." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 11230–11239.

Spurr, Adrian, Umar Iqbal, Pavlo Molchanov, Otmar Hilliges, and Jan Kautz (2020). "Weakly Supervised 3D Hand Pose Estimation via Biomechanical Constraints." In: *European Conference on Computer Vision (ECCV)*. Glasgow, Scotland.

Sridhar, Srinath, Franziska Mueller, Antti Oulasvirta, and Christian Theobalt (2015). "Fast and Robust Hand Tracking Using Detection-Guided Optimization." In: *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 3213–3221.

Sridhar, Srinath, Franziska Mueller, Michael Zollhöefer, Dan Casas, Antti Oulasvirta, and Christian Theobalt (2016). "Real-time Joint Tracking of a Hand Manipulating an Object from RGB-D Input." In: *European Conference on Computer Vision (ECCV)*.

Sridhar, Srinath, Antti Oulasvirta, and Christian Theobalt (2013). "Interactive Markerless Articulated Hand Motion Tracking using RGB and Depth Data." In: *International Conference on Computer Vision (ICCV)*. IEEE, pp. 2456–2463.

Stoll, Carsten, Nils Hasler, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt (2011). "Fast Articulated Motion Tracking using a Sums of Gaussians Body Model." In: *Proceedings of the International Conference on Computer Vision (ICCV)*. IEEE, pp. 951–958.

Tagliasacchi, Andrea, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly (Aug. 2015). "Robust Articulated-ICP for Real-Time Hand Tracking." In: *Computer Graphics Forum* 34.

Taylor, Jonathan, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Toby Sharp, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, et al. (2016). "Efficient and Precise Interactive Hand Tracking through Joint, Continuous Optimization of Pose and Correspondences." In: *ACM Transactions on Graphics (TOG)* 35.4, pp. 1–12.

Taylor, Jonathan, Vladimir Tankovich, Danhang Tang, Cem Keskin, David Kim, Philip Davidson, Adarsh Kowdle, and Shahram Izadi (2017). "Articulated Distance Fields for Ultra-fast Tracking of Hands Interacting." In: *ACM Transactions on Graphics (TOG)* 36.6, 244:1–244:12.

Tekin, Bugra, Federica Bogo, and Marc Pollefeys (June 2019). "H+O: Unified Egocentric Recognition of 3D Hand-Object Poses and Interactions." In: *Computer Vision and Pattern Recognition (CVPR)*.

Tkach, Anastasia, Mark Pauly, and Andrea Tagliasacchi (2016). "Sphere-Meshes for Real-time Hand Modeling and Tracking." In: *ACM Transactions on Graphics (TOG)* 35.6, 222:1–222:11.

Tkach, Anastasia, Andrea Tagliasacchi, Edoardo Remelli, Mark Pauly, and Andrew Fitzgibbon (2017). "Online Generative Model Personalization for Hand Tracking." In: *ACM Transactions on Graphics (TOG)* 36.6, 243:1–243:11.

Tompson, Jonathan, Murphy Stein, Yann Lecun, and Ken Perlin (2014). "Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks." In: *ACM Transactions on Graphics (TOG)* 33.5, pp. 1–10.

Tzionas, Dimitrios, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall (2016). "Capturing Hands in Action using

Discriminative Salient Points and Physics Simulation." In: *International Journal of Computer Vision (IJCV)*.

Ulyanov, Dmitry, Andrea Vedaldi, and Victor Lempitsky (2016). "Instance normalization: The missing ingredient for fast stylization." In: *arXiv preprint arXiv:1607.08022*.

Verschoor, Mickeal, Daniel Lobo, and Miguel A Otaduy (2018). "Soft hand simulation for smooth and robust natural interaction." In: *Virtual Reality and 3D User Interfaces (VR)*. IEEE, pp. 183–190.

Wan, Chengde, Thomas Probst, Luc Van Gool, and Angela Yao (2017). "Crossing Nets: Combining GANs and VAEs with a Shared Latent Space for Hand Pose Estimation." In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 680–689.

Wan, Chengde et al. (2019). "Self-Supervised 3D Hand Pose Estimation Through Training by Fitting." In: *Computer Vision and Pattern Recognition (CVPR)*.

Wang, Jiayi, Diogo Luvizon, Franziska Mueller, Florian Bernard, Adam Kortylewski, Dan Casas, and Christian Theobalt (2022). "HandFlow: Quantifying View-Dependent 3D Ambiguity in Two-Hand Reconstruction with Normalizing Flow." In: *Vision, Modeling and Visualization (VMV) Best Paper Honorable Mention*.

Wang, Jiayi, Franziska Mueller, Florian Bernard, Suzanne Sorli, Oleksandr Sotnychenko, Neng Qian, Miguel A. Otaduy, Dan Casas, and Christian Theobalt (Dec. 2020a). "RGB2Hands: Real-Time Tracking of 3D Hand Interactions from Monocular RGB Video." In: *ACM Transactions on Graphics (TOG)* 39.6.

Wang, Jiayi, Franziska Mueller, Florian Bernard, and Christian Theobalt (2020b). "Generative Model-Based Loss to the Rescue: A Method to Overcome Annotation Errors for Depth-Based Hand Pose Estimation." In: *Automatic Face and Gesture Recognition (FG)*. IEEE, pp. 93–100.

Wehrbein, Tom, Marco Rudolph, Bodo Rosenhahn, and Bastian Wandt (2021). "Probabilistic Monocular 3D Human Pose Estimation with Normalizing Flows." In: *International Conference on Computer Vision (ICCV)*.

Wheatland, Nkenge, Yingying Wang, Huaguang Song, Michael Neff, Victor Zordan, and Sophie Jörg (2015). "State of the art in hand and finger modeling and animation." In: *Computer Graphics Forum*. Vol. 34. 2. Wiley Online Library, pp. 735–760.

Winkler, Christina, Daniel Worrall, Emiel Hoogeboom, and Max Welling (2019). "Learning likelihoods with conditional normalizing flows." In: *arXiv preprint arXiv:1912.00042*.

Wöhlke, Jan, Shile Li, and Dongheui Lee (2018). "Model-based hand pose estimation for generalized hand shape with appearance normalization." In: *arXiv preprint arXiv:1807.00898*.

Wu, Xiaokun et al. (2018). "HandMap: Robust Hand Pose Estimation via Intermediate Dense Guidance Map Supervision." In: *European Conference on Computer Vision (ECCV)*.

Wu, Ying, J.Y. Lin, and T.S. Huang (2001). "Capturing natural hand articulation." In: *International Conference on Computer Vision (ICCV)*. Vol. 2, 426–432 vol.2.

Xiang, Donglai, Hanbyul Joo, and Yaser Sheikh (2019). "Monocular Total Capture: Posing face, body, and hands in the wild." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 10965–10974.

Xu, Chi and Li Cheng (2013). "Efficient Hand Pose Estimation from a Single Depth Image." In: *International Conference on Computer Vision (ICCV)*. IEEE, pp. 3456–3462.

Yang, Linlin and Angela Yao (2019). "Disentangling latent hands for image synthesis and pose estimation." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 9877–9886.

Ye, Qi and Tae-Kyun Kim (2018). "Occlusion-aware Hand Pose Estimation Using Hierarchical Mixture Density Network." In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 801–817.

Ye, Qi, Shanxin Yuan, and Tae-Kyun Kim (2016). "Spatial Attention Deep Net with Partial PSO for Hierarchical Hybrid Hand Pose Estimation." In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 346–361.

Yenamandra, Tarun, Florian Bernard, Jiayi Wang, Franziska Mueller, and Christian Theobalt (2019). "Convex Optimisation for Inverse Kinematics." In: *International Conference on 3D Vision (3DV)*. IEEE, pp. 318–327.

Yuan, Shanxin, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim (2017). "Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 4866–4874.

Yuan, Shanxin et al. (2018). "Depth-based 3d hand pose estimation: From current achievements to future goals." In: *Computer Vision and Pattern Recognition (CVPR)*.

Zhang, Baowen, Yangang Wang, Xiaoming Deng, Yinda Zhang, Ping Tan, Cuixia Ma, and Hongan Wang (2021). "Interacting Two-Hand 3D Pose and Shape Reconstruction from Single Color Image." In: *International Conference on Computer Vision (ICCV)*.

Zhang, Xiong, Qiang Li, Hong Mo, Wenbo Zhang, and Wen Zheng (2019). "End-to-End Hand Mesh Recovery From a Monocular RGB Image." In: *International Conference on Computer Vision (ICCV)*.

Zhao, Long, Xi Peng, Yuxiao Chen, Mubbasir Kapadia, and Dimitris N Metaxas (2020). "Knowledge as priors: Cross-modal knowledge generalization for datasets without superior knowledge." In: *Computer Vision and Pattern Recognition (CVPR)*, pp. 6528–6537.

Zhao, Wenping, Jianjie Zhang, Jianyuan Min, and Jinxiang Chai (2013). "Robust realtime physics-based motion control for human grasping." In: *ACM Transactions on Graphics (TOG)* 32.6, pp. 1–12.

Zhou, Xingyi, Qingfu Wan, Wei Zhang, Xiangyang Xue, and Yichen Wei (2016). "Model-based Deep Hand Pose Estimation." In: *IJCAI*. IJCAI'16. New York, New York, USA: AAAI Press, pp. 2421–2427. ISBN: 978-1-57735-770-4.

Zhou, Yi, Connelly Barnes, Lu Jingwan, Yang Jimei, and Li Hao (June 2019). "On the Continuity of Rotation Representations in Neural Networks." In: *Computer Vision and Pattern Recognition (CVPR)*.

Zimmermann, Christian and Thomas Brox (2017). "Learning to Estimate 3D Hand Pose from Single RGB Images." In: *International Conference on Computer Vision (ICCV)*.

Zimmermann, Christian, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox (2019). "FreiHAND: A Dataset for Markerless Capture of Hand Pose and Shape from Single RGB Images." In: *International Conference on Computer Vision (ICCV)*.