Hand-Object Contact Force Estimation From Markerless Visual Tracking

Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros, Abderrahmane Kheddar

▶ **To cite this version:**

Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros, Abderrahmane Kheddar. Hand-Object Contact Force Estimation From Markerless Visual Tracking. 2016. hal-01356138v1

HAL Id: hal-01356138

https://hal.science/hal-01356138v1

Preprint submitted on 25 Aug 2016 (v1), last revised 25 Sep 2017 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Hand-Object Contact Force Estimation From Markerless Visual Tracking

Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros and Abderrahmane Kheddar, *Senior Member, IEEE*

Abstract—We consider the problem of computing realistic contact forces during manipulation, backed with ground-truth measurements, using vision alone. Interaction forces are traditionally measured by mounting force transducers onto the manipulated objects or the hands. Those are costly, cumbersome, and alter the objects' physical properties and their perception by the human sense of touch. Our work establishes that interaction forces can be estimated in a cost-effective, reliable, non-intrusive way using vision. This is a complex and challenging problem. Indeed, in multi-contact, a given trajectory can generally be caused by an infinity of possible distributions. To alleviate the limitations of traditional models based on inverse optimization, we collect and release the first large-scale dataset on manipulation kinodynamics as 3.2 hours of synchronized force and motion measurements under 193 object-grasp configurations. We learn a mapping between high-level kinematic features based on the equations of motion and the underlying manipulation forces using recurrent neural networks (RNN). The RNN predictions are consistently refined using physics-based optimization through second-order cone programming (SOCP). We show that our method can successfully capture interaction forces compatible with both the observations and the way humans naturally manipulate objects, on an acquisition system no more complex than a single RGB-D camera.

Index Terms—Force sensing from vision, hand-object tracking, manipulation, pattern analysis, sensors, tracking.

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 [1], [2], [3]. 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 (Section 2). In particular, kinesiology research has resulted in successful attempts at modeling grip forces by inverse optimization, e.g., during static prehension [4] or two-finger circular motion [5]. 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).
- 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 4).
- We thoroughly validate our approach on ground-truth measurements (Section 5) and show that it can seamlessly be extended to visual tracking (Section 6).

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 7). Finally, we discuss thoroughly the current limitations, possible extensions and applications of our work (Section 8). A preliminary version of this research, focused on estimating normal forces from vision,

- *T.-H. Pham and A. Kheddar are with the CNRS-AIST Joint Robotics Laboratory, UMI3218/RL, Tsukuba, Japan, and the Interactive Digital Humans group of CNRS-University of Montpellier, LIRMM, UMR5506, Montpellier, France.*
- *N. Kyriazis, and A. A. Argyros are with the Institute of Computer Science, FORTH, Heraklion, Greece. A. A. Argyros is also with the Computer Science Department, University of Crete, Heraklion, Greece.*

appeared in [6]. 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].

2 Related Work

2.1 Monitoring Hand-Object Interactions

Current force transduction and sensing technologies are costly and may require frequent calibration. Mounting them onto objects biases physical properties such as shape, mass distribution and friction, while mounting them onto hands obstructs the human haptic sense, limiting the natural range of motion. In contrast, there is evidence that fingertip forces can be correlated to changes in the coloration of fingernails and surrounding skin [7], [8], [9]. These setups already suggest that computer vision can measure touch forces.

Used in conjunction with force sensing technologies, motion tracking can also provide information on body dynamics to explain how humans interact with their environment. A setup combining marker-based motion capture and force sensors was used in [10] to estimate hand joint compliance and synthesize interaction animations. While the use of motion capture markers does not directly interfere with the hand-object interactions, it is arguably invasive and difficult to deploy in the context of daily life activities. In this direction, the topic of markerless hand tracking was introduced in [11] and has lately received renewed attention in [12], [13], [14], [15], [16], [17], [18], [19], [20], [21]. During manipulation, hand-object interactions cause mutual occlusions that generative approaches can employ to enforce priors in the optimization process [22], [23], [24], [25]. In particular, force models can be used to select hand poses that are compatible with the observations through physical simulation [26], [27], [28]. In contrast with our approach, these models only need to capture physically plausible distributions rather than the actual forces being applied, which may substantially differ.

2.2 Biomechanical Models for Human Prehension

Prehension is an active research topic in the kinesiology field, an interest that stems from the remarkable dexterity and complexity of the human hand. As a result, inverse optimization approaches for manipulation have mostly resulted in models that, albeit sophisticated, rely on rather strong simplifying assumptions. The most common restriction is on the motion's dimensionality, e.g. static prehension [4]. Other approaches allow limited motion, e.g. circular [5], using a simplified grasp model in which individual fingers and hand surfaces are grouped into functional units named virtual fingers [29]. For instance, a hand holding a cup is seen as the thumb on one side and a virtual finger on the opposite side realizing the total wrench due to the four antagonist fingers. Under this formalism, the five-finger grasp is effectively seen as *two-finger*. In this simplified model, given the object's kinematics, the knowledge of one force fully determines the other. In reality, the force distribution problem is generally indeterminate as full-hand forces can compensate each other and cause the same motion.

The virtual finger model was also applied on nominal-internal forces during vertical and horizontal translational motions [30]. Internal forces represent the set of forces that humans apply in excess to the nominal forces that are physically required to create a given motion [31], [32]. For instance, when holding a cup statically, nominal forces directly compensate gravity, while internal forces secure the object through a firm grip but cancel each other out [33], [34]. Past studies showed that humans control internal forces to prevent slip, muscle fatigue or damaging fragile objects [35], [36], [37]. Overall, in reviewing several optimization-based models attempting to predict muscle activation patterns, [38] showed that the high redundancy of the human body makes it particularly difficult to identify clear optimization criteria in the way the central nervous system regulates human efforts at the musculoskeletal level.

2.3 Force Sensing From Vision

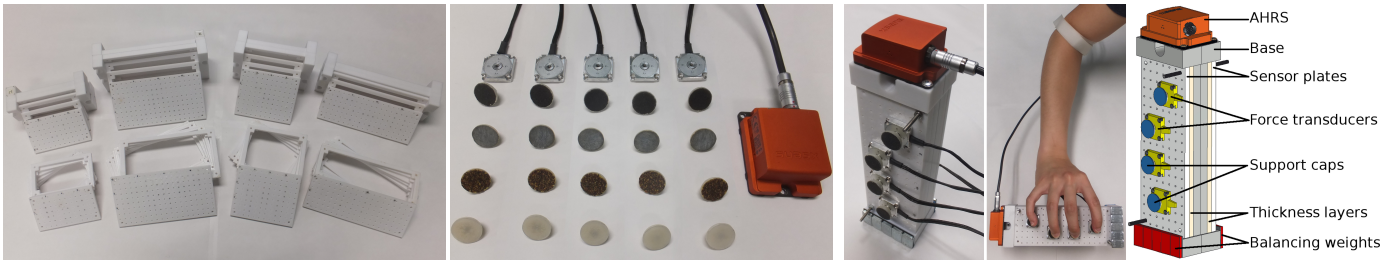
The force sensing from vision (FSV) framework presented in this paper is a continuation of our earlier work in [6], that was limited to 1D normal force measurements, four-finger grasps and relatively limited experimental conditions. In contrast, this paper is based on an extensive dataset of 3D force measurements on five-finger, diverse manipulation experiments. In addition, our past work used shallow multilayer perceptrons (MLP) to learn internal forces. Such an approach is difficult to generalize as the decomposition into nominal and internal components is not intrinsic, but rather depends on the objective function chosen to minimize nominal forces. While the extended approach we present here still builds upon the formulation of the force distribution problem as a second-order cone program (SOCP) [39], [40], we also capitalize on the recent success of deep learning applications to manipulation and monitoring of human activities [2], [41], [42] to construct a network that directly learns full 3D manipulation forces, avoiding the need for arbitrary constraints and hand-engineering [43].

Our work was also inspired by [44], which estimated ground reaction forces from motion capture using a damped spring model. Recently, forces were computed between the hand and deformable objects [45] and conversely by considering the human body elastic [46]. [19] showed that manipulation forces play a crucial role towards understanding hand-object interactions from vision and noted the challenge of obtaining the ground-truth contact points and forces humans use instinctively, which we address in our work.

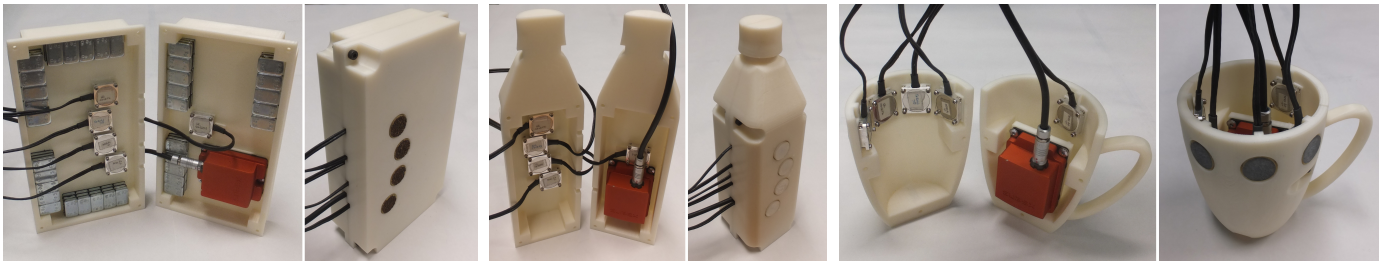
3 Manipulation Kinodynamics Dataset

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 [47], [48], whole-body and hand tracking [17], [49], and robotic grasping [50], [51]. In contrast, datasets viewing human manipulation not only from the angle of vision but also of

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.

Fig. 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).

touch have been more scarce so far. A notable example is the interaction capture technique of [10] 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.

3.1 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 [52], [53]. 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. 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 $8\,\mathrm{mm}$ steps on the surface. Thickness layers can be inserted in between to increase the grasp width by $5\,\mathrm{mm}$ increments, bringing the total grasp width range between $46\,\mathrm{mm}$ and $86\,\mathrm{mm}$. The force transducers are fitted with support caps of different surface textures: PET, sand paper of grit 40 (coarse), 150 (medium) and 320 (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 $56\times96\ \mathrm{mm}^2$. 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 [54], 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 7.

3.2 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 3.1. 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 $400\,\mathrm{g}$ 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 $60\,\mathrm{s}$. Every $10\,\mathrm{s}$, 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 $50\,\mathrm{g}$ balancing weight is attached to a randomly picked side, excluding sensor plates. Throughout the eight trials, we measure the effect of mass variations between $0\,\mathrm{g}$ and $350\,\mathrm{g}$ or $400\,\mathrm{g}$ and $750\,\mathrm{g}$ 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 $400\,\mathrm{Hz}$; 3D force measurements expressed in the reference frame of the object $\mathcal{R}_{\text{obj.}}$, subsampled from $500\,\mathrm{Hz}$ to $400\,\mathrm{Hz}$ to match the AHRS; the physical properties of the object: mass m, inertia matrix $\mathbf{J}$ about the center of mass $\mathbf{C}$; and the grasp parameters: for each finger $k \in \mathcal{F}$, the friction coefficient μ_k at contact point $\mathbf{P}_k^c$, and $\mathcal{R}_k = (\mathbf{n}_k, \mathbf{t}_k^x, \mathbf{t}_k^y)$ 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:

$$\|g_k \mathbf{t}_k^x + h_k \mathbf{t}_k^y\|_2 \leq \mu_k f_k, \tag{1}$$

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

$$\mathbf{F}_k = f_k \mathbf{n}_k + g_k \mathbf{t}_k^x + h_k \mathbf{t}_k^y. \tag{2}$$

3.3 Equations of Motion and Synchronization

Let $\boldsymbol{\mathcal{F}}_\mathrm{c}$ and $\boldsymbol{\tau}_c$ be the net force and torque due to individual contact forces, and $\boldsymbol{\mathcal{F}}_\mathrm{d}$ 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:

$$\begin{cases} \boldsymbol{\mathcal{F}}_\mathrm{c} = m\mathbf{a} - \boldsymbol{\mathcal{F}}_\mathrm{d} \\ \boldsymbol{\tau}_\mathrm{c} = \mathbf{J}_\mathbf{q} \cdot \boldsymbol{\alpha} + \boldsymbol{\omega} \times (\mathbf{J}_\mathbf{q} \cdot \boldsymbol{\omega}) - \boldsymbol{\tau}_\mathrm{d}, \end{cases} \tag{3}$$

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 $\pm 0.3\,\mathrm{m \cdot s^{-2}}$. For an object of mass $500\,\mathrm{g}$, 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 $0.33\,\mathrm{N}$. For each experiment, we compute the average net force $\Delta\boldsymbol{\mathcal{F}}_\mathrm{c}$ and torque $\Delta\boldsymbol{\tau}_\mathrm{c}$ discrepancies between AHRS and force transducers. We align their values by computing the minimal offsets $(\Delta\mathbf{F}_k)_{k\in\mathcal{F}}$ that result in $\Delta\boldsymbol{\mathcal{F}}_\mathrm{c}$ and $\Delta\boldsymbol{\tau}_\mathrm{c}$:

$$\min \{\mathcal{C}_{\boldsymbol{\mathcal{F}}_\mathrm{c}} + \mathcal{C}_{\boldsymbol{\tau}_\mathrm{c}} + \mathcal{C}_\text{var}\}, \tag{4}$$

with force-torque discrepancy and variation cost functions:

$$\begin{cases} \mathcal{C}_{\boldsymbol{\mathcal{F}}_\mathrm{c}}\left((\Delta\mathbf{F}_k)_k\right) = \left\| \Delta\boldsymbol{\mathcal{F}}_\mathrm{c} - \sum\limits_{k\in\mathcal{F}} [\Delta\mathbf{F}_k] \right\|_2^2 \\ \mathcal{C}_{\boldsymbol{\tau}_\mathrm{c}}\left((\Delta\mathbf{F}_k)_k\right) = \left\| \Delta\boldsymbol{\tau}_\mathrm{c} - \sum\limits_{k\in\mathcal{F}} \left[\overrightarrow{\mathbf{CP}_k} \times \Delta\mathbf{F}_k\right] \right\|_2^2 \\ \mathcal{C}_\text{var}\left((\Delta\mathbf{F}_k)_k\right) = \sum\limits_{k\in\mathcal{F}} \|\Delta\mathbf{F}_k\|_2^2 \end{cases} \tag{5}$$

In practice, it is preferrable to normalize $\mathcal{C}_{\boldsymbol{\mathcal{F}}_\mathrm{c}}$ and $\mathcal{C}_{\boldsymbol{\tau}_\mathrm{c}}$, e.g., with the initial discrepancies $\Delta\boldsymbol{\mathcal{F}}_\mathrm{c}$ 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.

4 Force Model

Based on the Newton-Euler equations, the net contact force $\boldsymbol{\mathcal{F}}_\mathrm{c}$ and torque $\boldsymbol{\tau}_\mathrm{c}$ are completely determined by the object's motion and physical properties. However, given $\boldsymbol{\mathcal{F}}_\mathrm{c}$ 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.

4.1 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) of the form:

$$\begin{aligned} \min \quad & \mathcal{C}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{P}\mathbf{x} + \mathbf{r}^T\mathbf{x} \\ \text{s.t.} \quad & \begin{cases} \|\mathbf{A}_j\mathbf{x} + \mathbf{b}_j\|_2 \leq \mathbf{c}_j^T\mathbf{x} + \mathbf{d}_j, \quad j = 1, \ldots, m \\ \mathbf{E}\mathbf{x} \leq \mathbf{f} \\ \mathbf{G}\mathbf{x} = \mathbf{h}. \end{cases} \end{aligned} \tag{6}$$

We express conditions of physical plausibility using the local decompositions of Eq. (2) as 15 optimization parameters:

$$\mathbf{x} = (f_1, g_1, h_1, \ldots, f_5, g_5, h_5)^T \tag{7}$$

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:

$$f_k \geq 0, \quad k = 1, \ldots, 5. \tag{8}$$

This can be rewritten in Eq. (6) with linear inequality matrices $\mathbf{E}$ and $\mathbf{f}$ of respective sizes $5{\times}15$ and $5{\times}1$, with:

$$\begin{aligned}\mathbf{E}(i,j) &= \begin{cases} -1 & \text{if} \quad j = 3(i-1)+1 \\ 0 & \text{else} \end{cases} \\ \mathbf{f}(i,1) &= 0. \end{aligned} \tag{9}$$

Friction. The Coulomb model of Eq. (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 $2{\times}15$, $2{\times}1$, $15{\times}1$ and $1{\times}1$, such that:

$$\mathbf{A}_k\mathbf{x} + \mathbf{b}_k = \begin{pmatrix} g_k \\ h_k \end{pmatrix} \quad \text{and} \quad \mathbf{c}_k^T\mathbf{x} + \mathbf{d}_k = (\mu_k f_k)\,. \tag{10}$$

Their elements are defined as follows:

$$\begin{aligned}\mathbf{A}_k(i,j) &= \begin{cases} 1 & \text{if} \quad j = 3(k-1)+1+i \\ 0 & \text{otherwise} \end{cases} \\ \mathbf{b}_k(i,1) &= 0 \\ \mathbf{c}_k(i,1) &= \begin{cases} \mu_k & \text{if} \quad i = 3(k-1)+1 \\ 0 & \text{otherwise} \end{cases} \\ \mathbf{d}_k(1,1) &= 0. \end{aligned} \tag{11}$$

Equations of motion. Recall from Eq. (3) that the net contact force $\boldsymbol{\mathcal{F}}_c$ and torque $\boldsymbol{\tau}_c$ can be determined from kinematic quantities only. The individual finger forces are such that:

$$\begin{cases} \boldsymbol{\mathcal{F}}_c = \sum\limits_{k\in\mathcal{F}} \mathbf{F}_k \\ \boldsymbol{\tau}_c = \sum\limits_{k\in\mathcal{F}} \left[\overrightarrow{\mathbf{CP}_k} \times \mathbf{F}_k\right]. \end{cases} \tag{12}$$

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

$$\begin{aligned} &\forall i = 1,\dots,3; \quad \forall j = 1,\dots,15; \quad \forall k = 1,\dots,5; \\ &\mathbf{G}(i,j) = \begin{cases} \mathbf{n}_k \cdot \mathbf{v}_i & \text{if} \quad j = 3(k-1)+1 \\ \mathbf{t}_k^x \cdot \mathbf{v}_i & \text{if} \quad j = 3(k-1)+2 \\ \mathbf{t}_k^y \cdot \mathbf{v}_i & \text{if} \quad j = 3(k-1)+3 \\ 0 & \text{otherwise} \end{cases} \\ &\mathbf{h}(i,1) = \boldsymbol{\mathcal{F}}_c \cdot \mathbf{v}_i \\ &\mathbf{G}(i+3,j) = \begin{cases} \left[\overrightarrow{\mathbf{CP}_k} \times \mathbf{n}_k\right] \cdot \mathbf{v}_i \text{ if } j{=}3(k{-}1){+}1 \\ \left[\overrightarrow{\mathbf{CP}_k} \times \mathbf{t}_k^x\right] \cdot \mathbf{v}_i \text{ if } j{=}3(k{-}1){+}2 \\ \left[\overrightarrow{\mathbf{CP}_k} \times \mathbf{t}_k^y\right] \cdot \mathbf{v}_i \text{ if } j{=}3(k{-}1){+}3 \\ 0 \quad \text{otherwise} \end{cases} \\ &\mathbf{h}(i+3,1) = \boldsymbol{\tau}_c \cdot \mathbf{v}_i \end{aligned} \tag{13}$$

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 [6]:

$$\mathcal{C}_{L^2}(\mathbf{x}) = \sum_{k\in\mathcal{F}} \left[{f_k}^2 + {g_k}^2 + {h_k}^2\right] = \sum_{k\in\mathcal{F}} \|\mathbf{F}_k\|_2^2\,. \tag{14}$$

Yet, the resulting forces can significantly differ from those humans really apply (see Fig. 2). Instead, we consider a cost minimizing the discrepancy with given target forces $\widetilde{\mathbf{F}}_k$:

$$\mathcal{C}_{\widetilde{\mathbf{F}}_k}(\mathbf{x}) = \sum_{k\in\mathcal{F}} \left\|\mathbf{F}_k - \widetilde{\mathbf{F}}_k\right\|_2^2 \tag{15}$$

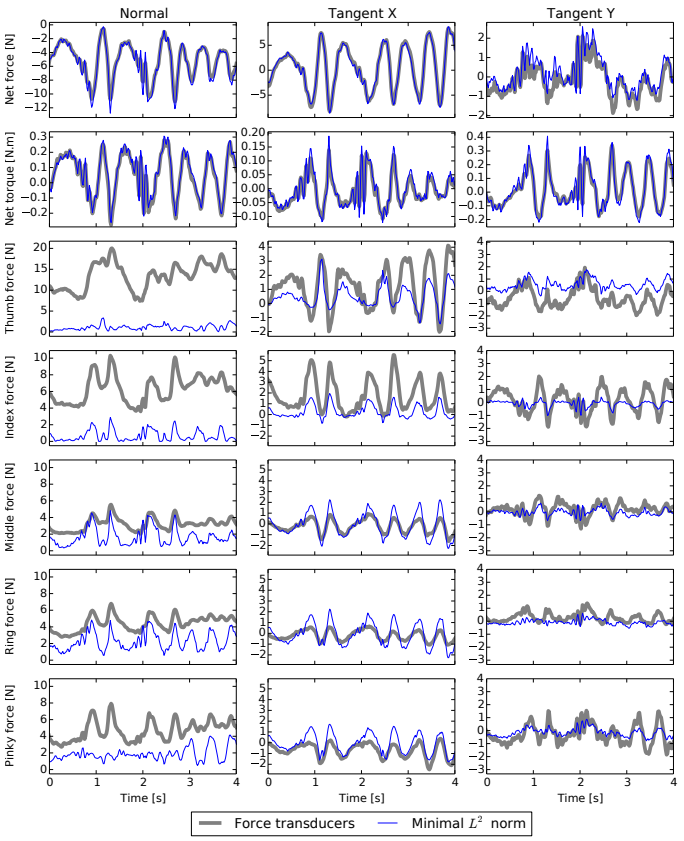


Fig. 2. Force distributions computed only by physics-based optimization are guaranteed to result in the observed motion (net force and torque) but can significantly differ from the real distributions at the finger level.

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

4.2 Learning Features

The criteria that is optimized by the central nervous system in hand-object manipulation is still unknown (see Section 2.2). A major obstacle to its identification is a dependency on musculoskeletal parameters that can be difficult to measure precisely [55]. 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, we group the available parameters into three categories:

- Object and grasp parameters: location of the center of mass $\mathbf{C}$ in $\mathcal{R}_{\text{obj.}}$, mass m, inertia matrix $\mathbf{J}$, contact point locations $\mathbf{P}_k$ and friction coefficients μ_k.
- Kinematic parameters: appearing in Eq. (3) are the object's orientation $\mathbf{q}$ in $\mathcal{R}_{\text{global}}$, rotational velocity $\boldsymbol{\omega}$, rotational acceleration $\boldsymbol{\alpha}$ and translational acceleration $\mathbf{a}$. $\mathbf{q}, \boldsymbol{\omega}, \mathbf{a}$ are directly measured by the AHRS. $\boldsymbol{\alpha}$ is obtained by simple numerical differentiation of $\boldsymbol{\omega}$. Alternatively, the relevant kinematic parameters can be obtained from visual tracking, through double differentiation of the object's pose and orientation.
- Force transducer measurements $\widetilde{\mathbf{F}}_k$.

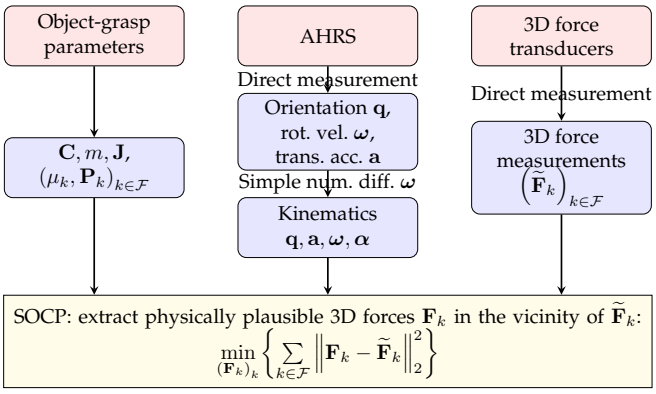


Fig. 3. For each experiment, we extract force distributions compatible with the observed motion in the vicinity of the transducer measurements.

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.

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 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. (12), the positions of the center of mass $\mathbf{C}$ and contact points $\mathbf{P}_k$ are meaningful not on their own but in relation to each other as $\overrightarrow{\mathbf{CP}}_k$. Similarly, from Eq. (3), we summarize the contributions of m, $\mathbf{a}$, $\mathbf{J}$, $\mathbf{q}$, $\boldsymbol{\omega}$, $\boldsymbol{\alpha}$ into the target net contact force $\boldsymbol{\mathcal{F}}_c$ and torque $\boldsymbol{\tau}_c$.

Recall that $\boldsymbol{\mathcal{F}}_c$ and $\boldsymbol{\tau}_c$ are expressed in $\mathcal{R}_{\text{global}}$. 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 $\boldsymbol{\mathcal{F}}_c$, $\boldsymbol{\tau}_c$ and $\overrightarrow{\mathbf{CP}}_k$ on $\mathcal{R}_{\text{obj.}}$. Thus, the input features stemming from the Newton-Euler equations are:

$$\forall (k, \mathbf{v}) \in \mathcal{F} \times \mathcal{R}_{\text{obj.}}, \quad \begin{cases} p^{\boldsymbol{\mathcal{F}}_c}_{\mathbf{v}} &= \boldsymbol{\mathcal{F}}_c \cdot \mathbf{v} \\ p^{\boldsymbol{\tau}_c}_{\mathbf{v}} &= \boldsymbol{\tau}_c \cdot \mathbf{v} \\ p^{\mathbf{P}_k}_{\mathbf{v}} &= \overrightarrow{\mathbf{CP}}_k \cdot \mathbf{v} \end{cases}. \tag{16}$$

In addition, we consider the average friction coefficient:

$$p^{\mu} = \langle \mu_k \rangle_{k\in\mathcal{F}} \tag{17}$$

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

$$\mathbf{K} = \left(p^{\boldsymbol{\mathcal{F}}_c}_{\mathbf{v}}, p^{\boldsymbol{\tau}_c}_{\mathbf{v}}, p^{\mathbf{P}_k}_{\mathbf{v}}, p^{\mu} \right)_{(k,\mathbf{v})\in\mathcal{F}\times\mathcal{R}_{\text{obj.}}} \tag{18}$$

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

$$\mathbf{D} = (\mathbf{F}_k \cdot \mathbf{v})_{(k,\mathbf{v})\in\mathcal{F}\times\mathcal{R}_{\text{obj.}}} \tag{19}$$

Note that attaching the frame to a chosen finger also helps preserve invariances througout objects and experiments. Using the thumb contact space $\mathcal{R}_{\text{th.}} = (\mathbf{t}_0^x, \mathbf{t}_0^y, \mathbf{n}_0)$ 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.

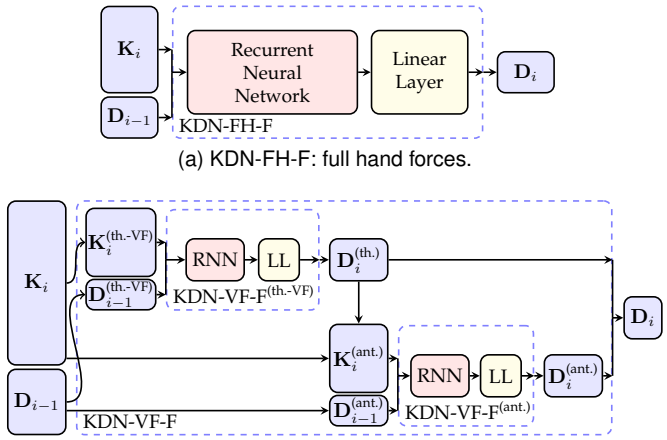


Fig. 4. Two RNN architectures learning the manipulation forces at each fingertip based on the current kinematics and past forces.

4.3 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:

$$\mathbf{D} = F(\mathbf{K}) \tag{20}$$

In [6], 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. (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:

$$\mathbf{D}_i = F\Big(\mathbf{K}_i, \mathbf{D}_{i-1}, (\mathbf{K}_j, \mathbf{D}_{j-1})_{j=1,i-1}\Big) \tag{21}$$

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) [56], with long short term memory (LSTM) neurons [57] 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:

$$\mathbf{D}_i = \text{KDN-FH-F}(\mathbf{K}_i, \mathbf{D}_{i-1}). \tag{22}$$

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:

$$\mathbf{D}_i^{(\text{th.})} = \text{KDN-VF-F}^{(\text{th.-VF})}\Big(\mathbf{K}_i^{(\text{th.-VF})}, \mathbf{D}_{i-1}^{(\text{th.})}\Big). \tag{23}$$

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

$$\mathbf{K}_i^{(\text{th.-VF})} = \left(p_{\mathbf{v}}^{\boldsymbol{\mathcal{F}}_c}, p_{\mathbf{v}}^{\boldsymbol{\tau}_c}, p_{\mathbf{v}}^{\mathbf{P}_{\text{th.}}}, p^{\mu_{\text{th.}}}, p_{\mathbf{v}}^{\mathbf{P}_{\text{ant.}}}, p^{\mu_{\text{ant.}}} \right)_{\mathbf{v} \in \mathcal{R}_{\text{th.}}}$$
$$\text{with} \quad \begin{cases} p_{\mathbf{v}}^{\mathbf{P}_{\text{ant.}}} = \left\langle p_{\mathbf{v}}^{\mathbf{P}_k} \right\rangle_{k \in \mathcal{F}_{\text{ant.}}} \\ p^{\mu_{\text{ant.}}} = \left\langle \mu_k \right\rangle_{k \in \mathcal{F}_{\text{ant.}}} \end{cases} \tag{24}$$

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

$$\forall \mathbf{v} \in \mathcal{R}_{\text{th.}}, \quad \begin{cases} p_{\mathbf{v}}^{\boldsymbol{\mathcal{F}}_{\text{ant.}}} = (\boldsymbol{\mathcal{F}}_c - \mathbf{F}_{\text{th.}}) \cdot \mathbf{v} \\ p_{\mathbf{v}}^{\boldsymbol{\tau}_{\text{ant.}}} = \left(\boldsymbol{\tau}_c - \left(\overrightarrow{\mathbf{CP}_{\text{th.}}} \times \mathbf{F}_{\text{th.}} \right) \right) \cdot \mathbf{v} \end{cases} \tag{25}$$

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

$$\mathbf{D}_i^{(\text{ant.})} = \text{KDN-VF-F}^{(\text{ant.})} \left(\mathbf{K}_i^{(\text{ant.})}, \mathbf{D}_{i-1}^{(\text{ant.})} \right)$$
$$\text{with} \quad \mathbf{K}_i^{(\text{ant.})} = \left(p_{\mathbf{v}}^{\boldsymbol{\mathcal{F}}_{\text{ant.}}}, p_{\mathbf{v}}^{\boldsymbol{\tau}_{\text{ant.}}}, p_{\mathbf{v}}^{\mathbf{P}_k}, p^{\mu_{\text{ant.}}} \right)_{(k,\mathbf{v}) \in \mathcal{F}_{\text{ant.}} \times \mathcal{R}_{\text{th.}}} \tag{26}$$

We depict KDN-FH-F and KDN-VF-F in Fig. 4.

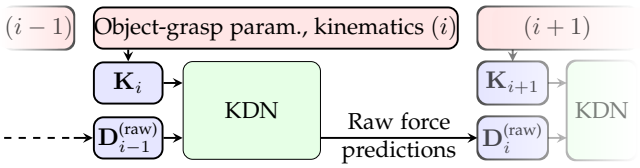
In order to further address the fact that the same motion can be due to different yet equally valid force distributions, we introduce alternative versions of KDN-FH-F and KDN-VF-F that associate current kinematics $\mathbf{K}_i$ and past forces $\mathbf{D}_{i-1}$ to force variations $\Delta\mathbf{D}_i$. 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.

5 Experiments

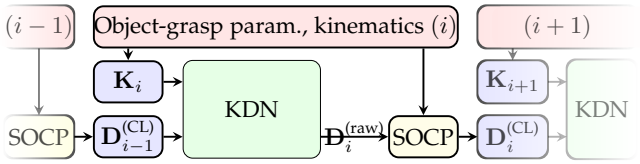
We train the four architectures KDN-FH-F, KDN-FH-Δ, KDN-VF-F, KDN-VF-Δ on the manipulation kinodynamics dataset of Section 3. Note that its sampling rate (400 Hz) far exceeds the frame rate of off-the-shelf RGB-D sensors such as Microsoft Kinect (30 fps) and Asus Xtion (60 fps). In order to be compatible with vision-based kinematics (Section 6), we down-sample the dataset to 60 Hz and split it for training (60 %), validation (20 %) and 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 [58] using stochastic gradient descent with a mean square error criterion and dropout [59] to avoid overfitting.

5.1 Force Reconstruction Model

From Eq. (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 5.2. Similarly, the predicted sequence also depends on the choice of the initial force distribution $\mathbf{D}_0$, which we address in Section 5.3. In this section, we discuss the reconstruction of physically plausible manipulation forces from KDN predictions and present our results on full-length experiments



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



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

Fig. 5. Open-loop and closed-loop force generation processes.

TABLE 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**)

with ground-truth initialization. Manipulation forces are obtained by projecting the components of $\mathbf{D}_i$ onto the local reference frame following Eq. (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. 5a. Using the SOCP described in Fig. 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 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. 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 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. 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

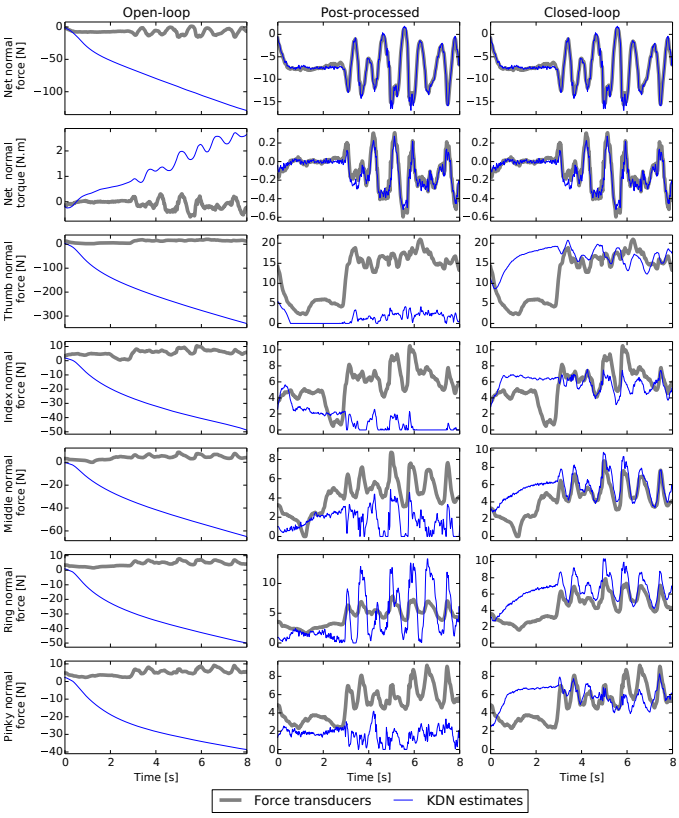


Fig. 6. Open-loop, post-processed and closed-loop force predictions for KDN-VF-Δ (normal components). In this example, the open-loop estimation drifts away from physically plausible solutions (negative normal forces). Compatibility with the observed motion is enforced through offline post-processing or closed-loop control at each time step.

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.

5.2 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 $1, 2, 4, 8, 16, 32\,\mathrm{s}$ (resp. $60, 120, 240, 480, 960, 1920$ samples) and computing the resulting estimation errors 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 $60.1\,\mathrm{s}$, standard deviation $3.8\,\mathrm{s}$). We report the results in Table 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 $1\,\mathrm{s}$ sequence duration and $2\,\mathrm{s}$ 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 $2\,\mathrm{s}$, 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.

5.3 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 $32.0\,\mathrm{s}$. 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 4.1. 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.

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 $8.0\,\mathrm{s}$ sequences. Finally, note that for any initial force distribution, the resulting sequence is constructed to be physically plausible given the observed motion and compatible 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.

6 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.

TABLE 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
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)

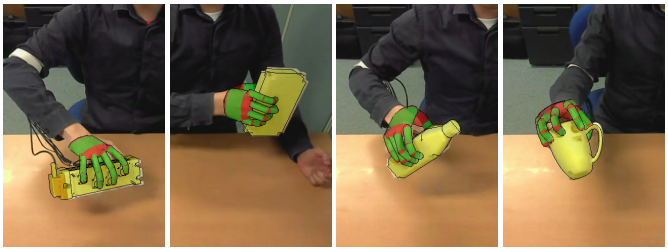
6.1 Model-Based Tracking

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 [60]. This choice was derived from our experience in [6] 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 6.2. The accuracy of the force estimates depends mostly on that of the contact points. Indicatively, simulating a Gaussian noise of standard deviation $5\,\mathrm{mm}$ (resp. $10\,\mathrm{mm}$) on the true contact points yields force reconstruction errors of zero mean (same net forces) and $0.87\,\mathrm{N}$ (resp. $1.54\,\mathrm{N}$) standard deviation. In our preliminary experiments, the average contact point estimation error was greater than $20\,\mathrm{mm}$. 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 [60]. 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.

6.2 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, in a configuration



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

Fig. 7. The hand and the object are tracked as a rigid compound.

TABLE 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)

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: 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 7.2. We depict sample tracking results in Fig. 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. 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 [61], [62] 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 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

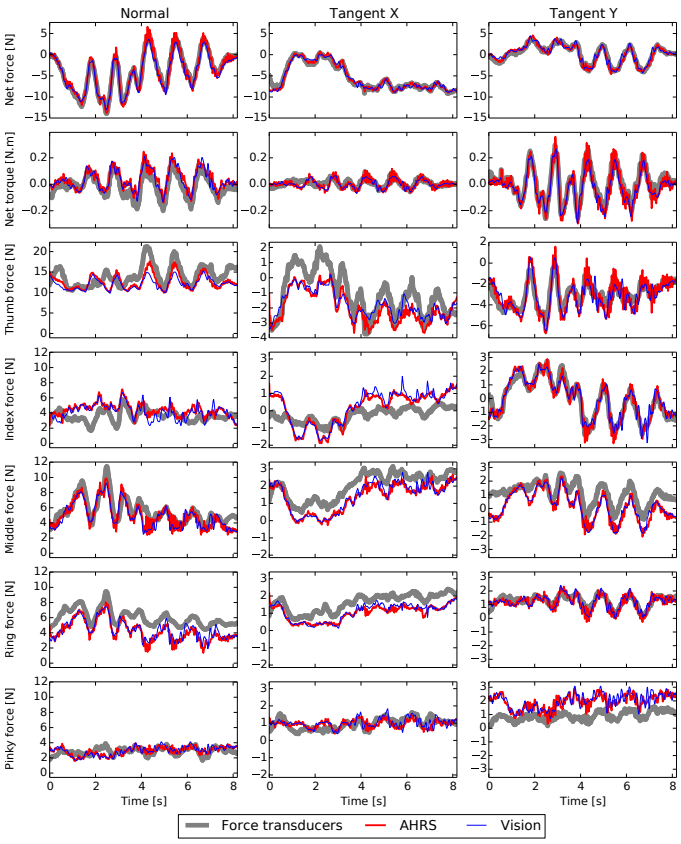


Fig. 8. Force estimates from AHRS measurements and visual tracking with closed-loop KDN-VF-F and random initialization.

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.

6.3 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 5, 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 5.3. 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 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 5, but also additional uncertainties from visual tracking. We depict an example of forces estimated from vision in Fig. 8.

TABLE 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	$\mathbf{-0.01\,(2.20)}$	$\mathbf{0.02\,(2.27)}$	$\mathbf{-0.04\,(2.30)}$
KDN-VF-Δ, PP	$5.40\,(27.61)$	$5.16\,(23.06)$	$5.94\,(24.54)$
KDN-VF-Δ, CL	$2.20\,(16.31)$	$3.87\,(19.99)$	$7.37\,(25.15)$

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.

7 Discussion

7.1 Visual Tracking Assumptions

In Section 6.1, 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 force transducers (that are not available in the required dimensions), or alternatively with dedicated force sensing surfaces [63], 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.

7.2 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. 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. 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 4.2 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.

7.3 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 $2470\,\mathrm{s}$, 60 samples per second). We report the computation time in Table 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 $5.29\,\mathrm{ms}$ 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 [58]. Third, SOCP solving is done with the CVXOPT library [64].

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 $(\mathbf{K}_i, \mathbf{D}_{i-1})$.

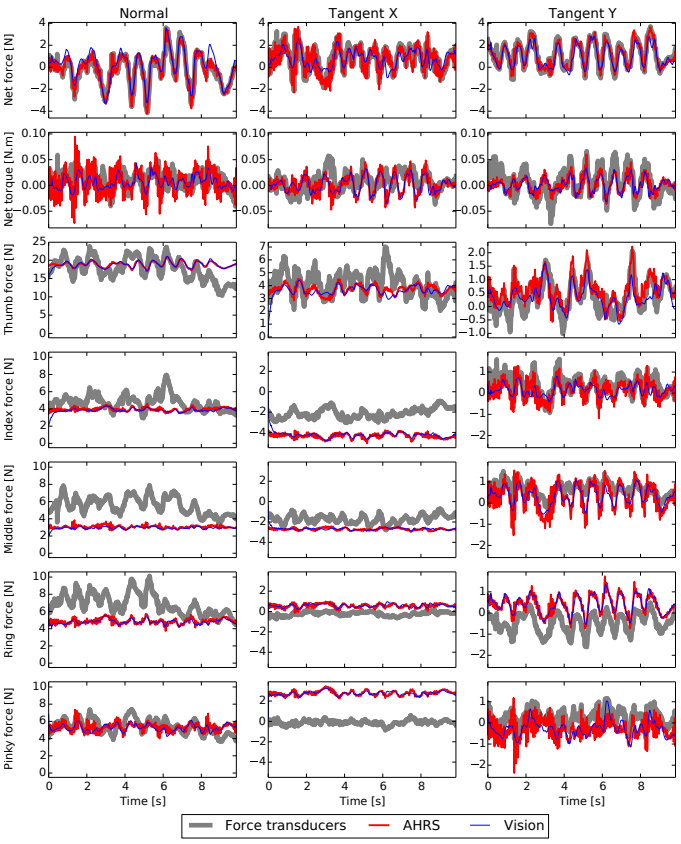


Fig. 9. Force estimates with non-prismatic grasp (mug).

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 $\mathbf{D}_i^{(\text{raw})}$.
4) We solve the SOCP and get the corrected forces $\mathbf{F}_i$.

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 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.

8 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 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. We present qualitative results on alternative object trackers [65], [66] in the supplementary material[2], with the contact points handpicked from the visual observations. 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 [67]. 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 [68]. In the long term, we plan to extend the force estimation framework to general articulated bodies for bi-manual grasping and whole-body interaction with the environment.

TABLE 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 %

Acknowledgments

This work was partially supported by the FP7 EU RoboHow.Cog project and the Japan Society for the Promotion of Science (JSPS): Kakenhi B No. 25280096.

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

REFERENCES

[1] A. Gupta, A. Kembhavi, and L. Davis, "Observing human-object interactions: Using spatial and functional compatibility for recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, 2009.

[2] Y. Zhu, Y. Zhao, and S. Chun Zhu, "Understanding tools: Task-oriented object modeling, learning and recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[3] C. Ye, Y. Yang, C. Fermuller, and Y. Aloimonos, "What can i do around here? deep functional scene understanding for cognitive robots," *arXiv preprint arXiv:1602.00032*, 2016.

[4] X. Niu, A. V. Terekhov, M. L. Latash, and V. M. Zatsiorsky, "Reconstruction of the unknown optimization cost functions from experimental recordings during static multi-finger prehension," *Motor control*, vol. 16, no. 2, 2012.

[5] G. Slota, M. Latash, and V. Zatsiorsky, "Grip forces during object manipulation: experiment, mathematical model, and validation," *Experimental Brain Research*, vol. 213, no. 1, 2011.

[6] T.-H. Pham, A. Kheddar, A. Qammaz, and A. A. Argyros, "Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

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

[8] Y. Sun, J. M. Hollerbach, and S. A. Mascaro, "Estimation of fingertip force direction with computer vision," *IEEE Trans. on Robotics*, vol. 25, no. 6, 2009.

[9] S. Urban, J. Bayer, C. Osendorfer, G. Westling, B. B. Edin, and P. 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.

[10] P. G. Kry and D. K. Pai, "Interaction capture and synthesis," *ACM Trans. on Graphics*, vol. 25, no. 3, 2006.

[11] J. M. Rehg and T. Kanade, "Visual tracking of high dof articulated structures: an application to human hand tracking," in *European Conference on Computer Vision*, 1994.

[12] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Efficient model-based 3d tracking of hand articulations using kinect," in *British Machine Vision Conference*. BMVA, 2011.

[13] M. de La Gorce, D. J. Fleet, and N. Paragios, "Model-based 3d hand pose estimation from monocular video," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, 2011.

[14] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun, "Hand pose estimation and hand shape classification using multi-layered randomized decision forests," in *European Conference on Computer Vision*, 2012.

[15] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun, "Realtime and robust hand tracking from depth," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[16] D. Tang, H. J. Chang, A. Tejani, and T.-K. Kim, "Latent regression forest: Structured estimation of 3d articulated hand posture," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

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

[18] P. Krejov, A. Gilbert, and R. Bowden, "Combining discriminative and model based approaches for hand pose estimation," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2015.

[19] G. Rogez, J. S. Supancic, and D. Ramanan, "Understanding everyday hands in action from rgb-d images," in *IEEE International Conference on Computer Vision*, 2015.

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

[21] J. Taylor, L. Bordeaux, T. Cashman, B. Corish, C. Keskin, E. Soto, D. Sweeney, J. Valentin, B. Luff, A. Topalian, E. Wood, S. Khamis, P. Kohli, T. Sharp, S. Izadi, R. Banks, A. Fitzgibbon, and J. Shotton, "Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences," *ACM Trans. on Graphics*, vol. 35, no. 4, 2016.

[22] I. Oikonomidis, N. Kyriazis, and 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.

[23] ——, "Tracking the articulated motion of two strongly interacting hands," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[24] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall, "Capturing hands in action using discriminative salient points and physics simulation," *International Journal of Computer Vision*, 2015.

[25] S. Sridhar, F. Mueller, M. Zollhöfer, D. Casas, A. Oulasvirta, and C. Theobalt, "Real-time joint tracking of a hand manipulating an object from rgb-d input," in *European Conference on Computer Vision*, 2016.

[26] N. Kyriazis and A. Argyros, "Physically plausible 3d scene tracking: The single actor hypothesis," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[27] W. Zhao, J. Zhang, J. Min, and J. Chai, "Robust realtime physics-based motion control for human grasping," *ACM Trans. on Graphics*, vol. 32, no. 6, 2013.

[28] 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*, vol. 32, no. 4, 2013.

[29] M. A. Arbib, T. Iberall, and D. Lyons, "Coordinated control programs for movements of the hand," *Hand function and the neocortex*, 1985.

[30] F. Gao, M. L. Latash, and V. M. Zatsiorsky, "Internal forces during object manipulation," *Experimental brain research*, vol. 165, no. 1, 2005.

[31] J. Kerr and B. Roth, "Analysis of multifingered hands," *International Journal of Robotics Research*, vol. 4, no. 4, 1986.

[32] T. Yoshikawa and K. Nagai, "Manipulating and grasping forces in manipulation by multifingered robot hands," *IEEE Trans. on Robotics and Automation*, vol. 7, no. 1, 1991.

[33] M. T. Mason and J. K. Salisbury, *Robot Hands and the Mechanics of Manipulation*. MIT Press, 1985.

[34] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*, 1st ed. CRC Press, Inc., 1994.

[35] J. R. Flanagan and R. S. Johansson, "Hand movements," *Encyclopedia of the human brain*, vol. 2, 2002.

[36] S. L. Gorniak, V. M. Zatsiorsky, and M. L. Latash, "Manipulation of a fragile object," *Experimental brain research*, vol. 202, no. 2, 2010.

[37] J. Park, T. Singh, V. M. Zatsiorsky, and M. L. Latash, "Optimality versus variability: effect of fatigue in multi-finger redundant tasks," *Experimental brain research*, vol. 216, no. 4, 2012.

[38] B. I. Prilutsky and V. M. Zatsiorsky, "Optimization-based models of muscle coordination," *Exercise and sport sciences reviews*, vol. 30, no. 1, 2002.

[39] M. S. Lobo, L. Vandenberghe, S. P. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear Algebra and its Applications*, vol. 284, 1998.

[40] S. P. Boyd and B. Wegbreit, "Fast computation of optimal contact forces," *IEEE Trans. on Robotics*, vol. 23, no. 6, 2007.

[41] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *International Journal of Robotics Research*, vol. 34, no. 4-5, 2015.

[42] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, 2016.

[43] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, 2013.

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

[45] M. Mohammadi, T. L. Baldi, S. Scheggi, and D. Prattichizzo, "Fingertip force estimation via inertial and magnetic sensors in deformable object manipulation," in *IEEE Haptics Symposium*, 2016.

[46] Y. Zhu, C. Jiang, Y. Zhao, D. Terzopoulos, and S.-C. Zhu, "Inferring forces and learning human utilities from videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[47] H. S. Koppula, A. Anand, T. Joachims, and A. Saxena, "Semantic labeling of 3d point clouds for indoor scenes," in *Advances in Neural Information Processing Systems*, 2011.

[48] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3d scene labeling," in *IEEE International Conference on Robotics and Automation*, 2014.

[49] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from RGBD images," in *IEEE International Conference on Robotics and Automation*, 2012.

[50] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic grasping of novel objects using vision," *International Journal of Robotics Research*, vol. 27, no. 2, 2008.

[51] B. Çalli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: The YCB object and model set and benchmarking protocols," *ICRA Tutorial*, 2015.

[52] C. Schedlinski and M. Link, "A survey of current inertia parameter identification methods," *Mechanical Systems and Signal Processing*, vol. 15, no. 1, 2001.

[53] K. S. Bhat, S. M. Seitz, J. Popović, and P. K. Khosla, "Computing the physical parameters of rigid-body motion from video," in *European Conference on Computer Vision*, 2002.

[54] 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*, vol. 46, no. 1, 2016.

[55] A. Erdemir, S. McLean, W. Herzog, and A. J. van den Bogert, "Model-based estimation of muscle forces exerted during movements," *Clinical Biomechanics*, vol. 22, no. 2, 2007.

[56] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, 1990.

[57] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, 1997.

[58] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A matlab-like environment for machine learning," in *BigLearn, NIPS Workshop*, 2011.

[59] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, 2014.

[60] N. Kyriazis and A. Argyros, "Scalable 3d tracking of multiple interacting objects," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[61] M. Fliess and H. Sira-Ramírez, "An algebraic framework for linear identification," *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 9, 2003.

[62] M. Mboup, C. Join, and M. Fliess, "Numerical differentiation with annihilators in noisy environment," *Numerical Algorithms*, vol. 50, no. 4, 2009.

[63] S. Stassi, V. Cauda, G. Canavese, and C. F. Pirri, "Flexible tactile sensing based on piezoresistive composites: A review," *Sensors*, vol. 14, no. 3, 2014.

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

[65] A. Krull, F. Michel, E. Brachmann, S. Gumhold, S. Ihrke, and C. Rother, "6-dof model based tracking via object coordinate regression," in *Asian Conference on Computer Vision*. Springer, 2014.

[66] J. Issac, M. Wüthrich, C. Garcia Cifuentes, J. Bohg, S. Trimpe, and S. Schaal, "Depth-based object tracking using a robust gaussian filter," in *IEEE International Conference on Robotics and Automation*, 2016.

[67] T.-H. Pham, A. Kheddar, A. Qammaz, and A. A. Argyros, "Capturing and reproducing hand-object interactions through vision-based force sensing," in *IEEE ICCV Workshop on Object Understanding for Interaction*, 2015.

[68] Y. Zheng and K. Yamane, "Evaluation of grasp force efficiency considering hand configuration and using novel generalized penetration distance algorithm," in *IEEE International Conference on Robotics and Automation*, 2013.

Tu-Hoa Pham is a Ph.D. candidate at the CNRS-AIST Joint Robotics Laboratory (JRL), UMI3218/RL, Tsukuba, Japan, and with the Interactive Digital Humans (IDH) team at CNRS-UM LIRMM, Montpellier, France, under the supervision of Prof. Abderrahmane Kheddar. He received the Dipl.-Ing. SUPAERO degree from Institut Supérieur de l'Aéronautique et de l'Espace and the M.Sc. degree in Applied Mathematics from Université Paul Sabatier, both in Toulouse, France, 2013. His research interests include humanoid robotics, machine learning and computer vision for monitoring of human activities and robot learning from demonstration.

Nikolaos Kyriazis is a Post-Doctoral Associate with the Computer Vision and Robotics Laboratory (CVRL) of the Foundation for Research and Technology – Heraklion (FORTH). He has studied at and received from the University of Crete (UoC) the BSc, MSc, and PhD diplomas in 2005, 2009 and 2014, respectively. He has also been affiliated with CVRL, FORTH, since 2005. His main research interests evolve around computational methods for observing and understanding the interaction of human and robotic systems with objects of their environment. Other research interests regard Physics Simulation, Computer Graphics, Software Engineering and Performance Computing.

Antonis A. Argyros is a Professor of Computer Science at the Computer Science Department, University of Crete and a researcher at the Institute of Computer Science, Foundation for Research and Technology-Hellas (FORTH) in Heraklion, Crete, Greece. His research interests fall in the areas of computer vision with emphasis on 2D/3D tracking, human gesture and posture recognition, 3D reconstruction and omnidirectional vision. He is also interested in applications of computer vision in the fields of robotics and smart environments. In these areas he has published more than 140 papers in scientific journals and refereed conference proceedings. Antonis Argyros has served as a general co-chair of ECCV 2010, as an Area Chair for ECCV 2016, as a co-organizer of HANDS 2015 and as an Associate Editor for IEEE ICRA 2016 and IEEE IROS 2016. He also serves as an Area Editor for the Computer Vision and Image Understanding Journal (CVIU) and as a member of the Editorial Boards of the IET Image Processing and IEEE Robotics & Automation Letters journals. He is also a member of the Strategy Task Group of the European Consortium for Informatics and Mathematics (ERCIM).

Abderrahmane Kheddar received the BS in Computer Science degree from the Institut National d'Informatique (ESI), Algiers, the MSc and PhD degree in robotics, both from the University of Pierre et Marie Curie, Paris. He is presently Directeur de Recherche at CNRS and the Director of the CNRS-AIST Joint Robotic Laboratory (JRL), UMI3218/RL, Tsukuba, Japan. He is also leading the Interactive Digital Humans (IDH) team at CNRS-University of Montpellier LIRMM, France. His research interests include haptics, humanoids and thought-based control using brain machine interfaces. He is a founding member of the IEEE/RAS chapter on haptics, the co-chair and founding member of the IEEE/RAS Technical committee on model-based optimization, He is a member of the steering committee of the IEEE Brain Initiative, Editor of the IEEE Transactions on Robotics and within the editorial board of some other robotics journals; he is a founding member of the IEEE Transactions on Haptics and served in its editorial board during three years (2007-2010). He is an IEEE senior member, and titular full member of the National Academy of Technology of France.