Contact force sensing from motion tracking

Tu-Hoa Pham

Collège Doctoral du Languedoc-Roussillon

THÈSE

Pour obtenir le grade de Docteur

Délivré par l'**Université de Montpellier**

Préparée au sein de l'école doctorale
Information Structures Systèmes (I2S)

Et de l'unité de recherche
Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier

Spécialité
Systèmes Automatiques et Microélectroniques

Présentée par **Tu-Hoa Pham**

Contact Force Sensing From Motion Tracking

Soutenue le 9 décembre 2016 devant le jury composé de

M. Éric Marchand	Professeur	Université de Rennes 1	Rapporteur
Mme. Véronique Perdereau	Professeur	Université Paris 6	Rapporteur
M. Antonis A. Argyros	Professeur	University of Crete	Examinateur
M. William Puech	Professeur	Université de Montpellier	Examinateur
M. Grégory Rogez	Chercheur	Inria Rhône-Alpes	Examinateur
M. Abderrahmane Kheddar	Directeur de Recherche	CNRS-UM LIRMM	Directeur de thèse

Université Montpellier

Acknowledgements

I would like to thank my research advisor, Prof. Abderrahmane Kheddar, for welcoming me in his team at JRL in 2011, while I was still a master student and very unsure about what I wanted to accomplish in life. I feel immensely grateful for the advice and support I received during these past three years of Ph.D. spent between Montpellier and Tsukuba, and for all the opportunities I received to improve both my research and myself.

It is a great honor for me to have Prof. Éric Marchand and Prof. Véronique Perdereau review this dissertation, and to have it examined by Prof. Antonis A. Argyros, Prof. William Puech, and Dr. Grégory Rogez.

I am doubly indebted to Antonis, for welcoming me in his lab in Crete prior to starting this Ph.D., and for having been a major mentor since then.

I would like to thank AIST and Prof. Eiichi Yoshida for hosting me in Japan and making JRL such a great environment to work and exchange ideas.

I would like to thank my colleagues as well as the several people who helped me broaden my research horizon through fruitful conversations: Don Joven Agravante, Hervé Audren, Stanislas Brossette, Stéphane Caron, Benjamin Chrétien, Giovanni De Magistris, Pierre Gergondet, Adrien Pajon, Antonio Paolillo, Damien Petit, Joris Vaillant, and many others.

I am thankful to my family for believing in me all this time, as well as my friends Arthur Dartois, Daniel Jartoux, Joan Massot, and Henri Ronse. I acknowledge Daniel as the source of all this trouble, by sending me that robotics internship offer five years ago.

I would like to thank my former professors at SUPAERO, Prof. Caroline Bérard and Prof. Yves Gourinat, for trusting me to do something good with my life even when it was hard to see things that way.

Finally, I dedicate this thesis to my amazing, beautiful, unconditionally loving girlfriend Jessica, who sacrificed so much of our time together to support me throughout this thesis.

Abstract

The human sense of touch is of fundamental importance in the way we perceive our environment, move ourselves, and purposefully interact with other objects or beings. Thus, contact forces are informative on both the realized task and the underlying intent. However, monitoring them with force transducers is a costly, cumbersome and intrusive process. In this thesis, we investigate the capture of haptic information from motion tracking. This is a challenging problem, as a given motion can generally be caused by an infinity of possible force distributions in multi-contact. In such scenarios, physics-based optimization alone may only capture force distributions that are physically compatible with a given motion, rather than those really applied. In contrast, machine learning techniques for the black-box modelling of kinematically and dynamically complex structures are often prone to generalization issues. We propose a formulation of the force distribution problem utilizing both approaches jointly rather than separately. We thus capture the variability in the way humans instinctively regulate contact forces while also ensuring their compatibility with the observed motion. We present our approach on both manipulation and whole-body interaction with the environment. We consistently back our findings with ground-truth measurements and provide extensive datasets to encourage and serve as benchmarks for future research on this new topic.

Keywords: force sensing from vision; motion capture; humanoid robotics.

Résumé

Le sens du toucher joue un rôle fondamental dans la façon dont nous percevons notre environnement, nous déplaçons, et interagissons délibérément avec d'autres objets ou êtres vivants. Ainsi, les forces de contact informent à la fois sur l'action réalisée et sa motivation. Néanmoins, l'utilisation de capteurs de force traditionnels est coûteuse, lourde, et intrusive. Dans cette thèse, nous examinons la perception haptique par la capture de mouvement. Ce problème est difficile du fait qu'un mouvement donné peut généralement être causé par une infinité de distributions de forces possibles, en multi-contact. Dans ce type de situations, l'optimisation sous contraintes physiques seule ne permet que de calculer des distributions de forces plausibles, plutôt que fidèles à celles appliquées en réalité. D'un autre côté, les méthodes d'apprentissage de type 'boîte noire' pour la modélisation de structures cinématiquement et dynamiquement complexes sont sujettes à des limitations en termes de capacité de généralisation. Nous proposons une formulation du problème de la distribution de forces exploitant ces deux approches ensemble plutôt que séparément. Nous capturons ainsi la variabilité dans la façon dont on contrôle instinctivement les forces de contact tout en nous assurant de leur compatibilité avec le mouvement observé. Nous présentons notre approche à la fois pour la manipulation et les interactions corps complet avec l'environnement. Nous validons systématiquement nos résultats avec des mesures de référence et fournissons des données exhausives pour encourager et évaluer les travaux futurs sur ce nouveau sujet.

Mots-clés: capture de force par vision; capture de mouvement; robotique humanoïde.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms

AHRS	Attitude and heading reference system
ANN	Artificial neural network
BSIP	Body segment inertial parameters
CNN	Convolutional neural network
CNS	Central nervous system
CoM	Center of mass
CPU	Central processing unit
DoF	Degree of freedom
ECT	Ensemble of collaborative trackers
fps	Frames per second
FSR	Force-sensing resistors
GPGPU	General-purpose computing on graphics processing units
GPU	Graphics processing unit
GRF	Ground reaction force
ICP	Iterative closest point
IMU	Inertial measurement unit
LfD	Learning from demonstration

LSTM	Long short-term memory
MAP	Muscle activation patterns
PSO	Particle swarm optimization
QP	Quadratic programming
RDF	Random decision forest
RGB-D	Red, blue, green (color) and depth
RNN	Recurrent neural network
SDF	Signed distance function
SOCP	Second-order cone program
SVM	Support vector machine

Hand-Object Manipulation

$\left(\mathbf{n}_k, \mathbf{t}_k^x, \mathbf{t}_k^y\right)$	Local contact space (normal-tangential) at contact k
(f_k, g_k, h_k)	Local force decomposition at contact k along $\left(\mathbf{n}_k, \mathbf{t}_k^x, \mathbf{t}_k^y\right)$
$\mathbf{G}$	Center of mass of the manipulated object
$\mathbf{F}_k$	Contact force applied at contact k
$\mathbf{F}_k^{(n)}, \mathbf{F}_k^{(i)}$	Nominal and internal components of contact force $\mathbf{F}_k$
$\mathbf{P}_k$	3D position of contact point k
$\mathscr{F}$	Finger set: thumb, index, middle, ring, pinky
$\mathbf{J}$	Inertia matrix in object frame, taken at the center of mass
m	Object mass
$\boldsymbol{\mathscr{F}}, \boldsymbol{\tau}$	Net force and torque exerted on the manipulated object
$\boldsymbol{\theta}$	Hand or object pose
$\mathbf{q} = (q_x, q_y, q_z, q_w)$	Quaternion of imaginary part (q_x, q_y, q_z) and real part q_w
$\boldsymbol{\omega}, \boldsymbol{\alpha}$	Rotational velocity and acceleration

$\mathbf{v}, \mathbf{a}$	Linear velocity and acceleration of the center of mass

Operators

$\frac{d^{(N)}.}{dt^{(N)}}$	N-th order time differentiation operator
$\mathbf{M}^{-1}$	Inverse of $\mathbf{M}$
$\mathbf{M}^{-T}$	Transpose of $\mathbf{M}$
$\mathbf{u} \cdot \mathbf{v}$	Dot product between $\mathbf{u}$ and $\mathbf{v}$
$\mathbf{u} \times \mathbf{v}$	Cross product between $\mathbf{u}$ and $\mathbf{v}$
$\overrightarrow{\mathbf{AB}}$	Vector from $\mathbf{A}$ to $\mathbf{B}$
$x^{(N)}$	N-th order time derivative of signal x

Whole-Body Interactions

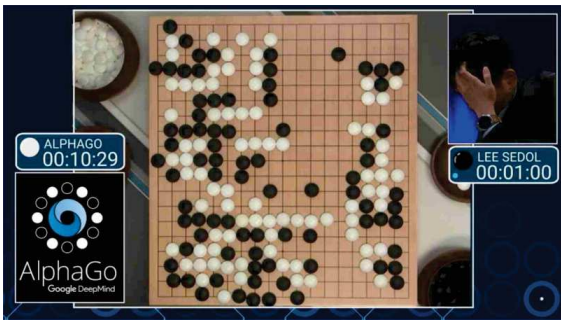
$\mathbf{R}_s$	Orientation matrix of body segment s in the global frame
$\boldsymbol{\omega}_s$	Angular velocity of body segment s in the body frame
$\dot{\mathscr{L}}_{\mathbf{G}}$	Angular momentum about $\mathbf{G}$
$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$	Bias vector including Coriolis, centrifugal forces and gravity terms
$\mathbf{G}$	Centroid of the articulated system
$\mathbf{G}_s$	Center of mass of body segment s
${}^{\mathbf{G}}\mathbf{F}_k$	Contact wrench at contact k transformed to $\mathbf{G}$
$\mathscr{G}$	Reference frame attached to the centroid and fixed with respect to a chosen body segment (e.g., pelvis)
${}^{\mathscr{G}}\mathbf{P}_k$	3D position of contact point k expressed in $\mathscr{G}$
${}^{\mathscr{G}}\mathbf{w}^{(gi)}$	gravito-inertial wrench $\mathbf{w}^{(gi)}$ expressed in $\mathscr{G}$
$\mathbf{P}_k$	3D position of contact point k
$\mathscr{C}_k = (\mathbf{x}_k, \mathbf{y}_k, \mathbf{z}_k)$	Local coordinate system at contact k
$\delta_{k,i}$	Activity flag for contact k at time step i

$\mathbf{w}^{(gi)}$	Gravito-inertial wrench taken at the centroid
$\mathbf{H}(\mathbf{q})$	Mass matrix
$\mathbf{J}_k$	Jacobian matrix at contact k
$\boldsymbol{\tau}^{(i)}$	Internal joint torques
$\mathscr{P}$	Linear momentum
$\mathbf{I}_s$	Inertia matrix of body segment s in the body frame, taken at $\mathbf{G}_s$
$\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$	Generalized coordinates, velocities and accelerations of the articulated system
$\mathbf{v}_s$	Linear velocity of $\mathbf{G}_s$ in the global frame
$\mathbf{F}_k = (\boldsymbol{\tau}_k, \mathbf{f}_k)$	External wrench at contact k, with $\boldsymbol{\tau}_k$ and $\mathbf{f}_k$ torque and force, respectively
${}^{\mathscr{C}_k}\mathbf{F}_k$	External wrench $\mathbf{F}_k$ expressed in contact space $\mathscr{C}_k$
m_s	Mass of body segment s

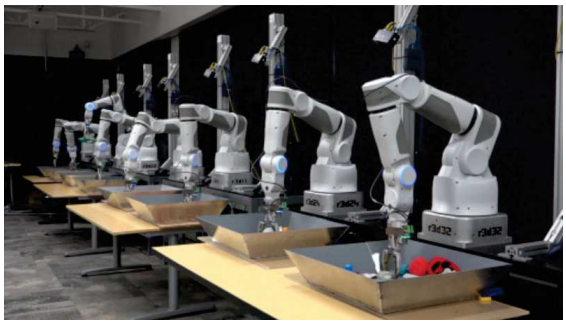
Introduction

The recent years have let us witness impressive advances in the fields of robotics and artificial intelligence. In October 2015, Google DeepMind's AlphaGo became the first computer program to defeat world-class champion Lee Sedol at go [SHM+16], a game long considered extremely difficult for computers to win due to its large branching factor. New techniques for large-scale machine learning are enabling robots to develop grasping skills without human supervision [LPKQ16], and progress in actuation and balancing systems are allowing biped robots to withstand human aggression with a stick. In parallel, augmented and virtual reality technologies are getting more and more immersive and affordable to customers. Extrapolating from here, one could imagine a future where self-driving cars take over the world by running over humans too absorbed in mixed reality activities to look around when crossing the street. Fortunately, this scenario remains (rather bad) science fiction. The winning AlphaGo system garnered hundreds of CPUs and GPUs in parallel for a very specialized purpose, illustrating the fact that it remains a major challenge to construct a truly general artificial intelligence. Robots still lack sensing abilities for both themselves and the environment, e.g., tactile sensors enabling the perception of subtle haptic clues for dexterous manipulation, or robust object identification and tracking from vision without relying on fiducial markers. Stimulating the human sense of touch also remains a challenge for virtual reality systems to achieve further immersion and embodiment.

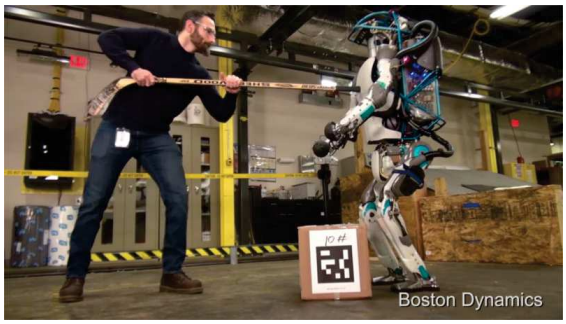
Overall, a central theme towards action understanding and control is that of haptic perception. Indeed, a privileged way humans interact with their environment is through touch, i.e., the application of contact forces to move objects, themselves and others. Being able to capture these by means of a simple and affordable setup would open a wide range of possibilities for multiple fields of research and engineering. For instance, in robotics, this could enable intuitive interfaces for learning from demonstration and human-robot interaction. In neuroscience, being able to estimate the forces applied during manipulation and reproducing them by means of haptic feedback could benefit the sensation of virtual or robotic embodiment for virtual reality and teleoperation. In rehabilitation, monitoring forces applied during manipulation and locomotion could help detect musculoskeletal conditions



(a) AlphaGo playing against Lee Sedol [SHM+16].



(b) Robot arms learning grasping [LPKQ16].



(c) Boston Dynamics Atlas robot pushed with a stick.

(d) Oculus Touch marker-based tracking system.

Figure 1: A few recent advances in artificial intelligence, robotics and virtual reality.

and guide movement training. In effect, force models are already being introduced in computer vision, for action understanding and robust motion capture based on physics.

Overall, haptic perception has been shown to be crucial in the human ability to grasp and manipulate objects [JW84]. However, traditional force sensing technologies are often costly, cumbersome, as well as of limited accuracy under repeated use. Importantly, they are also *intrusive*, in that mounting them onto objects can noticeably impact their physical properties. Alternatively, using wearable sensors (e.g., placed at the fingertips) hinders the natural perception of crucial properties (e.g., friction) by the human sense of touch. In both cases, instrumenting either is a time-consuming process that induces further limitations due to both the whole equipment needed (e.g., power supply, cabling) and its effect on the manipulation task (e.g., limitations on the natural range of motion). Computer vision research has resulted in multiple successful methods for monitoring motion information from markerless observations. Thus, a challenging question is: can we also estimate contact forces from vision? If successful, such a method would enable applications such as those discussed before, while also tremendously benefiting their usability. The completely non-intrusive estimation of contact forces would be particularly useful for monitoring and learning of daily activities in the context of home and service robotics.

However, this is an extremely challenging problem, for multiple reasons. First, before even considering forces, the markerless tracking of human subjects (whole-body or restricted to the hands) is itself an active topic in computer vision research. Indeed, such problems typically involve a large number of degrees of freedom, while visual observations may be subject to strong occlusions. In addition, the tracking method must account for, or be robust to, sensing limitations (e.g., number of cameras, resolution) and uncertainties (e.g., depth sensing noise). At this stage, it is unclear whether the state of the art in markerless visual tracking permits the accurate estimation of kinematic information (e.g., velocities and accelerations) on unconstrained interactions (e.g., rapid motions). Second, estimating when and where contact occurs between the human subject and the environment is not trivial due to mutual occlusions inherent to contact situations. Third, even when the motion and its characteristics are perfectly known, the force distribution problem is indeterminate in multi-contact. Indeed, while the knowledge of a force distribution completely characterizes the resulting kinematics, the converse is generally not true. Instead, in multi-contact, a desired net force can generally be distributed in an infinity of different configurations on a given set of contact points.

Still, spatiotemporal relationship between actor and objects can help enhance the joint understanding of both. In this thesis, we leverage the state of the art in computer vision, robotics and machine learning to investigate the problem of force sensing from vision (FSV). Typically, biomechanics approaches aim at solving the force distribution problem through inverse optimization, i.e., by searching for the criteria supposedly optimized by the central nervous system. While theoretically sound, such approaches have so far mainly produced models limited to very specific scenarios and grasping conditions (e.g., holding an object still between two fingers), due to the great complexity of the human body and the limited observability of physiological parameters without invasive surgery. Conversely, force models for physics simulation have recently been employed as optimization priors in hand-object markerless tracking. While helpful for this purpose, these approaches only aim at computing force distributions that are physically plausible, rather than the actual forces being applied.

In contrast, throughout this thesis, it is a primary concern for us to demonstrate the validity of the FSV framework we propose by consistently comparing the forces distributions estimated by our method with ground-truth measurements acquired experimentally on real tasks. We consider the force distribution problem both for hand-object manipulation and whole-body interaction with the environment. The core of our approach lies in the utilization of physics-based optimization and machine learning jointly, rather than separately. On its own, the former may be limited to physical plausibility, rather than fidelity to real force distribution patterns. The latter, by itself, is often subject to generalization issues, e.g., when

it is applied to examples that differ significantly from those present in the training dataset. By using them together, we capture the variability in the way humans naturally apply contact forces when interacting with objects and their environment, while ensuring that the resulting force distributions are compatible with the observed motion. The organization of this thesis is as follows:

- In Chapter 1, we review the state of the art in motion and force sensing technologies as well as visual tracking techniques and contact dynamics modeling.
- In Chapter 2, we introduce the topic of force sensing from vision in the context of manipulation. First, we evaluate the performance of a state-of-the-art hand-object tracker regarding motion and kinematics estimation. We then formulate an optimization problem estimating the minimal forces required to achieve the observed motion, and complement it with artificial neural networks that predict the additional forces humans naturally apply to secure the object in the grasp.
- In Chapter 3, we extend our work to 3D (normal and tangential) manipulation forces while accounting for time continuity and variability across object and grasp configurations with a new pipeline combining physics-based optimization and recurrent neural networks in mutual interaction. To assess the performance and extensibility of our approach, we also construct and release the first large-scale dataset on human manipulation kinodynamics with high-precision motion and force sensors.
- In Chapter 4, we challenge the estimation of whole-body contact forces in interaction with the environment from motion capture. We show that in such situations, forces estimated from optimization only differ significantly from the forces applied in reality, while conversely, force sensor measurements can be rather unreliable. We collect a new dataset on whole-body kinodynamics using an inertial motion capture system and external force sensors, and show that our approach can be successfully extended to challenging multi-contact configurations.

Chapter 1

Literature Review

The observation of humans in interaction with their environment is of great interest for multiple fields of research, such as robotics, computer vision, graphics and rehabilitation. Such interactions can be observed and encoded in terms of motion, i.e., the relative poses through time between one or multiple human actors and surrounding objects. From a lower-level perspective, humans interact with their environment in a privileged way through touch, i.e., the application of contact forces and torques onto surrounding objects, permitted by the remarkable dexterity of the human hand. As such, the monitoring of interaction forces is informative of both the resulting motions and the human intent. We first review the motion and force sensing technologies and their applications (Section 1.1). We discuss the state in the art in visual tracking (Section 1.2). We review existing kinematic and dynamic models for human motion analysis (Section 1.4). Finally, we discuss numerical techniques used for kinodynamic modeling (Section 1.5).

1.1 Monitoring Human Interactions With The Environment

In this section, we discuss the types of sensors used for motion and force monitoring and illustrate their combined utilization in past works.

1.1.1 Motion Sensors

Motion capture aims at tracking the movement of target objects or subjects. It is today commonly used to animate virtual characters or avatars in computer graphics and virtual reality. The tracking target can be rigid (e.g., a mug), articulated (e.g., whole-body tracking), or deformable (e.g., facial motion capture). The motion can be monitored in terms of positions (linear, angular) or the subsequent derivatives (e.g., velocity, acceleration).

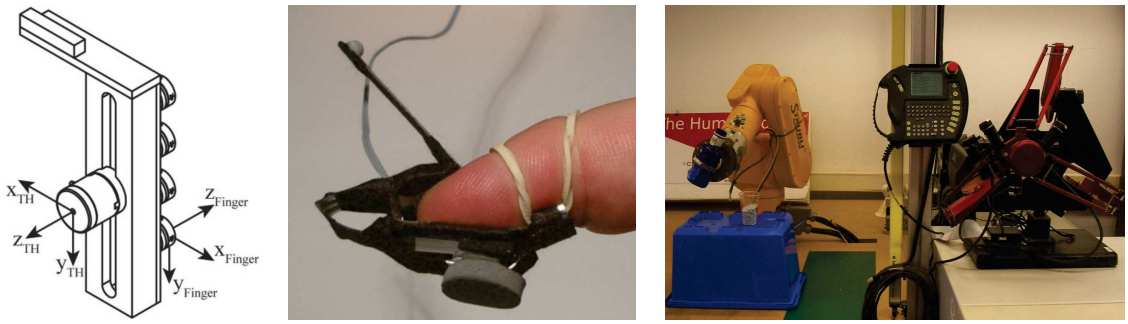
Motion capture techniques based on vision can be placed into two categories: marker-based or markerless. In the former, markers placed at prespecified landmarks on the tracking target are located in space from the visual observations (e.g., by thresholding), and mapped to the target kinematic structure [MG01, MHK06]. While marker-based methods generally yield very precise tracking, the required instrumentation is cumbersome. Markerless approaches aim at alleviating this issue by relying on visual observations only and were reviewed in [Pop07] in the context of human motion analysis. A problem inherent to vision-based systems is the possibility of mutual occlusions between multiple tracking targets (e.g., an object hiding the hand manipulating it, and conversely) or self-occlusions between different parts of the same target (e.g., the palm hiding some fingers). This issue can be alleviated by using multi-camera systems to limit the amount of occlusions. Such systems also allow the obtention of depth information by 3D reconstruction from multiple views [CBK03].

However, multi-camera systems are not very portable and require extensive calibration, making them difficult to use in uncontrolled environments. Inspired by human binocular vision, stereo vision systems use two calibrated cameras with parallel optical axes to seek corresponding points between view pairs and extract depth information [LK81]. Alternative methods used in consumer-grade RGB-D (color and depth) sensors include structured light [SS03], which consists in projecting a known pattern onto the scene and analyzing the corresponding deformations (e.g., Microsoft Kinect, Asus Xtion), and time-of-flight [GYB04], based on the delay between the emittance of a light pulse and its reflection by the objects of the scene (e.g., Microsoft Kinect v2, SoftKinetic DepthSense DS325).

Finally, strong occlusions can occur even when using multiple cameras, for example when multiple tracking targets interract with each other; in tight spaces; or in uncontrolled environments (e.g., outdoor). In such scenarios, body joint angles can be tracked using non optical systems worn on the subject's body. Inertial motion capture systems rely on inertial measurement units (IMUs) placed at specified landmarks on the subject's body (e.g., Perception Neuron, Xsens MVN Awinda). The joint angles are computed by sensor fusion and integration of the IMUs' accelerometer and gyroscope measurements [RLS09]. As such, inertial motion capture systems are prone to positional drift. External camera systems can be used in combination with inertial motion capture to provide absolute positioning.

1.1.2 Force Sensors

Force sensing is a key objective in understanding physical interactions between humans and their environment. Not only can tactile feedback provide valuable insight when setting up haptic interfaces, it is also of vital importance when monitoring manipulation tasks performed by robots, e.g., frail object grasping. Force sensors now come up in a variety of types and



(a) Handle instrumented with force sensors [WTZL15]

(b) Force transducer mounted on fingertip for interaction capture and synthesis [KP06].

(c) Haptic interface to demonstrate a robotic arm how to pour drinks based on force measurements during teleoperation [RJC13].

Figure 1.1: Instrumentation examples. (a): on object, (b): on hand, (c): haptic interface.

specifications (e.g., capacitive, piezoresistive, single or multi-axis) [CHP08], with particular requirements and applications in robotics [DMVS10, KCP15]. Common drawbacks of mechatronic force sensing devices reside in their extensive need for calibration, accuracy and repeatability limitations (e.g., hysteresis) and cost. Besides their sensing capabilities, another difficulty lies in their *intrusiveness*. For example, consider a human subject manipulating an object of given physical properties. In order to measure manipulation forces, sensors must be placed at the contact locations between the object and the hand. If mounted onto the object:

- the contact locations must be chosen in advance
- the object must be modified to fit force transducers at the specified locations as well as any additional instrumentation required (e.g., wires)
- such modifications can be cumbersome, time-consuming, and importantly affect the physical properties of the object (e.g., shape, mass distribution)

If mounted onto the hand:

- contacts can be placed arbitrarily but additional hand tracking is required to know their position throughout the experiment
- the force sensors can significantly impair the human haptic sense (e.g., friction perception) and limit the natural range of motion (e.g., joint angles, contact orientations)

We depict such instrumentation examples in Fig. 1.1.

A less intrusive approach could consist in covering the surface of the object with lightweight, flexible tactile sensing surfaces, reviewed in [SCCP14]. Doing so would allow the monitoring of manipulation forces on arbitrary contact points with minimal impact on

the object and the hand. However, such technologies are still limited in terms of captured dimensions (e.g., normal forces only) and precision (i.e., providing accurate measurements of the applied forces). During our preliminary experiments, we found that commercially available force-sensing resistors (FSR, Interlink Electronics), are better suited for contact detection than accurate force sensing, despite extensive calibration.

An alternative to pressure sensors proposed in [MA01] consists in instrumenting fingers with miniature LEDs and photodetectors to measure changes in fingernails coloration, that are then correlated to the touch force applied at fingertips. Later, this technology evolved to predict normal and shear forces, and even changes in posture, that appear to have different blood volume patterns [MA04]. Fingernail color and surrounding skin changes were also monitored and processed using an external camera system to estimate contact fingertip forces in [SHM08, SHM09, GHM13, UBO+13]. Conversely, computer graphics models were developed to simulate fingertip appearance changes based on simulated forces [AJK13]. This approach is, however, limited to fingertip contacts and requires extensive calibration for each individual user, since nail appearances can vary between subjects and through time. It is also limited by the necessity of having fingernails visible at all time and at high resolution, requiring appropriately mounted miniature cameras. Still, this result illustrates that the problem of estimating contact forces during manipulation can indeed be tackled by computer vision.

1.1.3 Applications of Motion and Force Monitoring

Observing a scene with a single RGB-D sensor in [KA13], Kyriazis et al. showed that the computation of interaction forces explaining the motion of *visible* objects could help infer the motion of objects *hidden* from the camera. Hand-object grasps were classified in terms of shape and contact configurations in multiple taxonomies [Cut89, LFNP14, FRS+16]. These taxonomies were augmented with contact force estimates in [RSR15b] to construct a dataset on functional grasp understanding. Also in action and scene understanding, Zhu et al. also used forces as a physical concept to model tool use [ZZCZ15] and learn human utilities [ZJZ+16], e.g., by quantifying comfort intervals while sitting on a chair in terms of forces exerted on body parts. Kry et al. proposed an acquisition setup combining marker-based motion capture and force transducers to estimate hand joint compliance and synthesize interaction animations [KP06].

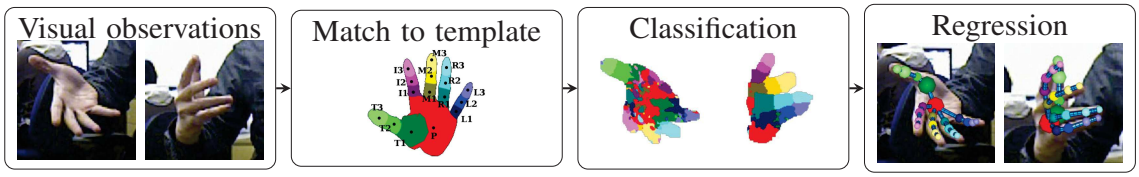
Besides computer vision, grasp taxonomies were used in robot learning from demonstration (LfD) for the planning of in-hand manipulation actions [PPB12]. The monitoring of contact forces is of critical importance for such dexterous actions. Towards this purpose, Liu et al. proposed a robotic fingertip equipped with a 6-axis force-torque sensor and a

rubber skin allowing the accurate and high-speed sensing of contact wrench and even contact location [LNP+15]. In [RJC13], Rozo et al. introduced an LfD framework to teach a robotic manipulator to place a ball in a box and pour a drink relying solely on force perception and haptic feedback. In the context of whole-body motion, [EKO15] modelled ground reaction forces (GRF) measured during human running experiments as polynomial splines to construct a general controller for bipedal running, independent of specific hardware constraints for the considered robot. Similarly, in graphics, the whole-body motion of a human subject was reconstructed from GRFs measured with consumer-grade pressure sensing platforms and a hand-tracking device in [HBL11].

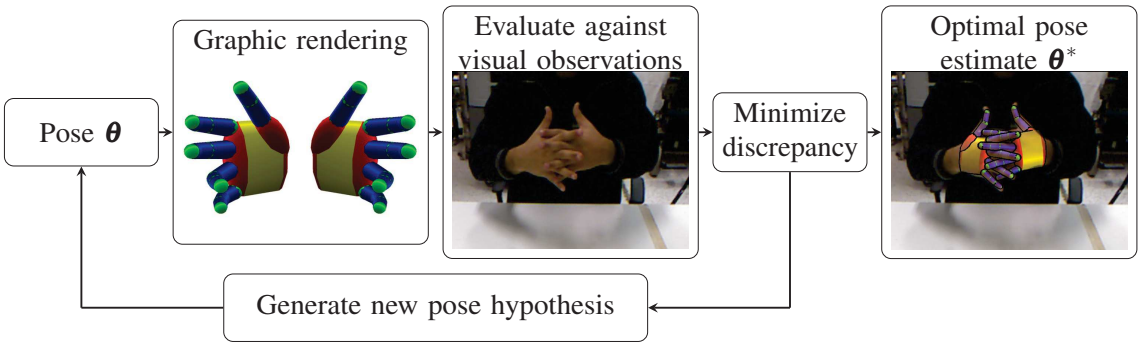
Still in computer graphics, ensuring that a desired motion is physically plausible through the application of compatible contact forces allows the generation of realistic-looking animations. In [Liu08] and [Liu09], Liu introduced an optimization framework allowing the production of such physically plausible manipulation animations with only inputs a starting grasp pose and a partial trajectory for the object. This work was later on extended in [YL12] to animate a hand manipulating objects from the motion capture of the objects and the subject's wrist, without tracking the motion of the individual fingers. Hand control strategies were also constructed in [AK13, BL14] to accomplish general actions such as reorienting a ball in a chosen direction rather than specifying its exact trajectory. Mordatch et al. introduced the contact-invariant optimization (CIO) method in [MTP12], enabling the synthesis of complex animations from high-level goals only by simultaneous optimization of contact and motion. This method was applied to dexterous manipulation in [MPT12]. In the context of visual servoing, i.e., the incorporation of visual information for robot control [ECR92, CH06], Agravante et al. combined haptic information and vision for the human-robot collaborative carrying of a table while preventing a ball on top from falling off [ACB+14].

1.2 Markerless Visual Tracking

Towards the non-intrusive monitoring of contact forces, our work capitalizes on the recent advances of markerless visual tracking for pose estimation and motion perception. Public libraries enabling the robust and efficient tracking of one or multiple objects are now routinely deployed in production or robotics research systems, e.g., ARToolKit [KB99], ViSP [MSC05], BLORT [MPR+10]. Such methods typically rely on feature tracking [MC05] (e.g., dots, contours, SIFT keypoints [Low04]) and can even scale to hundreds of objects in real time [PRR15] or articulated systems [CMC07, PRR14], provided accurate visual and kinematic models of the latter. In contrast, tracking human subjects is subject to different



(a) Bottom-up approach for hand tracking [TYK13].



(b) Top-down approach for the tracking of two hands in interaction [OKA12].

Figure 1.2: Bottom-up and top-down pose estimation methods.

constraints, e.g., body shape and appearance variety across individuals and populations, kinematic complexity and high dimensionality, self-occlusions between body parts. In this section, we review the state of the art in the markerless visual tracking of human subjects, either whole-body or restricted to the hand(s). Pose estimation methods can be classified into mostly bottom-up, mostly top-down, or hybrid. In bottom-up approaches, given visual observations of the subject, the first step is to identify the location of body parts in the image. The detected body parts are then assembled into a complete pose based on considerations such as body part proximity and temporal coherence. Conversely, in top-down approaches, a generative process creates subject pose hypotheses that are evaluated against actual observations. Final pose estimates are obtained by solving a multi-parameter optimization problem, e.g., the minimization of the discrepancy between the pose hypotheses rendered graphically and the visual observations. We depict representative examples of bottom-up and top-down pose estimation approaches in Fig. 1.2.

1.2.1 Bottom-Up Methods

Marker-based motion capture is a case of bottom-up pose identification, as body parts are first localized in space, then matched to a target kinematic structure. In the first work on markerless hand tracking [RK94], Rehg and Kanade extracted fingertip locations and finger bone central axes from the observed silhouette of the hand. Recent bottom-up approaches are typically data-driven and rely on discriminative models to learn a mapping between

visual inputs and poses from labeled examples. Wang and Popović used a colored glove to identify hand parts and track the hand in real time by nearest neighbor search in a hand-pose appearance database [WP09]. Romero et al. generated a database of synthetic hand images to track the hand without a colored glove [RKK09] and later extended this idea to synthetic hand-object poses [RKK10]. Random decision forests (RDF) [Bre01] were trained on synthetic data for hand tracking [KKKA11, KKKA12] and whole-body pose estimation [SFB$^+$11, SSK$^+$13]. Tang et al. explored alternative RDF models and addressed the discrepancy between synthetic data and real observations [TYK13, TCTK14]. In [TSLP14], Tompson et al. extracted hand pose features using RDFs in combination with convolutional neural networks (CNN) [LBBH98]. Rogez et al. recovered arm and hand poses using support vector machines (SVM) in the context of egocentric vision with a chest-mounted camera [RSR15a].

Bottom-up approaches are typically computationally cheap as they avoid the computation of graphic renderings and discrepancies to the visual observations. [SFB$^+$11] thus performed whole-body tracking at 200 frames per second, enabling its use as a side process for real-time applications (e.g., human-computer interaction, video gaming). Another advantage is that bottom-up approaches can operate on a per-frame basis, without relying on temporal coherence, while top-down approaches require manual initialization (at least) for the first frame of the sequence. However, the quality of the pose estimation is directly contingent on the training data and may not generalize well to previously unseen inputs, e.g., different hand poses, occlusion cases, or hand-object interactions.

1.2.2 Top-Down Methods

Rather than *discriminative*, top-down methods are *generative*, or model-based. Using an explicit model of the subject (e.g., 3D geometry, inertial parameters), pose hypotheses are optimized by evaluation against the visual observations. Global optimization is generally not possible due to the large dimensionality of the search space and the computational cost of rendering and evaluating pose hypotheses. Instead, local search is performed in the vicinity of an initial pose estimate, which can be initialized manually for the first frame of the sequence, or taken as the optimal pose found at the previous frame. In an early work on model-based whole-body tracking [GD96], Gavrila et al. decomposed the search space in a hierarchical manner, following the kinematic tree, by searching first for the head and torso, then the upper arms and thighs, then the forearms and lower legs. At each step, the pose of the corresponding limbs was found by discretizing the reduced search space and maximizing a similarity measure between visual observations given by a calibrated multi-camera setup and the synthesized appearance of the pose hypotheses in each camera view.

In [dLGFP11], de La Gorce et al. proposed an objective function differentiable with respect to the hand pose and its lighting, allowing its efficient optimization using a quasi-Newton method. Still, the 3D color rendering of a hand under different conditions of lighting remained particularly expensive, leading to a total computation time of approximately 40 s per frame (i.e., 0.025 fps). In addition to color, depth information has been increasingly used in recent approaches. The iterative closest point (ICP) algorithm [BM92], used to minimized the discrepancy between two point clouds, was adapted to whole-body tracking in [GPKT12] and ran at 125 fps on a single-threaded CPU implementation. However, ICP alone is easily trapped in local minima. Tagliasacchi et al. [TST+15] augmented the ICP with priors built from a hand pose database. Both approaches achieved the tracking of a hand from depth in real time. The signed distance function (SDF) [CL96] used previously in conjunction with ICP for surface mapping and tracking [NIH+11] was extended by Schmidt et al. to general articulated models [SNF14] such as the hand or the whole body.

Ballan et al. tracked two hands in interaction with an object [BTG+12] by searching for salient points (e.g., fingernails and fingertips) using a discriminative model, in combination with the local optimization of a differentiable cost function accounting for edges, optical flow and collisions. Tzionas et al. extended this approach with physical simulation to improve the realism of the hand-object pose estimates [TBS+15]. Oikonomidis et al. tracked a hand with particle swarm optimization (PSO) [KE95] using either a multi-camera setup [OKA10] or a single RGB-D sensor [OKA11a]. The use of stochastic optimization alone allows the incorporation of arbitrary priors in the optimization process regardless of differentiability constraints. The fact that two different objects cannot share the same physical space was implemented by penalizing interpenetrations between 3D shapes in the cost function, allowing the same framework to also track a hand in interaction with an object [OKA11b] or two strongly interacting hands [OKA12]. Complete occlusions were also treated through physics-based simulation [KA13]. Wang et. al proposed a contact-based sampling approach allowing the monitoring of subtle hand-object interactions during dexterous manipulation using a multi-camera setup [WMZ+13] and physics-based simulation.

In effect, top-down methods allow the treatment of occlusions not as a distractor, but rather as a source of information. A major advantage resides in their ability to tackle virtually any situation, provided the models of all the objects in the scene. However, their computational cost is generally considerable, although modern implementations (e.g., GPGPU) now enable real-time tracking. Still, reinitializing the pose search during the sequence remains problematic (e.g., as tracking errors accumulate through time or when the subject exits the field of view).

1.2.3 Hybrid Methods

Hybrid methods aim at combining the advantages of top-down and bottom-up approaches. Discriminative methods can provide a fast and rough pose estimate to efficiently initialize a slower but more refined search through local optimization.

In [SOT13], Sridhar et al. combined a generative method based on a Sum of Gaussians (SoG) model, with a linear SVM classifier detecting fingertips from depth maps. The SoG model was introduced earlier for whole-body tracking [SHG+11] to provide a differentiable cost function for fast local optimization. The SoG model was also used in combination with RDFs for the tracking of a hand in [SMOT15] and that of a hand manipulating an object [SMZ+16], in real time. Qian et al. [QSW+14] combined a fingertip detector with a generative model making use of both ICP for fast local optimization and PSO to explore the search space more thoroughly. In the approach of Sharp et al. [SKR+15], the discriminative model does not produce just a single good pose estimate but rather a distribution over possible hand poses, which are then fitted to the depth observations using a variant of the PSO algorithm. The resulting hand tracker was highly flexible with respect to camera placement, demonstrated great robustness to tracking failure, and ran in real time on a GPU implementation. This approach was extended in [TBC+16] with a smooth hand model and a differentiable cost function, enabling the use of gradient-based optimization techniques in the generative process for real-time tracking on CPU only.

Overall, hybrid methods have demonstrated promising results for the tracking of a hand in isolation. However, the tracking of multiple hands or hand-object interactions has been comparatively less studied. For these situations, current approaches suffer from the same drawbacks as bottom-up methods, in particular in the presence of strong occlusions.

1.3 Model-Based Hand-Object Tracking

While computationally expensive, top-down, model-based tracking methods allow the tracking of arbitrary subjects and objects in a unified framework. In this section, we review the FORTH hand-object tracking method [OKA10, OKA11a, OKA11b, OKA12, KA13], upon which we built our framework for force sensing from vision. First, we describe the requirements of the method (Section 1.3.1). We then summarize the hypothesize-and-test pose estimation strategy (Section 1.3.2). Finally, we discuss the incorporation of tracking priors in the optimization process (Section 1.3.3).

1.3.1 Observations and Models

We consider a set of N_C cameras calibrated intrinsically and extrinsically. At a given time step i, we denote by $F_{i,j}$ the *frame* acquired by camera j. We call *multiframe* M_i the set of images captured by all cameras at time step i, i.e., $M_i = \left\{F_{i,1}, \ldots, F_{i,N_C}\right\}$. We denote by S the sequence of N_S multiframes acquired through time, i.e., $S = \{M_1, \ldots, M_{N_S}\}$. In [OKA10, OKA11b], the multiframes are acquired by 8 color cameras observing the scene at 30 fps and 1280×960 resolution. In [OKA11a, OKA12, KA13], the scene is observed with a single RGB-D sensor providing synchronized color and depth maps at 30 fps and 640×480 resolution. Fig. 1.3a illustrates sample color and depth frames captured by a SoftKinetic RGB-D sensor.

Hands and objects are tracked based on their 3D shape. In the case of rigid objects, a dedicated 3D model must be provided (e.g., made available by the manufacturer or by CAD). The pose of a rigid object is then characterized by 6 DoF, i.e., its 3D orientation and the position $\mathbf{p} = (p_x, p_y, p_z)^T$ of a reference point of its geometry. For numerical reasons, 3D orientations are commonly represented by unit quaternions $\mathbf{q} = (q_x, q_y, q_z, q_w)^T$ among other representations and conventions (e.g., Euler angles, orthogonal matrices, etc.). The 6-DoF pose of a rigid object can thus be encoded by a 7-element vector:

$$\begin{aligned} &\boldsymbol{\theta}^{\text{rigid}} = (q_x, q_y, q_z, q_w, p_x, p_y, p_z)^T, \\ &\text{with} \quad \|\mathbf{q}\|_2 = 1. \end{aligned} \tag{1.1}$$

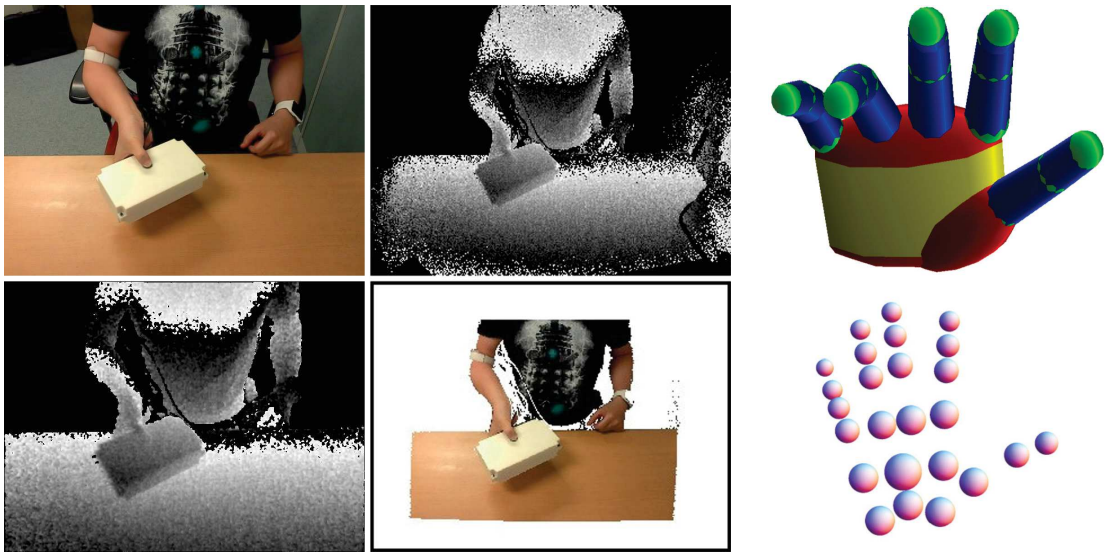
The hand is modeled as an articulated system of rigid bodies following [AHS03]: the palm and three segments for each of the five fingers. We denote the finger set by $\mathscr{F}$:

$$\mathscr{F} = \{\text{thumb}, \text{index}, \text{middle}, \text{ring}, \text{pinky}\}. \tag{1.2}$$

The palm is chosen as root of the kinematic tree and its global pose is encoded by 7 parameters as in Eq. (1.1). The finger poses are then characterized by 4 parameters each, i.e., 2 DoF for the finger base and 1 DoF for each of the two remaining joints. In total, the 26-DoF hand model is encoded by a 27-element vector:

$$\boldsymbol{\theta}^{\text{hand}} = \left(\boldsymbol{\theta}^{\text{palm}}, \left(\boldsymbol{\theta}^{k}\right)_{k \in \mathscr{F}}\right), \tag{1.3}$$

with $\boldsymbol{\theta}^{\text{palm}}$ the 7-parameter pose of the palm and $\boldsymbol{\theta}^{k}$ the 4 joint angles of finger $k \in \mathscr{F}$. Provided an instance of $\boldsymbol{\theta}^{\text{hand}}$, the 3D pose of each segment is computed by forward kinematics. To each segment is associated a visual representation using a combination of the following geometric primitives: cones, cylinders, ellipsoids and spheres, Autocollisions between different segments of the hand and interpenetration between hands and objects



(a) Raw color (top left) and depth (top right) are captured separately by two cameras. From the intrinsic parameters and relative positions, depth is registered to color (bottom left) and conversely (bottom right).

(b) Articulated 3D hand model (top) and sphere-based collision model (bottom) [OKA11b].

Figure 1.3: Tracking inputs and models. (a): RGB-D sensor observations, (b): hand model.

are computed using a simplified sphere-based collision model for faster computation. We represent both models in Fig. 1.3b.

1.3.2 Pose Estimation Strategy

In this section, we describe a representative pipeline for the tracking of a hand in isolation using a single RDB-D sensor, as performed in [OKA11a]. More complex cases are treated with the incorporation of tracking priors in the objective function, discussed in Section 1.3.3.

We consider a color image of the hand and the corresponding depth map as acquired by an RGB-D sensor at a given time step i. First, the area of interest is obtained by segmenting the hand from the rest of the color image by skin color detection [AL04]. We denote by o_s the 2D map of the segmented skin color. The depth of the skin-colored pixels is conserved from the raw depth map, while the rest is set to zero. We denote by o_d the resulting depth map. The segmented color and depth observations are $O = (o_s, o_d)$.

The objective is then to find the hand pose estimate $\boldsymbol{\theta}^*$ that minimizes a cost function $\mathscr{E}$, or energy, quantifying the discrepancy between observations O and pose hypotheses $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \{\mathscr{E}(\boldsymbol{\theta}, O)\}. \tag{1.4}$$

(a) Left to right: raw RGB, aligned depth, segmented hand, pose estimate. (b) Hand-object tracking.

Figure 1.4: Tracking: (a) hand in isolation [OKA11a], (b) two hands and multiple objects [KA14].

The energy $\mathscr{E}$ is comprised of two terms: a cost $\mathscr{E}_{\text{3D}}$ measuring the discrepancy between the visual observations and the rendered 3D hand model, and a cost $\mathscr{E}_{\text{kin}}$ that quantifies the likelihood of the hand pose hypothesis by itself, regardless of the observations:

$$\mathscr{E}(\boldsymbol{\theta}, O) = \lambda_{\text{3D}} \cdot \mathscr{E}_{\text{3D}}(\boldsymbol{\theta}, O) + \lambda_{\text{kin}} \cdot \mathscr{E}_{\text{kin}}(\boldsymbol{\theta}), \tag{1.5}$$

with λ_{3D} and λ_{kin} normalization coefficients adjusted manually for the optimization process. Given the camera intrinsic and extrinsic parameters, a hand pose hypothesis $\boldsymbol{\theta}$ can be rendered graphically to generate a synthetic depth map r_d. The cost function $\mathscr{E}_{\text{3D}}$ thus implements a discrepancy metric between synthetic and measured depth maps, e.g., the pixel-wise absolute difference between r_d and o_d. Regardless of the observations, $\mathscr{E}_{\text{kin}}$ penalizes kinematically implausible hand poses, e.g., those resulting in auto-collisions between finger segments. We depict the pose estimation process in Fig. 1.4a.

1.3.3 Incorporating Tracking Priors

Provided perfect observations, the optimization of the visual discrepancy cost function $\mathscr{E}_{\text{3D}}$ alone could theoretically lead to the real hand pose. In reality, observations are often partially missing (e.g., occlusions) or subject to measurement uncertainties (e.g., motion blur). To guide the pose estimation, tracking priors are incorporated through additional terms in the definition of the objective function $\mathscr{E}$ in Eq. (1.5). As reviewed in Section 1.2.2, the energy $\mathscr{E}$ is minimized by PSO. This choice is motivated by the limited number of hyperparameters to adjust manually (e.g., number of particles) and its efficiency in exploring the search space beyond local optima. Additionally, it does not impose any constraint on the objective function (e.g., differentiability). Thus, arbitrary priors can be incorporated in the optimization process in a unified computational framework.

In [OKA11a], the hand kinematic plausibility cost $\mathscr{E}_{\text{kin}}$ penalized overlapping adjacent fingers by comparing their abduction-adduction angles (i.e., how fingers are spread apart). This approach was also used for the tracking of two hands in interaction [OKA12]. However, this analytical formulation did not account for every possible collision (e.g., between thumb and pinky fingertips) and penalized certain valid poses (e.g., crossed index and middle fingers). Instead, [OKA11b] computed hand-hand and hand-object penetration volumes using a third-party physics engine [S$^+$05], exploiting the flexibility of PSO with respect to the objective function. In [KA13], the cost function was augmented with a term comparing the observations with physics-based simulation outputs [C$^+$13], allowing the pose recovery of fully occluded objects based solely on the tracking of the hand and the simulation of its effect on the objects of the scene, visible or not.

Another advantage of PSO is its parallel nature, enabling real-time implementations on the GPU [OKA11a, OKA12]. However, the computational complexity grows geometrically with the number of tracked subjects and objects, when accounting for them simultaneously. On the other hand, using multiple independent trackers scales better with the number of objects but does not account for occlusions between them. The scalability of generative methods was addressed in [KA14] through the concept of Ensemble of Collaborative Trackers (ECT), in which each individual trackers per object broadcast their results with each other. Fig. 1.4b depicts the tracking of two hands in interaction with multiple objects. In this thesis, we tracked hand-object interactions using a variant of ECT.

1.4 Modeling Contact Dynamics

In this section, we first review existing techniques for the modeling of the human body (Section 1.4.1). We then discuss the estimation of contact dynamics for the cases of whole-body contacts with the environment (Section 1.4.2) and hand-object interactions (Section 1.4.3).

1.4.1 Human Dynamic Model

The identification of the objective function optimized by the central nervous system in daily activities (e.g., locomotion, manipulation) is a long-standing problem in kinesiology research [Zat02]. In [PZ02], Prilutsky and Zatsiorsky reviewed the state of the art in the prediction of muscle activation patterns (MAP) from optimization, i.e., the understanding of how human efforts are regulated at the musculoskeletal level by the central nervous system (CNS). In doing so, they noted that a major difficulty lies in the high dimensionality of the human body, which allows 244 kinematic DoFs with approximately 630 muscles. As such, it

is a highly redundant system, making it difficult to consider biological parameters in isolation. In addition, the observability of such parameters (e.g., muscle forces, joint torques) may be limited without invasive surgery, which further hinders optimization-based techniques requiring ground-truth data.

Whole-body motion and forces are also linked through the equations of motion and the Body Segment Inertial Parameters (BSIP), i.e., the mass, position of the center of gravity and inertia tensor of each body segment. In an early work by Dempster [Dem55], eight cadavers were dismembered to provide data on mass, center of gravity, density and moments of inertia. Further measurements on cadavers were performed in [CCM+75] and compared to body segments models based on geometric primitives such as cylinders and ellipsoids, as done in [HJ64]. McConville et al. combined anthropomorphic measurements from 31 living male subjects with densities measured on cadavers to construct tables linking body segment measurements and inertial parameters [MCC+80]. Young et al. extended this approach on 46 living female subjects [YCS+83]. BSIP estimation on living subjects was also performed by Zatsiorsky et al. using gamma-ray in [ZS83, ZSC90b, ZSC90a]. Deleva et al. adjusted these results to alternative anthropomorphic conventions in [DL96]. Similarly, the tables of McConville et. al and Young et al. were adjusted by Dumas et. al in [DCV07], which we use in this thesis.

While anthropomorphic tables allow the fast computation of BSIP estimates from body segment measurements only, an important caveat resides in the specificities of the studied subjects, e.g., college-aged Caucasian males and females in [ZSC90b], whose results may not directly extend to other populations. Furthermore, the resulting BSIPs generally assume symmetry between left and right halves of the human body, which may be a rather inaccurate assumption, e.g., when particular muscles are atrophied due to neuromuscular diseases. Towards these issues, Jovic et al. proposed a hierarchical optimization framework for the online estimation of both robot and human BSIPs from direct motion and force-torque measurements [JEA+16], without further assumption. In [BV15], Bonnet and Venture used a Microsoft Kinect RGB-D sensor for visual tracking and a Nintendo Wii Balance Board for force-torque sensing, enabling the online estimation of BSIPs with consumer-grade sensors.

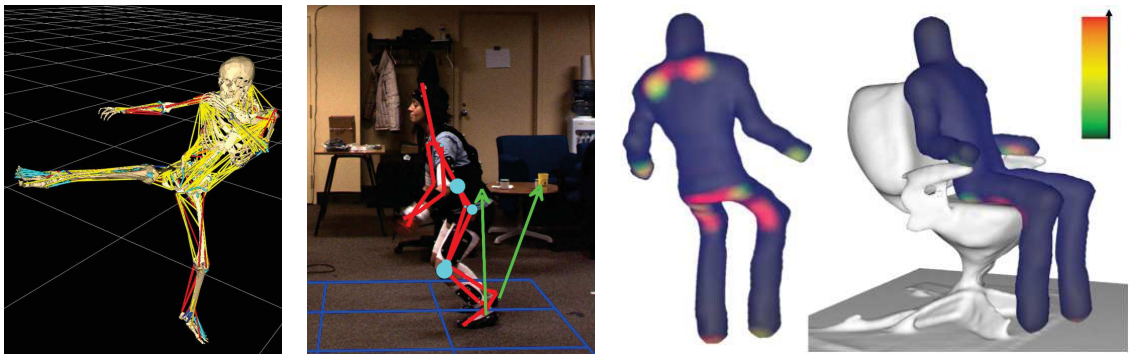
1.4.2 Whole-Body Dynamics

Whole-body motion and ground reaction forces can typically be measured accurately using commercially available solutions, e.g. marker-based motion capture and force plates, as reviewed in Section 1.1. Measuring muscle forces in vivo is difficult without invasive surgery. Instead, a common approach in biomechanics research is to first use the equations of motion to estimate joint torques through inverse dynamics [DA87], and compute muscle forces that

minimize criteria such as metabolic energy expenditure [AP01]. However, inverse dynamics solutions are contingent on the underlying BSIP model. The influence of BSIP estimation on inverse dynamics was studied in [PC99, RABF06, RHCF07, WDGJ14] in the context of gait analysis. In [MGPD15], Muller et al. estimated joint torques during overhead throwing and reported relative distances up to 70% between BSIP obtained with three different models of the literature. Overall, such approaches are limited by the difficulty to obtain ground-truth data, often limited to muscle excitation patterns measured with electromyography (EMG).

Alternatively, the human motion can be studied from the perspective of sensorimotor control optimality, reviewed in [Tod04]. A given task, e.g., taking a step foward, can generally be executed in multiple ways due to the redundancy of the human body. Still, it is commonly assumed that humans naturally execute movements that are optimal with respect to certain criteria, shaped by evolution and experience [Ale84]. Prilutsky and Zatsiorsky suggested that such criteria might be based on three major physiological cost functions: metabolic energy expenditure, muscle fatigue, and sense of perceived effort [PZ02]. In [Mom09], Mombaur formulated the generation of running motions as an *optimal control* problem, i.e., the computation of trajectories that respect a given set of constraint and are optimal with respect to a chosen cost function. Conversely, in *inverse optimal control*, the goal is to find the cost function that is optimized during the observed motion. In [LHP05], Liu et al. captured motion styles (e.g., sad / happy) in terms of preferences for applying torques at some joints rather that others to synthesize new walking and running animations. In [MTL10], Mombaur et al. identified cost functions optimized during human locomotion from motion capture and applied the resulting models to generate natural trajectories on a humanoid robot.

Very much related to our work, the estimation of contact dynamics was also addressed from a computer vision perspective. In [NYFS05], Nakamura et al. performed inverse dynamics on a detailed model of the human body to compute sensomatory information from motion capture, e.g., stimuli perceived at the level of the organs, muscles, tendons and ligaments. Our work was also inspired by Brubaker et al., who estimated joint torques from inverse dynamics and motion capture by parameterizing GRFs with a spring-based model [BSF09]. General contact configurations between the whole body and objects of the environment were also computed by considering the human body elastic [ZJZ$^+$16]. Zhu et al. used the resulting force estimates to learn human utilities when interacting with their environment, e.g., quantifying preferred poses when sitting in terms of contact forces between the body and the chair. While a major limitation to these works lies in the difficulty of validating the force models with ground-truth measurements, they also illustrate the interest of capturing contact dynamics for action understanding in computer vision, as illustrated in Fig. 1.5.



(a) Muscle tensions, low (yellow) to high (red) [NYFS05].

(b) External forces and joint torques (blue spheres) [BSF09].

(c) Contact forces when sitting from finite element analysis with deformable body model [ZJZ+16].

Figure 1.5: Dynamics estimation from motion capture for human and action understanding.

1.4.3 Prehension and Manipulation Dynamics

Besides whole-body dynamics estimation, the case of prehension and manipulation is another active topic of interest for kinesiology research. This interest stems from the remarkable dexterity of the human hand. It particular, it is a complex and redundant system, such that a given task can generally be executed using multiple force distributions. Similarly to the case of whole-body locomotion, it is difficult to identify clear criteria supposedly optimized by the CNS during multi-finger prehension. From experiments on thumb-index pinching [WJ84], Westling and Johansson showed that grip control is mostly influenced by the object's surface condition and its mass through the safety margin ratio, defined as the proportion of applied forces that are unnecessary to achieve the object's observed kinematics, with respect to the total forces applied. Cadoret and Smith precised that the influence of the surface condition is most important regarding its friction coefficient rather that its texture [CS96].

The notion of safety margin is largely related to that of nominal and internal forces. Humans do not manipulate objects using nominal closures (i.e., minimal grasp forces). They tend to "over-grasp" and produce workless internal forces, i.e. firmer grasps than mechanically required through the equations of motion. This grasping property is described by considering finger forces as two sets formalized in [KR86, YN91]: nominal forces responsible for the object's motion and internal forces that cancel each other out [MS85, MSZ94] and thus do not affect the object's kinematics. For instance, when holding a cup statically, nominal forces directly compensate gravity, while internal forces secure the object in place. Humans typically apply internal forces to prevent slip [JW84, FJ02] and control their magnitude to avoid muscle fatigue or damaging fragile objects [GZL10, PSZL12].

When manipulating objects, excessive forces applied by a finger are naturally compensated by others, making the problem of force sharing between fingers particularly challenging.

Thus, inverse optimization approaches for manipulation have mostly resulted in models that rely on rather strong simplifying assumptions. The most common restriction is on the motion's dimensionality. e.g., static prehension [NTLZ12]. Other approaches allow limited motion, but using a simplified grasp model in which individual fingers and hand surfaces are grouped into functional units named virtual fingers [AIL85, IBA86]. For instance, a hand holding a cup is seen as the thumb on one side and a virtual finger on the opposite side, that realizes the total wrench due to the four antagonist fingers. Under this formalism, a five-finger grasp is effectively seen as *two-finger*. The thumb-virtual finger model was used in conjunction with nominal-internal force decompositions on 1D horizontal, vertical and transversal cycles in [GLZ05], and to predict normal forces on 2D circular trajectories in [SLZ11]. In this simplified model, given the object's kinematics, the knowledge of one force fully determines the other through the equations of motion. This greatly conceals the issue of force sharing indeterminacy. since in reality, full-hand forces can compensate each other in an infinity of different distributions that all cause the same motion.

In computer vision and haptics, Mohammadi et al. computed forces between the hand and deformable objects by finite element analysis [MBSP16]. Rogez et al. showed that manipulation forces play a crucial role in hand-object interaction understanding[RSR15b], and noted the challenge of obtaining the ground-truth contact points and forces humans use instinctively, which we address in our work.

1.5 Numerical Techniques

In this section, we discuss the techniques used throughout this thesis for numerical differentiation (Section 1.5.1), physics-based optimization (Section 1.5.2), and time-series modeling (Section 1.5.3).

1.5.1 Numerical Differentiation

Through the Newton-Euler equations, contact forces applied during manipulation determine the resulting kinematics in terms of linear and rotational velocity and acceleration. These quantities are not directly provided by common positional tracking techniques, such as the markerless visual tracking approach of [KA14], that instead capture linear and rotational positions. Mathematically, velocities and accelerations can directly be computed from the first and second-order derivatives, respectively, of the tracked positions. In practice, this

is a delicate process due to tracking noise and sensing uncertainties. In such situations, straightforward finite difference leads to exploding velocities and accelerations. On the other hand, smoothing techniques may attenuate acceleration spikes occuring during manipulation.

A common approach for the numerical differentiation of noisy signals consists in finding a smooth approximation of the original signal. e.g., spline interpolation with a least-square criterion [ID04]. However, such approximations are constructed based on the observation of the whole signal, or at least a large portion of it. Instead, the well-known Gaussian smoothing in image processing can be implemented as a finite impulse response filter, making it suitable for real-time applications. Still, in the context of motion capture, while the visual acquisitions themselves (e.g., depth sensing) may be subject to Gaussian noise, the errors do not necessarily follow the same statistical properties (e.g., pose estimation errors can accumulate through time, rather than be uncorrelated). From the field of control engineering, Fliess and Sira-Ramírez introduced an algebraic framework for parameter identification in linear systems [FSR03]. This framework laid the foundation for the recent theory of model-free control [FJ09] as well as a new class of derivative estimators for noisy signals, independent of the noise statistical properties, termed algebraic numerical differentiators [MJF09].

In the original work of Mboup et al. [MJF09], the goal is to identify the n-th order derivative of a noisy signal x. With $N \geq n$, we consider its N-th order truncated Taylor expansion:

$$x_N(t) = \sum_{i=0}^{N} x^{(i)}(0)\frac{t^i}{i!}, \tag{1.6}$$

which is such that $\frac{d^{N+1}}{dt^{N+1}} x_N(t) = 0$. Moving into the frequency domain, Eq (1.6) becomes:

$$\begin{aligned} s^{N+1}\hat{x}_N(s) &= s^{N+1}\sum_{i=0}^{N} x^{(i)}(0)\frac{1}{s^{i+1}} = \sum_{i=0}^{N} s^{N-i}x^{(i)}(0), \\ &= s^N x(0) + \cdots + s^{N-n}x^{(n)}(0) + \cdots + x^{(N)}(0). \end{aligned} \tag{1.7}$$

The target derivative $x^{(n)}(0)$ is estimated by considering the terms $\left(s^{N-i}x^{(i)}(0)\right)_{i\neq n}$ as undesired perturbations that are to be "annihilated" using a linear differential operator Π and a complex-valued function ρ such that:

$$\Pi\hat{x}_N = \rho(s)x^{(n)}(0). \tag{1.8}$$

Mboup et al. proposed candidate Π and ρ operators that, moving back Eq. (1.8) into the time-domain, can be implemented as finite impulse response filters. This work was further extended to multidimensional signals, over possibly irregular sampling grids [RMR11a, RMR11b]. We produced an open-source implementation of these works through the course of this thesis [1].

1.5.2 Physics-Based Optimization

Optimization methods play a key part in many physics-inspired problems. In the context of rigid body dynamics, the Newton-Euler equations dictate that the net force $\mathscr{F}$ and torque $\boldsymbol{\tau}$ exerted on a body of mass m and inertia tensor $\mathbf{J_q}$ are linked to its linear acceleration $\mathbf{a}$, rotational velocity $\boldsymbol{\omega}$, and rotational acceleration $\boldsymbol{\alpha}$ by:

$$\begin{cases} \mathscr{F} = m\mathbf{a} \\ \boldsymbol{\tau} = \mathbf{J_q} \cdot \boldsymbol{\alpha} + \boldsymbol{\omega} \times (\mathbf{J_q} \cdot \boldsymbol{\omega}) \end{cases}. \tag{1.9}$$

Each contact force $\mathbf{F}_i^c$ can be decomposed along normal $\mathbf{n}_i$ and tangential $\mathbf{t}_i$ vectors as:

$$\mathbf{F}_i^c = f_i \mathbf{n}_i + \widetilde{g}_i \mathbf{t}_i. \tag{1.10}$$

Orienting $\mathbf{n}_i$ and $\mathbf{t}_i$ such that $f_i \geq 0$ and $\widetilde{g}_i \geq 0$ and denoting by μ the friction coefficient at contact i, the Coulomb friction model takes the following simple form:

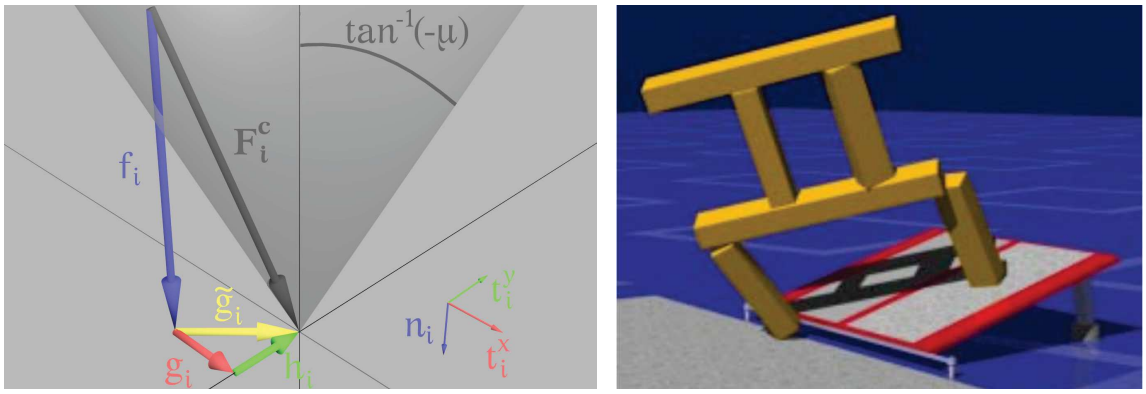
$$\widetilde{g}_i \leq \mu f_i. \tag{1.11}$$

In the case of dynamic friction, i.e., when the relative tangential velocity $\widetilde{\mathbf{v}}_i$ between the two surfaces in contact is non-zero, Eq. (1.11) becomes an equality and $\mathbf{t}_i$ is parallel to $\widetilde{\mathbf{v}}_i$.

In the context of dynamics simulation (see Fig. 1.6b), Baraff formulated Eq. (1.9) as equality constraints of a linear program (LP) to compute contact forces preventing interpenetration between rigid bodies [Bar89]. In [Bar91], Eq. (1.11) was incorporated as a linear inequality in the LP to model dynamic friction in conjunction with physics-based simulation. The case of static friction is more complex, since the direction of $\mathbf{t}_i$ is not known *a priori* in the 2D tangential plane. Instead, it is described by two orthogonal vectors $\mathbf{t}_i^x$ and $\mathbf{t}_i^y$, which we depict in Fig. 1.6a :

$$\mathbf{F}_i^c = f_i \mathbf{n}_i + g_i \mathbf{t}_i^x + h_i \mathbf{t}_i^y. \tag{1.12}$$

[1] https://github.com/ph4m/eand



(a) 3D normal-tangential force decomposition. (b) Rigid body dynamics simulation [Bar91].

Figure 1.6: Coulomb friction model and dynamics simulation with quadratic programming.

In that case, the Coulomb friction constraint becomes:

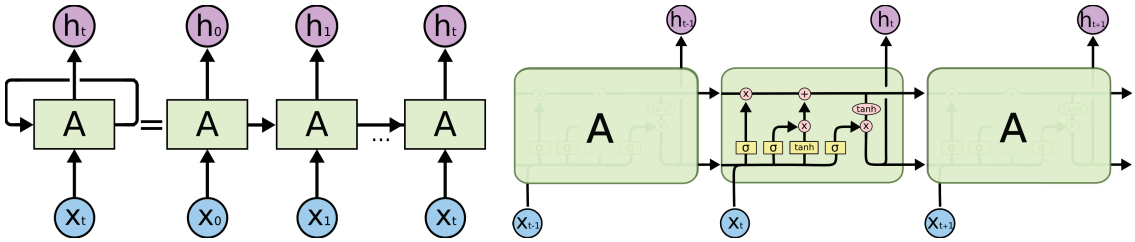
$$\left\| g_k \mathbf{t}_k^x + h_k \mathbf{t}_k^y \right\|_2 \leq \mu_k f_k, \tag{1.13}$$

which Baraff addressed using quadratic programming (QP) with a discretized friction cone. Cone constraints such as that of Eq. (1.13) were later addressed without linear approximations by means of second-order cone programming (SOCP) [LVBL98, BW07].

1.5.3 Neural Networks for Time Series Modeling

The recent successes of deep learning applications for whole-body control, manipulation and monitoring of human activities [MLA+15, LLS15, ZZCZ15, KS16], suggest that data-driven approaches can successfully account for model or perception uncertainties while avoiding the need for arbitrary constraints and hand-engineering [BCV13]. At the heart of these approaches lies the notion of artificial neural networks (ANNs), that take a scalar-valued vector as input and typically passes it through linear transformations and nonlinear activation functions (e.g., hyperbolic tangent). In the context of supervised learning, i.e., when both inputs and expected outputs are available, interconnection weights are updated by backpropagation, i.e., gradient descent with respect to a chosen cost function (e.g., squared prediction error).

In the recent review of [LBH15], LeCun et al. report that the recent advances in parallel programming on GPUs made it easier and faster for researchers to train large neural network models. Deep learning methods thus consist in the multi-layered stacking of simple modules. A popular architecture in image processing is that of the convolutional neural network (CNN) [LBBH98], designed to process multidimensional data (e.g., 2D images, 3D videos).



(a) Unrolled recurrent neural network. (b) Long short-term memory cell state.

Figure 1.7: RNN and LSTM graphic visualization [Ola15].

In this thesis, we capture the sequential nature of contact dynamics using recurrent neural networks (RNN) [Elm90]. In essence, RNNs are networks of neurons that maintain an internal state, updated with each input. Graphically, RNNs can be visualized as networks with loops that allow information to persist through time, as illustrated in Fig. 1.7a. Long short-term memory (LSTM) networks [HS97] implement an explicit memory mechanism and have demonstrated better performance at handling long-term dependencies than regular RNNs on multiple applications, such as machine translation [CVMG+14] or even in conjunction with CNNs for image captioning [VTBE15]. We subsequently use LSTMs to model contact dynamics as a particular type of time series.

Chapter 2

Towards Force Sensing From Vision: Observing Hand-Object Interactions to Infer Manipulation Forces

2.1 Introduction

Reliably capturing and reproducing human haptic interaction with surrounding objects by means of a cheap and simple set-up (e.g., a single RGB-D camera) would open considerable possibilities in computer vision, robotics, graphics, and rehabilitation. Computer vision research has resulted in several successful methods for capturing motion information. A challenging question is: to what extent can vision also capture haptic interaction? The latter is key for learning and understanding tasks, such as holding an object, pushing a chair or table, as well as enabling its reproduction from either virtual characters or physical (e.g., robotic) embodiments.

Contact forces are usually measured by means of haptic technologies such as force transducers. The main drawback of such technologies is that they are obtrusive. Computer vision techniques would therefore be an ideal alternative to circumvent this issue. Yet, is it possible to estimate forces from visual observation? There is evidence that haptic perception can be induced through illusion and substitution dominated by vision, e.g. [LCK+00]. We aim at exploring computer vision to infer the forces exerted by humans on surrounding objects. In particular, we consider hand-object grasping and manipulation. The problem is extremely complex. Indeed, establishing that a hand-object contact has occurred is difficult because of occlusions and tracking inaccuracies. Nevertheless, the detection of events like an object being lifted or discontinuities in body motion may provide useful hints towards

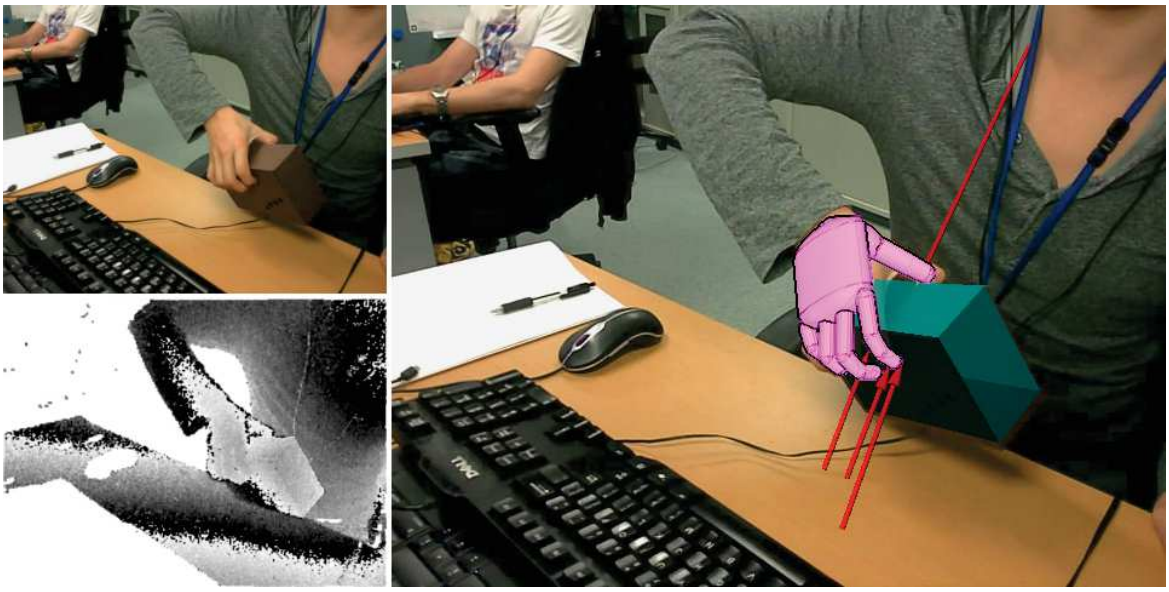


Figure 2.1: Using a single RGB-D camera, we track markerless hand-object manipulation tasks and estimate with high accuracy contact forces that are applied by human grasping throughout the motion.

disambiguating discrete events. Additionally, even if contact positions can be determined efficiently, the estimation of the applied forces is still challenging because of the inherent multiplicity of solutions.

We demonstrate that, by solely using computer vision, it is possible to compute interaction forces occurring in hand-object manipulation scenarios where object properties such as shape, contact friction, mass and inertia are known, along with the geometry of the human hand. First, we monitor both the hand and the object motions by using model-based 3D tracking (other visual tracking techniques can also be used if they meet performance requirements). From the tracking data, we estimate hand-object contact points through proximity detection. Algebraic filtering computes the object's kinematics, i.e. velocity and acceleration. Contact force distributions explaining the kinematic observations are then resolved using conic optimization. When manipulating objects, humans typically apply more (internal) forces than the (nominal) forces required from the Newton-Euler dynamics. Thus, we improve our estimation method by using neural networks to learn the amount and distribution of these internal forces among the fingers in contact. The experimental results obtained on datasets annotated with ground-truth measurements show the potential of the proposed method to infer hand-object contact forces that are both physically realistic and in agreement with the actual forces exerted by humans during grasping. To the best of our knowledge, this is

the first time that this problem is addressed and solved based solely on markerless visual observations.

2.2 Force Sensing From Vision

We consider a rigid body of mass m and inertia matrix $\mathbf{J}$ relative to the center of mass $\mathbf{G}$. For a given element e of the environment (e.g. human hand, table), let μ_e denote the corresponding Coulomb friction coefficient between the object and e. In this work, we assume these quantities known, e.g., obtained from the object's CAD model or existing identification techniques [SL01]. Interestingly, it has been shown that aspects of such information (e.g., mass distribution) can also be estimated by visual means [MB13]. We then consider a scenario where the object is grasped and manipulated by a human hand, with possible contacts with the environment. We observe the scene with a single RGB-D camera that we suppose calibrated intrinsically and extrinsically so that the direction of the gravity vector is known. Our goal is to estimate the interaction forces between the object and the user's hand, and between the object and the environment when such contacts occur. We address the problem of force sensing from vision (FSV) in four steps, as follows:

1. We track the object and the hand and perform, for each time step, vision-based proximity or collision detection to identify contacting fingers and corresponding contact points (Section 2.2.1).

2. Let $\boldsymbol{\theta}_i = (\mathbf{p}_\mathbf{G}, \mathbf{q})$ be the estimated 6-DoF object pose at instant i, with $\mathbf{p}_\mathbf{G}$ the 3D position of the center of mass and $\mathbf{q}$ the object's orientation, encoded by a quaternion. Based on the sequence of pose estimates $(\boldsymbol{\theta}_i)_{i\in[0,N]}$, we estimate for each frame the body's first and second-order kinematics, i.e. translational (resp. rotational) velocity $\mathbf{v}_i$ (resp. $\boldsymbol{\omega}_i$) and acceleration $\mathbf{a}_i$ (resp. $\boldsymbol{\alpha}_i$) (Section 2.2.2).

3. We compute a (nominal) force distribution explaining the object's state computed at step 2 following the Newton-Euler's laws of motion and Coulomb's friction model, using the contact points identified at step 1 (Section 2.2.4).

4. We learn and reproduce how humans naturally distribute among the fingers in contact (Section 2.2.6).

Each of these subproblems presents a number of challenges. First, the observation of manipulation tasks may be subject to mutual occlusions between the hand and the object. To overcome this issue, we address step 1 by means of model-based tracking as inspired

by [KA14]. Second, the limited camera acquisition frequency along with tracking errors can make the differentiation process of step 2 unstable. We tackle this issue by estimating derivatives using algebraic filtering derived from [MJF09]. Algebraic filtering was chosen for the sake of robustness, as it relies on no statistical assumption on the signal's noise. We then address step 3 by computing minimal force closure distributions as solutions of a second-order cone program (SOCP). Finally, step 4 stems from the fact that in contrast with [SHM08] where multiple photodetectors monitor each fingernail's blood flow individually, such microscopic features cannot be observed by a single RGB-D camera observing the whole scene. The object may indeed be grasped with more or less intensity without this being visible at a macroscopic scale. We tackle this statical indeterminacy with machine learning on usual human grasping practices.

2.2.1 Hand-Object Tracking

Our approach requires a good 3D pose estimate of the manipulated object together with that of the user's hand. To achieve this, we rely on a variant of the method proposed in [KA14] that is tailored to our needs. In [KA14], the model-based hand-object 3D tracking is formulated as an optimization problem, which seeks out the 3D object(s) pose and hand configuration that minimizes the discrepancy between hypotheses and actual observations. The optimization problem is solved based on PSO [ESK01].

Since this method estimates the generalized pose of a hand interacting with an object, it is straightforward to compute the 3D positions of the estimated fingertips in relation to the object's surface (i.e., contact points). Still, in our implementation of [KA14], we have incorporated one important modification. The original 3D hand-object tracking framework provides solutions that are compatible with visual observations and are physically plausible in the sense that the hand and the object do not share the same physical space (i.e., the hand does not penetrate the modeled volume of the object). However, occluded fingers may have different poses that respect the above constraints, making the estimation of contact points an under-constrained problem. To overcome this issue, we assume that contact points do not change significantly when they cannot be observed. Time and space coherency is thus enforced by penalizing solutions in which hidden contact points are far from their last observed position.

2.2.2 Numerical Differentiation for Kinematics

In theory, velocity and acceleration can be estimated by numerical differentiation of poses obtained from tracking. However, this process is highly dependent on two factors: (a) the

acquisition frequency of the RGB-D frames, and (b) the quality of the motion tracking. First, even a perfect tracking would result in poor velocity and acceleration estimates if performed over time steps far apart from each other, also depending on the way the hand moves. However, this is not a freely controllable parameter, as most commercial RGB-D cameras offer acquisition frame-rates capped between 30 and 60 fps. We present our results on a 30 fps SoftKinetic DepthSense 325 camera. Second, acceleration profiles occurring in manipulation tasks are naturally spiky (see for example Fig. 2.5). Therefore, numerical differentiation is challenging in that while the number of samples used for each derivative estimate must be sufficient to alleviate tracking errors, it must also be kept minimal to discern the sudden variations that acceleration profiles are subject to.

As an alternative to existing numerical differentiation methods, algebraic parameter estimation approaches [FSR03] led to a new class of derivative estimators called algebraic numerical differentiators [MJF09]. The tracking errors resulting from the employed model-based tracking framework seem to follow a Gaussian distribution, yet they are not independent of one another, which rules out the white noise formalism. Subsequently, and in order to keep the kinematics estimation process unbiased by the use of a particular tracking method, we implement the so-called minimal (κ,μ) algebraic numerical differentiators, which do not assume prior knowledge of the signal errors' statistical properties.

2.2.3 From Kinematics to Dynamics

We suppose the manipulated object subject to n_d non-contact forces $(\mathbf{F}_k^d)_{k\in[1,n_d]}$ applied at points $(\mathbf{P}_k^d)_{k\in\{1,\dots,n_d\}}$ (e.g., gravitation, electromagnetism). We consider them fully known based on the object's properties. We seek to estimate n_c contact forces $(\mathbf{F}_k^c)_{k\in[1,n_c]}$ applied at contact points with the hand or the environment $(\mathbf{P}_k^c)_{k\in[1,n_c]}$ that are obtained from tracking (Section 2.2.1). Using the object's kinematics as estimated in Section 2.2.2, its motion is governed by Newton-Euler equations. Therefore, the resulting net force $\mathscr{F}_c$ and torque $\boldsymbol{\tau}_c$ due to the contact forces are such that:

$$\begin{cases} \mathscr{F}_c = m\mathbf{a} - \mathscr{F}_d \\ \boldsymbol{\tau}_c = \mathbf{J}_\mathbf{q}\cdot\boldsymbol{\alpha} + \boldsymbol{\omega}\times(\mathbf{J}_\mathbf{q}\cdot\boldsymbol{\omega}) - \boldsymbol{\tau}_d, \end{cases} \tag{2.1}$$

with $\mathscr{F}_d$ and $\boldsymbol{\tau}_d$ the net force and torque due to non-contact forces, and $\mathbf{J}_\mathbf{q}$ the inertia matrix at orientation $\mathbf{q}$.

The contact forces are subject to friction, which we model using Coulomb's law. Let $\mathbf{n}_k$ be the unit contact normal oriented inwards the object at contact point $\mathbf{P}_k^c$. Let then $\mathbf{t}_k^x$ and $\mathbf{t}_k^y$ be two unit vectors orthogonal to each other and to the normal $\mathbf{n}_k$, thus defining the tangent

plane. Each contact force $\mathbf{F}_k^c$ is decomposed as follows:

$$\mathbf{F}_k^c = f_k \mathbf{n}_k + g_k \mathbf{t}_k^x + h_k \mathbf{t}_k^y, \tag{2.2}$$

With μ_k the friction coefficient at $\mathbf{P}_k^c$, Coulomb's law reads:

$$\left\| g_k \mathbf{t}_k^x + h_k \mathbf{t}_k^y \right\|_2 \leq \mu_k f_k, \tag{2.3}$$

which is a strict equality in the case of dynamic friction.

2.2.4 Nominal Forces From Cone Programming

We address the estimation of the minimal contact forces responsible for the observed motion (i.e., nominal forces) as a second-order cone program (SOCP) [LVBL98, BV04, BW07]:

$$\begin{aligned} \min \quad & \mathscr{C}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{P}\mathbf{x} + \mathbf{r}^T \mathbf{x} \\ \text{s.t.} \quad & \begin{cases} \left\| \mathbf{A}_j \mathbf{x} + \mathbf{b}_j \right\|_2 \leq \mathbf{c}_j^T \mathbf{x} + \mathbf{d}_j, \quad j = 1, \ldots, m \\ \mathbf{E}\mathbf{x} \leq \mathbf{f} \\ \mathbf{G}\mathbf{x} = \mathbf{h}. \end{cases} \end{aligned} \tag{2.4}$$

As we track the object and the user's hand, we can determine, at each timeframe, newly established and broken contacts, and also those that remain still and those that slide. Therefore, we are explicitly considering static and kinetic (i.e. dynamic) friction in the constraints formulation. With $n_{c,s}$ and $n_{c,k}$ the respective numbers of friction forces and n_c their sum, we construct the optimization vector as follows:

$$\begin{aligned} \mathbf{x} = (& f_1, g_1, h_1, \ldots, f_{n_{c,s}}, g_{n_{c,s}}, h_{n_{c,s}}, \\ & f_{n_{c,s}+1}, f_{n_{c,s}+2}, \ldots, f_{n_{c,s}+n_{c,k}})^T \end{aligned} \tag{2.5}$$

$\mathbf{x}$ is a vector of $3n_{c,s} + n_{c,k}$ elements. The SOCP formulation in Eq. (2.4) then allows the direct handling of Coulomb static friction as cone inequality constraints by defining, for each contact point $j = 1, \ldots, n_{c,s}$, matrices $\mathbf{A}_j$, $\mathbf{b}_j$, $\mathbf{c}_j$ and $\mathbf{d}_j$ such that:

$$\mathbf{A}_j \mathbf{x} + \mathbf{b}_j = \begin{pmatrix} g_j \\ h_j \end{pmatrix} \quad \text{and} \quad \mathbf{c}_j^T \mathbf{x} + \mathbf{d}_j = \begin{pmatrix} \mu_j f_j \end{pmatrix}. \tag{2.6}$$

Moreover, having each normal vector $\mathbf{n}_k$ oriented inwards the object, we formulate n_c linear inequality constraints such that $f_k \geq 0$. This is done by defining $\mathbf{E}$ as a $n_c \times (3n_{c,s} + n_{c,k})$

matrix and $\mathbf{f}$ as a n_c-element vector such that, for each contact point $j = 1, \ldots, n_{c,s} + n_{c,k}$:

$$\mathbf{E}_j \mathbf{x} = \begin{pmatrix} -f_j \end{pmatrix} \quad \text{and} \quad \mathbf{f}_j = \begin{pmatrix} 0 \end{pmatrix}, \tag{2.7}$$

with $\mathbf{E}_j$ and $\mathbf{f}_j$ the j-th rows of $\mathbf{E}$ and $\mathbf{f}$, respectively.

Equality constraints ensuring that the resulting contact force distribution explains the observed kinematics stem from Newton-Euler's equations, as combining Eq. (2.1) with contact force expressions from Eq. (2.2) directly yields six linear equations in $\mathbf{x}$. $\mathbf{G}$ and $\mathbf{h}$ are thus of respective size $6 \times (3n_{c,s} + n_{c,k})$ and 6×1, with rows 1 to 3 accounting for net force constraints, and rows 4 to 6 for net moment constraints.

As stated earlier in Section 2.2, there exists an infinity of possible force distributions for a given kinematics and set of contact points. We use the (squared) L^2 norm of the contact force distribution (i.e., the sum of squares of the individual components) as an indicator of the intensity of the grasp. We thus complete the SOCP with the following cost function:

$$\mathscr{C}_{L^2}(\mathbf{x}) = \sum_{k \in \mathscr{F}} \left[f_k^2 + g_k^2 + h_k^2 \right] = \sum_{k \in \mathscr{F}} \| \mathbf{F}_k^c \|_2^2, \tag{2.8}$$

where $\mathscr{F}$ is the set of contacting fingers. The objective function $\mathscr{C}_{L^2}$ allows to search for the optimal grasp in the L^2 sense, although other cost functions can be tested. Numerically, we formulate and solve the SOCP using the CVXOPT library for convex optimization [ADV13].

2.2.5 Reproducing Human Grasping Forces

Humans do not manipulate objects using nominal closures (i.e. minimal grasp forces). They tend to "over-grasp" and produce workless internal forces, i.e. firmer grasps than mechanically required. This human grasp property is described by considering finger forces as two sets [KR86, YN91]: nominal forces responsible for the object's motion, and internal forces that secure the object through a firm grip but do not affect the object's kinematics as they cancel each other out [MS85, MSZ94]. Studies showed that humans apply internal forces to prevent slip [JW84, FJ02] and control their magnitude to avoid muscle fatigue or damaging fragile objects [GZL10, PSZL12]. We extend the formulation of the SOCP to address such decompositions and construct a dataset on how humans apply internal forces when manipulating objects, extracted from tactile sensor measurements during real experiments.

Each finger force $\mathbf{F}_k$ is decomposed into a nominal component $\mathbf{F}_k^{(n)}$ and an internal component $\mathbf{F}_k^{(i)}$:

$$\begin{aligned} &\mathbf{F}_k = \mathbf{F}_k^{(n)} + \mathbf{F}_k^{(i)} \\ &\text{with} \quad \begin{cases} \mathbf{F}_k^{(n)} = f_k^{(n)}\mathbf{n}_k + g_k^{(n)}\mathbf{t}_k^x + h_k^{(n)}\mathbf{t}_k^y \\ \mathbf{F}_k^{(i)} = f_k^{(i)}\mathbf{n}_k + g_k^{(i)}\mathbf{t}_k^x + h_k^{(i)}\mathbf{t}_k^y. \end{cases} \end{aligned} \tag{2.9}$$

Although both forces are decomposed along the same contact frame $(\mathbf{n}_k, \mathbf{t}_k^x, \mathbf{t}_k^y)$ as in Eq. (2.2), note that nothing constraints $\mathbf{F}_k^{(n)}$ and $\mathbf{F}_k^{(i)}$ to be colinear. We subsequently redefine the optimization vector $\mathbf{x}$ by considering the nominal and internal components individually rather than their sum as in Eq. (2.5):

$$\begin{aligned} \mathbf{x} = (&f_1^{(n)}, g_1^{(n)}, h_1^{(n)}, f_1^{(i)}, g_1^{(i)}, h_1^{(i)}, \ldots, f_{n_{c,s}}^{(n)}, g_{n_{c,s}}^{(n)}, h_{n_{c,s}}^{(n)}, f_{n_{c,s}}^{(i)}, g_{n_{c,s}}^{(i)}, h_{n_{c,s}}^{(i)}, \\ &f_{n_{c,s}+1}, f_{n_{c,s}+2}, \ldots, f_{n_{c,s}+n_{c,k}})^T \end{aligned} \tag{2.10}$$

By definition, nominal forces are responsible for the object's motion through the Newton-Euler equations while internal forces are neutral regarding its state of equilibrium:

$$\begin{cases} \sum\limits_{k\in\mathscr{F}} \mathbf{F}_k^{(n)} = \boldsymbol{\mathscr{F}}_c, & \sum\limits_{k\in\mathscr{F}} \overrightarrow{\mathbf{GP}_k} \times \mathbf{F}_k^{(n)} = \boldsymbol{\tau}_c \\ \sum\limits_{k\in\mathscr{F}} \mathbf{F}_k^{(i)} = \mathbf{0}, & \sum\limits_{k\in\mathscr{F}} \overrightarrow{\mathbf{GP}_k} \times \mathbf{F}_k^{(i)} = \mathbf{0}. \end{cases} \tag{2.11}$$

Equation (2.11) provides a new set of constraints that we integrate into the SOCP of Section 2.2.4. Ensuring that the resulting distribution still obeys Coulomb's law of friction, we finally compute the distribution of nominal and internal forces that best match the tactile sensor measurements $(\widetilde{f}_k)_{k\in\mathscr{F}}$, using a new objective function:

$$\mathscr{C}_{\mathrm{d},\widetilde{f}_k}(\mathbf{x}) = \sum_{k\in\mathscr{F}} \left[\left\| \mathbf{F}_k^{(n)} \right\|_2^2 + \left(f_k^{(n)} + f_k^{(i)} - \widetilde{f}_k \right)^2 \right]. \tag{2.12}$$

The reason why we do not directly identify internal forces as the differences between the measurements $\widetilde{f}_k$ and the minimal forces resulting from the initial SOCP of Section 2.2.4 is that possible sensor measurement errors may lead them not to compensate each other. By integrating their computation into the SOCP, we ensure that the resulting internal forces $f_k^{(i)}$ bridge the gap between $f_k^{(n)}$ and measurements $\widetilde{f}_k$ without perturbing the object's observed kinematics. We illustrate the decomposition process in Fig. 2.2(a).

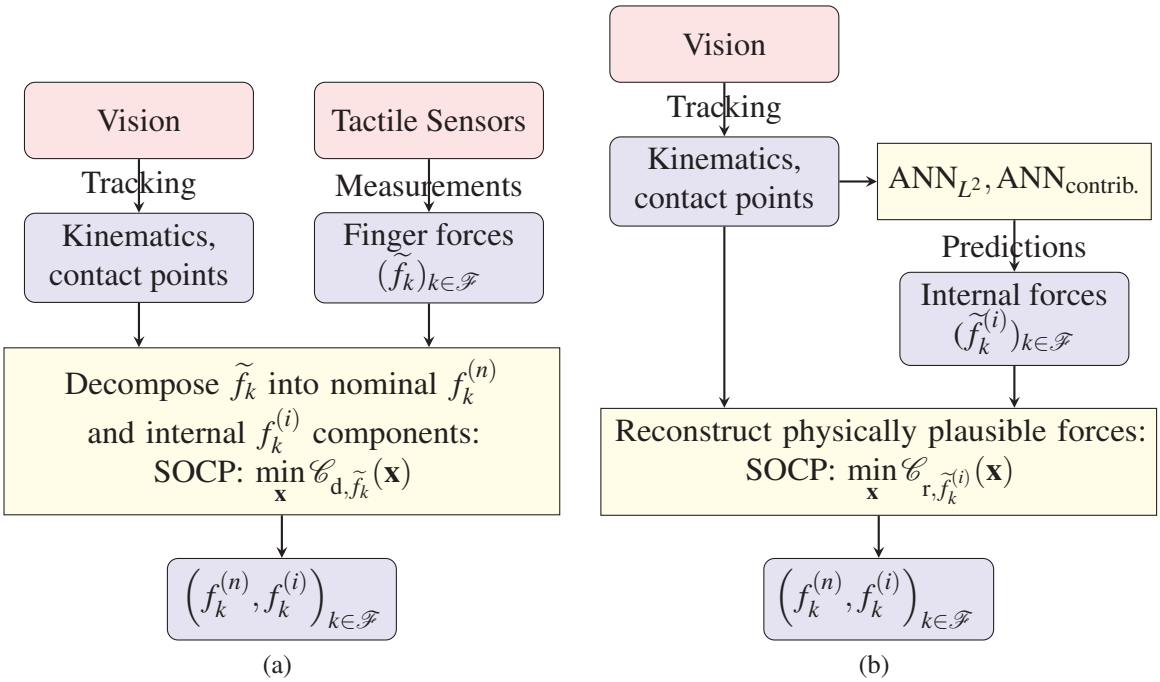


Figure 2.2: (a) Measurements from tactile sensors are used to estimate nominal and internal force decompositions from vision. (b) Full contact forces are reconstructed by combining ANN internal force predictions with an SOCP ensuring physical plausibility.

2.2.6 Learning Internal Force Distributions

Recent studies attempted to build mathematical models correlating grasp forces to kinematic data, yet limited to cyclic movement patterns and two-finger grasps [GLZ05, SLZ11], hence concealing the issue of static indeterminacy, i.e., the fact that in multi-contact, the knowledge of the motion does not suffice to completely characterize the underlying force distribution. In contrast, our approach learns how humans apply internal forces using artificial neural networks (ANN). We first construct an experimental dataset by having human operators manipulate an instrumented box (see Section 2.3) over tasks such as pick-and-place, lift and release, rotations, and unguided compositions of these. Experiments were conducted over a pool of six participants: three female (two right-handed, one left-handed), and three male (all right-handed) operators using their preferred hand on different contact and object mass configurations. Executing 160 manipulation experiments of approximately 10 s duration, we perform motion tracking and record the tactile sensor measurements to compute the best-matching decompositions $\left(f_k^{(n)}, f_k^{(i)}\right)$ following the SOCP of Section 2.2.5.

The next step is to learn the variations of internal forces $f_k^{(i)}$ with motion and grasping features. We select the learning parameters as those that directly impact the force distri-

butions through the Newton-Euler equations. Contact forces vary with the object's mass and acceleration, or more accurately including the contribution of gravity. We thus consider the target net contact force, which can be computed from the object's kinematics and the gravity vector $\mathbf{g}$ alone: $\boldsymbol{\mathscr{F}}_c = m \cdot (\mathbf{a} - \mathbf{g})$. We can consider this dependence as twofold: on the magnitude of $\boldsymbol{\mathscr{F}}_c$ itself, and on the relative orientation of $\boldsymbol{\mathscr{F}}_c$ with the contact normals, as in [GLZ05]:

$$p_1 = \|\boldsymbol{\mathscr{F}}_c\|_2 \tag{2.13}$$

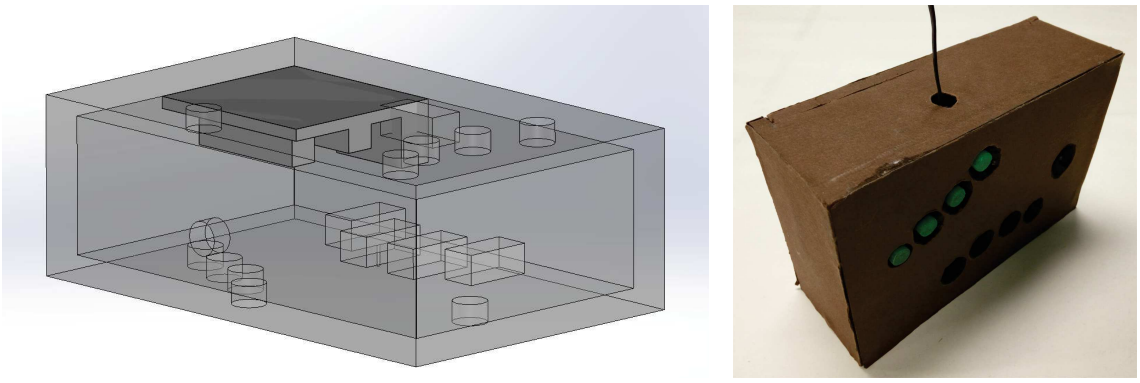
$$p_{2,k} = \mathbf{n}_k \cdot \mathbf{u}_{\boldsymbol{\mathscr{F}}_c}, \text{ with } \mathbf{u}_{\boldsymbol{\mathscr{F}}_c} = \frac{\boldsymbol{\mathscr{F}}_c}{\|\boldsymbol{\mathscr{F}}_c\|_2}. \tag{2.14}$$

Similarly, we consider the case of rotational kinematics through the magnitude of the net contact torque $\boldsymbol{\tau}_c$ of Eq. (2.1) and the individual torques each finger is able to generate:

$$p_3 = \|\boldsymbol{\tau}_c\|_2 \tag{2.15}$$

$$p_{4,k} = \left(\overrightarrow{\mathbf{GP}_k} \times \mathbf{n}_k\right) \cdot \mathbf{u}_{\boldsymbol{\tau}_c}, \text{ with } \mathbf{u}_{\boldsymbol{\tau}_c} = \frac{\boldsymbol{\tau}_c}{\|\boldsymbol{\tau}_c\|_2}. \tag{2.16}$$

Finally, we learn internal forces as a function of kinematics and grasp parameters $\left(p_1, (p_{2,k})_{k\in\mathscr{F}}, p_3, (p_{4,k})_{k\in\mathscr{F}}\right)$ using two ANNs: a first network, ANN_{L^2}, estimates the amount of internal forces applied, quantified as the L^2 norm of their distribution, while a second network, $\text{ANN}_{\text{contrib.}}$, jointly estimates the relative contribution of each finger in the grasp's intensity. The outputs of $\text{ANN}_{\text{contrib.}}$ are percentages constructed as the individual forces normalized with the overall L^2 norm. Note that, as that similar motions can stem from different force distributions, using a single ANN would mean linking similar inputs to highly varying individual forces. Yet, we observed that different grasp intensities still tend to be similarly shared among fingers, hence two ANNs to account for natural intensity variance but consistent decompositions. From the entire collected measurements, we only use as learning data those where the net forces computed from the observed kinematics and by summing up the tactile sensor measurements are within a specified threshold from each other. This allows us to avoid samples where visual tracking or tactile sensor measurements are unreliable, i.e., not compatible with each other with respect to the equations of motion. In our experiments, setting this threshold to $1.5N$ yields a final dataset of 8200 samples, which we partition into training and validation datasets to construct and assess different ANN configurations by cross-validation. Performing numerical resolution with the neuralnet package for statistical analysis software R [FGS12, R C14], we choose ANN_{L^2} and $\text{ANN}_{\text{contrib.}}$ with logistic neurons trained with resilient backpropagation and two hidden layers, with respectively 6 and 8 neurons in the first hidden layer, and 7 and 13 neurons in the second.



(a) CAD view: AHRS (upper plate) and five force sensors (cuboids) repositionable to assess different contact configurations (cylinders).

(b) Sensor configuration example (green) and other locations (holes).

Figure 2.3: Instrumented device for quantitative and qualitative evaluation.

		Central	Gaussian	Algebraic
Trans. acc. $[m \cdot s^{-2}]$	Avg.	-0.029	$\mathbf{-0.022}$	-0.024
	St.d.	1.686	1.627	**0.904**
Rot. vel. $[rad \cdot s^{-1}]$	Avg.	0.084	0.070	**0.052**
	St.d.	1.559	1.294	**1.241**

Table 2.1: Kinematics estimation errors (average and standard deviation) for central finite difference, Gaussian filtering, and algebraic filtering.

2.3 Experiments

In order to assess our approach, we perform manipulation experiments on a rectangular cuboid of dimensions 171mm $\times$ 111mm $\times$ 60mm. The simplified shape of this ground-truth object is chosen to meet sensing instrumentation constraints and offer several grasping possibilities. We instrument the box with two types of sensors. The first is an Xsens MTi-300 attitude and heading reference system (AHRS) motion sensor measuring reference rotational velocities and translational accelerations. Its purpose is to validate the numerical differentiation of tracking data by algebraic filtering, see Section 2.2.2. The second consists of five Honeywell FSG020WNPB piezoresistive one-axis force sensors that can be positioned at different predefined grasp spots on the box. We depict the instrumented device in Fig. 2.3. We evaluate the contact forces estimated from the SOCP in Section 2.2.4 with the force sensor measurements in terms of: (i) normal forces per finger, (ii) resulting net force, and (iii) sum of squares. We summarize the validation protocol in Fig. 2.4.

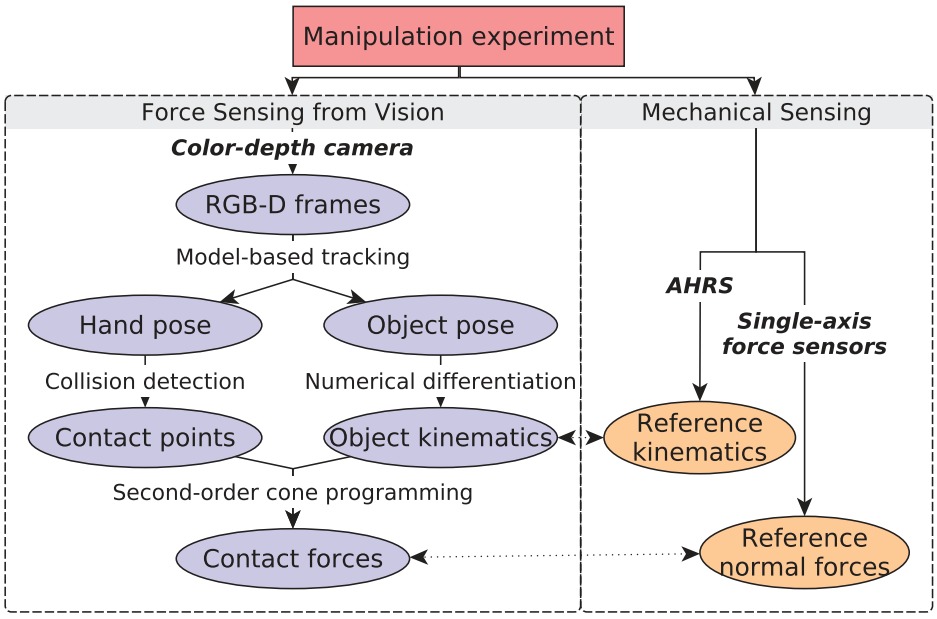


Figure 2.4: Validation protocol.

2.3.1 Kinematics From Vision vs AHRS

We assess the validity of our approach by executing motions emphasizing each of the three coordinates of both translation accelerations and rotational velocities, and comparing the kinematics estimated from vision to measurements from the Xsens MTi-300 AHRS. Statistical analysis of the estimation errors shows that algebraic numerical differentiation is well suited for kinematics estimation (see Table 2.1). Though on translational acceleration, its average error is slightly higher than with Gaussian filtering, its variance is also considerably lower. Its performance on rotational kinematics is also the best of all three tested approaches. We illustrate the results of the six-axis experiments in Fig. 2.5.

2.3.2 Nominal Forces From Vision-Based Kinematics

We now validate our vision-based force estimation framework using normal force sensors placed at pre-specified positions over the instrumented box. As a first validation step, contact points obtained from vision were compared to the expected contact points based on the sensors' locations and resulted in estimation errors of mean $-1.55mm$ and standard deviation $6.13mm$. Furthermore, we assessed the sensitivity of FSV to these uncertainties by comparing the force distributions obtained using either the contact points from vision or the tactile sensor positions. We found that FSV is relatively robust to such estimation errors, resulting in force uncertainties of mean $0.216N$ and standard deviation $1.548N$. Therefore,

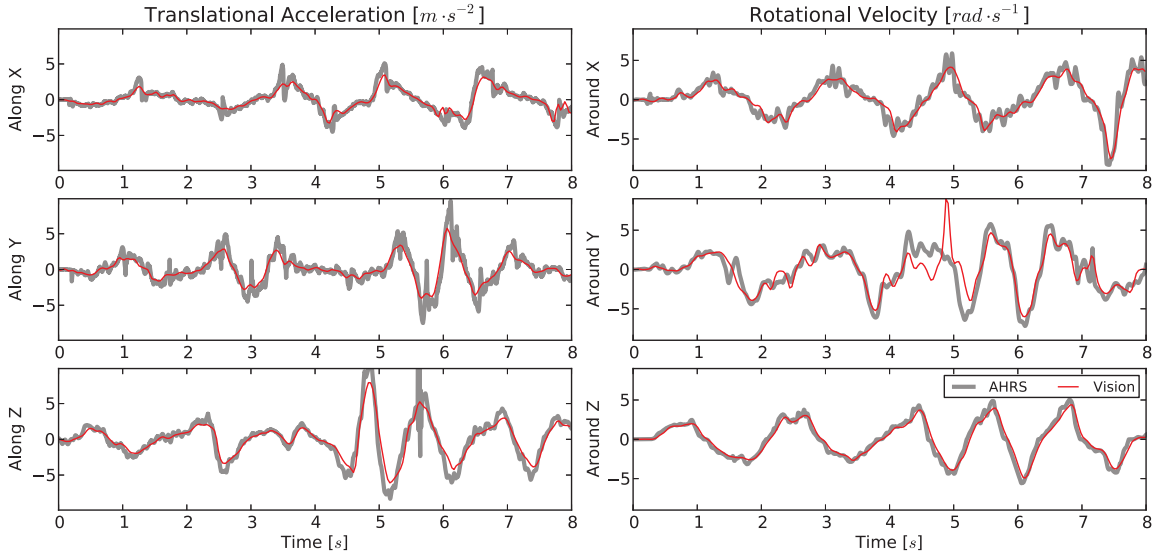


Figure 2.5: Comparison between vision-based kinematics and AHRS-embedded accelerometer and gyroscope.

we rely solely on vision-based kinematics and contact points for the rest of this work. When performing experiments, we also observed that the force applied by the pinky finger was consistently below the sensitivity threshold of our force sensors, hence we present our results on four-finger experiments. Still, as the force distribution problem introduced in Section 2.2 becomes statically indeterminate from three fingers, using four fingers maintains some high indeterminacy in the force distribution problem, and thus preserves the generality of our results. We represent the force sensor measurements along with FSV's outputs in Fig. 2.6.

As mentioned in Section 2.2.4, the comparison of the normal components from vision and from tactile sensors shows that the latter's measurements are overall greater. This illustrates the fact that humans seize objects harder than the required force closure, in contrast with the L^2-optimal grasp estimated from vision, which is visible in the sum of squares plot. Still, the resulting net forces are matching well, which demonstrates that FSV can successfully capture the object's motion characteristics and compute a force distribution that physically explains the observed kinematics.

2.3.3 Reconstructing Full Contact Force Distributions

By recording new manipulation experiments, we extract the kinematics and grasping parameters described in Section 2.2.6 over time $\left(p_1, (p_{2,k})_{k\in\mathscr{F}}, p_3, (p_{4,k})_{k\in\mathscr{F}}\right)$ and use the trained ANNs to predict the internal forces the human operator most likely apply throughout the experiment, $\left(\widetilde{f}_k^{(i)}\right)_{k\in\mathscr{F}}$. We finally construct the full contact force distributions using

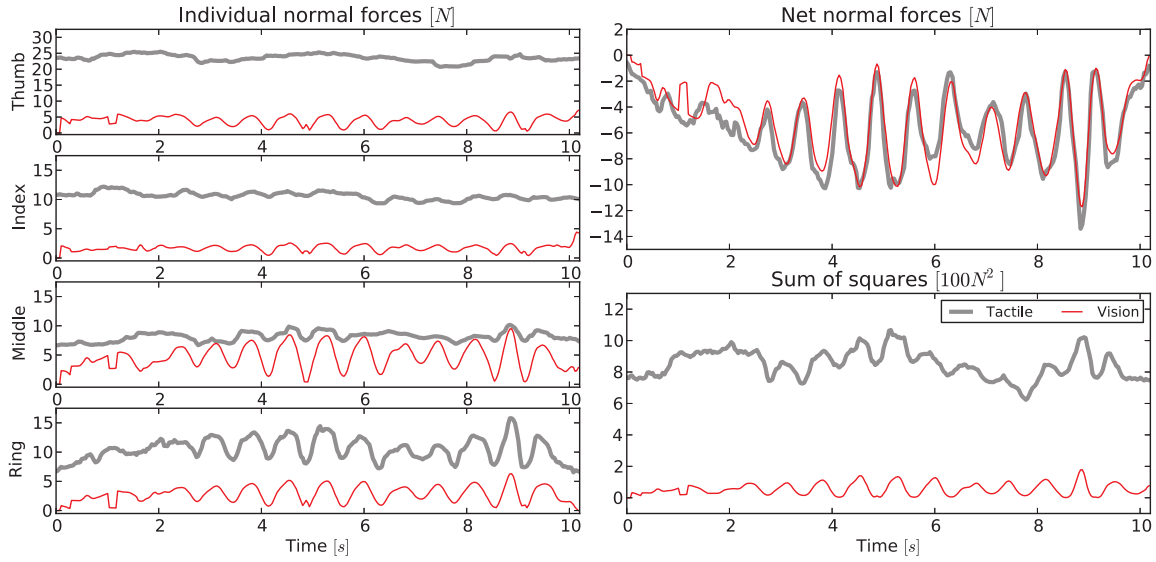


Figure 2.6: Contact forces from vision based on L^2 criterion are individually lower than tactile sensor measurements but result in the same net force.

a variant of the SOCP described in Section 2.2.5. For this new purpose, we formulate an objective function that aims at reconstructing full contact forces by minimization of, on one hand, the nominal components in isolation, and on the other hand, the discrepancy between optimal internal components and ANN predictions $\left(\widetilde{f}_k^{(i)}\right)_{k\in\mathscr{F}}$:

$$\mathscr{C}_{\mathrm{r},\widetilde{f}_k^{(i)}}(\mathbf{x}) = \sum_{k\in\mathscr{F}} \left[\left\| \mathbf{F}_k^{(n)} \right\|_2^2 + \left(f_k^{(i)} - \widetilde{f}_k^{(i)} \right)^2 \right]. \tag{2.17}$$

We illustrate the final estimation process in Fig. 2.2(b). By feeding the ANN internal force predictions into the SOCP, we ensure that the final internal forces $\left(\mathbf{F}_k^{(i)}\right)_{k\in\mathscr{F}}$ are not only consistent with natural grasping patterns but also physically correct and do not impact the object's observed kinematics through the resulting net force, as shown in Fig. 2.7.

2.3.4 Robustness Analysis

We investigate the robustness of our approach to features that do not appear in the training dataset. To this end, we train another instance of the ANNs described in Section 2.2.6, not over the entire dataset but on a partial subset relative to a single operator, on a single grasp pose, and a single mass configuration. We then evaluate the resulting ANNs on datasets obtained with another user, another grasp, and/or a 10% mass increase. We report the relative

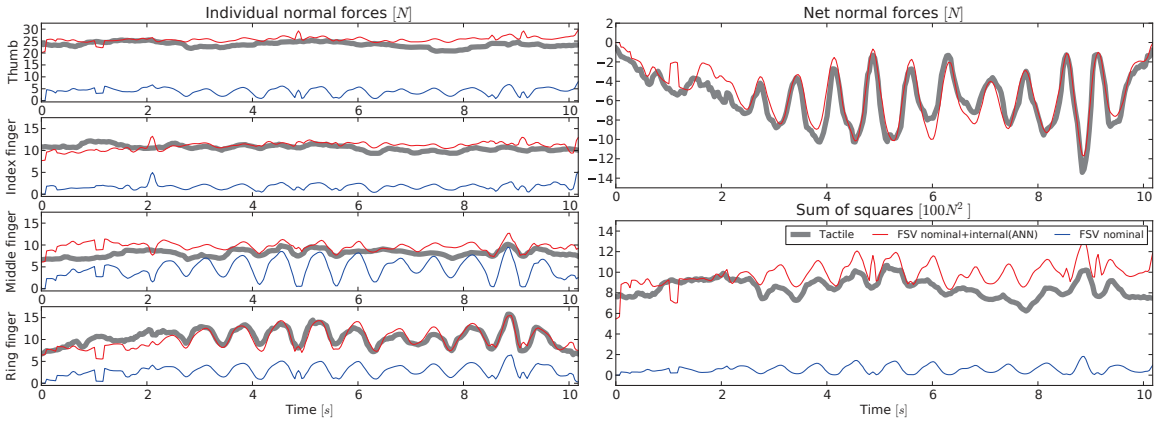


Figure 2.7: Artificial neural networks used in conjunction with cone programming successfully predict force distributions that both explain the observed motion and follow natural human force distribution patterns.

errors with respect to the tactile sensor measurements in Table 2.2 along with reference results from fully-trained ANNs.

First, it appears that ANNs trained over a single operator may be generalized to other users with no significant performance decrease, which suggests that humans tend to apply internal forces following similar patterns. Second, reasonable changes in mass do not seem to significantly impact the estimation accuracy either. This is allowed by the fact that in our problem formulation, mass is not a training variable by itself but is implicitly taken into account through the product $\mathscr{F}_c = m \cdot (\mathbf{a} - \mathbf{g})$. Under this formalism, manipulating a heavy object with a given kinematics is analogous to manipulating a lighter object with a higher acceleration. Therefore, the ANNs may accomodate mass changes provided that they were trained over a sufficient variety of kinematics. In the end, ANNs seem most sensitive to grasp pose changes. This may be explained by the fact that placing fingers differently may substantially change their synergies. Still, the performance decrease remains reasonable while force distributions, by construction, still explain the observed motion. Eventually, the main sensitivity to grasp poses is comforted by the fact that also changing user and mass does not decrease the estimation accuracy further.

2.4 Grasp Recovery by Force Optimization

As an application example, we now show that FSV can be used, along with grasp taxonomies, as an implicit force model to reconstruct physically realistic manipulation sequences from possibly incomplete visual observation and inaccurate visual tracking pose estimates.

			Part. training		Full training	
User	Mass	Grasp	Avg. [%]	St.d. [%]	Avg. [%]	St.d. [%]
◯	◯	◯	10.7	12.4	9.71	12.0
×	◯	◯	10.9	12.3	10.3	11.8
◯	×	◯	10.8	11.3	10.4	12.4
◯	◯	×	14.6	14.5	10.9	11.3
×	×	×	14.9	14.8	9.94	12.6

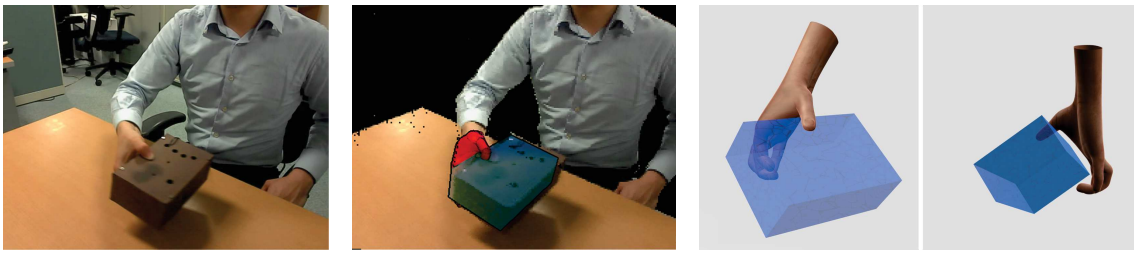
Table 2.2: Relative force estimation errors based on the exhaustivity of the training dataset. ◯ and × indicate features that respectively appear or not in the partial training dataset.

2.4.1 Initializing Reference Grasps

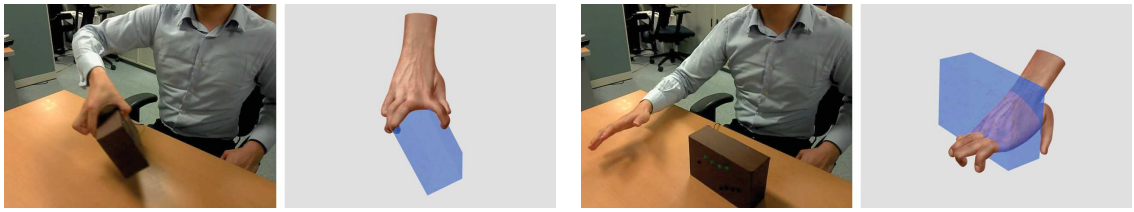
Manipulating an object with static contact points, we monitor the scene using a single RGB-D sensor and track the object and the hand jointly. Two common issues commonly arose during our experiments. First, tracking a manipulation scene sometimes led to mutual confusion, i.e., the object was mistaken for the hand, or conversely. Second, in some situations, self-occlusions produced physical incoherent hand-object poses despite visual consistency from the camera's point-of-view. We illustrate such mis-tracking examples in Fig. 2.8. In order to identify the real grasp being applied, we propose to optimize the force distribution in the vicinity of the hand pose estimate from tracking, taken as an initial guess rather than an absolute reference.

As described in Section 1.3.1, we represent the hand pose as a 27-parameter vector $\mathbf{H} = \left(\boldsymbol{\theta}^{\text{palm}}, \boldsymbol{\theta}^{\mathscr{F}}\right)$, with $\boldsymbol{\theta}^{\text{palm}}$ a 7-parameter vector encoding the palm pose (3D position and quaternion) and $\boldsymbol{\theta}^{\mathscr{F}}$ a 20-parameter vector encoding the finger poses (4 joint angles for each of the five fingers). With no prior knowledge on the instant the tracker loses the target, we initialize the pose search when contact occurs. This instant may be accurately obtained from the tracking of the object alone, i.e., without relying on the accuracy of the hand tracking and collision detection. Indeed, from the perspective of physics-based optimization, it can be implicitly defined as the moment at which the observed object kinematics cannot be explained by contacts with the environment alone, i.e., when the SOCP has no solution. We denote by $\widetilde{\mathbf{H}} = \left(\widetilde{\boldsymbol{\theta}}^{\text{palm}}, \widetilde{\boldsymbol{\theta}}^{\mathscr{F}}\right)$ such an initial hand pose from tracking.

The grasp taxonomy of [FbSRK09] describes a set of 17 grasp poses categorized by function and geometry (e.g., power grasp, large diameter). From it, we select a subset $\mathscr{P}$ of $N_{\mathscr{P}}$ grasp poses suitable to manipulate the object. In our experiments, we chose three grasps compatible with the instrumented device, e.g., firmly securing it from opposite sides or dexterously manipulating it from the corner, as illustrated in Fig. 2.9. We characterize



(a) Input color image. (b) Hand-object pose estimates. (c) Synthesized back and side views.



(d) Real thumb mistaken for thumb and index. (e) Lost tracking: hand-object interpenetration.

Figure 2.8: (a) Visible by the camera, (b) palm and thumb are successfully recognized. (c) However, the occluded finger poses are physically impossible as none hold the object. The accumulation of tracking errors can lead to (d) implausible and even (e) impossible poses.

each reference pose of the taxonomy $P \in \mathscr{P}$ by their finger poses $\boldsymbol{\theta}^{\mathscr{F},P}$. We finally initialise our search space with reference hand poses $\mathbf{H}^P$ specified as the union of the palm pose from tracking with finger configurations from the grasp taxonomy:

$$\forall P \in \mathscr{P}, \quad \mathbf{H}^P = \left(\widetilde{\boldsymbol{\theta}}^{\text{palm}}, \boldsymbol{\theta}^{\mathscr{F},P}\right) \tag{2.18}$$

We thus define an initial set of primitive hand poses $\left(\mathbf{H}^P\right)_{P \in \mathscr{P}}$.

2.4.2 Generating New Grasp Poses

With $\boldsymbol{\sigma}$ a 27-element vector of standard deviations for the hand pose parameters, we now construct new grasp candidates $\widetilde{\mathbf{H}}^P$ through Gaussian random sampling $\mathscr{N}(\cdot, \boldsymbol{\sigma})$ in the vicinity of each primitive hand pose $\mathbf{H}^P$:

$$\forall P \in \mathscr{P}, \quad \widetilde{\mathbf{H}}^P = \mathscr{N}(\mathbf{H}^P, \boldsymbol{\sigma}) \tag{2.19}$$

For each sampled grasp candidate $\widetilde{\mathbf{H}}^P$, we compute the 3D hand pose by forward kinematics. We then check for interpenetrations between the hand and the object using the SWIFT++ library for collision detection [EL01]. For each primitive composing the hand model, we consider that a contact occurs when it is within a chosen threshold to the object (in our

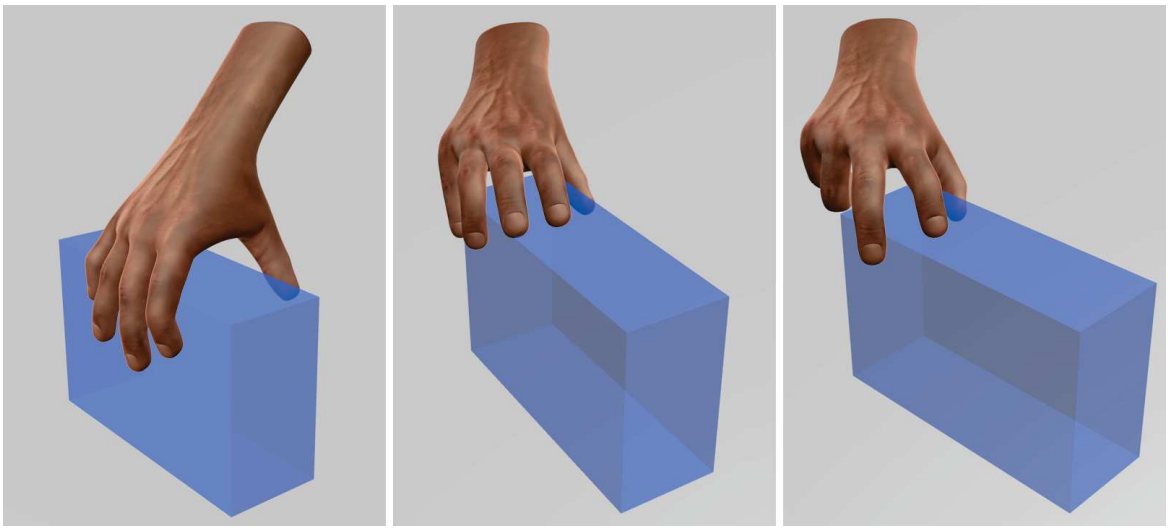


Figure 2.9: Reference grasps from left to right: large diameter, precision sphere and tripod.

experiments: 5 mm). If, for any primitive, penetration exceeds this threshold, the current grasp pose candidate $\widetilde{\mathbf{H}}^P$ is discarded from future consideration and a new one is sampled following Eq. (2.19).

Next, we ensure that the resulting contact points allow the manipulation of the object following the tracked object trajectory. We do so by formulating the SOCP of Section 2.2.4 with the contact locations estimated from collision detection and ensuring that, for each time step, with contacts fixed at the estimated locations, there exists a nominal force distribution $\left(\mathbf{F}_k^{(n)}\right)_{k\in\mathscr{F}}$ explaining the observed object kinematics through the equations of motion. If the grasp is not physically able to cause the desired motion, the grasp pose candidate is discarded. If it is, we predict the internal forces $\left(\mathbf{F}_k^{(i)}\right)_{k\in\mathscr{F}}$ humans are most likely to apply throughout the motion with the neural networks of Section 2.2.6. Finally, we denote by $(\mathbf{F}_k)_{k\in\mathscr{F}}$ the complete force distribution obtained by combining nominal and internal components in the FSV framework. Finally, we quantify the grasp intensity throughout the manipulation sequence by computing the L^2 norm of the total distribution, averaged over the duration of the experiment (N_S samples):

$$w\left(\widetilde{\mathbf{H}}^P\right) = \frac{1}{N_S}\sum_{j=1}^{N_S}\sqrt{\sum_{k\in\mathscr{F}}\|\mathbf{F}_k\|_2^2}. \tag{2.20}$$

In addition, we introduce an indicator $d_{\widetilde{\mathbf{H}}}$ on the discrepancy between the initial hand pose from tracking $\widetilde{\mathbf{H}}$ and the grasp candidate $\widetilde{\mathbf{H}}^P$, e.g., the L^2 distance :

$$d_{\widetilde{\mathbf{H}}}\left(\widetilde{\mathbf{H}}^P\right) = \left\|\widetilde{\mathbf{H}}^P - \widetilde{\mathbf{H}}\right\|_2. \tag{2.21}$$

Finally, we consider a weighted cost function $\mathscr{C}_{\widetilde{\mathbf{H}}}$ combining w and $d_{\widetilde{\mathbf{H}}}$:

$$\mathscr{C}_{\widetilde{\mathbf{H}}}\left(\widetilde{\mathbf{H}}^P\right) = w\left(\widetilde{\mathbf{H}}^P\right) + \alpha \cdot d_{\widetilde{\mathbf{H}}}\left(\widetilde{\mathbf{H}}^P\right). \tag{2.22}$$

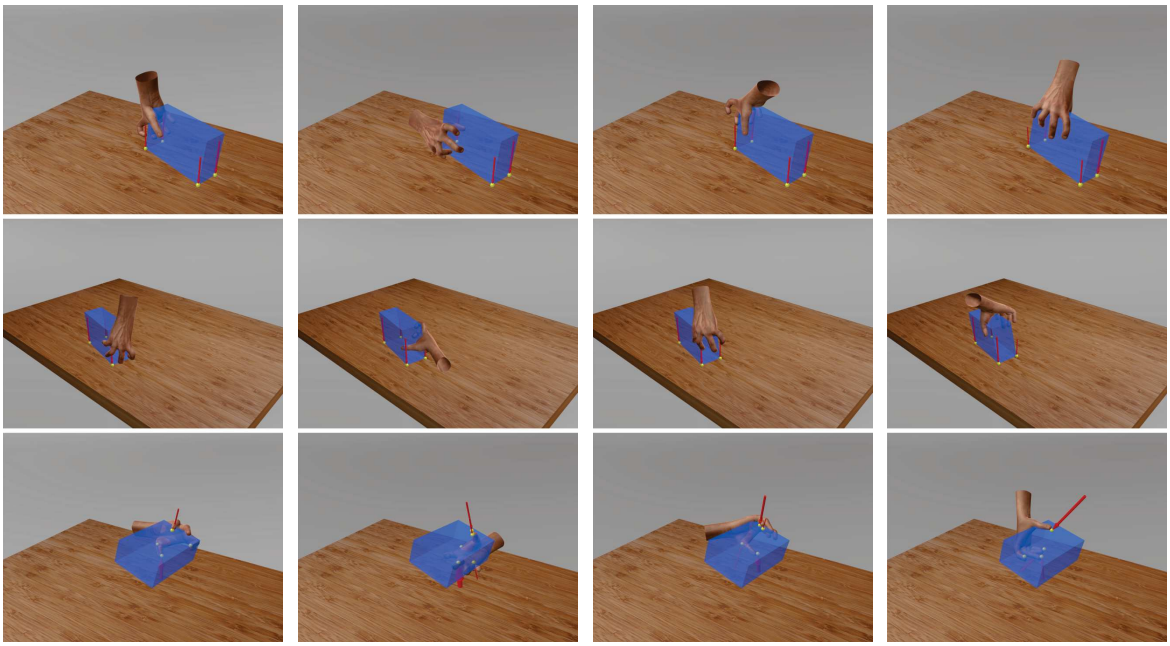
The weight α can be tuned to favor either the grasping energy along the observed trajectory or the fidelity with respect to the pose estimate from tracking. We subsequently present results for different values of α.

2.4.3 Results

In order to assess the validity of our approach, we consider a manipulation experiment involving a box being rapidly moved in arbitrary directions, both in translation and in rotation. Due to mutual occlusions between the hand and the object, at the beginning of the sequence, the pose estimates from model-based tracking match the observable features but are not physically plausible (see Fig. 2.8(a-c)). Throughout the rapid motion, the accumulation of tracking uncertainties results in completely wrong pose estimates (see Fig. 2.8(d-e)).

Using the numerical differentiation framework introduced in Section 2.2.2, we rely on the good estimation of the box pose to compute its kinematics over time. As described in Section 2.4.1, we combine grasp taxonomies and tracking data to initialize the search space. Following Section 2.4.2, we generate pose hypotheses by Gaussian sampling in the vicinity of the reference hand poses and verify their feasibility regarding interpenetrations and compatibility with the motion. In our experiments, we sampled 792 grasp candidates that were valid with respect to collision constraints in 10 min of computation time (Intel i7-4700MQ processor, single-threaded implementation). Rather than sampling the configuration space and checking for collisions, a possibly more efficient approach could be to sample the contact space, as explored in [WMZ+13]. We finally fed each grasp candidate into the FSV framework, yielding 493 grasp poses that were able to achieve the observed motion.

For each valid pose, we computed the grasp intensity w and discrepancy $d_{\widetilde{\mathbf{H}}}$ costs of Eqs. (2.20) and (2.21). Finally, we combined these values into the weighted cost $\mathscr{C}_{\widetilde{\mathbf{H}}}$ of Eq. (2.22) for different values of the normalization factor α. We depict the resulting optimal grasps in Fig. 2.10. As expected, minimizing the grasp intensity in priority (i.e., α small) yields grasps that are possibly far from the initial guess. Convertly, favorizing poses that are



(a) $\alpha = 0, w = 188.46$ (b) $\alpha = 1 \cdot 10^3, w = 236.86$ (c) $\alpha = 2 \cdot 10^3, w = 311.80$ (d) $\alpha = 3 \cdot 10^3, w = 349.19$

Figure 2.10: Each column represents the optimal solution yielded by our algorithm for increasing values of parameter α. The two first rows show the grasp candidate at the beginning of the experiment (front and back views). The third row corresponds to the same instant as the frame depicted in Fig. 2.8a. We can thus reconstruct various physically plausible grasps, that become closer to the initial observations as we increase α.

close to the tracking hypothesis (i.e., α big) allows the recovery of physically realistic grasps that match the actual observations despite inaccurate tracking data.

2.5 Summary and Discussion

Our work establishes that a single RGB-D camera can be used to capture interaction forces occurring in rigid object manipulation by a human hand without the need for visual markers or dedicated force sensing devices. Force sensing from vision is a novel and important contribution since it circumvents the intrusive instrumentation of object, environment and hands. Its exploitation can expand to the robotics field for daily on-line human activities monitoring, serving various purposes such as imitation learning.

Our method is validated with several experiments based on ground truth data and is able to estimate fairly accurately the force distributions applied during actual manipulation experiments. Although we confirmed that tracking noise is well mitigated by algebraic filtering, which produces truthful pose derivative estimates, guessing the hand-object contact

points under strong occlusions remains a challenging, open problem in computer vision. We achieved this by using a state-of-the-art model-based tracking method under the somewhat practical assumption that occluded fingers remain at their last observed position until they are visible again. While this assumption is fairly valid in numerous interesting cases, it is not true when considering tasks such as dexterous manipulation with finger repositioning or sliding. Still, this limitation does not call into question the force estimation framework *per se*, and could be alleviated by extending the markerless tracking method to multi-camera inputs, which would remain non-intrusive and keep an edge over tactile sensors regarding usability and cost.

With respect to computational performance, SOCP and internal force predictions are performed in real-time, and only hand-object tracking is computationally expensive. Given the recent developments on GPGPU implementations of hand-object tracking [KA14], our framework could be employed in real-time applications. This, combined with our reliance on a single camera, makes FSV suitable for daily observation and learning. Still, our approach is generic enough to accommodate any advance to the topic of 3D hand tracking and could be seamlessly extended to other methods, for instance when non real-time performance and a heavier setup are possible. Conversely, our framework could also be used as an implicit force model for physics-based tracking and motion editing, as human-like forces could augment the pose search with biomechanical considerations such as muscle fatigue or energy expenditure. We demonstrated such a use case to reconstruct physically plausible grasps in the presence of strong occlusions, which could also be incorporated in the tracking itself for interactive, physics-based correction.

Towards estimating contact forces from vision, we tackled the issue of static indeterminacy by applying machine learning techniques to internal forces. Rather than predicting new force distributions based on past observations, an alternative approach would be to formulate the evolution of the full contact forces following various objects and grasp taxonomies as an inverse optimal control problem. If invariants are found, they could be used to refine the cost function, which could result in more reliable contact forces than the nominal distributions computed by minimization of the grasp's L^2-norm. Extending the ground truth force measurement setup with embedded three-axis or force-torque miniature sensors would also benefit both learning and optimal control approaches. Further work could also address the case of surface contact models in place of point contacts (as the fingertip is deforming), namely for dexterous manipulations, or make use of synergy properties of the hand for bimanual tasks. Finally, combining our approach with visual SLAM or automated camera calibration methods would allow it to be deployed in unknown, varying environments, e.g. on mobile robots.

Acknowledgements

The work presented in this chapter was partially supported by the FP7 EU RoboHow.Cog project. It resulted in a conference paper [PKQA15b] and a workshop paper [PKQA15a], in collaboration with Ammar Qammaz from FORTH-ICS and Antonis A. Argyros from FORTH-ICS and the University of Crete.

Chapter 3

Hand-Object Contact Force Estimation From Markerless Visual Tracking

3.1 Introduction

Touch (i.e. physical contact) is of fundamental importance in the way we naturally interact with objects and in our perception of their physical and functional properties. Human manipulation remains little understood at the level of the underlying interaction forces, which are traditionally measured using force transducers. The latter are costly, cumbersome, and intrusive on both the object and the human haptic sense. Moreover, if mounted onto the hand, they often hinder or reduce the range of possible motions. Recent work has showed how the latter could be inferred from vision [GKD09, ZZCZ15, YYFA16]. Moreover advances in markerless visual tracking opened up the possibility for monitoring hand-object motions in a non-intrusive fashion. Computer vision techniques would therefore be an ideal substitute for current force sensing technologies.

This is an extremely challenging perspective. Indeed, tracking a hand interacting with an object is difficult due to strong mutual occlusions. Moreover, even when a manipulation trajectory is fully known, the force estimation problem is ill-posed or indeterminate in multi-contact. Indeed, given the physical properties of the object, there generally exists an infinity of force distributions resulting in the same motion (e.g. using different grip strengths– i.e. internal workless forces). While it is possible to compute physically plausible force distributions, capturing the real forces being applied is an open problem explored in multiple fields (see Section 1.4). In particular, kinesiology research has resulted in successful attempts at modeling grip forces by inverse optimization, e.g., during static prehension [NTLZ12] or two-finger circular motion [SLZ11]. Although these scenarios are of limited scope, this

suggests that it may be possible to construct a general model on human grasping, provided a rich dataset on manipulation kinodynamics (motion and forces).

In our work, we show that physics-based optimization can be used in conjunction with learning to capture manipulation forces from non-intrusive visual observation, on a setup as simple as a single RGB-D camera.

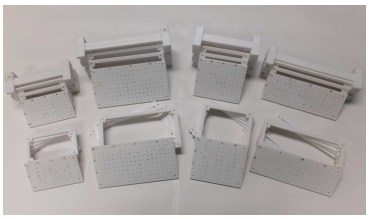
- We construct the first large-scale dataset on human manipulation kinodynamics, containing 3.2 hours of high-frequency measurements for 193 different object-grasp configurations (Section 3.2).
- We propose a force estimation framework that relies simultaneously on a recurrent neural network to predict forces that are consistent with the way humans naturally manipulate objects, and on a second-order cone program guaranteeing the physical correctness of the final force distribution (Section 3.3).
- We thoroughly validate our approach on ground-truth measurements (Section 3.4) and show that it can seamlessly be extended to visual tracking (Section 3.5).

Due to instrumentation constraints, our dataset is dedicated to constant contacts on prismatic grasps, i.e., with the thumb in direct opposition to the antagonist fingers. We discuss these limitations and show that the dual optimization-learning framework can still address scenarios beyond the focus of our study (Section 3.6). Finally, we discuss thoroughly the current limitations, possible extensions and applications of our work (Section 3.7). A preliminary version of this research, focusing on estimating normal forces from vision, was presented in Chapter 2 and appeared in [PKQA15b]. Our current study extends the latter idea and includes: an improved formulation of the optimization and learning models accounting for individual normal and tangential components, time-coherent manipulation forces, as well as algorithmic descriptions and extensive validation experiments that have not been presented before. To foster the research in this new topic, we make the manipulation kinodynamics dataset publicly available[1].

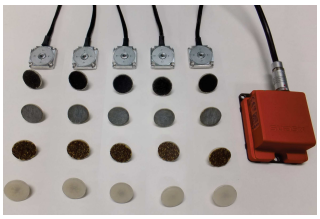
3.2 Manipulation Kinodynamics Dataset

Over the last years, the release of public datasets has massively benefitted the research in fields related to this work, such as object recognition and scene understanding [KAJS11, LBF14], whole-body and hand tracking [SPSS12, TSLP14], and robotic grasping [SDN08, ÇWS+15]. In contrast, datasets viewing human manipulation not only from the angle of vision but also

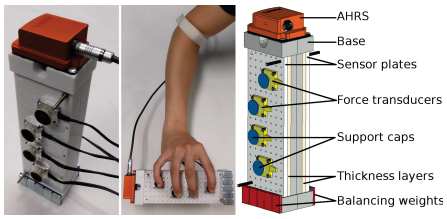
[1] https://github.com/jrl-umi3218/ManipulationKinodynamics.



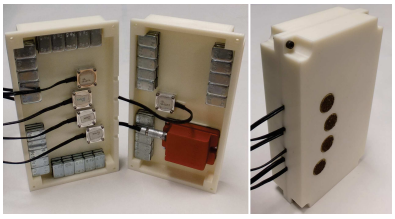
(a) AHRS base, thickness layers, sensor plates for repositionable transducers (four sizes).



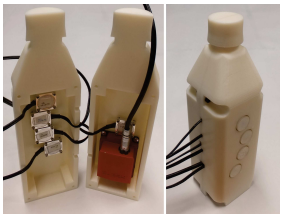
(b) 3D force transducers, support caps of various frictional characteristics, AHRS.



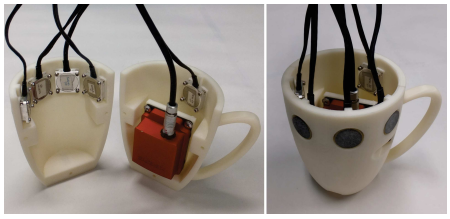
(c) Assembled instrumented device. The cables are tied to the subject's elbow to minimize force perturbations.



(d) 3D-printed box-shaped device with extra mass.



(e) Bottle-shaped device.



(f) Mug-shaped device for non-prismatic, spherical grasp.

Figure 3.1: We collect the manipulation kinodynamics dataset using dedicated instrumented devices of adjustable shape, friction, mass distribution and contact configuration (a-c). Additionally, we construct devices based on everyday objects, instrumented so as to allow intuitive interactions (d-f).

of touch have been more scarce so far. A notable example is the interaction capture technique of [KP06] for joint compliance estimation in graphics and synthesis of interaction animations. In this section, we introduce a new, extensive dataset dedicated to the kinodynamics of human manipulation.

3.2.1 Experimental Setup

Our objective is to construct a general force model capable of capturing the whole range of manipulation forces that are commonly applied during daily activities. The manipulation kinodynamics dataset was thus collected for diversity and genericity, regarding both the objects being manipulated and the way they are grasped. While using real objects may initially seem ideal, instrumenting them with force and motion sensors is impractical and makes it difficult and lengthy to collect a diverse dataset. Additionally, physical properties of arbitrary objects (e.g., inertia matrices) are seldom publicly available and must therefore be manually identified [SL01, BSPK02]. Finally, the instrumentation may result in measured forces that substantially differ from those that would have been applied on the original objects.

We address these caveats with dedicated instrumented devices, pictured in Fig. 3.1, composed of two symmetric parts for the thumb and the antagonist fingers. Each half consists of a base serving as support for an attitude and heading reference system (AHRS, Xsens MTi-300), and a sensor plate on which 3D precision force transducers (Tec Gihan USL06-H5-50N) can be positioned by 8 mm steps on the surface. Thickness layers can be inserted in between to increase the grasp width by 5 mm increments, bringing the total grasp width range between 46 mm and 86 mm. The force transducers are fitted with support caps of different surface textures: PET, sand paper of grit 40 (coarse), 150 (fine) and 320 (extra fine). The mass distribution can be adjusted with balancing weights inside and on the surface of the instrumented device. We 3D-print four sets of instrumented modules, with sensor plates of dimensions 80×152, 56×152, 80×96 and 56×96 mm^2. This setup allows the efficient collection of force and kinematics measurements under diverse grasp poses, friction conditions and mass distributions, obtained from the CAD models of the individual components.

Still, instrumentation constraints make it difficult to collect ground-truth measurements for arbitrary object shapes and grasps [FRS+16], which we consider essential to also prove the validity of any force prediction approach. Indeed, it would require a significantly heavier experimental setup to allow the individual adjustment of degrees of freedom such as local curvatures and finger repositioning. Note that these limits only apply to the dataset and not to the force estimation framework itself, which can still produce physically correct force distributions for such scenarios, although possibly different from the real forces being applied. We discuss these limitations and apply our algorithm to manipulation scenarios beyond the explicit scope of our study in Section 3.6.

3.2.2 The Dataset

Eleven right-handed volunteers, three females and eight males, took part as subjects in our experiments. Each subject was instructed to perform series of up to eight manipulation sequences as follows. For each series, the subject is given an instrumented box of randomly picked shape, thickness and surface texture as described in Section 3.2.1. The initial object configuration is completed by mounting the AHRS either at the top or at the bottom of the instrumented device, and at random with an additional 400 g mass inside. The subject is then instructed to perform manipulation tasks on eight variations of the initial configuration. Before each trial, the force transducers are placed on the box according to the subject's preferred grasp pose and their signals are adjusted following the manufacturer's recommended acquisition and calibration procedure. Each trial consists in the subject grasping the object and manipulating it for approximately 60 s. Every 10 s, in order to ensure the diversity

of the kinematics and forces present in the final dataset, the subject is given randomly picked instructions on speed, direction and task (e.g., slow forward pouring motion, fast left and right oscillations). After each trial, a 50 g balancing weight is attached to a randomly picked side, excluding sensor plates. Throughout the eight trials, we measure the effect of mass variations between 0 g and 350 g or 400 g and 750 g with the additional internal mass, arranged differently across series. Finally, the subject can interrupt the series whenever the object becomes uncomfortable to manipulate.

Overall, we collect motion and force measurements for 3.2 hours of manipulation experiments under 193 conditions of motion, friction, mass distribution and grasp. For each experiment, we provide: the global orientation $\mathbf{q}$, rotational velocity $\boldsymbol{\omega}$ and translational acceleration $\mathbf{a}$ measured by the AHRS at 400 Hz; 3D force measurements expressed in the reference frame of the object $\mathscr{R}_{\text{obj.}}$, subsampled from 500 Hz to 400 Hz to match the AHRS; the physical properties of the object: mass m, inertia matrix $\mathbf{J}$ about the center of mass $\mathbf{G}$; and the grasp parameters: for each finger $k \in \mathscr{F}$, the friction coefficient μ_k at contact point $\mathbf{P}_k^c$, and $\mathscr{R}_k = \left(\mathbf{n}_k, \mathbf{t}_k^x, \mathbf{t}_k^y\right)$ a local right-handed reference frame with $\mathbf{n}_k$ the normal to the surface oriented from the finger to the object. Friction coefficients are estimated by instructing the subjects to press and pull the force transducers until slipping and computing the maximum ratio between tangential and normal forces through the Coulomb model:

$$\left\| g_k \mathbf{t}_k^x + h_k \mathbf{t}_k^y \right\|_2 \leq \mu_k f_k, \tag{3.1}$$

with (f_k, g_k, h_k) the local decomposition of contact force $\mathbf{F}_k$:

$$\mathbf{F}_k = f_k \mathbf{n}_k + g_k \mathbf{t}_k^x + h_k \mathbf{t}_k^y. \tag{3.2}$$

3.2.3 Equations of Motion and Synchronization

Let $\boldsymbol{\mathscr{F}}_{\text{c}}$ and $\boldsymbol{\tau}_c$ be the net force and torque due to individual contact forces, and $\boldsymbol{\mathscr{F}}_{\text{d}}$ and $\boldsymbol{\tau}_d$ be the net force and torque due to non-contact forces (e.g., gravitation); the Newton-Euler equations of motion at the center of mass are:

$$\begin{cases} \boldsymbol{\mathscr{F}}_{\text{c}} = m\mathbf{a} - \boldsymbol{\mathscr{F}}_{\text{d}} \\ \boldsymbol{\tau}_{\text{c}} = \mathbf{J}_{\mathbf{q}} \cdot \boldsymbol{\alpha} + \boldsymbol{\omega} \times (\mathbf{J}_{\mathbf{q}} \cdot \boldsymbol{\omega}) - \boldsymbol{\tau}_{\text{d}}, \end{cases} \tag{3.3}$$

with $\mathbf{J}_{\mathbf{q}}$ the inertia matrix at orientation $\mathbf{q}$ and $\boldsymbol{\alpha}$ the rotational acceleration of the object, obtained by numerical differentiation of the AHRS rotational velocity measurements $\boldsymbol{\omega}$. The left hand side elements correspond to the contributions of the force transducer measurements while the right hand side elements can be computed from the object properties and

AHRS kinematics measurements. This allows us to synchronize the kinematic and dynamic measurements temporally while also accounting for sensor uncertainties.

First, the two signals can be synchronized temporally by computing the cross-correlation between the sequences of net forces obtained either from the AHRS or from the force transducers. Second, both the AHRS and the force transducers are subject to measurement errors, resulting in discrepancies in the resulting net force and torque. The specified AHRS maximum acceleration measurement error is of $\pm 0.3\,\mathrm{m}\cdot\mathrm{s}^{-2}$. For an object of mass 500 g, this amounts to net force errors up to $\pm 0.15\,\mathrm{N}$. In contrast, non-linearity and hysteresis can cause measurement errors up to $\pm 1\,\mathrm{N}$ per force transducer, i.e. $\pm 5\,\mathrm{N}$ at most on the net force. In practice, the average net force discrepancy between AHRS and force transducers throughout the whole dataset is 0.33 N. For each experiment, we compute the average net force $\Delta\boldsymbol{\mathscr{F}}_\mathrm{c}$ and torque $\Delta\boldsymbol{\tau}_\mathrm{c}$ discrepancies between AHRS and force transducers. We align their values by computing the minimal offsets $(\Delta\mathbf{F}_k)_{k\in\mathscr{F}}$ that result in $\Delta\boldsymbol{\mathscr{F}}_\mathrm{c}$ and $\Delta\boldsymbol{\tau}_\mathrm{c}$:

$$\min\left\{\mathscr{C}_{\boldsymbol{\mathscr{F}}_\mathrm{c}} + \mathscr{C}_{\boldsymbol{\tau}_\mathrm{c}} + \mathscr{C}_\mathrm{var}\right\}, \tag{3.4}$$

with force-torque discrepancy and variation cost functions:

$$\begin{cases} \mathscr{C}_{\boldsymbol{\mathscr{F}}_\mathrm{c}}\left((\Delta\mathbf{F}_k)_k\right) = \left\| \Delta\boldsymbol{\mathscr{F}}_\mathrm{c} - \sum\limits_{k\in\mathscr{F}} \left[\Delta\mathbf{F}_k\right] \right\|_2^2 \\ \mathscr{C}_{\boldsymbol{\tau}_\mathrm{c}}\left((\Delta\mathbf{F}_k)_k\right) = \left\| \Delta\boldsymbol{\tau}_\mathrm{c} - \sum\limits_{k\in\mathscr{F}} \left[\overrightarrow{\mathbf{GP}_k} \times \Delta\mathbf{F}_k\right] \right\|_2^2 \\ \mathscr{C}_\mathrm{var}\left((\Delta\mathbf{F}_k)_k\right) = \sum\limits_{k\in\mathscr{F}} \|\Delta\mathbf{F}_k\|_2^2 \end{cases} \tag{3.5}$$

In practice, it is preferrable to normalize $\mathscr{C}_{\boldsymbol{\mathscr{F}}_\mathrm{c}}$ and $\mathscr{C}_{\boldsymbol{\tau}_\mathrm{c}}$, e.g., with the initial discrepancies $\Delta\boldsymbol{\mathscr{F}}_\mathrm{c}$ and $\Delta\boldsymbol{\tau}_\mathrm{c}$ respectively. We solve the optimization problem using sequential least squares programming and correct the force transducer measurements with the resulting offsets.

3.3 Force Model

Based on the Newton-Euler equations, the net contact force $\boldsymbol{\mathscr{F}}_\mathrm{c}$ and torque $\boldsymbol{\tau}_\mathrm{c}$ are completely determined by the object's motion and physical properties. However, given $\boldsymbol{\mathscr{F}}_\mathrm{c}$ and $\boldsymbol{\tau}_\mathrm{c}$ can generally be achieved by an infinity of different force distributions. Our force model addresses these two aspects by combining physics-based optimization and learning to reconstruct force distributions that are both physically plausible and similar to actual human grasping.

3.3.1 Physics-Based Optimization for Manipulation

In this section, we formulate the Newton-Euler equations and Coulomb model as constraints of an optimization problem allowing the extraction of force distributions compatible with a given motion. We integrate these constraints in a second-order cone program (SOCP) [LVBL98, BV04, BW07] of the form:

$$\begin{aligned} \min \quad & \mathscr{C}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{P}\mathbf{x} + \mathbf{r}^T\mathbf{x} \\ \text{s.t.} \quad & \begin{cases} \left\|\mathbf{A}_j\mathbf{x} + \mathbf{b}_j\right\|_2 \leq \mathbf{c}_j^T\mathbf{x} + \mathbf{d}_j, \quad j = 1,\ldots,m \\ \mathbf{E}\mathbf{x} \leq \mathbf{f} \\ \mathbf{G}\mathbf{x} = \mathbf{h}. \end{cases} \end{aligned} \tag{3.6}$$

We express conditions of physical plausibility using the local decompositions of Eq. (3.2) as 15 optimization parameters:

$$\mathbf{x} = (f_1, g_1, h_1, \ldots, f_5, g_5, h_5)^T \tag{3.7}$$

Positivity. Recall that for each finger k, we choose the contact normal $\mathbf{n}_k$ oriented inwards the object. With this convention, the normal components f_k are non-negative:

$$f_k \geq 0, \quad k = 1,\ldots,5. \tag{3.8}$$

This can be rewritten in Eq. (3.6) with linear inequality matrices $\mathbf{E}$ and $\mathbf{f}$ of respective sizes 5×15 and 5×1, with:

$$\begin{aligned} \mathbf{E}(i,j) &= \begin{cases} -1 & \text{if} \quad j = 3(i-1)+1 \\ 0 & \text{else} \end{cases} \\ \mathbf{f}(i,1) &= 0. \end{aligned} \tag{3.9}$$

Friction. The Coulomb model of Eq. (3.1) can be written as five cone constraints, i.e., one per finger. For each finger k, the cone constraint matrices $\mathbf{A}_k$, $\mathbf{b}_k$, $\mathbf{c}_k$, $\mathbf{d}_k$, are of respective sizes 2×15, 2×1, 15×1 and 1×1, such that:

$$\mathbf{A}_k\mathbf{x} + \mathbf{b}_k = \begin{pmatrix} g_k \\ h_k \end{pmatrix} \quad \text{and} \quad \mathbf{c}_k^T\mathbf{x} + \mathbf{d}_k = (\mu_k f_k)\,. \tag{3.10}$$

Their elements are defined as follows:

$$\begin{aligned}
\mathbf{A}_k(i,j) &= \begin{cases} 1 & \text{if} \quad j = 3(k-1)+1+i \\ 0 & \text{otherwise} \end{cases} \\
\mathbf{b}_k(i,1) &= 0 \\
\mathbf{c}_k(i,1) &= \begin{cases} \mu_k & \text{if} \quad i = 3(k-1)+1 \\ 0 & \text{otherwise} \end{cases} \\
\mathbf{d}_k(1,1) &= 0.
\end{aligned} \tag{3.11}$$

Equations of motion. Recall from Eq. (3.3) that the net contact force $\mathscr{F}_{\mathrm{c}}$ and torque $\boldsymbol{\tau}_c$ can be determined from kinematic quantities only. The individual finger forces are such that:

$$\begin{cases} \mathscr{F}_{\mathrm{c}} = \displaystyle\sum_{k \in \mathscr{F}} \mathbf{F}_k \\ \boldsymbol{\tau}_{\mathrm{c}} = \displaystyle\sum_{k \in \mathscr{F}} \left[\overrightarrow{\mathbf{GP}_k} \times \mathbf{F}_k \right]. \end{cases} \tag{3.12}$$

We express the Newton-Euler equations in the global reference frame $\mathscr{R}_{\text{global}} = (\mathbf{v_1}, \mathbf{v_2}, \mathbf{v_3})$. The equality constraint matrices $\mathbf{G}$ and $\mathbf{h}$ are of respective sizes 6×15 and 6×1 with:

$$\begin{aligned}
&\forall i = 1,\ldots,3; \quad \forall j = 1,\ldots,15; \quad \forall k = 1,\ldots,5; \\
&\mathbf{G}(i,j) = \begin{cases} \mathbf{n}_k \cdot \mathbf{v}_i & \text{if} \quad j = 3(k-1)+1 \\ \mathbf{t}_k^x \cdot \mathbf{v}_i & \text{if} \quad j = 3(k-1)+2 \\ \mathbf{t}_k^y \cdot \mathbf{v}_i & \text{if} \quad j = 3(k-1)+3 \\ 0 & \text{otherwise} \end{cases} \\
&\mathbf{h}(i,1) = \mathscr{F}_{\mathrm{c}} \cdot \mathbf{v}_i \\
&\mathbf{G}(i+3,j) = \begin{cases} \left[\overrightarrow{\mathbf{GP}_k} \times \mathbf{n}_k \right] \cdot \mathbf{v}_i \text{ if } j = 3(k-1)+1 \\ \left[\overrightarrow{\mathbf{GP}_k} \times \mathbf{t}_k^x \right] \cdot \mathbf{v}_i \text{ if } j = 3(k-1)+2 \\ \left[\overrightarrow{\mathbf{GP}_k} \times \mathbf{t}_k^y \right] \cdot \mathbf{v}_i \text{ if } j = 3(k-1)+3 \\ 0 \quad \text{otherwise} \end{cases} \\
&\mathbf{h}(i+3,1) = \boldsymbol{\tau}_c \cdot \mathbf{v}_i
\end{aligned} \tag{3.13}$$

Cost function. Physically plausible force distributions can be computed with a cost function depending only on the optimization variables, e.g. minimal (squared) L^2 norm [PKQA15b]:

$$\mathscr{C}_{L^2}(\mathbf{x}) = \sum_{k \in \mathscr{F}} \left[{f_k}^2 + {g_k}^2 + {h_k}^2 \right] = \sum_{k \in \mathscr{F}} \|\mathbf{F}_k\|_2^2 . \tag{3.14}$$

Yet, the resulting forces can significantly differ from those humans really apply (see Fig. 3.2). Instead, we consider a cost minimizing the discrepancy with given target forces $\widetilde{\mathbf{F}}_k$:

$$\mathscr{C}_{\widetilde{\mathbf{F}}_k}(\mathbf{x}) = \sum_{k \in \mathscr{F}} \left\| \mathbf{F}_k - \widetilde{\mathbf{F}}_k \right\|_2^2 \tag{3.15}$$

In the following, we use $\mathscr{C}_{\widetilde{\mathbf{F}}_k}$ to correct force transducer measurements and neural network prediction uncertainties.

3.3.2 Learning Features

The criteria that are optimized by the central nervous system in hand-object manipulation are still unknown (see Section 1.4.3). A major obstacle to their identification is a dependency on musculoskeletal parameters that can be difficult to measure precisely [EMHvdB07]. Rather than explicitly considering such low-level parameters, the force model we propose in this work relies on an artificial neural network that predicts manipulation forces from high-level kinematic features. Based on the dataset presented in Section 3.2, we group the available parameters into three categories:

- Object and grasp parameters: location of the center of mass $\mathbf{G}$ in $\mathscr{R}_{\text{obj.}}$, mass m, inertia matrix $\mathbf{J}$, contact point locations $\mathbf{P}_k$ and friction coefficients μ_k.
- Kinematic parameters: appearing in Eq. (3.3) are the object's orientation $\mathbf{q}$ in $\mathscr{R}_{\text{global}}$, rotational velocity $\boldsymbol{\omega}$, rotational acceleration $\boldsymbol{\alpha}$ and translational acceleration $\mathbf{a}$. The quantities $\mathbf{q}, \boldsymbol{\omega}, \mathbf{a}$ are directly measured by the AHRS. $\boldsymbol{\alpha}$ is obtained by simple numerical differentiation of $\boldsymbol{\omega}$. Alternatively to the AHRS, these kinematic parameters can also be estimated from visual tracking, through double differentiation of the object's pose through time (see Section 3.5.2).
- Force transducer measurements $\widetilde{\mathbf{F}}_k$.

To alleviate sensing uncertainties, we extract physically plausible force distributions $\mathbf{F}_k$ in the vicinity of the possibly inaccurate measurements $\widetilde{\mathbf{F}}_k$, as depicted in Fig. 3.3.

The objective is then to learn the extracted force distributions $\mathbf{F}_k$ based on input parameters that depend only on the grasp, the object and its kinematics. We select these input features

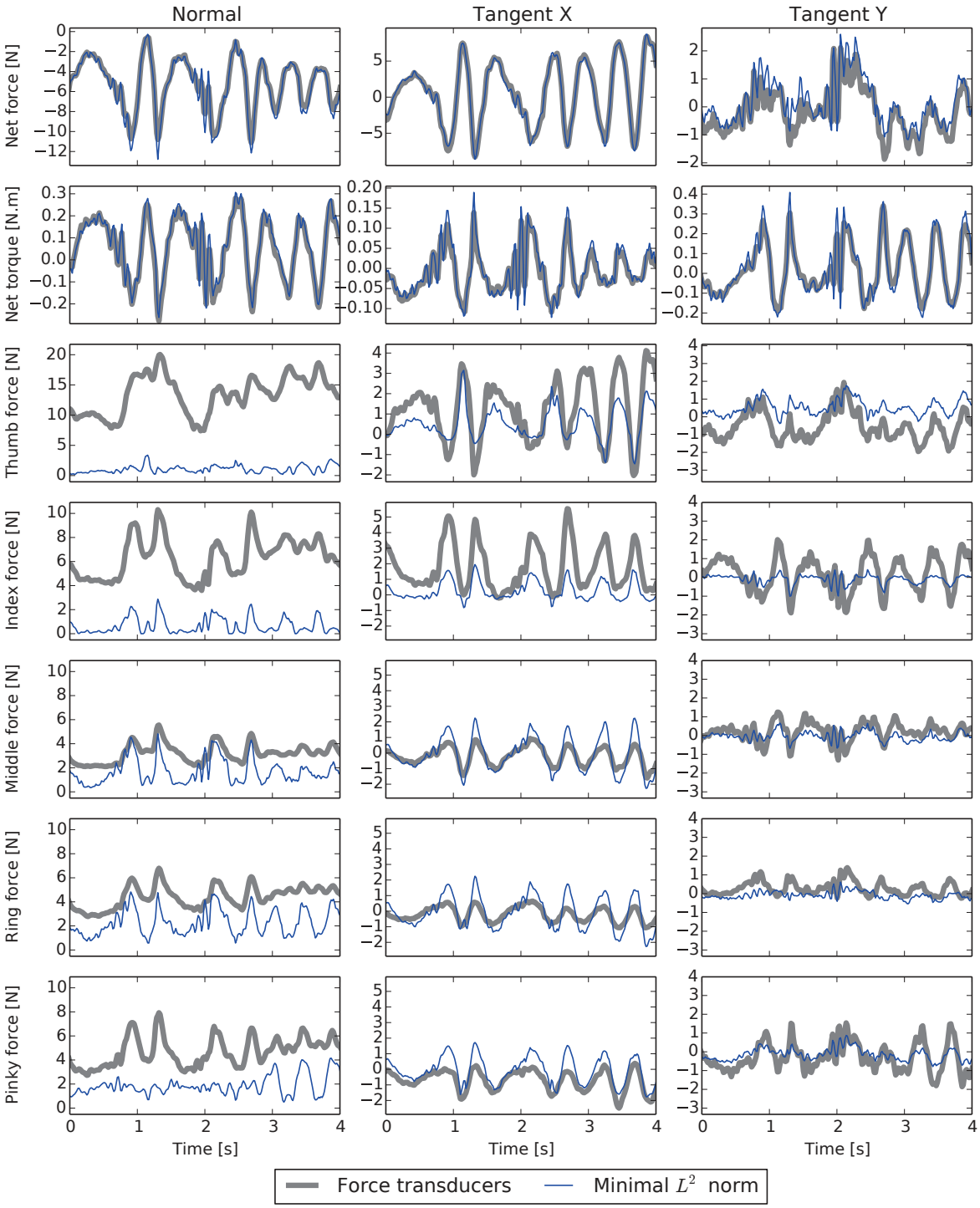


Figure 3.2: Force distributions computed only by physics-based optimization are guaranteed to result in the observed motion (net force and torque) but can significantly differ from the real distributions at the finger level.

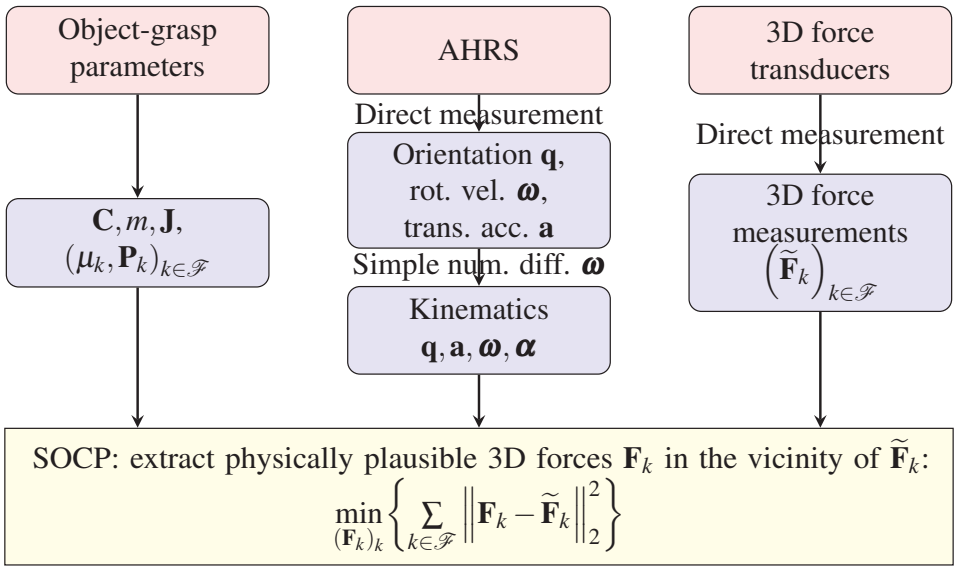


Figure 3.3: For each experiment, we extract force distributions compatible with the observed motion in the vicinity of the transducer measurements.

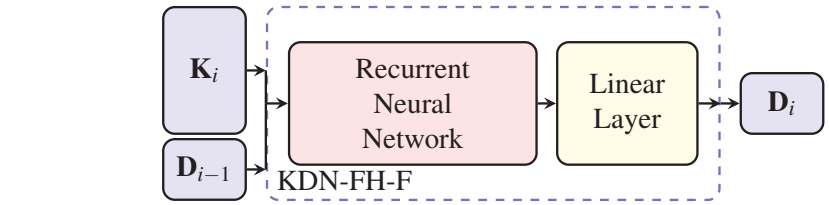
based on their contribution to the Newton-Euler equations of motion. A first approach could be to take the raw parameters listed above. However, their influence is often determined not individually but rather in interaction with other parameters. From Eq. (3.12), the positions of the center of mass $\mathbf{G}$ and contact points $\mathbf{P}_k$ are meaningful not on their own but in relation to each other as $\overrightarrow{\mathbf{GP}_k}$. Similarly, from Eq. (3.3), we summarize the contributions of m, $\mathbf{a}$, $\mathbf{J}$, $\mathbf{q}$, $\boldsymbol{\omega}$, $\boldsymbol{\alpha}$ into the target net contact force $\boldsymbol{\mathscr{F}}_c$ and torque $\boldsymbol{\tau}_c$.

Recall that $\boldsymbol{\mathscr{F}}_c$ and $\boldsymbol{\tau}_c$ are expressed in $\mathscr{R}_{\text{global}}$. Since the dataset focuses on static grasps, for each experiment, the contact points are constant in any frame attached to the object. We account for translational and rotational invariances by projecting $\boldsymbol{\mathscr{F}}_c$, $\boldsymbol{\tau}_c$ and $\overrightarrow{\mathbf{GP}_k}$ on $\mathscr{R}_{\text{obj.}}$. Thus, the input features stemming from the Newton-Euler equations are:

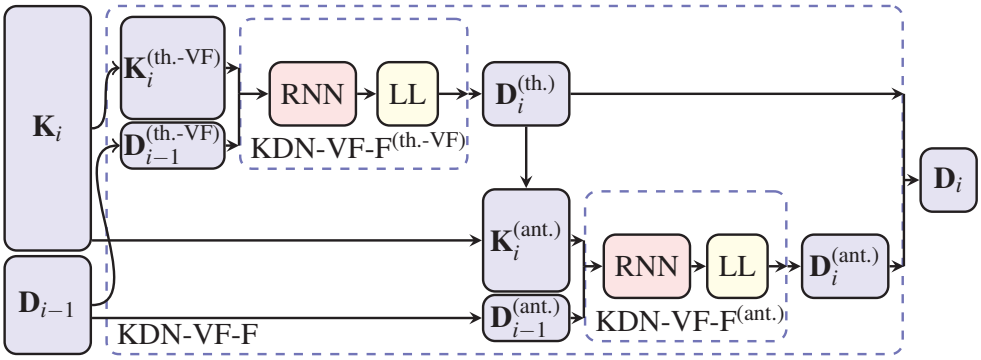
$$\forall (k, \mathbf{v}) \in \mathscr{F} \times \mathscr{R}_{\text{obj.}}, \quad \begin{cases} p_{\mathbf{v}}^{\boldsymbol{\mathscr{F}}_c} &= \boldsymbol{\mathscr{F}}_c \cdot \mathbf{v} \\ p_{\mathbf{v}}^{\boldsymbol{\tau}_c} &= \boldsymbol{\tau}_c \cdot \mathbf{v} \\ p_{\mathbf{v}}^{\mathbf{P}_k} &= \overrightarrow{\mathbf{GP}_k} \cdot \mathbf{v} \end{cases} . \tag{3.16}$$

In addition, we consider the average friction coefficient:

$$p^{\mu} = \langle \mu_k \rangle_{k\in\mathscr{F}} \tag{3.17}$$



(a) KDN-FH-F: full hand forces.



(b) KDN-VF-F: two-stage thumb-virtual finger network.

Figure 3.4: Two RNN architectures learning the manipulation forces at each fingertip based on the current kinematics and past forces.

We regroup these parameters, derived from the grasp-object properties and kinematics, into a 22-element vector $\mathbf{K}$:

$$\mathbf{K} = \left(p_{\mathbf{v}}^{\mathscr{F}_c}, p_{\mathbf{v}}^{\boldsymbol{\tau}_c}, p_{\mathbf{v}}^{\mathbf{P}_k}, p^{\mu} \right)_{(k,\mathbf{v}) \in \mathscr{F} \times \mathscr{R}_{\text{obj.}}} \tag{3.18}$$

Similarly, we denote by $\mathbf{D}$ the 15-element vector of the force distribution expressed in the local frame:

$$\mathbf{D} = (\mathbf{F}_k \cdot \mathbf{v})_{(k,\mathbf{v}) \in \mathscr{F} \times \mathscr{R}_{\text{obj.}}} \tag{3.19}$$

Note that attaching the frame to a chosen finger also helps preserve invariances througout objects and experiments. Using the thumb contact space $\mathscr{R}_{\text{th.}} = \left(\mathbf{t}_0^x, \mathbf{t}_0^y, \mathbf{n}_0\right)$ with $\mathbf{t}_0^y$ towards the palm, all four antagonist fingers share the same coordinate along $\mathbf{n}_0$, hence reducing $\mathbf{K}$ to 19 elements.

3.3.3 Neural Network Modelling

Given an object-grasp configuration, the goal of our work is to obtain an accurate estimate of the force distribution applied to achieve an observed motion, e.g. by reconstructing a force

distribution function F such that:

$$\mathbf{D} = F(\mathbf{K}) \tag{3.20}$$

In [PKQA15b], we approximated such a function with an MLP learning internal forces. Yet, our previous formulation has two important limitations:

- Similar tasks can be achieved with different force distributions, i.e., multiple values of $\mathbf{D}$ can be associated to the same value of $\mathbf{K}$. As such, different distributions would tend to be averaged albeit equally valid.
- In Eq. (3.20), consecutive force distributions are independent through time. Instead, since contact is never broken, we should expect that the force distribution $\mathbf{D}_i$ at timestamp i depends not only on the corresponding task parameters $\mathbf{K}_i$ but also on the past.

Therefore, we adopt the following alternative formulation:

$$\mathbf{D}_i = F\left(\mathbf{K}_i, \mathbf{D}_{i-1}, \left(\mathbf{K}_j, \mathbf{D}_{j-1}\right)_{j=1,i-1}\right) \tag{3.21}$$

Through the dependency on past kinodynamics, the first limitation is also mitigated since forces are distinguished based on $\mathbf{K}_i$ trajectories rather than single samples.

We capture the sequential nature of manipulation kinodynamics using recurrent neural networks (RNN) [Elm90], with long short term memory (LSTM) neurons [HS97] that allow for better learning of long-term dependencies. In this work, we investigate four kinodynamics network (KDN) architectures. The first model we propose, KDN-FH-F, directly predicts full hand forces $\mathbf{D}_i$ from the current kinematics $\mathbf{K}_i$ and previous distribution $\mathbf{D}_{i-1}$ using a single RNN:

$$\mathbf{D}_i = \text{KDN-FH-F}(\mathbf{K}_i, \mathbf{D}_{i-1}). \tag{3.22}$$

Alternatively, we propose a two-stage network inspired by the virtual finger model, KDN-VF-F. A first RNN estimates thumb forces $\mathbf{D}_i^{(\text{th.})}$ based on parameters reducing the full grasp to a thumb and virtual finger:

$$\mathbf{D}_i^{(\text{th.})} = \text{KDN-VF-F}^{(\text{th.-VF})}\left(\mathbf{K}_i^{(\text{th.-VF})}, \mathbf{D}_{i-1}^{(\text{th.})}\right). \tag{3.23}$$

We associate the virtual finger with the centroid of the antagonist fingers $\mathscr{F}_{\text{ant.}}$ and their average friction coefficient:

$$\begin{aligned} \mathbf{K}_i^{(\text{th.-VF})} &= \left(p_{\mathbf{v}}^{\boldsymbol{\mathscr{F}}_c}, p_{\mathbf{v}}^{\boldsymbol{\tau}_c}, p_{\mathbf{v}}^{\mathbf{P}_{\text{th.}}}, p^{\mu_{\text{th.}}}, p_{\mathbf{v}}^{\mathbf{P}_{\text{ant.}}}, p^{\mu_{\text{ant.}}} \right)_{\mathbf{v} \in \mathscr{R}_{\text{th.}}} \\ \text{with} \quad & \begin{cases} p_{\mathbf{v}}^{\mathbf{P}_{\text{ant.}}} = \left\langle p_{\mathbf{v}}^{\mathbf{P}_k} \right\rangle_{k \in \mathscr{F}_{\text{ant.}}} \\ p^{\mu_{\text{ant.}}} = \langle \mu_k \rangle_{k \in \mathscr{F}_{\text{ant.}}} \end{cases} \end{aligned} \tag{3.24}$$

We compute the total wrench due to the antagonist fingers based on the contribution of the estimated thumb force $\mathbf{F}_{\text{th.}}$:

$$\forall \mathbf{v} \in \mathscr{R}_{\text{th.}}, \quad \begin{cases} p_{\mathbf{v}}^{\boldsymbol{\mathscr{F}}_{\text{ant.}}} = (\boldsymbol{\mathscr{F}}_c - \mathbf{F}_{\text{th.}}) \cdot \mathbf{v} \\ p_{\mathbf{v}}^{\boldsymbol{\tau}_{\text{ant.}}} = \left(\boldsymbol{\tau}_c - \left(\overrightarrow{\mathbf{GP}_{\text{th.}}} \times \mathbf{F}_{\text{th.}} \right) \right) \cdot \mathbf{v} \end{cases} \tag{3.25}$$

The second stage of the network learns the resulting distribution $\mathbf{D}_i^{(\text{ant.})}$ over the antagonist fingers:

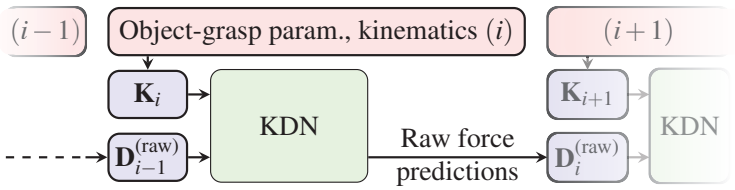
$$\begin{aligned} \mathbf{D}_i^{(\text{ant.})} &= \text{KDN-VF-F}^{(\text{ant.})} \left(\mathbf{K}_i^{(\text{ant.})}, \mathbf{D}_{i-1}^{(\text{ant.})} \right) \\ \text{with} \quad \mathbf{K}_i^{(\text{ant.})} &= \left(p_{\mathbf{v}}^{\boldsymbol{\mathscr{F}}_{\text{ant.}}}, p_{\mathbf{v}}^{\boldsymbol{\tau}_{\text{ant.}}}, p_{\mathbf{v}}^{\mathbf{P}_k}, p^{\mu_{\text{ant.}}} \right)_{(k,\mathbf{v}) \in \mathscr{F}_{\text{ant.}} \times \mathscr{R}_{\text{th.}}} \end{aligned} \tag{3.26}$$

We depict KDN-FH-F and KDN-VF-F in Fig. 3.4.

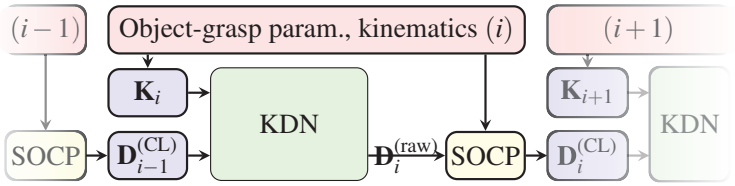
In order to further address the fact that the same motion can be due to different yet equally valid force distributions, we introduce alternative versions of KDN-FH-F and KDN-VF-F that associate current kinematics $\mathbf{K}_i$ and past forces $\mathbf{D}_{i-1}$ to force variations $\Delta\mathbf{D}_i$. In doing so, we explicitly associate the same output to two sequences that differ by a constant internal force distribution. We denote these alternative architectures by KDN-FH-Δ and KDN-VF-Δ. Full manipulation forces are then reconstructed by sequentially adding predicted force variations. As such, these architectures are prone to drift and may require additional control.

3.4 Experiments

We train the four architectures KDN-FH-F, KDN-FH-Δ, KDN-VF-F, KDN-VF-Δ on the manipulation kinodynamics dataset of Section 3.2. Note that its sampling rate (400 Hz) far exceeds the frame rate of off-the-shelf RGB-D sensors such as Microsoft Kinect (30 fps) and Asus Xtion (60 fps). In order to be compatible with vision-based kinematics (Section 3.5), we down-sample the dataset to 60 Hz and split it for training (60 %), validation (20 %) and



(a) Open-loop: force predictions are not checked for physical plausibility.



(b) Closed-loop: force predictions are corrected between time steps.

Figure 3.5: Open-loop and closed-loop force generation processes.

testing (20 %). In KDN-FH-F and KDN-FH-Δ, the RNN contains two hidden-layers of size 256. In KDN-VF-F and KDN-VF-Δ, each RNN stage contains a single hidden-layer of size 256. The networks are implemented and trained within the Torch7 framework [CKF11] using stochastic gradient descent with a mean square error criterion and dropout [SHK+14] to avoid overfitting.

3.4.1 Force Reconstruction Model

From Eq. (3.21), each force distribution $\mathbf{D}_i$ is computed from the corresponding kinematics $\mathbf{K}_i$ and the distribution at the previous time step $\mathbf{D}_{i-1}$. Due to this sequential process, the predicted forces may drift away from the transducer measurements throughout the experiment. We assess the influence of the experiment duration in Section 3.4.2. Similarly, the predicted sequence also depends on the choice of the initial force distribution $\mathbf{D}_0$, which we address in Section 3.4.3. In this section, we discuss the reconstruction of physically plausible manipulation forces from KDN predictions and present our results on full-length experiments with ground-truth initialization. Manipulation forces are obtained by projecting the components of $\mathbf{D}_i$ onto the local reference frame following Eq. (3.19). Since the Newton-Euler and Coulomb laws are not explicitly enforced by the RNNs, the raw predictions are not guaranteed to result in the observed motion. We depict the open-loop prediction process in Fig. 3.5a. Using the SOCP described in Fig. 3.3 with the KDN outputs instead of the force transducer measurements, the sequence of raw predictions can be post-processed to yield physically plausible force distributions in their vicinity. Another important point is that the training sequences are physically coherent. Thus, repeatedly feeding incompatible kinematics and forces into the KDN may result in growing prediction errors. We tackle this

Table 3.1: Force Estimation Errors on Full-Length Manipulation Sequences

	Open-loop	Post-processed	Closed-loop
KDN-FH-F	0.49 (4.14)	0.44 (4.07)	**0.16** (**3.54**)
KDN-FH-Δ	−43.67 (156.72)	0.60 (**4.74**)	**0.50** (11.03)
KDN-VF-F	0.29 (3.19)	0.29 (3.13)	**0.12** (**2.60**)
KDN-VF-Δ	1145.06 (3984.86)	3.54 (11.80)	**2.32** (**6.60**)

issue by integrating the SOCP in closed-loop with the KDN such that force predictions are consistently corrected between time steps. We depict the closed-loop prediction process in Fig. 3.5b.

We compute the estimation errors (average and standard deviation) for the four network architectures using open-loop prediction, offline post-processing or closed-loop prediction and report the results in Table 3.1. In general, post-processing and closed-loop prediction perform better than open-loop prediction. This is especially the case for the networks estimating force variations $\Delta\mathbf{D}_i$, as these tend to be rather unstable and prone to drift. For instance, in Fig. 3.6, the open-loop predictions rapidly drift away from the net force and torque producing the target kinematics. Additionally, the individual normal forces become negative, which would mean that fingertips pull rather than press on the contact surface. Offline post-processing looks for physically valid forces in the vicinity of negative raw predictions, finally yielding distributions of minimal norm. In contrast, closed-loop prediction can help the network recover from incorrect predictions and maintain human-like grasping forces. Overall, the networks predicting force distributions generally perform better than those estimating force variations. For those, post-processing does not appear to significantly improve the open-loop estimations, which shows that these RNNs are rather successful at capturing the relationship between kinematics and underlying forces. Finally, the better accuracy of KDN-VF-F indicates that the virtual finger model can be a useful tool to decouple the static indeterminacy stemming from the thumb and antagonist fingers. Still, the two-stage architecture makes KDN-VF-Δ more prone to drift since thumb force predictions cannot be corrected alone before computing the antagonist forces.

3.4.2 Force Drift Over Time

Due to the infinity of force distributions compatible with a given motion, the force predictions are likely to deviate from the transducer measurements over time. We quantify this effect by splitting the experiments into sub-sequences of maximum duration 1, 2, 4, 8, 16, 32 s (resp. 60, 120, 240, 480, 960, 1920 samples) and computing the resulting estimation errors

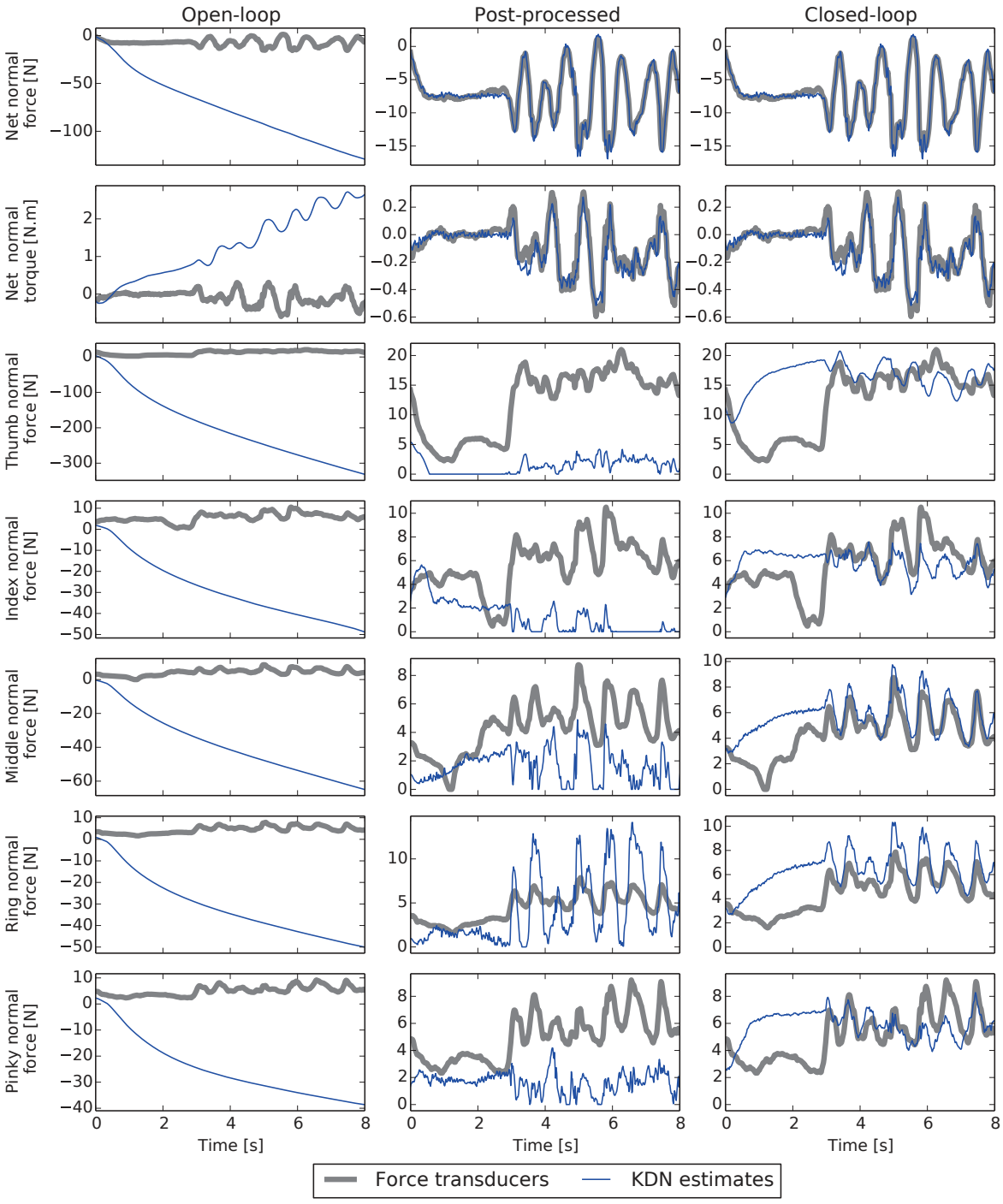


Figure 3.6: Open-loop, post-processed and closed-loop force predictions for KDN-VF-Δ (normal components). In this example, the open-loop estimation drifts away from physically plausible solutions (negative normal forces). Compatibility with the observed motion is enforced through offline post-processing or closed-loop control at each time step.

for the four architectures with ground-truth initialization and offline post-processing or closed-loop prediction. For completeness, we reproduce the estimation errors over the full length sequences (average duration 60.1 s, standard deviation 3.8 s). We report the results in Table 3.2.

In line with the observations made on the full-length experiments, KDN-VF-Δ is the worst-performing network for every sequence duration, whereas KDN-VF-F is consistently best-performing or closely behind. This indicates again that decoupling thumb and antagonist redundancies is a viable strategy, yet more unstable in the presence of force variation uncertainties. We also observed that KDN-FH-Δ yields better results than its full force counterpart KDN-FH-F on the 1 s sequence duration and 2 s to a lesser extent. Recall that the $\Delta\mathbf{D}_i$ networks were introduced to accommodate the possibility of having the same motion caused by an infinity of force distributions. It appears here that KDN-FH-Δ is better at matching the real force variations on short sequences. Still, the applicability of this result on real manipulation tasks is limited due to the two following aspects. First, for sequence lengths greater than 2 s, the accumulation of $\Delta\mathbf{D}_i$ prediction errors becomes predominant. Second, the accuracy of the predicted force sequence is contingent on its initialization on the real forces being applied as measured by force transducers, which, ultimately, the force estimation framework aims at completely circumventing.

3.4.3 Force Sequence Initialization

Manipulation forces are sequentially computed based on an initial distribution that can be adjusted freely. We assess the force variability following non ground-truth initialization for sequences of maximum duration 4.0, 8.0, 16.0 and 32.0 s. Each sequence is initialized as follows. Using the average and standard deviation $\boldsymbol{\mu}, \boldsymbol{\sigma}$ of each finger force throughout the manipulation kinodynamics dataset, we pick a random sample $\widetilde{\mathbf{D}}_0$ following the normal distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$. We then correct $\widetilde{\mathbf{D}}_0$ using the SOCP of Section 3.3.1. Thus, we ensure that the resulting distribution $\mathbf{D}_0$ is compatible with the initial kinematics $\mathbf{K}_0$. We report the force estimation errors for random and ground-truth initialization in Table 3.3.

Expectedly, ground-truth initialization yields better force estimates overall. Still, for each architecture, the performance difference decreases with the sequence duration. Indeed, even when starting from the same distribution, the predicted sequence is likely to deviate from the transducer measurements due to the infinity of force variations producing the same motion. This mitigates the importance of the force initialization over time. In the case of the best-performing network, KDN-VF-F (closed-loop), the difference is actually minor even starting from 8.0 s sequences. Finally, note that for any initial force distribution, the resulting sequence is constructed to be physically plausible given the observed motion and compatible

Table 3.2: Force Estimation Drift Through Time

	1.0 s	2.0 s	4.0 s	8.0 s	16.0 s	32.0 s	Full length
KDN-FH-F, post-processed	−0.21 (2.06)	−0.21 (2.43)	−0.13 (2.86)	−0.04 (3.22)	**0.07** (3.54)	0.19 (3.76)	0.44 (4.07)
KDN-FH-F, closed-loop	−0.13 (2.20)	−0.12 (2.47)	−0.07 (2.80)	**0.00** (3.07)	0.06 (3.24)	0.08 (3.33)	0.16 (3.54)
KDN-FH-Δ, post-processed	**0.00** (**1.80**)	0.15 (2.42)	0.36 (3.22)	0.56 (3.89)	0.68 (4.34)	0.56 (4.62)	0.60 (4.74)
KDN-FH-Δ, closed-loop	0.02 (1.87)	0.11 (2.48)	0.27 (3.44)	0.45 (5.14)	0.58 (7.39)	0.57 (9.32)	0.50 (11.03)
KDN-VF-F, post-processed	0.07 (2.09)	0.13 (2.51)	0.20 (2.82)	0.25 (2.99)	0.27 (3.07)	0.28 (3.11)	0.29 (3.13)
KDN-VF-F, closed-loop	0.02 (1.86)	**0.04** (**2.16**)	**0.07** (**2.38**)	0.10 (**2.50**)	0.11 (**2.56**)	**0.12** (**2.58**)	**0.12** (**2.60**)
KDN-VF-Δ, post-processed	0.43 (2.93)	0.87 (4.47)	1.64 (7.11)	2.37 (9.33)	2.90 (10.61)	2.94 (11.13)	3.54 (11.80)
KDN-VF-Δ, closed-loop	0.41 (2.47)	0.76 (3.45)	1.24 (4.74)	1.69 (5.69)	1.99 (6.17)	2.15 (6.43)	2.32 (6.60)

Table 3.3: Influence of Force Prediction Initialization

	4.0 s		8.0 s		16.0 s		32.0 s	
	Reference	Random	Reference	Random	Reference	Random	Reference	Random
KDN-FH-F, PP	−0.13 (**2.86**)	**−0.00** (3.42)	**−0.04** (**3.22**)	0.12 (3.60)	**0.07** (**3.54**)	0.21 (3.76)	**0.19** (**3.76**)	**0.19** (3.80)
KDN-FH-F, CL	**−0.07** (**2.80**)	0.09 (3.36)	**0.00** (**3.07**)	0.10 (3.43)	**0.06** (**3.24**)	0.09 (3.42)	0.08 (**3.33**)	**0.06** (3.36)
KDN-FH-Δ, PP	0.36 (**3.22**)	**0.34** (3.72)	**0.56** (**3.89**)	0.52 (4.25)	0.68 (**4.34**)	**0.64** (4.49)	0.56 (**4.62**)	**0.52** (4.73)
KDN-FH-Δ, CL	**0.27** (**3.44**)	0.37 (4.08)	**0.45** (**5.14**)	0.53 (5.75)	**0.58** (7.39)	0.63 (**7.35**)	0.57 (**9.32**)	**0.56** (9.59)
KDN-VF-F, PP	**0.20** (**2.82**)	0.22 (3.01)	**0.25** (**2.99**)	0.27 (3.08)	**0.27** (**3.07**)	0.28 (3.13)	**0.28** (**3.11**)	0.29 (3.14)
KDN-VF-F, CL	**0.07** (**2.38**)	0.12 (2.61)	**0.10** (**2.50**)	0.12 (2.63)	**0.11** (**2.56**)	0.13 (2.63)	**0.12** (**2.58**)	0.13 (2.63)
KDN-VF-Δ, PP	**1.64** (**7.11**)	1.79 (7.55)	**2.37** (**9.33**)	**2.37** (9.50)	2.90 (10.61)	**2.70** (**10.32**)	**2.94** (11.13)	2.99 (**11.10**)
KDN-VF-Δ, CL	**1.24** (**4.74**)	1.27 (5.11)	**1.69** (**5.69**)	1.75 (5.86)	**1.99** (**6.17**)	2.06 (6.29)	**2.15** (**6.43**)	2.18 (6.47)

with the forces a human could likely apply, based on the manipulation kinodynamics dataset. This allows the generation of force sequences following different profiles for the same motion (e.g., light or strong starting grasp). This method can also be used to reinitialize the prediction model when the resulting distributions are unreliable, as it may happen in the presence of motion tracking discontinuities.

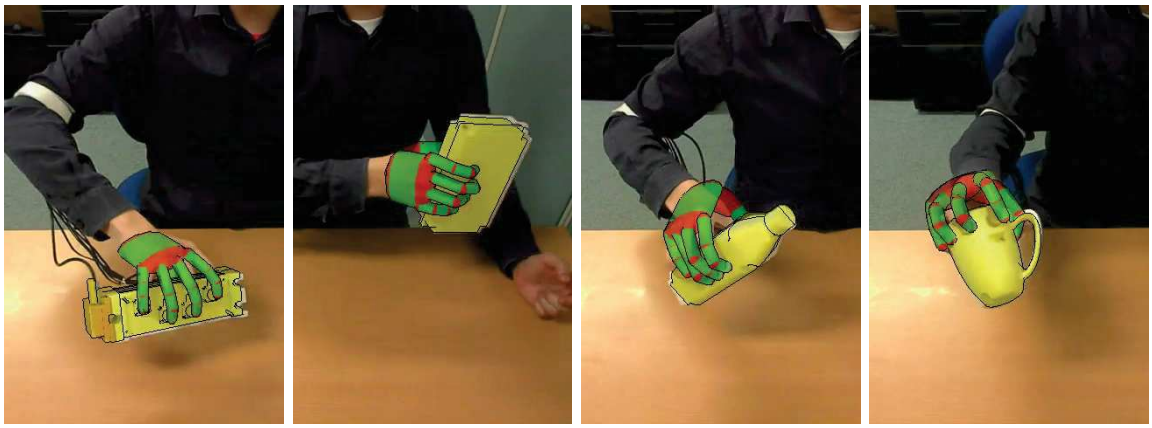
3.5 Force Sensing From Vision

In the previous sections, we showed that the finger forces applied during manipulation can be inferred based on the kinematics of the object, as measured by a high-performance AHRS. Now, we propose to estimate the object's kinematics from markerless visual tracking, thus circumventing the need for any instrumentation whatsoever.

3.5.1 Model-Based Tracking

Along with the physical properties of the manipulated object, the force estimation framework requires its kinematics and the location of the contact points over which forces are distributed. Object kinematics and contact points can be attained by means of tracking the hand and the manipulated object in 3D. Given such a successful 3D tracking, the kinematics can readily be computed from the motion of the object, and the contact points by reasoning about the proximity of the object and the fingers of the hand. Achieving hand-object tracking at the level of accuracy and robustness that is required for visual force estimation is a challenging task. We recorded experiments for quantitative evaluation using a SoftKinetic DepthSense 325 sensor. In the recorded sequences, the motion of the hand-object compound was such that a wide range of linear and angular velocities was explored. In practice, such motions frequently induce high levels of motion blur and strong (in some cases, complete) occlusions. There is also considerable noise in the depth measurements provided by the sensor which, in some cases, is systematic (e.g. slanted surface artifacts).

We used the 3D hand-object tracking method of [KA14]. This choice was derived from our experience in [PKQA15b] which showed the efficacy and flexibility of the Ensemble of Collaborative Trackers (ECT) when dealing with more than a single object or hand. Through extensive quantitative experiments, we found that ECT yields accurate object kinematics estimates, as we discuss in Section 3.5.2. The accuracy of the force estimates depends mostly on that of the contact points. Indicatively, simulating a Gaussian noise of standard deviation 5 mm (resp. 10 mm) on the true contact points yields force reconstruction errors of zero mean (same net forces) and 0.87 N (resp. 1.54 N) standard deviation. In our preliminary



(a) Dataset obj. (b) Box. (c) Bottle. (d) Mug.

Figure 3.7: The hand and the object are tracked as a rigid compound.

experiments, the average contact point estimation error was greater than 20 mm. It should be noted that tracking the object alone fails due to the object occlusions by the manipulating hand not being accounted for. To deal with this problem, we capitalize on the observation that in the scenarios we are interested in, the hand achieves a firm grasp that changes only slightly when moving the object around. Under this assumption, as soon as the hand grasps the object, the hand and the object can be viewed as a single rigid compound. Thus, in a first step, we track hand-object interaction with [KA14]. We then select a frame where the mutual hand-object occlusions are minimal. For that particular frame, we execute anew the optimization step by incorporating an extra term in the objective function that favors a hand pose where the fingertips touch the object at the known contact points. This leads to a hand-object configuration that is most compatible to observations, while respecting the contact point soft constraints. To arrive at this configuration, both the configuration of the hand and the object are revised. This configuration is then considered as a rigid compound which is used to track the whole sequence anew. The first tracking pass involves the optimization of 34 parameters per frame, 27 for the hand and 7 for the object. The second pass corresponds to 7 parameters only: the rigid transform of the compound.

3.5.2 Kinematics Estimation From Visual Tracking

With the camera calibrated intrinsically and extrinsically such that the gravity vector is known, we record and process 12 tracking experiments using the following objects. First, the instrumented device used in Section 3.2, in a configuration that does not appear in the manipulation kinodynamics dataset (mass 279 g). Second, three objects used in daily activities, 3D-printed and equipped with AHRS and force transducers for ground truth:

Table 3.4: Kinematics Estimation Errors from Tracking

	Central	Gaussian	Algebraic
Trans. acc. $[\mathrm{m}\cdot\mathrm{s}^{-2}]$	$0.31(25.36)$	$-0.02(2.92)$	$-0.05(3.03)$
Rot. vel. $[\mathrm{rad}\cdot\mathrm{s}^{-1}]$	$0.14(446.45)$	$-0.05(30.94)$	$0.01(31.76)$
Force [N]	$1.18(8.94)$	$0.01(0.72)$	$0.01(0.75)$

a cuboid box (856 g), a small bottle (453 g), and a mug (174 g). We use the latter as an application of the force model on non-prismatic grasps in Section 3.6.2. We depict sample tracking results in Fig. 3.7.

Given the pose of the object throughout the experiment, we estimate its first and second-order kinematics by numerical differentiation. This process is rather delicate as noise in the estimated trajectory generates spikes in its derivatives, i.e. velocity and acceleration, therefore forces. The effects of noise can usually be mitigated by smoothing the original signal over several samples or using appropriate filters, e.g. Gaussian. However, force profiles occurring in manipulation tasks are naturally spiky (see Fig. 3.6), as simply moving and stopping an object yields successive acceleration vectors in opposite directions. Therefore, smoothing the trajectory of the object comes at the expense of the ability to discern sudden variations in acceleration profiles, which is crucial.

As an alternative to classical numerical differentiation methods, we investigate the use of algebraic numerical differentiators [FSR03, MJF09] which do not assume any statistical properties on the signal's noise. We compare the kinematics estimates to the AHRS measurements on translational acceleration and rotational velocity. In order to quantify the effect on force estimation, we also compute the decomposition of the force transducer measurements on AHRS and vision-based kinematics. Denoting by $T_s = 1/60\,\mathrm{s}$ the time period between frames, we find an optimal Gaussian kernel of standard deviation $\sigma = 3T_s$ truncated at $\pm 4\sigma$. Similarly, the (κ, μ) algebraic numerical differentiator performs best as a filter of half width $4T_s$ with parameters $\kappa = \mu = 0.5$. We report the resulting kinematics estimation errors in Table 3.4.

On typical tracking sequences, smoothing techniques appear necessary to compute reliable kinematics estimates. Both the Gaussian and algebraic filters yield reasonable force discrepancies despite possible tracking uncertainties and discontinuities. Overall, while the Gaussian filter seems to perform slightly better than the algebraic filter, the latter also requires significantly less samples per estimate. This allows for a shorter lag for real time applications while also better capturing high frequency force variations, at the cost of a slightly larger sensitivity to tracking noise.

3.5.3 Force Prediction From Vision-Based Kinematics

Using a single camera, we track manipulation experiments and estimate the object's kinematics with algebraic filtering. In Section 3.4, although the four network architectures are trained on AHRS data, the object's kinematics is used as an input without consideration of the way it is measured. Thus, the trained networks can seamlessly generate force sequences from vision-based kinematics. In order to be completely independent of ground-truth sensing, we use the random initialization process described in Section 3.4.3. We compute the resulting estimation errors with respect to ground-truth force transducer measurements, along with, for reference, force predictions derived from the AHRS kinematics, none of these being used in the vision-based estimation process. We report our results in Table 3.5.

Under the same initialization conditions, forces computed from vision are comparable to forces computed from AHRS measurements. The decrease in accuracy is most noticeable on networks estimating force variations $\Delta\mathbf{D}_i$ due to a higher tendency to drift, as discussed in Section 3.4, but also additional uncertainties from visual tracking. We depict an example of forces estimated from vision in Fig. 3.8. Tracking discontinuities (e.g., lost hand-object pose), following second-order differentiation, are perceived by the force estimation framework as acceleration spikes and result in sudden fingertip force variations. These errors accumulate in the case of $\Delta\mathbf{D}_i$ networks since each prediction is directly relative to the preceding sample. When erroneous kinematics can be identified, their impact can be mitigated by reinitializing the prediction process based on the last reliable sample. However, while doing so is straightforward when AHRS measurements are available, it is difficult from the tracked kinematics alone, since acceleration spikes are not necessarily due to discontinuities but can also stem from actual sudden motions. Overall, KDN-VF-F appears the most resilient architecture to visual tracking uncertainties.

3.6 Discussion

3.6.1 Visual Tracking Assumptions

In Section 3.5.1, we suppose the contact points known and use them to compute a static grasp throughout the motion. Note that our force estimation framework itself is independent of the tracking method employed as long as reliable motion and contact information can be provided. The difficulty for us was to collect ground-truth measurements to validate our approach. Therefore, we forced the positioning of the fingertips at desired locations for both the real objects and the visual tracking system. Indeed, to allow arbitrary finger placement, the experimental apparatus should be covered with an array of high-precision 3D

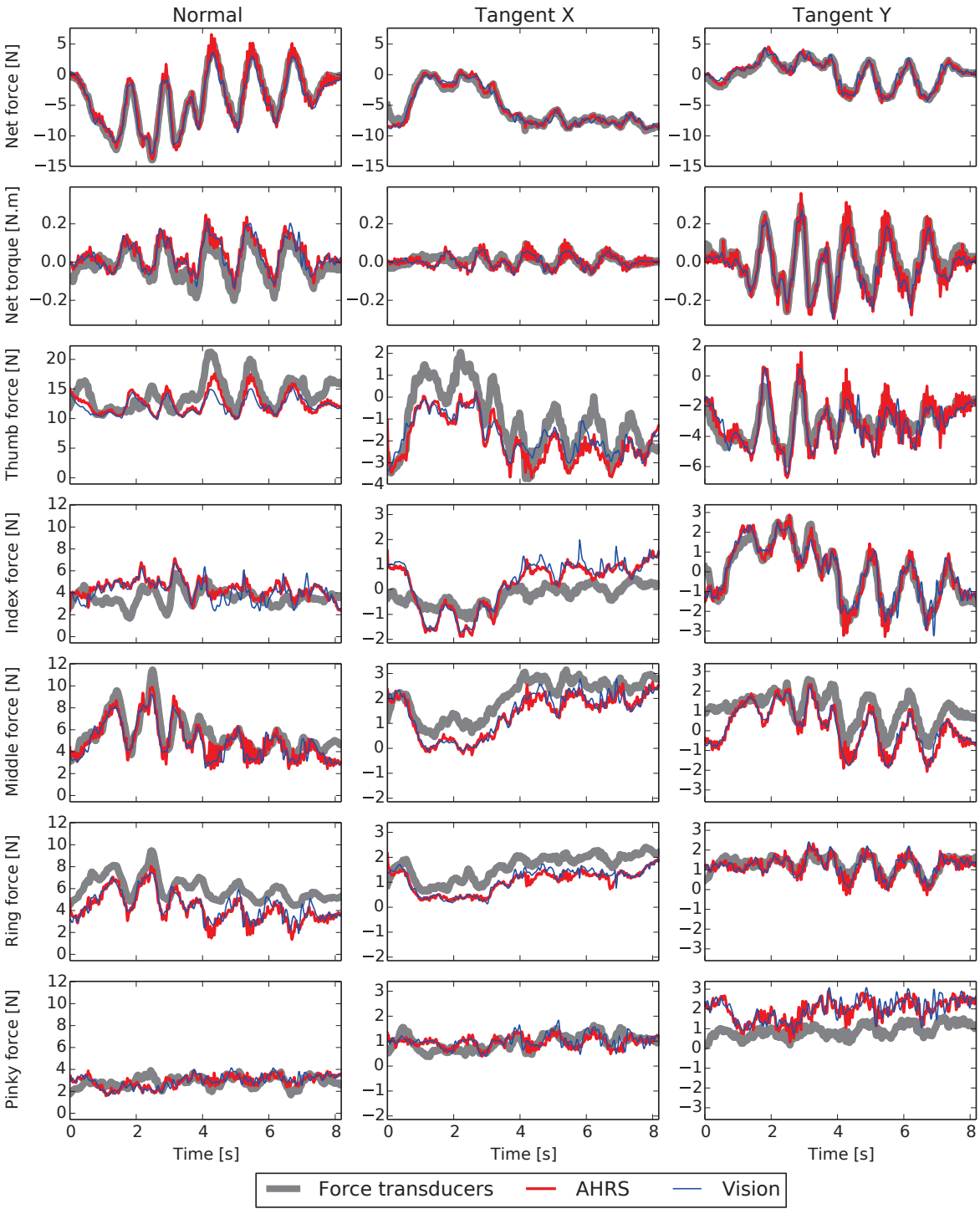


Figure 3.8: Force estimates from AHRS measurements and visual tracking with closed-loop KDN-VF-F and random initialization.

force transducers (that are not available in the required dimensions), or alternatively with

Table 3.5: Force Estimation Errors From Visual Tracking

Kinematics	AHRS	AHRS	Vision
Initialization	ground truth	random	random
KDN-FH-F, PP	$-1.10\,(2.95)$	$-1.12\,(2.95)$	$-1.18\,(3.11)$
KDN-FH-F, CL	$-1.37\,(3.12)$	$-1.37\,(3.13)$	$-1.25\,(3.61)$
KDN-FH-Δ, PP	$0.72\,(3.38)$	$0.85\,(3.42)$	$0.94\,(3.39)$
KDN-FH-Δ, CL	$1.21\,(5.80)$	$2.27\,(11.86)$	$3.50\,(17.28)$
KDN-VF-F, PP	$0.18\,(2.64)$	$0.14\,(2.68)$	$0.15\,(2.69)$
KDN-VF-F, CL	$\mathbf{-0.01}\,(\mathbf{2.20})$	$\mathbf{0.02}\,(\mathbf{2.27})$	$\mathbf{-0.04}\,(\mathbf{2.30})$
KDN-VF-Δ, PP	$5.40\,(27.61)$	$5.16\,(23.06)$	$5.94\,(24.54)$
KDN-VF-Δ, CL	$2.20\,(16.31)$	$3.87\,(19.99)$	$7.37\,(25.15)$

dedicated force sensing surfaces [SCCP14], generally limited in accuracy and range (e.g., normal forces only).

Our force estimation framework can readily challenge in-hand manipulation scenarios with more sophisticated tracking systems (e.g., multi-camera). Again, assessing such tasks is limited by the difficulty of measuring the actual forces without obstructing the subject's haptic sense, which we consider essential in our demonstration. In effect, the tracking method we describe does not introduce any constraint besides those relative to the ground-truth instrumentation, while making it possible to monitor manipulation forces using a single off-the-shelf depth sensor.

3.6.2 Beyond Prismatic Grasps

For the sake of completeness, we evaluate the force estimation framework on a non-prismatic grasp. We construct a mug-shaped instrumented device, pictured in Fig. 3.7d, and arrange the force transducers on a circle, with the contact normals pointing towards the center. We then compute force distributions from visual tracking and AHRS measurements using the model trained on prismatic grasps. We depict the resulting predictions in Fig. 3.9. We observe the following. First, by considering the hand and the object as a single rigid compound, we are able to track the mug fairly accurately using a single depth sensor, despite it being essentially rotationally symmetric, except for a handle that is easily occluded. Second, in general, the RNN predictions do not follow the subtle force variations along the normal $\mathbf{n}_k$ and tangential directions $\mathbf{t}_k^x$ as closely as the tangential directions $\mathbf{t}_k^y$. Indeed, recall from Section 3.3.2 that the individual $\mathbf{t}_k^y$ per finger are defined, uniformly, as oriented towards the

palm. This property is preserved in the case of the mug. However, while for prismatic grasps the $\mathbf{n}_k$ are collinear with each other and perpendicular to the $\mathbf{t}_k^x$, couplings appear between and among each set in the case of the mug. Still, although RNN predictions and force transducer measurements can quite differ, the SOCP ensures that the final distributions are physically plausible based solely on the observed kinematics and the object-grasp properties, regardless of the RNN training dataset.

While we could imagine extending the force estimation framework further by training new network architectures on arbitrary grasps, this is difficult in practice. The ground-truth instrumentation used in the manipulation kinodynamics dataset captures 11 degrees of freedom for the contact space (grasp width and 2D tangential position of each finger on the tangential space). In contrast, for general grasps, the instrumentation should allow 25 degrees of freedom (5 per finger, ignoring the transducer orientations about the normal axes). Due to a greater contact space dimensionality, it would require significantly more experiments to obtain a dataset that is both diverse and extensive, as well as a much heavier experimental setup to be able to fine-tune the position and roll-pitch of each transducer independently.

3.6.3 Computational Performance

On a computer equiped with an Intel i7-4700MQ CPU (quad-core 2.40GHz) and an NVIDIA GTX 780M GPU, we apply the KDN-VF-F closed-loop architecture on the testing dataset (39 experiments, total duration 2470 s, 60 samples per second). We report the computation time in Table 3.6. While at first the computation time appears greater than the dataset duration, the decomposition per process shows that the current implementation is actually rather sub-optimal. In fact, the three core components of our approach take only 5.29 ms per sample. First, algebraic differentiators implemented as finite impulse response filters are of minor impact on the computation time. Second, RNN predictions are parallelized on the GPU using the Torch7 framework [CKF11]. Third, SOCP solving is done with the CVXOPT library [ADV13].

In the current implementation, we construct the RNN input vectors and SOCP constraint matrices within their respective frameworks. A typical iteration is as follows:

1. Given the current kinematics and the SOCP corrected forces $\mathbf{F}_{i-1}$ at the previous step, we construct the RNN input vector $(\mathbf{K}_i, \mathbf{D}_{i-1})$.

2. The network produces a raw force prediction $\mathbf{D}_i^{(\text{raw})}$.

3. We assemble SOCP constraint matrices from the target kinematics and the cost function from $\mathbf{D}_i^{(\text{raw})}$.

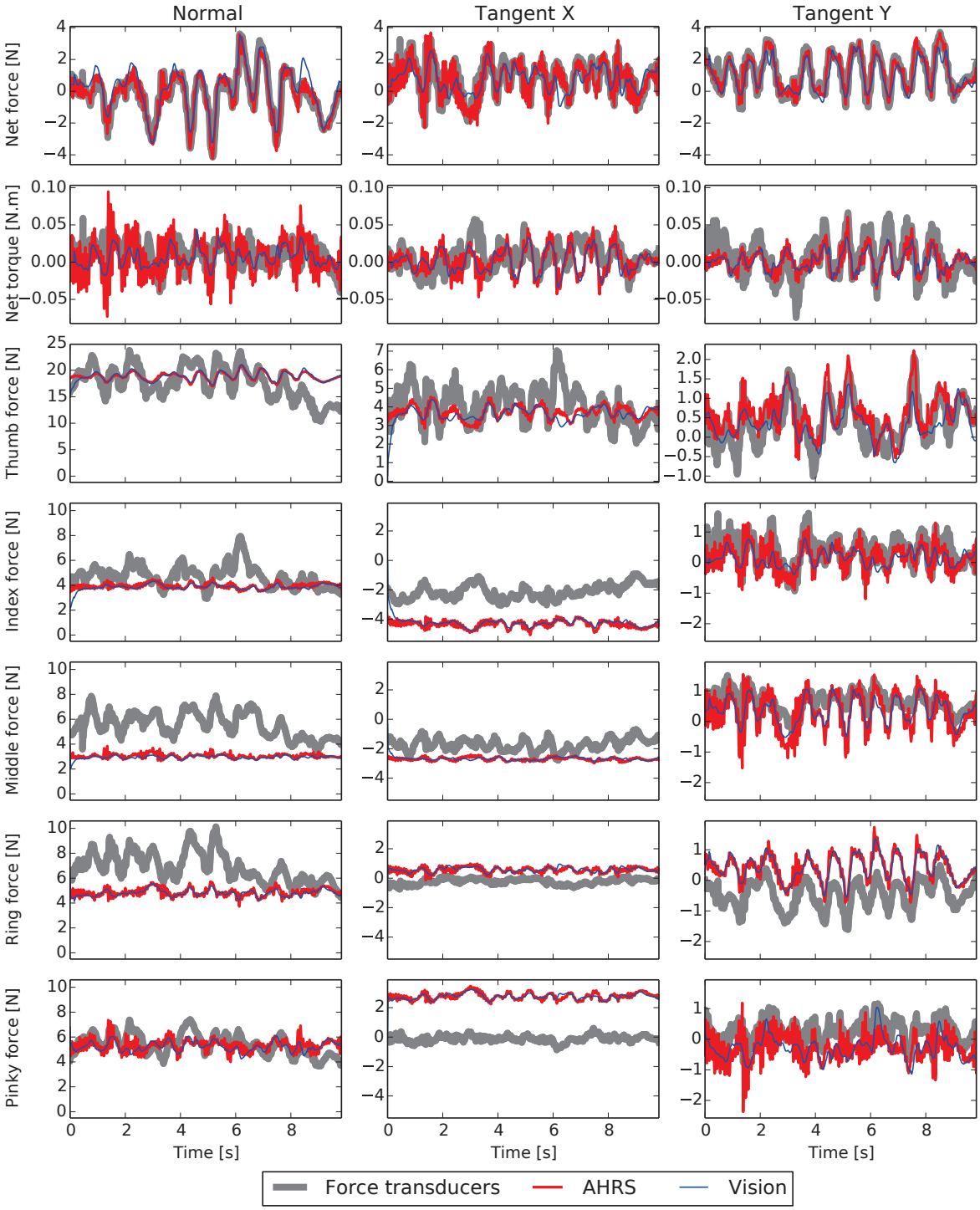


Figure 3.9: Force estimates with non-prismatic grasp (mug).

4. We solve the SOCP and get the corrected forces $\mathbf{F}_i$.

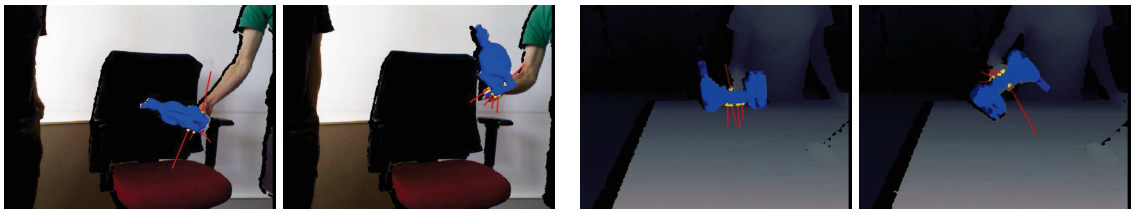
Table 3.6: Computation Time Decomposition by Process

	Total	Per sample	Per timestep
Experiment duration	2470.0 s	16.67 ms	100.00 %
Computation time	3521.4 s	23.76 ms	142.57 %
Algebraic diff.	22.3 s	0.15 ms	0.90 %
RNN prediction	120.4 s	0.81 ms	4.87 %
↪ Data formatting	86.2 s	0.58 ms	3.49 %
SOCP correction	641.8 s	4.33 ms	25.98 %
↪ Initialization	659.0 s	4.45 ms	26.68 %
Lua/Python bridge	1991.7 s	13.44 ms	80.64 %

Steps 1 and 2 are executed in Lua for Torch7, while steps 3 and 4 are executed in Python for CVXOPT. Both being interpreted languages explains part of the overhead in preparing the data for each process. However, the majority of the computation time is actually spent on managing the two interpreters in succession, as represented by the Lua/Python bridge value in Table 3.6, which measures the time elapsed between steps 2 and 3, and between steps 4 and 1 (next iteration). Note that no calculation is performed during that time, only spent on switching between Lua and Python contexts. For this reason, simply implementing our method within a unified computational framework would certainly yield a tremendous increase in performance enabling real-time use. Other possible improvements at the numerical level include refactoring data structures to reduce redundancies and update constraint matrices only when needed, initializing the SOCP search at the RNN predictions, and rewrite the physical plausibility problem as a quadratic program (QP) using a discretized friction cone.

3.7 Conclusion and Future Work

Our work establishes that monitoring hand-object interaction forces at the fingertip level, a problem that is traditionally addressed with costly, cumbersome and intrusive force transducers, can be addressed in a cheap, reliable and transparent way using vision. Based on the first large-scale dataset on manipulation kinodynamics, the approach we present estimates force distributions that are compatible with both physics and real human grasping patterns. While the case of static prismatic grasps may appear restrictive, this limitation is only relative to the instrumentation required to collect ground-truth measurements, essential to prove the validity of the approach. Provided such an experimental setup, we expect that our method can be seamlessly extended to arbitrary grasps. Note that, even without, the current SOCP



(a) 'Samurai' sequence, overlay on registered color. (b) 'Drill' sequence, overlay on depth map.

Figure 3.10: Qualitative force predictions (red) with manually picked contact points (yellow) on alternative object tracking datasets: (a) [KMB+14], (b) [IWGC+16].

formulation is independent of the dataset used to train the networks and always produces distributions that result in the observed motion. Finally, even limited to prismatic grasps, the estimation of 3D forces for all five fingers on arbitrary motions greatly extends the state of the art in interaction capture. Using our approach, it is achieved with a setup as simple as a single RGB-D camera, enabling its use for monitoring of human activities and robot learning from demonstration in daily settings.

Our approach is readily compatible with any method providing accurate object kinematics. For qualitative evaluation, we applied our technique to the alternative object trackers and datasets of [KMB+14, IWGC+16] with the contact points handpicked from the visual observations. We depict snapshots of these experiments in Fig. 3.10 and invite the reader to refer to the supplementary material of [PKAK16] for video results[2]. When the situation allows a richer setup, a multi-camera system can also be used to track the hand and the object separately. Our future work involves alleviating the limitations induced by the ground-truth instrumentation. In order to monitor non rigid grasps, we aim to apply the force estimation framework in conjunction with tracking to guide the pose search as an implicit model for grasp plausibility and realism [PKQA15a]. Additionally, the generalization to arbitrary grasps could be addressed by considering the variability of manipulation forces with grasp and object properties as an inverse optimal control problem. The manipulation kinodynamics dataset could thus be used to refine the force optimization problem with physiological criteria, e.g., grasp efficiency [ZY13]. In the long term, we plan to extend the force estimation framework to general articulated bodies for bi-manual grasping. In Chapter 4, we extend our approach to whole-body interactions with the environment.

[2]https://www.youtube.com/watch?v=NhNV3tCcbd0

Acknowledgments

The work presented in this chapter was partially supported by the FP7 EU RoboHow.Cog project and the Japan Society for the Promotion of Science (JSPS): Kakenhi B No. 25280096. It was submitted as a journal paper [PKAK16], in collaboration with Nikolaos Kyriazis from FORTH-ICS and Antonis A. Argyros from FORTH-ICS and the University of Crete.

Chapter 4

Whole-Body Contact Force Sensing From Motion Capture

4.1 Introduction

Humans purposefully interact with their environment through physical contact to manipulate and move themselves or objects. The contact forces that are applied during a given task are informative on both the resulting motion and the underlying intent. Thus, force sensing has direct applications in research fields such as robot learning from demonstration and control [RJC13, EKO15], physics-based animation [HBL11, ZSZ+14] and visual tracking [KA13, PKQA15a]. Contact forces are typically measured using force transducers that are costly, cumbersome and of limited, varying accuracy under repeated stress [DMVS10]. In this work, we propose a method to infer human whole-body contact forces from motion capture alone. If combined with markerless visual tracking technologies [MPA15], this would enable the non-intrusive monitoring of contact forces in daily activities. However, the problem is very challenging.

By means of the equations of motion for articulated rigid bodies, the knowledge of external and internal forces uniquely determines the resulting kinematics. In contrast, the reverse problem is generally indeterminate in multi-contact with the environment and the knowledge of a given motion may not suffice to fully characterize the underlying force distribution. For instance, one can stand still while applying foot forces of varying magnitude in opposite, lateral directions. The force distribution problem in whole-body locomotion is an active research topic in multiple fields (Section 1.4.2). In Chapters 2 and 3 we proposed a combined optimization and learning approach for force sensing from vision in the context of manipulation. However, these approaches do not directly extend to the case of whole-body

multi-contact with the environment due to, in particular, the higher dimensionality of the human body and the variety of possible contact configurations. We address it as follows:

- Akin to our data-driven approach for manipulation, we collect real measurements on whole-body kinodynamics, in the form of 100 min of motion and force measurements for diverse contact configurations (Section 4.2).
- We propose a force estimation framework relying jointly on a recurrent neural network that learns how humans instinctively distribute contact forces while accounting for varying multi-contact configurations, and a second-order cone program that guarantees the physical plausibility of the resulting distributions with respect to the whole-body equations of motion (Section 4.3).
- We consistently validate our approach with ground-truth measurements throughout our work and demonstrate its accuracy on challenging scenarios (Section 4.4).

Finally, we discuss the current limitations of our work as well as possible applications and extensions (Section 4.5). To accelerate the research on this new topic and encourage alternative implementations, we make our datasets and algorithms publicly available [1] .

4.2 Whole-Body Kinodynamics Dataset

4.2.1 Experimental Setup

We collect kinodynamic measurements (motion and forces) on human activities using two types of sensors in parallel. First, the human whole-body motion is tracked using a motion capture system (Xsens MVN Awinda) consisting of 17 wireless inertial measurement units (IMU) and batteries strapped at specified body landmarks. The choice of this motion capture technology is motivated by our intention to collect human kinodynamic measurements in confined and eventually outdoor environments. Vision-based systems (e.g., Vicon) are limited by strong occlusions occurring in whole-body interactions with the environment, and difficult to apply in uncontrolled environments on the fly (e.g., outdoor).

The motion of the subject's body, modeled as a 23-segment skeleton, is recorded at 100 Hz. For each sample, the system provides the 6-DoF pose of each segment as well as the corresponding linear and rotational velocity and acceleration. Contact forces at the subject's feet are monitored with instrumented shoes (Xsens ForceShoe), equipped with 6-DoF force-torque sensors at the heel and toes and IMUs recording the sensor orientations. We measure

[1] The dataset and algorithms will be released at https://github.com/jrl-umi3218/WholeBodyKinodynamics.

(a) Motion capture suit, contact force (red) and torque (yellow) visualization.

(b) Shoes equipped with inertial measurement units and force-torque sensors.

Figure 4.1: Acquisition system for whole-body kinematics and contact forces.

other interaction forces with the environment using an additional 6-DoF force-torque sensor (ATI Mini-45) held in the subject's hand. All force-torque measurements are recorded at 100 Hz. We depict our acquisition setup in Fig. 4.1.

Being based on inertial measurements, the motion capture system is prone to drift compared to marker-based tracking methods (e.g., Vicon). We are working on a solution to attenuate this problem. Similarly, wearable force sensors can be of lower accuracy than force plates due to repeated pressure and deformations. Still, a major benefit of our lightweight setup is the efficient and continuous acquisition of kinematics and contact forces on highly-dynamic motions through time, which is generally not possible with static force plates. Additionally, the simultaneous monitoring of the whole-body motion and forces allows their correction in two steps. First, low-frequency sensing inaccuracies (e.g., drift) for both

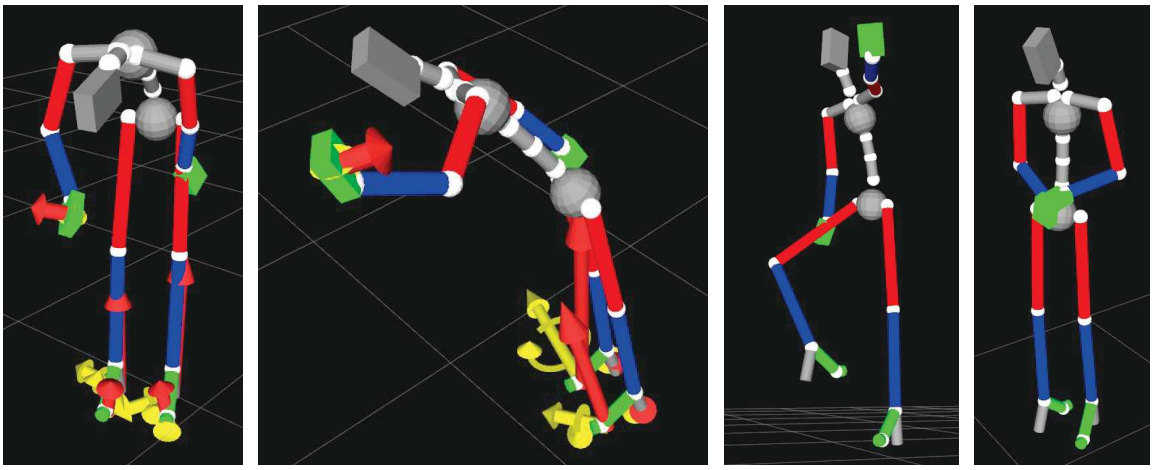
types of sensors can be corrected in isolation, based on physical considerations described in Section 4.2.2. Second, physical consistency between whole-body kinematics and contact force measurements can be enforced through the equations of motion for articulated systems of rigid bodies (see Section 4.3.1).

4.2.2 Preparing Measurements for Dynamics Analysis

The Xsens MVN Awinda system captures the motion of the subject using a 23-segment skeleton. At each time step, the whole-body pose is encoded by 161 parameters, i.e., 7 parameters per segment (3D position and orientation as quaternion). To facilitate the dynamics analysis, we transform the motion capture output into a kinematic tree rooted at the subject's pelvis and link the 23 segments with 22 spherical joints allowing 3 rotational degrees of freedom. In practice, we represent spherical joints with 3 chained revolute joints for compatibility with the Unified Robot Description Format (URDF)[2]. Thus, the whole-body pose is summarized, without loss, as a 73-element vector $\mathbf{q}$ containing the quaternion and position of the base link in the global frame and 66 joint angles. We augment the kinematic tree with body segment inertial parameters (i.e., mass, center of gravity, inertia tensor) computed from the subject's weight and measurements with the anthropomorphic tables of Dumas et al. [DCV07]. The resulting dynamic model is implemented using the RBDyn library for rigid body dynamics [V$^+$13]. Corrections based on the equations of motion are discussed in Section 4.3.1. Prior to that, motion capture and force sensing errors can be mitigated based on the following considerations.

Recall that the whole-body tracking system employed in this work is based on inertial measurement units. Positions and orientations are computed by sensor fusion and successive integrations of acceleration and velocity measurements, making the pose estimation sensitive to drift over time. As such, contact configurations cannot be directly identified based solely on the reported body segment positions. For instance, we found that contacts between the feet and the ground during walking could not be consistently segmented by simple thresholding on their vertical positions. Instead, we thresholded the force sensor measurements to know when a contact occurs. Noticeable pose estimation errors also appeared when considering contacts others than with the ground, e.g., between the hand and the environment, as depicted in Figs. 4.2a and 4.2b. In such cases, we manually constrained the palm to be either horizontal or vertical. We also observed body segments in unrealistic orientations, illustrated in Figs. 4.2c and 4.2d, due to the motion capture system not enforcing joint limits in the skeletal tracking. We implemented them manually in our kinematic model and limited the range of the revolute

[2]http://wiki.ros.org/urdf



(a) Contact with table. (b) Contact with wall. (c) Raised right foot. (d) Static stance.

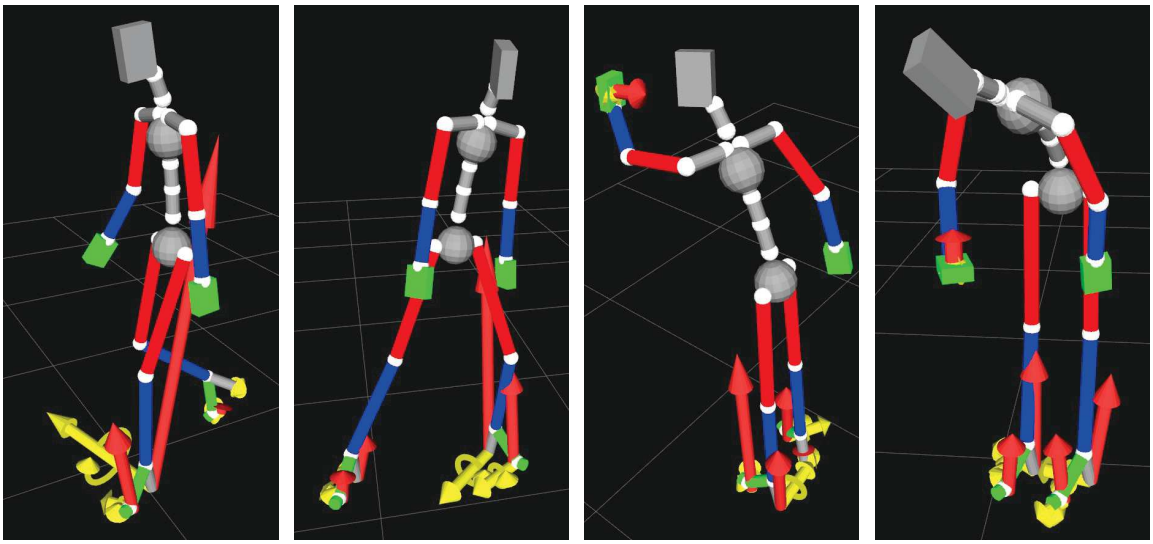
Figure 4.2: Erroneous tracking examples. (a): against a table, the right hand should be horizontal with the contact normal pointing upwards. (b): against a wall, the hand should be vertical with the contact normal in the horizontal plane. (c): right foot flipped backwards when raised on a foot stand. (d): foot orientation drift with subject standing still.

joints. Note that this material limitation does not affect the generality of our approach and can be fully cirvumvented with additional visual observations, at the cost of portability and flexibility for the experimental setup.

Finally, the force measurements are subject to noise, either from the sensors themselves or due to interferences in the wireless transmission. We attenuate it by smoothing all signals with a Gaussian filter of kernel $\sigma = 0.05\,\mathrm{s}$. Second, a slow-varying bias can appear in the force-torque measurements due to repeated stress and battery drain. We compute the bias through time by averaging the signals that persist when a sensor is not in contact with the environment, which should only be caused by the inertia of the moving parts attached to the sensing surface (e.g., force shoe external sole).

4.2.3 Experiments and Data Collection

In a preliminary study, four male volunteers took part as subjects in our experiments. Their weights (between 69.6 kg and 79.8 kg, plus the 3.5 kg acquisition system), heights (between 1.79 m and 1.94 m), and individual body segment lengths were measured to initialize the motion capture skeletal tracking model and BSIPs following [DCV07]. All sensors (motion and force-torque) were calibrated and reset between experiments following the manufacturers' recommended acquisition procedure to reduce the effects of measurement drift and hysteresis. The subjects were instructed to perform the following tasks:



(a) Walking. (b) Upper body balancing. (c) Wall support. (d) Leaning on a table.

Figure 4.3: Sample poses from the whole-body kinodynamics dataset.

- Walking at different speeds (slow, normal, fast) and following different trajectories (circular, back and forth)
- Moving the upper body while maintaining the feet static
- Taking support against a wall with the left or right hand
- Leaning on a table with the left or right hand

We illustrate these experiments in Fig. 4.3. The goal of the first task is to allow neural networks to capture the centroidal dynamics relationship between motion and forces in bipedal contact. The second task follows the same principle and also provides examples of static indeterminacy, i.e., how humans apply forces that cancel each other out and do not affect their state of equilibrium. The third and fourth tasks go further and are typical scenarios where the straightforward minimization of multi-contact forces leads to distributions that are physically plausible but not representative of those humans really apply, as discussed in Section 4.3.

For each experiment, we record the 6-DoF pose of the 23 segments through time as estimated by the motion capture suit and convert it to an equivalent 73-element vector of generalized coordinates $\mathbf{q}$. The inertial motion capture system readily provides the angular and linear velocities and accelerations of each segment in the global frame. We construct the whole-body velocity (resp. acceleration) vector $\dot{\mathbf{q}}$ (resp. $\ddot{\mathbf{q}}$) from the angular and linear velocities (resp. accelerations) of the base link in the global frame and from the joint

velocities (resp. accelerations) of the children segments, computed by projecting the segment angular velocities (resp. accelerations) onto the individual parent frames. We then perform a preliminary correction of both the motion capture and force-torque sensor measurements as described in Section 4.2.2. Overall, we construct a dataset of total duration 100 min comprising synchronized motion and force-torque measurements on 51 experiments.

4.3 Force Sensing From Whole-Body Motion

4.3.1 Whole-Body Force Optimization

We consider an articulated system of rigid bodies subject to $N_{\boldsymbol{\tau}}$ internal joint torques $\boldsymbol{\tau}^{(i)}$:

$$\boldsymbol{\tau}^{(i)} = \left(\tau_1^{(i)}, \ldots, \tau_{N_{\boldsymbol{\tau}}}^{(i)}\right)^T \tag{4.1}$$

and $N_{\mathbf{F}}$ external wrenches $\mathbf{F}_k = (\boldsymbol{\tau}_k, \mathbf{f}_k)$, with $\boldsymbol{\tau}_k$ and $\mathbf{f}_k$ the respective external torque and force at contact k, expressed in the global frame. Considering the (free) position and orientation of the base link, the number of degrees of freedom is $N_{DoF} = N_{\boldsymbol{\tau}} + 6$. We denote by $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$ the respective generalized coordinates, velocity and acceleration of the articulated system. The whole-body equations of motion can be expressed as:

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \mathbf{0}_6 \\ \boldsymbol{\tau}^{(i)} \end{bmatrix} + \sum_{k=1}^{N_{\mathbf{F}}} \mathbf{J}_k^T \mathbf{F}_k, \tag{4.2}$$

with:

- $\mathbf{H}(\mathbf{q})$ the $N_{DoF} \times N_{DoF}$ mass matrix,
- $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ the $N_{DoF} \times 1$ bias vector of the Coriolis, centrifugal forces and gravity terms,
- $\mathbf{J}_k$ the $N_{DoF} \times 6$ Jacobian matrix at contact k,
- $\mathbf{0}_6$ the 6×1 internal wrench directly applied at the base link (zero for floating base).

For each contact k, we denote by $\mathbf{z}_k$ the (uniquely defined) normal vector oriented from the environment to the body and choose two orthogonal vectors $\mathbf{x}_k$ and $\mathbf{y}_k$ in the tangential plane. We can thus express the external wrench $\mathbf{F}_k$ in the contact space $\mathscr{C}_k$:

$$\begin{aligned} {}^{\mathscr{C}_k}\mathbf{F}_k &= \left(\tau_k^{\mathbf{x}}, \tau_k^{\mathbf{y}}, \tau_k^{\mathbf{z}}, f_k^{\mathbf{x}}, f_k^{\mathbf{y}}, f_k^{\mathbf{z}}\right)^T, \\ \text{with} \quad &\begin{cases} \boldsymbol{\tau}_k = \tau_k^{\mathbf{x}}\mathbf{x}_k + \tau_k^{\mathbf{y}}\mathbf{y}_k + \tau_k^{\mathbf{z}}\mathbf{z}_k \\ \mathbf{f}_k = f_k^{\mathbf{x}}\mathbf{x}_k + f_k^{\mathbf{y}}\mathbf{y}_k + f_k^{\mathbf{z}}\mathbf{z}_k \end{cases}. \end{aligned} \tag{4.3}$$

Having chosen $\mathbf{z}_k$ oriented towards the body, the normal force component is such that:

$$f_k^{\mathbf{z}} \geq 0. \tag{4.4}$$

The Coulomb model with friction coefficient μ_k requires that:

$$\left\| f_k^{\mathbf{x}}\mathbf{x} + f_k^{\mathbf{y}}\mathbf{y} \right\|_2 \leq \mu_k f_k^{\mathbf{z}}. \tag{4.5}$$

Eq. (4.2), (4.4) and (4.5) can be respectively incorporated as equality, linear inequality and cone constraints of a second-order cone program (SOCP) of optimization parameters:

$$\begin{aligned} \mathbf{x} &= \left(\boldsymbol{\tau}^{(i)}, \left({}^{\mathscr{C}_k}\mathbf{F}_k \right)_{k=1,N_{\mathbf{F}}} \right), \\ &= \left(\tau_1^{(i)}, \ldots, \tau_{N_{\boldsymbol{\tau}}}^{(i)}, \tau_1^{\mathbf{x}}, \tau_1^{\mathbf{y}}, \tau_1^{\mathbf{z}}, \ldots, f_{N_{\mathbf{F}}}^{\mathbf{x}}, f_{N_{\mathbf{F}}}^{\mathbf{y}}, f_{N_{\mathbf{F}}}^{\mathbf{z}} \right)^T. \end{aligned} \tag{4.6}$$

$\mathbf{x}$ is a vector of size $N_{\boldsymbol{\tau}} + 6N_{\mathbf{F}}$.

4.3.2 Force Correction and Reconstruction

From this formulation, it is directly possible to construct physically plausible force distributions by minimizing a cost function depending only on the optimization parameters, e.g., the (squared) L^2 norm of the internal and external wrenches:

$$\begin{aligned} \mathscr{C}_{L^2}(\mathbf{x}) &= \|\mathbf{x}\|_2^2, \\ &= \left\| \boldsymbol{\tau}^{(i)} \right\|_2^2 + \sum_{k=1}^{N_{\mathbf{F}}} \|\mathbf{F}_k\|_2^2. \end{aligned} \tag{4.7}$$

The resulting forces, by construction, are necessarily compatible with the observed motion. However, in multi-contact, when there exists more than a single distribution for a given task, there is no guarantee that the L^2-optimal distribution coincides with the actual forces being applied. As discussed in Section 1.4.2, the identification of the cost function supposedly optimized by the human central nervous system is still an active research topic. In our work, rather than trying to reconstruct an explicit formulation of the force distribution cost function, we instead propose to build an implicit model relying on machine learning techniques to capture how humans naturally distribute interaction forces in multi-contact.

Recall that due to measurement uncertainties, the preliminary correction process described in Section 4.2.1 does not fully ensure that the force-torque measurements are physically consistent with the motion. For this purpose, we formulate a cost function for the SOCP that

quantifies the distance between the external wrenches to optimize $\mathbf{F}_k$ and target wrenches $\widetilde{\mathbf{F}}_k$:

$$\mathscr{C}_{\text{disc}}(\mathbf{x}) = \varepsilon \left\| \boldsymbol{\tau}^{(i)} \right\|_2^2 + \sum_{k=1}^{N_{\mathbf{F}}} \left\| \mathbf{F}_k - \widetilde{\mathbf{F}}_k \right\|^2, \tag{4.8}$$

with ε an optimization weight for the internal joint torques. With ε big, the SOCP searches for external wrench distributions that minimize the magnitude of the joint torques. Conversely, with ε small, we extract the force distributions that are the closest to the reference $\widetilde{\mathbf{F}}_k$ while being physically compatible with the observed motion, regardless of the joint torques. Experimentally, we set $\varepsilon = 10^{-6}$, non-zero for numerical resolution of the SOCP with the CVXOPT library for convex optimization [ADV13].

By taking for $\widetilde{\mathbf{F}}_k$ the force-torque sensor measurements acquired experimentally, the resulting cost function can be used to mitigate sensing uncertainties by extracting physically correct force distributions in the vicinity of the uncertain measurements. Alternatively, in Section 4.3.4, we take for $\widetilde{\mathbf{F}}_k$ force-torque predictions estimated by a neural network based on whole-body kinematic observations and correct them with the same SOCP formulation. Fig. 4.4 illustrates how the accumulation of individual measurement errors over all force sensors can result in a measured net force that is in considerable disagreement with physics. In the following, we denote by ground truth the physically realistic distributions obtained by correcting the sensor measurements with the SOCP (relative to the dynamic model).

4.3.3 Learning Features

Our goal is to construct a mapping $\mathscr{F}$ between a set of input features $\mathbf{K}$ representing the whole-body kinematics and contact configuration, and output features $\mathbf{D}$ representing the underlying dynamics, i.e., external wrenches:

$$\mathbf{D} = \mathscr{F}(\mathbf{K}). \tag{4.9}$$

For the sake of generality, we aim at modelling human force distribution patterns based on an optimal selection of high-level features rather than a large set of hand-engineered parameters. Akin to the case of manipulation, we select those based on their contributions on the equations of motion. However, while the Newton-Euler equations for rigid bodies allow the extraction of relevant parameters from a limited set of well-identified physical quantities (see Section 3.3.2), the complete equations of motion for articulated bodies are significantly more complex (see Eq. (4.2)).

Instead of the complete equations of motion, we consider, for feature extraction, the Newton-Euler equations for centroidal dynamics. For each element s of the set of $N_{\mathscr{S}} = 22$

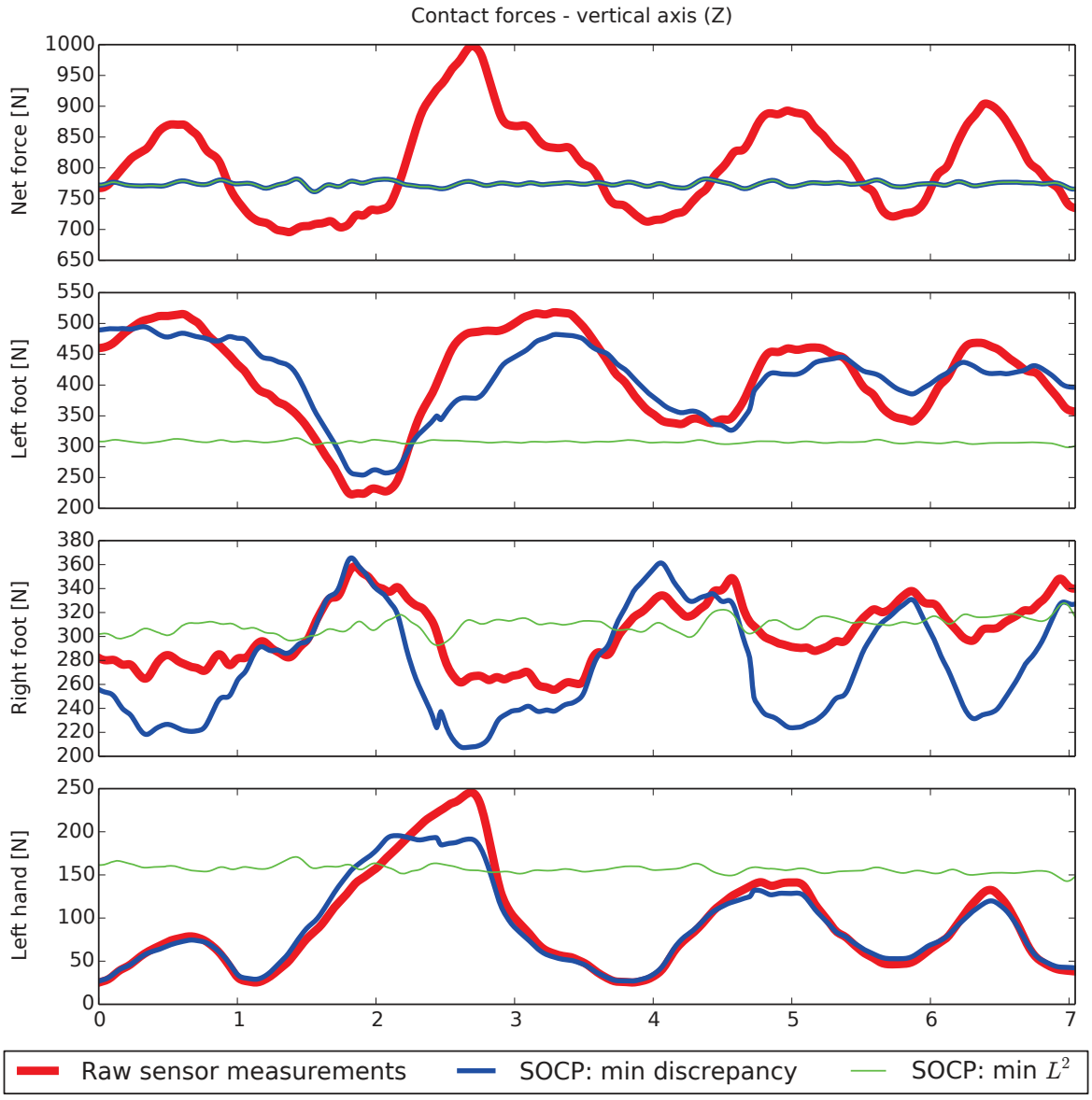


Figure 4.4: In this sequence, the subject stays still while applying varying forces in triple contact with the environment. The equations of motion dictate that the net contact force should be constant (top row), which is not apparent on the force sensor measuments (red line) due to sensing uncertainties. Forces compatible with the observed kinematics can be computed using an SOCP (green and blue lines). The minimization of the L^2 norm alone yields forces that are physically plausible but differ significantly from the measurements. Instead, minimizing the discrepancy to the uncertain measurements yields forces that are realistic both physically and compared to actual distributions.

body segments $\mathscr{S}$, we denote by m_s its mass and $\mathbf{G}_s$ its CoM. In the global frame, we denote by $\mathbf{v}_s$ the linear velocity of $\mathbf{G}_s$ and $\mathbf{R}_s$ its orientation matrix. In the segment frame, we denote by $\boldsymbol{\omega}_s$ and $\mathbf{I}_s$ its local angular velocity and inertia tensor, respectively. With m the total mass

of the articulated system and **G** its centroid, the linear momentum $\mathscr{P}$ and angular momentum $\mathscr{L}_{\mathbf{G}}$ at **G** are defined by:

$$\begin{cases} \mathscr{P} = \sum_{s \in \mathscr{S}} m_s \mathbf{v}_s, \\ \mathscr{L}_{\mathbf{G}} = \sum_{s \in \mathscr{S}} m_s \overrightarrow{\mathbf{GG}_s} \times \mathbf{v}_s + \mathbf{R}_s \mathbf{I}_s \boldsymbol{\omega}_s. \end{cases} \tag{4.10}$$

With $\dot{\mathscr{L}}_{\mathbf{G}}$ and $\dot{\mathscr{P}}$ the time derivatives of the angular and linear momenta, respectively, $\mathbf{g}$ the gravity vector and ${}^{\mathbf{G}}\mathbf{F}_k$ the contact wrench at contact k transformed to **G**, the Newton-Euler equations for centroidal dynamics state that:

$$\begin{bmatrix} \dot{\mathscr{L}}_{\mathbf{G}} \\ \dot{\mathscr{P}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ m\mathbf{g} \end{bmatrix} + \sum_{k=1}^{N_{\mathbf{F}}} {}^{\mathbf{G}}\mathbf{F}_k. \tag{4.11}$$

We gather gravity, linear and angular momenta as a gravito-inertial wrench $\mathbf{w}^{(gi)}$ [CPN16]:

$$\mathbf{w}^{(gi)} = \begin{bmatrix} \dot{\mathscr{L}}_{\mathbf{G}} \\ \dot{\mathscr{P}} - m\mathbf{g} \end{bmatrix} \tag{4.12}$$

Denoting by $\mathbf{P}_k$ the location of contact k, Eq. (4.11) can thus be rewritten as:

$$\mathbf{w}^{(gi)} = \sum_{k=1}^{N_{\mathbf{F}}} \begin{bmatrix} \boldsymbol{\tau}_k + \overrightarrow{\mathbf{GP}_k} \times \mathbf{f}_k \\ \mathbf{f}_k \end{bmatrix}. \tag{4.13}$$

The left-hand side of Eq. (4.13) is a purely kinematic term that can be directly computed from the whole-body pose and its derivatives, while the right-hand side summarizes the contributions of each contact wrench. This representation makes it a good candidate for the selection of optimal learning features extracting the gist of locomotory dynamics.

Recall that the whole-body motion is expressed in the world frame. To account for translational and rotational invariances, we extract and express learning features in a reference frame $\mathscr{G}$ of origin the centroid **G** and of orientation fixed with respect to a chosen body frame (e.g., that of the pelvis). Thus, walking straight to the North is perceptually equivalent to walking straight to the East. Based on Eq. (4.13), we extract as kinematics-based features:

- ${}^{\mathscr{G}}\mathbf{w}^{(gi)}$ the 6-element gravito-inertial wrench expressed in $\mathscr{G}$,
- ${}^{\mathscr{G}}\mathbf{P}_k$ the 3D position of each contact k expressed in $\mathscr{G}$.

This results in $6 + 3N_{\mathbf{F}}$ features for a motion involving $N_{\mathbf{F}}$ contacts. In practice, the mapping $\mathscr{F}$ of Eq. (4.9) may require a fixed-size input vector. Thus, the formalism adopted should

account for varying contacts. Instead of only considering the location of the contacts ${}^{\mathscr{G}}\mathbf{P}_{k,i}$ that are active at timestamp i, we monitor a set of N_c possible contact locations through time and encode contact configurations using parameters $\delta_{k,i}$ such that:

$$\delta_{k,i} = \begin{cases} 1 & \text{if contact } k \text{ is active at time step } i, \\ 0 & \text{otherwise.} \end{cases} \tag{4.14}$$

In our experiments, we continuously monitor the position of the subject's feet and hands. As discussed in Section 4.2.2, we extract the parameters δ_k by thresholding on the force sensor measurements, although this could be achieved by body and object segmentation when visual observations are available. Overall, the complete input features at timestamp i are:

$$\mathbf{K}_i = \left({}^{\mathscr{G}}\mathbf{w}_i^{(gi)}, \left({}^{\mathscr{G}}\mathbf{P}_{k,i}, \delta_{k,i} \right)_{k=1,N_c} \right)^T \tag{4.15}$$

$\mathbf{K}_i$ is a vector of size $6+7N_c$. Similarly, we define as output features the contact wrenches (i.e., forces and torques) applied at each monitored potential contact point k, expressed in $\mathscr{G}$:

$$\mathbf{D}_i = \left(\left({}^{\mathscr{G}}\mathbf{F}_{k,i} \right)_{k=1,N_c} \right)^T \tag{4.16}$$

For each monitored point k, the wrench ${}^{\mathscr{G}}\mathbf{F}_{k,i}$ is zero if k is not in contact with the environment. $\mathbf{D}_i$ is a vector of size $6N_c$.

4.3.4 Neural Network Model

In Eq. (4.9), the mapping $\mathscr{F}$ does not account for temporal continuity. As such, consecutive force distributions are independent of each other. Instead, we introduce a dependency on both the current motion and the past trajectory using the following formulation:

$$\mathbf{D}_i = \mathscr{F}\left(\left(\mathbf{K}_j \right)_{j=1,i} \right). \tag{4.17}$$

We model this time series structure using recurrent neural networks (RNN) [Elm90] with long short-term memory neurons (LSTM) [HS97]. A simple network architecture, which we denote by WBN-D (whole-body network, direct), thus consists in a simple RNN directly mapping $\mathbf{K}_i$ to $\mathbf{D}_i$ while keeping track of long-term dependencies to the past:

$$\mathbf{D}_i = \text{WBN-D}\left(\mathbf{K}_i \right). \tag{4.18}$$

A typical iteration at timestamp i is as follows:

1. from the whole-body motion, compute the kinematics-based input features $\mathbf{K}_i$
2. feed $\mathbf{K}_i$ into WBN-D, get raw predicted dynamic features $\mathbf{D}_i^{(\text{raw})}$
3. project $\mathbf{D}_i^{(\text{raw})}$ from $\mathscr{G}$ to the global frame to extract contact wrench predictions $\widetilde{\mathbf{F}}_{k,i}$

We illustrate the WBN-D architecture in Fig. 4.5a. Although the RNN is expected to implicitly capture the relationship between kinematics and forces, the raw predicted forces are not guaranteed to fully comply with the whole-body equations of motion and friction constraints. Therefore, we compute physically plausible solutions $\mathbf{F}_k$ in the vicinity of the raw wrench predictions $\widetilde{\mathbf{F}}_k$ using the SOCP of section 4.3.1, with the complete equations of motion of Eq. (4.2) and the discrepancy cost function of Eq. (4.8). This step can be done offline, after the prediction of the complete raw wrench sequence from kinematics alone.

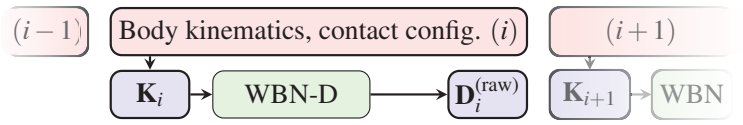
Alternatively, we propose an architecture that implements a feedback loop, WBN-F, allowing the correction of raw wrenches between consecutive predictions:

$$\mathbf{D}_i = \text{WBN-F}\left(\mathbf{K}_i, \mathbf{D}_{i-1}\right). \tag{4.19}$$

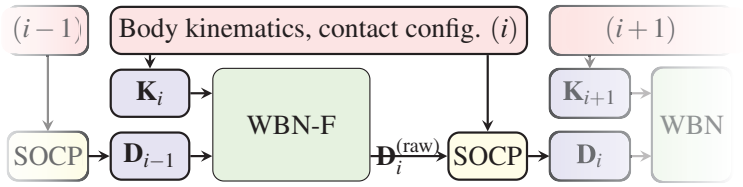
For prediction, the force distribution sequence is initialized with the distribution $\mathbf{D}_0$ of minimal L^2 norm, as described in Eq. (4.7). Such a distribution is computed from the kinematics alone. Subsequent iterations i are then as follows:

1. fetch the previous dynamic features $\mathbf{D}_{i-1}$
2. from the current whole-body motion, compute the current kinematic features $\mathbf{K}_i$
3. feed $\mathbf{K}_i, \mathbf{D}_{i-1}$ into WBN-F, get raw predicted dynamic features $\mathbf{D}_i^{(\text{raw})}$
4. project $\mathbf{D}_i^{(\text{raw})}$ from $\mathscr{G}$ to the global frame to extract contact wrench predictions $\widetilde{\mathbf{F}}_{k,i}$
5. feed $\widetilde{\mathbf{F}}_{k,i}$ into an SOCP accounting for the whole-body equations of motion
6. extract physically plausible forces $\mathbf{F}_{k,i}$ in the vicinity of the raw predictions
7. project corrected forces $\mathbf{F}_{k,i}$ from the global frame to $\mathscr{G}$ to extract dynamic features $\mathbf{D}_i$

We depict the WBN-F architecture in Fig. 4.5b.



(a) Forces are direcly computed from the kinematics and contact configuration.



(b) Force predictions are corrected between consecutive time steps.

Figure 4.5: Direct and feedback whole-body network architectures.

Table 4.1: Force Estimation Errors [N] on Testing Set (16 min)

	Raw	Corrected
Force sensors	$-4.58\,(46.1)$	ground truth
Min. L^2	N/A	$2.19\,(46.0)$
WBN-D	$0.75\,(38.4)$	$0.89\,(29.4)$
WBN-F	$1.26\,(48.1)$	$0.77\,(47.3)$

4.4 Experiments

4.4.1 Results on Complete Dataset

For the purpose of training, validation and testing, we construct a random partitioning of the whole-body kinodynamics dataset into three subsets of respective size 70 %, 15 % and 15 %. We implement the WBN-D and WBD-F neural network architectures within the Torch7 framework [CKF11]. Both architectures take the kinematics features $\mathbf{K}_i$ as input, as well as $\mathbf{D}_{i-1}$ for WBD-F, pass them into two LSTM hidden layers of size 256, and compose the results with a linear layer returning the dynamics features $\mathbf{D}_i$. We train the networks using mini-batch stochastic gradient descent with a standard regression criterion (mean square error) and dropout to avoid overfitting [SHK+14]. The SOCP correction is implemented using the CVXOPT library for convex optimization [ADV13].

In Table 4.1, we summarize the average error and standard deviation (between parentheses) between ground truth and the following force data:

- raw force sensor measurements

- forces obtained from direct L^2 norm minimization
- WBN-D outputs: raw and corrected offline
- WBN-F outputs: raw and corrected between consecutive iterations

We observe the following:

- Force-torque sensors are rather imprecise by themselves without physics-based correction (Table 4.1, first row, first column), in terms of bias (average error) and repeatibility (standard deviation).
- On the other hand, forces computed with a direct L^2 criterion also greatly differ from actual measurements (see Fig. 4.4). Thus, physics-based optimization, by itself, is not enough to capture the real forces being applied.
- Finally, the accuracy of all methods relying on learning and optimization is at least comparable (MBN-F) or significantly better (MBN-D) than that of the force-torque sensors.

Our main outcome is thus that, provided a rich dataset on human kinodynamics, the method we propose can outperform physical force sensing both in terms of accuracy and usability.

Comparing the force prediction architectures in detail, we observe that, in the absence of SOCP correction (first column), the WBN-D architectures performs better than WBN-F. This is expected, since feeding past raw predictions does not bring new information to the system. Additionally, all networks are trained on physically coherent data, i.e., input kinematics $\mathbf{K}_i$ and output dynamics $\mathbf{D}_i$ are compatible with respect to the equations of motion. However, raw neural network predictions $\mathbf{D}_i^{(\text{raw})}$ are not guaranteed to be compatible with $\mathbf{K}_i$. This directly impacts the WBN-F architecture, since possibly inaccurate force predictions are repeatedly fed back into the network (see Eq. (4.18)), which does not happen for the direct architecture WBN-D (see Eq. (4.19)). With SOCP correction (second column), we observe significant improvement in the accuracy of WBN-D but not WBN-F. A possibility is that the embedded SOCP correction interferes with the recurrent neural network prediction and internal state update processes. As future work, we aim at combining convex optimization and learning into a unified computational framework [YFW16] or guiding the neural network training with physics-based constraints [SE16].

4.4.2 Results on Restricted Training

During walking, most of the time is spent with only one foot on the ground. In single contact, the equations of centroidal dynamics, see Eq. (4.11), dictate that the contact wrench

can be uniquely identified from the body kinematics. Therefore, it may not be necessary to extensively train neural networks on such examples. Instead, the prediction accuracy may suffer if multi-contact examples (where the difficulty resides) represent a minority of the dataset. We assess this effect by training the previous neural network architectures not on the whole dataset, but on two sets containing only either walking or multi-contact examples. Both are again randomly partitioned into training (70 %), validation (15 %) and testing (15 %) subsets. We denote by WBN-D-W and WBN-F-W the respective direct and feedback architectures trained on walking examples only, and by WBN-D-M and WBN-F-M the networks trained on multi-contact examples.

We apply each network type on both its own testing set and that of the other type. We illustrate the application of WBN-D-W and WBN-D-M on a triple contact example (leaning on a table) in Fig. 4.6 and on walking in Fig. 4.7. In both cases, the raw predictions are corrected with the SOCP to ensure their physical compatibility with the observed motion. As it can be expected, the architecture trained only on walking fails at capturing the actual force distributions applied by humans in multi-contact (see Fig 4.6). In contrast, the architecture that was not extensively trained on walking accurately reconstructs contact forces even on such scenarios (see Fig. 4.7). This confirms that physics-based optimization is a valuable complement to recurrent neural networks for the latter to focus on multi-contact indeterminacy.

4.5 Discussion and Future Work

Our work introduces a novel method for the inference of contact forces applied by human subjects from their motion only. Our system estimates forces that are both physically plausible and in agreement with ground-truth measurements, even in challenging contact configurations where the force distribution problem is highly indeterminate. Trained on our (public) dataset, the neural network architectures can be applied to any centroidal representation, while the SOCP can be formulated for any articulated body. As such, our approach can be seamlessly generalized to any whole-body tracking system. Applying our method to markerless visual tracking would thus enable fully non-intrusive monitoring of whole-body forces in daily activities. Still, our approach would certainly benefit from the further collection of ground-truth force measurements on even more rich motion and contact configurations. Another possibility could be to integrate convex optimization and learning into a unified computational framework [YFW16]. In the long term, we also plan to apply our framework to force-based robot learning from demonstration, on-line multi-contact motion retargeting and knowledge-based multi-contact planning and control.

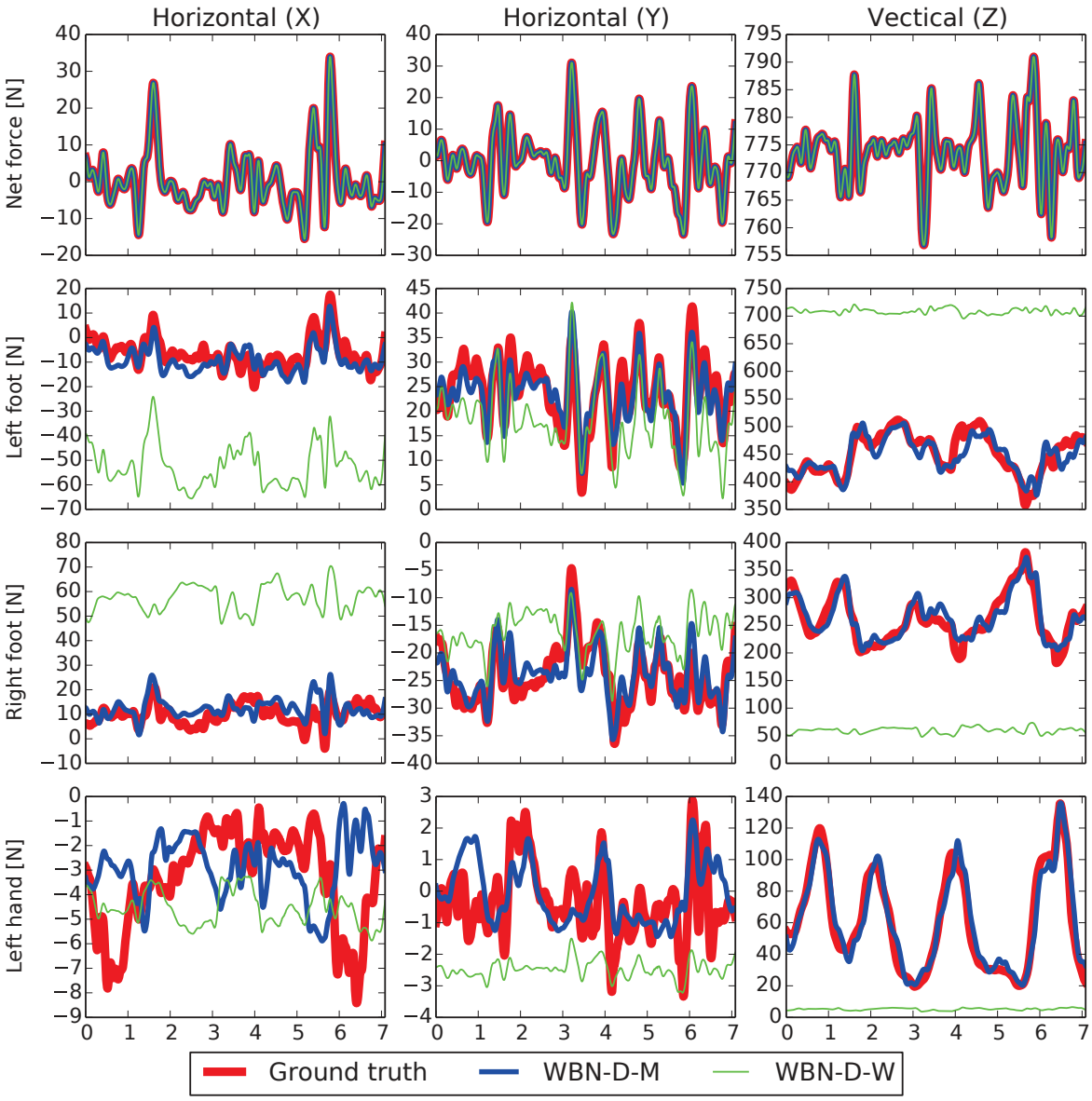


Figure 4.6: Triple contact example. Trained on similar examples, WBN-D-M successfully estimates the actual forces being applied. In contrast, WBN-D-W predicts physically valid but significantly different force distributions.

Acknowledgments

The work presented in this chapter was partially supported by the H2020 RIA COMANOID project[3] and the Japan Society for the Promotion of Science (JSPS): Kakenhi B No. 25280096. It resulted in a conference paper [PBCK16], in collaboration with Adrien Bufort and Stéphane Caron from CNRS-UM LIRMM.

[3]http://www.comanoid.eu

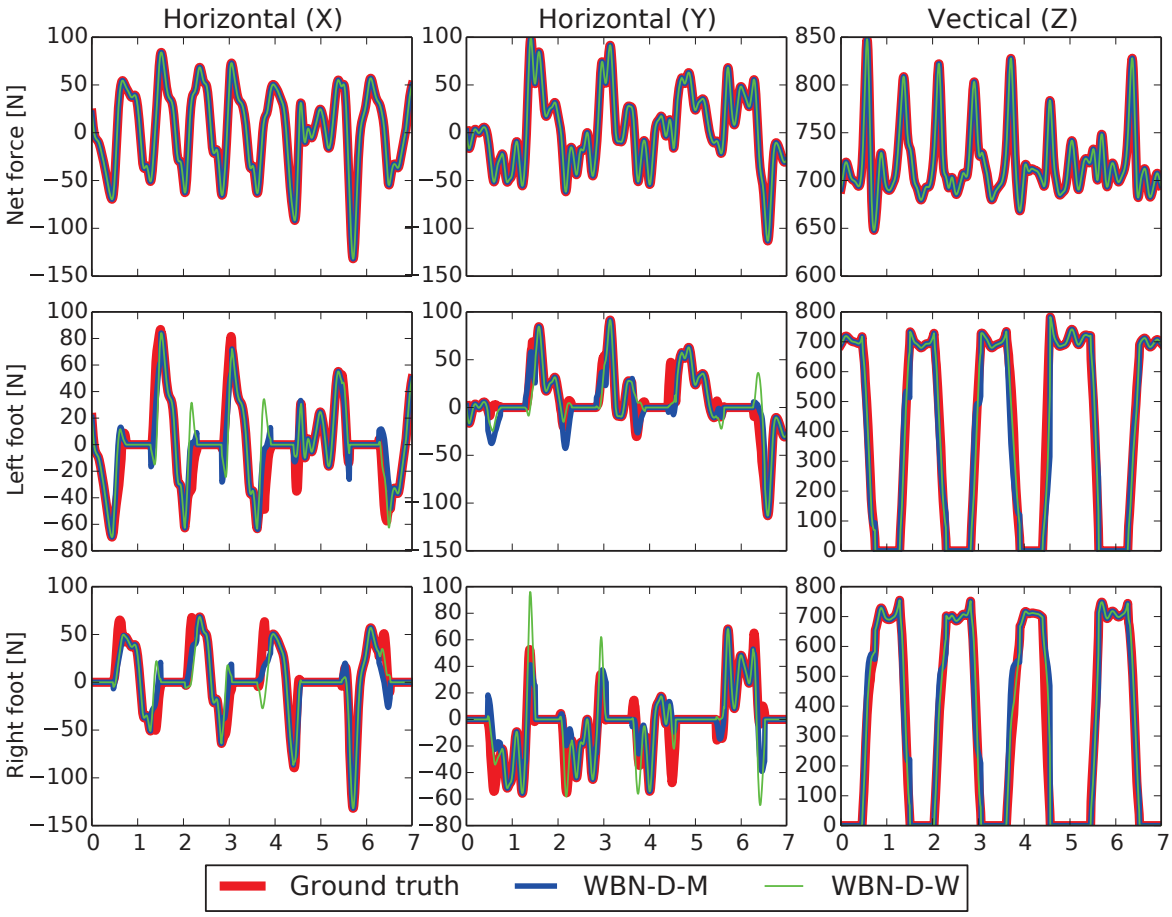


Figure 4.7: Walking example. Despite not having been extensively trained on such examples, the performance of WBN-D-M used in conjunction with physics-based optimization is comparable to that of WBN-D-W.

Conclusion

In this thesis, we demonstrated that the estimation of interaction contact forces, a problem that pertains to the human sense of *touch*, could be addressed through the lens of *vision*. In doing so, we formulated the novel problem of force sensing from motion capture, in which the goal is to estimate the *real*, rather than just *physically plausible*, contact forces that humans exert when interacting with their environment, based only on the observation of their motion. This, in turn, makes it possible to circumvent the traditional approach that consists in instrumenting objects or actors with costly, cumbersome, intrusive force transducers in monitoring motion together with force data.

Towards force sensing from vision, we first challenged the estimation of normal forces during manipulation. We constructed a first ground-truth acquisition prototype measuring real normal forces applied by human subjects. Performing model-based visual tracking using a single RGB-D sensor, we estimated the object's kinematics by differentiating the captured pose through time. From the Newton-Euler equations, we computed the net force sequence explaining the observed motion and compared it to the net forces obtained by summing up the force transducer measurements. We thus confirmed that the state of the art in markerless hand-object tracking could indeed bridge the first gap between motion and (net) forces. However, we observed that the actual forces measured during real experiments were significantly greater than those obtained by straightforward physics-based optimization, a caveat neglected by force models used only for simulation. We addressed this indeterminacy with artificial neural networks capturing how humans naturally distribute internal forces in addition to the nominal forces required to achieve a given motion.

We then made our approach more robust and general, in particular by suppressing the reliance of an arbitrary cost function decomposing full force measurements into nominal and internal components. Instead, we harnessed recent advances in machine learning and trained recurrent neural networks on a novel, large-scale dataset on human manipulation kinodynamics. We developed a new ground-truth acquisition setup measuring 3D contact forces with five precision force transducers, repositionable on the object's surface, and the object's kinematics with AHRS-embedded accelerometers and gyroscopes. Together with

reconfigurable properties such as shape, friction and mass distribution, this allowed us to gather a dataset comprising 3.2 h of motion and force measurements under 193 different object-grasp configurations. We then showed that recurrent neural networks trained on this dataset could successfully be applied to kinematics estimated from vision. We thus formulated a generic force estimation framework in which recurrent neural networks trained on the manipulation kinodynamics dataset were applied to kinematics estimated from vision, and their predictions interactively corrected between consecutive time steps by a second-order cone program ensuring their physical consistency.

Finally, we proposed an extension of our force estimation framework from the case of manipulation to that of whole-body contacts with the environment. Similarly, we observed that in multi-contact, forces obtained from optimization only significantly differed from those really applied. We collected a new database on whole-body kinodynamics using an inertial motion capture suit in combination with force-torque sensors under the subject's shoes and held in-hand. In essence, the generality of our previous formulation allowed its extension to whole-body contacts simply by transposing the SOCP from the Newton-Euler equations for a 6-DoF rigid object to the equations of motion for a 72-DoF articulated system, while training the RNN on the whole-body centroidal dynamics. Performing experiments both on walking and challenging multi-contact configurations, we noted that our approach could actually outperform physical force sensing, in particular due to the large amount of noise and uncertainties high-capacity force-torque sensors are subject to, under repeated stress.

Although we obtained significant results towards the estimation of contact forces from motion capture, we believe it is still a new path of research for which we are only laying the foundations. In the case of manipulation, the quality of the force estimation is still contingent on the accuracy of the hand-object tracking. In particular, while we were consistently able to capture the object's kinematics, locating the contact points with sufficient accuracy has proven to be particularly challenging in the current PSO-based tracking framework. Instead of sampling pose hypotheses in the configuration space, an alternative could be to sample the contact space and optimize hand poses together with the resulting manipulation forces. Removing the current (artificial) constraint on grasp staticity, would thus enable the monitoring of contact forces during in-hand manipulation. Still, the acquisition of ground-truth measurements, which we consider crucial to validate any force model, is extremely difficult for truly arbitrary grasps and especially contact orientations. In other words, by trying to get rid of force transducers (the goal of this thesis), we fully experience how cumbersome they are! Still, our approach was purposely made generic to accomodate further advances in both visual tracking and force sensing technologies. In effect, we already show its applicability to alternative object tracking datasets. In the case of whole-body interaction,

our approach would also benefit from the collection of a broader dataset involving even more complex contact configurations and motions. Alternatively, we would also like to combine our whole-body force model with kinematics estimated from markerless visual tracking.

We also identified other mid-term possible developments. First, we would like to investigate the incorporation of physics-based constraints directly in the training of the neural networks, allowing the reconstruction of physically correct and human-like force distributions in a unified computational framework. Conversely, the force distribution problem could be considered again from the perspective of inverse optimal control. Using our large-scale datasets on both manipulation and whole-body kinodynamics, we could look for invariants in the way humans distribute forces on their environment while also regulating internal forces and joint torques. Alternatively, our force estimation framework could also be used for robot learning from demonstration, by providing a natural way to teach tasks using only vision instead of heavy physical haptic interfaces. We would also like to investigate the use of FSV not alone, but in complement of low-cost force transducers. Such a combination could be mutually beneficial. On one hand, cheap sensors providing for instance only low-accuracy, normal force measurements, could be corrected to be compatible with the observed motion and also augmented with predicted tangential components. On the other hand, uncertain force measurements could still serve as a reference for FSV. This would thus alleviate the current biggest limitation of the current system, namely that it only predicts the force distributions that are *most likely* given past observations and can be 'deceived' by applying forces that significantly deviate from the norm (e.g., by gripping stronger than natural).

In the long term, we would like to consider the estimation of contact forces for manipulation and whole-body interactions together, rather than separately. Instead of considering only one rigid body or one articulated system, subject to external forces, we could instead consider a set of multiple agents, either active (i.e., that) or passive (i.e., objects) with respect to the application of contact forces, each with their own dynamic model. This would enable the consideration of interaction scenarios involving synergies between actors, e.g., for bimanual prehension or multiple persons carrying large objects in cooperation. Alternatively, we would also like to investigate the use of additional types of sensors, as long as these remain minimally intrusive. In particular, physiological sensors such as electroencephalography (EEG) headsets or surface electromyography (EMG) electrodes may provide valuable information in discerning otherwise dynamically equivalent force distributions. Finally, in our work, we made the assumption that the physical properties of the manipulated object were readily available (e.g., through inertial parameter identification or CAD). Conversely, we could explore the reverse problem, i.e., retrieve the physical properties of manipulated objects using only visual observations on how people naturally interact with them.

Publications

Accepted / Published:

- Tu-Hoa Pham, Abderrahmane Kheddar, Ammar Qammaz, and Antonis A. Argyros. Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2015
  - Paper: https://hal.archives-ouvertes.fr/hal-01356136/document
  - Video: https://www.youtube.com/watch?v=mtWwkOJkeXM
- Tu-Hoa Pham, Abderrahmane Kheddar, Ammar Qammaz, and Antonis A Argyros. Capturing and reproducing hand-object interactions through vision-based force sensing. In *IEEE ICCV Workshop on Object Understanding for Interaction*, December 2015
  - Paper: https://hal.archives-ouvertes.fr/hal-01372238/document
- Tu-Hoa Pham, Adrien Bufort, Stéphane Caron, and Abderrahmane Kheddar. Whole-body contact force sensing from motion capture. In *IEEE/SICE International Symposium on System Integration*, 2016
  - Paper: https://hal.archives-ouvertes.fr/hal-01372531/document

Submitted:

- Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros, and Abderrahmane Kheddar. Hand-object contact force estimation from markerless visual tracking. Submitted to IEEE Trans. on Pattern Analysis and Machine Intelligence, August 2016
  - Preprint: https://hal.archives-ouvertes.fr/hal-01356138/document
  - Dataset: https://github.com/jrl-umi3218/ManipulationKinodynamics
  - Video: https://www.youtube.com/watch?v=NhNV3tCcbd0

References

[ACB+14] Don Joven Agravante, Andrea Cherubini, Antoine Bussy, Pierre Gergondet, and Abderrahmane Kheddar. Collaborative human-humanoid carrying using vision and haptic sensing. In *IEEE International Conference on Robotics and Automation*, pages 607–612, 2014.

[ADV13] M.S. Andersen, J. Dahl, and L. Vandenberghe. Cvxopt: A python package for convex optimization. abel.ee.ucla.edu/cvxopt, 2013.

[AHS03] Irene Albrecht, Jörg Haber, and Hans-Peter Seidel. Construction and animation of anatomically based human hand models. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 98–109. Eurographics Association, 2003.

[AIL85] M. A. Arbib, A. R. Iberall, and D. Lyons. Coordinated control programs for movements of the hand. *Hand function and the neocortex*, pages 111–129, 1985.

[AJK13] S. Andrews, M. Jarvis, and P.G. Kry. Data-driven fingertip appearance for interactive hand simulation. *ACM SIGGRAPH Conference on Motion in Games*, 2013.

[AK13] S. Andrews and P. G. Kry. Goal directed multi-finger manipulation: Control policies and analysis. *Computers & Graphics*, 37(7):830 – 839, 2013.

[AL04] Antonis A. Argyros and Manolis I. A. Lourakis. Real-time tracking of multiple skin-colored objects with a possibly moving camera. In *European Conference on Computer Vision*, pages 368–379. Springer, 2004.

[Ale84] R. McN. Alexander. The gaits of bipedal and quadrupedal animals. *The International Journal of Robotics Research*, 3(2):49–59, 1984.

[AP01] Frank C. Anderson and Marcus G. Pandy. Dynamic optimization of human walking. *Journal of Biomechanical Engineering*, 123(5):381–390, 2001.

[Bar89] David Baraff. Analytical methods for dynamic simulation of non-penetrating rigid bodies. In *ACM SIGGRAPH*, 1989.

[Bar91] David Baraff. Coping with friction for non-penetrating rigid body simulation. In *ACM SIGGRAPH*, 1991.

[BCV13] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[BL14] Yunfei Bai and C. Karen Liu. Dexterous manipulation using both palm and fingers. In *IEEE International Conference on Robotics and Automation*, pages 1560–1565, 2014.

[BM92] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[Bre01] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[BSF09] M. A. Brubaker, L. Sigal, and D. J. Fleet. Estimating Contact Dynamics. In *IEEE International Conference on Computer Vision*, 2009.

[BSPK02] Kiran S. Bhat, Steven M. Seitz, Jovan Popović, and Pradeep K. Khosla. Computing the physical parameters of rigid-body motion from video. In *European Conference on Computer Vision*, pages 551–565, 2002.

[BTG+12] L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys. Motion capture of hands in action using discriminative salient points. In *European Conference on Computer Vision*, 2012.

[BV04] Stephen P. Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[BV15] Vincent Bonnet and Gentiane Venture. Fast determination of the planar body segment inertial parameters using affordable sensors. *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, 23(4):628–635, 2015.

[BW07] Stephen P. Boyd and Ben Wegbreit. Fast computation of optimal contact forces. *IEEE Trans. on Robotics*, 23(6):1117–1132, 2007.

[C+13] Erwin Coumans et al. Bullet physics library. 2013.

[CBK03] German K. M. Cheung, Simon Baker, and Takeo Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–77, 2003.

[CCM+75] R. F. Chandler, Charles E. Clauser, John T. McConville, H. M. Reynolds, and John W. Young. Investigation of inertial properties of the human body. Technical report, DTIC Document, 1975.

[CH06] François Chaumette and Seth Hutchinson. Visual servo control. i. basic approaches. *IEEE Robotics & Automation Magazine*, 13(4):82–90, 2006.

[CHP08] M. R. Cutkosky, R. D. Howe, and W. R. Provancher. *Springer Handbook of Robotics*, chapter Force and Tactile Sensors, pages 455–476. Springer, Berlin, Heidelberg, 2008.

[CKF11] Ronan Collobert, Koray Kavukcuoglu, and Clément Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011.

[CL96] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques (SIGGRAPH 96)*, pages 303–312. ACM, 1996.

[CMC07] Andrew I. Comport, Éric Marchand, and François Chaumette. Kinematic sets for real-time robust articulated object tracking. *Image and Vision Computing*, 25(3):374–391, 2007.

[CPN16] Stéphane Caron, Quang-Cuong Pham, and Yoshihiko Nakamura. Zmp support areas for multi-contact mobility under frictional constraints. *IEEE Trans. on Robotics*, September 2016.

[CS96] Geneviève Cadoret and Allan M. Smith. Friction, not texture, dictates grip forces used during object manipulation. *Journal of Neurophysiology*, 75(5):1963–1969, 1996.

[Cut89] Mark R. Cutkosky. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Trans. on Robotics and Automation*, 5(3):269–279, 1989.

[CVMG+14] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[ÇWS+15] Berk Çalli, Aaron Walsman, Arjun Singh, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M. Dollar. Benchmarking in manipulation research: The YCB object and model set and benchmarking protocols. *ICRA Tutorial*, 2015.

[DA87] D. T. Davy and M. L. Audu. A dynamic optimization technique for predicting muscle forces in the swing phase of gait. *Journal of Biomechanics*, 20(2):187–201, 1987.

[DCV07] R. Dumas, L. Cheze, and J.-P. Verriest. Adjustments to mcconville et al. and young et al. body segment inertial parameters. *Journal of Biomechanics*, 40(3):543–553, 2007.

[Dem55] Wilfred Taylor Dempster. *Space Requirements of the Seated Operator: Geometric, Kinematic, and Mechanical Aspects of the Body with Special Reference to Limbs*. NTIS, 1955.

[DL96] Paolo De Leva. Adjustments to zatsiorsky-seluyanov's segment inertia parameters. *Journal of Biomechanics*, 29(9):1223–1230, 1996.

[dLGFP11] Martin de La Gorce, David J Fleet, and Nikos Paragios. Model-based 3d hand pose estimation from monocular video. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33(9):1793–1805, 2011.

[DMVS10] Ravinder S. Dahiya, Giorgio Metta, Maurizio Valle, and Giulio Sandini. Tactile sensing - from humans to humanoids. *IEEE Trans. on Robotics*, 26(1):1–20, 2010.

[ECR92] Bernard Espiau, François Chaumette, and Patrick Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313–326, 1992.

[EKO15] Johannes Englsberger, Pawel Kozlowski, and Christian Ott. Biologically inspired dead-beat controller for bipedal running in 3d. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 989–996, 2015.

[EL01] Stephen A. Ehmann and Ming C. Lin. Accurate and fast proximity queries between polyhedra using convex surface decomposition. In *Computer Graphics Forum*, volume 20, pages 500–511. Wiley Online Library, 2001.

[Elm90] Jeffrey L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.

[EMHvdB07] Ahmet Erdemir, Scott McLean, Walter Herzog, and Antonie J. van den Bogert. Model-based estimation of muscle forces exerted during movements. *Clinical Biomechanics*, 22(2):131–154, 2007.

[ESK01] Russell C. Eberhart, Yuhui Shi, and James Kennedy. *Swarm Intelligence (The Morgan Kaufmann Series in Evolutionary Computation)*. Morgan Kaufmann, 1 edition, 2001.

[FbSRK09] Thomas Feix, Heinz bodo Schmiedmayer, Javier Romero, and Danica Kragić. A comprehensive grasp taxonomy. In *RSS Conference: Workshop on Understanding the Human Hand for Advancing Robotic Manipulation*, 2009.

[FGS12] Stefan Fritsch, Frauke Guenther, and Marc Suling. Package 'neuralnet', version 1.32. http://CRAN.R-project.org/package=neuralnet, 2012.

[FJ02] J. Randall Flanagan and Roland S. Johansson. Hand movements. *Encyclopedia of the Human Brain*, 2:399–414, 2002.

[FJ09] Michel Fliess and Cédric Join. Model-free control and intelligent PID controllers: towards a possible trivialization of nonlinear control? In *15th IFAC Symposium on System Identification*, July 2009.

[FRS+16] T. Feix, J. Romero, H. B. Schmiedmayer, A. M. Dollar, and D. Kragic. The GRASP taxonomy of human grasp types. *IEEE Trans. on Human-Machine Systems*, 46(1), 2016.

[FSR03] Michel Fliess and Hebertt Sira-Ramírez. An algebraic framework for linear identification. *ESAIM: Control, Optimisation and Calculus of Variations*, 9:151–168, 2003.

[GD96] Dariu M. Gavrila and Larry S. Davis. 3-d model-based tracking of humans in action: a multi-view approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 73–80, 1996.

[GHM13] Thomas Grieve, John M. Hollerbach, and Stephen A. Mascaro. Force prediction by fingernail imaging using active appearance models. In *World Haptics*, pages 181–186, 2013.

[GKD09] A. Gupta, A. Kembhavi, and L. S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(10), 2009.

[GLZ05] Fan Gao, Mark L. Latash, and Vladimir M. Zatsiorsky. Internal forces during object manipulation. *Experimental Brain Research*, 165(1):69–83, 2005.

[GPKT12] Varun Ganapathi, Christian Plagemann, Daphne Koller, and Sebastian Thrun. Real-time human pose tracking from range data. In *European Conference on Computer Vision*, pages 738–751. Springer, 2012.

[GYB04] S. Burak Gokturk, Hakan Yalcin, and Cyrus Bamji. A time-of-flight depth sensor-system description, issues and solutions. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, pages 35–35, 2004.

[GZL10] Stacey L. Gorniak, Vladimir M. Zatsiorsky, and Mark L. Latash. Manipulation of a fragile object. *Experimental Brain Research*, 202(2), 2010.

[HBL11] Sehoon Ha, Yunfei Bai, and C. Karen Liu. Human motion reconstruction from force sensors. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 129–138, 2011.

[HJ64] Ernest P. Hanavan Jr. A mathematical model of the human body. Technical report, DTIC Document, 1964.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[IBA86] Thea Iberall, Geoffrey Bingham, and M. A. Arbib. Opposition space as a structuring concept for the analysis of skilled hand movements. *Experimental Brain Research series*, 15:158–173, 1986.

[ID04] Salim Ibrir and Sette Diop. A numerical procedure for filtering and efficient high-order signal differentiation. *International Journal of Applied Mathematics and Computer Science*, 14(2):201–208, 2004.

[IWGC+16] Jan Issac, Manuel Wüthrich, Cristina Garcia Cifuentes, Jeannette Bohg, Sebastian Trimpe, and Stefan Schaal. Depth-based object tracking using a robust gaussian filter. In *IEEE International Conference on Robotics and Automation*, 2016.

[JEA+16] Jovana Jovic, Adrien Escande, Ko Ayusawa, Eiichi Yoshida, Abderrahmane Kheddar, and Gentiane Venture. Humanoid and human inertia parameter identification using hierarchical optimization. *IEEE Trans. on Robotics*, 32(3):726–735, 2016.

[JW84] R. S. Johansson and G. Westling. Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects. *Experimental Brain Research*, 56(3):550–564, 1984.

[KA13] Nikolaos Kyriazis and Antonis A. Argyros. Physically plausible 3d scene tracking: The single actor hypothesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[KA14] Nikolaos Kyriazis and Antonis A. Argyros. Scalable 3d tracking of multiple interacting objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3430–3437, 2014.

[KAJS11] Hema S. Koppula, Abhishek Anand, Thorsten Joachims, and Ashutosh Saxena. Semantic labeling of 3d point clouds for indoor scenes. In *Advances in Neural Information Processing Systems*, pages 244–252, 2011.

[KB99] Hirokazu Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *IEEE and ACM International Workshop on Augmented Reality*, pages 85–94, 1999.

[KCP15] Zhanat Kappassov, Juan-Antonio Corrales, and Véronique Perdereau. Tactile sensing in dexterous robot hands — review. *Robotics and Autonomous Systems*, 74:195–220, 2015.

[KE95] J. Kennedy and R. Eberhart. Particle swarm optimization. In *IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948, 1995.

[KKKA11] Cem Keskin, Furkan Kıraç, Yunus Emre Kara, and Lale Akarun. Real time hand pose estimation using depth sensors. In *IEEE ICCV Workshop on Consumer Depth Cameras for Computer Vision*, 2011.

[KKKA12] Cem Keskin, Furkan Kıraç, Yunus Emre Kara, and Lale Akarun. Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In *European Conference on Computer Vision*, pages 852–863. Springer, 2012.

[KMB+14] Alexander Krull, Frank Michel, Eric Brachmann, Stefan Gumhold, Stephan Ihrke, and Carsten Rother. 6-dof model based tracking via object coordinate regression. In *Asian Conference on Computer Vision*. Springer, 2014.

[KP06] Paul G. Kry and Dinesh K. Pai. Interaction capture and synthesis. *ACM Trans. on Graphics*, 25(3):872–880, 2006.

[KR86] Jeffrey Kerr and Bernard Roth. Analysis of multifingered hands. *International Journal of Robotics Research*, 4(4):3–17, 1986.

[KS16] Hema S. Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 38(1):14–29, 2016.

[LBBH98] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[LBF14] Kevin Lai, Liefeng Bo, and Dieter Fox. Unsupervised feature learning for 3d scene labeling. In *IEEE International Conference on Robotics and Automation*, 2014.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[LCK+00] Anatole Lécuyer, Sabine Coquillart, Abderrahmane Kheddar, Paul Richard, and Philippe Coiffet. Pseudo-haptic feedback: Can isometric input devices simulate force feedback? In *IEEE Virtual Reality Conference*, pages 83–90, New Brunswick, NJ, 2000.

[LFNP14] Jia Liu, Fangxiaoyu Feng, Yuzuko C Nakamura, and Nancy S Pollard. A taxonomy of everyday grasps in action. In *IEEE-RAS International Conference on Humanoid Robots*, pages 573–580, 2014.

[LHP05] C. Karen Liu, Aaron Hertzmann, and Zoran Popović. Learning physics-based motion style with nonlinear inverse optimization. *ACM Trans. on Graphics*, 24(3):1071–1081, 2005.

[Liu08] C. Karen Liu. Synthesis of interactive hand manipulation. In Markus H. Gross and Doug L. James, editors, *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 163–171, 2008.

[Liu09] C. Karen Liu. Dextrous manipulation from a grasping pose. *ACM Trans. on Graphics*, 28(3):59:1–59:6, 2009.

[LK81] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, volume 81, pages 674–679, 1981.

[LLS15] Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *International Journal of Robotics Research*, 34(4-5):705–724, 2015.

[LNP+15] Hongbin Liu, Kien Cuong Nguyen, Véronique Perdereau, Joao Bimbo, Junghwan Back, Matthew Godden, Lakmal D. Seneviratne, and Kaspar Althoefer. Finger contact sensing and the application in dexterous hand manipulation. *Autonomous Robots*, 39(1):25–41, 2015.

[Low04] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[LPKQ16] Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with large-scale data collection. In *International Symposium on Experimental Robotics*, 2016.

[LVBL98] Miguel Sousa Lobo, Lieyen Vandenberghe, Stephen P. Boyd, and Hervé Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193–228, 1998.

[MA01] Stephen A. Mascaro and Harry H. Asada. Photoplethysmograph fingernail sensors for measuring finger forces without haptic obstruction. *IEEE Trans. on Robotics and Automation*, 2001.

[MA04] Stephen A. Mascaro and Harry H. Asada. Measurement of finger posture and three-axis fingertip touch force using fingernail sensors. *IEEE Trans. on Robotics and Automation*, 20(1):26–35, 2004.

[MB13] Artashes Mkhitaryan and Darius Burschka. Visual estimation of object density distribution through observation of its impulse response. In *International Conference on Computer Vision Theory and Applications*, pages 586–595, 2013.

[MBSP16] Mostafa Mohammadi, Tommaso Lisini Baldi, Stefano Scheggi, and Domenico Prattichizzo. Fingertip force estimation via inertial and magnetic sensors in deformable object manipulation. In *IEEE Haptics Symposium*, pages 284–289, 2016.

[MC05] Éric Marchand and François Chaumette. Feature tracking for visual servoing purposes. *Robotics and Autonomous Systems*, 52(1):53–70, 2005.

[MCC+80] John T. McConville, Charles E. Clauser, Thomas D. Churchill, Jaime Cuzzi, and Ints Kaleps. Anthropometric relationships of body and body segment moments of inertia. Technical report, DTIC Document, 1980.

[MG01] Thomas B. Moeslund and Erik Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, 2001.

[MGPD15] Antoine Muller, Coralie Germain, Charles Pontonnier, and Georges Dumont. A comparative study of 3 body segment inertial parameters scaling rules. *Computer Methods in Biomechanics and Biomedical Engineering*, 18(sup1):2010–2011, 2015.

[MHK06] Thomas B. Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006.

[MJF09] Mamadou Mboup, Cédric Join, and Michel Fliess. Numerical differentiation with annihilators in noisy environment. *Numerical Algorithms*, 50(4):439–467, 2009.

[MLA+15] Igor Mordatch, Kendall Lowrey, Galen Andrew, Zoran Popovic, and Emanuel V. Todorov. Interactive control of diverse complex characters with neural networks. In *Advances in Neural Information Processing Systems*, pages 3132–3140, 2015.

[Mom09] Katja Mombaur. Using optimization to create self-stable human-like running. *Robotica*, 27(03):321–330, 2009.

[MPA15] Damien Michel, Kostas Panagiotakis, and Antonis A. Argyros. Tracking the articulated motion of the human body with two rgbd cameras. *Machine Vision Applications*, 26(1):41–54, 2015.

[MPR+10] Thomas Mörwald, Johann Prankl, Andreas Richtsfeld, Michael Zillich, and Markus Vincze. Blort - the blocks world robotic vision toolbox. In *ICRA Workshop Best Practice in 3D Perception and Modeling for Mobile Manipulation*, 2010.

[MPT12] Igor Mordatch, Zoran Popović, and Emanuel Todorov. Contact-invariant optimization for hand manipulation. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '12, pages 137–144, Aire-la-Ville, Switzerland, Switzerland, 2012.

[MS85] Matthew T. Mason and J. Kenneth Salisbury. *Robot Hands and the Mechanics of Manipulation*. MIT Press, 1985.

[MSC05] Éric Marchand, Fabien Spindler, and François Chaumette. Visp for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robotics & Automation Magazine*, 12(4):40–52, 2005.

[MSZ94] Richard M. Murray, S. Shankar Sastry, and Li Zexiang. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., 1st edition, 1994.

[MTL10] Katja Mombaur, Anh Truong, and Jean-Paul Laumond. From human to humanoid locomotion — an inverse optimal control approach. *Autonomous robots*, 28(3):369–383, 2010.

[MTP12] Igor Mordatch, Emanuel Todorov, and Zoran Popović. Discovery of complex behaviors through contact-invariant optimization. *ACM Trans. on Graphics.*, 31(4):43:1–43:8, 2012.

[NIH+11] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, 2011.

[NTLZ12] Xun Niu, Alexander V. Terekhov, Mark L. Latash, and Vladimir M. Zatsiorsky. Reconstruction of the unknown optimization cost functions from experimental recordings during static multi-finger prehension. *Motor Control*, 16(2):195, 2012.

[NYFS05] Yoshihiko Nakamura, Katsu Yamane, Yusuke Fujita, and Ichiro Suzuki. Somatosensory computation for man-machine interface from motion-capture data and musculoskeletal human model. *IEEE Trans. on Robotics*, 21(1):58–66, 2005.

[OKA10] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A. Argyros. Markerless and efficient 26-DOF hand pose recovery. In *Asian Conference on Computer Vision*, Queenstown, New Zealand, 2010.

[OKA11a] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *British Machine Vision Conference*, 2011.

[OKA11b] 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 *IEEE International Conference on Computer Vision*, 2011.

[OKA12] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A. Argyros. Tracking the articulated motion of two strongly interacting hands. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2012.

[Ola15] Christopher Olah. Understanding lstm networks. http://colah.github.io/, 2015.

[PBCK16] Tu-Hoa Pham, Adrien Bufort, Stéphane Caron, and Abderrahmane Kheddar. Whole-body contact force sensing from motion capture. In *IEEE/SICE International Symposium on System Integration*, 2016.

[PC99] DJ Pearsall and PA Costigan. The effect of segment parameter error on gait analysis results. *Gait & Posture*, 9(3):173–183, 1999.

[PKAK16] Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros, and Abderrahmane Kheddar. Hand-object contact force estimation from markerless visual tracking. Submitted to IEEE Trans. on Pattern Analysis and Machine Intelligence, August 2016.

[PKQA15a] Tu-Hoa Pham, Abderrahmane Kheddar, Ammar Qammaz, and Antonis A Argyros. Capturing and reproducing hand-object interactions through vision-based force sensing. In *IEEE ICCV Workshop on Object Understanding for Interaction*, December 2015.

[PKQA15b] Tu-Hoa Pham, Abderrahmane Kheddar, Ammar Qammaz, and Antonis A. Argyros. Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2015.

[Pop07] Ronald Poppe. Vision-based human motion analysis: An overview. *Computer Vision and Image Understanding*, 108(1):4–18, 2007.

[PPB12] Urbain Prieur, Véronique Perdereau, and Alexandre Bernardino. Modeling and planning high-level in-hand manipulation actions from human knowledge and active learning from demonstration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1330–1336, 2012.

[PRR14] Karl Pauwels, Leonardo Rubio, and Eduardo Ros. Real-time model-based articulated object pose detection and tracking with variable rigidity constraints. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3994–4001, Columbus, Ohio, 2014.

[PRR15] Karl Pauwels, Leonardo Rubio, and Eduardo Ros. Real-time pose detection and tracking of hundreds of objects. *IEEE Trans. on Circuits and Systems for Video Technology*, 2015.

[PSZL12] Jaebum Park, Tarkeshwar Singh, Vladimir M. Zatsiorsky, and Mark L. Latash. Optimality versus variability: effect of fatigue in multi-finger redundant tasks. *Experimental Brain Research*, 216(4):591–607, 2012.

[PZ02] Boris I. Prilutsky and Vladimir M. Zatsiorsky. Optimization-based models of muscle coordination. *Exercise and Sport Sciences Reviews*, 30(1):32, 2002.

[QSW+14] Chen Qian, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun. Realtime and robust hand tracking from depth. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[R C14] R Core Team. R: A language and environment for statistical computing. R-project.org, 2014.

[RABF06] Guillaume Rao, David Amarantini, Eric Berton, and Daniel Favier. Influence of body segments' parameters estimation models on inverse dynamics solutions during gait. *Journal of Biomechanics*, 39(8):1531–1536, 2006.

[RHCF07] Jeffrey A. Reinbolt, Raphael T. Haftka, Terese L. Chmielewski, and Benjamin J. Fregly. Are patient-specific joint and inertial parameters necessary for accurate inverse dynamics analyses of gait? *IEEE Trans. on Biomedical Engineering*, 54(5):782–793, 2007.

[RJC13] Leonel Rozo, Pablo Jiménez, and Torras Carme. A robot learning from demonstration framework to perform force-based manipulation tasks. *Intelligent Service Robotics*, 6(1):33–51, 2013.

[RK94] James M. Rehg and Takeo Kanade. Visual tracking of high dof articulated structures: an application to human hand tracking. In *European Conference on Computer Vision*, pages 35–46. Springer Berlin Heidelberg, 1994.

[RKK09] J. Romero, H. Kjellström, and D. Kragic. Monocular real-time 3d articulated hand pose estimation. In *IEEE-RAS International Conference on Humanoid Robots*, pages 87–92, 2009.

[RKK10] Javier Romero, Hedvig Kjellström, and Danica Kragic. Hands in action: real-time 3d reconstruction of hands in interaction with objects. In *IEEE International Conference on Robotics and Automation*, pages 458–463, 2010.

[RLS09] Daniel Roetenberg, Henk Luinge, and Per Slycke. Xsens mvn: full 6dof human motion tracking using miniature inertial sensors. *Xsens Motion Technologies BV, Tech. Rep*, 2009.

[RMR11a] Samer Riachy, Mamadou Mboup, and Jean-Pierre Richard. Multivariate numerical differentiation. *Journal of Computational and Applied Mathematics*, 2011.

[RMR11b] Samer Riachy, Mamadou Mboup, and Jean-Pierre Richard. Numerical differentiation on irregular grids. In *IFAC World Congress*, Milan, Italie, 2011.

[RSR15a] Grégory Rogez, James S. Supancic, and Deva Ramanan. First-person pose recognition using egocentric workspaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4325–4333, 2015.

[RSR15b] Grégory Rogez, James S. Supancic, and Deva Ramanan. Understanding everyday hands in action from rgb-d images. In *IEEE International Conference on Computer Vision*, pages 3889–3897, 2015.

[S+05] Russell Smith et al. Open dynamics engine. 2005.

[SCCP14] Stefano Stassi, Valentina Cauda, Giancarlo Canavese, and Candido Fabrizio Pirri. Flexible tactile sensing based on piezoresistive composites: A review. *Sensors*, 14(3):5296–5332, 2014.

[SDN08] Ashutosh Saxena, Justin Driemeyer, and Andrew Y. Ng. Robotic grasping of novel objects using vision. *International Journal of Robotics Research*, 27(2):157–173, 2008.

[SE16] Russell Stewart and Stefano Ermon. Label-free supervision of neural networks with physics and domain knowledge. *arXiv preprint arXiv:1609.05566*, 2016.

[SFB+11] Jamie Shotton, Andrew Fitzgibbon, Andrew Blake, Alex Kipman, Mark Finocchio, Richard Moore, and Toby Sharp. Real-time human pose recognition in parts from a single depth image. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011.

[SHG+11] Carsten Stoll, Nils Hasler, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt. Fast articulated motion tracking using a sums of gaussians body model. In *IEEE International Conference on Computer Vision*, pages 951–958, 2011.

[SHK+14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[SHM08] Yu Sun, John M. Hollerbach, and Stephen A. Mascaro. Predicting fingertip forces by imaging coloration changes in the fingernail and surrounding skin. *IEEE Trans. on Biomedical Engineering*, 55(10):2363–2371, 2008.

[SHM09] Yu Sun, John M. Hollerbach, and Stephen A. Mascaro. Estimation of fingertip force direction with computer vision. *IEEE Trans. on Robotics*, 25(6), 2009.

[SHM+16] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[SKR+15] Toby Sharp, Cem Keskin, Duncan Robertson, Jonathan Taylor, Jamie Shotton, David Kim, Christoph Rhemann, Ido Leichter, Alon Vinnikov, Yichen Wei, et al. Accurate, robust, and flexible real-time hand tracking. In *ACM Conference on Human Factors in Computing Systems*, pages 3633–3642, 2015.

[SL01] Carsten Schedlinski and Michael Link. A survey of current inertia parameter identification methods. *Mechanical Systems and Signal Processing*, 15(1):189 – 211, 2001.

[SLZ11] Gregory P. Slota, Mark L. Latash, and Vladimir M. Zatsiorsky. Grip forces during object manipulation: experiment, mathematical model, and validation. *Experimental Brain Research*, 213(1):125–139, 2011.

[SMOT15] Srinath Sridhar, Franziska Mueller, Antti Oulasvirta, and Christian Theobalt. Fast and robust hand tracking using detection-guided optimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[SMZ+16] 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 *European Conference on Computer Vision*, 2016.

[SNF14] Tanner Schmidt, Richard Newcombe, and Dieter Fox. Dart: Dense articulated real-time tracking. *Robotics: Science and Systems*, 2014.

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

[SPSS12] Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena. Unstructured human activity detection from RGBD images. In *IEEE International Conference on Robotics and Automation*, pages 842–849, 2012.

[SS03] Daniel Scharstein and Richard Szeliski. High-accuracy stereo depth maps using structured light. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages I–195, 2003.

[SSK+13] Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013.

[TBC+16] Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, Arran Topalian, Erroll Wood, Sameh Khamis, Pushmeet Kohli, Toby Sharp,

Shahram Izadi, Richard Banks, Andrew Fitzgibbon, and Jamie Shotton. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. *ACM Trans. on Graphics*, 35(4):143, 2016.

[TBS+15] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient points and physics simulation. *International Journal of Computer Vision*, pages 1–22, 2015.

[TCTK14] Danhang Tang, Hyung Jin Chang, Alykhan Tejani, and Tae-Kyun Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[Tod04] Emanuel Todorov. Optimality principles in sensorimotor control. *Nature Neuroscience*, 7(9):907–915, 2004.

[TSLP14] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Trans. on Graphics*, 33, 2014.

[TST+15] Andrea Tagliasacchi, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. Robust articulated-icp for real-time hand tracking. In *Computer Graphics Forum*, volume 34, pages 101–114, 2015.

[TYK13] Danhang Tang, Tsz-Ho Yu, and Tae-Kyun Kim. Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In *IEEE International Conference on Computer Vision*, pages 3224–3231, 2013.

[UBO+13] Sebastian Urban, Justin Bayer, Christian Osendorfer, Göran Westling, Benoni B. Edin, and Patrick van der Smagt. Computing grip force and torque from finger nail images using gaussian processes. In *IEEE-RSJ International Conference on Intelligent Robots and Systems*, 2013.

[V+13] Joris Vaillant et al. Rbdyn. https://github.com/jorisv/RBDyn, 2013.

[VTBE15] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2015.

[WDGJ14] Mariska Wesseling, Friedl De Groote, and Ilse Jonkers. The effect of perturbing body segment parameters on calculated joint moments and muscle forces during gait. *Journal of Biomechanics*, 47(2):596–601, 2014.

[WJ84] G. Westling and R. S. Johansson. Factors influencing the force control during precision grip. *Experimental Brain Research*, 53(2):277–284, 1984.

[WMZ+13] Y. Wang, J. Min, J. Zhang, Y. Liu, F. Xu, Q. Dai, and J. Chai. Video-based hand manipulation capture through composite motion control. *ACM Trans. on Graphics*, 32(4):43, 2013.

[WP09] Robert Y. Wang and Jovan Popović. Real-time hand-tracking with a color glove. In *ACM Trans. on Graphics*, volume 28, page 63, 2009.

[WTZL15] Yen-Hsun Wu, Thomas S. Truglio, Vladimir M. Zatsiorsky, and Mark L. Latash. Learning to combine high variability with high precision: Lack of transfer to a different task. *Journal of Motor Behavior*, 47(2):153–165, 2015.

[YCS+83] Joseph W. Young, Richard F. Chandler, Clyde C. Snow, Kathleen M. Robinette, Gregory F. Zehner, and Maureen S. Lofberg. Anthropometric and mass distribution characteristics of the adult female. Technical report, DTIC Document, 1983.

[YFW16] Zheng Yan, Jianchao Fan, and Jun Wang. A collective neurodynamic approach to constrained global optimization. *IEEE Trans. on Neural Networks and Learning Systems*, 2016.

[YL12] Yuting Ye and C. Karen Liu. Synthesis of detailed hand manipulations using contact sampling. *ACM Trans. on Graphics*, 31(4):1–10, 2012.

[YN91] T. Yoshikawa and K. Nagai. Manipulating and grasping forces in manipulation by multifingered robot hands. *IEEE Trans. on Robotics and Automation*, 7(1):67–77, 1991.

[YYFA16] Chengxi Ye, Yezhou Yang, Cornelia Fermuller, and Yiannis Aloimonos. What can i do around here? deep functional scene understanding for cognitive robots. *arXiv preprint arXiv:1602.00032*, 2016.

[Zat02] Vladimir M. Zatsiorsky. *Kinetics of human motion*. Human Kinetics, 2002.

[ZJZ+16] Yixin Zhu, Chenfanfu Jiang, Yibiao Zhao, Demetri Terzopoulos, and Song-Chun Zhu. Inferring forces and learning human utilities from videos. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[ZS83] V. M. Zatsiorsky and V. Seluyanov. The mass and inertia characteristics of the main segments of the human body. *Biomechanics VIII-B*, 56(2):1152–1159, 1983.

[ZSC90a] V. M. Zatsiorsky, V. Seluyanov, and L. Chugunova. In vivo body segment inertial parameters determination using a gamma-scanner method. *Biomechanics of Human Movement: Applications in Rehabilitation, Sports and Ergonomics*, pages 186–202, 1990.

[ZSC90b] V. M. Zatsiorsky, V. N. Seluyanov, and L. G. Chugunova. Methods of determining mass-inertial characteristics of human body segments. *Contemporary Problems of Biomechanics*, 272:291, 1990.

[ZSZ+14] Peizhao Zhang, Kristin Siu, Jianjie Zhang, C. Karen Liu, and Jinxiang Chai. Leveraging depth cameras and wearable pressure sensors for full-body kinematics and dynamics capture. *ACM Trans. on Graphics*, 33, 2014.

[ZY13] Yu Zheng and Keisaku Yamane. Evaluation of grasp force efficiency considering hand configuration and using novel generalized penetration distance algorithm. In *IEEE International Conference on Robotics and Automation*, pages 1580–1587, 2013.

[ZZCZ15] Yixin Zhu, Yibiao Zhao, and Song Chun Zhu. Understanding tools: Task-oriented object modeling, learning and recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.