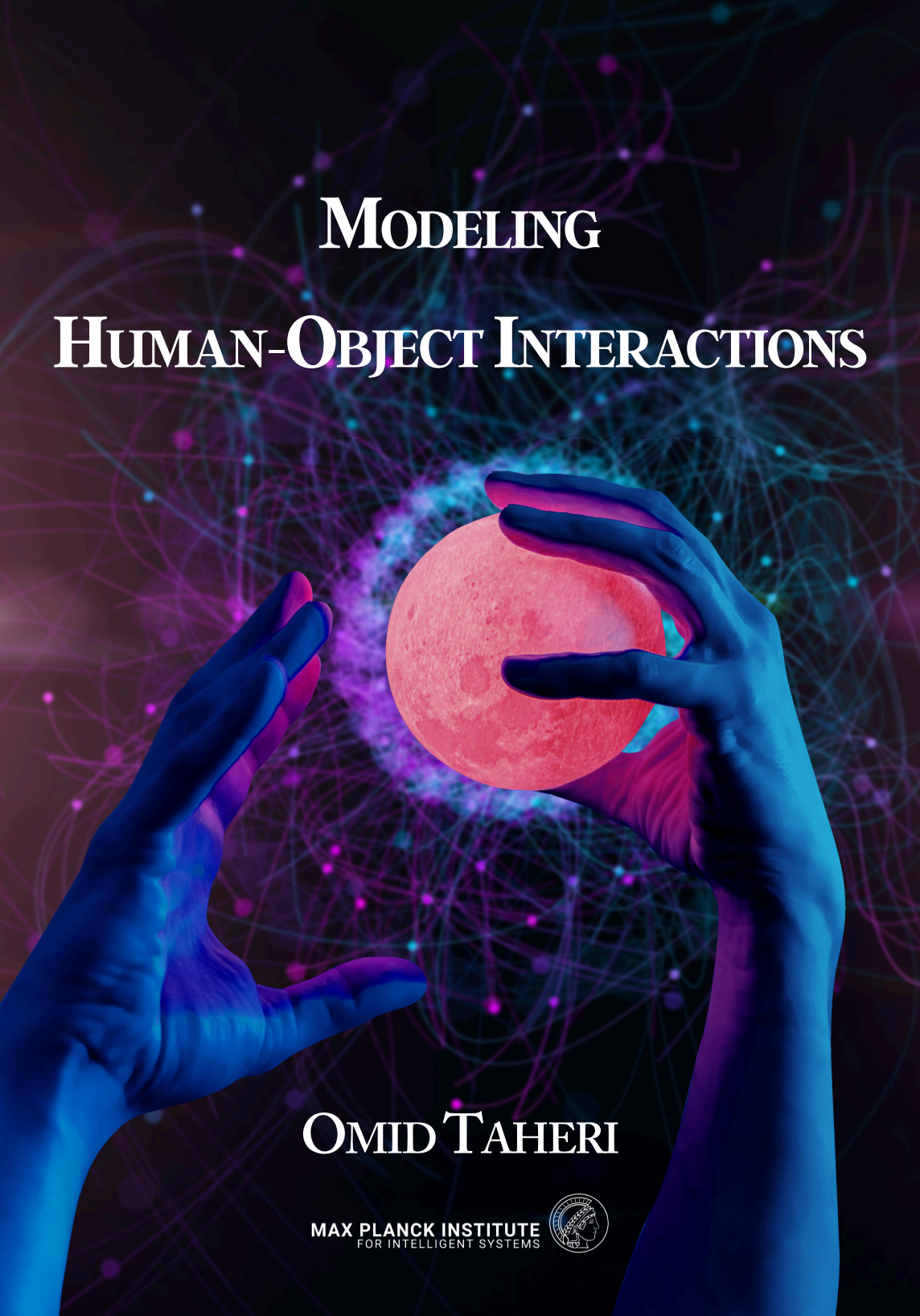
Modeling Human-Object Interactions



Omid Taheri

Max Planck Institute for Intelligent Systems

Modeling Dynamic 3D Human-Object Interactions: From Capture to Synthesis

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von

Omid Taheri

aus Marvdasht, Iran

Tübingen
2024

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	04.07.2024
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Gerard Pons-Moll
2. Berichterstatter:	Prof. Dr. Michael J. Black
3. Berichterstatterin:	Prof. Dr. Angela Dai

To Mom and Dad,

for your unconditional support, sacrifices, and faith along the way.

تقدیم به پدر و مادر فداکارم.

زندگی صحنه‌ی یکتای هنرمندی ماست

هر کسی نغمه‌ی خود خواند و از صحنه رود

صحنه پیوسته به جاست

خرم آن نغمه که مردم بسپارند به یاد

ژاله اصفهانی

Life's a unique stage for our artistry's display,
Each soul sings its own song, then departs from the fray.
The stage remains steadfast, ever there to stay,
Fortunate the melodies that in hearts forever play.

- Jaleh Esfahani

Abstract

Modeling digital humans that move and interact realistically with virtual 3D worlds has emerged as an essential research area recently, with significant applications in computer graphics, virtual and augmented reality, telepresence, the Metaverse, and assistive technologies. In particular, human-object interaction, encompassing full-body motion, hand-object grasping, and object manipulation, lies at the core of how humans execute tasks and represents the complex and diverse nature of human behavior. Therefore, accurate modeling of these interactions would enable us to simulate avatars to perform tasks, enhance animation realism, and develop applications that better perceive and respond to human behavior. Despite its importance, this remains a challenging problem, due to several factors such as the complexity of human motion, the variance of interaction based on the task, and the lack of rich datasets capturing the complexity of real-world interactions. Prior methods have made progress, but limitations persist as they often focus on individual aspects of interaction, such as body, hand, or object motion, without considering the holistic interplay among these components. This Ph.D. thesis addresses these challenges and contributes to the advancement of human-object interaction modeling through the development of novel datasets, methods, and algorithms.

The first major contribution of this research is the introduction of the GRAB dataset. Training computer models to understand and synthesize realistic avatar interactions requires a rich dataset containing accurate hand-object grasps, complex 3D object shapes, contact information, and full-body motion over time. However, such a comprehensive dataset has been missing, as capturing it comes with numerous challenges. Existing datasets predominantly focus on hand-only interactions and face challenges such as occlusions and inaccurate tracking tools. We overcome these limitations by considering full-body interactions, using a high-quality motion capture system, and adopting state-of-the-art methods to accurately track detailed body, head, hand, and object shapes and motions. Going beyond existing datasets, we collect GRAB, the first human-object-interaction dataset containing the 3D body and object motion, accurate hand-object interaction, head motion, and detailed contact. Utilizing GRAB, we train GrabNet, a model that generates state-of-the-art hand grasps for unseen 3D objects. Overall, GRAB and GrabNet serve as a foundation for further research and have enabled the development of numerous methods for modeling and synthesizing human-object interactions.

The initial step of "walking" toward and "grasping" an object is a requirement for any object-interaction motion, and is essential for realistic avatar motion. Despite its importance, previous work has mainly focused on static grasps or the major limbs of the body, ignoring the hands and head. To address this, we need to generate full-body motions, with realistic hand grasps and head poses simultaneously. This is challenging due to the extensive state-space of poses, the necessity for consistency between body and hands while maintaining physical constraints, and the vital role of the head during interactions. To address this, we introduce GOAL, the first model that takes an initial 3D body and object and generates the avatar's motion that walks and grasps the object with realistic body pose, head orientation, hand grasp, and foot-ground contact. To achieve this we exploit complementary grasp representations, namely, SMPL-X body pose and vertex offsets between the body and object, and a novel interaction-aware feature that transforms body-to-object distance to better localize the object with respect to the body. These contributions facilitated the generation of realistic avatar motions, advancing the development of digital human modeling.

To authentically simulate virtual avatar motions, in addition to walking and grasping objects, avatars should have realistic and physically plausible finger motions while manipulating objects. While seemingly trivial for humans, this involves satisfying numerous semantic and physical constraints, making it a complex challenge to address. Consequently, there has been a lack of methods capable of automatically generating hand motions that are consistent with the full-body pose and object manipulation. GRIP bridges this gap, by taking the 3D motion of the body and the object and synthesizing realistic motion for both left and right hands before, during, and after object manipulation. GRIP leverages spatio-temporal information between the body and the object to extract rich temporal interaction cues, which help generalization to new objects. Moreover, it introduces a motion temporal consistency constraint applied in the latent space, leading to generating consistent interaction motions. Overall, GRIP "upgrades" sequences of body and object motion to include detailed hand-object interaction and further enhances the realism and applicability of digital human modeling.

In conclusion, this Ph.D. thesis advances the field of 3D human-object interaction modeling by offering novel, holistic, and practical approaches. By developing new resources such as the GRAB dataset and models like GrabNet, GOAL, and GRIP, we are not just incrementally improving the field but transforming the way we approach, understand, and solve problems in human-object interaction modeling. As the digital era advances and the Metaverse concept becomes more prevalent, our work marks a crucial step towards creating more realistic and immersive virtual worlds where digital humans interact with their environment in an authentically human-like way. We envision our contributions will inspire and guide future research to continue exploring, innovating, and enhancing the capabilities of digital human modeling.

Zusammenfassung

Modellierung digitaler Menschen, die sich realistisch in virtuellen 3D-Welten bewegen und interagieren, hat sich kürzlich als ein wesentliches Forschungsgebiet herauskristallisiert, mit bedeutenden Anwendungen in der Computergrafik, virtueller und erweiterter Realität, Telepräsenz, dem Metaverse und assistiven Technologien. Insbesondere liegt die Interaktion zwischen Mensch und Objekt, die die gesamte Körperbewegung, das Greifen von Objekten mit der Hand und die Objektmanipulation umfasst, im Kern dessen, wie Menschen Aufgaben ausführen, und repräsentiert die komplexe und vielfältige Natur menschlichen Verhaltens. Daher würde eine genaue Modellierung dieser Interaktionen es uns ermöglichen, Avatare zu simulieren, die Aufgaben ausführen, den Realismus der Animation zu verbessern und Anwendungen zu entwickeln, die menschliches Verhalten besser wahrnehmen und darauf reagieren. Trotz seiner Bedeutung bleibt dies aufgrund mehrerer Faktoren wie der Komplexität der menschlichen Bewegung, der Varianz der Interaktion basierend auf der Aufgabe und dem Mangel an umfangreichen Datensätzen, die die Komplexität realer Interaktionen erfassen, ein herausforderndes Problem. Frühere Methoden haben Fortschritte gemacht, aber Einschränkungen bestehen weiterhin, da sie sich oft auf einzelne Aspekte der Interaktion konzentrieren, wie Körper-, Hand- oder Objektbewegung, ohne das ganzheitliche Zusammenspiel dieser Komponenten zu berücksichtigen. Diese Doktorarbeit befasst sich mit diesen Herausforderungen und trägt zur Weiterentwicklung der Modellierung der Interaktion zwischen Mensch und Objekt bei, durch die Entwicklung neuer Datensätze, Methoden und Algorithmen.

Der erste bedeutende Beitrag dieser Forschung ist die Einführung des GRAB-Datensatzes. Um Computermodelle zu trainieren, die realistische Avatar-Interaktionen verstehen und synthetisieren, wird ein umfangreicher Datensatz benötigt, der genaue Hand-Objekt-Griffe, komplexe 3D-Objektformen, Kontaktinformationen und die gesamte Körperbewegung über die Zeit enthält. Ein solch umfassender Datensatz hat jedoch gefehlt, da seine Erfassung zahlreiche Herausforderungen mit sich bringt. Bestehende Datensätze konzentrieren sich überwiegend auf Hand-Interaktionen und stehen vor Herausforderungen wie Verdeckungen und ungenauen Tracking-Tools. Wir überwinden diese Einschränkungen, indem wir vollständige Körperinteraktionen berücksichtigen, ein hochwertiges Motion-Capture-System verwenden und modernste Methoden anwenden, um detaillierte Körper-, Kopf-, Hand- und Objektformen und -bewegungen genau zu verfolgen. Über bestehende Datensätze hinausgehend,

sammeln wir GRAB, den ersten Mensch-Objekt-Interaktionsdatensatz, der die 3D-Bewegung von Körper und Objekt, genaue Hand-Objekt-Interaktion, Kopfbewegung und detaillierten Kontakt enthält. Mit GRAB trainieren wir GrabNet, ein Modell, das state-of-the-art Handgriffe für ungesehene 3D-Objekte generiert. Insgesamt dienen GRAB und GrabNet als Grundlage für weitere Forschungen und haben die Entwicklung zahlreicher Methoden zur Modellierung und Synthese von Mensch-Objekt-Interaktionen ermöglicht.

Der erste Schritt des „Gehens“ zu und des „Greifens“ eines Objekts ist eine Voraussetzung für jede Objekt-Interaktionsbewegung und essentiell für realistische Avatar-Bewegungen. Trotz seiner Bedeutung lag der Schwerpunkt bisheriger Arbeiten hauptsächlich auf statischen Griffen oder den Hauptgliedmaßen des Körpers und ignorierte die Hände und den Kopf. Um dies anzugehen, müssen wir vollständige Körperbewegungen mit realistischen Handgriffen und Kopfpositionen gleichzeitig generieren. Dies ist aufgrund des umfangreichen Zustandsraums der Posen, der Notwendigkeit der Konsistenz zwischen Körper und Händen unter Beibehaltung physischer Einschränkungen und der wichtigen Rolle des Kopfes während der Interaktionen herausfordernd. Um dies zu adressieren, führen wir GOAL ein, das erste Modell, das einen anfänglichen 3D-Körper und ein Objekt nimmt und die Bewegung des Avatars generiert, der das Objekt mit realistischer Körperhaltung, Kopforientierung, Handgriff und Fuß-Boden-Kontakt geht und greift. Um dies zu erreichen, nutzen wir komplementäre Griffdarstellungen, nämlich die SMPL-X-Körperpose und Vertex-Versätze zwischen Körper und Objekt, und ein neuartiges interaktionsbewusstes Merkmal, das die Körper-zu-Objekt-Distanz transformiert, um das Objekt in Bezug auf den Körper besser zu lokalisieren. Diese Beiträge erleichterten die Generierung realistischer Avatar-Bewegungen und förderten die Entwicklung der digitalen Menschenmodellierung.

Um virtuelle Avatar-Bewegungen authentisch zu simulieren, zusätzlich zum Gehen und Greifen von Objekten, sollten Avatare realistische und physikalisch plausible Fingerbewegungen während der Objektmanipulation aufweisen. Obwohl dies für Menschen scheinbar trivial ist, erfordert es die Erfüllung zahlreicher semantischer und physischer Einschränkungen, was es zu einer komplexen Herausforderung macht. Infolgedessen gab es einen Mangel an Methoden, die in der Lage sind, automatisch Handbewegungen zu generieren, die mit der gesamten Körperhaltung und der Objektmanipulation konsistent sind. GRIP schließt diese Lücke, indem es die 3D-Bewegung des Körpers und des Objekts nimmt und realistische Bewegungen für beide Hände vor, während und nach der Objektmanipulation synthetisiert. GRIP nutzt raumzeitliche Informationen zwischen Körper und Objekt, um reichhaltige zeitliche Interaktionshinweise zu extrahieren, die die Generalisierung auf neue Objekte unterstützen. Darüber hinaus führt es eine Bewegungskonsistenzbeschränkung im

latenten Raum ein, die zu konsistenten Interaktionsbewegungen führt. Insgesamt „verbessert“ GRIP Sequenzen von Körper- und Objektbewegungen, indem es detaillierte Hand-Objekt-Interaktionen einbezieht und den Realismus und die Anwendbarkeit der digitalen Menschenmodellierung weiter erhöht.

Zusammenfassend fördert diese Doktorarbeit das Feld der 3D-Mensch-Objekt-Interaktionsmodellierung, indem sie neuartige, ganzheitliche und praktische Ansätze bietet. Durch die Entwicklung neuer Ressourcen wie des GRAB-Datensatzes und Modelle wie GrabNet, GOAL und GRIP verbessern wir nicht nur schrittweise das Feld, sondern transformieren die Art und Weise, wie wir Probleme in der Modellierung der Interaktion zwischen Mensch und Objekt angehen, verstehen und lösen. Da das digitale Zeitalter voranschreitet und das Konzept des Metaverse immer verbreiteter wird, markiert unsere Arbeit einen entscheidenden Schritt in Richtung der Schaffung realistischerer und immersiverer virtueller Welten, in denen digitale Menschen auf authentisch menschliche Weise mit ihrer Umgebung interagieren. Wir stellen uns vor, dass unsere Beiträge zukünftige Forschungen inspirieren und anleiten werden, um weiterhin zu erforschen, zu innovieren und die Fähigkeiten der digitalen Menschenmodellierung zu verbessern.

Acknowledgements

As I reflect upon the past few years, it is hard to believe that this journey has reached its conclusion. It has been an incredible journey of personal and professional growth and many challenges, that I did not traverse alone, and it is only appropriate that I take the time to acknowledge those who have supported me along the way.

First and foremost, my deepest gratitude goes to my supervisor, Michael Black. His trust , faith , and unwavering support have been my pillars of strength throughout this journey. His guidance has not only transformed my academic understanding but also provided valuable life lessons. The transition from my different background into this field was challenging, but his support, patience, and unique research approach made it possible. His mentoring was more than just academic; he created an environment that encouraged open discussions and exploration, making learning a holistic process. This has significantly influenced my development as a scholar.

My heartfelt thanks also go to my co-supervisor, Dimitrios Tzionas for his tireless support and guidance. His patience, open-mindedness, and friendship were the light that always guided me through. His presence was felt in every step, making the journey smoother and less intimidating. His contributions to my work have been invaluable, especially our late-night discussions during all deadlines, which were always effective and worked in the end ;).

Next, my sincerest appreciation goes to all the members of our department, Perceiving Systems. You were my second family, the people who made me feel at home when I was away from home. Your company, discussions, and endless supply of cake (:D) made the department more than just a place of work. So, thank you Vassilis, Partha, Soubhik, Lea, Shashank, Radek, Sai, Markos, Mert, Nikos, Yao, Muhammed, Victoria, Qianli, Arjun, Alex, Haiwen, Ahmed, Paola, Priyanka, Timo, Omri, Peter, Marilyn, Yuliang, Enes, Yinghao, Anurag, Joachim, Giorgio, Anstasios, Hanz, Yufeng, Yufei, Sergi, Suraj, Rick, Artur, Nadine, Paul, Sergey, Mohammed, Vannesa, David, Jinlong, Hongwei, Nitin, Eric, Benjamin, Silvia, Aamir, Siyu, Nima, Rahul, Yandong, Srisha, Lea (x2), Nefeli, Kiran, Camila, Elia, Eric, Claudia, Florian, Arina, Tomasz, Tithy, Prerana, Mirela, and other people who were not part of PS but still supported in many ways, Bala, Rama, Berna, Axel, Katja, Malte, Wojciech, Sina, Angela, Miriam, Ines, Rebecca. Special thanks to Melanie, Nicole, Johanna, and Cordelia for making our lives easier in the lab. Also, a big thank

you to the capture/data team, Tsvetelina, Markus, Mason, Tobias, Andrea, Senya, Galina, Taylor, Alpar, Filippe, and everyone who was part of this journey.

I would like to thank Yi, Yang, Duygu, and Sören for hosting me at Adobe during my internship and making it into an amazing experience.

Special thanks to Prof. Pons-Moll for reviewing this thesis, Prof. Dai, and Prof. Geiger for agreeing to be part of my examination committee, and Prof. Kuchenbecker for being part of my TAC and adding invaluable insights into my PhD studies.

Thanks to Leila, Sara, and Katherine, and all IMPRS-IS people for their support throughout my PhD program.

I extend my sincere gratitude to the Max Planck Institute, the International Max Planck Research School for Intelligent Systems (IMPRS-IS), and the German Federal Ministry of Education and Research (BMBF) for their financial support and contributions to my research.

I wish to express my gratitude towards the renowned Iranian singer, Homayoun Shajarian. His voice was my constant companion, providing the soundtrack to my work and life. It served as a source of inspiration and solace during challenging times.

Moreover, I must acknowledge the roots that have held me firm – my family. To my parents, Ayaz and Zahra; words cannot adequately express my gratitude for your unwavering belief in me. You have sacrificed so much, always prioritizing my needs above your own. Your love and support have been the invisible hands lifting me up at every step. My gratitude extends to my siblings, Maryam, Hamid, Saeed, and Ali, whose love and encouragement have been a constant source of strength.

Finally, I would like to sincerely acknowledge Elahe for her support, patience, and encouragement throughout significant parts of my Ph.D. journey.

In conclusion, it is impossible to name everyone who has played a role in my journey, but every contribution, whether big or small, has not gone unnoticed. I am privileged and humbled to have shared this journey with you all.
(cover photo was designed using adobe express content).

Thank you, from the bottom of my heart.

باسپاس فراوان

Omid Taheri
Tübingen, Germany, July 4, 2024

List of Papers

I. *GRAB: A dataset of whole-body human grasping of objects.*
Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas.
In European Conference on Computer Vision (ECCV), (**2020**).

II. *GOAL: Generating 4D whole-body motion for hand-object grasping.*
Omid Taheri, Vasileios Choutas, Michael J. Black, and Dimitrios Tzionas.
In Conference on Computer Vision and Pattern Recognition (CVPR), **2022**.

III. *GRIP: Generating Interaction Poses Using Latent Consistency and Spatial Cues.*
Omid Taheri, Yi Zhou, Dimitrios Tzionas, Yang Zhou, Duygu Ceylan, Sören Pirk, and Michael J. Black.
In International Conference on 3D Vision (3DV), **2024**.

Papers not included in the thesis:

IV. *ARCTIC: A Dataset for Dexterous Bimanual Hand-Object Manipulation.*
Zicong Fan. **Omid Taheri**, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J. Black, and Otmar Hilliges,
In Conference on Computer Vision and Pattern Recognition (CVPR), **2023**

V. *InterCap: Joint Markerless 3D Tracking of Humans and Objects in Interaction.*
Yinghao Huang, **Omid Taheri**, Michael J. Black, and Dimitrios Tzionas.
In German Conference on Pattern Recognition (GCPR), **2022**.

VI. *IPMAN: 3D Human Pose Estimation via Intuitive Physics.*
Shashank Tripathi, Lea Müller, Chun-Hao P. Huang, **Omid Taheri**, Michael J. Black, and Dimitrios Tzionas.
In Conference on Computer Vision and Pattern Recognition (CVPR), **2023**.

Contribution report

Papers I, II, and III are used for this thesis and are mainly the work of the author. Paper IV is the main work of Zicong Fan and the author of this thesis mainly supervised the project. especially the data capture process and protocol. Paper V is the main work of Yinghao Huang where the author contributed with visualizations, data analysis, and discussion of results. Paper VI is the main work of Shashank Tripathi and the author mainly contributed to implementing baselines, evaluations of the method, and discussion of results.

Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols that will be later used within the body of the document.

Abbreviations

Here we list the frequently used abbreviations in the thesis.

MoCap	Motion Capture
AMT	Amazon Mechanical Turk
AR	Augmented Reality
BPS	Basis Point Set
CAD	Computer-Aided Design
cVAE	Conditional Variational Autoencoder
DoF	Degrees of Freedom
GD	Gradient Descent
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
GT	Ground-Truth
IAA	Interaction-Aware Attention
IMU	Inertial Measurement Unit
LBS	Linear Blend Skinning
LSTM	Long-Short Term Memory
ML	Machine Learning
MPJPE	Mean Per-Joint Point Error
MPVPE	Mean Per-Vertex Point Error
NN	Neural Network
PCA	Principal Component Analysis
RGB	Red Green Blue (color model)
RNN	Recurrent Neural Network
SMPL	Skinned Multi-Person Linear model
SMPL+H	SMPL+ Hands
SMPL-X	SMPL eXpressive
SOTA	State-of-the-art
V2V	Vertex-to-Vertex
Vicon	Brand name of a motion capture system
VR	Virtual Reality

Math Symbols

Here we list the mathematical symbols used in the thesis.

β ..... Body shape parameters
$\boldsymbol{E}$ ..... Energy function
ψ ..... Facial expression parameters
$\boldsymbol{\Theta}_{t\text{-}5:t}$ ..... SMPL-X parameters of the last 5 frames
θ ..... Body Pose parameters
γ ..... Body translation parameters
$J(\beta)$ ..... 3D joints of the SMPLX kinematic skeleton
N_b ..... Number of vertices in the 3D body mesh

Chapter 2

$\varepsilon_{contact}$ ..... Threshold for contact
$\mathcal{M}_b$ ..... Penetrating sub-mesh of the body
$\mathcal{R}_b$ ..... Intersection rings of the body mesh
$\mathcal{R}_o$ ..... Intersection rings of the object mesh
M_b ..... 3D body mesh
$d_{o \to b}$ ..... Distance from object to body
d ..... Distance
F_b ..... Triangles of the 3D body mesh
$V_b^{\mathcal{C}}$ ..... Vertices of the body in contact
V_b ..... Vertices of the 3D body mesh
$V_o^{\mathcal{C}}$ ..... Vertices of the object in contact
V_o ..... Vertices of the object mesh

Chapter 3

γ ..... Translation in MANO model
θ_{wrist} ..... Wrist rotation in MANO model
BPS_o ..... Basis point set representation for object
D ..... Distances between the hand and the object
Z ..... Grasping embedding space

Chapter 4

$\boldsymbol{b}_g^h$ ..... BPS representation of the hand in the goal frame
$\boldsymbol{b}^{\circ}$ ..... Basis Point Set representation of the object
$\boldsymbol{d}$ ..... Body to object distance
$\boldsymbol{z}_g$ ..... Static grasp latent code
$\boldsymbol{q}$ ..... Head orientation vector
$\hat{\boldsymbol{q}}$ ..... Ground Truth head orientation vector
$\boldsymbol{d}^{b \to \circ}$ ..... 3D offset vectors from body to object

$\boldsymbol{d}^{h\to\circ}$ ..... Offset vectors from hand vertices to the object
$\hat{\boldsymbol{d}}^{h\to\circ}$ ..... Predicted offset vectors from hand vertices to the object
$\boldsymbol{d}^{h}_{t\to g}$ ..... Hand vertex offsets from the current frame to the goal frame
N ..... Sampled Vertices
$\boldsymbol{v}_t$ ..... Sampled body vertices in the current frame
$\dot{\boldsymbol{v}}_t$ ..... Velocities of the sampled body vertices in the current frame
$\boldsymbol{v}^h$ ..... Hand vertices

Chapter 5

$\boldsymbol{h^A}$ ..... Ambient Sensor features
$\boldsymbol{d}$ ..... Average hand-to-object distance
$\dot{\boldsymbol{d}}$ ..... Rate of change of hand-to-object distance
$\boldsymbol{\theta^{la}_p}, \boldsymbol{\theta^{ra}_p}$ ..... Noisy left and right arm pose predictions from ANet
$\boldsymbol{h^P}$ ..... Proximity Sensor features
$\boldsymbol{\theta^r}, \boldsymbol{\theta^l}$ ..... Right and left hand pose parameters

Units

Here we list the units.

° ..... Degrees
fps ..... Frames per Second
m ..... Meter
mm ..... Millimeter
MP ..... Megapixel

"In our hands lies the power to shape our world."

— Nelson Mandela

1

Introduction

Contents

1.1 Motivation

The rapidly expanding Metaverse, Virtual/Augmented reality, and telepresence will revolutionize the way we interact and communicate in the future. Nowadays, the emergence of large language models (LLMs) has dramatically changed various aspects of our lives, including how we interact with computers by providing intelligent instructions or verbal assistance. Additionally, the COVID-19 pandemic and quarantines have made clear the urgent need for new and efficient ways to meet each other, communicate and solve tasks. Envision a future where intelligent virtual assistants or robots not only provide support in decision-making, similar to LLMs, but also are embodied in highly realistic avatars and engage in meaningful interactions with their human counterparts, objects, or environments.

Enabling this vision requires virtual avatars that look and behave like real humans. While photorealistic avatars are widely studied, modeling the realistic

motion of avatars, especially when interacting with objects and environments, has barely been explored. Humans are constantly moving, and their motions are mostly goal-oriented; we move by walking on the ground, sleep lying on a bed, rest sitting on a chair, and work using touchscreens and keyboards. We exploit the affordances of the natural environment, and we design objects to better "afford" our bodies. Human-object interaction, encompassing full-body motion, hand-object grasping, and object manipulation, lies at the core of how humans execute tasks to achieve their goals and represent the complex and diverse nature of human behavior. Therefore, accurate modeling of these interactions would enable us to simulate avatars to perform tasks in Augmented and Virtual Reality, enhance animation realism, and develop applications that can better perceive and respond to human behavior.

1.2 Problem Statement

Although seemingly trivial for adult humans, generating realistic hand-object interactions for avatars is a complex challenge. For example, think of how we drink from a cup in real life. We walk towards the object with our feet contacting the floor, orient our head to look at the object, lean our torso and extend our arms to reach it, dexterously pose our hands to establish fine contact and grasp it, bring it to our mouth, making contact with the lips, and finally, we tilt the head to drink. Grasping of the same object by the same person will vary depending on the relative "starting" position of the hand with respect to the object and for different purposes like lifting the object, passing it to another person, washing it, or drinking from it. Humans are able to gracefully execute these steps, yet these are challenging and involve motion planning, motor control, and spatial awareness. Therefore, generating such motions is challenging for computers, because they need to represent and perceive the 3D shape of the object, the 3D shape and anatomy of the human hand, the goal of the task, the spatial 3D hand-object relationship, and the object's affordances (e.g., a knife's handle is graspable; the blade is not).

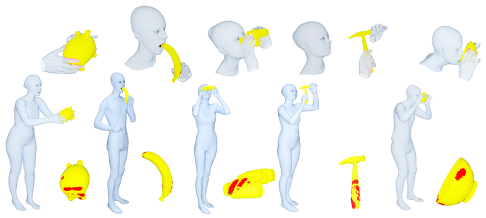
Prior work has made some progress in this direction, but some limitations remain to be addressed. First, simulating accurate human-object interactions requires a strong model of such interactions, and learning this model requires data. However,

capturing such data is not simple and existing datasets often lack the necessary detail of the body, head, hands, and object motion and the contact between them [1–6]. Grasping involves both gross and subtle motions, as humans involve their whole body and dexterous finger motion to manipulate objects. Therefore, objects contact multiple body parts and not just the hands. This is difficult to capture with images because the regions of contact are occluded [7–9]. Pressure sensors or other physical instrumentation, however, are also not a full solution as they can impair natural human-object interaction and do not capture accurate full-body motion [10–13]. Consequently, there are no existing datasets of complex human-object interaction that contain full-body motion, 3D body shape, and detailed body-object contact. Second, traditional methods for human motion generation have mainly focused on individual aspects of interaction [6, 14–27]. Some work has focused only on bodies or hands in "isolation", with no scene or object context [24–27]. Other work focuses on bodies interacting with scenes but ignores the hands and head [28–34]. Similarly, work on generating hand grasps often ignores the body and is mainly focused on static grasps [35–42]. However, these are just parts of the problem. What we really need, instead, is to generate the motion of full-body avatars grasping objects, by jointly considering the body, head, hands, and the object, and the holistic interplay among these components.

1.3 Contributions

This research presents significant contributions to the field of human-object interaction modeling through the development of novel datasets, methods, and algorithms. These contributions address the current limitations and provide comprehensive solutions for generating realistic motion of full-body avatars grasping and interacting with objects. The main contributions are threefold, addressing the need for comprehensive datasets, incorporating whole-body motion during an interaction, and generating intricate hand-object interactions throughout the manipulation. Each part is organized into a chapter, detailing the advancements made in the specific area.

1.3.1 GRAB: A Novel Dataset for Human-Object Interaction

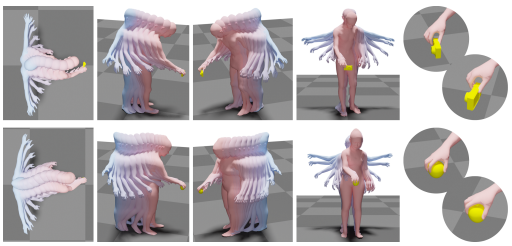


The first major contribution of this research is the introduction of GRAB, a novel dataset that captures complex and rich human-object interactions in 3D. Most previous work focuses on prehensile "grasps" [1, 2, 8, 43, 44], where a single human hand is stably lifting or using an object. However, human grasping and using everyday objects involve the whole body, are fundamentally three-dimensional, and contact occurs between objects and "multiple" body parts. Such whole-body grasping [4] has received much less attention [3, 16] than single-hand object grasping. To model this, a dataset of humans interacting with varied objects is needed, capturing the 3D surface of both the whole body and objects.

We address the limitations of the previous datasets in GRAB, by using state-of-the-art tools and methods. First, we use a high-quality motion capture system to overcome the occlusion challenge, accurately capturing the motion of markers placed on the body and object surfaces. Our precisely designed marker layouts on the body and objects help prevent occlusions and accurately track objects and detailed *whole body* motions. Next, we adapt MoSh++ [45] and extend it to reconstruct facial motion in addition to the 3D shape and motion of the body and hands from the tracked Motion Capture (MoCap) markers. For increased accuracy, we capture a 3D scan of each participant, fit the SMPL-X body model [5] to it, and use the resulting body shape during the MoSh++ process. As a result, we obtain detailed 3D meshes for both the object and the human (with a full body, articulated fingers, and face) moving over time while in interaction. Using these meshes, we then infer the accurate body-object contact.

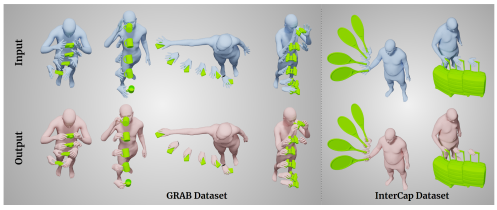
GRAB fills the data gap, as it is the first human-object interaction dataset containing the 3D body and object motion, accurate hand-object interaction, head motion, and detailed contact. This is a unique dataset, that goes well beyond existing ones for modeling and understanding how humans grasp and manipulate objects, how their full body is involved, and how interaction varies with the task. Subsequently, to show the value of our dataset for machine learning, we use the dataset to train GrabNet, which generates state-of-the-art static hand grasps for unseen 3D objects.

1.3.2 GOAL: Generating Avatar Motion for Grasping Objects



Although GrabNet generates accurate hand-object grasps, they are still static and do not involve the whole body and head pose for grasping. The initial step of "walking" towards an object and "grasping" it, is a fundamental and common aspect of many object-interaction motions, and is essential for realistic avatar motion. Some prior work focuses on generating walking motions [24, 25, 27, 46], however, none of them consider grasping an object or interacting with it. Therefore, we introduce GOAL, the first model that takes a 3D object, its position, and a starting 3D body pose and shape as input, and generates avatars that walk and grasp the objects with realistic body pose, head orientation, hand grasp, and foot-ground contact. Prior work only generates SMPL-X body pose parameters [24] to represent the motions. Here, we exploit complementary grasp representations by additionally regressing vertex offsets between the body and object, and a head direction vector. Then, we use them in an optimization step to further improve the grasp quality, and body and head pose. Moreover, we introduce a novel interaction-aware feature that transforms body-to-object distance into a richer representation, which helps the networks to better localize the object with respect to the body. This leads to generating smooth motions without noticeable foot sliding.

1.3.3 GRIP: Interaction Model for Hand Motion Synthesis



GOAL generates the avatar motion only up to the grasping of the objects. For an authentic virtual avatar motion, avatars should not only walk and grasp objects but also have intricate hand-object interactions afterward, including subtle finger motion and accurate grasps throughout the interaction with objects. While seemingly trivial for humans, this involves satisfying numerous semantic and physical constraints, making it a complex challenge to address.

With GRIP, we bridge this gap by introducing a learning-based interaction model that takes the 3D motion of the object and the body without finger articulations, and synthesizes realistic motion for both left and right hands before, during, and after object manipulation. GRIP leverages the spatio-temporal information between the body and the object to extract rich temporal interaction cues that help generalization to new objects. Moreover, to generate a consistent motion between frames, it introduces a motion temporal consistency constraint applied in the latent space. This leverages a shared decoder, which regulates motion inconsistency and leads to generating consistent interaction motions. Overall, GRIP "upgrades" sequences of body and object motion without finger articulation to include detailed hand-object interaction and further enhances the realism and applicability of digital human modeling.

1.4 Summary

In conclusion, this research marks a significant stride in the field of digital human modeling, particularly in the domain of human-object interactions. We first introduce the GRAB dataset, which provides a rich dataset to study human-object interaction motion, containing the 3D body and object motion, accurate hand-object interaction, head motion, and detailed contact. Utilizing this dataset,

we train GrabNet, a generative model that predicts hand grasps given the 3D object shape. However, GrabNet generates static grasps and focuses solely on the hands. To overcome these limitations, we introduce GOAL, a model that generates human *whole-body* motion for walking and grasping 3D objects. Directly generating SMPL-X parameters to represent full-body grasps leads to inaccurate hand-object contacts. To improve the grasp quality, GOAL regresses complementary grasp representations in addition to the SMPL-X pose parameters, namely vertex offsets between the body and object, and a head direction vector. Furthermore, GOAL employs an interaction-aware feature that transforms body-to-object distance into a richer representation, producing smooth and realistic motion. Despite generating realistic motions, GOAL's motions stop after grasping the objects and do not model the finger motions and accurate grasps throughout the interaction with objects. Finally, to tackle this, we propose GRIP, which takes the 3D motion of the body and object and synthesizes realistic motion for both left and right hands before, during, and after object manipulation. GRIP uses a latent temporal consistency and novel spatio-temporal feature extractors that help generate smooth and accurate motions, further enhancing the realism and applicability of digital human modeling.

However, the significance of this work lies not just in these immediate contributions but in the opportunities it may create for future research and development. By beginning the development of these foundational datasets and models, we have created a starting point for subsequent exploration and innovation. The applications of this research may extend beyond academia, with potential implications for various fields including telepresence, augmented reality, and assistive technologies.

Ultimately, this research offers a step toward a future where virtual avatars not only appear more lifelike but also interact with a level of complexity that reflects human behavior. The progress made in this thesis may help to lay the groundwork for a new phase in human-computer interaction, where the boundaries of realism are gradually explored, making our interactions in the digital world more immersive and intuitive.

1.5 Thesis Organization

This thesis is organized into several chapters, each focusing on a distinct aspect of understanding and modeling human-object interaction motions in 3D. The structure of the thesis is as follows:

- **Chapter 2: GRAB**: This chapter presents the design, creation, and evaluation of the GRAB dataset, which captures complex and rich human-object interactions in 3D. We detail the methodology used to obtain accurate motion capture data for both the human body and objects, as well as the resulting dataset's unique features and applications.
- **Chapter 3: GrabNet**: Building on the GRAB dataset, this chapter introduces GrabNet, a generative model that predicts hand grasps given a 3D object shape. We describe the model's architecture, training process, and results, demonstrating its ability to generate state-of-the-art static hand grasps for unseen 3D objects.
- **Chapter 4: GOAL**: In this chapter, we present GOAL, a model that generates whole-body motion for walking and grasping 3D objects. We outline the model's architecture, novel features for improving grasp quality and generating smooth motion, and the experimental evaluation of the generated motion sequences.
- **Chapter 5: GRIP**: This chapter focuses on GRIP, a learning-based interaction model that synthesizes realistic motion for both hands before, during, and after object manipulation, given the 3D motion of the body and object. We discuss the model's architecture, the novel spatio-temporal extractors and latent temporal consistency, and the resulting improvements in motion quality and realism.
- **Chapter 6: Conclusion**: This final chapter summarizes the key findings and contributions of the thesis, highlighting the novel approaches to understanding and modeling human-object interactions in 3D. We reflect on the implications of the work, discuss potential future directions, and conclude with a review of the thesis's broader impact on the field of human-object interaction research.

- **Appendices**: The appendices contain supplementary material that supports the main content of the thesis. This includes mathematical derivations, additional experimental results, technical specifications, and other relevant information that provides further insights into the methods and concepts discussed in the main chapters.

Data is a precious thing and will last longer than the systems themselves.

— Tim Berners-Lee

2

GRAB: The 4D Human-Object Interaction Dataset

Contents

2.1 Overview

Training computers to understand, model, and synthesize human grasping requires a rich dataset containing complex 3D object shapes, detailed contact information, hand pose and shape, and the 3D body motion over time. While "grasping" is commonly thought of as a single hand stably lifting an object, we capture the motion of the entire body and adopt the generalized notion of "whole-body grasps". To achieve this, we collect a new dataset, called *GRAB* (GRasping Actions with Bodies), of whole-body grasps, containing full 3D shape and pose sequences of 10 subjects interacting with 51 everyday objects of varying shape and size. Given MoCap markers, we fit the full 3D body shape and pose, including the articulated face and hands, as well as the 3D object pose. This gives detailed 3D meshes over time, from which we compute contact between the body and object. This is a unique dataset, that goes well beyond existing ones for modeling and understanding how humans grasp and manipulate objects, how their full body is involved, and how interaction varies with the task. The dataset and code are available for research purposes at https://grab.is.tue.mpg.de.

2.2 Introduction

A key goal of computer vision is to estimate human-object interactions from video to help understand human behavior. Doing so requires a strong model of such interactions and learning this model requires data. However, collecting such data is not straightforward due to the broad and nuanced human motions during the grasping actions such as the entire body and intricate finger movements to handle objects. Therefore, in addition to hands, contact happens between the rest of the body and the objects. Additionally, capturing these contacts is challenging from the images or other sensors. Consequently, there are no existing datasets of complex human-object interaction that contain full-body motion, 3D body shape, and detailed body-object contact. To fill this gap, we capture a novel dataset of full-body 3D humans dynamically interacting with 3D objects as illustrated in Fig.

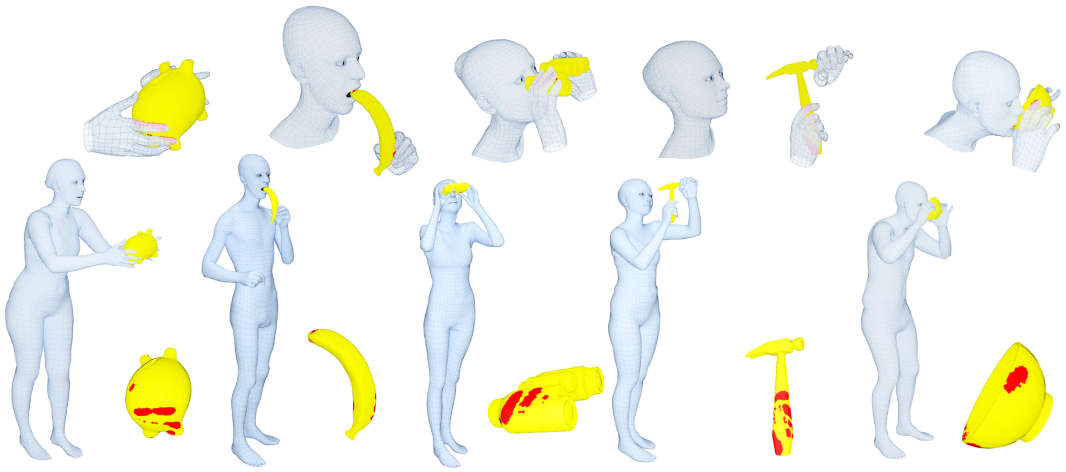


Figure 2.1: Example "whole-body grasps" from the GRAB dataset. A "grasp" is usually thought of as a single hand interacting with an object. Using objects, however, may involve more than just a single hand. From left to right: (i) passing a piggy bank, (ii) eating a banana, (iii) looking through binoculars, (iv) using a hammer, (v) drinking from a bowl. Contact between the object and the body is shown in red on the object; here contact areas are spatially extended to aid visualization.

2.1. By accurately tracking 3D body and object shape, we reason about contact resulting in a dataset with detail and richness beyond existing grasping datasets.

Most previous work focuses on prehensile "grasps" [43]; i.e. a single human hand stably lifting or using an object. The hands, however, are only part of the story. For example, as infants, our earliest grasps involve bringing objects to the mouth [47]. Consider the example of drinking from a bowl in Fig. 2.1 (right). To do so, we must pose our body so that we can reach the bowl, we orient our head to see it, we move our arm and hand to stably lift it, and then we bring it to our mouth, making contact with the lips, and finally we tilt the head to drink. As this and other examples in the figure illustrate, human grasping and use of everyday objects involves the *whole body*. Such interactions are fundamentally *three-dimensional*, and contact occurs between objects and multiple body parts.

Dataset: Such whole-body grasping [4] has received much less attention [3, 16] than single hand-object grasping [1, 2, 8, 43, 44]. To model such grasping we need a dataset of humans interacting with varied objects, capturing the full 3D surface of both the body and objects. To solve this problem we adapt recent motion capture techniques, to construct a new rich dataset called **GRAB** for "*GRasping Actions with Bodies.*" Specifically, we adapt MoSh++ [45] in two ways. First, MoSh++

estimates the 3D shape and motion of the body and hands from MoCap markers; here we extend this to include facial motion. For increased accuracy, we first capture a 3D scan of each subject and fit the SMPL-X body model [5] to it. Then MoSh++ is used to recover the pose of the body, hands, and face. Note that the face is important because it is involved in many interactions; see in Fig. 2.1 (second from left) how the mouth opens to eat a banana. Second, we also accurately capture the motion of 3D objects as they are being manipulated by the subjects. To this end, we use small hemispherical markers on the objects and show that these do not impact grasping behavior. As a result, we obtain detailed 3D meshes for both the object and the human (with a full body, articulated fingers, and face) moving over time while in interaction, as shown in Fig. 2.1. Using these meshes we then infer the body-object contact (red regions in Fig. 2.1). Unlike [13], this gives both the contact and the full body/hand pose over time. Interaction is dynamic, including in-hand manipulation and re-grasping. GRAB captures 10 different people (5 male and 5 female) interacting with 51 objects from [13]. Interaction takes place in 4 different contexts: lifting, handing over, passing from one hand to the other, and using, depending on the affordances and functionality of the object.

Applications: GRAB supports multiple uses of interest to the community. First, we show how GRAB can be used to gain insights into hand-object contact in everyday scenarios. This includes studying the affordances of objects, contact heatmaps, and the important areas on the body and hands for interaction. Second, there is significant interest in training models to grasp 3D objects [48]. Thus, we use GRAB to train a conditional variational autoencoder (cVAE) to generate plausible grasps for unseen 3D objects (Chapter 3). Given a randomly posed 3D object, we predict plausible hand parameters (wrist pose and finger articulation) appropriate for grasping the object. Then, by conditioning on a new 3D object shape, we sample from the learned latent space and generate hand grasps for this object. We evaluate both quantitatively and qualitatively the resulting grasps and show that they look natural.

In summary, this work makes the following contributions: (1) we introduce a unique dataset capturing real "whole-body grasps" of 3D objects, including full-body human motion, object motion, in-hand manipulation and re-grasps; (2) to capture this, we adapt MoSh++ to solve for the body, face, and hands of SMPL-X to obtain

detailed moving 3D meshes of body and objects; (3) using these meshes and tracked 3D objects we compute plausible contact between the object and the human and provide an analysis of observed patterns; (4) in Chapter 3 we show the value of our dataset for machine learning, by training a novel conditional neural network to generate 3D hand grasps for unseen 3D objects. The dataset, models, and code are available for research purposes at https://grab.is.tue.mpg.de.

2.3 Related Work

Despite progress in understanding how humans interact with objects, existing datasets, models, and methodologies reveal significant gaps and limitations. Studies have mostly focused on hand grasps, with less attention given to whole-body interactions. It is challenging to capture these complex interactions, especially when contact is involved. While there have been attempts to address these issues, they often fail to fully capture the complexity of real-world interactions. This section reviews the current research and highlights the strengths and weaknesses of prior work.

2.3.1 Grasp Definition

Hand Grasps: Hands are crucial for grasping and manipulating objects. For this reason, many studies focus on understanding grasps and defining taxonomies [1, 2, 8, 10, 43, 44]. These works have explored the object shape and purpose of grasps [1], contact areas on the hand captured by sinking objects in ink [44], pose and contact areas [8] captured with an integrated data-glove [11] and tactile-glove [10], or number of fingers in contact with the object and thumb position [2]. A key element for these studies is capturing accurate hand poses, relative hand-object configurations and contact areas.

Whole-Body Grasps: Often people use more than a single hand to interact with objects. However, there are not many works in the literature on this topic [3, 4]. Borras et al. [3] use MoCap data [20] of people interacting with a scene with multi-contact, and present a body pose taxonomy for such whole-body grasps. Hsiao et al. [4] focus on imitation learning with a database of whole-body grasp

demonstrations with a human teleoperating a simulated robot. Although these works go into the right direction, they use unrealistic humanoid models and simple objects [3, 4] or synthetic ones [4]. Instead, we use the SMPL-X model [5] to capture "whole-body", face and dexterous in-hand interactions.

2.3.2 Capturing Interactions

Interaction-Motion Tracking: MoCap is often used to capture, synthesize or evaluate humans interacting with scenes. Lee et al. [30] capture a 3D body skeleton interacting with a 3D scene and show how to synthesize new motions in new scenes. Wang et al. [49] capture a 3D body skeleton interacting with a large geometric objects. Han et al. [50] present a method for automatic labeling of hand markers, to speed up hand tracking for VR. Le et al. [51] capture a hand interacting with a phone to study the "comfortable areas", while Feit et al. [52] capture two hands interacting with a keyboard to study typing patterns. Other works [38, 39] focus on graphics applications. Kry et al. [39] capture a hand interacting with a 3D shape primitive, instrumented with a force sensor. Pollard et al. [38] capture the motion of a hand to learn a controller for physically based grasping. Mandery et al. [20] sit between the above works, capturing humans interacting with both big and handheld objects, but without articulated faces and fingers. None of the previous work captures full 3D bodies, hands and faces together with 3D object manipulation and contact.

Capturing Contact: Capturing human-object contact is hard because the human and object heavily occlude each other. One approach is instrumentation with touch and pressure sensors, but this might bias natural grasps. Pham et al. [7] predefine contact points on objects to place force transducers. More recent advances in tactile sensors allow accurate recognition of tactile patterns and handheld objects [53]. Some approaches [8] use a data glove [11] with an embedded tactile glove [10, 54] but this combination is complicated and the two modalities can be hard to synchronize. A microscopic-domain tactile sensor [55] is introduced in [9], but is not easy to use on human hands. Mascaro et al. [12] attach a minimally invasive camera to detect changes in the coloration of fingernails due to pressure. Brahmbhatt et al. [13] use a thermal camera to directly observe the "thermal print" of a hand on the grasped object. However, for this they only

capture static grasps that last long enough for heat transfer. Consequently, even recent datasets that capture realistic hand-object [56, 57] or body-scene [6, 19] interaction avoid directly measuring contact.

2.3.3 3D Interaction Models for Data Collection

Learning a model of human-object interactions is useful for graphics and robotics to help avatars [36, 37] or robots [58] interact with their surroundings, and for vision [59, 60] to help reconstruct interactions from ambiguous data. However, there is a chicken-and-egg problem; to capture or synthesize data to learn a model, one needs such a model in the first place. For this reason, the community has long used hand-crafted approaches that exploit contact and physics, for body-scene [6, 14, 61–63], body-object [15, 16, 18], or hand-object [17, 35, 40, 64–70] scenarios. These approaches compute contact approximately; this contact may be rough for humans modeled as 3D skeletons [16, 18] or shape primitives [14, 15, 66], or relatively accurate when using 3D meshes, whether generic [35, 67], personalized [61, 62, 68, 69], or based on 3D statistical models [6, 17].

Hand Grasps: To collect training data, several works [35, 65] use synthetic Poser [71] hand models, manually articulated to grasp 3D shape primitives. Contact points and forces are also annotated [35] through proximity and inter-penetration of 3D meshes. In contrast, Hasson et al. [17] use the robotics method GraspIt [72] to automatically generate 3D MANO [73] grasps for ShapeNet [74] objects and render synthetic images of the hand-object interaction. However, GraspIt optimizes for hand-crafted grasp metrics that do not necessarily reflect the distribution of human grasps (see Sup. Mat. Sec. C.2 of [17], and [75]). Alternatively, Garcia-Hernando et al. [56] use magnetic sensors to reconstruct a 3D hand skeleton and rigid object poses; they capture 6 subjects interacting with 4 objects. This dataset is used by [76, 77] to learn to estimate 3D hand and object poses, but suffers from noisy poses and significant inter-penetrations (see Sec. 5.2 of [17]).

Body Grasps: For bodies, Kim et al. [18] use synthetic data to learn to detect contact points on a 3D object, and then fit an interacting 3D body skeleton to them. Savva et al. [19] use RGB-D to capture 3D body skeletons of 5 subjects interacting in 30 3D scenes, to learn to synthesize interactions [19], affordance detection [78],

or to reconstruct interaction from videos [60]. Mandery et al. [20] use optical MoCap to capture 43 subjects interacting with 41 tracked objects, both large and small. This is similar to our effort but they do not capture fingers or 3D body shape, so cannot reason about contact. Corona et al. [79] use this dataset to learn context-aware body motion prediction. Starke et al. [37] use Xsens IMU sensors [80] to capture the main body of a subject interacting with large objects, and learn to synthesize avatar motion in virtual worlds. Hassan et al. [6] use RGB-D and 3D scene constraints to capture 20 humans as SMPL-X [5] meshes interacting with 12 static 3D scenes, but do not capture object manipulation. Zhang et al. [81] use this data to learn to generate 3D scene-aware humans.

We see that only parts of our problem have been studied. We draw inspiration from prior work, in particular [4, 6, 13, 20]. We go beyond these by introducing a new dataset of real "whole-body" grasps, as described in the next section.

2.4 Dataset

To manipulate an object, the human needs to approach its 3D surface and bring their skin to come in *physical contact* to apply forces. Realistically capturing such human-object interactions, especially with "whole-body grasps", is a challenging problem. First, the object may occlude the body and vice-versa, resulting in *ambiguous* observations. Second, for physical interactions, it is crucial to reconstruct an accurate and detailed 3D *surface* for both the human and the object. Additionally, the capture has to work across multiple scales (body, fingers, and face) and for objects of varying complexity. We address these challenges with a unique combination of state-of-the-art solutions that we adapt to the problem.

There is a fundamental trade-off with current technology; one has to choose between (a) accurate motion with instrumentation and without natural RGB images, or (b) less accurate motion but with RGB images. Here we take the former approach; for an extensive discussion please refer to Sec. 2.5.4

Figure 2.2: MoCap markers used to capture the motion of 3D objects. Example 3D printed objects from [13]. We glue 1.5 mm radius hemi-spherical markers (the gray dots) on the objects. These makers are small enough to be unobtrusive. The 6 objects on the right are mostly used by one or more hands, while the 6 on the left involve "whole-body grasps".

2.4.1 Human Body Model

We model the human with the SMPL-X [5] 3D body model. SMPL-X jointly models the body with an articulated face and fingers; this expressive body model is critical to capture physical interactions. More formally, SMPL-X is a differentiable function $M_b(\beta, \theta, \psi, \gamma)$ that is parameterized by body shape β, pose θ, facial expression ψ and translation γ. The output is a 3D mesh $M_b = (V_b, F_b)$ with $N_b = 10475$ vertices $V_b \in \mathbb{R}^{(N_b \times 3)}$ and triangles $F_b = 20,908$. The shape parameters $\beta \in \mathbb{R}^{10}$ are coefficients in a learned low-dimensional linear shape space, created via PCA on 3D meshes of roughly $4,000$ different people [82]. This lets SMPL-X represent different subject identities with the same mesh topology. The 3D joints, $J(\beta)$, of a kinematic skeleton are regressed from the body shaped defined by β. The skeleton

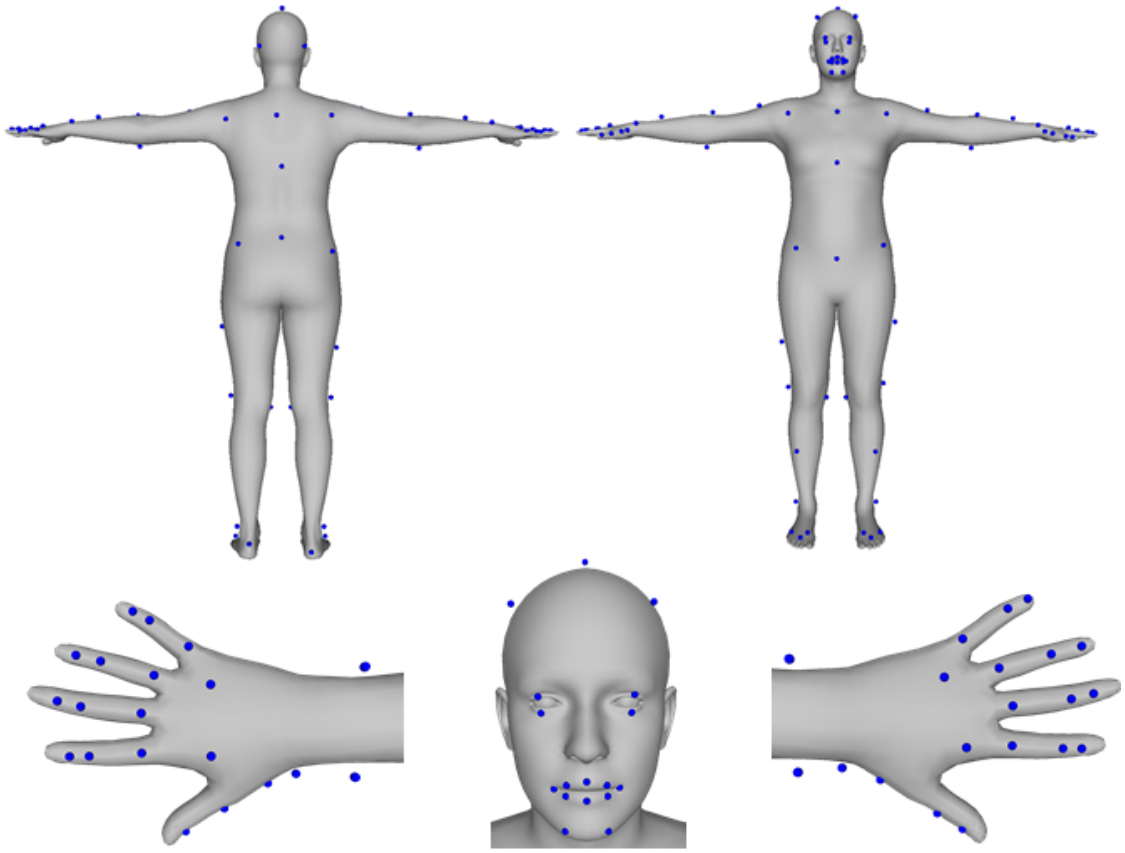


Figure 2.3: MoCap markers used to capture human motion. We attach 99 reflective markers per subject; 49 for the body, 14 for the face and 36 for the fingers. We use spherical 4.5 mm radius markers for the body and hemi-spherical 1.5 mm radius ones for the hands and face.

has 55 joints in total; 22 for the body, 15 joints per hand for finger articulation, and 3 for the neck and eyes. Corrective blend shapes are added to the body shape and then posed body is defined by linear blend skinning with this underlying skeleton. The overall pose parameters $\theta = (\theta_b, \theta_f, \theta_h)$ are comprised of $\theta_b \in \mathbb{R}^{66}$ and $\theta_f \in \mathbb{R}^9$ parameters in axis-angle representation for the main body and face joints correspondingly, with 3 degrees of freedom (DoF) per joint, and $\theta_h \in \mathbb{R}^{60}$ parameters in a lower-dimensional pose space for both hands, i.e. 30 DoF per hand following [17]. For more details, please see [5].

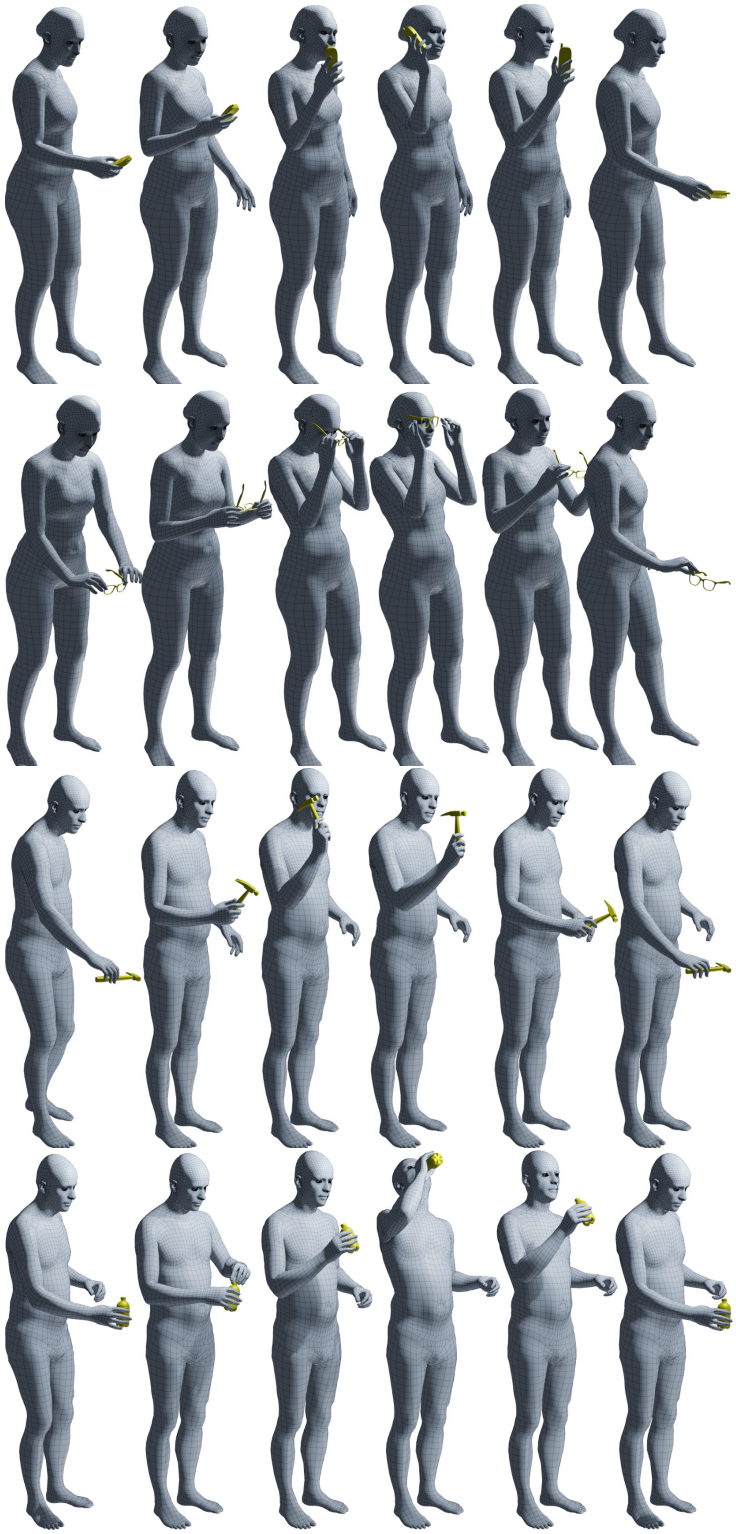


Figure 2.4: We capture humans interacting with objects over time and reconstruct sequences of 3D meshes for both, as described in Sec. 2.4.2 and Sec. 2.4.3. Note the realistic and plausible placement of objects in the hands, and the "whole-body" involvement.

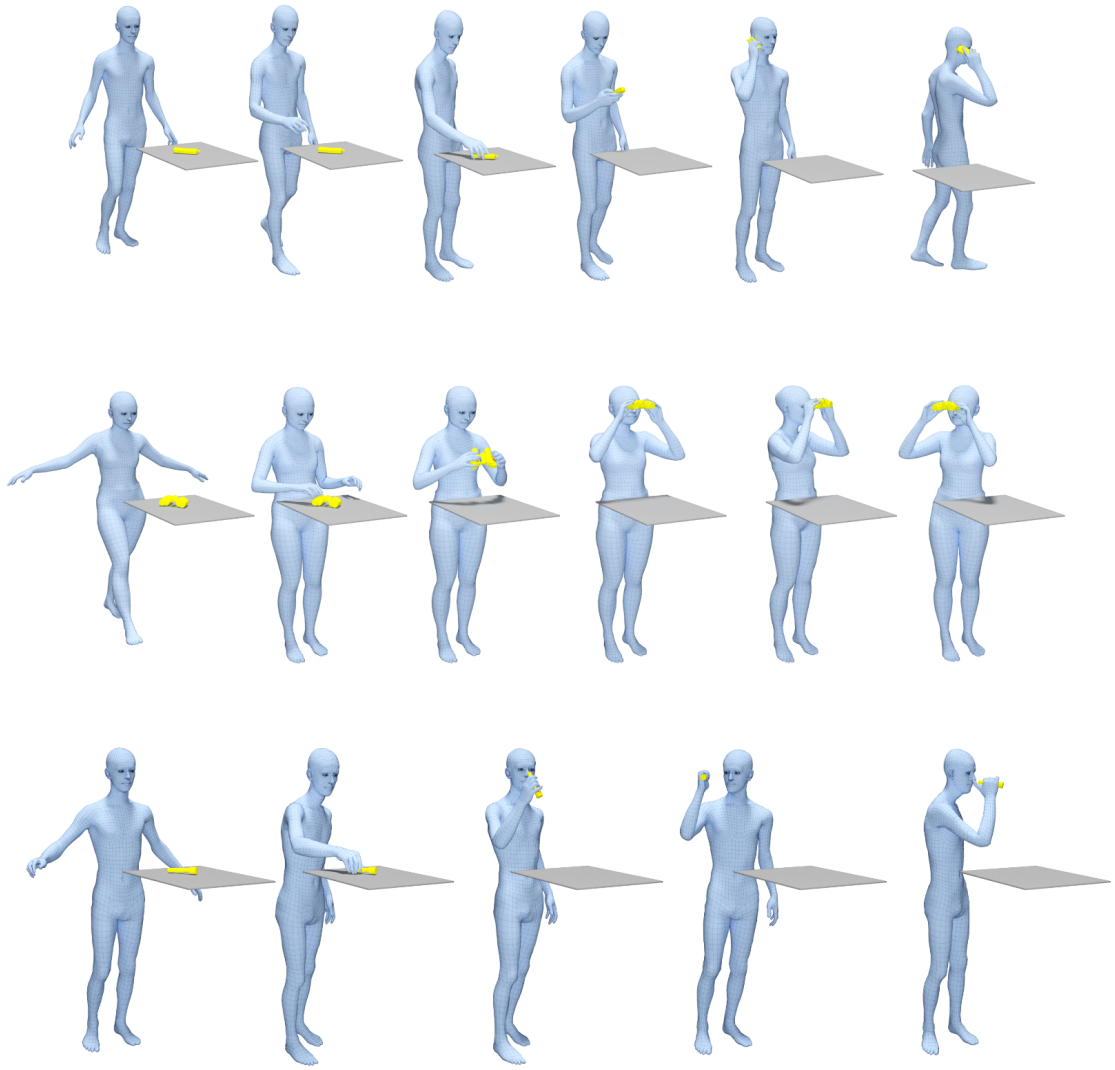


Figure 2.5: We capture humans interacting with objects over time and reconstruct sequences of 3D meshes for both. Here we show 3 sequences for "talking on the phone", "using binoculars", and "using a flashlight". Note the realistic and plausible hand-object interaction and the "whole-body" involvement.

2.4.2 Motion Capture (MoCap)

We use a Vicon system with 54 infrared "Vantage 16" [83] cameras that capture 16 MP at 120 fps. The large number of cameras minimizes occlusions and the high frame rate captures temporal details of contact. The high resolution allows small (1.5 mm radius) hemispherical markers. This minimizes their influence on finger and face motion and does not alter how people grasp objects. Details of the

marker setup are shown in Fig. 2.2 and 2.3. Even with many cameras, motion capture of the body, face, and hands, together with objects, is uncommon because it is so challenging. MoCap, markers become occluded, labels are swapped, and ghost makers appear. MoCap cleaning was done by four trained technicians using Vicon's Shōgun-Post software.

Capturing Human MoCap: To capture human motion, we use the marker set of Fig. 2.3. The body markers are attached on a tight body suit with a velcro-based mounting at a distance of roughly $d_b = 9.5$ mm from the body surface. The hand and face markers are attached directly to the skin with special removable glue, therefore the distance to these is roughly $d_h = d_f \approx 0$ mm. Importantly, no hand glove is used and hand markers are placed only on the dorsal side, leaving the palmar side completely uninstrumented, for natural interactions.

Capturing Objects: To reconstruct interactions accurately, it is important to know the precise 3D object surface geometry. We, therefore, use the CAD object models of [13], and 3D print them with a Stratasys Fortus 360mc [84] printer; see Fig. 2.2. Each object o is then represented by a known 3D mesh with vertices V_o. To capture object motion, we attach on the 1.5 mm hemi-spherical markers with strong glue directly to the object surface. We use at least 8 markers per object, empirically distributing them on the object so that at least 3 of them are always observed. The size and placement of the markers makes them unobtrusive. In Appendix A we show empirical evidence that makers have minimal influence on grasping.

2.4.3 From MoCap Markers to 3D Surfaces

Model-Marker Correspondences: For the human body we define, a priori, the rough marker placement on the body as shown in Fig. 2.3. Exact marker locations on individual subjects are then computed automatically using MoSh++ [45]. In contrast to the body, the objects have different shapes and mesh topologies. Markers are placed according to the object shape, affordances and expected occlusions during interaction; Fig. 2.2, Therefore, we annotate object-specific vertex-marker correspondences and do this once per object.

Human and Object Tracking: To ensure accurate human shape, we capture a 3D scan of each subject and fit SMPL-X to it following [73]. We fit these

personalized SMPL-X models to our cleaned 3D marker observations using MoSh++ [45]. Specifically, we optimize over pose, θ, expressions, ψ, and translation, γ, while keeping the known shape, β, fixed. The weights of MoSh++ for the finger and face data terms are tuned on a synthetic dataset to have accurate motions (see Sec. 2.4.4).

Objects are simpler than humans because they are rigid and we know their exact 3D shape. Given three or more detected markers, we solve for the rigid object pose $\theta_o \in \mathbb{R}^6$. Here we track the human and object separately and on a per-frame basis. Figures 2.4 and Fig. 2.5 show representative motions from our dataset and demonstrate that our approach captures realistic interactions and reconstructs detailed 3D meshes for both the human and the object over time.

In Fig. 2.8 we show more qualitative results with closeups on the interacting parts to show the accuracy of the tracked motions.

2.4.4 Adapting MoSh++

We adapt MoSh++ [45] for capturing the whole body (including the hands and face). The human and object are tracked independently and on a per-frame basis, for simplicity. We make two small changes to MoSh++. First, we use the ground-truth body shape, obtained from a 3D scan. Consequently, we do not use MoSh++ to estimate body shape. Second, we extend MoSh++ to estimate the parameters of the SMPL-X body model. This means extending it to capture facial pose and expression parameters. Additionally, we estimate the rigid 6 DoF pose of the objects using their known shape and the detected markers.

To adapt MoSh++ to capture faces, we need to tune the parameters of the model. For this [45] follows a data-driven approach; they capture the SSM dataset with an optical MoCap system synchronized with a 3D body scanner and use the scans for computing a reconstruction quality metric. However, SSM has markers only on the main body, while also the fingers of the scans are very noisy.

Capturing such a dataset, with clear scan regions for both the body, the face and all fingers, as well as synced MoCap for them, is too challenging. Instead, we follow a more practical approach and create a synthetic dataset by animating SMPL-X and generating virtual markers on the moving meshes. To bridge the domain gap, we simulate noise for marker position and visibility; we randomly

Mosh++ version	MoSh Stage-I (mm)		MoSh Stage-II (mm)	
	mean ± std	median	mean ± std	median
Vanila	4.76 ± 1.03	4.55	5.59 ± 1.86	5.28
Our adapted	**3.09 ± 0.55**	**2.80**	**4.86 ± 1.83**	**4.48**

Table 2.1: Evaluation of MoSh++ on the synthetic dataset. We compare the vanilla [45] to our adapted version. For the first stage of MoSh++, Stage-I, we report the distance of the latent marker placement compared to ground-truth marker locations, and for the second stage of MoSh++, Stage-II, we report the average vertex-to-vertex error between estimated and ground-truth meshes.

add 3D Gaussian noise with 1 mm variance in marker positions, as in [45], and randomly drop up to 5 markers per frame.

Unfortunately, there is no existing dataset with rich SMPL-X sequences. However, its model formulation is compatible to SMPL [85] for the body, FLAME [86] for the face, and MANO [73] for the hands. Therefore, we resort to datasets specific to each part to animate the body, face, and hands. For the *body*, we employ DFAUST [87] that captures 10 subjects performing 10 sequences each. We split the subjects into 6 for training and 4 for a withheld test set. We compute personalized SMPL-X mesh templates by registering the model to one scan per person as in [5, 73], and pose their body according to the registrations of DFAUST. For the *hand*, we employ the hand-only MANO model registrations of [73]. From the 1554 hand poses, we hold out 155 for the test set and use the rest for training. We then add hand motion to each body sequence by randomly choosing 15 hand poses and interpolating between them. For the *face*, we employ sequences of FLAME parameters from [88, 89]; the latter covers extreme facial expressions, while the former has everyday speaking expressions. We randomly choose 100 sequences from each dataset, splitting them into 60 for the training set and 40 for the withheld test set.

We use this dataset to set the weights following the approach in [45]. Table 2.1 compares a standard version of [45] with our adapted version on the synthetic test set for both stages of MoSh++. For the first stage of MoSh++ (Stage-I) we report the distance of the latent marker placement compared to ground-truth marker locations in mm. For this stage, we start from random marker placement guesses in the 1-ring neighborhood of the ground-truth locations. We repeat this three times with different random seeds for selecting 12 frames of MoSh++; see [45]. For the second stage (Stage-II), we use the optimized latent marker placements resulting

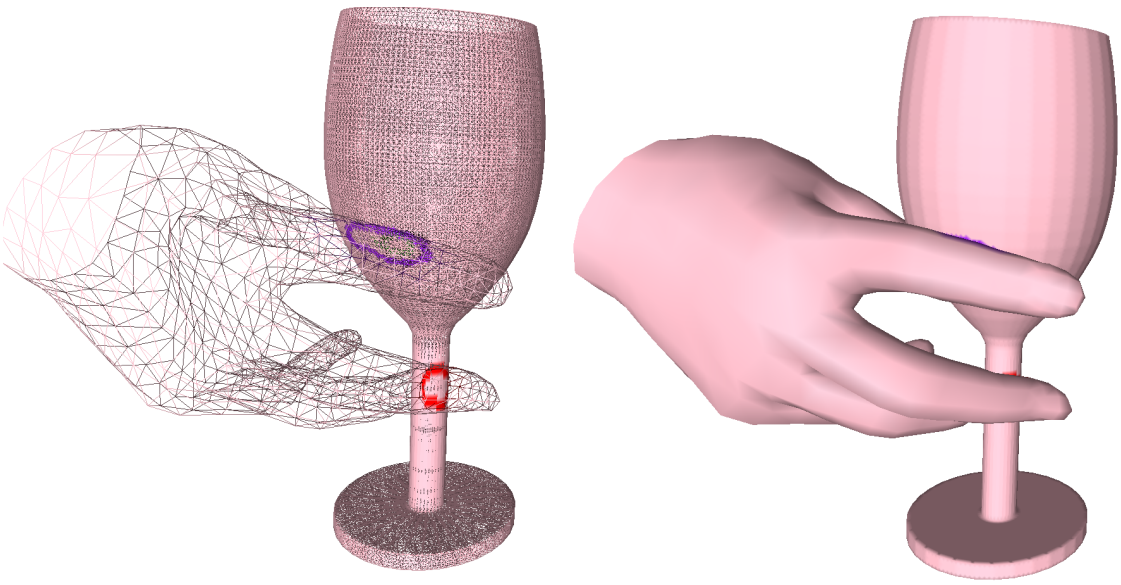


Figure 2.6: Detection of "intersection ring" triangles during contact annotation (Sec. 2.4.5).

from each random seed of the Stage-I and report the average vertex-to-vertex error between estimated and ground-truth meshes in mm. In each stage, we choose the weights that minimize the reported error. Our adapted version shows a clear improvement, by fitting the whole body, hands, and face, with weights λ tuned on our synthetic dataset. In contrast, [45] tunes only the body weights on their SSM dataset, it fits the hands with empirical weights and does not fit the face.

2.4.5 Contact Annotation

Since contact cannot be directly observed, we estimate it using 3D proximity between the 3D human and object meshes. In theory, they come in contact when the distance between them is zero. In practice, however, we relax this and define contact when the distance, $d \leq \varepsilon_{contact}$, for a threshold $\varepsilon_{contact}$. This helps address: (1) measurement and fitting errors, (2) limited mesh resolution, (3) the fact that human soft tissue deforms when grasping an object, while the SMPL-X model cannot model this.

Given these issues, accurately estimating contact is challenging. Consider the hand grasping a wine glass in Fig. 2.6 , where the color rings indicate intersections. Ideally, the glass should be in contact with the thumb, index and middle fingers. "Contact under-shooting" results in fingers hovering close to the object surface, but

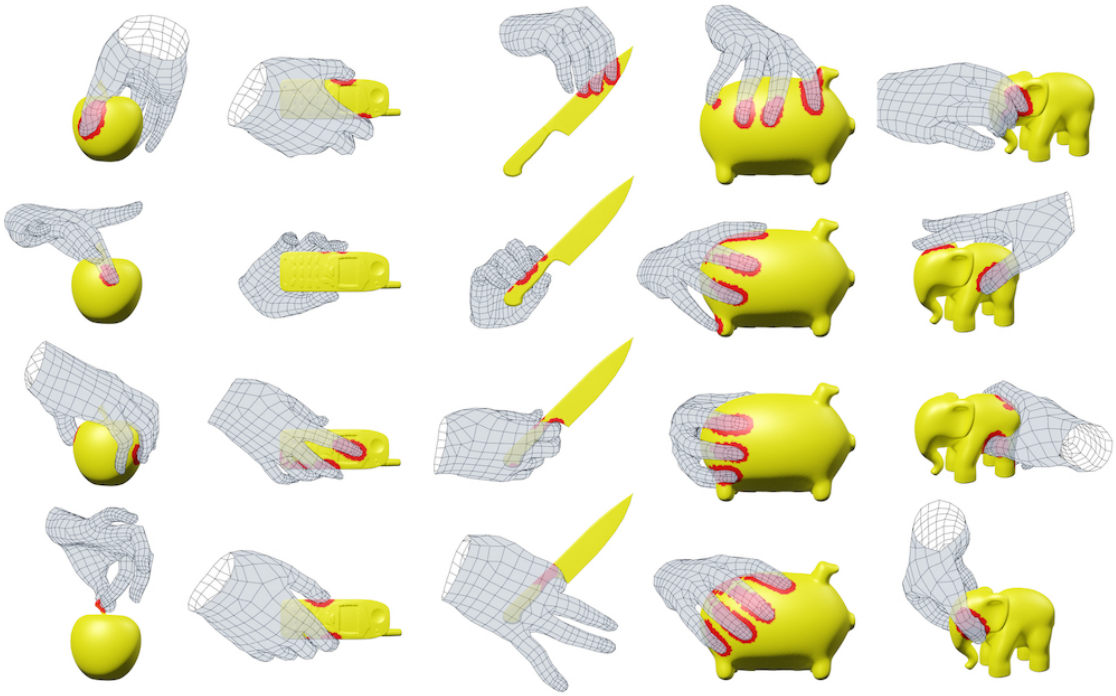


Figure 2.7: Accurate tracking lets us compute realistic contact areas (red) for each frame (Sec. 2.4.5). For illustration, we render only the hand of SMPL-X and spatially extend the red contact areas for visibility.

not on it, like the thumb. "Contact over-shooting", results in fingers penetrating the object surface around the contact area, like the index (purple intersections) and middle finger (red intersections). The latter case is especially problematic for thin objects where a penetrating finger can pass through the object, intersecting it on two sides. In this example, we want to annotate contact only with the outer surface of the object and not the inner one.

We account for "contact over-shooting" cases with an efficient heuristic. We use a fast method [5, 90] to detect intersections, cluster them in connected "intersection rings", $\mathcal{R}_b \subsetneq V_b$ and $\mathcal{R}_o \subsetneq V_o$, and label them with the intersecting body part, seen as purple and red rings in Fig. 2.6. The "intersection ring", $\mathcal{R}_b$, segments the body mesh M_b to give the "penetrating sub-mesh" $\mathcal{M}_b \subsetneq M_b$. (1) When a body part gives only one intersection, we annotate the points $V_o^{\mathcal{C}} \subset V_o$ on the object all vertices enclosed by the ring $\mathcal{R}_o$ as being in contact. We then annotate as contact points, $V_b^{\mathcal{C}} \subset V_b$, on the body all vertices that lie close to $V_o^{\mathcal{C}}$ with a distance $d_{o \to b} \leq \varepsilon_{contact}$. (2) In case of multiple intersections i we take into account only the ring $\mathcal{R}_b^i$ corresponding to the largest intersection subset, $\mathcal{M}_b^i$.

For body parts that are not found in contact above, there is the possibility of "contact under-shooting". To address this, we compute the distance from each object vertex V_o, to each non-intersecting body vertex V_b. We then annotate as contact vertices, $V_o^{\mathcal{C}}$ and $V_b^{\mathcal{C}}$, the ones with $d_{o \to b} \leq \varepsilon_{contact}$. We empirically find that $\varepsilon_{contact} = 4.5$ mm works well for our purposes.

In Fig. 2.7 we show examples of the computed contacts (shown in red) between the hand and object. Additionally, in Fig. 2.8 we show representative full-body interactions from GRAB with closeups of the interacting body parts and their contact areas (shown in red). Note that our dataset contains interactions between the objects and other body parts rather than only hands.

2.4.6 Dataset Protocol

Human-object interaction depends on various factors including the human body shape, object shape and affordances, object functionality, or interaction intent, to name a few. We, therefore, capture 10 people (5 men and 5 women), of various sizes and nationalities, interacting with the objects of [13]; see example objects in Fig. 2.2. All subjects gave informed written consent to share their data for research purposes.

For each object, we capture interactions with 4 different intents, namely "use" and "pass" (to someone), borrowed from [13], as well as "lift" and "off-hand pass" (from one hand to the other). Figure 2.4 shows some example 3D capture sequences for the "use" intent. For each sequence we: (i) we randomize initial object placement to increase motion variance, (ii) we instruct the subject to follow an intent, (iii) the subject starts from a T-pose and approaches the object, (iv) they perform the instructed task, and (v) they leave the object and return to a T-pose. In Fig. 2.9 we show examples of different intents captured in the GRAB dataset.

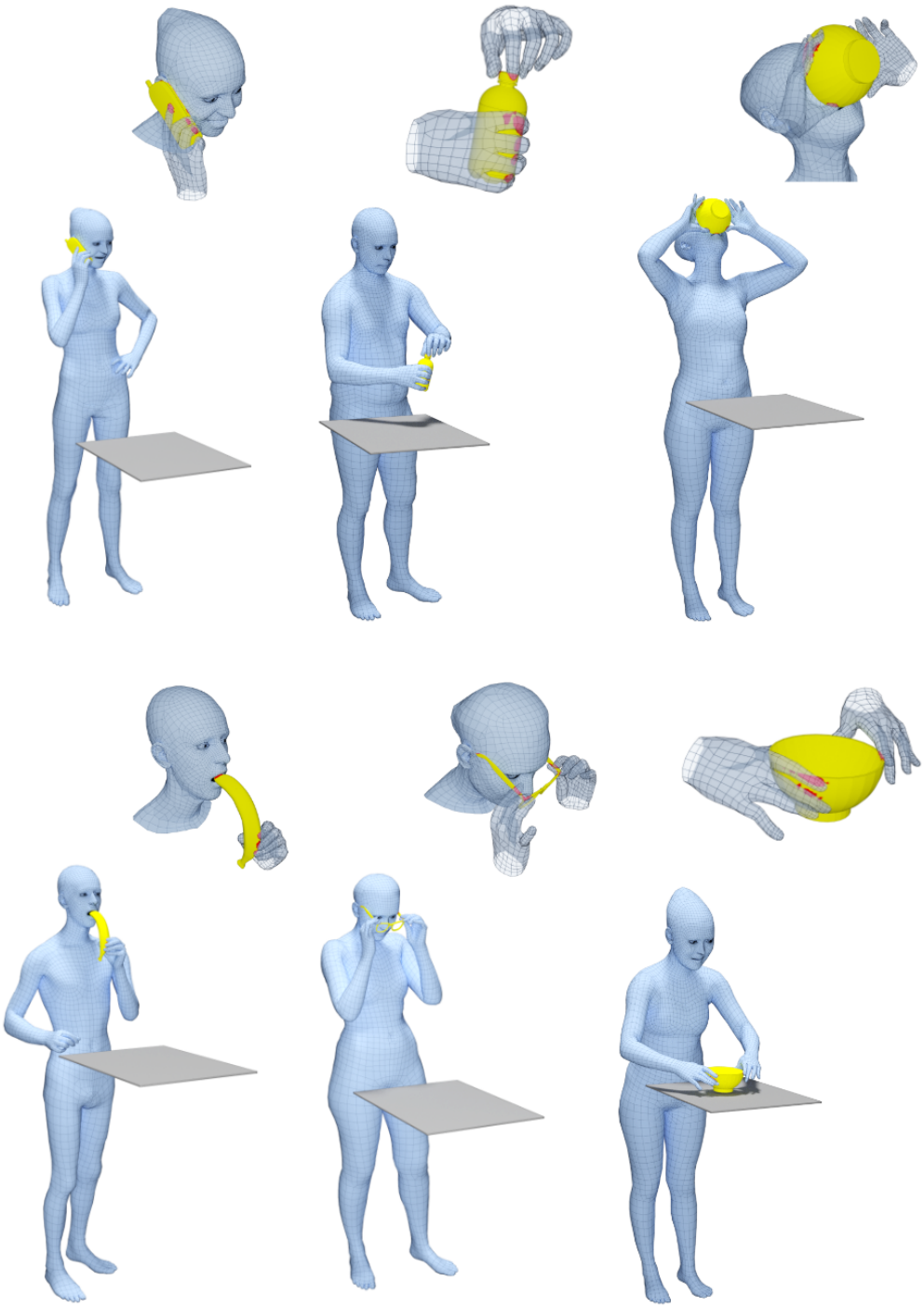


Figure 2.8: We show several full-body interaction poses from GRAB with closeups of the interacting parts. Note the accurate body, hand, and face motions during the interaction. Using the tracked 3D meshes we compute the contact areas between the body and objects as shown in red.

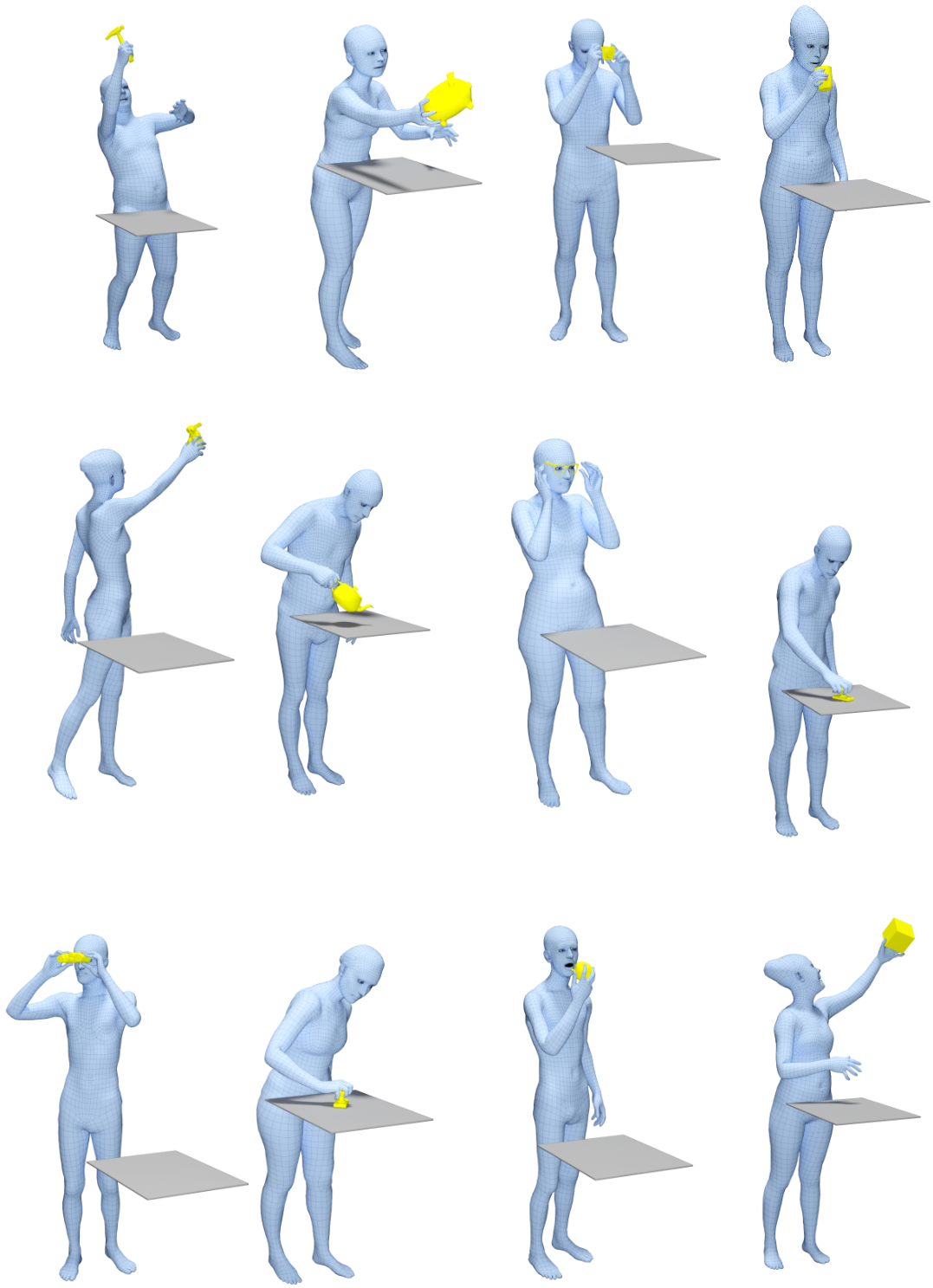


Figure 2.9: We show several examples of the 4 intents in our dataset, namely: "use", "pass", "lift", and "off-hand pass". The "use" intent changes based on the affordances of the object, as shown in the figure.

Table 2.2: Size of the GRAB dataset. GRAB is sufficiently large to enable training of data-driven models of grasping as shown in Chapter 3.

Intent	"Use"	"Pass"	"Lift"	"Off-hand"	Total
# Sequences	579	414	274	67	1334
# Frames	605.796	335.733	603.381	77.549	1.622.459

2.5 Analysis

2.5.1 Dataset Stats

The dataset contains 1334 sequences and over 1.6M frames of MoCap; Table 2.2 provides a detailed breakdown. Here we analyze those frames where we have detected contact between the human and the object. We assume that the object is static on a table and can move only due to grasping. Consequently, consider contact frames to be those in which the object's position deviates in the vertical direction by at least 5 mm from its initial position and in which at least 50 body vertices are in contact with the object. This results in $952,514$ contact frames that we analyze below. The exact thresholds of these contact heuristics have little influence on our analysis, see Sec. 2.5.3

By uniquely capturing the whole body, and not just the hand, interesting interaction patterns arise. By focusing on "use" sequences that highlight the object functionality, we observe that 92% of contact frames involve the right hand, 39% the left hand, 31% both hands, and 8% involve the head. For the first category the per-finger contact likelihood, from thumb to pinky, is 100%, 96%, 92%, 79%, 39% and for the palm 24%. For more results and the detailed percentage of the contact areas please see Appendix A.

2.5.2 Contact Heatmaps

To visualize the fine-grained contact information, we integrate over time the binary per-frame contact maps, and generate "heatmaps" encoding the contact likelihood

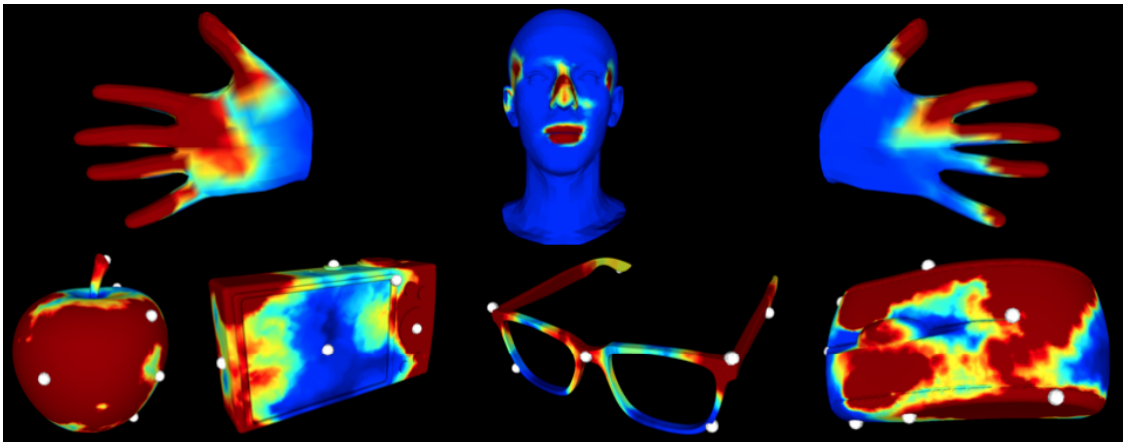


Figure 2.10: Contact "heatmaps". **Top:** For the body we focus on "use" sequences to show "whole-body grasps" where the hands and face are involved. **Bottom:** For objects we include all intents. Object markers (light gray) are unobtrusive and can lie on "hot" (red) contact areas.

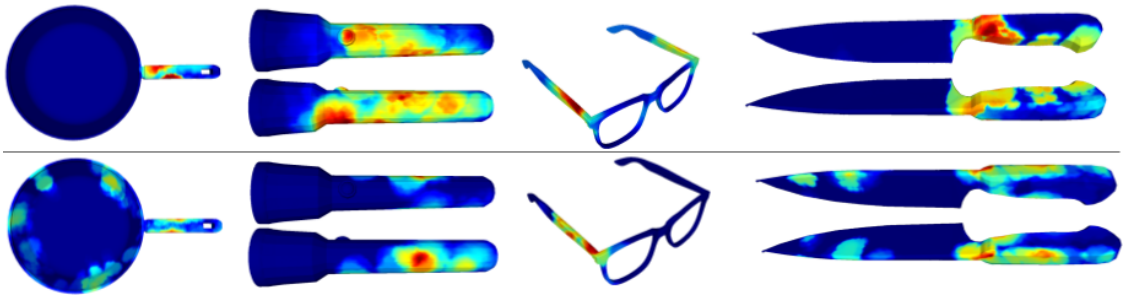


Figure 2.11: Effect of interaction intent on contact during grasping. We show the "use" (top) and "pass" (bottom) intents for 4 different objects.

of contact across the whole body surface. Figure 2.10 (left) shows such "heatmaps" for "use" sequences. "Hot" areas (red) denote a high likelihood of contact, while "cold" areas (blue) denote a low likelihood. We see that both the hands and face are important for using everyday objects, highlighting the importance of capturing the whole interacting body. For the face, the "hot" areas are the lips, the nose, the temporal head area, and the ear. For hands, the fingers are more frequently in contact than the palm, with more contact on the right hand than the left. The palm seems more important for right-hand grasps than for left-hand ones, possibly because all our subjects are right-handed. Contact patterns are also influenced by the size of the object and the size of the hand (see .

Figure 2.11 shows the effect of the intent on the contact maps of 4 objects. Contact for "use" sequences comply with the functionality of the object; e.g. people do not touch the knife blade or the hot area of the pan, but they do contact the

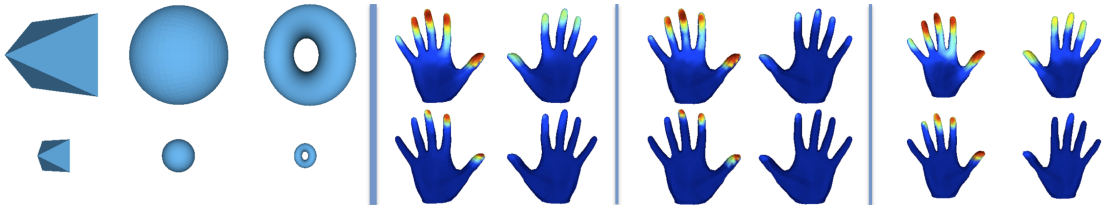


Figure 2.12: Effect of object size on contact during grasping. We show contact "heatmaps" for all subjects, the biggest subject, and the smallest subject.

on/off button of the flashlight. For "pass" sequences subjects tend to contact one side of the object irrespective of affordances, leaving the other one free to be grasped by the receiving person.

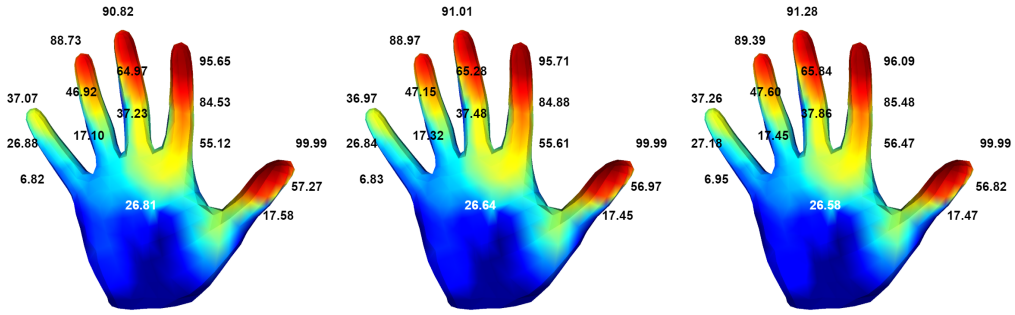
For natural interactions, it is important to have a minimally intrusive setup. While our MoCap markers are small and unobtrusive (Figure 2.2), we ask whether subjects may be biased in their grasps by these markers. Figure 2.10 (Bottom) shows contact "heatmaps" for some objects across all intents. These clearly show that markers are often located in "hot" areas, suggesting that subjects do not avoid grasping these locations. Further analysis based on K-means clustering of grasps can be found in Appendix A.

Figure 2.12 shows the effect of hand and object size, for our 5 shape primitive objects, on the grasp type. Smaller objects are grasped mainly by the dominant hand, while for bigger objects the other hand is involved too. Small-sized subjects have a bigger tendency for bimanual grasps, and involve more fingers and contact around MCP joints. Additionally, we see that subjects with bigger hands handle most of the interaction motions with a single hand.

2.5.3 Influence of Contact Heuristic Thresholds

We use several heuristics to determine contact frames, see Sec. 2.4.5. For the contact "heatmap" analysis we take all the contact frames for which the object is being manipulated, i.e. it is off the table. Because the heatmap is integrated over many frames, small variations in the heuristics have little impact on the contact patterns.

To show this empirically, we perform a sensitivity analysis by changing our thresholds. Figure 2.13 shows "heatmaps" for "use" sequences for several setups,



Filter setting	H: 2mm , C: 40 vertices	H: 5mm , C: 50 vertices	H: 15mm , C: 60 vertices

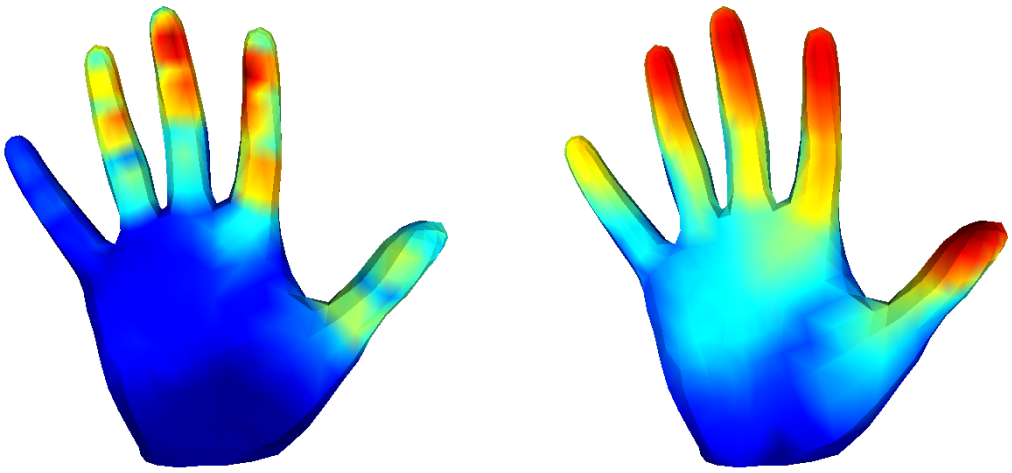
Figure 2.13: Sensitivity analysis for contact heuristics. We follow the format of Fig. 2.10 and show "heatmaps" and contact likelihoods in percentages % for a subset of "all" sequences with different setups (columns), as indicated in the labels. The symbol H denotes the minimum difference between the object's vertical position from its initial one (resting on a table). The symbol C denotes the minimum number of object vertices that we require to be in contact with each finger. The figure shows that threshold choices have a minimal effect when integrated over many frames to create "heatmaps".

following the format of Fig. 2.10 The results verify our hypothesis that the heuristics have minimal influence on the computed contacts.

2.5.4 MoCap VS RGB Images

Capturing *accurate* human-object interactions while also capturing *natural* RGB images is very challenging. Some recent datasets [56, 57] capture hand-only interactions with objects and include RGB images, but the images capture only the hand and not the whole body [56, 57] and are not fully natural due to visible instrumentation on the hand [56]. Please note that this latter point is fundamental. Currently, one must choose between accurate grasping, which requires instrumentation, or natural images, which reduces the accuracy of ground truth.

Both methods [56, 57] suffer from severe hand-object inter-penetrations. Garcia-Hernando et al. [56] originally reconstruct a hand skeleton interacting with 4 object meshes, and their method was reported to have an average *skeleton* penetration depth of 11.0 ± 8.9 mm (see Sec. 5.2 of [17]). Similarly, we compute the *surface* penetration between the hands and the 3D object meshes for [57] and find the mean to be 4.36 ± 0.94 mm. Although the hand inter-penetration of [57] is not as severe as [56], it suffers from not having realistic contact with objects. In Fig. 2.14 we



HO-3D dataset [57] GRAB dataset (ours)

Figure 2.14: Contact "heatmaps" for HO-3D [57] (left) and GRAB (right), for the right hand. The hotter the color, the more frequently that hand part is in contact with objects. During grasping and manipulation, the thumb finger and all fingertips play a central role. This is evident with GRAB but not with [57].

compare the contact "heatmaps" for [57] (left) and GRAB (right). Note that for [57], the thumb and all fingertips are rarely in contact, whereas for GRAB they are frequently in contact. The latter is much more realistic given the central role of the thumb and fingertips in object manipulation. This points to an important technical problem, showing the challenges in accurately tracking the hands.

We conclude that state-of-the-art interaction methods, which also capture RGB images, suffer from intense occlusions and penetrations along with non-realistic contact between the hand and the objects. Such data is not good to learn an accurate data-driven model of 3D interactions. In contrast, our "use" grasps have only 3.25 ± 0.68 mm average *surface* penetration, which is significantly lower than [56, 57], and realistic contact between the body and objects, while containing more challenging scenarios, namely dexterous in-hand manipulation and capture of the whole body instead of only the hand.

This is attributed to our precision-focused setup, which increases accuracy at the expense of not capturing RGB images, due to the uniform and artificial texture of the MoCap body suit and the 3D printed objects. We believe that this is a sensible trade-off; one can use our accurate 3D mesh reconstructions to learn a model of 3D interactions, and use it as a prior in future work to improve methods like [56, 57] for the hand or the whole body.

Acknowledging the limitations and complexities associated with capturing *accurate* human-object interactions along with *natural* RGB images, we endeavored to address and overcome these challenges in our subsequent work, ARCTIC [91]. In our ARCTIC paper, we significantly advanced our approach to tackling the issues faced in previous datasets. ARCTIC presents a comprehensive dataset featuring two hands manipulating objects in a dexterous manner, encompassing 2.1M video frames paired with precise 3D hand and object meshes, and detailed, dynamic contact information. This dataset uniquely includes bi-manual articulation of objects, such as scissors or laptops, showcasing the synchronized evolution of hand poses and object states over time.

2.5.5 MoCap VS 3D Scan Sequences

For accurate human shapes, we capture a dense 3D scan for each subject to which we fit a personalized 3D SMPL-X template mesh. However, 3D scanning does not scale up for capturing human-object interaction sequences. This would produce huge amounts of data, the processing of which would be a major undertaking. Moreover, object tracking under occlusions would be still very challenging, as finding scan-to-model correspondences is a hard ill-posed problem. Instead, with MoCap a minimum of 3 marker observations is enough for reliable object pose estimation. The placement of many small markers on the objects means that we can always estimate object pose. Using MoSh++ for the body, given a ground truth body shape, produces accurate meshes that are on par with 3D scanning but much more practical to capture. We follow therefore this practical and scalable approach; we use a high-end optical MoCap system (Sec. 2.4) and fit full 3D meshes to MoCap markers for both the human and the object.

2.5.6 Penetration Analysis

Although our method is fairly accurate, there can still be inter-penetrations. In this section, we evaluate the human-object mesh inter-penetration. and we report the results in Fig. 2.15. We evaluate the degree of penetration for “use” sequences,

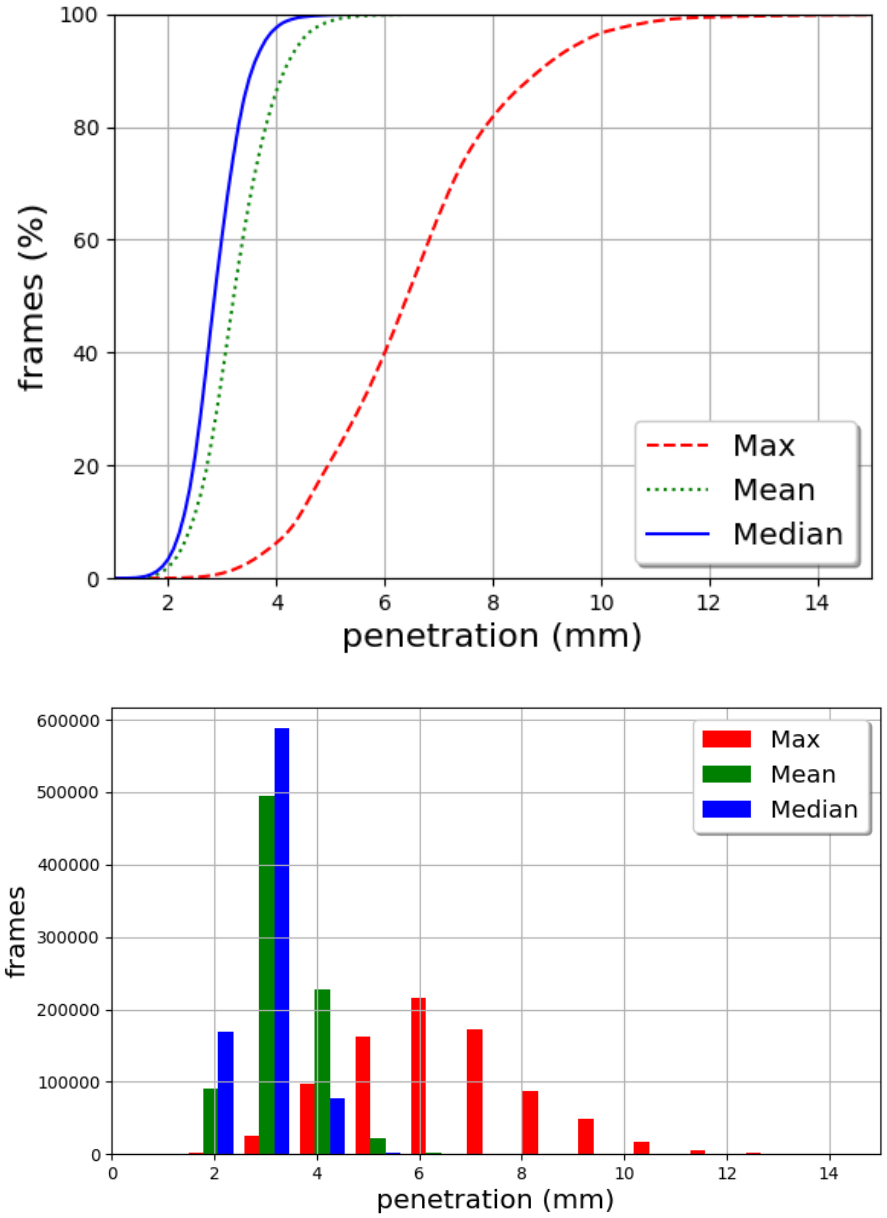


Figure 2.15: Penetration plots for grasp with the "use" intent. For each frame we store the max (red), mean (green), and median (blue) vertex penetration. **(Top)**: percentage of frames (Y-axis) below a varying penetration error (X-axis). **(Bottom)**: bar-plot for the number of frames (Y-axis) with a specific (quantized for binning) penetration (X-axis). The mean penetration is 3.25 ± 0.68 mm.

that pose the most realistic occlusions and capture challenges. "Use" grasps have 3.25 ± 0.68 mm average penetration, which effectively corresponds to the missing soft-tissue deformation. Please note that there is no model of the human body with articulated fingers and face that captures such soft-tissue deformation with contact. In addition, 67% of "use" grasps have ≤ 3.5 mm penetration, 86% ≤ 4.0

mm, 96% $\leq$ 4.5 mm and 99.9% has $\leq$ 5.8 mm.

2.6 Conclusion

We provide a new dataset to the community that goes beyond previous motion capture or grasping datasets. GRAB has shown to be be useful for a wide range of problems in the community and has been well received. In the next chapter, we show that it provides enough data and variability to train a novel network to predict object grasping, as we demonstrate with GrabNet (Chapter 3). But there is much more that can be done. Importantly, GRAB includes the full body motion, enabling much richer modeling than GrabNet.

Limitations: By focusing on accurate MoCap, we do not have synced image data. However, GRAB can support image-based inference [92, 93] by enabling rendering of synthetic human-object interaction [17, 94, 95] or learning priors to regularizing ill-posed inference of human-object interaction from 2D images [60].

Future Work: GRAB can support learning human-object interaction models [19, 60], robotic grasping from imitation [96], learning mappings from MoCap markers to meshes [50], rendering synthetic images [17, 94, 95], inferring object shape/pose from interaction [97, 98], or analysis of temporal patterns [99].

Grasping the unseen is the great challenge of our era.

— Stephen Hawking

3

GrabNet: Generating Static Grasps for 3D Objects

Contents

3.1 Introduction

The ability to interact with the environment through grasping and manipulating objects is a foundation of human behaviour. This capability has fascinated and challenged researchers, inspiring research in various fields, including robotics, computer vision, and graphics. With the advent of immersive technologies such as virtual and augmented reality, the demand for sophisticated hand-object interaction models

has surged. Therefore, accurate hand pose estimation and realistic grasp synthesis for unseen objects became essential for the development of these technologies.

However, modeling accurate hand-object interaction comes with several challenges. Many existing works have tackled the problem from a robotics perspective, focusing on the prediction of the gripper pose for robotic arms [100], which is far from the human grasp. Moreover, some methods primarily rely on a handcrafted or optimization-based refinement process, which can lead to unrealistic and implausible hand grasps. Additionally, using inaccurate hand models such as skeletons leads to unrealistic finger-object grasps with penetrations. Some methods use more realistic hand models like MANO [73], however, these methods are often limited in their ability to generalize to new unseen objects due to the lack of accurate human-object interaction datasets.

This thesis presents GrabNet, a novel model designed to address these challenges and gaps in the literature. GrabNet employs a two-stage process consisting of CoarseNet and RefineNet to generate plausible 3D grasps for unseen objects. CoarseNet generates an initial grasp using a Conditional Variational Autoencoder (cVAE), while RefineNet is a neural network that refines the initial pose based on the distances between the hand and object meshes. This neural network approach contrasts with the optimization-based refinement process commonly used in the literature. For this We utilize the Basis Point Set (BPS) [101] representation, to encode the object shape as a set of distances from the basis points to the nearest object points. This representation provides a robust and generalizable method for 3D object representation, leading to generating realistic hand-object grasps during inference. Additionally, we leverage the GRAB dataset Chapter 2 to learn realistic candidate contact points on the hand surface, as opposed to previous methods that use handcrafted constraints. This approach ensures a more data-driven and realistic hand-object interaction.

In the following sections of this chapter, we will provide a detailed description of the GrabNet model, its components, and the methodology behind its design. We will also present comprehensive experimental results that demonstrate its effectiveness in generating plausible 3D hand grasps for unseen objects.

3.2 Related Work

This section reviews the relevant literature on generating plausible 3D hand grasps for unseen 3D objects. We will discuss four areas of research that have direct relevance to this study: Hand Grasp and Contact Capturing, grasp synthesis methods, 3D object representations, and hand pose estimation.

3.2.1 Hand Grasp and Contact Capturing

Extensive research has been conducted in the domain of capturing and identifying human grasp patterns [99, 102–105]. This is crucial to understanding how the human hand adapts its pose to ensure a secure grasp on an object and has many applications in robotics, computer graphics, and computer vision. Some works in this area have utilized various sensors to capture contacts, such as stretch-sensing gloves [106], touch screens [107], or other modalities [13, 108]. These, however, interfere with natural interactions. Some others like [108] used detected keypoints to track hand and object pose and paired them with thermal contact, but lack the required accuracy. Here we use, GRAB [41] dataset that provides a very accurate dataset of human-object interaction as described in Chapter 2.

3.2.2 Grasp Synthesis Methods

Synthesizing grasps is a critical component in many robotic and graphic applications, resulting in very extensive literature [109–123]. Grasp synthesis is useful for graphics and robotics to help characters [36, 37] or robots [58] interact with their surroundings, and for vision applications [59, 60] to help reconstruct interactions from ambiguous data. Many traditional works take an optimization approach [123, 124] to satisfy hand-crafted stability and grasp constraints. These methods suffer from unnatural poses and interactions. Some other works take a data-driven approach [125, 126] and try machine learning methods to learn a grasp representation completely from data. These methods, however, mainly generate inaccurate results.

To get the best of both, recent methods [100, 121, 127] take a hybrid approach and combine the deep learning approaches with an analytical approach for grasp synthesis. In contrast to Mousavian et al. [100], our network latent-space captures not only the 6 DoF pre-grasp pose (gripper for [100]/wrist for MANO), but also the fully articulated human hand pose. Karunratanakul et al. [128] propose to use learnable representation for modeling hand-object interaction that can be used without contact post-processing. Unlike others, we employ a two-step approach to generate a coarse grasp and then refine it using fully learnable methods. This differs from previous works that use an optimization process for grasp refinement, and we show it produces more realistic and plausible hand grasps, making it suitable for tasks that require more intricate manipulation.

3.2.3 3D Object Representations

The representation of a 3D object is a critical component in computer vision and graphics [121, 129–136]. Traditional methods often use surface-based or voxel-based representations, which are not robust when dealing with previously unseen objects. Recent methods use Signed Distance Functions (SDF) [137], Occupancy Networks [138], or various Implicit Fields [139] to represent objects. Inspired by these, our model uses a relative of the SDF, the Basis Point Set (BPS) [101], which encodes the object shape as a set of distances from the basis points to the nearest object points. This representation allows our model to easily generalize to new object shapes.

3.2.4 Hand Pose Estimation

Hand pose estimation aims to predict the pose of a hand and object given certain conditions or input data, such as RGB or RGB-D images [140–149]. Accurate reconstruction of such poses requires accurate modeling and understanding of hand-object interaction. Previous studies such as [17, 150] use handcrafted weights to enforce contact between specific hand areas and objects. This leads to unnatural grasps that lack diversity. In contrast, GrabNet employs learned weights from the GRAB dataset to model accurate interaction between the hands and the object.

3.3 Method

We use the GRAB dataset to train neural models that generate plausible 3D MANO [73] grasps for a previously unseen 3D object.

Our model called GrabNet, consists of two main modules. First, we employ a conditional variational autoencoder (cVAE) [151], called CoarseNet, which generates an initial hand grasp. For this, the encoder Q takes the ground-truth MANO pose and translation and maps them to a grasping embedding space Z. To represent the object shape, we use the Basis Point Set (BPS) [101] representation, which is represented as a set of distances from fixed basis points to the nearest object points. This is very similar to the Unsigned Distance Field representation and has been shown to generalize easily to new object shapes. For the reconstruction we sample from the latent space and concatenate it to the object shape representation and pass it to the decoder, P. As output, the decoder reconstructs the hand pose and translation corresponding to the grasp.

CoarseNet's grasps are mostly reasonable, but their realism, in terms of penetration and hand-object contact, can improve. Therefore, we propose a second network,called RefineNet, to further improve the grasps. Since the hand generated by CoarseNet are already very close to the object surface, we propose to refine their pose using the distances D between the hand and the object. THis provides a very rich information about the proximity between the hand and the object and is shown to be very effective. To refine grasps, we first simulate noisy grasps by perturbing ground-truth hand grasps with random noise (see Sec. 3.3.2 and then train RefineNet to regress the ground truth grasps. As input RefineNet takes hand pose and D of a perturbed hand and regresses the ground-truth hand-poses. Additionally, we observe that applying the refinement step 3 times achieves a better performance compared to only 1 time. The overall architecture of GrabNet is shown in Fig. 3.1.

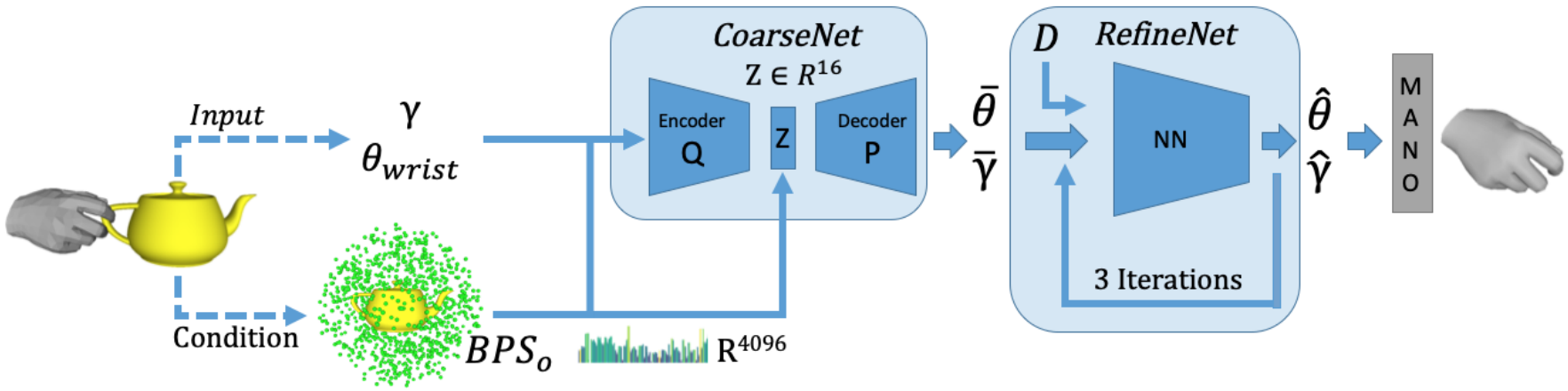


Figure 3.1: GrabNet architecture. GrabNet generates MANO [73] grasps for unseen object shapes, encoded with a BPS [101] representation. It is comprised of two main modules. First, with CoarseNet we predict an initially plausible grasp. Second, we refine this with RefineNet to produce better contacts with the object.

3.3.1 Hand Model

To represent the hands we use the MANO parametric hand model. Similar to SMPL-X, MANO is a statistical model that is parametrized with hand pose θ_h, hand translation, γ_h, and hand shape, β_h, and map them to a 3D hand mesh. The pose parameters represent the hand joint rotations, and the shape parameters control the person-specific deformations of the hand; see [68] for more details. To represent the hand pose, we use the continuous 6D rotation representation from [152], $\theta_h \in R^{96}$.

3.3.2 Data Preparation

For training, we gather all frames with right-hand grasps that involve some minimal contact.

CoarseNet: We use only right-hand grasps from the GRAB dataset, but left-hand poses could also be mirrored to appear as right-hand ones for data augmentation. To select the right-hand frames for training GrabNet we use the following rules. (i) The right hand should be in contact with the object (The right thumb and at least one more finger should be in contact.). (ii) The left hand should not have any contact. (iii) The object's vertical position should be at least 5 mm different from its initial one (i.e. it should be lifted from the resting table). (iv) A finger is considered a contacting finger when it is in contact with at least 50 object vertices. With these filters, we make sure that we have only stable grasps with which to train GrabNet.

To model arbitrary shapes, we use the basis point set [101] representation b_o for all our objects. This representation is very similar to the SDF representation and helps our networks generalize to new object shapes. For computational efficiency, we precompute b_o and load it from memory during training. Here, we sample basis points randomly in a sphere of 150 mm radius, which is big enough to cover our objects. We empirically found 4096 basis points to be enough to capture the object shapes for our purpose. We then center each training sample, i.e. hand-object grasp, at the centroid of the object and compute the $b_o \in R^{4096}$ representation for the object. This is performed by computing the distances from each BPS

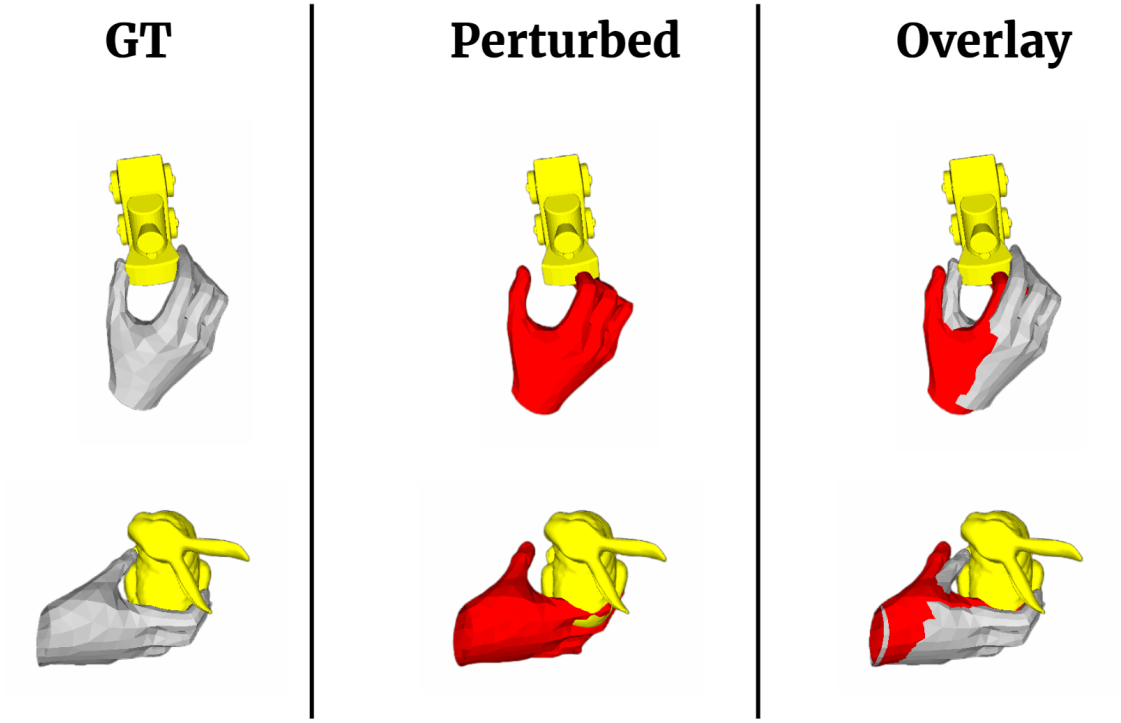


Figure 3.2: We show two examples of the perturbed hand poses for RefineNet training. The perturbed grasps have both penetration (red-index) and no-contact (red-thumb) cases. RefineNet is trained to take the perturbed grasps and regress the ground-truth ones.

point to the closest point on the object. This is later used as a condition in our CoarseNet network for generating a grasp.

RefineNet: The GRAB dataset provides MANO hand parameters for both hands in addition to the full-body SMPL-X parameters. To prepare the training data for RefineNet we add Gaussian noise to the Ground Truth MANO parameters of the selected data for GrabNet. This is to simulate noisy grasps generated by CoarseNet by adding penetrations or no-contact poses. For this the perturbation needs to be minimal as the generated grasps from CoarseNet are not very noisy. We empirically find the following noise parameters result to accurate grasps: $\mathcal{N}(\mu = 0, \sigma^2 = 0.2)$, $\mathcal{N}(\mu = 0, \sigma^2 = 0.004)$, and $\mathcal{N}(\mu = 0, \sigma^2 = 0.05)$ for MANO finger joints rotation, root rotation, and translation respectively. In Fig. 3.2 we show two examples of the perturbed grasps (red) compared to the ground-truth ones (gray).

Out of our 51 objects, borrowed from [13], we hold out 4 objects for the validation set (“apple”, “toothbrush”, “elephant” and “hand”), 6 objects for the test

set (“mug”, “wineglass”, “camera”, “binoculars”, “frying pan” and “toothpaste”), and use the remaining 41 objects for the training set.

The training, validation and test splits contain roughly 320k, 31k and 65k grasping data points, correspondingly.

3.3.3 Network Architecture

We pass the object shape b_o along with the initial MANO wrist rotation, θ_{wrist}, and translation γ to the encoder $Q(Z|\theta_{wrist}, \gamma, b_o)$ that produces a latent grasp code $Z \in R^{16}$. The decoder $P(\bar{\theta}, \bar{\gamma}|Z, b_o)$ maps Z and b_o to MANO parameters with full finger articulation $\bar{\theta}$, to generate a 3D grasping hand. The outputs of the decoder are MANO translation ($\hat{\gamma} \in R^3$) and joint angles ($\hat{\theta} \in R^{96}$) in the continuous 6-dimensional representation of [152].

For the training loss, we use the standard cVAE loss terms given by

$$\mathcal{L}_{\text{cVAE}} = \mathbb{E}_{q(Z|\theta_{wrist},\gamma,b_o)}[\log P(\bar{\theta}, \bar{\gamma}|Z, b_o)] - \lambda_{kl}\text{KL}(Q(Z|\theta_{wrist}, \gamma, b_o)||P(Z)) \quad (3.1)$$

where λ_{kl} is the weight regularizer, controlling the trade-off between the reconstruction and the KL divergence term.

Furthermore, we define an L2 loss on the hand mesh vertices, defined as:

$$\mathcal{L}_{\text{L2}} = \sum_i w_i \|v_i - \hat{v}_i\|^2 \quad (3.2)$$

where v_i and $\hat{v}_i$ are the original and reconstructed vertex positions, respectively, and w_i are the learned vertex weights based on the vertex number, learned from GRAB in contrast to handcrafted ones [6] or weights learned from artificial data [17].

Additionally, we incorporate a data term on MANO mesh edges, formulated as an L1 loss:

$$\mathcal{L}_{\text{L1}} = \sum_i |e_i - \hat{e}_i| \quad (3.3)$$

where e_i and $\hat{e}_i$ are the original and reconstructed mesh edges, respectively.

Finally, we introduce a penetration loss $\mathcal{L}_{\text{penetration}}$ and a contact loss $\mathcal{L}_{\text{contact}}$. For the contact loss, we consider a vertex to be in contact if it is closer than

a threshold, here 2.5 mm, to the object, and compare it with the ground truth ones. The contact loss is thus formulated as:

$$\mathcal{L}_{\text{contact}} = \sum_i \mathbb{I}[d_i < \delta](d_i - d_{\text{gt},i})^2 \tag{3.4}$$

where d_i is the distance of the i^{th} vertex from the object, $\delta = 2.5$ mm is the contact threshold, $d_{\text{gt},i}$ is the ground truth distance, and $\mathbb{I}[\ \cdot\ ]$ is the indicator function, which equals 1 if the vertex is considered in contact and 0 otherwise.

For the penetration loss, we compute whether a vertex of the hand is inside the object and penalize it accordingly. The penetration loss can be represented as:

$$\mathcal{L}_{\text{penetration}} = \sum_i \max(0, -d_i) \tag{3.5}$$

where d_i is the signed distance of the i^{th} vertex from the object's surface, being negative when the vertex is inside the object.

The total loss is then the weighted sum of all the individual loss terms:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cVAE}} + \lambda_1\mathcal{L}_{\text{L1}} + \lambda_2\mathcal{L}_{\text{L2}} + \lambda_3\mathcal{L}_{\text{penetration}} + \lambda_4\mathcal{L}_{\text{contact}} \tag{3.6}$$

where λ_1, λ_2, λ_3, and λ_4 are the weights for the L1, L2, penetration, and contact losses, respectively.

At inference time, given an unseen object's BPS shape, b_o, we sample from the latent space Z, concatenate them, and decode our sample to generate a MANO grasp. Using our validation set, we found that 16 dimensions for the latent space result in generating more realistic grasps compared to other latent-code dimentions.

RefineNet: The grasps estimated by CoarseNet are plausible but can be refined for improved contacts. For this, RefineNet takes as input the initial grasp ($\bar{\theta}$, $\bar{\gamma}$) and the distances D from MANO vertices to the object mesh. The distances are weighted according to the vertex contact likelihood learned from GRAB, shown in Sec. 3.3.3. Then, RefineNet estimates refined MANO parameters ($\hat{\theta}$, $\hat{\gamma}$) in 3 iterative steps similar to [93], to give the final grasp. This iterative refinement is the key to adding fine-grained hand-object grasp and generating a realistic final grasp. The network architecture consists of 3 fully connected residual networks with skip connections between them.

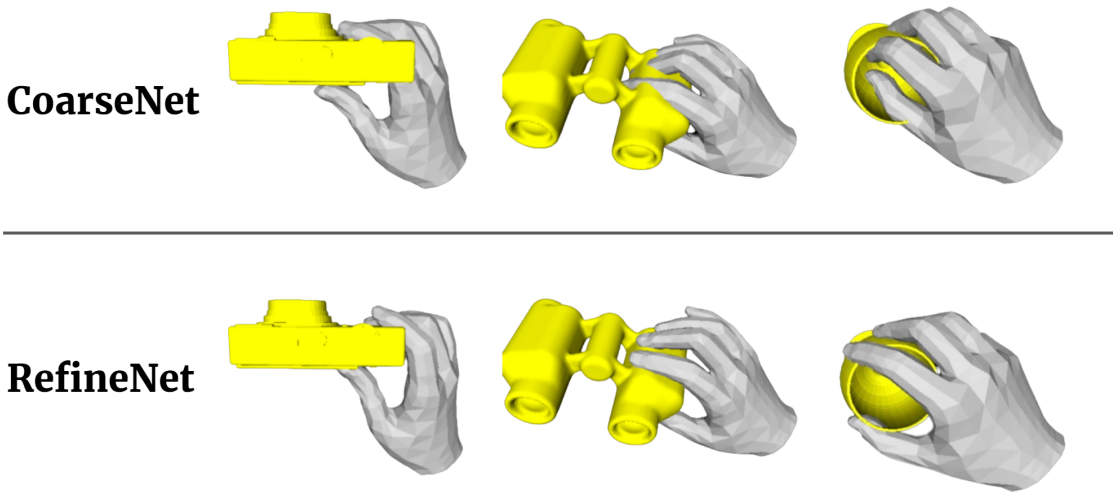


Figure 3.3: Grasp quality before and after using RefineNet. We show the initial grasps generate using CoarseNet (Top) and the refined grasps after using RefineNet (Bottom). The results show that RefineNet is able to remove penetrations and make the grasps more realistic and physically plausible.

To show the effect of our RefineNet, in Fig. 3.3 we show the grasp quality before and after using RefineNet. The results show that RefineNet makes the initial results from CoarseNet much more realistic and physically plausible.

To train RefineNet, we generate a synthetic dataset of noisy grasps as explained in the previous section. Then, we train RefineNet to recover the ground-truth hand poses. We use the same training losses as for CoarseNet.

The CoarseNet and RefineNet are trained for 16 and 23 epochs respectively with the learning rate starting from $5e-4$, decreasing on validation error plateau to 0.1 times, and early stopping after 8 epochs with no improvement in validation error. Both networks are trained separately.

3.4 Evaluation

3.4.1 Quantitative

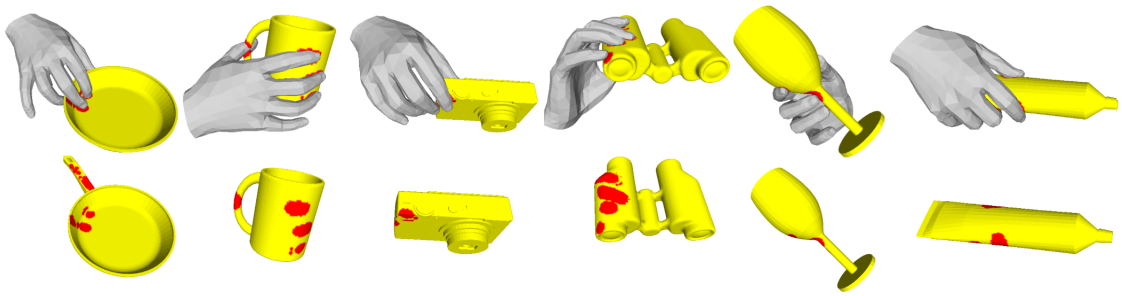
We first quantitatively evaluate the two main components, by computing the reconstruction Mean Per Vertex Position Error (MPVPE), as shown in Tab. 3.1.

Please note that the two networks are trained separately, however during the inference they are connected together. The results show that the components, that are trained separately, work reasonably well and generate accurate results.

Table 3.1: Evaluation of Mean Per Vertex Position Error (MPVPE) for CoarseNet and RefineNet for train, test, and validation sets of the grab dataset.

	MPVPE (mm)		
Model	Training	Validation	Test
CoarseNet	12.1	14.1	18.4
RefNet	3.7	4.1	4.4

Effectively evaluating the realism of the generated grasps is challenging through the available metrics like penetration. However, humans inherently possess the ability to evaluate the realism of generated grasps. Therefore, to evaluate GrabNet generated grasps, we perform a user study through Amazon Mechanical Turk (AMT) [153]. For the perceptual study, we take 6 test objects from the dataset and, for each object, we generate 20 grasps, mix them with 20 ground-truth grasps, and show them with a rotating 3D viewpoint to participants. Then we ask participants how much they agree with the statement "Humans can grasp this object as the video shows" on a 5-level Likert scale (5 is "strongly agree" and 1 is "strongly disagree"). To filter out the noisy subjects, namely the ones who do not understand the task or give random answers, we use catch trials that show implausible grasps. We remove participants who rate these catch trials as realistic; see Appendix B for details. Table 3.2 (left) shows the participants' scores for both ground truth and generated grasps. Additionally, in Fig. 3.4 we show representative grasps with their corresponding scores from the perceptual study below them. Also, in the figure we show the frequency of each likert score for all the generated grasps as a bar plot, which shows a higher number for the scores 4 and 5 compared to others. Overall, the results from our user study reveal that the generated grasps by *GrabNet* exhibit a close approximation to the ground truth, with mean scores of 4.12 and 4.38 respectively, suggesting a high level of accuracy in our method's grasp generation. However, there still exists a subtle, yet discernible preference for the ground truth grasps, underscoring the inherent challenge and room for improvement in perfectly mimicking human grasp patterns.



3.21	3.88	4.06	4.5	4.5	4.75

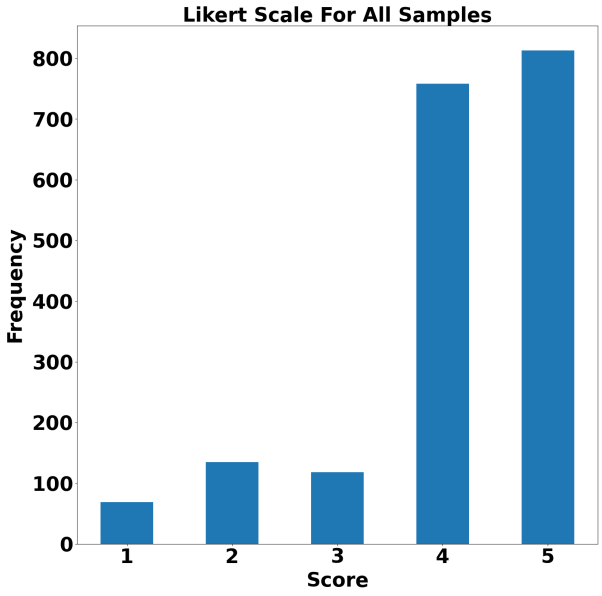


Figure 3.4: (Top) Grasps generated by GrabNet for unseen objects; grasps look natural. As a by-product of 3D mesh generation, we compute the red contact areas. For each grasp, the number below it corresponds to the average Likert score from all annotators. (Bottom) The frequency of each Likert score is shown for all generated grasps.

3.4.2 Qualitative

GrabNet.: Given an unseen 3D object, we first obtain an initial coarse grasp with CoarseNet, and then pass this to RefineNet to refine it and obtain the final grasp. For simplicity, the two networks are trained separately, but we expect end-to-end refinement to be beneficial, as in [17]. Figure 3.4 (top) shows some generated examples. We show more qualitative results in Fig. 3.6. Overall, the

Table 3.2: GrabNet evaluation for 6 test objects. The "AMT" column shows the perceptual study results; grasp quality is rated from 1 (worst) to 5 (best). The "vertices" and "contact" columns evaluate grasps against the closest ground-truth one.

	AMT				Vertices	Contact
	Generation		Ground Truth		cm	%
Test Object	mean	std	mean	std	N=100	N=20
binoculars	4.09	0.93	4.27	0.80	2.56	4.00
camera	4.40	0.79	4.34	0.76	2.90	3.75
frying pan	3.19	1.30	4.49	0.67	3.58	4.16
mug	4.13	1.00	4.36	0.78	1.96	3.25
toothpaste	4.56	0.67	4.42	0.77	1.78	5.39
wineglass	4.32	0.88	4.43	0.79	1.92	4.56
Average	4.12	1.04	4.38	0.77	2.45	4.18

qualitative results show that the generated grasps look realistic.

Contact: As a by-product of our 3D grasp predictions, we can compute contact between the 3D hand and object meshes, following Sec. 2.4.5. Contacts for GrabNet generated grasps are shown with red in Figure 3.4 (top). Other methods for contact prediction, like [13], are pure bottom-up approaches that label a vertex as in contact or not, without explicit reasoning about the hand structure. In contrast, we follow a top-down approach; we first generate a 3D grasping hand, and then compute contact with explicit hand-geometry-aware reasoning.

Figure 3.5 shows examples of contact areas (red) generated by [13] (Bottom) and our approach (Top). The method of [13] gives only 10 predictions per object, some with zero contact. Also, a hand is supposed to touch the whole red area; this is often not aligned with the human hand shape. Our contact is a product of MANO-based inference. In other words, we first generate the 3D hand meshes and then compute the contact between the hand and the object based on their proximity. Therefore, this is by construction aligned with the hand shape. Also, one can draw infinite samples from our learned grasping latent space. For further evaluation, we follow a protocol similar to [13] for our data. For every unseen test object, we generate 20 grasps, and for each one, we find both the closest ground-truth contact map and the closest ground-truth hand vertices, for comparison. Table 3.2 (right) reports the average error over all 20 predictions, in % for the former

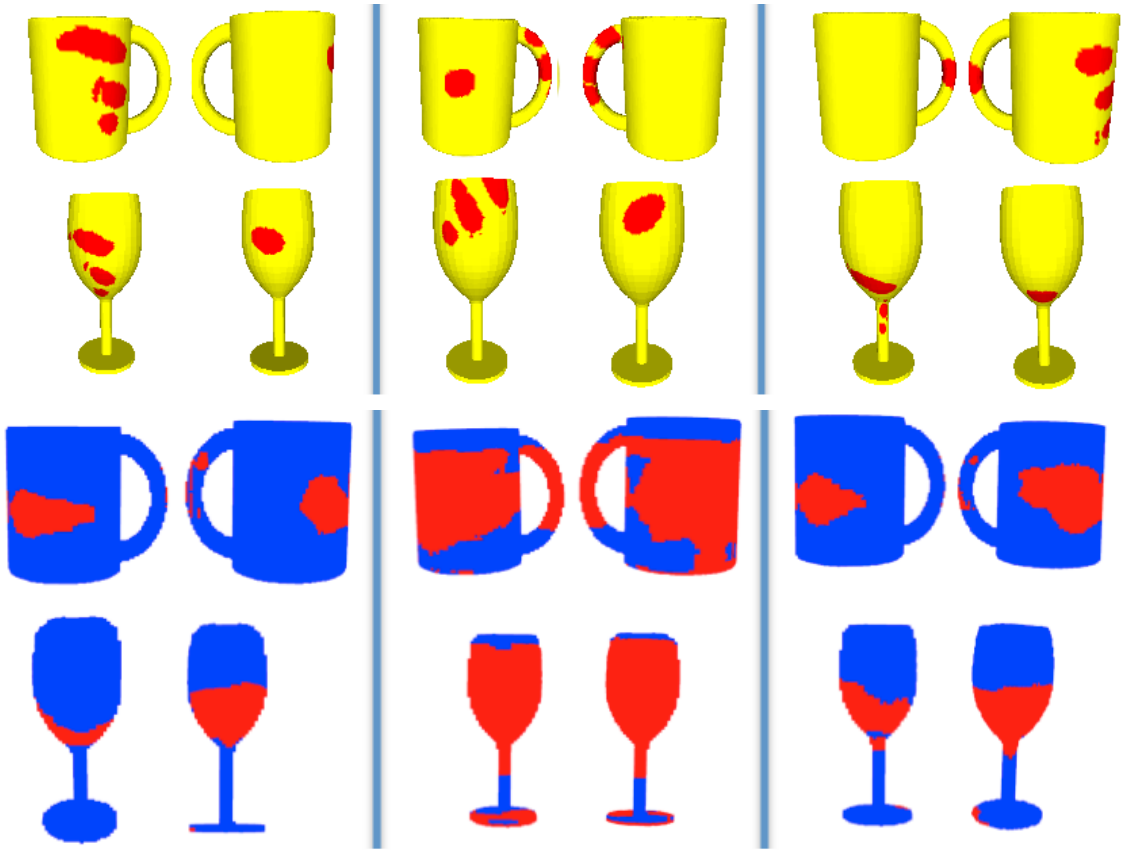


Figure 3.5: Comparison of GrabNet contacts (Top) to ContactDB [13] ones (Bottom). For GrabNet results, we first generate a 3D hand grasp and then compute the contact maps based on the proximity between the hand and the object. For ContactDB, the contact maps are directly generated as the network output. For each estimation we render two views, following the presentation style of [13].

and cm for the latter case. We observe that the average error in vertices ranges around 2.45 cm, indicating a commendable accuracy in vertex prediction. The contact errors, represented as a percentage, are consistently below 5%, showcasing the model's ability to generate realistic grasp resulting in accurate contact areas. Both metrics show the model's capability to generate grasps that are not only geometrically accurate but also practically feasible.

3.5 Conclusion

We introduce GrabNet, a learning-based method trained on GRAB, to generate realistic hand grasps for previously unseen 3D objects. We employ a two-stage

process consisting of CoarseNet and RefineNet, where CoarseNet generates an initial coarse grasp and then RefineNet refines this initial grasp based on the distances between the hand and object meshes. This approach contrasts with the optimization-based refinement process commonly used in the literature. We utilize a robust 3D object representation, the Basis Point Set (BPS) [101], leading to generating realistic hand-object grasps during inference and generalizing well to new object shapes. The evaluation shows that our framework is able to synthesize natural and physically plausible hand grasps.

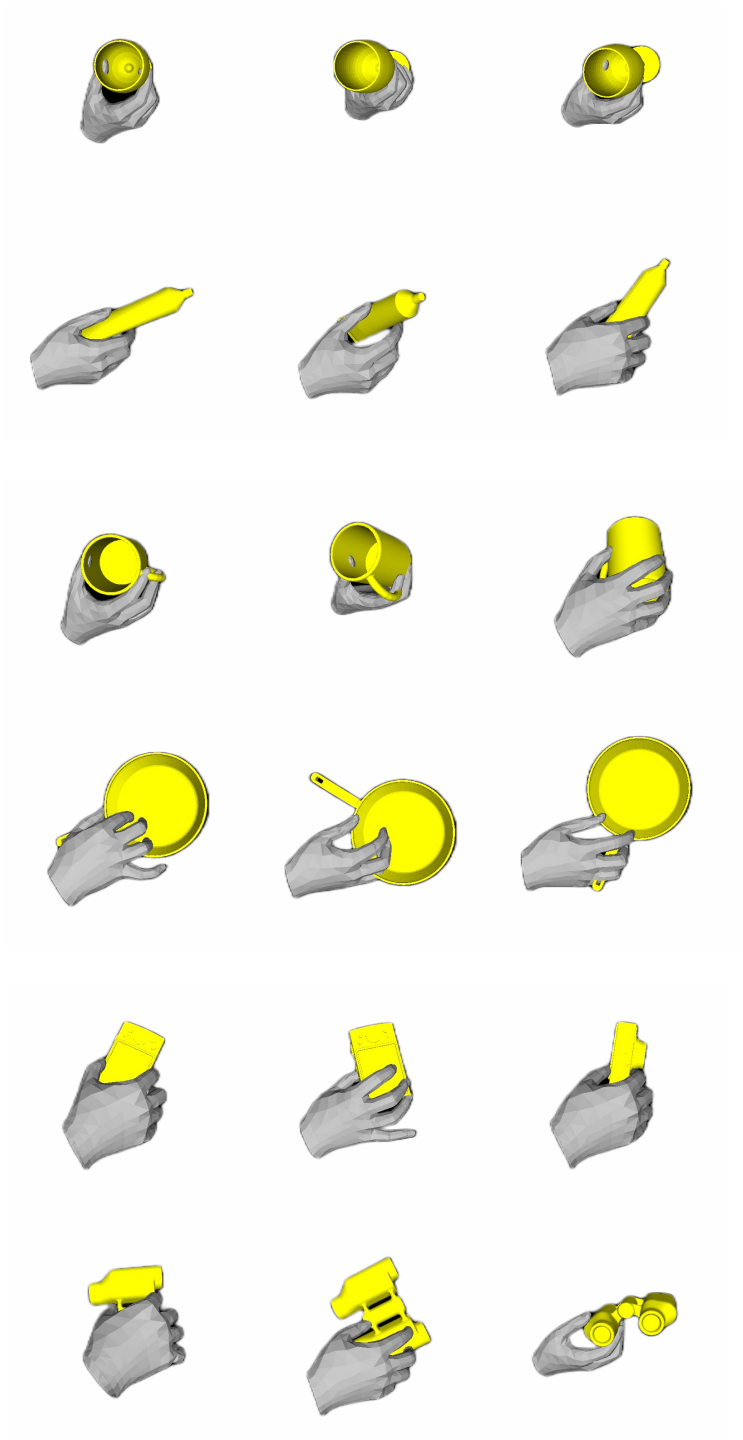


Figure 3.6: Generated grasp from GrabNet for the 6 unseen objects in the test-set. Each row shows 3 generated samples for the same object.

To achieve a goal you have never achieved before, you must start doing things you have never done before.

— John C. Maxwell

4

GOAL: Generating Body Motion to Grasp 3D Objects

Contents

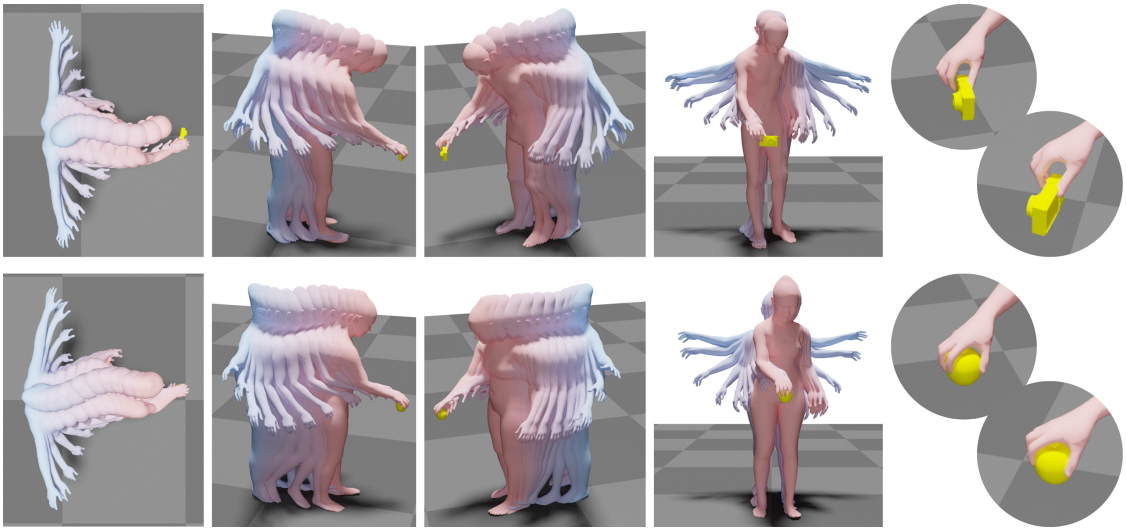


Figure 4.1: GOAL generates whole-body motions for approaching and grasping an unseen 3D object. The figure shows *generated motions* for 2 people (top, bottom), each grasping a different novel object. For each sequence we show 4 different views (left to right), as well as zoomed-in circular snapshots of the final grasp. GOAL is the first method to generate such a natural motion and grasp for the full body.

4.1 Overview

Generating digital humans that move realistically has many applications and is widely studied, but existing methods focus on the major limbs of the body, ignoring the hands and head. Hands have been separately studied (eg. Chapter 3) but the focus has been on generating realistic static grasps of objects. To synthesize virtual characters that *interact* with the world, we need to generate full-body motions and realistic hand grasps simultaneously. Both sub-problems are challenging on their own and, together, the state-space of poses is significantly larger, the scales of hand and body motions differ, and the whole-body posture and the hand grasp must agree, satisfy physical constraints, and be plausible. Additionally, the head is involved because the avatar must look at the object to interact with it. For the first time, we address the problem of generating full-body, hand, and head motions of an avatar grasping an unknown object. As input, our method, called GOAL, takes a 3D object, its position, and a starting 3D body pose and shape. GOAL outputs a sequence of whole-body poses (see Fig. 4.1) using two novel networks. First, *GNet* generates a *goal* whole-body grasp with a realistic body, head, arm,

and hand pose, as well as hand-object contact. Second, *MNet* generates the *motion* between the starting and goal pose. This is challenging, as it requires the avatar to walk towards the object with foot-ground contact, orient the head towards it, reach out, and grasp it with a realistic hand pose and hand-object contact. To achieve this, the networks exploit a representation that combines SMPL-X body parameters and 3D vertex offsets. We train and evaluate GOAL, both qualitatively and quantitatively, on the GRAB dataset. Results show that GOAL generalizes well to unseen objects, outperforming baselines. A perceptual study shows that GOAL's generated motions approach the realism of GRAB's ground truth. GOAL takes a step towards synthesizing realistic full-body object grasping. Our models and code are available for research purposes at https://goal.is.tue.mpg.de.

4.2 Introduction

Virtual humans are important for movies, games, AR/VR, and the Metaverse. Not only do they need to look realistic, but also move and *interact* realistically. Most work on human motion generation has focused only on bodies, without the head and hands. Often, these bodies are considered in "isolation", with no scene or object context. Other work focuses on bodies interacting with scenes, but ignores the hands. Similarly, work on generating hand grasps often ignores the body. We argue that these are all just parts of the problem. What we really need, instead, is to generate motion of *full-body* avatars *grasping* objects, by jointly considering the body, head, hands, and the object. We address this here for the first time.

The problem is challenging and multifaceted. Think of how we grasp objects in real life (see Fig. 4.2); we walk towards the object with our feet contacting the floor, we orient our head to look at the object, lean our torso and extend our arms to reach it, and dexterously pose our hands to establish fine contact and grasp it. Humans are able to gracefully execute these steps, yet, these are challenging and involve motion planning, motor control, and spatial awareness. Some of these steps have been studied separately, but we cannot simply combine the partial solutions since the entire action must be *coordinated*. This is challenging because: (1) full bodies have a much higher-dimensional state space than bodies or hands alone; (2)

the body and hands have very different sizes, motion scales and level of dexterity; (3) the body, head, and hands must move in a coordinated fashion. Currently, there are no automatic tools to generate such coordinated full-body grasping motions.

We address this with *GOAL*, which stands for *Generating Object-interActing whoLe-body motions*. GOAL generates whole-body avatar motion for grasping an unknown object, by jointly considering the body, head, hands, and the object. GOAL takes three *inputs*: (1) a 3D object, (2) its position and orientation, and (3) a "starting" 3D body pose and shape, positioned near the object and roughly oriented towards it. As output, GOAL generates a sequence of 3D body poses from the starting pose through to an object grasp. To do so, GOAL uses two novel networks (for an overview see Fig. 4.3): (1) First, GNet generates a "goal" whole-body grasp, with a realistic body pose, head pose, arm pose, and hand pose, as well as realistic finger-object and foot-ground contact. GNet is formulated as a conditional variational auto-encoder (cVAE), thus, it learns a distribution over grasping poses, and can generate a variety of "goal" grasps. (2) Then, MNet inpaints the motion between the "starting" and "goal" poses, by generating a sequence of whole-body poses in an auto-regressive fashion. This is challenging because the avatar needs to walk by taking several steps proportional to the distance to the object (see Fig. 4.1), while having natural foot-floor contact without "skating", and continuously orient the head to look at the object. Then, when it is near the object, it needs to slow down, stop walking, lean the torso, extend the arms to reach the object. It must also pose the hand to contact the object and grasp it. All body parts need to move gracefully and in full coordination, so that the motion looks natural.

Achieving this level of realism requires technical novelties. GOAL draws inspiration by recent work [27, 154, 155], but goes beyond this to uniquely infer both SMPL-X [5] parameters and 3D offsets. GNet infers 3D hand-to-object vertex offsets to give spatial awareness and guide object grasping. MNet infers 3D SMPL-X vertex offsets to guide SMPL-X deformation from the previous to the current frame. These offsets lie in 3D Euclidean space, thus, they can be more accurately inferred than SMPL-X parameters and are used in an offline optimization scheme to refine SMPL-X poses. We train GNet and MNet on the GRAB [41] dataset, which contains whole-body SMPL-X humans grasping objects (see Chapter 2).

We evaluate GOAL, both quantitatively and qualitatively, on withheld parts of the GRAB dataset. Specifically, we withhold 5 objects for testing. Results show

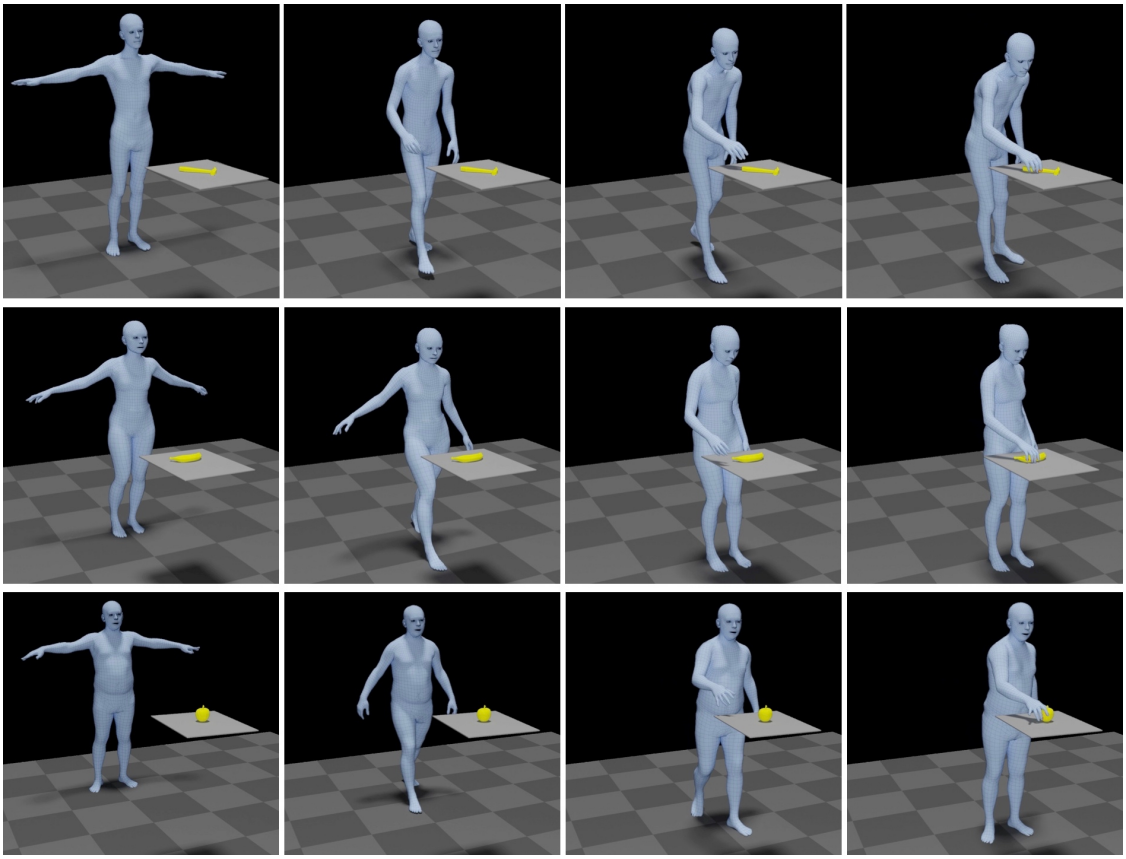


Figure 4.2: Grasping an object involves several motions. We walk towards the object with our feet contacting the floor, orient our head to look at it, lean our torso, extend our arms, and pose our hand to contact and grasp the object. The depicted examples use motions captured in the GRAB dataset (Chapter 2) [41].

that GOAL generalizes well and produces natural motions for full-body walking and object grasping; see Fig. 4.1. Quantitative evaluation shows that GOAL outperforms baselines, and ablation studies show a positive contribution of all major components. A perceptual study verifies the above while showing that GOAL's generated motions achieve a level of realism comparable to GRAB's ground-truth motions.

To conclude, GOAL takes a step towards automatic whole-body grasp motion generation for realistic avatars. Models and code are available for research purposes.

4.3 Related Work

In this section, we review relevant literature in the field of human motion generation within 3D scenes. We delve into the evolution from isolated body movements to

complex full-body interactions, highlighting key data-driven approaches, their limitations, and how they pave the way for our novel full-body motion generation model.

4.3.1 Motion Generation Methods

Motion generation for bodies "in isolation": Research on human motion generation has a long history [21–23]. However, even recent methods [24–27], mostly study the body "in isolation"; i.e., with no scene context. Most methods generate the motion of 3D skeletons [25, 26, 156–159], while others [24, 27, 160] generate the motion of a human model like SMPL [85]. Typically, 1-2 seconds of motion synthesis is referred to as "long term". Early deep-learning methods employ RNNs [157, 161, 162], however, they struggle with discontinuities between the observed and predicted poses, and with long-range spatial relations across time. Other methods account for these with phase-functioned feed-forward neural networks [37, 163], i.e. by conditioning the network weights on phase. However, these focus on cyclic motions. More recent methods [24, 25, 164, 165] adopt an attention [166] mechanism.

Motion generation for bodies in 3D scenes: Most early methods extend MoCap databases with manual and sparse annotations for foot and hand contact [28–31]. Then, they fit motion to contacts with optimization and space-time constraints for 3D body motion re-targeting [28], and animating bodies that move over 3D terrain [29–31].

To avoid big MoCap datasets, some methods use deep reinforcement learning (RL) for body-scene [32–34] or hand-object [167, 168] interactions. These methods show promising results for navigating terrain with varying height and gaps [32, 33], sitting on chairs [34, 37], using a hammer and opening a door [167], and for in-hand object re-orientation [168]. Generalization to new bodies, object geometry, and interaction types remains a challenge.

Others follow a 3D geometric approach. Pirk et al. [99] place virtual sensors on objects to sense the flow of points sampled on an agent interacting with these and build functional object descriptors. Al-Asqhar et al. [169] re-target body motion by encoding human joints w.r.t. fixed points sampled on a scene. Ho et al. [170] use body and object vertices to compute per-frame "interaction meshes",

and minimize their Laplacian deformation to re-target body motion. These pure geometric methods are not robust to real-world noise.

In contrast, we fall in the category of data-driven methods. Corona et al. [79] generate the context-aware motion of a human skeleton interacting with objects, where "context" is encoded as a directed graph connecting person and object nodes. More relevant are methods for generating motion between a "start" and a "goal" pose in a 3D scene. Hassan et al. [171] estimate a "goal" position and interaction direction on an object, plan a 3D path from a start body pose to this, and finally generate a sequence of body poses with an auto-regressive cVAE for walking and interacting, e.g., sitting on a chair. Wang et al. [172] first estimate several "sub-goal" positions and bodies, divide these into short start/end pairs to synthesize short-term motions, and finally stitch these together in a long motion with an optimization process.

Motion generation for hands: ElKoura and Singh [36] estimate physically plausible hand poses for musical instruments, using a learned low-dimensional pose space. Pollard et al. [38] use MoCap to learn a controller for physically-based grasping. Kry et al. [39] capture hand MoCap and forces with sensors on objects, and use these to build "interaction trajectories", and synthesize and re-target motions with physics simulation. More related to us, Lie et al. [40] take as input MoCap data of body and object motion, and add the missing hand motion to the body, by first searching for feasible contact point trajectories, and then generating smooth hand motion with space-time optimization that satisfies the estimated contacts.

4.3.2 Static Pose Generation

Pose generation for bodies in 3D scenes: Early methods use either contact annotations [173] or detections [18] on 3D objects, and fit 3D skeletons to these. Other methods use physics simulation to reason about contacts and sitting comfort [174–176]. Focusing on rooms instead of single objects, Grabner et al. [177] predict all areas on a 3D scene mesh where a 3D human mesh can sit, using proximity and intersection metrics. Recent methods [81, 178, 179] use deep learning to generate static humans interacting with a scene. Zhang et al. [81] learn a cVAE to generate SMPL-X [5] poses, conditioned on an input depth image and semantic segmentation

of the scene. Zhang et al. [178] use an explicit scene-centric representation of interaction, while Hassan et al. [179] use a human-centric representation.

Pose generation for hand-object grasps: As explained in Chapter 3, we predict MANO hand grasps for unseen 3D object meshes, by first predicting a rough hand grasp, and then refining it using distance and contact metrics. Grady et al. [42] refine grasps by first estimating contacts on both the hand and the object and then refining the hand with optimization to satisfy the inferred contacts.

4.3.3 Motion for full-body interactions

People use their body and hands together to interact with the world. Hsiao et al. [4] build a database of whole-body grasps with a human operating an avatar, and perform imitation learning. Borras et al. [3] capture whole-body MoCap data [20] of people interacting with scene objects and handheld objects, using a humanoid model, and define a pose taxonomy. As explained in Chapter 2, we capture whole-body SMPL-X [5] interactions with handheld objects but learn a cVAE that generates only static grasping hands, due to the task complexity. Merel et al. [180] use deep RL and human MoCap demonstrations to learn a vision-guided neural controller for picking up and carrying boxes, or catching/throwing a ball.

Summary: The community has focused on parts of the problem (either the body or the hands) or used unrealistic bodies. GOAL learns to generate full-body SMPL-X motions, from walking to approach an object up to grasping it, given only a 3D object and a starting human pose.

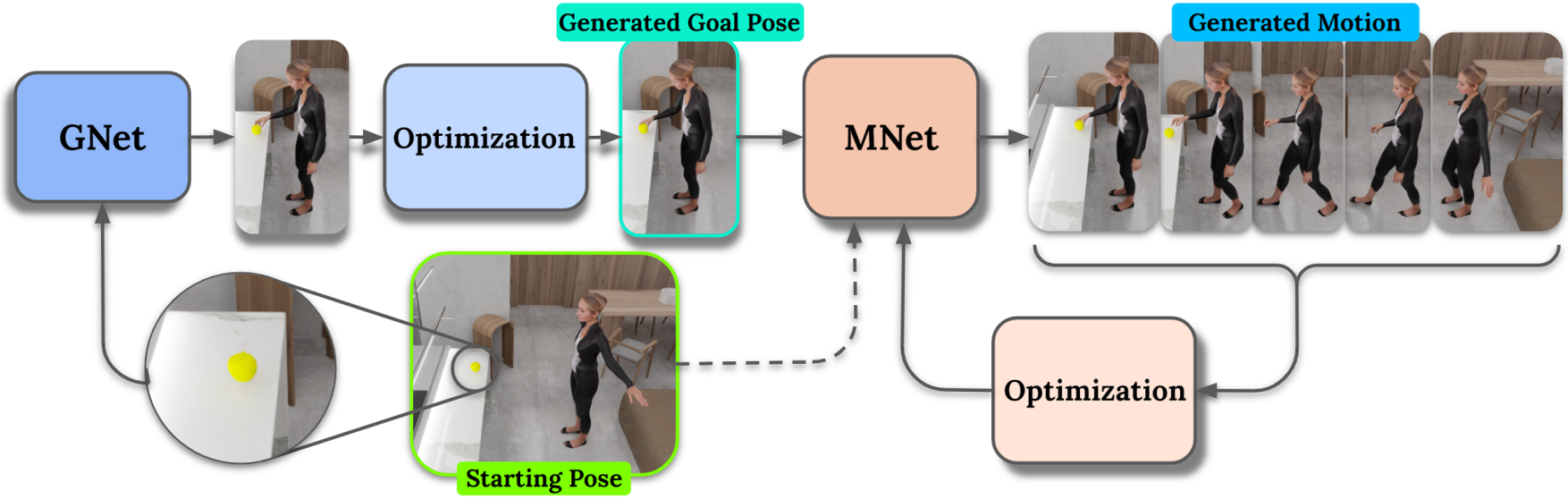


Figure 4.3: Overview of GOAL architecture. There are two main stages: (1) GNet takes as input a 3D object, its position and orientation, and generates a "goal" whole-body grasping pose. GNet's output pose is refined with optimization post processing to look more realistic and physically plausible. (2) MNet takes as input a starting human pose and GNet's "goal" pose, and generates the motion in between as a sequence of poses in an auto-regressive fashion. MNet's output poses are refined with optimization post processing to better "reach" the "goal" pose.

4.4 Method

An overview of our method, GOAL, is shown in Fig. 4.3. GOAL takes three *inputs*: (1) a 3D object, (2) its position and orientation, and (3) a "starting" 3D body pose and shape, positioned near the object (roughly $0.5 - 1.5$ m) and oriented towards it (roughly $\pm 10°$). As *output*, GOAL generates a sequence of SMPL-X poses with two main networks: (1) GNet synthesizes a "goal" SMPL-X mesh that grasps the 3D object with a realistic body pose and hand-object contact; (2) MNet "inpaints" the motion from the "start" to the "goal" pose, by generating a sequence of "moving" SMPL-X bodies in an auto-regressive way. Without loss of generality, we model right-handed grasps; which can be transferred to the left hand easily through "mirroring" data and retraining. Extending these to two-handed grasps, with or without hand coordination, is left for future work.

4.4.1 Human Model

In line with the body model defined in Chapter 2, we continue to use the SMPL-X [5] model, which jointly represents the body, head, face, and hands. However, given our focus on body areas important for interactions, we employ GRAB's [41] contact heatmaps to sample N vertices on these critical areas to represent the body. Let $\boldsymbol{\Theta} = \{\boldsymbol{\theta}, \boldsymbol{\gamma}\}$ represent the articulated pose $\boldsymbol{\theta} \in \mathbb{R}^{55\times 6}$ [152] and translation $\boldsymbol{\gamma} \in \mathbb{R}^3$ of the body, which are carried forward from Chapter 2.

4.4.2 Interaction-Aware Attention

Two common representations for body-object interaction are vertex-to-vertex distances and binary contact labels for mesh vertices; however, the former carries information that is irrelevant to the interaction (e.g., vertices far away from the object), while the latter is too compact and carries no information about 3D proximity before/after contact.

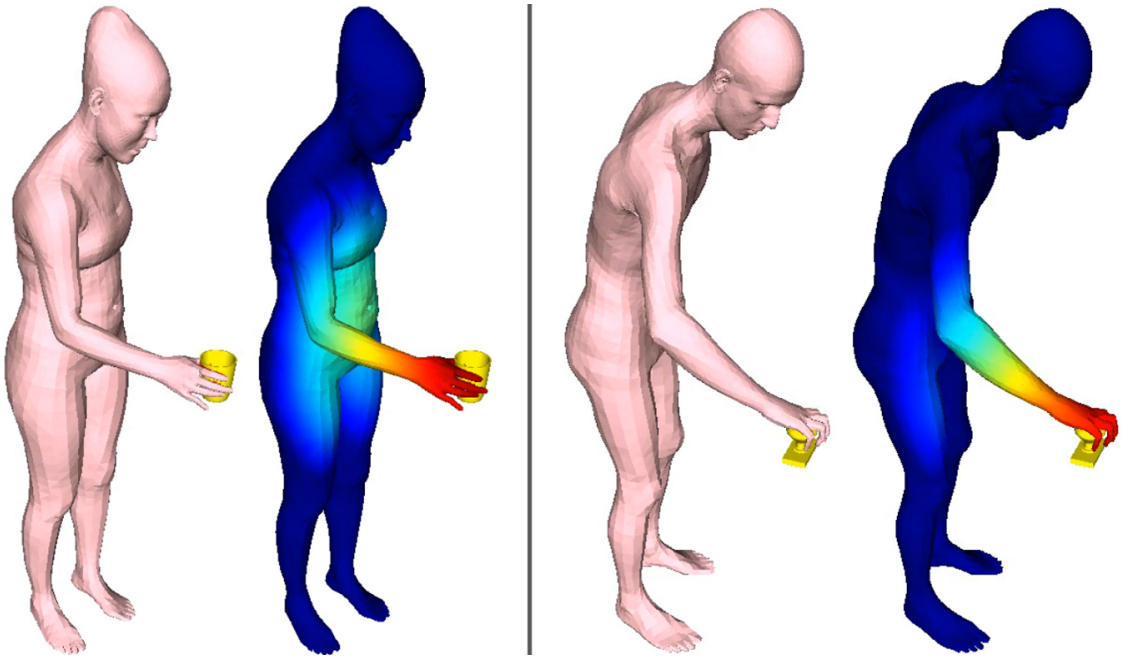


Figure 4.4: Visualization of the "interaction-aware attention" (IAA) for the body-to-object vertex distances, $I_w(\boldsymbol{d})$, of Sec. 4.4.2. For each pair: **(Left)** Input 3D meshes for the human (pink) and the object (yellow). **(Right)** The color-coded body mesh visualizes the interaction-aware attention; blue denotes body vertices that are far from the object (i.e., irrelevant for the specific interaction), and red denotes vertices that are near the object (i.e., very relevant).

Here, we use vertex-to-vertex distances, but introduce a new "interaction-aware attention" (IAA) that focuses more on body vertices that are important for interaction (e.g., hands for grasping, feet for walking) and less on irrelevant vertices (e.g., knees are less relevant than hands for grasping). Our "interaction-aware" attention is formulated as:

$$I_w(d_i) = e^{-wd_i}, \quad \forall i \in 1, 2, ..., N \tag{4.1}$$

where $w > 0$ is a scalar weight, d_i is the i^{th} element of the body-to-object distance vector $\boldsymbol{d} \in \mathbb{R}^N_+$, representing the distance. N is the number of sampled vertices on SMPL-X; we sample $N_b = 400$ for the body (including the hands) and $N_h = 99$ for each hand. Our IAA gives exponentially more attention to vertices relevant for interaction. As visualized in Fig. 4.4, this focuses attention on body areas that are meaningful for interaction. We set $w = 5$, which empirically results in realistic grasps.

4.4.3 GNet - Grasp Network

A detailed architecture of the GNet network and its optimization-based post processing is shown in Fig. 4.5. GNet is a conditional variational auto-encoder (cVAE) [151] that generates a static whole-body grasp, conditioned on the given object and its pose. To do this, we first encode whole-body grasps into an embedding space.

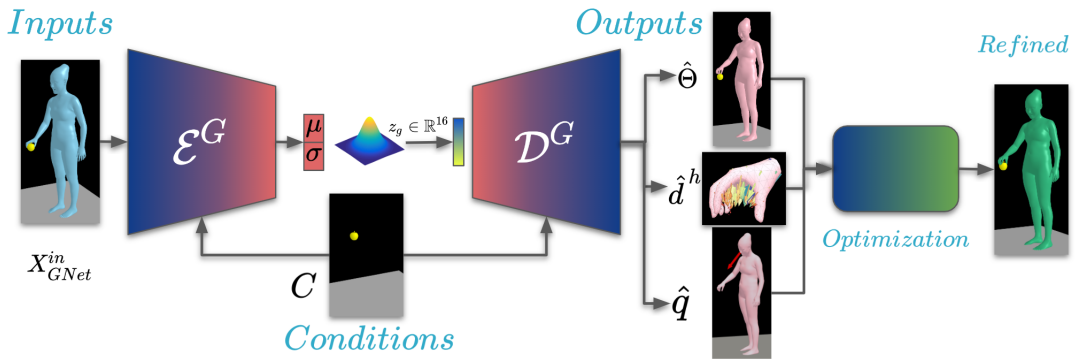


Figure 4.5: Architectural overview of the GNet network, as well as the optimization post-processing step (right-most part).

Input: GNet's encoder takes as input:

$$\boldsymbol{X}^{\text{in}}_{\text{GNet}} = \left[\boldsymbol{\Theta}, \boldsymbol{\beta}, \boldsymbol{v}, \boldsymbol{q}, \boldsymbol{\gamma}^{\circ}, \boldsymbol{b}^{\circ}, \boldsymbol{d}^{b\rightarrow\circ}\right], \tag{4.2}$$

where $\boldsymbol{\Theta}$ and $\boldsymbol{\beta}$ are SMPL-X's pose and shape parameters, respectively, $\boldsymbol{v} \in \mathbb{R}^{N_b \times 3}$ are the 3D coordinates of the N_b sampled SMPL-X vertices, $\boldsymbol{q} \in \mathbb{R}^3$ is a unit vector for head orientation, $\boldsymbol{\gamma}^{\circ} \in \mathbb{R}^3$ is the object translation, and $\boldsymbol{b}^{\circ} \in \mathbb{R}^{1024}$ is the Basis Point Set (BPS) [101] representation of the 3D object shape. Finally, $\boldsymbol{d}^{b\rightarrow\circ} \in \mathbb{R}^{N_b \times 3}$ denotes 3D offset vectors that encode the body-to-object proximity. for each of the sampled body vertices, $\boldsymbol{v}$, $\boldsymbol{d}^{b\rightarrow\circ}$ contains a 3D offset vector representing the distance vector to the closest object vertex in $\boldsymbol{v}^{\circ}$.

At training time, GNet's encoder, $\mathcal{E}^{\text{G}}$, maps the inputs X to the parameters of a normal distribution, $\{\boldsymbol{\mu}, \boldsymbol{\sigma}\} \in \mathbb{R}^{16}$. At inference time, we "skip" the encoder, and sample a latent whole-body grasp code, $\boldsymbol{z}_g \in \mathbb{R}^{16}$, from this distribution.

Output: GNet's decoder, $\mathcal{D}^{\text{G}}$, takes the grasp code, $\boldsymbol{z}_g$, and the input conditions for the object, $\boldsymbol{C} = [\boldsymbol{b}^{\circ}, \boldsymbol{\gamma}^{\circ}]$, and infers SMPL-X pose parameters $\hat{\boldsymbol{\Theta}}$, the head

direction vector $\hat{\boldsymbol{q}}$, and 3D offset vectors $\hat{\boldsymbol{d}}^{h\rightarrow\circ}$ from the N_h sampled hand vertices, $\boldsymbol{v}_h \subset \boldsymbol{v}$, to the closest object vertex. We call $\hat{\boldsymbol{q}}$ and $\hat{\boldsymbol{d}}^{h\rightarrow\circ}$ interaction features.

Output space: We make two empirical observations: (1) Networks struggle to predict accurate SMPL-X parameters, possibly due to their non-Euclidean space. (2) Networks predict interaction features in a Euclidean space much more precisely. These observations are in line with recent work [27, 154, 155]. However, we go beyond prior work by inferring 3D *offsets* together with SMPL-X parameters, instead of regressing vertex positions and fitting SMPL-X to these. We leverage these features in an optimization step to refine our SMPL-X pose predictions. We show a representation of the interaction features in Fig. 4.6.

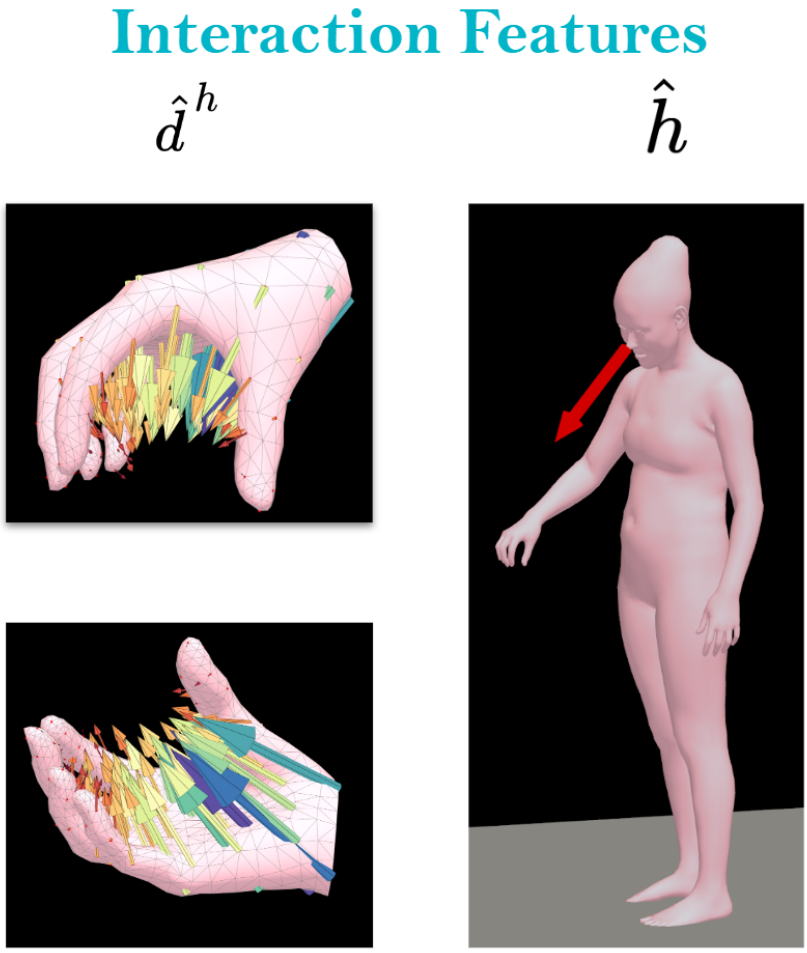


Figure 4.6: Visualization of the proposed interaction features (only the right hand is shown for simplification) used as the output of GNet. We show that the networks are able to generate these features more accurately than non-euclidean ones and that these features are complementary to the body pose for interaction.

Training: GNet is trained with the loss:

$$\begin{aligned}\mathcal{L}_{\text{GNet}} = \lambda_{\boldsymbol{v}}\mathcal{L}_{\boldsymbol{v}} + \lambda_{\boldsymbol{v}}^{h}\mathcal{L}_{\boldsymbol{v}}^{h} + \lambda_{\boldsymbol{\Theta}}\mathcal{L}_{\boldsymbol{\Theta}} + \\ \lambda_{\mathtt{q}}\mathcal{L}_{\mathtt{q}} + \lambda_{d}^{h\rightarrow\circ}\mathcal{L}_{d}^{h\rightarrow\circ} + \lambda_{KL}\mathcal{L}_{KL},\end{aligned} \tag{4.3}$$

where $\mathcal{L}_{\boldsymbol{v}} = \|\boldsymbol{v} - \hat{\boldsymbol{v}}\|_1$, $\mathcal{L}_{\boldsymbol{v}}^{h} = \|\boldsymbol{v}^{h} - \hat{\boldsymbol{v}}^{h}\|_1$, $\mathcal{L}_{\boldsymbol{\Theta}} = \|\boldsymbol{\Theta} - \hat{\boldsymbol{\Theta}}\|_2$, $\mathcal{L}_{\mathtt{q}} = \|\boldsymbol{q} - \hat{\boldsymbol{q}}\|_2$, $\mathcal{L}_{d}^{h\rightarrow\circ} = \|\boldsymbol{d}^{h\rightarrow\circ} - \hat{\boldsymbol{d}}^{h\rightarrow\circ}\|_1$, $\mathcal{L}_{KL}$ is the Kullback-Leibler divergence, and λ are weights. Hat variables are inferred; non-hat ones are ground truth. GNet's encoder and decoder use fully-connected layers with skip connections. For architecture details, see Fig. 4.5.

Optimization: We use the predicted offsets to refine our SMPL-X predictions with optimization post processing. Specifically, we *optimize* over SMPL-X parameters, $\boldsymbol{\Theta}$, initialized with GNet's predictions. Instead of hand-crafted contact constraints [17, 172, 181] during optimization, we use data-driven constraints *generated* from GNet, namely: (1) hand-to-object vertex offsets, (2) the head-orientation vector, (3) pose coupling to the initial value, and (4) foot-ground penetration. In technical terms, to refine the hand to realistically grasp the object, we define a term that penalizes differences between GNet's inferred offsets $\hat{\boldsymbol{d}}^{h\rightarrow\circ}$, and offsets $\boldsymbol{d}^{h\rightarrow\circ}$, computed online during the optimization, from SMPL-X's hand vertices to the closest object vertices:

$$\boldsymbol{E}_{\boldsymbol{d}}^{h\rightarrow\circ}(\boldsymbol{\theta}, \boldsymbol{\gamma}; \hat{\boldsymbol{d}}^{h\rightarrow\circ}) = \|\boldsymbol{d}^{h\rightarrow\circ} - \hat{\boldsymbol{d}}^{h\rightarrow\circ}\|_1. \tag{4.4}$$

Coupling for pose, $\boldsymbol{\theta}$, and translation, $\boldsymbol{\gamma}$, parameters discourages deviation from GNet's inferred values, $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\gamma}}$:

$$\boldsymbol{E}_{\boldsymbol{\theta}}(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}) = \|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}\|_2, \quad \boldsymbol{E}_{\boldsymbol{\gamma}}(\boldsymbol{\gamma}; \hat{\boldsymbol{\gamma}}) = \|\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}\|_1. \tag{4.5}$$

Similarly, head-orientation coupling is formulated as:

$$\boldsymbol{E}_{\mathtt{q}}(\boldsymbol{\theta}, \boldsymbol{\gamma}; \hat{\boldsymbol{q}}) = \|\boldsymbol{q}(\boldsymbol{\theta}, \boldsymbol{\gamma}) - \hat{\boldsymbol{q}}\|_1. \tag{4.6}$$

To encourage ground contact and discourage penetration, given the current SMPL-X parameters in every optimization step, we find, online, the lowest vertex of the body along the "y" vertical axis and encourage its y-coordinate to be zero:

$$\boldsymbol{E}_{\mathtt{f}} = |\boldsymbol{v}_y(k)|, \quad k = \arg\min_i \boldsymbol{v}_y(i), \tag{4.7}$$

where i and k are vertex indices in the body mesh. Our final energy is a combination of the above terms:

$$\boldsymbol{E}_{\text{GNet}} = \lambda_d^{h\to\circ}\boldsymbol{E}_d^{h\to\circ} + \lambda_{\boldsymbol{\theta}}\boldsymbol{E}_{\boldsymbol{\theta}} + \lambda_{\gamma}\boldsymbol{E}_{\gamma} + \lambda_{\mathbf{q}}\boldsymbol{E}_{\mathbf{q}} + \lambda_f\boldsymbol{E}_f. \tag{4.8}$$

We do the optimization using Adam optimizer with a learning rate starting from $10e^{-2}$ and gradually decreasing on plateau until the gradient is lower than $10e^{-5}$.

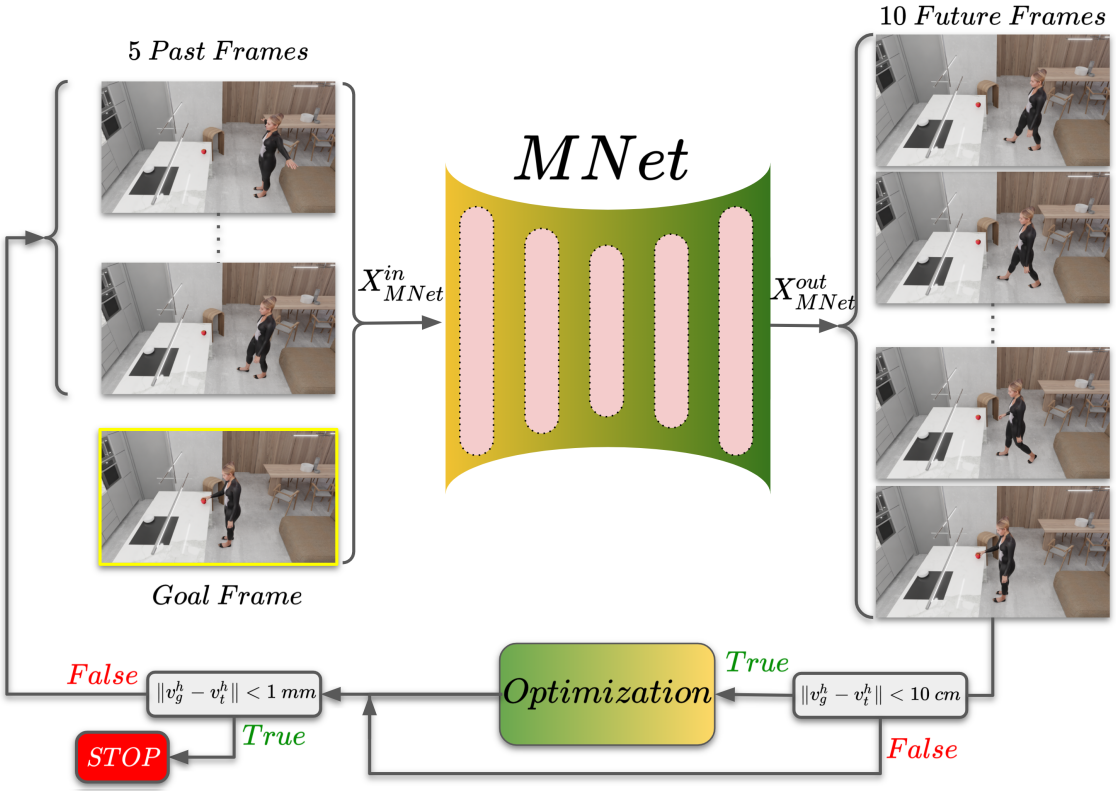


Figure 4.7: Architectural overview of the MNet network, as well as the optimization post-processing step (bottom part).

4.4.4 MNet - Motion Network

In Fig. 4.7 we show the architectural overview of MNet and its optimization-based post-processing. MNet is an auto-regressive network that generates the body motion in several iterations. It generates the motion from the "start" to the "goal" frame; the latter is generated by GNet, described above in Sec. 4.4.3. The length of a sequence depends on several factors, such as the object location w.r.t. the body

and motion speed. Therefore, to generate motion of arbitrary duration, we use an auto-regressive network architecture [37, 171].

Input: MNet takes as input (auto-regressive fashion):

$$\boldsymbol{X}^{in}_{\text{MNet}} = [\boldsymbol{\Theta}_{t\text{-}5:t}, \boldsymbol{\beta}, \boldsymbol{v}_t, \dot{\boldsymbol{v}}_t, d^h_{t\rightarrow g}, \boldsymbol{b}^h_g], \tag{4.9}$$

where t is the current frame, $\boldsymbol{\Theta}_{t\text{-}5:t}$ are SMPL-X parameters of the last 5 frames, $\boldsymbol{\beta}$ is the subject's shape, $\boldsymbol{v}_t$ and $\dot{\boldsymbol{v}}_t$ are the locations and velocities of the N_b sampled body vertices in the current frame, and $\boldsymbol{d}^h_{t\rightarrow g}$ are the hand vertex offsets from the current frame, t, to the "goal" frame, g. Finally, $\boldsymbol{b}^h_g$ is the BPS representation [101] of the hand in the "goal" grasping frame using the same BPS basis points as for the object; using the same basis points for the BPS representation encodes the spatial relationship between the hand and the object in the "goal" frame, and is empirically important for "guiding" the motion towards a realistic grasp.

We empirically find that using as input the pose, $\boldsymbol{\Theta}$, of more than 1 past frame leads to a smoother motion prediction, in agreement with Starke et al. [37]; using more than 5 frames does not lead to noticeable improvement.

Output: MNet produces as output:

$$\boldsymbol{X}^{\text{out}}_{\text{MNet}} = [\Delta\boldsymbol{\Theta}_{t:t+10}, \Delta\boldsymbol{v}_{t:t+10}, \Delta\boldsymbol{d}^h_{t:t+10}] \tag{4.10}$$

where $t : t{+}10$ is the future 10 frames, $\Delta\boldsymbol{\Theta}_{t:t+10}$ is the change of SMPL-X parameters, $\Delta\boldsymbol{v}_{t:t+10}$ is the change of SMPL-X vertex positions, and $\Delta\boldsymbol{d}^h_{t:t+10}$ is the change of hand vertex offsets. All changes, Δ, are relative to the current frame, t.

Output space: MNet focuses both on SMPL-X parameters and Euclidean-space interaction features, similarly to GNet. This empirically helps inference; the generated motion is smoother and better "reaches" the "goal" grasp.

Auto-regression: MNet estimates SMPL-X parameters for 10 future poses, and then we use them to compute the features of the last 5 frames and feed them back to MNet as inputs for the next iteration, along with other inputs shown in Eq. (4.9). For architecture details, see Fig. 4.7. Unlike HuMoR [182] where in each iteration only 1 future frame is generated, MNet's generated motion improves when generating more number of future frames; note that the improvement saturates for 10 future frames, see Tab. 4.2-right in Sec. 4.5.2.

Training: MNet is trained with the loss:

$$\mathcal{L}_{\text{MNet}} = \lambda_{\boldsymbol{v}}\mathcal{L}_{\boldsymbol{v}} + \lambda_{\boldsymbol{v}}^{h}\mathcal{L}_{\boldsymbol{v}}^{h} + \lambda_{\boldsymbol{\Theta}}\mathcal{L}_{\boldsymbol{\Theta}} + \lambda_{\boldsymbol{d}}^{h\rightarrow\circ}\mathcal{L}_{\boldsymbol{d}}^{h\rightarrow\circ} + \lambda_{\boldsymbol{v}}^{\mathtt{f}}\mathcal{L}_{\boldsymbol{v}}^{\mathtt{f}}, \tag{4.11}$$

where the losses on body vertices, $\mathcal{L}_{\boldsymbol{v}}$, hand vertices, $\mathcal{L}_{\boldsymbol{v}}^{h}$, SMPL-X parameters, $\mathcal{L}_{\boldsymbol{\Theta}}$, and hand-to-object offsets, $\mathcal{L}_{\boldsymbol{d}}^{h\rightarrow\circ}$ are borrowed from Eq. (4.3). Finally, $\mathcal{L}_{\boldsymbol{v}}^{\mathtt{f}} = \|\boldsymbol{v}^{\mathtt{f}} - \hat{\boldsymbol{v}}^{\mathtt{f}}\|_1$ is a new loss on foot vertices that are close to the ground. This loss and the input velocities of Eq. (4.9) help foot-ground contact and reduce sliding; see the video on our website.

Optimization: We refine MNet's generated motion with a post-processing optimization such that the final hand grasp gets closer to the "goal" grasp generated by GNet. Since we need precision only when the hand is close to the object, we conduct optimization only when MNet's estimated hand vertices get closer than 10 cm to the "goal" hand vertex positions. Following GNet's scheme, we use MNet outputs listed in Eq. (4.10) as constraints, instead of hand-crafted ones. Specifically, we first compute the average per-vertex velocity of MNet's predicted hands, $\dot{\boldsymbol{v}}_t^h$. Then, we linearly interpolate hand vertices for the next frame, $\mathbf{v}_{t+1}^h$, between the current, $\mathbf{v}_t^h$, and "goal" ones, $\mathbf{v}_g^h$:

$$\mathbf{v}_{t+1}^h = \mathbf{v}_t^h + \|\dot{\mathbf{v}}_t^h\|\mathbf{l}, \quad \text{where} \quad \mathbf{l} = \frac{\mathbf{v}_g^h - \mathbf{v}_t^h}{\|\mathbf{v}_g^h - \mathbf{v}_t^h\|} \tag{4.12}$$

where $\|\dot{\mathbf{v}}_t^h\|$ is the average-velocity magnitude, and $\mathbf{l}$ is a unit vector pointing from current to "goal" hand vertices. In practice, we "force" hands to move towards the "goal" grasp in a locally linear trajectory; this is simple and intuitive, but might result in some hand-object penetrations. Since our focus here is the hand grasp, for the rest of the body we keep MNet's predicted pose and velocity intact.

The energy function for optimization is:

$$\boldsymbol{E}_{\text{MNet}} = \lambda_{\boldsymbol{\Theta}}\boldsymbol{E}_{\boldsymbol{\Theta}} + \lambda_{\boldsymbol{v}}^{h}\boldsymbol{E}_{\boldsymbol{v}}^{h}, \tag{4.13}$$

where the term $\boldsymbol{E}_{\boldsymbol{\Theta}}$ is on SMPL-X parameters, and $\boldsymbol{E}_{\boldsymbol{v}}^{h}$ is on hand vertices; their definition is similar to $\mathcal{L}_{\boldsymbol{v}}$ and $\mathcal{L}_{\boldsymbol{\Theta}}$ from Eq. (4.3).

For both GNet's and MNet's optimization step, we refine the inferred SMPL-X bodies with gradient descent using Adam [183].

4.4.5 Data Preparation

We use the GRAB dataset from Chapter 2 for training and evaluating the models.

GNet: GNet generates static whole-body grasps. Therefore, from the GRAB dataset, we collect all frames with right-hand grasps, for which subjects grasp the object in a stable way. For this, we follow the selection criteria used for GrabNet's [41] training data in Chapter 3. We then center the object at the origin along the horizontal plane, i.e., while preserving its height. This is important as the object height changes the body pose for grasping. In total, we collect $160K$, $26K$, and $12.5K$ frames for the training, testing, and validation set, respectively.

MNet: On the other hand, since MNet generates motion, from each sequence of GRAB, we gather all frames from the starting one up to the frame where the right hand first establishes a stable grasp. For this, we use the same selection criteria as above for GNet. We then create several sub-sequences by sliding a 21-frame long window over each sequence with a stride of 1 frame. For each sub-sequence, we consider the first 10 frames as "past" motion, the last 10 frames as "future" motion, and the middle one as the "current" frame. Please note that after experiments we found that only 5 past frames are sufficient to get accurate results from MNet. Then, following Starke et al. [37], we make all "past" and "future" frames relative to the body coordinate system of the "current" frame, while keeping the gravity direction always upward. In total, we collect roughly $40K$, $7K$, and $3K$ motion sub-sequences for the training, testing, and validation sets, respectively.

4.5 Experiments

4.5.1 Qualitative Evaluation

GNet: Figure 4.8 shows representative generated static grasps before and after optimization. Before optimization, body and head poses are plausible, but hand grasps can be improved (pink). The optimization refines hands for more realistic

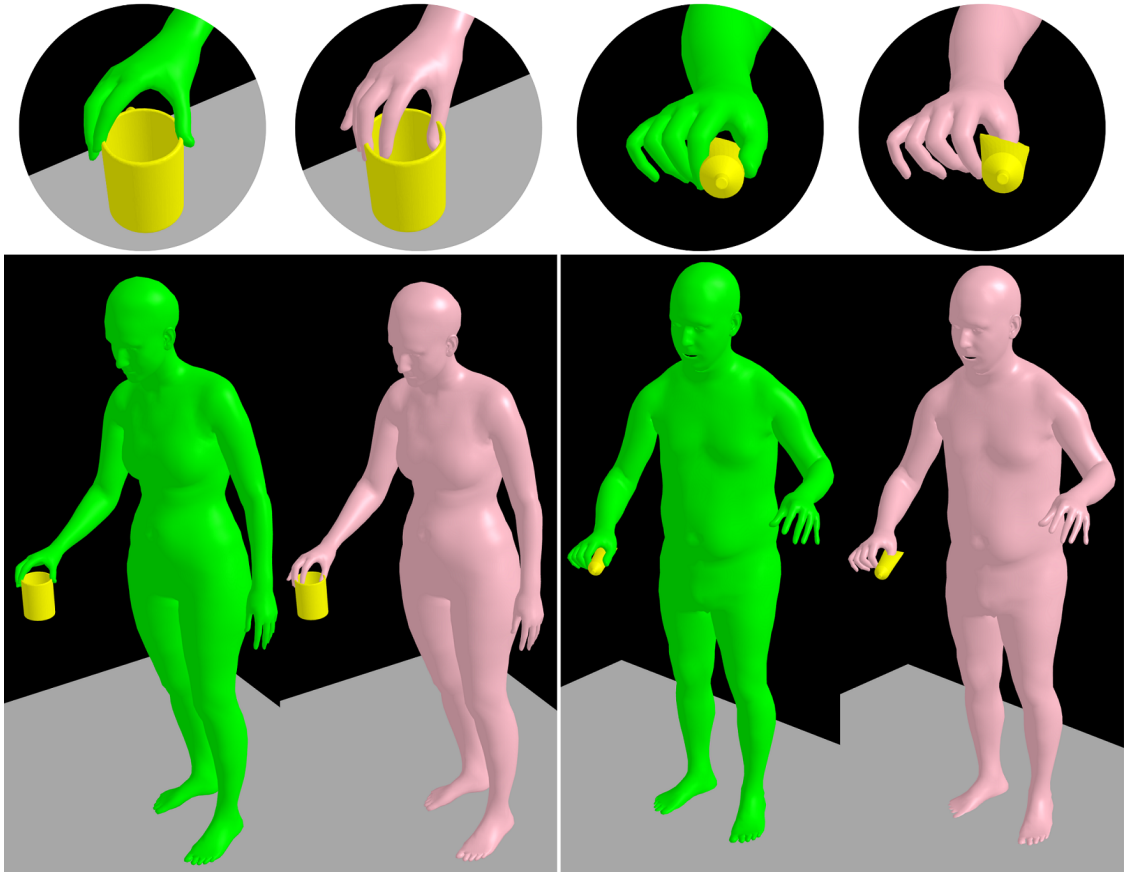


Figure 4.8: GNet's generated SMPL-X grasp poses (Sec. 4.4.3) before (pink) and after optimization (green). Results show that optimization-based post-processing effectively refines the initial prediction towards a more realistic and physically plausible grasp.

and physically plausible grasps (green). Figure 4.9 shows GNet 's generalization ability with grasps generated for 2 unseen and complex objects from the YCB dataset [184].

MNet: Figure 4.10 shows motions generated for several test object shapes and locations, body shapes and, "start" poses. In Fig. 4.11 we show more qualitative results of GOAL from different views and with close-ups on hands. The results show the accuracy of the generated motions and grasp. We observe that GOAL can generalize well to new unseen input conditions, including object shape, location, and body starting pose. Despite this, we observe some rare failure cases where our results have penetration artifacts, which are shown in Sec. 4.5.5.

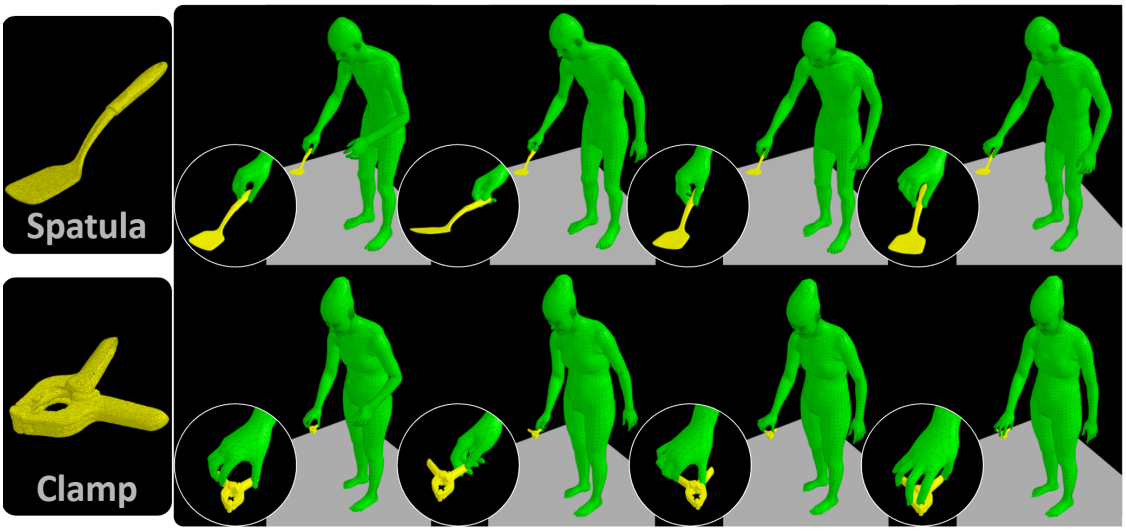


Figure 4.9: We show how GOAL generalizes by sampling 4 grasps from GNet's latent space for 2 complex, unseen objects from the YCB [184] dataset.

4.5.2 Quantitative Evaluation

GNet: Table 4.1 reports the penetration volume (cm^3) and contact ratio [81] of four models: (1) "GrabNet" [41], which generates MANO grasps, (2) "GrabNet-SMPL-X", a variant that uses SMPL-X, (3) GNet without optimization, and (4) "GNet" with optimization. Here the penetration volume is calculated by determining the intersection between two meshes and subsequently measuring its volume in cubic centimeters (cm^3). The contact ratio, on the other hand, is derived by identifying the number of ground truth contact frames, establishing the number of contact frames present in the generated grasps, and computing the ensuing percentage. We see that generating whole-body grasps (row 2) is harder than hand-only grasps (row 1), yet "GNet" (row 4) outperforms baselines. Thus, post-processing optimization helps improve contact and reduce penetration volume. Note that small penetrations are inevitable as SMPL-X does not model soft tissue deformation; see [42].

Foot sliding: We evaluate foot-ground contact with a "foot sliding" metric. For each frame, we find the closest body vertex to the ground and compute its velocity. For contact vertices velocity should ideally be zero; if it is higher than 1 *cm* per frame, we consider the foot to be "sliding". GOAL generates sequences with 13.7% "foot-sliding" frames; GRAB's ground truth has 6.7%. Although there

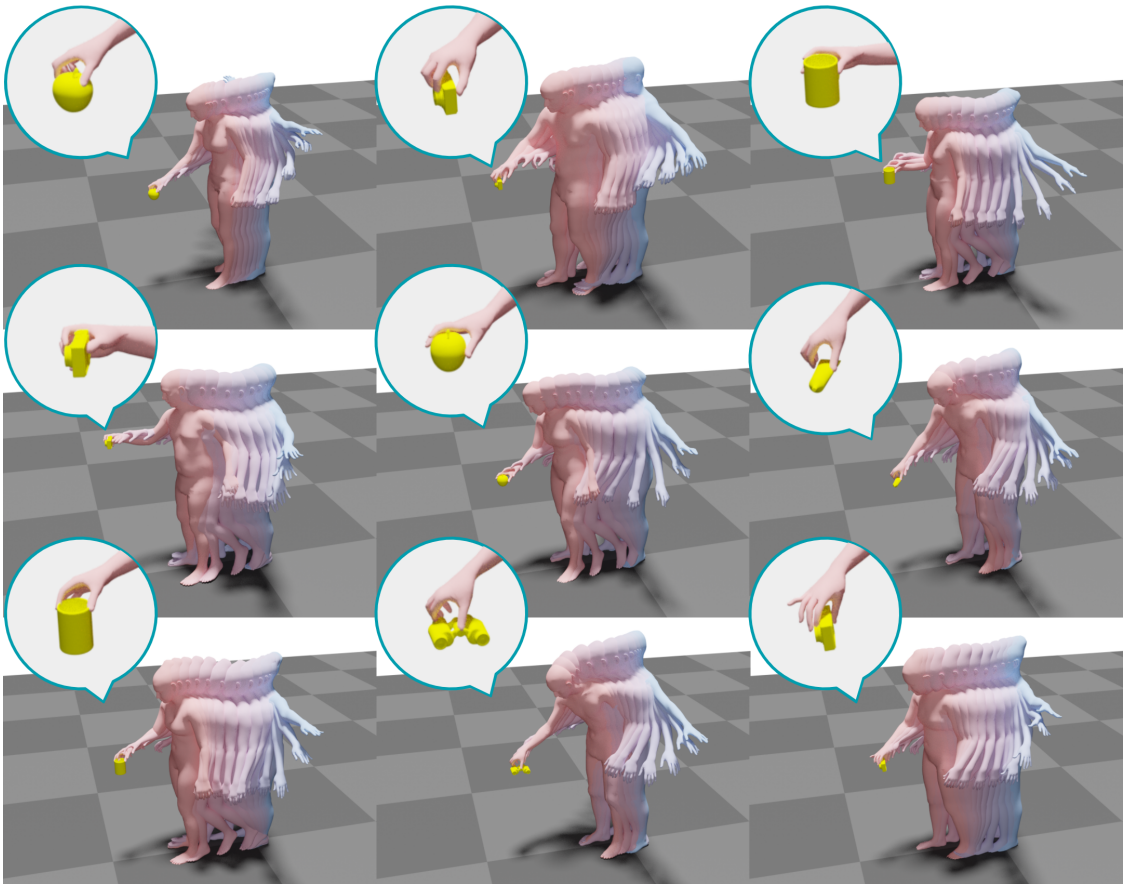


Figure 4.10: Representative motions generated by GOAL (GNet and MNet) for several test-object shapes, locations, body shapes, and "start" poses.

	Grasp Synthesis	Penetration Vol. (cm^3) ↓	Contact Ratio [81]
1.	GrabNet [41]	2.65	**1.00**
2.	GrabNet-SMPL-X	7.33	0.87
3.	GNet-w/o-opt	5.32	0.87
4.	GNet (ours)	**2.22**	**1.00**
	GRAB (GT)	1.95	1.00

Table 4.1: Penetration and contact-ratio evaluation for GNet. We compare GNet with-/out optimization and GrabNet variants.

is room for improvement, GNet's feet "slide" less than existing work [27, 171], which is around 27% (on other datasets).

4.5.3 Ablation Study

MNet – IAA & output features: Tab. 4.2 compares MNet with similar models that infer: (1) only SMPL-X pose parameters, "MNet-Pose", (2) only markers akin

Figure 4.11: Results generated by GOAL shown from different views with close-ups of the hand grasps. The results represent the smooth motion of the human body, hands, and head approaching to grasp a 3D object.

to MOJO [27], "MNet-Marker", and (3) MNet outputs without Interaction-Aware Attention, "MNet-w/o-IAA". We report vertex-to-vertex (V2V) errors for the full body, hands and feet on GRAB's held-out test set. Errors drop with: (1) using IAA features and (2) jointly inferring SMPL-X pose and marker offsets as output. We empirically observe that our combination of inputs and outputs, inspired from work on character control [37], leads to more realistic results.

Motion Network	V2V (mm) ↓		
	Body	Hand	Feet
1. MNet-Pose	22.0	30.2	10.4
2. MNet-Marker [27]	21.1	30.1	**9.8**
3. MNet-w/o-IAA	21.0	29.1	10.5
4. MNet (ours)	**19.7**	**28.0**	9.9

Table 4.2: Effect of MNet outputs and use of "Interaction Aware Attention" (IAA) on the V2V error. "Pose" refers to SMPL-X pose parameters, and "Markers" to a MOJO-like [27] setup for the whole body. The results show a reduction in errors by using our output spaces (row 3) and using IAA features (row 4).

Number of	V2V (mm) ↓		
Output Frames	Body	Hand	Feet
1	26.7	35.7	17.9
2	21.5	29.6	13.2
3	20.3	28.5	12.2
5	19.7	28.1	10.5
10	**19.7**	**28.0**	**9.9**

Table 4.3: Effect of MNet's number of output frames on the performance. As the number of MNet's output frames increase, results improve but saturate around 10 frames.

MNet – Number of output frames: We train 5 networks with outputs ranging from 1 to 10 frames, and report in Tab. 4.3 the vertex-to-vertex (V2V) error for the body, feet, and hands between the generated and ground-truth meshes. Results show that generating more frames in each iteration of our auto-regressive scheme helps generate better results. We empirically observe that, when inferring a small number of future frames, sometimes the motion does not converge to a grasp and hands gradually deviate away from the object, instead of contacting it.

4.5.4 Perceptual Evaluation

We evaluate GNet and MNet by generating grasping poses and motions, respectively, on GRAB's test set, and running a perceptual study via Amazon Mechanical Turk.

Metric	GNet	GNet + Opt	Ground-truth [41]
Overall Grasping Pose ↑	3.89 ± 0.93	$\mathbf{3.98} \pm 0.94$	3.78 ± 1.06
Foot-Ground Contact ↑	3.98 ± 1.06	$\mathbf{4.10} \pm 0.93$	3.82 ± 1.11
Hand-Object Grasp ↑	2.70 ± 1.37	3.63 ± 1.16	$\mathbf{3.98} \pm 1.04$
Head Orientation ↑	3.83 ± 1.01	$\mathbf{4.01} \pm 0.97$	3.84 ± 1.07
Average ↑	3.60 ± 1.22	$\mathbf{3.93} \pm 1.02$	3.86 ± 1.07

Table 4.4: Perceptual study for GNet with-/out optimization. Subjects rate the realism of the grasp from 1 (unrealistic) to 5 (very realistic). We report the average rating value ± the standard deviation, computed across all valid participants. The optimization step ("GNet + Opt") improves all four studied features.

GNet: For each test-set object, we generate 2 "goal" whole-body grasps and render a "turntable animation" of these, before and after optimization, and the corresponding ground-truth grasps. Participants rate the quality of 4 features: (1) grasping pose, (2) foot-ground contact, (3) hand-object grasp, and (4) head orientation. They rate the realism of each feature on a Likert scale of scores between 1 (unrealistic) to 5 (very realistic). Each grasp is evaluated by at least 10 Participants. To remove invalid ratings, e.g., those who do not understand the task, we use catch trials similar to Chapter 3. Results are shown in Tab. 4.4. The optimization step improves the realism of whole-body grasps. Moreover, it improves head orientation compared to the ground truth; this is because some GRAB subjects look away from the object while grasping it, while GNet produces heads oriented towards the object due to the explicit head orientation, $\boldsymbol{q}$. The same applies also for foot-ground contact, due to the explicit foot term, $\mathcal{L}_v^{\mathbf{f}}$, in Eq. (4.11). Overall, the quality of the generated grasps is close to the ground truth.

MNet: We show generated and ground-truth sequences to participants, and ask them to rate the quality of: (1) overall body motion, (2) foot-ground contact, (3) hand-object grasp at the end of motion, (4) head orientation. Table 4.5 shows the results; GOAL generates grasping motions that approach the realism of ground truth. Note that MNet has a harder task than GNet as it generates a full motion. Also, Tabs. 4.4 and 4.5 show that ground truth is rated higher for motions than for static poses; this is harder for MNet to match.

Metric	GOAL	Ground-truth [41]
Overall Body Motion ↑	3.74 ± 0.97	$\mathbf{4.20} \pm 0.90$
Foot-Ground Contact ↑	3.88 ± 1.14	$\mathbf{4.18} \pm 1.05$
Final Hand-Object Grasp ↑	3.66 ± 1.05	$\mathbf{4.32} \pm 0.91$
Head Orientation ↑	3.86 ± 1.03	$\mathbf{4.18} \pm 1.00$
Average ↑	3.79 ± 1.05	$\mathbf{4.22} \pm 0.97$

Table 4.5: Evaluation of MNet motions. Participants rate the generated and ground-truth motions on a Likert scale of 1 (unrealistic) to 5 (very realistic) for 4 factors:overall body motion realism, feet-ground contact, final hand-object grasp, and head orientation.

4.5.5 Failure Cases

Despite generating mostly realistic motions, the MNet optimization sometimes results in small hand-object penetration before the "goal" grasping frame; we show two examples in Fig. 4.12. This is due to linearly interpolating the motion between the "current" and "goal" frames during optimization, and could be solved in future work by adding a penetration loss, and potentially by replacing the linear interpolation with a more involved approach.

In addition, in some cases we observe "foot sliding", especially when the "starting" body is placed further than $1.5m$ from the object. Figure 4.13 shows some "foot-sliding" cases in comparison to the ground-truth motion. While our main focus here is to generate grasping motion, future work should look into combining GOAL with longer walking-motion generation methods [27, 171].

4.6 Conclusion

We introduce GOAL, the first model to generate realistic human motions to grasp previously unseen 3D objects. We use two novel networks (GNet and MNet) to first generate a static "goal" grasp and then inpaint the motion between the frames. We exploit the ability of both networks to infer interaction features in Euclidean space and introduce an optimization step after each network to improve the quality of the grasps and motion based on the regressed features. The evaluation shows that our framework is able to synthesize natural and physically plausible grasping motions.

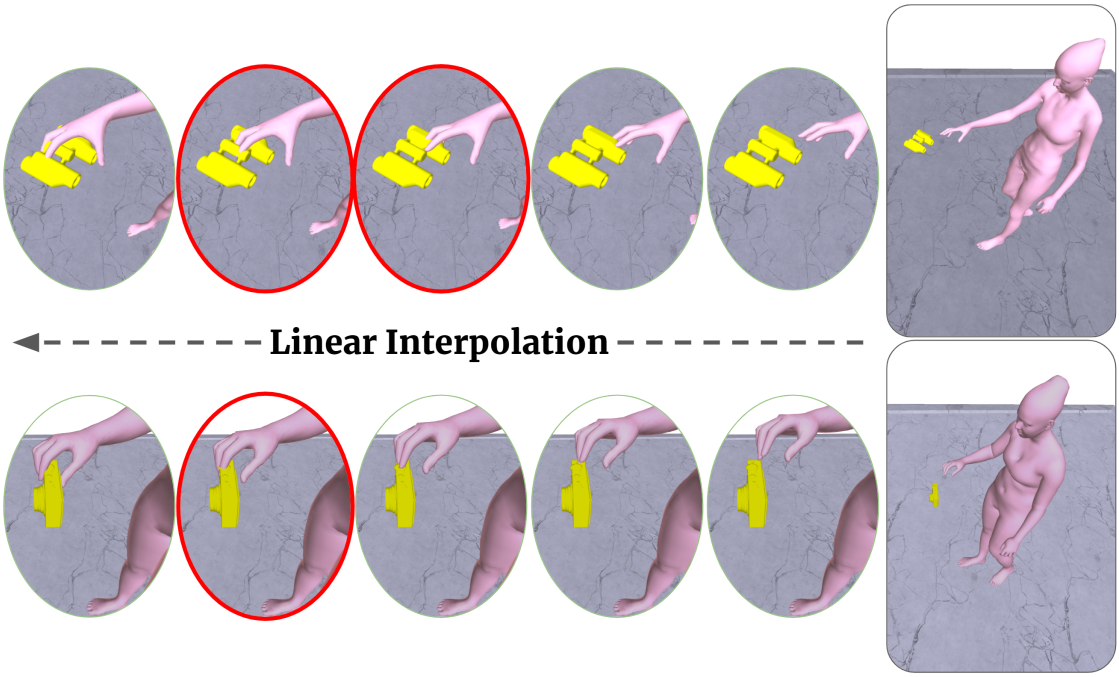


Figure 4.12: Two penetration failure cases during MNet's optimization post-processing with linear interpolation. In the figure the hand approaches the object from right to left, and the red ovals highlight hand-object penetrations.

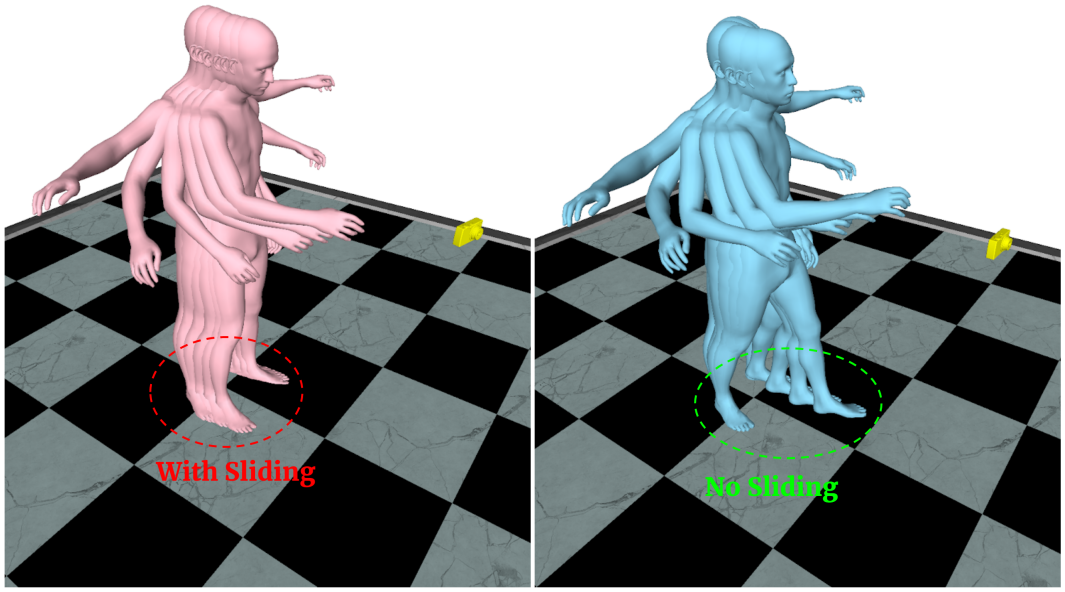


Figure 4.13: A failure case of "foot sliding" generated by MNet (left), and compared to the corresponding ground-truth motion (right). Note that for the ground truth (right) the right foot maintains contact with the floor, while the left foot moves in the air for walking.

Future work: GOAL opens up many possibilities for future studies on grasping motion generation. Even though GOAL generates realistic grasping motions, it is

constrained to be close to the object and can not generate motions when the body is far away. We plan to extend this to synthesize longer walking motions, prior to interaction with objects. In addition, here we focus on human-object interaction; we plan to combine GOAL with human-scene interaction models.

Details matter, it's worth waiting to get it right.

— Steve Jobs

5

GRIP: Generating Hands Motion for Object Interaction

Contents

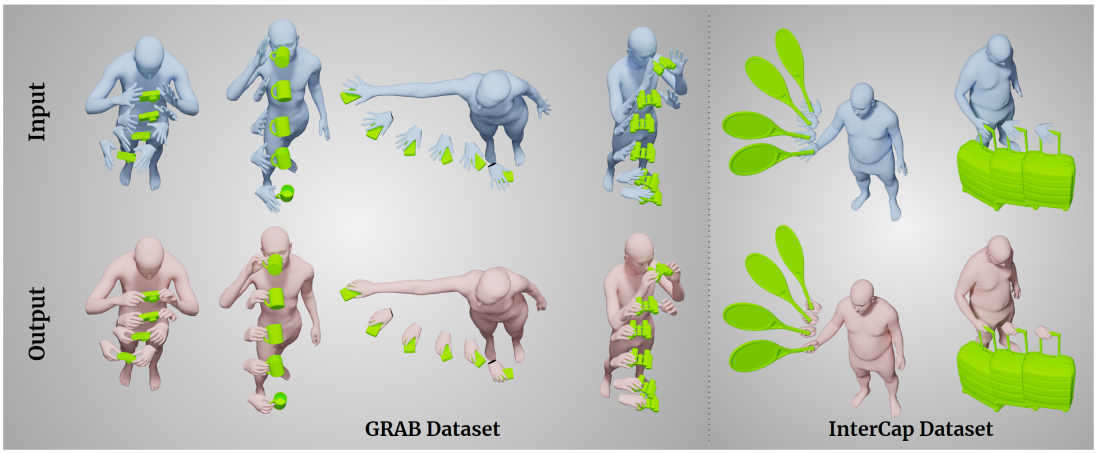


Figure 5.1: GRIP generates realistic hand-object interaction poses (pink), given the easy-to-acquire body and object motion without fingers (blue) – notice that the input hand pose is constant. GRIP animates the hands to be consistent with the body and object, producing realistic poses in various scenarios like pre-/post-grasp hand opening, and single or bi-manual grasps. It also works with various object shapes and sizes, and on different datasets like GRAB [41] (left) and InterCap [185] (right).

5.1 Overview

Hands are dexterous and highly versatile manipulators that are central to how humans interact with objects and their environment. Consequently, modeling realistic hand-object interactions, including the subtle motion of individual fingers, is critical for applications in computer graphics, computer vision, and mixed reality.

Prior work on capturing and modeling humans interacting with objects in 3D focuses on the body and object motion, often ignoring hand pose. We go one step ahead in GrabNet and GOAL works (Chapters 3 and 4) where we focus on generating hand poses as well. However, we still only consider the pre-interaction motions up to the grasping moment, i.e. before the hands interact with the objects and move them. To overcome this, we introduce GRIP, a learning-based method that takes, as input, the 3D motion of the body and the object, and synthesizes realistic motion for both left and right hands before, during, and after object interaction. Our goal is to add realistic hand motions to a body, based on the relative motion of the body and object during an interaction.

As a preliminary step before synthesizing the hand motion, we first use a network, ANet, to denoise the arm motion. Then, we leverage the spatio-temporal relationship

between the body and the object to extract two types of novel temporal interaction cues, and use them in a two-stage inference pipeline to generate the hand motion. In the first stage, we introduce a new approach to enforce motion temporal consistency in the latent space (LTC) and generate consistent interaction motions. In the second stage, GRIP generates refined hand poses to avoid hand-object penetrations. Given sequences of body and object motion during an interaction task, GRIP "upgrades" them to include hand poses for hand-object interaction. Quantitative experiments and perceptual studies demonstrate that GRIP outperforms baseline methods and generalizes to unseen objects and motions from different motion-capture datasets. Our models and code will be available for research purposes at https://grip.is.tue.mpg.de.

5.2 Introduction

Digital humans that move and interact naturally with virtual 3D worlds have many applications in synthetic data creation, games, XR, and telepresence. In particular, physically plausible hand-object interaction is critical for realism. Unfortunately, automatically generating hand motions that are consistent with the world is challenging and no fully general solutions exist. Prior work on capturing and modeling humans interacting with objects in 3D focuses on the body and object motion, often ignoring hand pose, which plays a critical role in understanding how humans interact with objects. In our previous works GrabNet and GOAL (Chapters 3 and 4), we started addressing this gap by integrating hand pose modeling. While this was a step forward, our focus remained on the pre-interaction motions, specifically up to the moment of grasping. This means we mainly studied the movements leading up to when the hands touch the objects, but not what happens after, such as how the hands manipulate or move the objects.

The problem is challenging since different object shapes require different types of interaction and hand grasps, such as a power grasp to grab an apple, a delicate three-finger pinching of a cup handle, and bimanual collaboration to hold a pair of binoculars with two hands. Performing these actions is effortless for adult humans; however, even small errors, such as hand-object penetrations or subtly

misplaced fingers, can significantly affect the perceived realism of generated grasps for virtual avatars.

Here we consider generating realistic grasps where a 3D animation of the body and object is given, either from motion capture (MoCap), reconstructed from videos, or from an animator. In contrast to GRAB, motion capture data rarely contains hands as they are difficult to capture, requiring small markers that are often occluded and require high-resolution cameras. In some cases, despite being tracked, hands and arms are very noisy [185]. Objects, in contrast, are easier to track. Figure 5.1 (top) illustrates this scenario with body and object motion, from GRAB [41] and InterCap [185], but with only rigid hands. The goal is to transform this data into a more natural animation by synthesizing the appropriate hand-object interaction, as illustrated in Fig. 5.1 (bottom). With this approach we can "upgrade" existing datasets to support research on human-object interaction.

To this end, we introduce *GRIP*, which stands for *GeneRating Interaction Poses*, a learned model that generates realistic hand motions for interactions with a variety of previously-unseen objects. Previous work in this direction focuses only on static grasping [41, 42], requires an initial hand pose that is then improved [186], or only considers single-hand grasps [186, 187]. Going beyond these approaches, our method directly infers dynamic hand motion, both in a single-hand or bimanual scenario, conditioned only on the object and body motion.

Our contributions are two-fold. First, we propose a set of virtual "hand sensors" to extract rich spatio-temporal interaction cues between the body and the object. Specifically, we introduce an *Ambient Sensor* that senses the object shape and motion within the hands' broader reaching region, as well as a *Proximity Sensor* that captures fine-grained geometric features and a more nuanced distance field between the hand and object surface within the hands' closer region. While virtual sensors have been used in prior work, our novel contribution is the innovative use of a distance-based representation combined with interaction-aware attention [188]. This unique combination significantly improves results and generalizes to unseen objects and motions.

Second, we propose an *arm denoising* network and a novel two-stage *hand inference* pipeline to leverage these features and generate realistic interaction motions. Since arm motions from tracking or reconstruction can be noisy, we first

use an arm denoising network, ANet to refine arm motion. For the hand inference, our goal is to achieve near real-time performance, therefore, to avoid iterative optimization, like previous methods, we design two networks. First, the *Consistency Network (CNet)* takes features from both *hand sensors* and generates smooth and consistent hand interaction motions. Achieving this is challenging, as motions need to be realistic, temporally consistent, and natural. Naively applying temporal smoothness terms to the final output hand motion, cf. [37, 188], will break the contact consistency and lead to high-frequency changes in contact areas. To overcome this, we propose a novel *Latent Temporal Consistency (LTC)* solution. Specifically, we jointly learn global and residual latent codes to represent two successive frames and apply temporal consistency in the latent space, as shown in Fig. 5.4. Then, to mitigate any inconsistency between the two global latent codes, the key insight is to decode them using a "shared" network to generate consistent hand poses. We use LTC in both ANet and CNet to ensure consistency in the motions.

The generated hand poses from CNet bring fingers very close to the object surface, allowing the *Proximity Sensor* to capture a more nuanced distance field. Therefore, in the second stage, we recompute the *Proximity Sensor* features and use a refinement network, RNet, to add subtle refinements and resolve penetrations in the interaction frames.

GRIP is trained to generate both left- and right-hand motion simultaneously, enabling realistic modeling of single-hand and bi-manual interactions. In contrast to other methods, which only focus on contact frames [41, 42], our model is able to generate dynamic hand motions *before*, *during*, and *after* the interaction with objects. Additionally, unlike [186], which requires expensive optimization in the pose refinement step, our framework consists only of feed-forward neural networks. By predicting realistic hand and finger motions, GRIP can be used to increase the realism of an avatar's interaction in AR/VR applications, refine noisy hand-object interaction motions (Fig. 5.12-top), enrich existing interaction datasets that do not contain realistic finger motions (Fig. 5.12-bottom), or capture new datasets with dexterous interactions but without explicitly tracking fingers.

We evaluate GRIP quantitatively and qualitatively on a withheld test set from the GRAB dataset, with 5 unseen objects and motions. The results show that our method generates accurate hand motions involving object grasping and

manipulation. We also show that GRIP generalizes to other MoCap datasets and larger objects, not present in GRAB, by generating hand grasps for unseen objects from the MoGaze [189] and InterCap [185] datasets (see Fig. 5.12). The quantitative evaluation shows that GRIP outperforms baselines, while our ablation studies explore the efficacy of our *latent temporal consistency*, *hand sensors*, and other design choices. Finally, we perform a perceptual study to evaluate the quality of the generated hand interaction motions. The results indicate that hand-object interaction sequences generated by GRIP achieve a level of realism similar to GRAB's ground-truth motions.

5.3 Related Work

Despite the many advances in the field of motion synthesis for human avatars, generating accurate hand motion is still a challenging and unsolved problem. While many approaches focus on improving static grasps [41, 42] with manually designed heuristics [17, 150], more recent techniques consider dynamic grasp generation [186, 187]. Such methods are still limited in computation time, bimanual operations, and the generalization to novel objects and grasping patterns. We review the most relevant methods below.

Static Grasp Generation: Generating static grasps has been widely studied in robotics, computer graphics, and computer vision. Common approaches in graphics and robotics use physics-based control to generate novel hand grasps for a given 3D object. This includes using reference poses to optimize generated grasps [38], using hand pose and force closure [39, 190], or pruning grasp candidates through physics-based analysis [191–194]. Some recent methods take a data-driven approach and generate hand grasps by training on large hand-object interaction datasets [13, 41, 108, 181, 195–198]. Most of these approaches either estimate the grasping-hand pose directly [13, 108, 195], based on model parameters [5, 85] or by employing an implicit representation [186, 197]. Other approaches further refine the initially generated grasps by using a neural network [41] or by leveraging predicted contact maps [42, 195].

Dynamic Grasp Generation: Generating hand-object grasping motions is more challenging than static grasp generation. Most previous methods approach

this by generating contact constraints and by resolving them through optimization-based methods [40, 193, 199–201]. Despite being physically plausible, the generated hand motions lack realism and are prone to interaction artifacts. More recently, reinforcement learning (RL) has been used for hand-only and full-body scenarios [32, 202–206]. Christen et al. [207] employ physics simulation along with RL for dynamic grasp synthesis; however, their method requires reference hand-grasps and dynamic features of the object. A key challenge of these methods is generalization to new object geometries and hand configurations. Zhang et al. [187] use a distance-based spatial representation between hands and objects and train a network to generate right-handed object manipulation motions. To avoid interaction artifacts, [186] propose an object-centric spatio-temporal representation and refine it with a neural network. The refined representation is then used in an optimization step to recover the hand-interaction motion. Unlike our approach, most of these methods treat each hand separately, making generated hand-collaboration and bi-manual grasps unrealistic.

Object and Scene Interaction: Some early work uses foot and hand contact annotations from MoCap datasets with optimization-based methods to extend or retarget human motions to scenes [28–31]. Alternatively, deep reinforcement learning can be used to generate body-scene [32–34] or hand-object [167, 168, 207] interactions. Other methods use descriptors for dynamic interactions [99, 208], encode the joint motions of humans w.r.t. scene points [169], or use Laplacian deformation between body and object vertices to define a representation for modeling interactions [170]. As geometry-based approaches are not robust to real-world noise, some methods take a data-driven approach to predict action and motion sequences [209] or to generate key frames of motions and then complete them with data-driven or optimization-based techniques [171, 172, 188].

Hand-Object Interaction Tracking: For graphics applications, hand motions have traditionally been animated by artists [187]. While MoCap can be used to capture hand motion datasets [13, 56, 68, 108, 210], such captures are technically challenging, limiting the amount of such data in the world. For the MoGaze [189], KIT [20], and BEHAVE [211] datasets, human motions are tracked during interaction with objects, but the fingers and palm, are not explicitly captured. In Chapter 2 we capture accurate hand-object interactions with a high-accuracy MoCap system,

but this approach does not scale. Zhang et al. [187] propose a method for real-time hand motion synthesis, given the wrist and object motion. However, this does not generalize to new object shapes and full-body motions. InterCap [185] captures full-body and hand interactions with objects, but hand poses are noisy.

Summary: Previous methods suffer from one or more of generalization ability, computation time, an initial hand pose requirement, or model only single-hand interactions. Our data-driven method, GRIP, addresses these limitations and efficently generates realistic motions for both hands interacting with novel objects.

5.4 Method

Our goal is to add realistic hand poses to a body, based on the relative motion of the body and object during an interaction. To correctly estimate the hand interaction motion, we need to model how and when the object grasp happens. These cues can be found in the object's geometry and the correlated body-object motion trajectories. For example, if the distance between a wrist and the object is decreasing, the hand is approaching the object, but if it becomes constant and the object starts moving, we can infer it is grasped.

To represent such information, we design two virtual "hand sensors"; (1) the *Ambient Sensor* obtains the object's geometric features and its spatial relation to the hands and (2) the *Proximity Sensor* obtains a fine-grained distance field from different hand regions to the object surface.

However, if the arm motion is noisy, these computed features will also be inaccurate. Therefore, as a preliminary step, we use an arm denoising network, ANet, as shown in Fig. 5.2, which takes the noisy arm motion and refines it while enforcing the temporal motion consistency.

Then, we propose a two-stage hand prediction framework to generate hand motion, as illustrated in Fig. 5.2. In the first stage, since we do not have an initial hand pose, we use a mean hand to compute the features of the hand sensors to predict both hand poses. To consider temporal information, we feed our model with the body poses and the hand sensors' features of the current and next frame, in addition to the hand-to-object distance and velocity in the next n frames (typically

10, but this can be varied). In the second stage, based on the predicted hand poses, we recompute the *Proximity Sensor* feature and refine the predictions to enhance interaction accuracy and reduce hand-object penetrations.

We design a two-stage neural network. The first, *CNet*, enforces temporal consistency in hand motions, while the second, *RNet*, only performs a per-frame refinement. Unlike most other motion prediction methods that enforce temporal smoothness in the last step, we handle it in the first stage. This has the advantage of applying smoothness before detailed finger motions are added in the refinement step. Furthermore, since the first step already provides smooth hand poses, the features computed by the *Proximity Sensor* in the second stage are temporally consistent between frames, which keeps the consistent contact regions and smooth motion in the refined hand poses.

Details about each hand sensor and the neural networks are provided in the following sections.

5.4.1 Body and Hand Representations

To model the body and hand motion, we use the SMPL-X [5] model as introduced in Chapter 2. It can represent fine-detailed motion and accurate physical interactions, which are critical for object-interaction motions. Here, we predict only the parameters of the hands: the right-hand pose, $\boldsymbol{\theta}^{\boldsymbol{r}} \in \mathbb{R}^{15\times 6}$ and the left-hand pose, $\boldsymbol{\theta}^{\boldsymbol{l}} \in \mathbb{R}^{15\times 6}$ [152]. In addition, to efficiently represent the hand surface, we follow [188] and sample 99 vertices on each hand; these are denoted v^l and v^r, for the left and right hand, respectively.

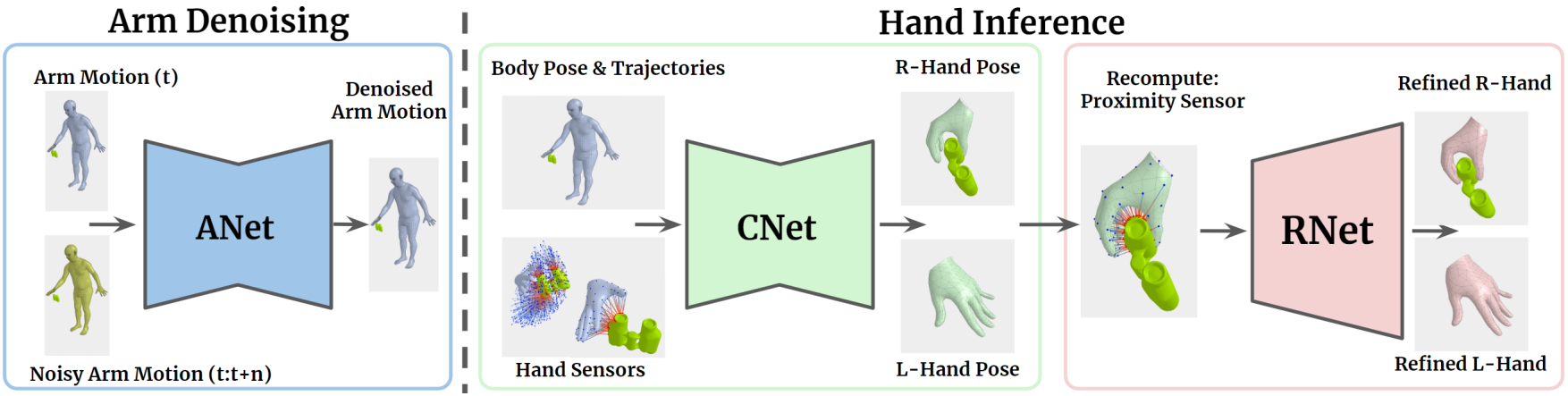


Figure 5.2: Overview of GRIP. We first denoise the arm motion using the ANet network. We then predict hand interaction motion in two stages: (CNet) Given the hand-object spatial features, extracted using our *Hand Sensors*, body pose and trajectories in two consecutive frames, CNet predicts both left- and right-hand poses. (RNet) Based on the predicted hand poses, we recompute the *Proximity Sensor* feature and refine the hand poses with RNet to enhance interaction accuracy and reduce possible penetrations.

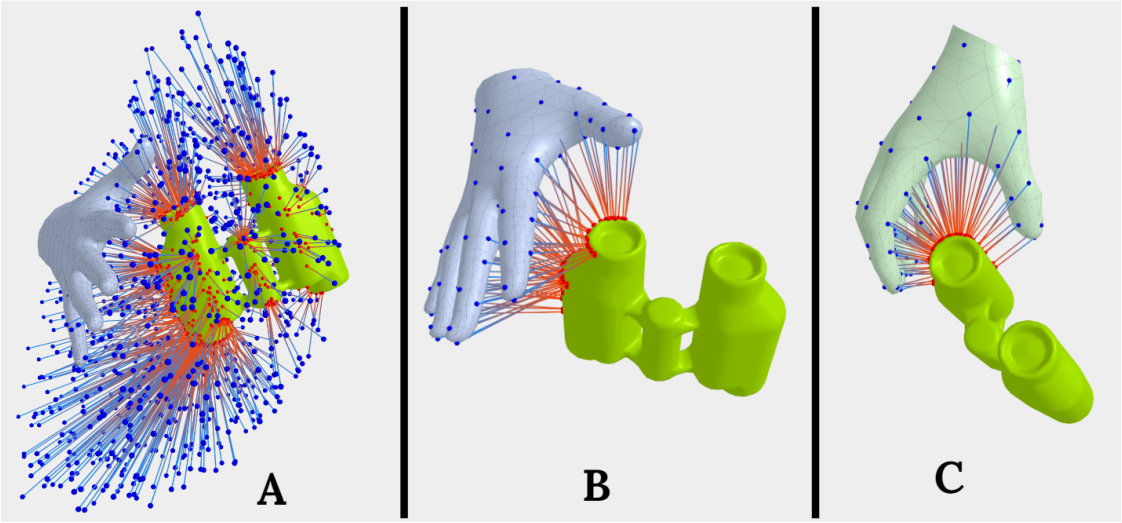


Figure 5.3: Visualization of our Hand Sensors (only right-hand for simplicity). **(A)** *Ambient Sensor* points (blue) and their computed distances to the closest object points (red). This sensor captures the object geometry and distance to the hands. **(B)** *Proximity Sensor* feature computation for CNet's inputs with mean-hand pose initialization. **(C)** Recomputing the Proximity Sensor values for RNet, using the hand poses generated by CNet. Note that the corresponding points on the object change for each finger compared to (B).

5.4.2 Ambient Sensor

To sense the location and shape of the object, we uniformly sample a set of 1024 points in a hemisphere that is rigidly attached to each hand and centered at the base middle-finger joint, as shown in Fig. 5.3-A. For each motion frame, we compute the distance, d, from each of these points to the closest vertex on the object surface. This allows us to capture detailed information about the object shape and the relative distance between the hands and the object. The former informs the hand pose to adapt to certain shapes, while the latter helps predict the state of the hand motion, such as the pre-grasp and pre-release opening, and to keep consistent contact during the interaction.

Unlike commonly used voxel grids [37, 187], which provide a binary and discrete spatial representation, our novel *Ambient Sensor* provides a continuous representation as it uses a distance-based representation. Furthermore, we pass the distances, d, through the interaction-aware attention transformation (Eq. (5.1))

proposed by [188], with $w = 5$, to emphasize points closer to the object surface

$$I_w(d) = \exp(-w \times d), \quad w > 0. \tag{5.1}$$

The ablation studies in Tab. 5.1 and comparison with voxel-based ambient sensors show these unique combination captures rich spatial hand-object relations, improves results, and generalizes to unseen objects and motions.

5.4.3 Proximity Sensor

Although the *Ambient Sensors* capture important interaction information, they do not encode the distance of specific hand regions to the surface of the object; this is essential to know the contact areas. Therefore, we use the sampled hand vertices $\boldsymbol{v}$ and compute their closest distance to the object surface. Since we do not have the hand pose in the beginning, we initialize the hand with the mean pose from SMPL-X [5] and compute the proximity features in the first stage of prediction, as shown in Fig. 5.3-B. In the second stage, we recompute the *Proximity Sensors*' values using the hand poses generated from the first stage, as shown in Fig. 5.3-C.

In contrast to the *Ambient Sensor*, the *Proximity Sensor* provides fine-grained geometric details. This more nuanced information about interaction is essential to generate hand poses with fewer penetrations and better contacts, particularly when the hands are very close to the object's surface. Thus, for the *Proximity Sensor*, we apply the transformation in Eq. (5.1) with a higher weight ($w = 50$) w.r.t. the *Ambient Sensor*, to put emphasis on the vertices closer to the object.

5.4.4 Consistency Network (CNet)

CNet is a novel encoder-decoder neural network that takes the body motion and hand sensor features of two consecutive frames at time t and $t + 1$ to predict the hand poses of both frames. The two frames will be used in our proposed *Latent Temporal Consistency (LTC)* algorithm to enforce temporal and contact consistency for the final prediction. CNet additionally takes the average hand-to-object distance $\boldsymbol{d}$ in the future n frames, from t to $t + n$, where $n = 10$ by default, as input to

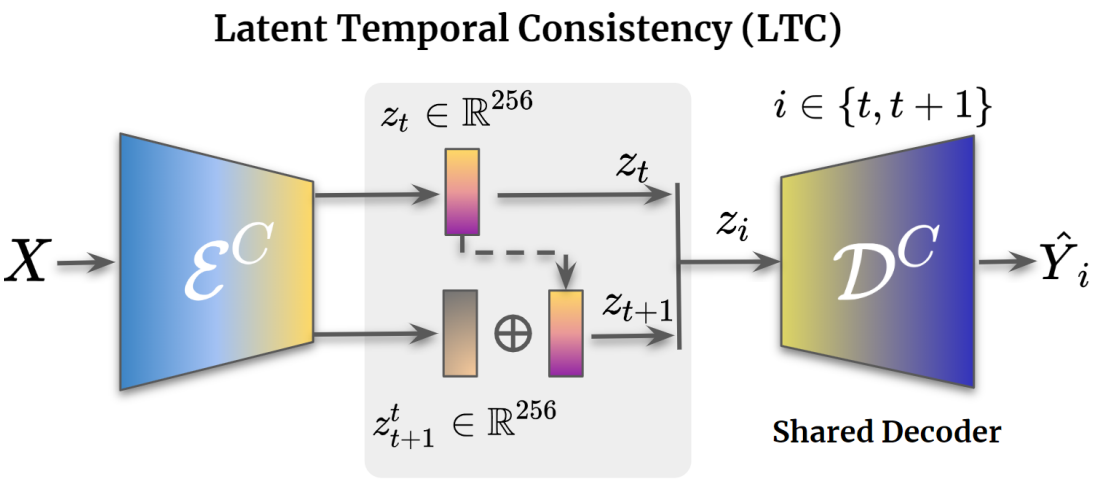


Figure 5.4: CNet Architecture. We propose the LTC algorithm that enforces consistency between two successive frames in the latent space (see Sec. 5.4.4 for more details).

better disambiguate the grasp and release moments. The detailed architecture of CNet is illustrated in Fig. 5.4. The inputs to the network are:

$$X = \left[\boldsymbol{\beta}, \boldsymbol{\theta}_{t:t+1}, \boldsymbol{h}^{\boldsymbol{A}}_{t:t+1}, \boldsymbol{h}^{\boldsymbol{P}}_{t:t+1}, \bar{\boldsymbol{d}}_{t:t+n}, \dot{\bar{\boldsymbol{d}}}_{t:t+n}\right] \tag{5.2}$$

where $t : t+i$ denotes i motion frames in the future including the current frame, $\boldsymbol{\theta}_{t:t+1}$ are the SMPL-X joint angles without considering the global root joint, $\boldsymbol{h}^{\boldsymbol{A}}_{t:t+1}$ and $\boldsymbol{h}^{\boldsymbol{P}}_{t:t+1}$ are the hand *Ambient Sensor* and *Proximity Sensor* values for both left and right hands, and $\bar{\boldsymbol{d}}_{t:t+n} \in \mathbb{R}^{2\times 99}$ and $\dot{\bar{\boldsymbol{d}}}_{t:t+n} \in \mathbb{R}^{2\times 99}$ are the average of hand-to-object distance and its rate of change for sampled hand vertices in the n future frames.

5.4.5 Latent Temporal Consistency (LTC)

In addition to physically plausible hand-object contact, an important factor in the realism of interaction motions is consistent dynamics and contact areas between consecutive frames. To enforce these, we smooth the motion in the latent space of hand motions rather than in the output space, as we noticed the latter adds high-frequency noise to the contact areas throughout the motion. As shown in Fig. 5.4, the encoder, $\mathcal{E}^{\mathrm{C}}$, maps the input X to two latent codes, $z_t, z^t_{t+1} \in \mathbb{R}^{256}$, where z_t denotes the global latent code for a hand pose in the current frame and z^t_{t+1} is the relative latent code for the next frame with respect to the current frame. We

compute the global latent code for the next frame by adding the two latent codes as $z_{t+1} = z_t + z^t_{t+1}$; see Fig. 5.4. We then pass each global latent code individually to a shared decoder, $\mathcal{D}^{\mathrm{C}}$, to get the outputs $\hat{Y}$. The shared decoder helps regulate inconsistency between the two global latent codes, as it is represented and penalized in the final hand poses. The output of CNet is:

$$\hat{Y} = \left[\hat{\boldsymbol{\theta}}^{\boldsymbol{r}}_{t:t+1}, \hat{\boldsymbol{\theta}}^{\boldsymbol{l}}_{t:t+1}, \hat{\boldsymbol{h}}^{\boldsymbol{P}}_{t:t+1},\right] \tag{5.3}$$

where $\hat{\boldsymbol{\theta}}^{\boldsymbol{r}}_{t:t+1}$, $\hat{\boldsymbol{\theta}}^{\boldsymbol{l}}_{t:t+1} \in \mathbb{R}^{15\times 6}$ are right-/left-hand poses in the current and next frame, and $\hat{\boldsymbol{h}}^{\boldsymbol{P}}_{t:t+1}$ are the inferred *Proximity Sensor* values; the latter ones have been shown to increase realism and lower errors [188].

Generating hand poses in the current and next frame allows for defining consistency and smoothness losses between them. Evaluations in Tab. 5.1 show that the motions generated with our LTC algorithm achieve a lower error and better consistency compared to baselines with no enforced consistency or with consistency in the output space.

We use fully-connected dense residual blocks with skip connections for both encoder and decoder, and train CNet end-to-end. The training loss is defined as

$$\mathcal{L} = \lambda_{\boldsymbol{v}}\mathcal{L}_{\boldsymbol{v}} + \lambda_{\boldsymbol{h}^{\boldsymbol{P}}}\mathcal{L}_{\boldsymbol{h}^{\boldsymbol{P}}} + \lambda_{\theta}\mathcal{L}_{\theta}, \tag{5.4}$$

where $\mathcal{L}_{\boldsymbol{v}} = \|\boldsymbol{v} - \hat{\boldsymbol{v}}\|_1$ is a loss on the hand vertices $\boldsymbol{v}$, $\mathcal{L}_{\theta} = \|\hat{\boldsymbol{\theta}}^{\boldsymbol{l}} - \boldsymbol{\theta}^{\boldsymbol{l}}\|_2 + \|\hat{\boldsymbol{\theta}}^{\boldsymbol{r}} - \boldsymbol{\theta}^{\boldsymbol{r}}\|_2$ is on the joint rotations of both hands and $\mathcal{L}_{\boldsymbol{h}^{\boldsymbol{P}}} = \|\hat{\boldsymbol{h}}^{\boldsymbol{P}} - \boldsymbol{h}^{\boldsymbol{P}}\|_1$ is on the hand-to-object distances, both directly estimated from the network and derived from the estimated hand poses.

5.4.6 Arm Denoising Network (ANet)

For the hand sensors in CNet to capture rich information between the hand and the object, the motion of these two should be very accurate and without noise. Therefore, as shown in Fig. 5.2 (left), we train ANet to first refine the arm motion before passing to CNet. It takes as input both arms' pose in the current frame, $\boldsymbol{\theta}^{\boldsymbol{la}}$ and $\boldsymbol{\theta}^{\boldsymbol{ra}}$, and the noisy poses of the future frame, $\boldsymbol{\theta}^{\boldsymbol{la}}_{\boldsymbol{p}}$ and $\boldsymbol{\theta}^{\boldsymbol{ra}}_{\boldsymbol{p}}$, and gives the denoised

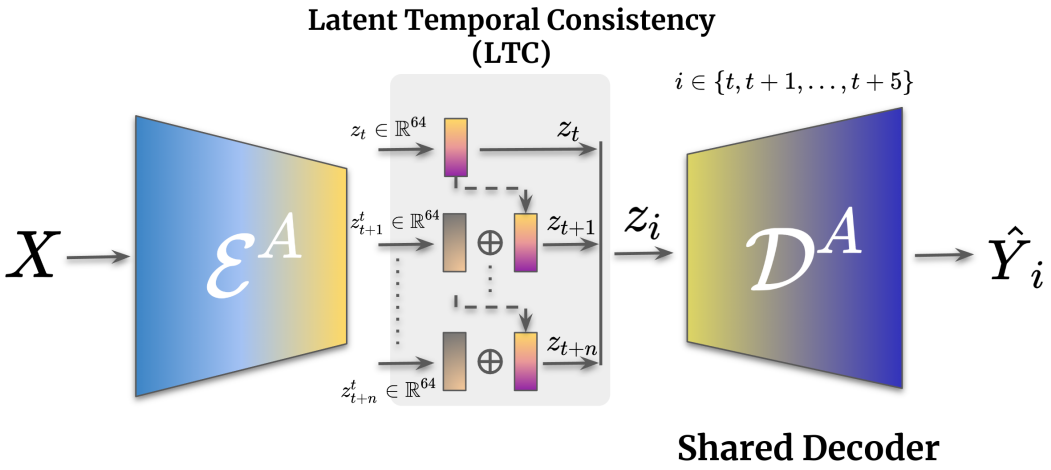


Figure 5.5: Architecture overview of ANet. Similar to CNet, we use the LTC algorithm to ensure motion consistency of the denoised arm motions. For this, the encoder maps the input to a global latent code in the current frame and relative latent codes in the future frames. Then a shared decoder is used to generate the denoised motions.

arm poses. We use a similar architecture to CNet, and enforce the consistency between the denoised poses in the latent space of the network using LTC.

For an architectural overview of ANet see Fig. 5.5. As input, A-Net takes the arm motion and hand sensor features of the current Ground Truth frame along with five noisy future frames. As output it gives the denoised arm poses for the five future frames, following [188]. To ensure motion consistency between the successive frames of the denoised motions, we use the LTC algorithm similar to CNet, as explained in Sec. 5.4.5. For this, the encoder, E^A, maps the input to five latent representations for each arm pose, as shown in Fig. 5.5. Then we apply the latent temporal consistency algorithm by adding the residual latent codes, z_i^t, to the global latent code, z_t. Finally, we use a shared decoder, D^A, to decode the denoised motions. Both encoder and decoder have 4 fully-connected residual layers with skip connections in between.

5.4.7 Refinement Network (RNet)

The motions generated by CNet are in the right ballpark but can be refined further to improve realism and remove possible penetrations. To this end, we train a refinement network, RNet. We use the generated hand poses from CNet to

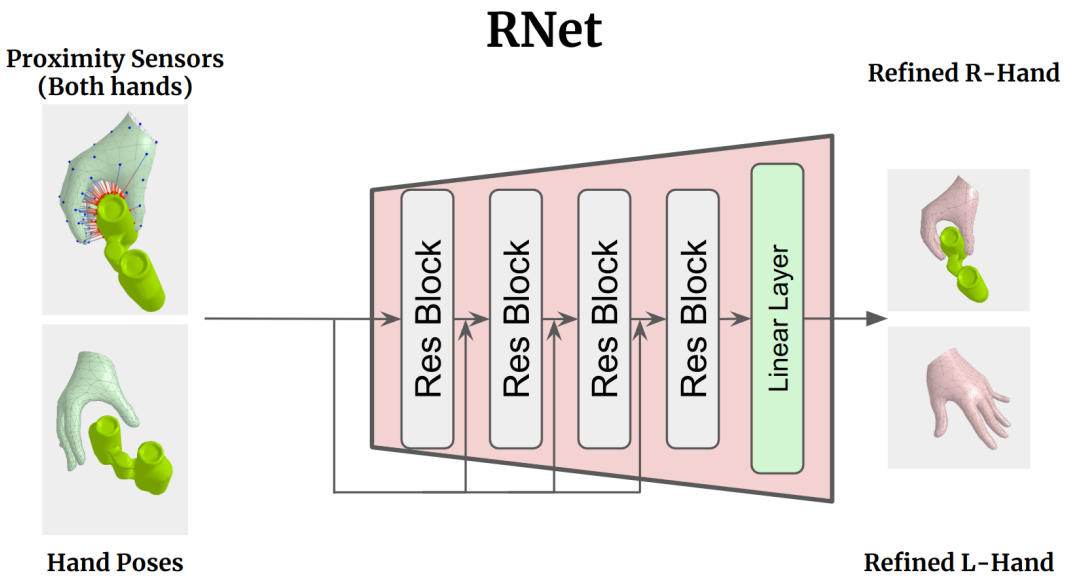


Figure 5.6: Architecture overview of RNet. As input, it takes hand poses and proximity sensor values, and generates the refined hand poses. The network consists of 4 residual blocks with skip connections and an output linear layer.

recompute *Proximity Sensor* features, $\boldsymbol{h}_{\theta}^{\boldsymbol{P}}$, similar to CNet inputs (see Fig. 5.3-C). Then RNet takes $\boldsymbol{h}_{\theta}^{\boldsymbol{P}}$ and the hand poses, $\hat{\boldsymbol{\theta}}^{\boldsymbol{l}}$ and $\hat{\boldsymbol{\theta}}^{\boldsymbol{r}}$, and outputs the refined hand poses. To keep the motion dynamics, generated from CNet, we train RNet to refine hand poses only in the interaction frames and not to change the pose when hands are far away from the object surface. In addition to the CNet output, we train RNet on perturbed training data to simulate noisy inputs. Training losses are similar to those used for CNet in Eq. (5.4).

For the architecture overview of RNet please see Fig. 5.6. RNet takes, as input, hand poses and proximity sensor values of a motion frame and, as output, generates the refined hand poses for both the left and right hand. The network consists of 4 residual blocks with skip connections and an output linear layer.

5.4.8 Data

The GRAB dataset Chapter 2 is used to train our GRIP model.

CNet data: CNet generates hand interaction motion based on the body and object motion in a sequence. We use all the training and test sequences from GRAB for training and testing CNet, respectively. In addition to hand-object grasp

frames, we consider the per-grasp and post-grasp motion frames of each sequence to generalize to the hand poses throuout the whole sequence. In total, we use 1335 motion sequences, performed on 51 3D objects. To split the dataset, we use the motions performed on “mug”, “apple”, “camera”, “binoculars”, and “toothpaste” as the test set,“fryingpan”, “toothbrush”, “elephant”, and “hand” as the validation set, and the rest as the training set. In total, we have $329K$, $52K$, and $24K$ motion frames for the training, testing, and validation set, respectively.

RNet data: RNet refines the motions generated from CNet, therefore, we use the output of CNet as the main data source for RNet. In addition, to model more severe penetration and interaction artifacts, we prepare a synthetic dataset by perturbing the ground-truth data in GRAB. For this, we add Gaussian noise with a standard deviation of 0.3 to the axis-angle rotation representation of the hand poses.
ANet data: ANet is trained to refine noisy arms motion. To prepare the training data, we add Gaussian noise to the shoulder and elbow joints of the ground-truth motion data. The noise is added to the axis-angle rotation of the joints and has 0.01 and 0.03 standard deviations for the shoulder and elbow joints, respectively.

5.5 Experiments

5.5.1 Evaluation Metrics

We use the standard “Mean Per-Joint Position Error” (MPJPE) and “Mean Per-Vertex Position Error” (MPVPE), which represent the Euclidean distance between the ground-truth and estimated hand joints and vertices, respectively.
Intersection Volume (IV): This measures the intersection volume between the hand and the object to assess the realism, i.e., the physical plausibility, of the generated grasps.
Contact Consistency (CC): This evaluates the consistency of contacts for the grasping frames of generated grasp motions, i.e., the finger sliding on the object surface. We use ground-truth motions to select grasp frames, and, for generated motions, compute the deviation distance from the contact areas on the object.

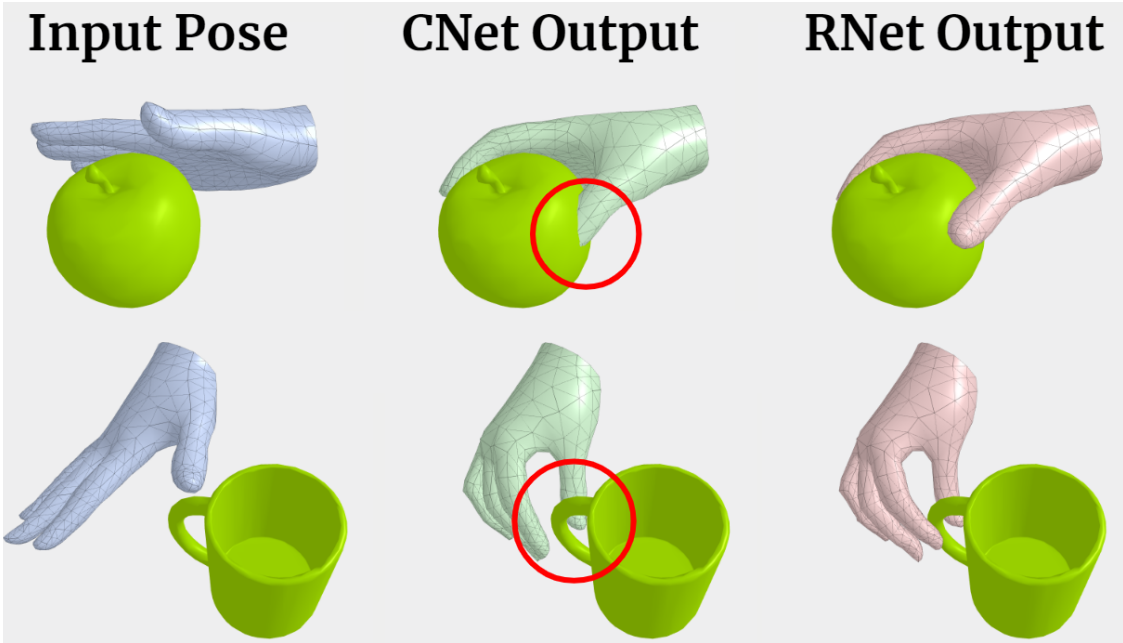


Figure 5.7: Comparing CNet and RNet generated grasps. Results show that RNet effectively refines the penetration and "non-contact" artifacts (red circles) of the CNet results.

For this, we proceed as follows: Let F denote the set of grasp frames selected from the ground truth motions. For each grasp frame $f \in F$, Let $C_g(f)$ denote the contact area on the object for the generated grasp motion and $C_t(f)$ denote the contact area on the object from the ground truth. The deviation distance for frame f is computed as:

$$D(f) = \text{distance}(C_g(f), C_t(f))$$

Then the Contact Consistency is computed as the average deviation distance over all grasp frames in F:

$$CC = \frac{1}{|F|} \sum_{f \in F} D(f)$$

Where:

- $|F|$ represents the number of grasp frames.
- $D(f)$ represents the deviation distance for frame f.

5.5.2 Qualitative Evaluation

Results show that CNet generates reasonable and smooth hand grasps, but sometimes with artifacts like hand-object interpenetration. After applying the refinement network, RNet, the results look more realistic and physically plausible. In Fig. 5.7 we show examples of generated grasps using CNet and after applying the RNet refinement.

Figures 5.1 and 5.10 show several representative hand motions generated with GRIP, including pre-/post-grasp hand opening, single-hand grasps, and bi-manual grasps for different unseen object shapes. Overall, the generated hand motions are reasonable, smooth, and consistent. For more results, please see Appendix C.

In Fig. 5.8 we show more qualitative results generated on unseen objects, using GRIP. The top row shows input body and object motion, and the bottom row shows generated hand poses. We show close-ups of the generated hand poses, in single and bimanual scenarios, to show the accuracy of the generated grasps. In Fig. 5.9 we provide results for successive frames of a motion sequence to show the consistency of the generated hand poses over time. Additionally, the results show that our method is able to refine the noisy arm poses from the InterCap dataset.

In Fig. 5.11, we show representative scores for the ManipNet grasps from our user study. These results confirm several limitations of ManipNet which GRIP addresses, making it easy to apply in real-world scenarios.

Performance on Other Datasets: GRIP is trained on the GRAB dataset, which only has small hand-held objects. High-quality data of hand-object interaction with large objects is rare. Despite training on small objects, our virtual hand sensors help generalize to larger objects, as they only sense the interaction areas locally and not the whole object. To highlight GRIP's generalization capability, we show generated interaction poses for *unseen* large objects from the InterCap [185] and MoGaze [189] datasets in Fig. 5.12 and Fig. 5.1-right, and compare them with the original hand poses.

Cross-Object Grasp Transfer: We show that GRIP can be used to transfer grasping motions from one object to another one. To test whether our method generalizes well to different object shapes and motions, we use GRIP to transfer

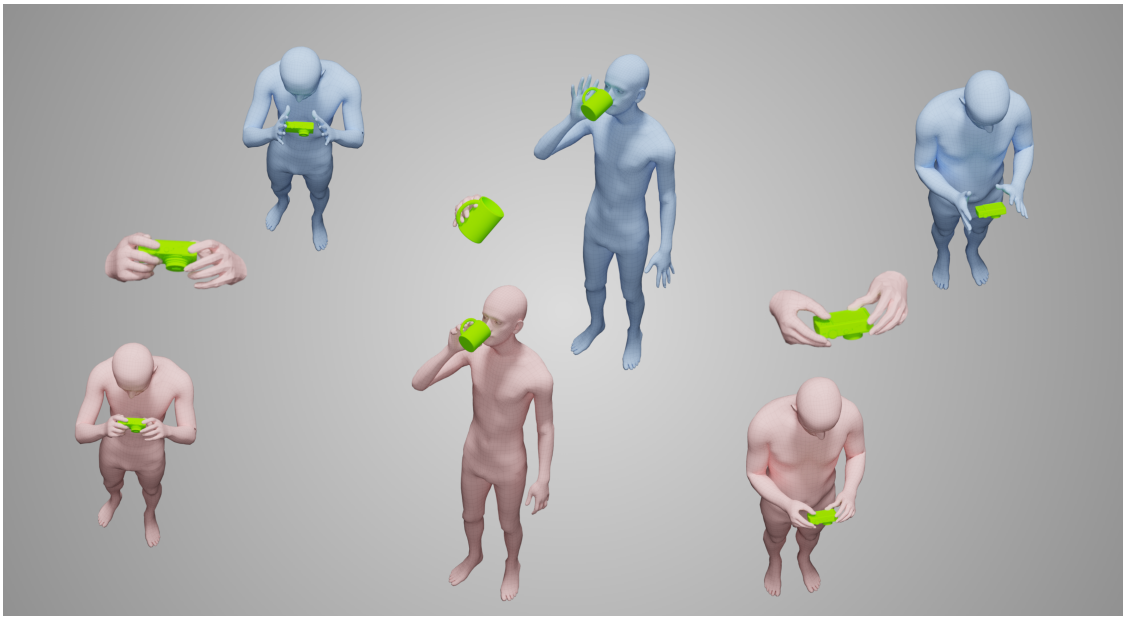


Figure 5.8: Generated results with GRIP for unseen objects. (Top row) input body and object, (bottom row) generated hand poses. We show close-ups of the generated hand poses in single and bimanual scenarios, to show the accuracy of the generated grasps.

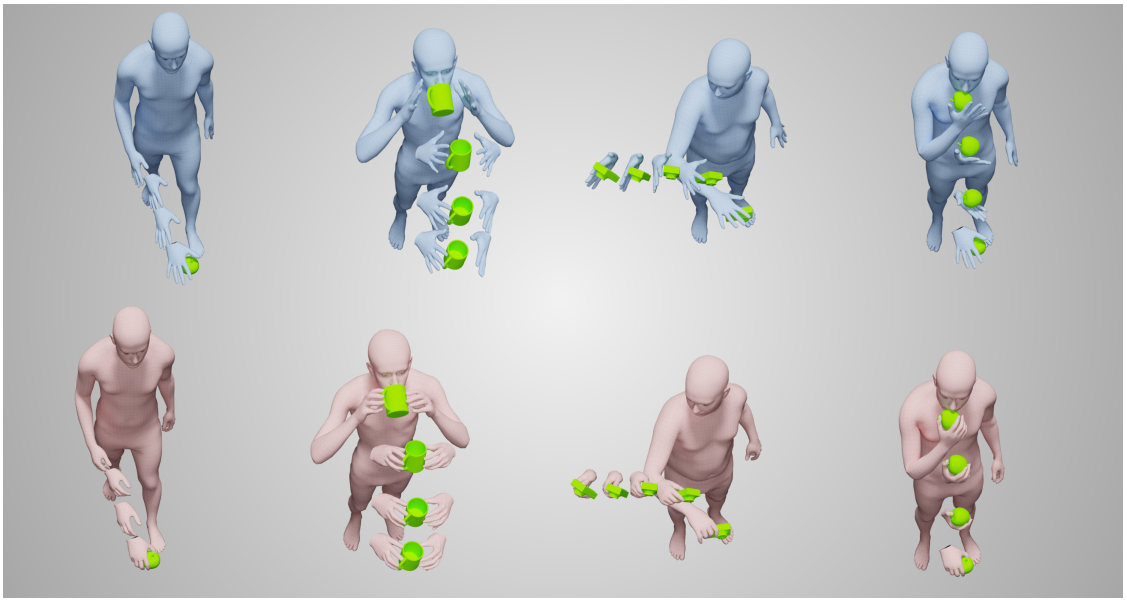


Figure 5.9: Generated hand motions using GRIP. (Top row) input body and object motion. (Bottom row) generated hand poses. We provide results for successive frames of the same motion to show the consistency of the generated motions over time.

the input interaction motion from a source object to a target object. Given a sequence of body and object motion without hand poses, we replace the source

Figure 5.10: GRIP results. We show various generated grasps, in single and bimanual scenarios, for different objects shapes. The input (flat, non-articulated) hands are shown with blue meshes, and GRIP's generated hands (articulated) with pink meshes.

object with a target object that is roughly of the same size. We then compute the hand sensor features for the new object geometry and use GRIP to generate hand interaction poses for the new object.

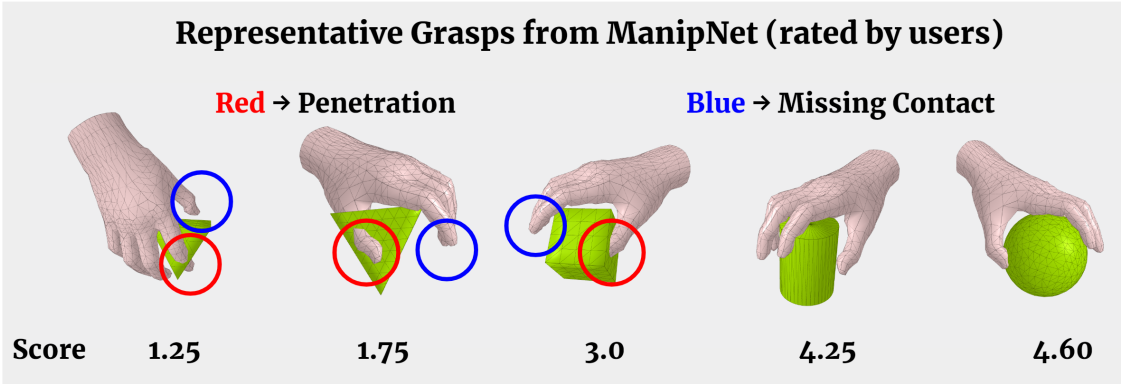


Figure 5.11: representative scores for ManipNet [187] grasps from our user study.

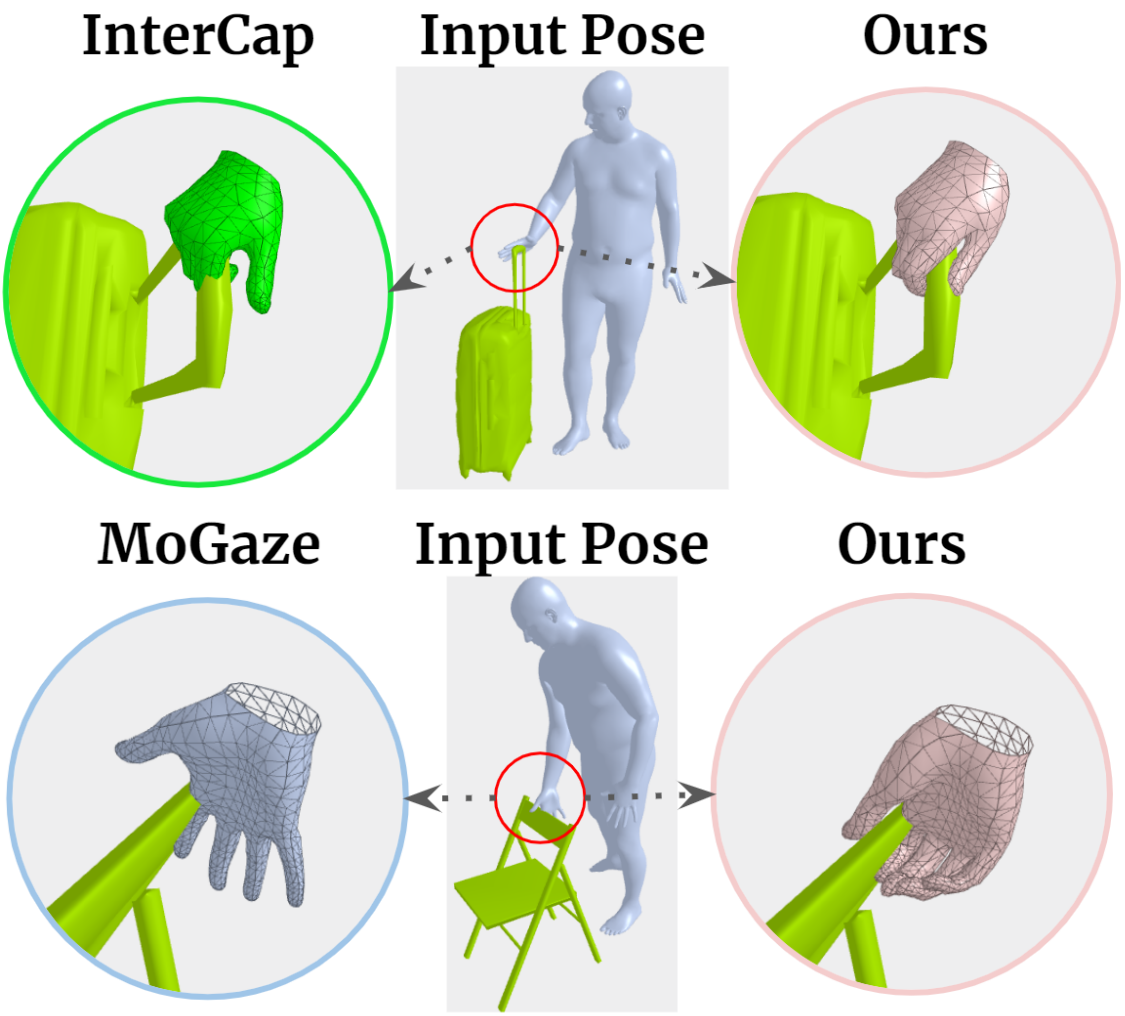


Figure 5.12: Our generated grasps (pink circles) for large objects from InterCap [185] and MoGaze [189] , and comparison with the original grasps from these datasets.

Qualitative results show that our method is able to generate realistic hand motions for the target object and generalizes well to the new object's shape and motion. In Fig. 5.13 we show two examples of the grasp transfer application. The top row shows that the hands adapt well to the target object geometry, "elephant", and the bottom row shows a change in the grasp type (e.g., thumb contact area) due to the smaller size of the target object, "sphere". This is useful for synthetic data generation because a single motion capture sequence can be repurposed to generate many different synthetic human-object interactions. This is also useful for FX where actors are captured handling a "dummy" object that is replaced by a 3D graphics object; this is a common scenario in film production.

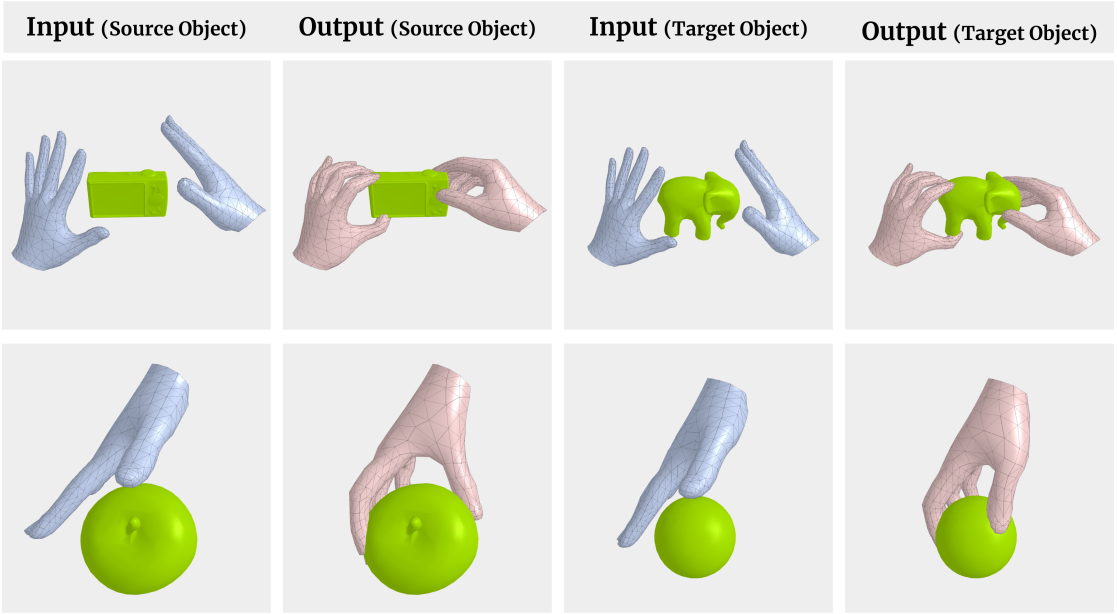


Figure 5.13: Grasp transfer from a source object to a target one. Given a sequence of body and object motion without hand poses, we replace the source object with a target one and use GRIP to generate hand interaction poses for the new object. The top row shows grasp transfer from "camera" to an "elephant" geometry, and the bottom row shows grasp transfer from an "apple" to a small "sphere". Notice how the hands adapt to the new object shape (top row) and the change in the grasp type (bottom row).

Method ↓	**MPVPE** (mm) ↓		**MPJPE** (mm) ↓		**CC** (mm) ↓	
	R-Hand	L-Hand	R-Hand	L-Hand	R-Hand	L-Hand
	Hand Sensors Ablation					
GRIP (w/o Ambient)	9.56	6.72	7.08	4.99	15.03	9.48
GRIP (w/o Proximity)	9.62	6.82	7.11	5.09	15.64	9.10
	Latent Temporal Consistency (LTC) Evaluation					
GRIP (w/o Consist.)	8.17	6.18	5.99	**4.53**	13.01	7.66
GRIP (output Consist.)	9.31	7.11	6.81	5.31	13.21	8.18
GRIP (w/o RNet)	8.19	6.58	6.10	4.95	11.44	7.03
GRIP (fullmodel)	**7.88**	**6.17**	**5.85**	4.62	**10.56**	**6.25**

Table 5.1: (Top) We show the effect of our "Hand Sensors" by comparing variants of GRIP without our sensors' features; GRIP results in lower errors. **(Bottom)** The effect of the LTC algorithm is explored by comparing GRIP against a network without LTC (*w/o Consist.*) and one with consistency on the output poses (*output Consist.*). The GRIP-generated motions have lower errors.

5.5.3 Ablation Study

Latent Temporal Consistency (LTC): To evaluate the importance of our proposed temporal consistency algorithm for interaction motions, we compare our

	# of Future Frames	MPVPE (mm) ↓ R-Hand	MPVPE (mm) ↓ L-Hand
1.	0	9.21	8.18
2.	3	8.94	7.78
3.	5	8.34	7.29
4.	10	**7.88**	**6.17**

Table 5.2: Evaluating the trade-off for the real-time performance and accuracy of GRIP by comparing different numbers of future frames. Results demonstrate that by using up to 10 future frames, the metric errors decrease.

network with two baselines, namely: (1) a network without enforced consistency (*w/o Consist.*) and (2) a network with consistency applied directly on the generated hand poses (*output Consist.*). As seen in Tab. 5.1-bottom our LTC method that smooths the latent space representation not only reduces the CC error, but also results in lower errors in MPVPE and MPJPE.

Hand Sensors: To evaluate the effect of our *Ambient Sensor* and *Proximity Sensor*, we train different baselines of GRIP by removing these features, (*w/o Ambient*) and (*w/o Proximity*). We compare MPVPE, MPJPE, and CC between the generated hand motions and the ground truth. Results in Tab. 5.1-top show that our distance-based hand sensors provide rich interaction information to the network that leads to lower errors and consistent motions.

RNet: In Tab. 5.1 we evaluate our refinement network, RNet, by comparing the results of GRIP with RNet (*fullmodel*) and without it (*w/o RNet*). The table verifies that the refinement step helps reduce the hand MPJPE and MPVPE errors and enhance motion consistency.

Number of Future Frames: In Tab. 5.2 we show the effect of using a different number of future motion frames on the accuracy of the hand poses by comparing different variants of GRIP. The table verifies that using more future frames (up to 10 frames) lets the network generate more accurate poses. This is a trade-off between a real-time performance (row 1) and a higher accuracy with some latency (rows 2-4). Empirically, we observe that performance saturates for more than 10 frames, in accordance with [188].

Metric	GRIP (w/o RNet)	GRIP	Ground truth [41]
Hand-Object Grasp ↑	4.09 ± 0.89	4.11 ± 0.85	4.12 ± 0.90
Hand Motion Smoothness ↑	3.88 ± 1.06	3.91 ± 1.04	3.98 ± 1.03
Contact Consistency ↑	4.02 ± 1.01	4.09 ± 0.95	4.13 ± 0.95
In-Hand Manipulation ↑	3.96 ± 1.01	3.97 ± 0.99	4.01 ± 1.00
Average ↑	3.99 ± 1.00	4.02 ± 0.96	4.06 ± 0.97

Table 5.3: Perceptual evaluation of GRIP results, without and with RNet, compared with the ManipNet [187] results and ground truth [41]. The participants rate the realism of the generated grasps from 1 (unrealistic) to 5 (very realistic). The table reports the mean ± std, computed for all valid study participants. Results show that GRIP generated grasps are more realistic than ManipNet and that RNet improves the grasps of CNet.

5.5.4 Perceptual Study (Comparison to ManipNet)

We evaluate the hand motions generated from CNet and RNet with a perceptual study on AMT and compare them with ManipNet results and the GT motions. For GRAB's test-set motion sequences, we use GRIP to generate the interacting hand poses. We then create videos of the generated motions from CNet, the refined motions from RNet, and the corresponding ground truth. To compare with ManipNet, we extracted their moving meshes from their demo and rendered them in the same format as GRIP results.

The participants rate the realism of the hand motions based on 4 criteria: (1) hand-object grasp, (2) hand motion smoothness, (3) contact consistency, and (4) in-hand manipulations. Each motion is evaluated by at least 10 different participants. The ratings are on a 5-level Likert scale, where 1 means unrealistic and 5 means very realistic. We use catch trials similar to [41, 188] to identify invalid ratings and remove them; Tab. 5.3 shows the evaluation results.

The study shows that the GRIP-generated hand motions are very realistic and close to the ground-truth ones. In addition, the scores are slightly higher when motions are refined by RNet, especially for Contact Consistency (CC), which shows the effectiveness of our LTC algorithm. Furthermore, we see a lower rating for ManipNet results compared to our results. Additionally, in Tab. 5.1 we show the computed penetration errors for ManipNet, which is 13% higher than ours. While the test data is different (simpler for ManipNet), these results confirm several limitations of ManipNet such as single-hand inference, poor generalization to new objects,

Metric ↓	Model ↓	GRAB-T (0.01)	GRAB-T (0.02)	GRAB-R (0.3)	GRAB-R (0.5)
MPVPE (mm)	TOCH	16.0 → **11.8**	31.9 → **13.9**	**6.30** → 11.5	**10.3** → 11.0
	GRIP	17.4 → **10.3**	34.2 → **13.1**	6.21 → **4.62**	10.5 → **6.72**
MPJPE (mm)	TOCH	16.0 → **9.93**	31.9 → **12.3**	**4.58** → 9.58	**7.53** → 9.12
	GRIP	16.9 → **9.70**	33.8 → **12.8**	4.26 → **3.21**	7.64 → **4.18**

Table 5.4: Comparison of GRIP (only RNet) performance with TOCH [186] on the perturbed test-sets from GRAB. Following TOCH, we perturb the hand pose (-R) and translation (-T) by adding Gaussian noise to them. The numbers in parentheses (top) show the noise magnitude. The table reports the metrics before and after using each method.

and no full-body setting. GRIP addresses these issues, making it easy to apply in real-world scenarios. For representative grasps and failures please see Appendix C.

5.5.5 Comparison to TOCH

To evaluate the performance of ANet and RNet, we compare them to TOCH [186] on refining perturbed test-sets from GRAB. To do this, similar to [186], we perturb the motions by adding Gaussian noise, with different magnitudes, to the pose (GRAB-R) and translation (GRAB-T) of both hands. To keep the original motion dynamics, generated from CNet, RNet is trained to only refine hand-pose (i.e., rotation perturbations), therefore we refine perturbed translation using ANet and perturbed rotations using RNet. We provide the full-comparison results in Tab. 5.4. Results show that the combination of ANet and RNet performs better in refining noisy hand interactions.

5.5.6 Baselines

To evaluate GRIP's performance, in Tab. 5.5 we compare the penetration volume (cm^3) and contact ratio [81] of two baselines (rows 1, 2) with our models (rows 3, 4), namely: (1) "GrabNet" [41], which generates MANO grasps, (2) a trained "GrabNet-SMPL-X" variant, which generates full-body SMPL-X grasps, (3) GRIP (w/o RNet), and (4) GRIP (w/ RNet). Results show that our full model (row 4) performs better than the baselines in generating grasps. Please note that our

Grasp Synthesis	Penetration (cm^3) ↓	Contact Ratio ↑
1. GrabNet [41]	2.65	**1.00**
2. GrabNet-SMPL-X	7.33	0.87
3. GRIP (w/o-RNet)	3.18	0.96
4. GRIP (w/ -RNet)	**2.38**	**1.00**
GRAB (GT)	1.95	1.00

Table 5.5: Penetration and contact-ratio metrics for two GrabNet baselines and GRIP models with-/out RNet. We report penetration volume and contact ratio of different baselines on the sequences of test set that have hand-object contact.

model generates the *motion* of *both hands* during object interaction, while the baselines only generate *static* grasps of *one* hand. Also, note that small penetrations (1.95 cm^3) are inevitable even in the ground-truth, since SMPL-X does not model soft-tissue deformation [42].

5.6 Runtime

Due to its pure learning-based pipeline, GRIP is able to generate hand poses rapidly. We find that a full forward pass of our method on a single V100-16GB GPU, including the CNet inference, recomputing proximity sensor values, and RNet forward pass, takes 0.022 seconds, which is equivalent to 45 fps. Therefore, GRIP can be used to synthesize hands for avatars in interactive applications like video games and mixed reality settings, which are mostly running at 30 fps. Please notice that our network still relies on mean hand-to-object distance in the future 10 frames, which causes a fixed 10-frame latency (1/3 of a second) in real-time applications. This is the trade-off to have more accurate poses with latency instead of real-time performance with lower accuracy, as shown in Tab. 5.2

5.7 Conclusion

We propose GRIP, a data-driven method that directly generates realistic interaction motions for both hands given the animated body and target object. Our method's novelties include (1) a two-stage hand prediction approach using two networks for

coarse and fine grasping, (2) the combination of two novel hand sensors, and (3) a latent-space temporal consistency modeling. As a result, compared with previous methods, GRIP is able to predict hand poses from scratch, generalize to novel object shapes, adapt to bimanual interactions, and generate subtle hand poses with temporal consistency. Those benefits will allow GRIP to be used in scenarios like capturing new datasets of human-object interaction *without* the difficulty of tracking the hands. This can also be used to add hands to previous datasets [20, 189] "for free", and to synthesize hands for avatars in video games and mixed reality settings.

Limitations and Future Work: Although the inference time for GRIP is very fast, it relies on the mean hand-to-object distance in the future 10 frames to guide the prediction of grasps. This causes a fixed 10-frame latency in interactive applications. It may be possible to learn to anticipate movement and reduce this delay. Extending the method to human-scene interaction would be interesting.

"The hand is the visible part of the brain."

— Immanuel Kant

6

Conclusion

Contents

6.1 Contributions

The completion of this doctoral thesis has yielded important advancements in the field of human-object interaction modeling. Through the development of novel datasets, methods, and algorithms, we have expanded the boundaries of understanding and simulating human-object interactions, capturing the depth and complexity of these areas. This work has broadened the scope of 3D human-object interaction modeling to incorporate full-body avatars, enabling the generation of dynamic and intricate interaction sequences. Our contributions can be categorized into four distinct areas, each building upon the other to create a robust and comprehensive exploration of the field.

GRAB: The cornerstone of this research is the creation of the GRAB dataset, a novel dataset that captures intricate and complex 3D human-object interactions. Prior work primarily focused on prehensile grasps, where a single human hand lifts or interacts with an object. Our research, however, recognized the necessity to incorporate the entire body in such motions to accurately capture the full scope of human-object interactions.

The GRAB dataset filled a crucial gap in the existing data resources by capturing accurate 3D body and object motion. In creating this resource, we overcome the challenges of previous datasets using innovative techniques and state-of-the-art technologies. We used a high-resolution motion capture system to accurately track detailed whole-body motions and object positions, employing meticulously designed marker layouts on the body and objects. Additionally, We employed an adapted and extended version of MoSH++, allowing us to reconstruct facial motion in addition to capturing the 3D shape and motion of the body and hands from the tracked markers as well as the object motion.

As a result, GRAB is the first dataset of its kind, which contains 3D motion of the body and objects, precise hand-object interaction, head motion, and detailed body-object contact information. This unique and comprehensive resource extends beyond existing datasets, paving the way for research into how humans interact with objects in various tasks using their entire body, modeling and understanding how humans grasp and manipulate objects, and how the interaction varies with the task.

GrabNet: The development of GrabNet was the next substantial stride in this research. Utilizing the GRAB dataset, GrabNet was trained as a model capable of generating accurate static hand grasps given an unseen 3D object. This work achieves state-of-the-art performance in modeling hand-object grasps, moving beyond previous models. This application reinforces the usefulness of GRAB in the field and lays the foundation for our subsequent contributions, building upon the initial step of static grasps to generate dynamic, whole-body interactions.

GOAL: Our next contribution lies in the development of GOAL, the first model designed to generate avatar motion to walk and grasp 3D objects. The innovation of GOAL emerged from the recognition of the limitations in GrabNet's ability to generate dynamic, whole-body grasping movements. Although GrabNet advanced static hand-object grasps, GOAL elevated this by simulating the realistic body and head pose involved in walking toward and grasping an object. GOAL takes a 3D object, its position, and a starting 3D body pose and shape and generates avatars that walk and grasp the objects with realistic body pose, head orientation, hand grasp, and foot-ground contact.

Unlike prior work, we exploit a complementary set of grasp representations, including vertex offsets between the body and object and a head direction, in

the network output, resulting in a more accurate body and head pose during the grasp. Additionally, A key feature of GOAL is the introduction of an interaction-aware feature transformation that encodes body-to-object distance into a richer representation. This helps the networks to better localize the object with respect to the body, which leads to generating smooth motions without noticeable foot sliding. As such, GOAL stands as the first model for simulating avatars that walk and grasp objects, incorporating realistic body pose, head orientation, hand grasp, and foot-ground contact.

GRIP: The fourth major contribution of our research is the development of GRIP, a learning-based interaction model designed to synthesize realistic hand motion before, during, and after object manipulation. While GOAL generates the avatar motion up to the accurate grasping of the objects, we recognized that the realism of avatar interactions largely depends on the realistic and physically plausible finger motion during object manipulation. While this involves satisfying numerous semantic and physical constraints and might seem trivial for humans, it presents a significant challenge in the realm of digital human modeling.

GRIP bridges this gap by taking the 3D motion of the object and the body without finger articulation and synthesizing realistic motion for both hands. This is achieved by leveraging the rich temporal interaction cues drawn from the spatio-temporal information between the body and the object, which helps the model to generalize well to new objects.

Furthermore, to generate a consistent motion between frames, we introduce a motion temporal consistency applied in the latent space, which together with a shared decoder regulates motion inconsistency and leads to generating consistent interaction motions. In summary, GRIP advances digital human modeling by "upgrading" sequences of body and object motion without hand articulation to include detailed hand-object interaction, which ultimately enhances the realism and applicability of virtual avatars.

6.2 Future Work

We believe that despite the significant advancements made in this research, several intriguing research avenues remain unexplored and some new ones open. Our

contributions provide rich grounds for future investigation.

Motion Synthesis via Action Labels: The present study has centered around synthesizing realistic human-object interactions using 3D representations. However, an exciting avenue of future research would be the automatic generation of both body and object motion, given an action label. The challenge lies in abstracting a multitude of potential human-object interactions into concise and interpretable action labels, ensuring generated motions align with the intended actions. The interaction motion could vary for the same action label but with a different initial position of the body and object, object shape, object category, and many other factors. The models presented in this thesis, especially GOAL and GRIP, could serve as a foundational starting point for this work, given their ability to generate holistic human-object interactions.

Video-based Interaction Reconstruction: Video data offer a rich source of information on human-object interactions. However, the challenge remains in converting 2D video footage into precise 3D models that can accurately represent these interactions. Furthermore, videos, being unstructured data, may contain noise or discrepancies that need to be effectively managed to produce reliable outputs. Current methods mainly focus on reconstructing human motions from images and mostly ignore the objects and interactions. Future research could explore methods of reconstructing body and object motion from videos, extracting valuable interaction data from the abundant, albeit unstructured, video material available online. The GRAB dataset, with an emphasis on accurate tracking of human-object interaction, could be instrumental in training models to perform such tasks and extract accurate information from video sources.

Leveraging Large Video Data: Building upon the idea of video-based interaction reconstruction, another promising research area involves leveraging the vast amounts of video data available for interaction learning. Machine learning models could be trained on these data, improving their ability to understand and reproduce realistic interactions. The methods presented in this thesis serve as an initial step in this direction by introducing effective representations and methods for learning interactions. In other words, instead of directly reconstructing human and object interaction motion from the videos, one can focus on additionally extracting rich features about the interaction such as contact, proximity field between the body and object, and their relative motions.

Exploring Optimal 3D Interaction Representations: While our research advanced the field, there is still room to explore the optimal 3D representations for human-object interaction. The challenge here is twofold: identifying which interaction features are crucial for representing the objects, body, and their interaction, and determining the most computationally efficient way to encode these features into the network without losing critical data. Current 3D representations for objects and scenes such as SDF, Voxel-grid, and implicit representations mainly focus on representing the shape and do not consider the interaction. Future work can explore new rich representations to lead to a more efficient and accurate synthesis of complex human-object interactions. The contributions of our research, particularly the GRAB dataset, have set a foundation for further investigation into this area.

Integrated Scene Navigation and Interaction: Current methods largely treat scene navigation and human-object interaction as separate tasks. However, real-world interactions involve continuous, seamless transitions between navigating an environment and interacting with objects. The challenge would be to merge the nuances of spatial navigation with the intricacies of object interaction, ensuring avatars not only reach their target but also interact with it convincingly. While both scene navigation and human-object interaction are crucial for realistic avatar behavior, they present distinct research challenges. Scene navigation operates on a larger scale, emphasizing route planning and object avoidance, with motions often spaced out and relying on multiple sensory inputs. In contrast, hand-object interactions demand fine-grained motions and precise manipulations based on the object's shape and position. Simply applying methods from hand-object interactions to scene navigation overlooks these nuances. Future work could aim to develop a holistic model that integrates these two components, generating natural and authentic virtual human behavior. The current models, GOAL and GRIP, could serve as the cornerstone for this future development with their respective abilities to generate realistic whole-body motions and detailed hand interactions.

Text-to-Interaction Synthesis: Finally, an intriguing future direction lies in developing a system that can generate accurate object interaction motion from detailed text descriptions. This could facilitate the creation of highly specific and custom interactions, based on user-specified textual prompts. This future direction aligns with the current state-of-the-art in Natural Language Processing (NLP),

where systems can generate meaningful outputs based on textual inputs. Currently, there is a lack of data for aligned interaction motions with the textual descriptions. The main challenge lies in the fact that language descriptions are not as details as the motion sequences. For instance, a simple directive like "making coffee" can encompass myriad variations contingent on one's position relative to 3D objects and the environment. The proposed direction of leveraging large video data could be one potential solution to solve the lack of data. But still, our understanding of object interaction, as highlighted in this thesis, will be fundamental. Therefore, the approaches proposed in this thesis, especially in terms of generating accurate hand-object interactions, could facilitate such advancements.

6.3 Summary

In summary, this Ph.D. thesis has deepened insights into three-dimensional human-object interaction, starting with the foundational GRAB dataset and advancing with the introduction of the GrabNet, GOAL, and GRIP models. These models, while significant, represent just one facet of a multifaceted domain. Each offers a unique perspective on the interaction process, from static grasping with GrabNet to the realistic finger articulation for object interaction by GRIP.

However, it is crucial to note that, like all research, this study has its limitations. While the models provide a nuanced representation of human-object interactions, there remain challenges in scalability, applicability to diverse scenarios, and real-time implementation, among others. Additionally, this thesis mainly focuses on the body motion that leads to grasping an object and less on moving or manipulating the objects. Future directions could encompass the automatic generation of body and object motion from action labels, leveraging video data for interaction learning, and probing optimal 3D interaction representations. The eventual goal, albeit ambitious, is a more comprehensive and authentic digital representation of humans, paving the way for enhanced virtual reality, ergonomics, and humanoid robotics applications.

Understanding human-object interactions remains an intricate endeavor, with both realized potentials and uncharted territories. This research underscores the importance of continued exploration in this domain, aiming to seamlessly integrate the digital and physical realms.

Appendices

Vision is the art of seeing what is invisible to others.

— Jonathan Swift

A

GRAB: The Human-Object Interaction Dataset

Contents

A.1 Protocol Details

Here we provide details of the data capture pipeline. We capture motions with 4 intents: "use", "pass", "lift", and "off-hand pass". For each sequence we randomize the object position and pose on a resting table, the height of which is also randomized between 75 cm and 120 cm to increase motion variance. We capture the following intents:

"Use": For the objects that have a clear everyday use (e.g. drinking from cup), we ask the subject to naturally use them. In case of multiple use cases (e.g. digital/analog photo camera) we capture multiple sequences. For objects without a clear use (e.g. cylinder) the subject has to grasp them and inspect them.

"Pass": The subject is asked to pass the object to a predefined direction, that is randomized (e.g. bottom-left, top-right, etc), to increase motion variance.

"Lift": The subject is asked to grasp the object, lift it stably in any natural way they can imagine, then leave it on the table in any random pose, and repeated this

several times with re-grasping. This increases grasp variance, by encouraging the exploration of contact configurations and relative hand-object orientations.

"Off-Hand pass": As a form of bimanual manipulation, the subject grasps the object with the off-hand, passes it to the dominant hand, and uses it (see "use").

We capture MoCap markers placed on the body, face and fingers, as well as on the object (Chapter 2, Sec 2.4). Additionally, we capture markers attached to the floor and the table, for potential future use. All subjects gave informed consent to share their motion data for research purposes.

A.2 Computing Contact

Here we provide some additional intuition to Sec. Sec. 2.4.5. In particular, we explain how we deal with noise in the reconstructed moving meshes to produce clean contact data.

Figure A.1 (left): Consider the illustrated example of a 3D cup. Its mesh thickness is thin, i.e. it has an outer and inner surface that are different, yet close to each other. In Fig. A.1 (top-left) the thumb and index fingers of a grasping hand penetrate both the outer and inner surface. This is due to noise, fitting errors, and because existing models do not model contact-dependent skin deformations.

For these examples the actual contact area is the one on the outer object surface. To annotate only this, following Sec. Sec. 2.4.5, we first compute all colliding triangles and cluster them in connected "rings" (Fig. A.1, top-left). For each "ring" we compute the corresponding penetrating hand areas (Fig. A.1, bottom). The hand areas that contact the inner surface are a subset of the ones that contact the outer one. Then, we remove (big red circles in Fig. A.1 bottom) the purple and green groups, we keep only the remaining "rings", and annotate the vertices enclosed by them as contact vertices (Fig. A.1, top-right).

Figure A.1 (right): The above procedure gives binary contact annotations ("contact" on "not in contact"). Contact labels, however, can be more fine-grained, e.g. with the label of the corresponding hand part (Fig. A.1, right), or even with the point on the 3D hand surface (Fig. A.3). For the former example, we find the object vertices that are in contact, and for each one we find the closest SMPL-X/MANO bone, and assign its ID as the contact label.

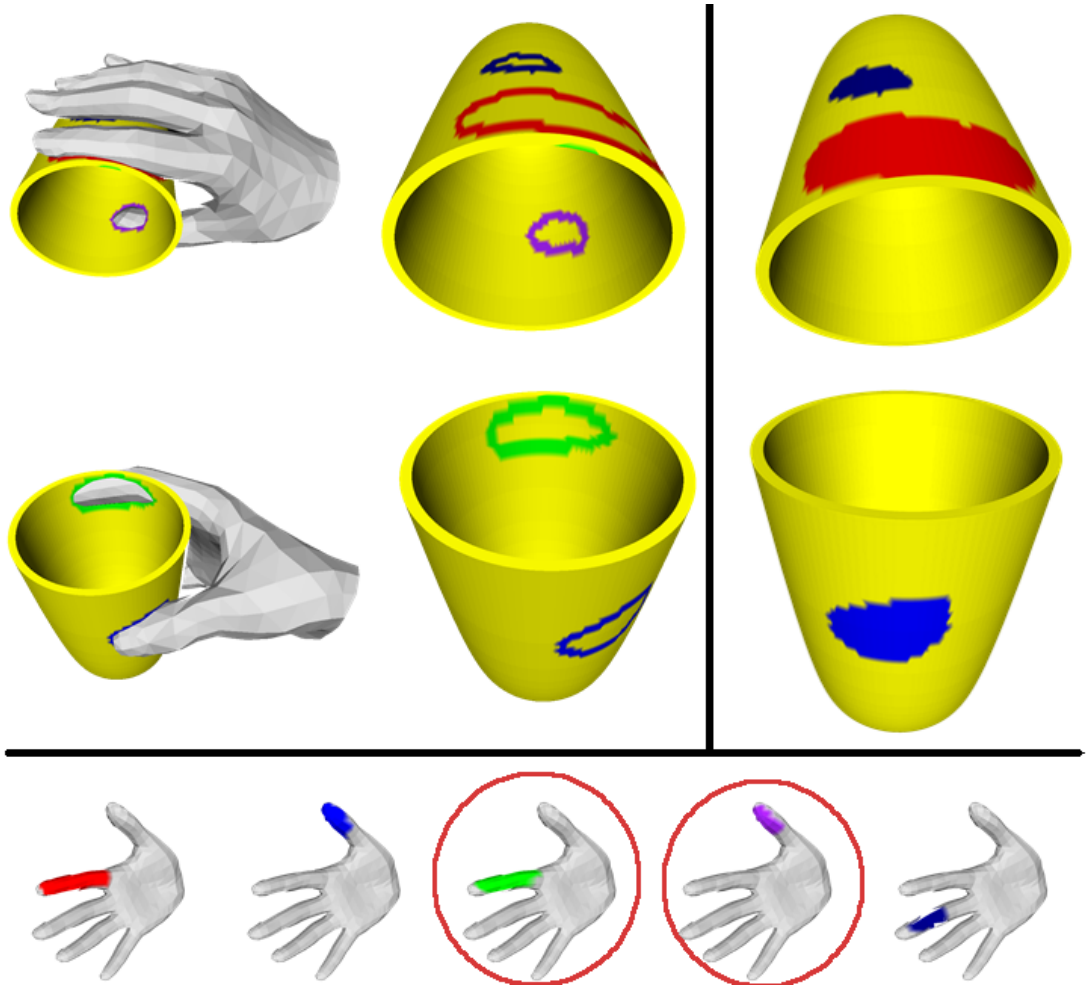


Figure A.1: Annotating contact areas for a hand grasping a cup: (top-left) "rings" of colliding triangles, color-coded for each finger, (bottom) penetrating hand areas that correspond to each "ring"; the ones corresponding to the green and purple vertices (red circled hand parts) penetrate the inner cup surface and are ignored, (top-right) the final filtered "rings" and the enclosed vertices are annotated as contact areas. The contact labels are binary; color is used here only for visualization purposes.

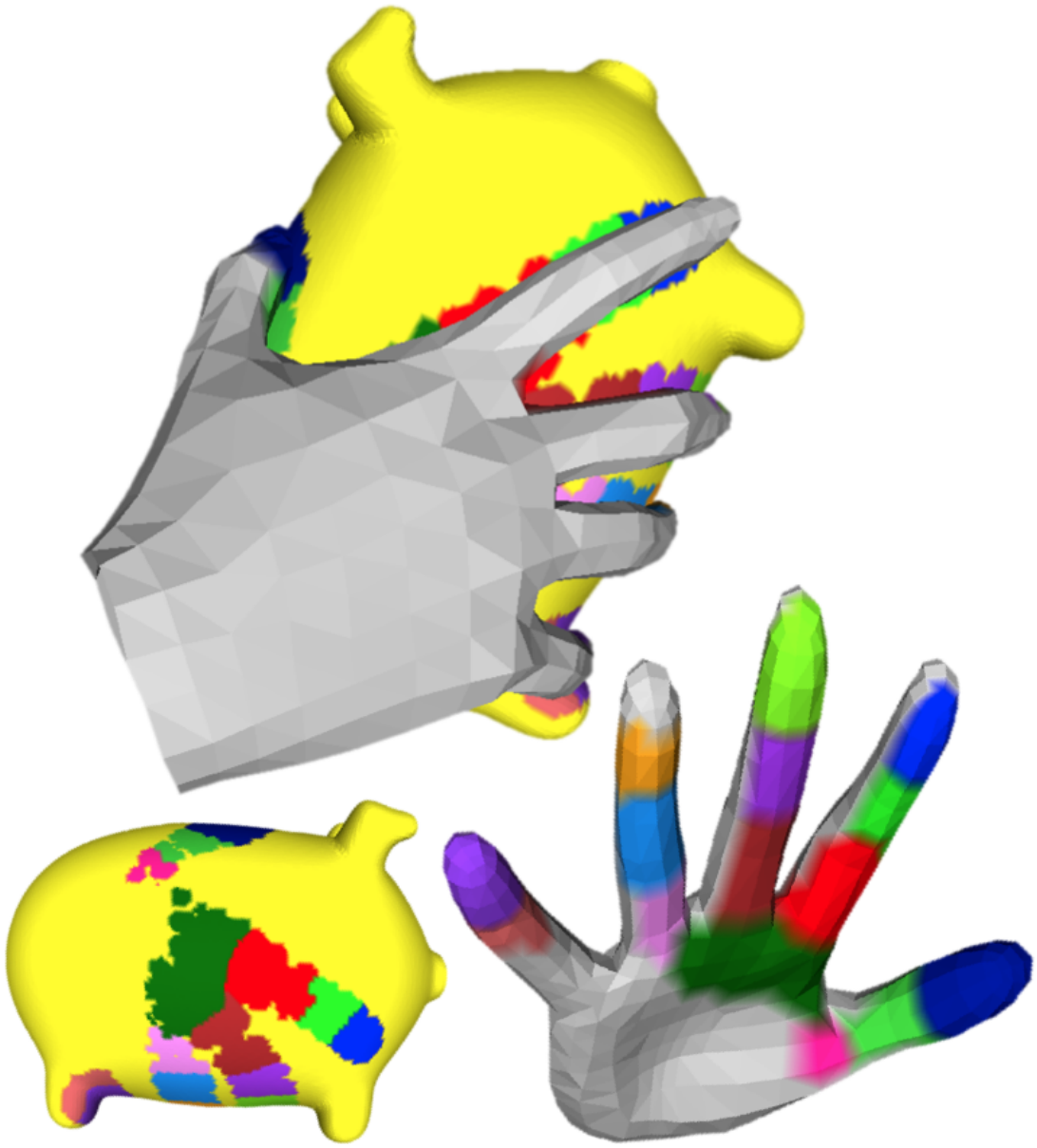


Figure A.2: Contact labels can though be more fine-grained, e.g. using the contacting hand parts or hand vertices. Here we see an example of the former case. (top) Each color represents a contact area caused by a different hand part. (bottom) Contact areas are shown also on the object and unposed hand for clarity. Note that the size of contact areas is expanded for illustration purposes.

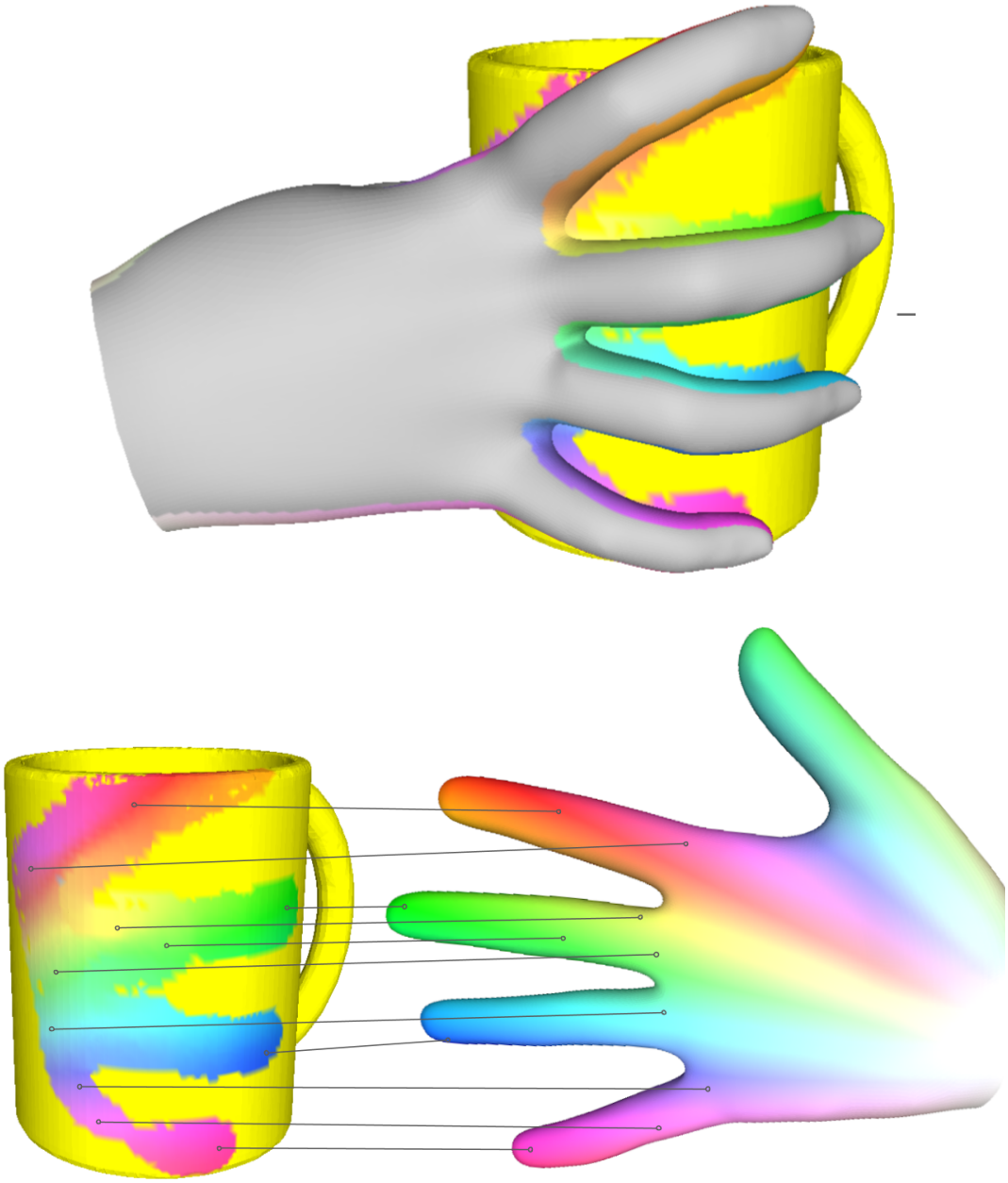


Figure A.3: (Right): Fine-grained contact labels. In contrast to the binary contact labels of Fig. 2.7 and Fig. 3.5 of Chapter 3, and the part-based contact labels of Fig. A.1, here we show an example of much more fine-grained labels. **(left)** A 3D hand-object grasp configuration. **(middle)** The object alone. **(right)** The hand in canonical pose. We highlight different points on the 3D surface of the inner hand with color gradients. The contact between the hand and the object defines surface correspondences between them (shown as lines).

A.2.1 Heatmaps Analysis for Various Intents

Fig. A.4 provides additional fine-grained numbers for in-contact parts of the body. Each row corresponds to a motion intent in the GRAB dataset. For each intent, the right column shows the contact percentage and "heatmap" for the right hand, left hand, and head across all frames and relative to *all* body vertices. In the three left columns, the "heatmaps" and percentages are relative to only *each part's* vertices and for only the frames for which these parts are in contact (left hand, right hand, and head), for visualization purposes. For example, for the "use" sequences (second row in Fig A.4), the right hand was in contact for 90.62% of all frames, and in those contact frames the thumb fingertip was in contact for 99.88% of them.

A.3 Bias from MoCap Markers

A natural question arises - are subjects biased in their grasps by MoCap markers? We empirically place more markers in areas less likely to be contacted, according to object affordances. To account for potential occlusions, though, we have to place some markers in other areas as well. For this reason, we still expect our markers to be contacted.

Our subjects did not complain about discomfort or bias, yet we need more evidence for this. Apart from the analysis in the main manuscript (see Sec. 2.5 and Fig. 2.10), here we perform k-means clustering (k=20) on our grasps, and visualize each cluster center, i.e. a grasping hand, and the grasped object. We observe that several clusters (typically 3-6 out of 20) show that fingers do come in contact with markers. Figure A.5 shows for 5 objects (rows) 3 representative contacting clusters (columns). We believe this is good additional empirical evidence that our 1.5 mm radius hemispherical markers cause no or minimal bias.

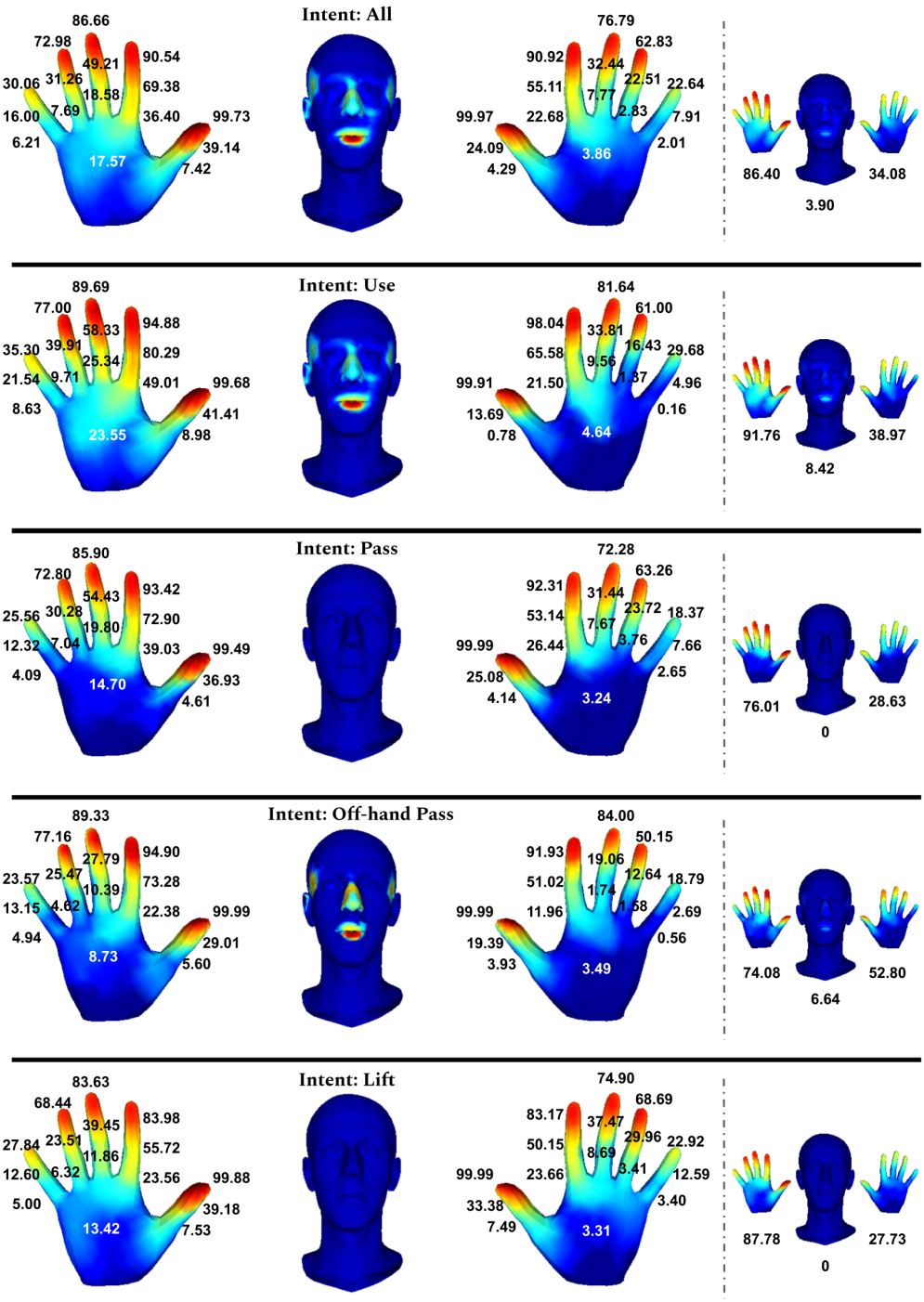


Figure A.4: Contact "heatmaps" and percentages for all intents in GRAB, for various body parts. Each row corresponds to an intent in the GRAB dataset. For each intent, the right column shows the results for each part (right hand, left hand, and head) across all frames and relative to *all* body vertices. In the three left columns, the results are relative to only *each part's* vertices and for only frames for which these parts are in contact (left hand, right hand, and head), for visualization purposes.

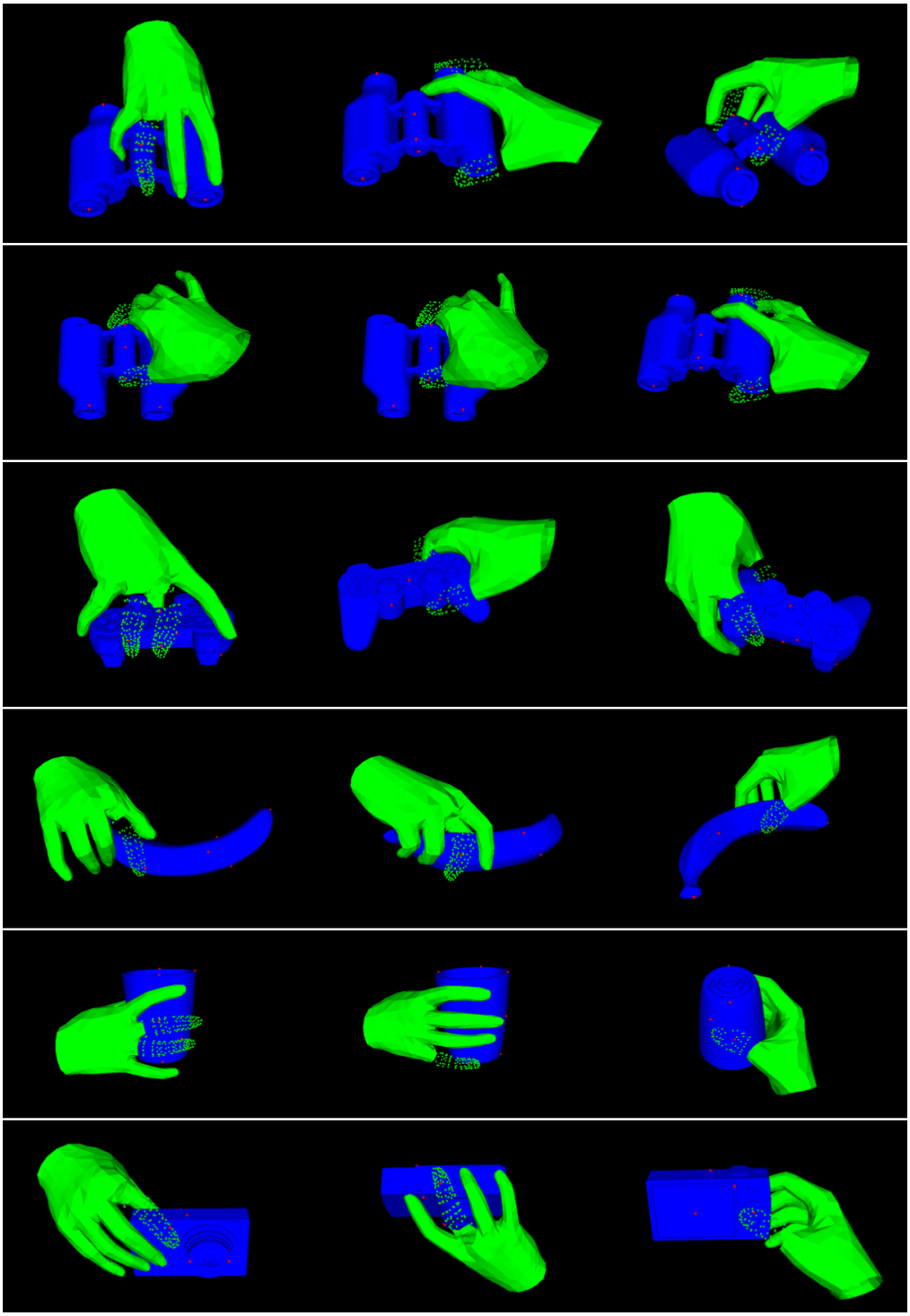


Figure A.5: Do subjects avoid markers? To answer this, we perform k-means clustering (k=20) on our grasps, and visualize each cluster center, i.e. a grasping MANO (green), and the grasped object (blue). We observe that several cluster centers (columns) per object (rows) show that subjects contact MoCap markers (red); here we show 3 clusters for 5 objects. For fingers that contact markers we render only the vertices, to allow to see the markers (best viewed on screen).

We have two hands. One to help ourselves, the second to help others.

— Audrey Hepburn

B

GrabNet: Generating Static Grasps for 3D Objects

Contents

B.1 Results: Success and Failure Cases

Figures B.2, B.3 and B.4 provide a wide variety of qualitative GrabNet results. More specifically, they show 10 different grasps (rows) generated for 6 unseen objects (columns). The three figures show three different viewpoints (one view per figure) for the same grasp of the 10×6 grid. We see that most grasps look natural and plausible, as GrabNet is learned from high-quality GRAB captures. More results with a rotating viewpoint are shown in the video on our website.

GrabNet can still generate some failure cases. These are mostly cases of penetrating fingers; there are not many cases of contacting fingers that fly away from the object. Penetrations are observed mostly for objects with thin parts (cup handle, wine glass, bowl). We found the frying pan to be the most challenging object, due to its comparably big size along with its thin surface walls and handle. This might be due to the sparse BPS_o representation for 3D object shapes capturing mostly their bigger parts. Furthermore, at the moment we use a penetration and

a contact term in the training loss of GrabNet as soft constraints, since here we focus on a data-driven method. One could add an optimization stage to refine the regressed grasp with hard penetration and contact constraints.

The results show the value of GRAB for training data-driven models, but also point to room for improvement for GrabNet's modeling and training scheme.

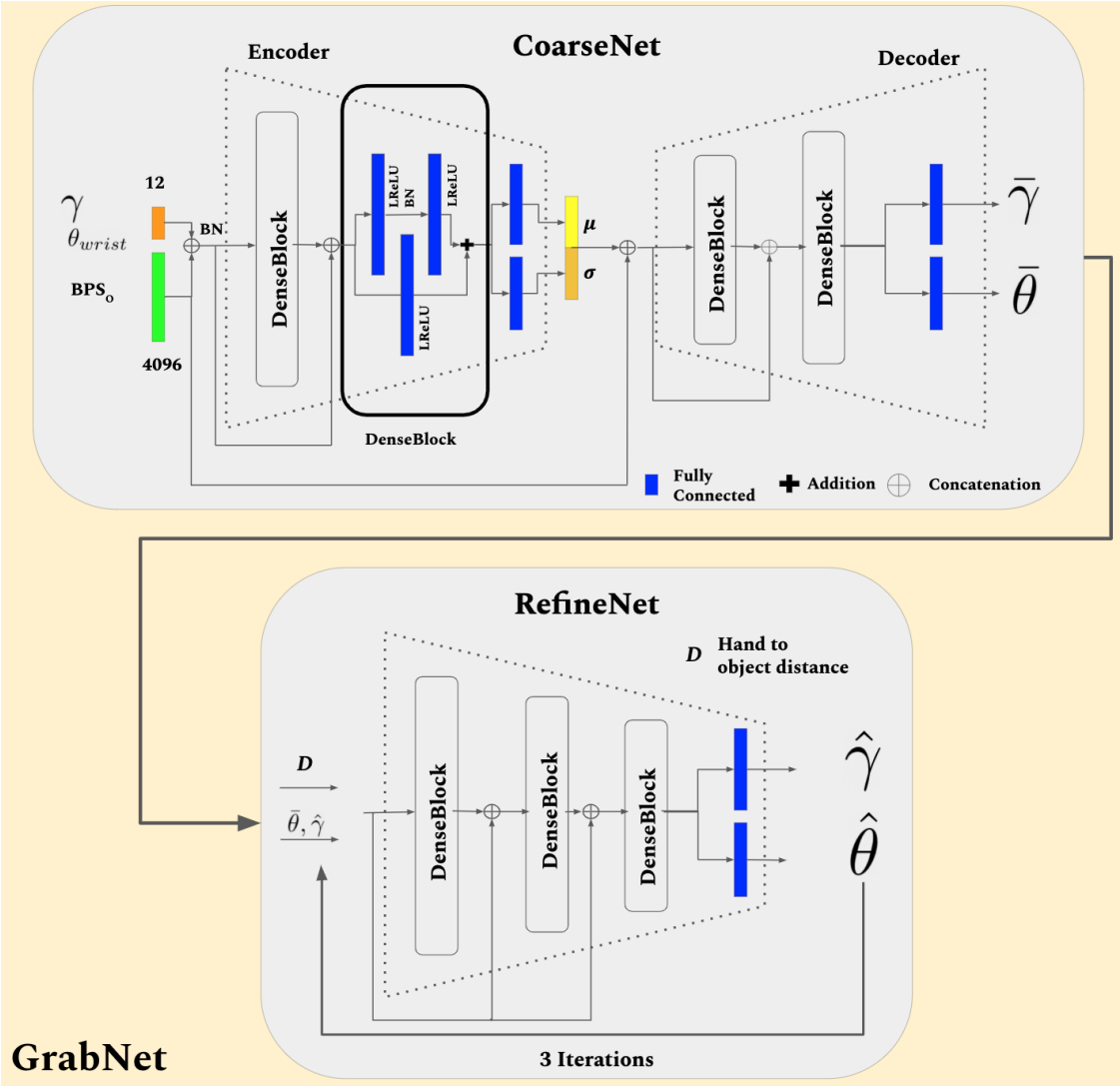


Figure B.1: GrabNet Architecture. For the encoder input, we concatenate the BPS representation of the object with MANO parameters, while for decoder input we concatenate it with a sample from latent space. The decoder gives the MANO hand parameters which we pass to the MANO model to obtain the 3D hand mesh.

B.2 GrabNet Implementation Details

A detailed architecture for GrabNet is shown in Fig. B.1. For CoarseNet, we concatenate the object BPS_o representation with MANO hand parameters as input to the encoder, and also concatenate it with the latent code as the input (condition) to the decoder. The outputs of the decoder are MANO translation ($\gamma \in R^3$) and joint angles ($\theta \in R^{96}$) in the continuous 6-dimensional representation of [152].

Using our validation set, we found out that 16 dimensions for the latent space results in generating better grasps. Qualitative results are provided in Fig B.2.

For RefineNet we take the output of the CoarseNet (MANO parameters) and first compute the distances of MANO vertices to the object vertices. We then pass the distances with the MANO parameters to the network. RefineNet refines the input grasp through 3 iterations. The CoarseNet and RefineNet are trained for 16 and 23 epochs respectively with the learning rate starting from $5e-4$, decreasing on validation error plateau to 0.1 times, and early stopping after 8 epochs with no improvement in validation error. Both networks are trained separately.

B.3 Filtering out Unreliable Turkers

As mentioned in Chapter 2, along with ground-truth (GRAB) and GrabNet-generated grasps, we pass to Turkers noisy grasps generated by perturbing ground-truth ones. These noisy grasps are our test for spotting unreliable Turkers, that either select their answers randomly or misunderstand the task. Specifically, we remove the ones that gave a rating of 3 or more (indicating good realism) for at least 20% of these noisy grasps. In total we removed 54 out of 170 Turkers.

Figure B.2: Visualization of 10 different grasps (rows) generated by GrabNet for 6 unseen objects (columns). Conditioned on the BPS_o representation of unseen 3D object shapes, we sample from the learned 16 dimensional latent grasping space Z of CoarseNet. We then concatenate BPS_o to the Z sample and pass them to the decoder of CoarseNet, which outputs the coarse grasping MANO hand model parameters. We then pass the coarse grasps to the RefineNet to get the final grasps. Some failure cases are highlighted in red. Different viewpoints for the results of Fig. B.3, B.4.

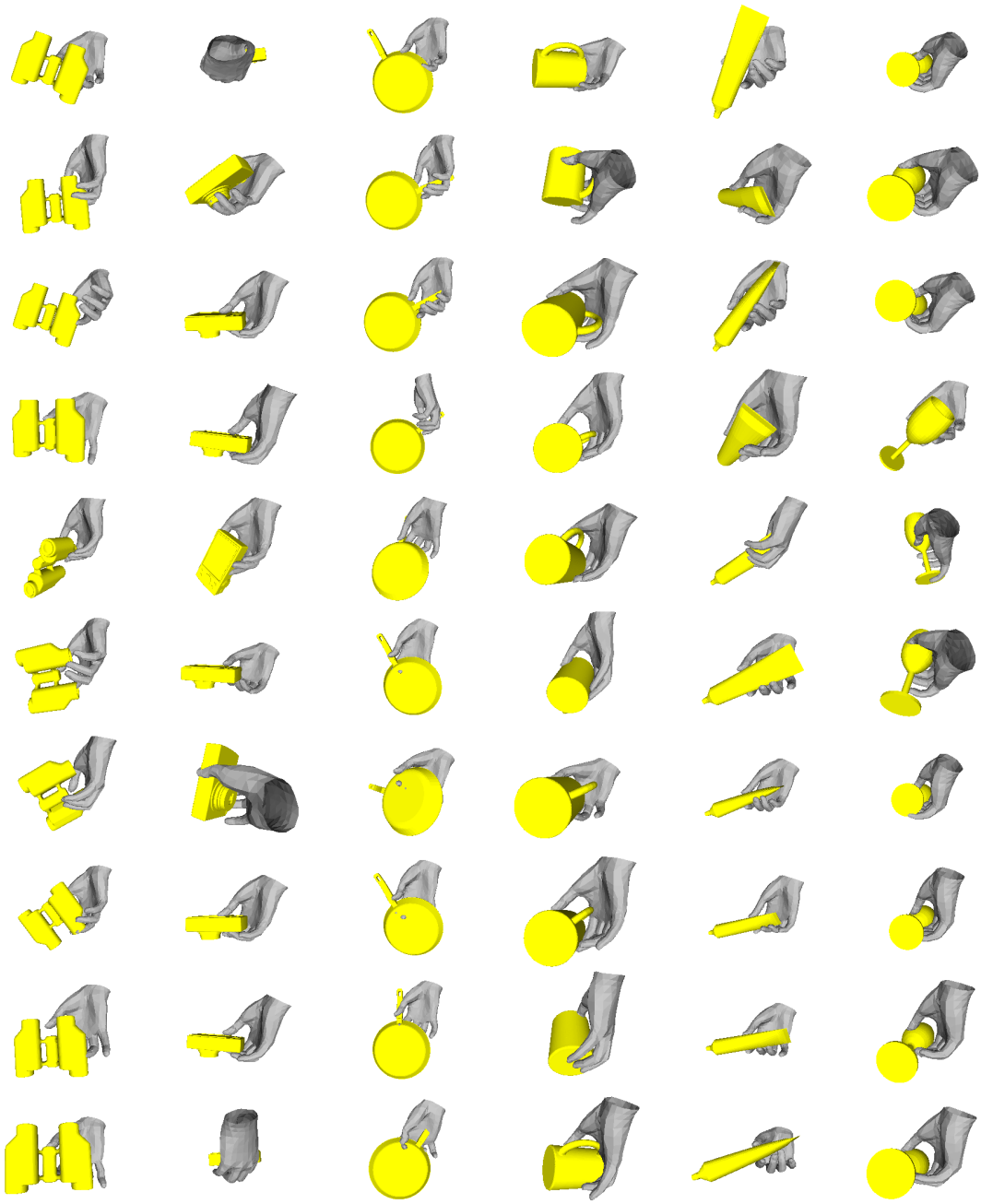


Figure B.3: Visualization of 10 different grasps (rows) generated by GrabNet for 6 unseen objects (columns). Conditioned on the BPS_o representation of unseen 3D object shapes, we sample from the learned 16 dimensional latent grasping space Z of CoarseNet. We then concatenate BPS_o to the Z sample and pass them to the decoder of CoarseNet, which outputs the coarse grasping MANO hand model parameters. We then pass the coarse grasps to the RefineNet to get the final grasps. Some failure cases are highlighted in red. Different viewpoints for the results of Fig. B.2, B.4.

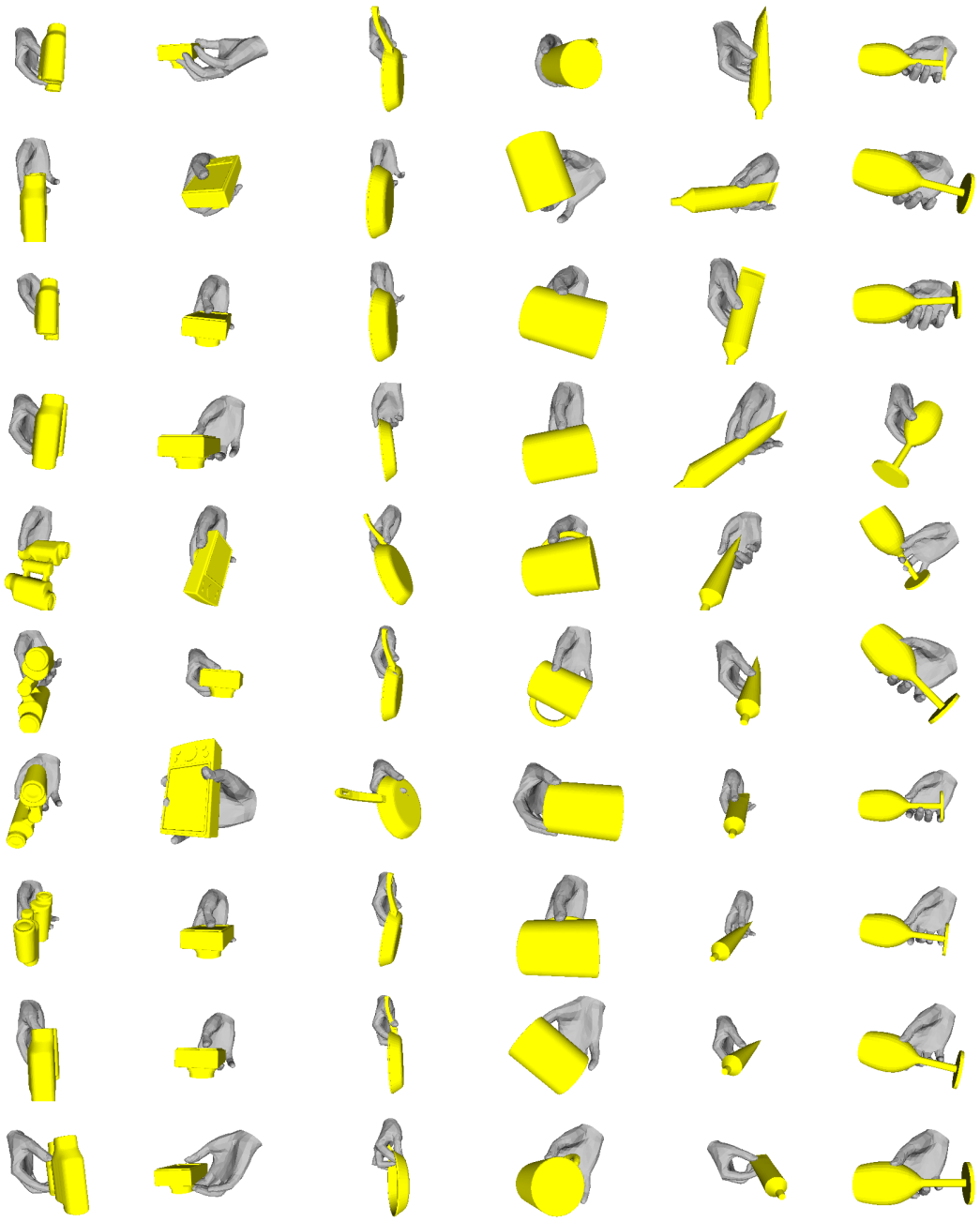


Figure B.4: Visualization of 10 different grasps (rows) generated by GrabNet for 6 unseen objects (columns). Conditioned on the BPS_o representation of unseen 3D object shapes, we sample from the learned 16 dimensional latent grasping space Z of CoarseNet. We then concatenate BPS_o to the Z sample and pass them to the decoder of CoarseNet, which outputs the coarse grasping MANO hand model parameters. We then pass the coarse grasps to the RefineNet to get the final grasps. Some failure cases are highlighted in red. Different viewpoints for the results of Fig. B.2, B.3.

The human hand has the power to grasp not only physically but metaphorically.

— Heidi Barr

C

GRIP: Hand Interaction Poses for Object and Body Motion

Contents

C.1 Physics Simulation

Our main goal is to generate visually plausible hand-object interaction motions, however we also evaluate the physical plausibility of our results, which may be important for the real-world applications. Following prior methods [17, 150, 197], we evaluate the generated grasps in a Bullet physics simulation. We fix the body position and apply gravity to the object. A small object displacement (<1 mm) after 5 physics simulation steps is counted as a "stable" grasp. For all generated grasps, CNet and RNet have 93% and 97% stability, respectively. This suggests that the synthesized hand poses are not just visually pleasing but also physically realistic.

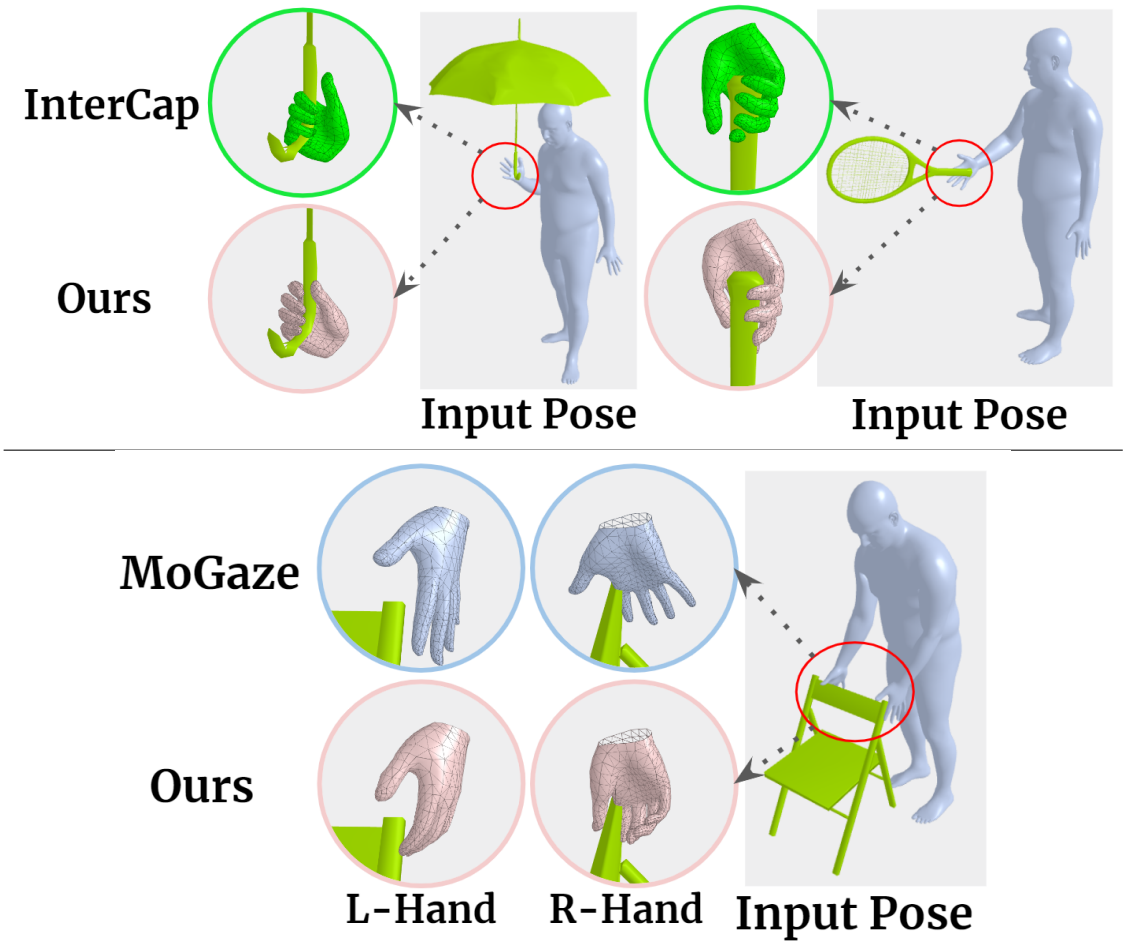


Figure C.1: GRIP's performance to generate hand grasps for large objects. We generate hand poses on unseen large objects from Intercap (Top) and MoGaze (Bottom) datasets. These objects have larger 3D structures compared to the 3D objects during training, however, our hand sensors are not distracted by the extended objects due to their locality. Thus, GRIP is able to generate plausible grasps for such objects.

C.2 Performance on Large Objects

In Fig. C.1 we show more qualitative results of our method performance to generate hand grasps for large objects. Note that these objects have extended 3D structure compared with all the training objects in the GRAB dataset. What is important to note here is that our hand sensors are not distracted by the extended objects due to their locality. Thus GRIP is able to generate plausible grasps for such objects.

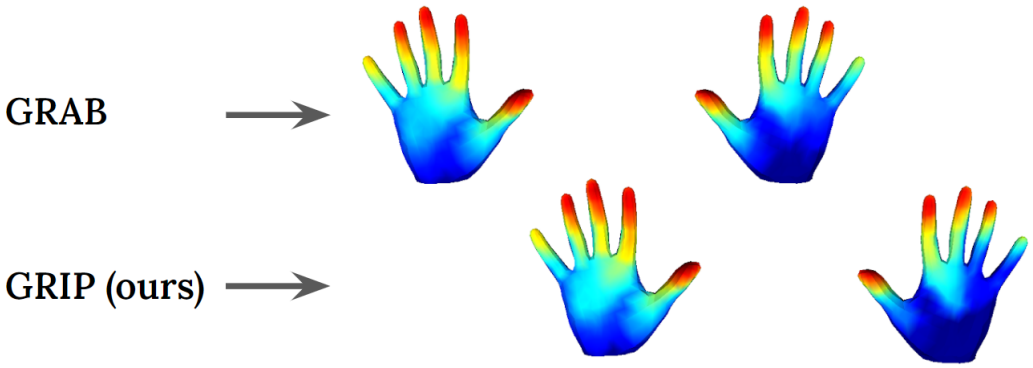


Figure C.2: Comparison of the contact heatmaps from GRAB and GRIP. We compute contact vertices on both the left and right-hand and aggregate them across all frames. Results show that GRIP contact maps are similar to GRAB, which is indicative of the realism of the generated hand grasps.

C.3 Grasp Analysis

To further evaluate the quality of the generated grasps from GRIP, we compare the aggregated contact heatmaps from our method with GRAB [46]. For each motion frame in the test set, we compute the contact vertices on both hands based on their distance to the object surface, similar to GRAB. We then aggregate the contact maps across all frames to compute the overall contact heatmap. Figure C.2 (top) shows the contact heatmap from GRAB and (bottom) shows the heatmaps for GRIP. Areas with a high likelihood of contact are shown with “hot" (red) colors and with a low likelihood of contact are shown with “cool” (blue) colors. We see that GRIP contact maps follow a similar pattern to GRAB, and have higher contact likelihood on the fingertips. The similarity suggests that generated grasps exhibit similar contacts as real grasps.

References

[1] Mark R Cutkosky. "On grasp choice, grasp models, and the design of hands for manufacturing tasks". In: *IEEE Transactions on Robotics and Automation* 5.3 (1989), pp. 269–279.

[2] Thomas Feix, Javier Romero, Heinz-Bodo Schmiedmayer, Aaron M. Dollar, and Danica Kragic. "The GRASP Taxonomy of Human Grasp Types". In: *IEEE Transactions on Human-Machine Systems* 46.1 (2016), pp. 66–77.

[3] Júlia Borras and Tamim Asfour. "A whole-body pose taxonomy for loco-manipulation tasks". In: *International Conference on Intelligent Robots and Systems (IROS)*. 2015, pp. 1578–1585.

[4] Kaijen Hsiao and Tomas Lozano-Perez. "Imitation Learning of Whole-Body Grasps". In: *International Conference on Intelligent Robots and Systems (IROS)*. 2006, pp. 5657–5662.

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

[6] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. "Resolving 3D Human Pose Ambiguities with 3D Scene Constrains". In: *International Conference on Computer Vision (ICCV)*. 2019.

[7] T. Pham, N. Kyriazis, A. A. Argyros, and A. Kheddar. "Hand-Object Contact Force Estimation from Markerless Visual Tracking". In: *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 40.12 (2018), pp. 2883–2896.

[8] Keni Bernardin, Koichi Ogawara, Katsushi Ikeuchi, and Ruediger Dillmann. "A sensor fusion approach for recognizing continuous human grasping sequences using hidden Markov models". In: *IEEE Transactions on Robotics (T-RO)* 21.1 (2005), pp. 47–57.

[9] Micah K. Johnson, Forrester Cole, Alvin Raj, and Edward H. Adelson. "Microgeometry Capture using an Elastomeric Sensor". In: *Transactions on Graphics (TOG)* 30.4 (2011), 46:1–46:8.

[10] *Pressure Profile Systems Inc. (PPS)*. https://pressureprofile.com.

[11] *CyberGlove III Data Glove*. http://www.cyberglovesystems.com/cyberglove-iii.

[12] Stephen A Mascaro and H Harry Asada. "Photoplethysmograph fingernail sensors for measuring finger forces without haptic obstruction". In: *IEEE Transactions on Robotics and Automation (TRA)* 17.5 (2001), pp. 698–708.

[13] Samarth Brahmbhatt, Cusuh Ham, Charles C. Kemp, and James Hays. "ContactDB: Analyzing and Predicting Grasp Contact via Thermal Imaging". In: *Computer Vision and Pattern Recognition (CVPR)*. 2019.

[14] Marcus A. Brubaker, David J. Fleet, and Aaron Hertzmann. "Physics-Based Person Tracking Using the Anthropomorphic Walker". In: *International Journal of Computer Vision (IJCV)* 87.1 (2009), p. 140.

[15] H. Kjellstrom, D. Kragic, and M. J. Black. "Tracking people interacting with objects". In: *Computer Vision and Pattern Recognition (CVPR)*. 2010.

[16] Zongmian Li, Jiri Sedlar, Justin Carpentier, Ivan Laptev, Nicolas Mansard, and Josef Sivic. "Estimating 3D Motion and Forces of Person-Object Interactions From Monocular Video". In: *Computer Vision and Pattern Recognition (CVPR)*. 2019.

[17] Yana Hasson, Gül Varol, Dimitris Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. "Learning joint reconstruction of hands and manipulated objects". In: *Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 11807–11816.

[18] Vladimir G. Kim, Siddhartha Chaudhuri, Leonidas Guibas, and Thomas Funkhouser. "Shape2pose: Human-centric shape analysis". In: *Transactions on Graphics (TOG)* 33.4 (2014), 120:1–120:12.

[19] Manolis Savva, Angel X Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. "PiGraphs: Learning interaction snapshots from observations". In: *Transactions on Graphics (TOG)* 35.4 (2016), 139:1–139:12.

[20] Christian Mandery, Ömer Terlemez, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. "The KIT whole-body human motion database". In: *International Conference on Advanced Robotics (ICAR)*. 2015, pp. 329–336.

[21] Norman I. Badler, Cary B. Phillips, and Bonnie Lynn Webber. *Simulating Humans: Computer Graphics Animation and Control*. USA: Oxford University Press, Inc., 1993.

[22] Matthew Brand and Aaron Hertzmann. "Style machines". In: *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*. 2000, pp. 183–192.

[23] Jack M. Wang, David J. Fleet, and Aaron Hertzmann. "Gaussian Process Dynamical Models for Human Motion". In: *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 30.2 (2008), pp. 283–298.

[24] Mathis Petrovich, Michael J. Black, and Gül Varol. "Action-Conditioned 3D Human Motion Synthesis with Transformer VAE". In: *International Conference on Computer Vision (ICCV)*. 2021, pp. 10985–10995.

[25] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. "History Repeats Itself: Human Motion Prediction via Motion Attention". In: *European Conference on Computer Vision (ECCV)*. Vol. 12359. 2020, pp. 474–489.

[26] Ye Yuan and Kris Kitani. "DLow: Diversifying Latent Flows for Diverse Human Motion Prediction". In: *European Conference on Computer Vision (ECCV)*. Vol. 12354. 2020, pp. 346–364.

[27] Yan Zhang, Michael J. Black, and Siyu Tang. "We Are More Than Our Joints: Predicting How 3D Bodies Move". In: *Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 3372–3382.

[28] Michael Gleicher. "Retargetting motion to new characters". In: *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*. 1998, pp. 33–42.

[29] Jehee Lee, Jinxiang Chai, Paul S. A. Reitsma, Jessica K. Hodgins, and Nancy S. Pollard. "Interactive control of avatars animated with human motion data". In: *Transactions on Graphics (TOG)* 21.3 (2002), pp. 491–500.

[30] Kang Hoon Lee, Myung Geol Choi, and Jehee Lee. "Motion patches: Building blocks for virtual environments annotated with motion data". In: *Transactions on Graphics (TOG)* 25.3 (2006), pp. 898–906.

[31] Mubbasir Kapadia, Xu Xianghao, Maurizio Nitti, Marcelo Kallmann, Stelian Coros, Robert W. Sumner, and Markus Gross. "Precision: Precomputing environment semantics for contact-rich character animation". In: *Symposium on Interactive 3D Graphics (SI3D)*. 2016.

[32] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. "DeepMimic: Example-guided deep reinforcement learning of physics-based character skills". In: *Transactions on Graphics (TOG)* 37.4 (2018), 143:1–143:14.

[33] Xue Bin Peng, Glen Berseth, and Michiel Van de Panne. "Terrain-adaptive locomotion skills using deep reinforcement learning". In: *Transactions on Graphics (TOG)* 35.4 (2016), 81:1–81:12.

[34] Yu-Wei Chao, Jimei Yang, Weifeng Chen, and Jia Deng. "Learning to Sit: Synthesizing Human-Chair Interactions via Hierarchical Control". In: *Conference on Artificial Intelligence (AAAI)*. 2021, pp. 5887–5895.

[35] Grégory Rogez, James S. Supančič III, and Deva Ramanan. "Understanding Everyday Hands in Action from RGB-D Images". In: *International Conference on Computer Vision (ICCV)*. 2015.

[36] George ElKoura and Karan Singh. "Handrix: Animating the human hand". In: *Symposium on Computer Animation (SCA)*. 2003, pp. 110–119.

[37] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. "Neural State Machine for Character-Scene Interactions". In: *Transactions on Graphics (TOG)* 38.6 (2019), 209:1–209:14.

[38] Nancy S. Pollard and Victor Brian Zordan. "Physically based grasping control from example". In: *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*. 2005, pp. 311–318.

[39] Paul G. Kry and Dinesh K. Pai. “Interaction capture and synthesis”. In: *Transactions on Graphics (TOG)* 25.3 (2006), pp. 872–880.

[40] Yuting Ye and C. Karen Liu. “Synthesis of detailed hand manipulations using contact sampling”. In: *Transactions on Graphics (TOG)* 31.4 (2012), 41:1–41:10.

[41] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. “GRAB: A Dataset of Whole-Body Human Grasping of Objects”. In: *European Conference on Computer Vision (ECCV)*. Vol. 12349. 2020, pp. 581–600.

[42] Patrick Grady, Chengcheng Tang, Christopher D. Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C. Kemp. “ContactOpt: Optimizing Contact To Improve Grasps”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 1471–1481.

[43] John R Napier. “The prehensile movements of the human hand”. In: *The Journal of bone and joint surgery* 38.4 (1956), pp. 902–913.

[44] Noriko Kamakura, Michiko Matsuo, Harumi Ishii, Fumiko Mitsuboshi, and Yoriko Miura. “Patterns of Static Prehension in Normal Hands”. In: *American Journal of Occupational Therapy* 34.7 (1980), pp. 437–445.

[45] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. “AMASS: Archive of Motion Capture as Surface Shapes”. In: *International Conference on Computer Vision (ICCV)*. 2019.

[46] Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. “TEACH: Temporal Action Compositions for 3D Humans”. In: *International Conference on 3D Vision (3DV)*. 2022.

[47] Holly A Ruff. “Infants’ manipulative exploration of objects: Effects of age and object characteristics.” In: *Developmental Psychology* 20.1 (1984), p. 9.

[48] Anis Sahbani, Sahar El-Khoury, and Philippe Bidaud. “An overview of 3D object grasp synthesis algorithms”. In: *Robotics and Autonomous Systems (RAS)* 60.3 (2012), pp. 326–336.

[49] Zhe Wang, Liyan Chen, Shaurya Rathore, Daeyun Shin, and Charless Fowlkes. “Geometric Pose Affordance: 3D Human Pose with Scene Constraints”. In: *arXiv:1905.07718* (2019).

[50] Shangchen Han, Beibei Liu, Robert Wang, Yuting Ye, Christopher D Twigg, and Kenrick Kin. “Online optical marker-based hand tracking with deep labels”. In: *Transactions on Graphics (TOG)* 37.4 (2018), 166:1–166:10.

[51] Huy Viet Le, Sven Mayer, Patrick Bader, and Niels Henze. “Fingers’ Range and Comfortable Area for One-Handed Smartphone Interaction Beyond the Touchscreen”. In: *CHI Conference on Human Factors in Computing Systems*. 2018.

[52] Anna Maria Feit, Daryl Weir, and Antti Oulasvirta. “How we type: Movement strategies and performance in everyday typing”. In: *CHI Conference on Human Factors in Computing Systems*. 2016.

[53] Subramanian Sundaram, Petr Kellnhofer, Yunzhu Li, Jun-Yan Zhu, Antonio Torralba, and Wojciech Matusik. "Learning the signatures of the human grasp using a scalable tactile glove". In: *Nature* 569.7758 (2019), pp. 698–702.

[54] *Tekscan Grip System: Tactile Grip Force and Pressure Sensing.* https://www.tekscan.com/products-solutions/systems/grip-system.

[55] *GelSight tactile sensor.* http://www.gelsight.com.

[56] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. "First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations". In: *Computer Vision and Pattern Recognition (CVPR)*. 2018.

[57] Shreyas Hampali, Markus Oberweger, Mahdi Rad, and Vincent Lepetit. "HO-3D: A Multi-User, Multi-Object Dataset for Joint 3D Hand-Object Pose Estimation". In: *Computer Vision and Pattern Recognition (CVPR)*. 2019.

[58] Ankur Handa, Karl Van Wyk, Wei Yang, Jacky Liang, Yu-Wei Chao, Qian Wan, Stan Birchfield, Nathan Ratliff, and Dieter Fox. "DexPilot: Vision Based Teleoperation of Dexterous Robotic Hand-Arm System". In: *International Conference on Robotics and Automation (ICRA)*. 2019.

[59] Henning Hamer, Juergen Gall, Thibaut Weise, and Luc Van Gool. "An Object-Dependent Hand Pose Prior from Sparse Training Data". In: *Computer Vision and Pattern Recognition (CVPR)*. 2010.

[60] Aron Monszpart, Paul Guerrero, Duygu Ceylan, Ersin Yumer, and Niloy J Mitra. "iMapper: Interaction-guided scene mapping from monocular videos". In: *Transactions on Graphics (TOG)* 38.4 (2019), 92:1–92:15.

[61] N. Hasler, B. Rosenhahn, T. Thormahlen, M. Wand, J. Gall, and H. Seidel. "Markerless Motion Capture with unsynchronized moving cameras". In: *Computer Vision and Pattern Recognition (CVPR)*. 2009.

[62] B. Rosenhahn, C. Schmaltz, T. Brox, J. Weickert, D. Cremers, and H. Seidel. "Markerless motion capture of man-machine interaction". In: *Computer Vision and Pattern Recognition (CVPR)*. 2008.

[63] M. Yamamoto and K. Yagishita. "Scene constraints-aided tracking of human body". In: *Computer Vision and Pattern Recognition (CVPR)*. 2000.

[64] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A. Argyros. "Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints". In: *International Conference on Computer Vision (ICCV)*. 2011.

[65] Javier Romero, Hedvig Kjellström, and Danica Kragic. "Hands in action: Real-time 3D reconstruction of hands in interaction with objects". In: *International Conference on Robotics and Automation (ICRA)*. 2010.

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

[67] Aggeliki Tsoli and Antonis A. Argyros. “Joint 3D tracking of a deformable object in interaction with a hand”. In: *European Conference on Computer Vision (ECCV)*. 2018.

[68] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. “Capturing Hands in Action using Discriminative Salient Points and Physics Simulation”. In: *International Journal of Computer Vision (IJCV)* 118.2 (2016), pp. 172–193.

[69] Yangang Wang, Jianyuan Min, Jianjie Zhang, Yebin Liu, Feng Xu, Qionghai Dai, and Jinxiang Chai. “Video-based hand manipulation capture through composite motion control”. In: *Transactions on Graphics (TOG)* 32.4 (2013), 43:1–43:14.

[70] Hao Zhang, Zi-Hao Bo, Jun-Hai Yong, and Feng Xu. “InteractionFusion: Real-time reconstruction of hand poses and deformable objects in hand-object interactions”. In: *Transactions on Graphics (TOG)* 38.4 (2019), 48:1–48:11.

[71] *POSER: 3D Rendering and Animation Software*. https://www.posersoftware.com.

[72] Andrew T. Miller and Peter K. Allen. “Graspit! A versatile simulator for robotic grasping”. In: *IEEE Robotics Automation Magazine (RAM)* 11.4 (2004), pp. 110–122.

[73] Javier Romero, Dimitrios Tzionas, and Michael J. Black. “Embodied hands: Modeling and capturing hands and bodies together”. In: *Transactions on Graphics (TOG)* 36.6 (2017), 245:1–245:17.

[74] Angel X. Chang, homas A. Funkhouser, eonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. “ShapeNet: An Information-Rich 3D Model Repository”. In: *arXiv:1512.03012* (2015).

[75] Corey Goldfeder, Matei T. Ciocarlie, Hao Dang, and Peter K. Allen. “The Columbia grasp database”. In: *International Conference on Robotics and Automation (ICRA)*. 2009.

[76] M. Kokic, D. Kragic, and J. Bohg. “Learning Task-Oriented Grasping From Human Activity Datasets”. In: *IEEE Robotics and Automation Letters (RA-L)* 5.2 (2020), pp. 3352–3359.

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

[78] Manolis Savva, Angel X Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. “SceneGrok: Inferring action maps in 3D environments”. In: *Transactions on Graphics (TOG)* 33.6 (2014), 212:1–212:10.

[79] Enric Corona, Albert Pumarola, Guillem Alenyà, and Francesc Moreno-Noguer. “Context-aware Human Motion Prediction”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 6990–6999.

[80] *XSENS: Inertial Motion Capture*. https://www.xsens.com/motion-capture.

[81] Yan Zhang, Mohamed Hassan, Heiko Neumann, Michael J. Black, and Siyu Tang. "Generating 3D People in Scenes without People". In: *Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 6193–6203.

[82] Kathleen M. Robinette, Sherri Blackwell, Hein Daanen, Mark Boehmer, Scott Fleming, Tina Brill, David Hoeferlin, and Dennis Burnsides. *Civilian American and European Surface Anthropometry Resource (CAESAR) Final Report*. Tech. rep. AFRL-HE-WP-TR-2002-0169. US Air Force Research Laboratory, 2002.

[83] *Vicon Vantage: Cutting edge, flagship camera with intelligent feedback and resolution*. https://www.vicon.com/hardware/cameras/vantage.

[84] *Stratasys Fortus 360mc: 3D printing*. https://www.stratasys.com/resources/search/white-papers/fortus-360mc-400mc.

[85] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. "SMPL: A Skinned Multi-Person Linear Model". In: *Transactions on Graphics (TOG)* 34.6 (2015), 248:1–248:16.

[86] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. "Learning a model of facial shape and expression from 4D scans". In: *ACM Transactions on Graphics (TOG)* 36.6 (2017), 194:1–194:17.

[87] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J Black. "Dynamic FAUST: Registering human bodies in motion". In: *Computer Vision and Pattern Recognition (CVPR)*. 2017.

[88] Daniel Cudeiro, Timo Bolkart, Cassidy Laidlaw, Anurag Ranjan, and Michael Black. "Capture, Learning, and Synthesis of 3D Speaking Styles". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[89] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. "Generating 3D faces using Convolutional Mesh Autoencoders". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

[90] Tero Karras. "Maximizing Parallelism in the Construction of BVHs, Octrees, and K-d Trees". In: *ACM SIGGRAPH/Eurographics Conference on High-Performance Graphics*. 2012.

[91] Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J. Black, and Otmar Hilliges. "ARCTIC: A Dataset for Dexterous Bimanual Hand-Object Manipulation". In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023.

[92] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. "Monocular Expressive Body Regression through Body-Driven Attention". In: *European Conference on Computer Vision (ECCV)*. 2020.

[93] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. "End-to-end Recovery of Human Shape and Pose". In: *Computer Vision and Pattern Recognition (CVPR)*. 2018.

[94] Anurag Ranjan, David T. Hoffmann, Dimitrios Tzionas, Siyu Tang, Javier Romero, and Michael J. Black. "Learning Multi-Human Optical Flow". In: *International Journal of Computer Vision (IJCV)* (2020).

[95] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. "Learning from Synthetic Humans". In: *Computer Vision and Pattern Recognition (CVPR)*. 2017.

[96] Tim Welschehold, Christian Dornhege, and Wolfram Burgard. "Learning manipulation actions from human demonstrations". In: *International Conference on Intelligent Robots and Systems (IROS)*. 2016.

[97] Feryal M. Behbahani, Guillem Singla—Buxarrais, and A. Aldo Faisal. "Haptic SLAM: An Ideal Observer Model for Bayesian Inference of Object Shape and Hand Pose from Contact Dynamics". In: *Haptics: Perception, Devices, Control, and Applications*. 2016.

[98] M. Oberweger, P. Wohlhart, and V. Lepetit. "Generalized Feedback Loop for Joint Hand-Object Pose Estimation". In: *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 42.8 (2020), pp. 1898–1912.

[99] Sören Pirk, Vojtech Krs, Kaimo Hu, Suren Deepak Rajasekaran, Hao Kang, Yusuke Yoshiyasu, Bedrich Benes, and Leonidas J. Guibas. "Understanding and exploiting object interaction landscapes". In: *Transactions on Graphics (TOG)* 36.3 (2017), 31:1–31:14.

[100] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. "6-DOF GraspNet: Variational Grasp Generation for Object Manipulation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019.

[101] Sergey Prokudin, Christoph Lassner, and Javier Romero. "Efficient Learning on Point Clouds with Basis Point Sets". In: *Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 4331–4340.

[102] De-An Huang, Minghuang Ma, Wei-Chiu Ma, and Kris M. Kitani. "How do we use our hands? Discovering a diverse set of common grasps". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 666–675.

[103] Yezhou Yang, Cornelia Fermüller, Yi Li, and Yiannis Aloimonos. "Grasp type revisited: A modern perspective on a classical feature for vision". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015), pp. 400–408.

[104] Yuzuko C. Nakamura, Daniel M. Troniak, Alberto Rodriguez, M. T. Mason, and Nancy S. Pollard. "The complexities of grasping in the wild". In: *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)* (2017), pp. 233–240.

[105] Guido Heumer, Heni Ben Amor, Matthias Weber, and Bernhard Jung. "Grasp Recognition with Uncalibrated Data Gloves - A Comparison of Classification Methods". In: *2007 IEEE Virtual Reality Conference* (2007), pp. 19–26.

[106] Oliver Glauser, Shihao Wu, Daniele Panozzo, Otmar Hilliges, and Olga Sorkine-Hornung. "Interactive hand pose estimation using a stretch-sensing soft glove". In: *ACM Transactions on Graphics (TOG)* 38 (2019), pp. 1–15.

[107] Steffen Puhlmann, Fabian Heinemann, Oliver Brock, and Marianne Maertens. "A compact representation of human single-object grasping". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2016), pp. 1954–1959.

[108] Samarth Brahmbhatt, Chengcheng Tang, Christopher D. Twigg, Charles C. Kemp, and James Hays. "ContactPose: A Dataset of Grasps with Object Contact and Hand Pose". In: *European Conference on Computer Vision (ECCV)*. Vol. 12358. 2020, pp. 361–378.

[109] George ElKoura and Karan Singh. "Handrix: animating the human hand". In: *Symposium on Computer Animation*. 2003.

[110] Iasonas Oikonomidis, Nikolaos Kyriazis, and Antonis A. Argyros. "Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints". In: *2011 International Conference on Computer Vision* (2011), pp. 2088–2095.

[111] Jun-Sik Kim and Jung Min Park. "Physics-based hand interaction with virtual objects". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)* (2015), pp. 3814–3819.

[112] Nancy S. Pollard and Victor B. Zordan. "Physically based grasping control from example". In: *Symposium on Computer Animation*. 2005.

[113] Dafni Antotsiou, Guillermo Garcia-Hernando, and Tae-Kyun Kim. "Task-Oriented Hand Motion Retargeting for Dexterous Manipulation Imitation". In: *ECCV Workshops*. 2018.

[114] Christoph W. Borst and Arun P. Indugula. "Realistic virtual grasping". In: *IEEE Proceedings. VR 2005. Virtual Reality, 2005.* (2005), pp. 91–98.

[115] Hans Rijpkema and Michael Girard. "Computer animation of knowledge-based human grasping". In: *Proceedings of the 18th annual conference on Computer graphics and interactive techniques* (1991).

[116] Tanner Schmidt, Katharina Hertkorn, Richard A. Newcombe, Zoltán-Csaba Márton, Michael Suppa, and Dieter Fox. "Depth-based tracking with physical constraints for robot manipulation". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)* (2015), pp. 119–126.

[117] S Brahmbhatt, A Handa, J Hays, and D Fox. "ContactGrasp: Functional Multi-finger Grasp Synthesis from Contact". In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2019.

[118] Paul G. Kry and Dinesh K. Pai. “Interaction capture and synthesis”. In: *ACM Trans. Graph.* 25 (2005), pp. 872–880.

[119] Y. Li, Jiaxin L. Fu, and Nancy S. Pollard. “Data-Driven Grasp Synthesis Using Shape Matching and Task-Based Pruning”. In: *IEEE Transactions on Visualization and Computer Graphics* 13 (2007), pp. 732–747.

[120] Srinath Sridhar, Franziska Mueller, Michael Zollhöfer, Dan Casas, Antti Oulasvirta, and Christian Theobalt. “Real-Time Joint Tracking of a Hand Manipulating an Object from RGB-D Input”. In: *ArXiv* abs/1610.04889 (2016).

[121] Jeffrey Mahler, Matthew Matl, Vishal Satish, Michael Danielczuk, Bill DeRose, Stephen McKinley, and Ken Goldberg. “Learning ambidextrous robot grasping policies”. In: *Science Robotics* 4 (2019).

[122] Wenping Zhao, Jianjie Zhang, Jianyuan Min, and Jinxiang Chai. “Robust realtime physics-based motion control for human grasping”. In: *ACM Transactions on Graphics (TOG)* 32 (2013), pp. 1–12.

[123] Jungwon Seo, Soonkyum Kim, and Vijay R. Kumar. “Planar, bimanual, whole-arm grasping”. In: *2012 IEEE International Conference on Robotics and Automation* (2012), pp. 3271–3277.

[124] Robert Krug, Dimitar Dimitrov, Krzysztof Andrzej Charusta, and Boyko Iliev. “On the efficient computation of independent contact regions for force closure grasps”. In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2010), pp. 586–591.

[125] Lerrel Pinto and Abhinav Kumar Gupta. “Supersizing self-supervision: Learning to grasp from 50K tries and 700 robot hours”. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)* (2015), pp. 3406–3413.

[126] Joseph Redmon and Anelia Angelova. “Real-time grasp detection using convolutional neural networks”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)* (2014), pp. 1316–1322.

[127] Jeffrey Mahler, Matthew Matl, Xinyu Liu, Albert Li, David V. Gealy, and Ken Goldberg. “Dex-Net 3.0: Computing Robust Vacuum Suction Grasp Targets in Point Clouds Using a New Analytic Model and Deep Learning”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)* (2017), pp. 1–8.

[128] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael J. Black, Krikamol Muandet, and Siyu Tang. “Grasping Field: Learning Implicit Representations for Human Grasps”. In: *2020 International Conference on 3D Vision (3DV)* (2020), pp. 333–344.

[129] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. “A Point Set Generation Network for 3D Object Reconstruction from a Single Image”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 2463–2471.

[130] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Joshua B. Tenenbaum. “MarrNet: 3D Shape Reconstruction via 2.5D Sketches”. In: *NIPS*. 2017.

[131] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, W. Liu, and Yu-Gang Jiang. “Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images”. In: *European Conference on Computer Vision*. 2018.

[132] Zhiqin Chen, Andrea Tagliasacchi, and Hao Zhang. “BSP-Net: Generating Compact Meshes via Binary Space Partitioning”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), pp. 42–51.

[133] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. “AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation”. In: *arXiv: Computer Vision and Pattern Recognition* (2018).

[134] Hiroharu Kato, Y. Ushiku, and Tatsuya Harada. “Neural 3D Mesh Renderer”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2017), pp. 3907–3916.

[135] Daniel Maturana and Sebastian A. Scherer. “VoxNet: A 3D Convolutional Neural Network for real-time object recognition”. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2015), pp. 922–928.

[136] Mateusz Michalkiewicz, Jhony Kaesemodel Pontes, Dominic Jack, Mahsa Baktash, and Anders P. Eriksson. “Deep Level Sets: Implicit Surface Representations for 3D Shape Inference”. In: *ArXiv* abs/1901.06802 (2019).

[137] Jeong Joon Park, Peter R. Florence, Julian Straub, Richard A. Newcombe, and S. Lovegrove. “DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), pp. 165–174.

[138] Lars M. Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. “Occupancy Networks: Learning 3D Reconstruction in Function Space”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 4455–4465.

[139] Zhiqin Chen and Hao Zhang. “Learning Implicit Fields for Generative Shape Modeling”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 5932–5941.

[140] James Steven Supančič, Grégory Rogez, Yi Yang, Jamie Shotton, and Deva Ramanan. “Depth-Based Hand Pose Estimation: Data, Methods, and Challenges”. In: *2015 IEEE International Conference on Computer Vision (ICCV)* (2015), pp. 1868–1876.

[141] Javier Romero, Hedvig Kjellström, and Danica Kragic. “Monocular real-time 3D articulated hand pose estimation”. In: *2009 9th IEEE-RAS International Conference on Humanoid Robots* (2009), pp. 87–92.

[142] Dimitrios Tzionas and Juergen Gall. “3D Object Reconstruction from Hand-Object Interactions”. In: *2015 IEEE International Conference on Computer Vision (ICCV)* (2015), pp. 729–737.

[143] Grégory Rogez, Maryam Khademi, James Steven Supančič, José M. M. Montiel, and Deva Ramanan. “3D Hand Pose Detection in Egocentric RGB-D Images”. In: *ECCV Workshops*. 2014.

[144] Aggeliki Tsoli and Antonis A. Argyros. “Joint 3D Tracking of a Deformable Object in Interaction with a Hand”. In: *European Conference on Computer Vision*. 2018.

[145] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. “Capturing Hands in Action Using Discriminative Salient Points and Physics Simulation”. In: *International Journal of Computer Vision* 118 (2015), pp. 172–193.

[146] He Wang, Sören Pirk, Ersin Yumer, Vladimir G. Kim, Ozan Sener, Srinath Sridhar, and Leonidas J. Guibas. “Learning a Generative Model for Multi-Step Human-Object Interactions from Videos”. In: *Computer Graphics Forum* 38 (2019).

[147] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F. Fouhey. “Understanding Human Hands in Contact at Internet Scale”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), pp. 9866–9875.

[148] Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros, and Abderrahmane Kheddar. “Hand-Object Contact Force Estimation from Markerless Visual Tracking”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (2018), pp. 2883–2896.

[149] Grégory Rogez, James Steven Supančič, and Deva Ramanan. “First-person pose recognition using egocentric workspaces”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015), pp. 4325–4333.

[150] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. “Leveraging Photometric Consistency over Time for Sparsely Supervised Hand-Object Reconstruction”. In: *CoRR* abs/2004.13449 (2020).

[151] Diederik P. Kingma and Max Welling. “Auto-encoding variational bayes”. In: *International Conference on Learning Representations (ICLR)*. 2014.

[152] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. “On the Continuity of Rotation Representations in Neural Networks”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 5745–5753.

[153] *Amazon Mechanical Turk*. https://www.mturk.com.

[154] Mihai Zanfir, Andrei Zanfir, Eduard Gabriel Bazavan, William T. Freeman, Rahul Sukthankar, and Cristian Sminchisescu. “THUNDR: Transformer-Based 3D Human Reconstruction With Markers”. In: *International Conference on Computer Vision (ICCV)*. 2021, pp. 12971–12980.

[155] Matthew Loper, Naureen Mahmood, and Michael J. Black. “MoSh: Motion and Shape Capture from Sparse Markers”. In: *Transactions on Graphics (TOG)* 33.6 (2014), 220:1–220:13.

[156] Julieta Martinez, Michael J. Black, and Javier Romero. "On Human Motion Prediction Using Recurrent Neural Networks". In: *Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 4674–4683.

[157] Anand Gopalakrishnan, Ankur Mali, Dan Kifer, Lee Giles, and Alexander G. Ororbia. "A Neural Temporal Model for Human Motion Prediction". In: *Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 12116–12125.

[158] Daniel Holden, Jun Saito, and Taku Komura. "A deep learning framework for character motion synthesis and editing". In: *Transactions on Graphics (TOG)* 35.4 (2016), 138:1–138:11.

[159] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. "Learning Trajectory Dependencies for Human Motion Prediction". In: *International Conference on Computer Vision (ICCV)*. 2019, pp. 9488–9496.

[160] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. "Action2Motion: Conditioned Generation of 3D Human Motions". In: *International Conference on Multimedia (MM)*. 2020, pp. 2021–2029.

[161] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. "A Simple yet Effective Baseline for 3D Human Pose Estimation". In: *International Conference on Computer Vision (ICCV)*. 2017, pp. 2659–2668.

[162] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. "Recurrent Network Models for Human Dynamics". In: *International Conference on Computer Vision (ICCV)*. 2015, pp. 4346–4354.

[163] Daniel Holden, Taku Komura, and Jun Saito. "Phase-functioned neural networks for character control". In: *Transactions on Graphics (TOG)* 36.4 (2017), pp. 1–13.

[164] Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. "AI Choreographer: Music Conditioned 3D Dance Generation With AIST++". In: *International Conference on Computer Vision (ICCV)*. 2021, pp. 13401–13412.

[165] Yongyi Tang, Lin Ma, Wei Liu, and Wei-Shi Zheng. "Long-Term Human Motion Prediction by Modeling Motion Context and Enhancing Motion Dynamics". In: *International Joint Conference on Artificial Intelligence (IJCAI)*. 2018, pp. 935–941.

[166] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. "Attention is All you Need". In: *Conference on Neural Information Processing Systems (NeurIPS)*. 2017, pp. 5998–6008.

[167] Guillermo Garcia-Hernando, Edward Johns, and Tae-Kyun Kim. "Physics-Based Dexterous Manipulations with Estimated Hand Poses and Residual Reinforcement Learning". In: *International Conference on Intelligent Robots and Systems (IROS)*. 2020, pp. 9561–9568.

[168] Tao Chen, Jie Xu, and Pulkit Agrawal. "A System for General In-Hand Object Re-Orientation". In: *Conference on Robot Learning (CoRL)* (2021).

[169] Rami Ali Al-Asqhar, Taku Komura, and Myung Geol Choi. “Relationship descriptors for interactive motion adaptation”. In: *Symposium on Computer Animation (SCA)*. 2013, pp. 45–53.

[170] Edmond S. L. Ho, Taku Komura, and Chiew-Lan Tai. “Spatial relationship preserving character motion adaptation”. In: *Transactions on Graphics (TOG)* 29.4 (2010), 33:1–33:8.

[171] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael J. Black. “Stochastic Scene-Aware Motion Prediction”. In: *International Conference on Computer Vision (ICCV)*. 2021, pp. 11374–11384.

[172] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. “Synthesizing Long-Term 3D Human Motion and Interaction in 3D Scenes”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 9401–9411.

[173] Juncong Lin, Takeo Igarashi, Jun Mitani, Minghong Liao, and Ying He. “A sketching interface for sitting pose design in the virtual environment”. In: *Transactions on Visualization and Computer Graphics (TVCG)* 18.11 (2012), pp. 1979–1991.

[174] Changgu Kang and Sung-Hee Lee. “Environment-adaptive contact poses for virtual characters”. In: *Computer Graphics Forum (CGF)* 33.7 (2014), pp. 1–10.

[175] Kurt Leimer, Andreas Winkler, Stefan Ohrhallinger, and Przemyslaw Musialski. “Pose to Seat: Automated design of body-supporting surfaces”. In: *Computer Aided Geometric Design (CAGD)* 79 (2020).

[176] Youyi Zheng, Han Liu, Julie Dorsey, and Niloy J. Mitra. “Ergonomics-inspired reshaping and exploration of collections of models”. In: *Transactions on Visualization and Computer Graphics (TVCG)* 22.6 (2015), pp. 1732–1744.

[177] Helmut Grabner, Juergen Gall, and Luc Van Gool. “What makes a chair a chair?” In: *Computer Vision and Pattern Recognition (CVPR)*. 2011, pp. 1529–1536.

[178] Siwei Zhang, Yan Zhang, Qianli Ma, Michael J. Black, and Siyu Tang. “PLACE: Proximity Learning of Articulation and Contact in 3D Environments”. In: *International Conference on 3D Vision (3DV)*. 2020, pp. 642–651.

[179] Mohamed Hassan, Partha Ghosh, Joachim Tesch, Dimitrios Tzionas, and Michael J. Black. “Populating 3D Scenes by Learning Human-Scene Interaction”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 14708–14718.

[180] Josh Merel, Saran Tunyasuvunakool, Arun Ahuja, Yuval Tassa, Leonard Hasenclever, Vu Pham, Tom Erez, Greg Wayne, and Nicolas Heess. “Catch & Carry: Reusable neural controllers for vision-guided whole-body tasks”. In: *Transactions on Graphics (TOG)* 39.4 (2020), p. 39.

[181] Enric Corona, Albert Pumarola, Guillem Alenyà, Francesc Moreno-Noguer, and Gregory Rogez. “GanHand: Predicting Human Grasp Affordances in Multi-Object Scenes”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 5030–5040.

[182] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. "HuMoR: 3D Human Motion Model for Robust Pose Estimation". In: *International Conference on Computer Vision (ICCV)*. 2021, pp. 11488–11499.

[183] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations (ICLR)*. 2015.

[184] Berk Çalli, Arjun Singh, Aaron Walsman, Siddhartha S. Srinivasa, Pieter Abbeel, and Aaron M. Dollar. "The YCB object and Model set: Towards common benchmarks for manipulation research". In: *International Conference on Advanced Robotics (ICAR)*. 2015, pp. 510–517.

[185] Yinghao Huang, Omid Taheri, Michael J. Black, and Dimitrios Tzionas. "InterCap: Joint Markerless 3D Tracking of Humans and Objects in Interaction". In: *German Conference on Pattern Recognition (GCPR)*. Vol. 13485. Lecture Notes in Computer Science. Springer. 2022, pp. 281–299.

[186] Keyang Zhou, Bharat Lal Bhatnagar, Jan Eric Lenssen, and Gerard Pons-Moll. "TOCH: Spatio-Temporal Object Correspondence to Hand for Motion Refinement". In: *European Conference on Computer Vision (ECCV)*. 2022.

[187] He Zhang, Yuting Ye, Takaaki Shiratori, and Taku Komura. "ManipNet: Neural manipulation synthesis with a hand-object spatial representation". In: *Transactions on Graphics (TOG)* 40.4 (2021), 121:1–121:14.

[188] Omid Taheri, Vasileios Choutas, Michael J. Black, and Dimitrios Tzionas. "GOAL: Generating 4D Whole-Body Motion for Hand-Object Grasping". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022. URL: https://goal.is.tue.mpg.de.

[189] Philipp Kratzer, Simon Bihlmaier, Niteesh Balachandra Midlagajni, Rohit Prakash, Marc Toussaint, and Jim Mainprice. "MoGaze: A Dataset of Full-Body Motions that Includes Workspace Geometry and Eye-Gaze". In: *IEEE Robotics and Automation Letters* 6.2 (2021), pp. 367–373.

[190] Sahar El-Khoury, Anis Sahbani, and Philippe Bidaud. "3D Objects Grasps Synthesis: A Survey". In: *IFToMM World Congress on Mechanism and Machine Science*. 2011.

[191] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. "Data-Driven Grasp Synthesis—A Survey". In: *IEEE Transactions on Robotics* 30 (2014), pp. 289–309.

[192] Ying Li, Jiaxin L. Fu, and Nancy S. Pollard. "Data-Driven Grasp Synthesis Using Shape Matching and Task-Based Pruning". In: *Transactions on Visualization and Computer Graphics (TVCG)* 13.4 (2007), pp. 732–747.

[193] Igor Mordatch, Zoran Popovic, and Emanuel Todorov. "Contact-Invariant Optimization for Hand Manipulation". In: *Symposium on Computer Animation (SCA)*. 2012, pp. 137–144.

[194] Beatriz León, Stefan Ulbrich, Rosen Diankov, Gustavo Puche, Markus Przybylski, Antonio Morales, Tamim Asfour, Sami Moisio, Jeannette Bohg, and James Kuffner. "OpenGRASP: A Toolkit for Robot Grasping Simulation". In: *Simulation, Modeling, and Programming for Autonomous Robots - Second International Conference, SIMPAR 2010, Darmstadt, Germany, November 15-18, 2010. Proceedings*. Vol. 6472. Lecture Notes in Computer Science. Springer, 2010, pp. 109–120.

[195] Hanwen Jiang, Shaowei Liu, Jiashun Wang, and Xiaolong Wang. "Hand-Object Contact Consistency Reasoning for Human Grasps Generation". In: *Proceedings of the International Conference on Computer Vision*. 2021.

[196] Korrawe Karunratanakul, Adrian Spurr, Zicong Fan, Otmar Hilliges, and Siyu Tang. "A Skeleton-Driven Neural Occupancy Representation for Articulated Hands". In: *2021 International Conference on 3D Vision (3DV)*. 2021, pp. 11–21.

[197] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael J. Black, Krikamol Muandet, and Siyu Tang. "Grasping Field: Learning Implicit Representations for Human Grasps". In: *International Conference on 3D Vision (3DV)*. 2020, pp. 333–344.

[198] Tianqiang Zhu, Rina Wu, Xiangbo Lin, and Yi Sun. "Toward Human-Like Grasp: Dexterous Grasping via Semantic Representation of Object-Hand". In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 15721–15731.

[199] Karen C. Liu. "Dextrous manipulation from a grasping pose". In: *Transactions on Graphics (TOG)* 28.3 (2009), p. 59.

[200] Yuting Ye and Karen C. Liu. "Synthesis of detailed hand manipulations using contact sampling". In: *Transactions on Graphics (TOG)* 31.4 (2012), 41:1–41:10.

[201] Wenping Zhao, Jianjie Zhang, Jianyuan Min, and Jinxiang Chai. "Robust realtime physics-based motion control for human grasping". In: *Transactions on Graphics (TOG)* 32.6 (2013), 207:1–207:12.

[202] Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Józefowicz, Bob McGrew, Jakub W. Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, Jonas Schneider, Szymon Sidor, Josh Tobin, Peter Welinder, Lilian Weng, and Wojciech Zaremba. "Learning dexterous in-hand manipulation". In: *Int. J. Robotics Res.* 39.1 (2020).

[203] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. "Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations". In: *Proceedings of Robotics: Science and Systems*. Pittsburgh, Pennsylvania, 2018.

[204] Kevin Bergamin, Simon Clavet, Daniel Holden, and James Richard Forbes. "DReCon: Data-Driven Responsive Control of Physics-Based Characters". In: *ACM Trans. Graph.* 38.6 (2019).

[205] Soohwan Park, Hoseok Ryu, Seyoung Lee, Sunmin Lee, and Jehee Lee. "Learning Predict-and-Simulate Policies from Unorganized Human Motion Data". In: *ACM Trans. Graph.* 38.6 (2019).

[206] Xue Bin Peng, Glen Berseth, Kangkang Yin, and Michiel Van De Panne. "DeepLoco: Dynamic Locomotion Skills Using Hierarchical Deep Reinforcement Learning". In: *ACM Trans. Graph.* 36.4 (2017).

[207] Sammy Christen, Muhammed Kocabas, Emre Aksan, Jemin Hwangbo, Jie Song, and Otmar Hilliges. "D-Grasp: Physically Plausible Dynamic Grasp Synthesis for Hand-Object Interactions". In: *Computer Vision and Pattern Recognition (CVPR)*. 2022.

[208] Sören Pirk, Olga Diamanti, Boris Thibert, Danfei Xu, and Leonidas J. Guibas. "Shape-aware Spatio-temporal Descriptors for Interaction Classification". In: *ICIP* (2017).

[209] He Wang, Sören Pirk, Ersin Yumer, Vladimir G. Kim, Ozan Sener, Srinath Sridhar, and Leonidas J. Guibas. "Learning a Generative Model for Multi-Step Human-Object Interactions from Videos". In: *Computer Graphics Forum* 38.2 (2019), pp. 367–378.

[210] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. "HOnnotate: A Method for 3D Annotation of Hand and Object Poses". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 3193–3203.

[211] Bharat Lal Bhatnagar, Xianghui Xie, Ilya A Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. "Behave: Dataset and method for tracking human object interactions". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 15935–15946.