Physically Plausible 3D Scene Tracking: The Single Actor Hypothesis

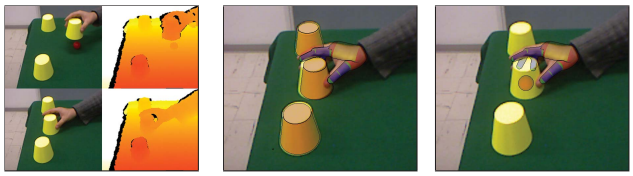
Nikolaos Kyriazis, Antonis Argyros
Institute of Computer Science, FORTH and Computer Science Department, University of Crete
{kyriazis, argyros}@ics.forth.gr

Abstract

In several hand-object(s) interaction scenarios, the change in the objects' state is a direct consequence of the hand's motion. This has a straightforward representation in Newtonian dynamics. We present the first approach that exploits this observation to perform model-based 3D tracking of a table-top scene comprising passive objects and an active hand. Our forward modelling of 3D hand-object(s) interaction regards both the appearance and the physical state of the scene and is parameterized over the hand motion (26 DoFs) between two successive instants in time. We demonstrate that our approach manages to track the 3D pose of all objects and the 3D pose and articulation of the hand by only searching for the parameters of the hand motion. In the proposed framework, covert scene state is inferred by connecting it to the overt state, through the incorporation of physics. Thus, our tracking approach treats a variety of challenging observability issues in a principled manner, without the need to resort to heuristics.

1. Introduction

One of the major goals of computer vision is to extract meaningful interpretations of the world based on the analysis of visual data. This work focuses on a scenario, where the hand of a human actor interacts with a number of objects placed on a table. A fundamental step towards the interpretation of this interaction is the monitoring of the state of the scene, i.e., the 3D position and orientation of the objects and the 3D position, orientation and full articulation of the actor's hand. A key observation is that in such a scenario, the human hand is the *single actor* and *scene state changes can be attributed to the actions of the human hand and their induced consequences*. Given that the physical world and its visual observations are determined by the laws of physics, we focus on how computer vision may benefit from *explicitly* accounting for these laws. Thus, we model the dynamics of a hand interacting with a physical world. Moreover, we make a distinction between active and passive entities, to come up with effective, physically plausible interpretations of scenes, exhibiting complex hand-object(s) interaction.



(a) (b) (c)

Figure 1: The exploitation of the *single actor hypothesis* through *physics modelling*, allows physically plausible, heuristic-free 3D tracking of hand-object interactions. (a) RGBD observation of a hand interacting with objects. (b), (c) By searching for hand motion only, we are able to track the 3D state of the entire scene. The state can be *overt* (partially visible hand and objects (b)) or even *covert* (totally occluded objects like the ball-inside-the-cup (c)).

We monitor the 3D state of the scene by means of tracking (Fig. 1), which is defined as an optimization problem. The objective function is a quantification of the discrepancy between a given hypothesis over the scene state and observations, and is parameterized over a hand motion between two successive instants in time. A hypothesized hand motion is simulated in a physics-based simulation environment that reflects the latest state of the scene, as it has been tracked up to that point in time. The simulated hand-object(s) interaction yields an expectation over the appearance of such a hypothesis, that regards both the hand and the object(s). A comparison of this expectation to actual observations quantifies the compatibility of the hand motion hypothesis to the data. A highly preferable hypothesis is one that explains (a) where the hand is in the new tracking frame and (b) the consequences of its interaction with the scene, as those are reflected in the observations. The expectation and comparison mechanisms are implemented as a forward model that accounts for the dynamics and the ap-

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.9

pearance of a scene. This model is turned into an inference mechanism over the physical state of the scene by means of black-box optimization.

We show that under the *single actor hypothesis*, our approach is able to track complex scenes. At the same time, by modelling dynamics, we bring scene understanding to a level where various problems (constrained observability, scene cardinality, *etc*.) are resolved effortlessly, uniformly and without the need to resort to heuristics.

2. Relevant work

Our work aims at deriving physically plausible interpretations of the interaction of a human with the environment. In this context, we are interested in approaches that study the interaction of humans with their environment and/or incorporate physics/dynamics to improve vision processes.

Several researchers have exploited dynamics, by introducing interesting abstractions of physical phenomena, in order to tackle scene understanding problems. Brand *et al*. [3] exploited the physical notion of causality to perform qualitative reasoning in computer vision problems. Mann *et al*. [14] methodically generated 2D hypotheses for simple scenes viewed by a single camera. The roles of the scene's constituents were then ranked based on *physical plausibility*. Delamarre [7] assigned physical behavior to a contour model that drove a reconstructive optimization process. Papadourakis and Argyros [19] identified the physical notion of object permanence as the ambiguity resolver for the case of multiple objects tracking. Gupta *et al*. [9] used the notion of physical stability to hypothesize physically plausible 3D scene interpretations.

Several other approaches consider dynamics explicitly, but restrict understanding to either the actor or a single object, only. Human body dynamics has been exploited towards the formation of strong yet compact priors [25]. Urtasun et al. [23] modeled the dynamics of the golf swing motion to track golf swings in 3D from a single camera. Popović and Witkin [21] rectified 3D motion capture data to make it compliant to physical constraints. Vondrak *et al*. [24] fused motion planning and contact dynamics to track humans from multiple cameras and a ground assumption. Brubaker *et al*. [4] employed realistic metaphors of the lower body dynamics to estimate and predict walking. Going further, they incorporated a friction model for a ground that affords human motion upon it [5]. Bhat *et al*. [2] performed 3D tracking of an object by searching over parameterized experiments that optimally projected back to an image sequence. Duff and Wyatt [8] used physical simulation and search heuristics to track a fast moving ball, despite occlusions and for the 2D case. In previous work [13], we performed 3D motion estimation for a bouncing ball, from a single camera and despite severe occlusions by exploiting dynamics modelling. Ye and Liu [26] synthesized physically plausible hand movements, from pour or absent hand observations, that explained the manipulation of objects with known trajectories from a hand whose rough location was also known.

There are also approaches that go beyond abstractions of dynamics while considering ensembles of entities rather than entities in isolation. Metaxas and Terzopoulos [15] defined a continuous Kalman filter that was able to track a deformable object. Although interesting, the proposed approach is of limited extensibility and is susceptible to overfitting. Salzmann and Urtasun [22] approached the problem of 3D tracking by attributing motion of parts to net forces that act upon them at each tracking frame. Because of the lack of explicit structure, this method is also susceptible to overfitting. Scaling to different types of interaction or introducing more structure is not straightforward.

Kjellstrom et al. [12] improved the estimation of the 3D pose of the human body while in interaction with easy to track objects, by constraining the hands. In previous work we tracked the constellation of a hand and an object from multiple cameras [17], and the full articulation of two interacting hands from a RGBD sensor [18], all in 3D, by employing synthetic 3D models. Ballan et al. [1] captured delicate interaction between two hands and an object from multiple high resolution cameras using 3D models of high fidelity. While inspired by the fundamentals of interactions, none of these approaches considered dynamics directly. Moreover, in all cases, the consideration of more objects would require the increase of the problem dimensionality.

2.1. Our approach

In this work, by considering (a) the dynamics of a scene as the core representation in a dynamics simulator, (b) 3D rendering as an appearance forward model and (c) black-box optimization as the solution to decoupling inference from modelling, we come up with a framework that introduces a novel forward model for dynamic scenes. We use this framework to tackle a series of challenging vision-based tracking scenarios. To the best of our knowledge, none of the existing techniques can cope with the complexity of these scenarios. Still, all of them are treated invariably and are handled effectively within the proposed framework.

As an example, in a table-top scenario, regardless of how many cameras overlook the scene, there are always problems related to *observability* due to occlusions. While a hand transports an object, we do not directly perceive the hand touching it. Still, we know that touching, *i.e*. force exertion, happens, because otherwise the object could not be lifted. Our recently proposed method in [17] demonstrates multicamera-based joint hand-object tracking that is performed based on two criteria: (a) the appearance of the hand-object ensemble matches the observations and (b) the

hand does not share the same space as the object. Both criteria are fulfilled by a hand that has proper articulation, is close to the object but does not touch it. But if the hand manages to lift/transport the object, it is clear that the above interpretation is not plausible. Being physics-based, our framework is forced to compute a plausible solution.

Another challenging observability issue is severe occlusions, *e.g*. caused because of *containment*. In the example of the shell game (Fig. 1), a ball being covered by a cup can no longer be seen. However, as humans, we do hold expectations over an evolving scene, despite the complexity of interaction and the severe and temporally extended occlusions. This has been successfully identified in [19], where "object permanence", *i.e*. the expectation of an occluded object reappearing close to its occluder, gave rise to a discrete logic that can handle challenging tracking scenarios. Still, the "object permanence" principle will fail if the ball-in-the-cup passes over a hole of the table. Containment and the resulting occlusions go beyond the heuristic of "object permanence". The laws of physics guarantee that a ball that is trapped between a cup and the table, has to travel inside the cup being moved by a hand. Such cases, where *lack of observation* can be remedied by the consideration of physics, are effortlessly handled within our framework.

Another important issue is *scene cardinality*. In the approaches taken by [1, 12, 17, 18], tracking additional rigid entities requires increasing the problem dimensionality which, in turn, makes optimization increasingly harder. Instead, within our framework and as long as the *single actor hypothesis* holds, tracking scenes of different cardinalities does not alter the dimensionality of the problem.

3. Methodology

Dynamics, as a rich and powerful modelling tool, constitutes an excellent framework where the *single actor hypothesis* is naturally expressed. It is rich because it introduces new types of data, such as mass, energy, friction, restitution, *etc*. Additionally, it is powerful because the predictive power of dynamics is the most elaborate reflection of how entities interact in a truly physical world.

The approach can be summarized as follows. A hand motion is sought that best explains the evolution of a scene between two consecutive time instants t and $t+1$. Hand motion is parameterized as the transition from a reference hand pose h_t (*e.g*. provided by tracking) to a new hand pose h_{t+1} and, thus, is defined by h_{t+1}, alone. As new observations arrive, a new tracking frame is defined, for which the tracking solution is established by a hypothesize-and-test fashion, driven by Particle Swarm Optimization (PSO) [11]. Hypotheses of hand motion are tested in a physics-based simulation environment and the outlook of the induced scene state is rendered into maps that are comparable to the observations. The discrepancy between observations and rendered hypotheses is quantified in an objective function that is minimized through PSO. The sought solution is a physically plausible scene interpretation that is most compatible with the observations.

3.1. Forward model

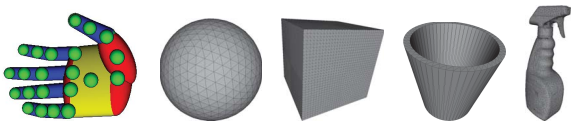
We use a forward model that regards the physical state of a scene and its appearance, as observed by a camera. This model is parameterized over a hand motion, *i.e*. two hand poses (a source and a target) in successive time instants. Given a hand motion, forward modelling produces two different outputs. First, through dynamics simulation, it updates the poses and velocities of objects, as these have been altered due to the hypothesized hand motion. Second, the resulting scene state is rendered so that a direct comparison between hypotheses and actual observations is possible.

3.1.1 Dynamics model

We are interested in table-top scenes, that consist of a static table, multiple objects and a right hand, all in 3D. All entities are represented in a dynamics simulator (*Bullet* [6]). Entities are essentially represented as 3D shapes with inertia tensors, masses, friction and restitution coefficients. Inertia tensors and masses reflect a body's resistance towards accelerations. Friction coefficients express the amount of energy that is transferred from body collisions to tangential accelerations. Restitution factors modulate the amount of energy that is lost during collisions. All of the above hardly reflect realistic conditions and only bare relative significance, since they are simulator- or application-specific. Still, the selected simulator can generate realistic dynamic behaviour, which is the key in extracting physically plausible scene interpretations.

Bullet supports collision shape representations such as analytical expressions, convex hulls and arbitrary shapes in the form of triangular meshes. We represent a table as an appropriately sized and positioned parallelepiped. Its mass and inertia are infinite so that it is immovable. Shapes can be provided as a pre-production or post-production specification of everyday objects (Fig. 2). For the case of shapes like boxes, ellipses, cylinders and their compounds, analytical representations are used as collision models. In all other cases, we resort to triangular shape approximations. All objects are considered to be of equal mass and the inertia tensors are automatically computed by *Bullet,* through shape analysis. The gravitational force is always exerted, at all objects, in a direction that is opposite to the normal vector of the table's top surface, and with a magnitude of $10m/s^2$.

The human hand is a special case. It's 3D structure is defined similarly to [16] and thus is represented by 27 parameters, that regard the absolute 3D pose of the palm and



(a) (b) (c) (d) (e)

Figure 2: The physical entities that are considered in our framework. The hand model (a) comprises 22 ellipses and 15 cylinders, appropriately positioned, rotated and scaled. The collision spheres (green) inside the hand model give it physical substance. We consider a variety of object specifications such as a ball and and box (analytical expressions) a cup (designed and then printed) and a bottle (3D scanned).

the relative articulation of the fingers. In contrast to the rest of the objects, the hand's collision model is abstracted in order to make modelling and inference tractable. The hand is able to change the state of the scene by means of forces that are the result of its accelerated surface contacting the surface of the objects. We approximate the effective surface of the hand by a compound of spheres that are strategically inscribed at various locations inside the 3D volume of the hand's structure (Fig. 2(a)). If s_k is the k-th sphere of the collision model and its 3D position is given through the application of the kinematics function $\mathbf{K}_k(h)$ for a hand pose h, then for a hand motion from h_t to h_{t+1}, s_k is given a velocity $\vec{v}_k = \left(\mathbf{K}_k\left(h_{t+1}\right) - \mathbf{K}_k\left(h_t\right)\right)/\Delta t$. Due to their mass, velocity and friction, these spheres can act as point forces on the surface of the scene's objects. The spheres of the hand's collision model are not allowed to rotate, so that all tangential collision energy is transferred to the colliding object and is not spent on the rotation of the spheres, too. This enables the hand to pick up objects without allowing them to slide through rolling spheres. The collisions among the spheres are ignored so as to better approximate the flexibility of the hand's surface, by accounting for the whole hand collision model as a union rather as a collection of independent entities. By modulating the mass and friction coefficient of the spheres the hand becomes less/more capable of manipulating heavy or slippery objects. No gravitational force is assigned to the spheres as it is assumed to always be eliminated by the torques of the hand joints.

For a hand motion, a given state of the rest of the scene and a time step, simulation of dynamics is responsible for evolving the scene into a new, physically plausible state. The hand spheres bare kinetic energy and transfer that energy, through collision, to other objects. Dynamics simulation is responsible for applying collision checking, force direction estimation and preservation of energy and momentum in order to transform the old scene state to the new one. All states contain information that regards pose (position and orientation, and thus, potential energy) and velocity (and therefore kinetic energy) for every entity being simulated.

In notation, if h_t is the initial hand pose of a hand motion and h_{t+1} is the target hand pose, then the next scene state S_{t+1} is computed via the simulation process $\mathbf{S}$, as a function of the current scene state S_t:

$$S_{t+1} = \mathbf{S}\left(h_t, h_{t+1}, S_t\right), \tag{1}$$

where S_k is the full description of the physical state at time k for N entities:

$$\mathrm{S}_k = \{\{s_i, m_i, I_i, F_i, R_i, \vec{p}_i, \vec{q}_i, \vec{v}_i, \vec{\alpha}_i\} \,|i = 1 \ldots N\} \tag{2}$$

In Eq.(2), s_i is the collision shape, m_i is the mass, I_i is the inertia tensor, F_i is the friction coefficient, R_i is the restitution coefficient, $\vec{p}_i$ is the position, $\vec{q}_i$ is the orientation, $\vec{v}_i$ is the linear velocity and $\vec{\alpha}_i$ is the angular velocity of body i, at time step k. For each individual object, the applied forces and torques are the accumulated result of the total simulated interaction. All vectors are in 3D. The time step Δt is inferred from t and $t+1$.

3.1.2 Appearance model

Every hand motion hypothesis yields a new expectation over the physical state of the scene. This expectation needs to be made comparable to observations so that the corresponding hypothesis can be evaluated. A hypothesis scores well when its simulated expectations over the scene evolution match the new observations well.

In this work, observations come from an RGBD camera that provides an RGB image and an aligned depth image. Both images are in standard VGA resolution. Foreground pixels are identified through background subtraction [27] on the RGB part of the observations. A background model is built from a scene where the expected foreground, the actor and the objects, is missing. Once the pixels of interest I_l^o have been identified, the respective depth values in the depth image are extracted in a filtered depth image I_d^o (Fig. 3(d)).

Every physical state S_k that is generated by function $\mathbf{S}$ contains enough information so that data comparable to I_l^o and I_d^o can be generated (Fig. 3(e)). In detail, given the calibration information c of the RGBD camera, S_k can be rasterized. Each element in S_k is represented by a shape that is (or can be approximated as) a triangular mesh. For the special case of the hand, we follow an approach similar to that of [16]. All meshes are rendered with respect to their positions and orientations, in a virtual camera described by c. From this rendering, two channels of information are kept, (a) a map I_l^r that is set for pixels occupied by geometry projections and (b) a map I_d^r that holds the depth value of each rendered pixel in I_l^r. In notation, an observation function $\mathbf{O}$ generates two maps at time step k:

$$\{I_l^o, I_d^o\} = \mathbf{O}\left(k\right). \tag{3}$$

(a) (b) (c) (d) (e) (f) (g)

Figure 3: (a), (b) RGBD input. (d) Masked depth image I_d^o. For a hand motion hypothesis h (c), a synthetic depth map I_d^r is rendered (e). The difference between I_d^o and I_d^r (f) yields the fitness of h. The best scoring h, computed by PSO, is the tracking solution (g) for the current frame.

Given a state S_k and calibration information c, a rendering function $\mathbf{R}$ generates comparable maps:

$$\{I_l^r, I_d^r\} = \mathbf{R}\left(S_k, c\right). \tag{4}$$

3.2. Inference

In order to infer total state change from new observations we formulate an optimization problem, which we solve for the hand motion alone. All scene changes are attributed to hand intervention, which, in optimization terms, amounts to 27 parameters. Thus, at any tracking iteration at time t, given (a) the hand position h_t and (b) the state of the scene S_t, we seek for a new hand pose h_{t+1} defined as

$$h_{t+1} \triangleq \arg\min_{h} \mathbf{E}\left(h_t, h, S_t, c, t\right). \tag{5}$$

h_{t+1} must be such that the motion of the hand from h_t to h_{t+1} best explains the observed evolution of the scene. The resulting scene state S_{t+1} that accompanies h_{t+1} is provided by Eq. (1). Function $\mathbf{E}$ defines a penalty to be minimized over hand motion hypotheses. It is a linear combination of a prior term $\mathbf{P}$ and a data term $\mathbf{D}$:

$$\mathbf{E}\left(h', h'', S, c, t\right) = \lambda_P \mathbf{P}\left(h''\right) + \lambda_D \mathbf{D}\left(h', h'', S, c, t\right) \tag{6}$$

During optimization, the range of all possible hand motions is considered. However, we need to penalize for hand motions that contain inter-penetrations of distinct hand sub-parts (*e.g.* fingers). Thus, $\mathbf{P}$ is defined as:

$$\mathbf{P}\left(h\right) = \sum_{\{i,j\} \in \mathbf{C_M}(h)} \mathbf{P_D}\left(s_i, s_j\right), \tag{7}$$

where function $\mathbf{C_M}$ provides the collision check pairs for the sub-parts, s_k is the k-th collision element and function $\mathbf{P_D}$ computes pair-wise penetration depth, that is computed by the simulator. The data term $\mathbf{D}$ combines equations (1), (3) and (4) to quantify the difference between the observation of a scene and the expected outcome of a hand motion hypothesis:

$$\mathbf{D}\left(h', h'', S, c, t\right) = \mathbf{F}\left(\mathbf{O}\left(t\right), \mathbf{R}\left(\mathbf{S}\left(h', h'', S\right), c\right)\right) \tag{8}$$

with

$$\mathbf{F}\left(o, r\right) = \lambda_F \mathbf{D_D}\left(o, r\right) + \left(1 - \lambda_F\right) \mathbf{D_S}\left(o, r\right), \tag{9}$$

$$\mathbf{D_D}\left(I_d^o, I_d^r\right) = \frac{\sum \min\left(\left|I_d^o - I_d^r\right|, T_d\right)/T_d}{\sum\left(I_l^o \vee I_l^r\right) + \varepsilon}, \tag{10}$$

$$\mathbf{D_S}\left(I_l^o, I_l^r\right) = 1 - \frac{2\sum\left(I_l^o \wedge I_l^r\right)}{\sum\left(I_l^o \vee I_l^r\right) + \sum\left(I_l^o \wedge I_l^r\right) + \varepsilon}. \tag{11}$$

All operands are $2D$ maps and all operators, arithmetic and logical, are pixel-wise. Summations are performed over the area of the $2D$ operands. A small term ϵ is added to denominators to avoid divisions by zero.

$\mathbf{D_D}$ represents depth comparisons between observed and hypothesized scenes (Fig. 3(f)). All differences are clamped in the range $[0, T_d]$, so that overly large differences (mostly due to noise) don't dominate. Then, differences are normalized in the range $[0, 1]$. Subsequently, they are reduced in a normalized sum, over the entire image.

$\mathbf{D_S}$ represents the overlap of rendered and observed silhouettes. We prefer hypotheses where all observed silhouettes are accounted for by predicted silhouettes. This is complementary to depth differences and acts as a safeguard against multiple local minima of $\mathbf{D_D}$. More specifically, we try to avoid hypotheses that yield strong depth matching in only small parts of the image. Values are in the range $[0, 1]$.

In order to minimize $\mathbf{E}$ we use Particle Swarm Optimization (PSO) [11]. PSO has been selected because of its optimization performance [10]. Additionally, it is parallel in nature, which allows for accelerated execution in parallel architectures. Minimizing $\mathbf{E}$ amounts to invoking it several thousand times per tracking frame. Every invocation involves 3D rendering and dynamics simulation, both being computationally demanding tasks. GPU architectures are used in order to accelerate rendering and multicore CPU architectures are exploited for the acceleration of dynamics simulation for each PSO generation.

3.3. Tracking loop

Given a table-top scene containing objects and a human hand, initialization is performed. The table is detected by means of RANSAC plane fitting, and from its normal vector the gravity vector's direction is inferred. 3D models of the objects that can be found in the scene are assumed to be available in a database. The initial registration (estimating 3D position and orientation) of these models to the actual 3D point cloud provided by the RGBD camera is performed using the method described in [20]. The initial configuration of the hand is predefined (Fig. 5).

A new iteration of tracking is performed as soon as new observations become available at time t. As a first step, foreground segmentation is applied. A hypothesis of a new hand pose h_t amounts to a relative motion with respect to

h_{t-1}. All simulations conducted to estimate S_t are evolutions of S_{t-1}. PSO is delegated with the task of minimizing the penalty function $\mathbf{E}$, for the new observations, in order to find the minimizer hand motion. The PSO population of hand poses is initialized in the vicinity of the solution for the previous frame, so that search is more efficient on image sequences that are sampled densely in time. As PSO searches for the fittest hypothesis, multiple scenarios of interaction are simulated, rendered and compared to actual observations. Although the actor's arm/body is not modelled and belongs to the foreground, it does not influence inference. This is because the computations of Eq. (5) are performed inside a 2D bounding box of the image projection of each and every modelled entity (hand, objects). Being tight, such bounding boxes contain very few observations of the actor other than the hand. The hypothesis that optimally explains the evolution of a scene in terms of appearance is dubbed as the tracking solution for the current frame. The scene state that accompanies the winning hypothesis replaces S_{t-1} for the next tracking frame.

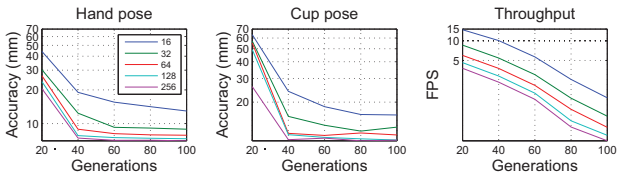
4. Experiments

For the evaluation of the presented framework we employed a parallel implementation. We used GPU threads for 3D rendering and objective evaluation and CPU threads for dynamics simulation. All experiments were executed on a machine with the following specifications: quad-core Intel i7 920 CPU, 6 GBs RAM and a 1581GFlops Nvidia GTX 580 GPU with 1.5 GBs RAM. For the image acquisition process we employed a Kinect sensor and the OpenNI framework. Acquisition was performed at a $30fps$ rate and therefore the dynamics simulation time interval was set to $\Delta t = 1/30s$. This interval was subsequently subdivided into smaller intervals ($10\times$), so as to remedy the lack of robust continuous collision detection for complex geometries.

Unless otherwise stated, PSO parameterization amounted to 64 particles and 100 generations. Each hypothesis was rendered in a series of surfaces (one per simulated entity) of 50×50 pixels each. The effective area over the observations that rendering and evaluation considered at each step was based on per-entity bounding box computations for the solution of the previous tracking step. The values $\lambda_P = 10$, $\lambda_D = 1$ and $\lambda_F = 0.9$ were used in all experiments. The mass of the hand model collision spheres was set to 1 and the friction factor to 10. This combination of factors yields a powerful and dexterous hand, that still requires at least two opposing fingers in order to pick up objects. T_d was set to $40mm$.

4.1. Quantitative evaluation

For the problem of scene tracking and when a hand is involved, it is very difficult to acquire ground truth information. In an effort to produce data that can be used as a ground truth we conducted the following experiment. A human hand grasped a cup firmly, lifted it and moved it around in various angles. In a sequence of 500 frames, the hand grasped the cup firmly in the last 370 frames (see the 1st column of Fig. 5). By construction, the pose of the hand was correlated with the pose of the cup. We tracked this scene and thus gained access to the inferred poses of the hand and the cup. We measured the standard deviation of the distance between the estimated position of the hand and that of the object, during grasping, and we found this to be $3.7mm$. Similarly, the standard deviation of the difference in hand/cup poses was equal to $1.3°$. These measurements indicate that tracking was successful in capturing the tight spatial hand/object relationship, despite the object's rotational symmetry, that hinders orientation estimation, and despite the strong hand-hand and hand-cup occlusions.



(a) (b) (c)

Figure 4: Quantitative results. Distinct curves correspond to different particle counts. Logarithmic scale was used in the vertical axes for better resolution over small value differences.

In a second experiment, we tested the optimizer's ability to effectively solve the tracking problem by using synthetic data for which ground truth was available. Such data (depths and silhouettes) were produced from the tracking result of the previous experiment. Being an output of our framework, these data are compatible by construction. We tested different budgets (*i.e.*, allowed count of objective function evaluations) that were distributed across various particle and generation count combinations. For each budget, a tracking experiment was repeated 100 times and the distance between the resulting track and the ground truth was measured using the accuracy measure proposed in [17]. The corresponding accuracy results are shown in Fig. 4(a) and Fig. 4(b), where each point represents the median error across all tracking repetitions. The tracking frame rate for each budget is shown in Fig. 4(c).

What can be seen from Fig. 4(a) and Fig. 4(b) is that generally, as budget grows, a better accuracy is achieved. As expected, we can trade accuracy for speed. In the synthetic experiments even low budgets suffice for adequately accurate tracking of both the hand and the object. There, budgets that yielded as much as $3fps$ could produce adequately accurate tracks. For real-world experiments, exhibiting non-

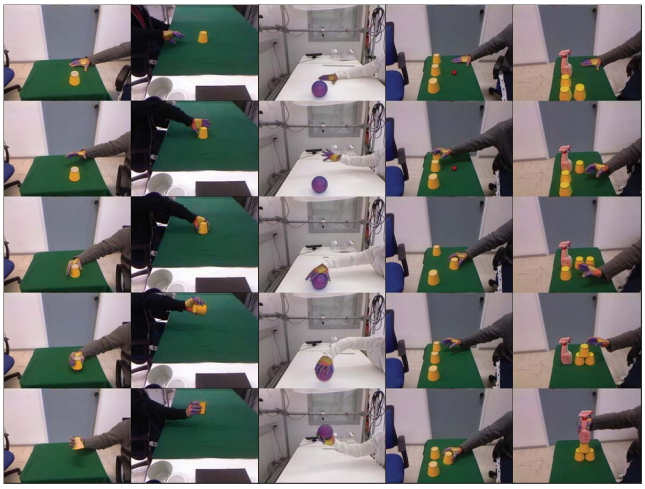


Figure 5: Tracking results, superimposed over the respective RGB input. Leftmost column: the sequence that was used for quantitative assessment. Rest columns: qualitative evaluation. Detailed presentation of the entire sequences that highlight the efficacy of the proposed method can be found in the supplementary material.

ideal observations and inexact modelling, we resorted to a greater budget that still yielded acceptable performance (64 particles and 100 generations, $0.5fps$).

4.2. Qualitative evaluation

We conducted a series of real-life experiments to test whether (a) our modeling of dynamics and appearance is adequate to match real observations (b) tracking complex interactions can be achieved with the proposed optimization framework and, (c) our *single actor hypothesis* is well reflected "in the wild". In all experiments, optimization was performed only on the parameters of the actor's hand. The (correct) motion of all objects was inferred as a consequence of the hand's motion that best explained the observations in total. All videos were recorded at 30fps. They were processed in $1\times$ and $2\times$ speeds, in order to simulate both normal and faster motion, yielding identical tracking performance. The recorded sequences and the respective results can be viewed at `http://youtu.be/0RCsQPXeHRQ`.

The first experiment considered a hand and a cup (2^{nd} column of Fig. 5). At this footage the hand picked up the cup and put it back on the table in an upside-down orientation. As it can be verified in Fig. 5, the proposed method successfully provided a physically plausible track.

In the second experiment a hand lifted and manipulated a plastic bowling ball that was barely graspable due to its size (3^{rd} column of Fig. 5). Given enough friction, our hand modelling was able to explain the lifting and manipulation of the object. Even when almost the entire hand was occluded by the ball we came up with plausible hypotheses. In both previous situations, consulting physics yielded that an object that was to remain in the air required finger support, which, even if barely observed, was hypothesized in order to achieve overall consistency with the observations.

In the third experiment we considered a more elaborate case of interaction that induced cascaded occlusions. We demonstrated the shell game with three cups and one ball (4^{th} column of Fig. 5). One cup trapped the ball and was moved around, moving the ball inside it and pushing other cups when in its way. At some point in time, a chained interaction occurred. The hand pushed an empty cup, which in turn pushed the cup containing the ball. As the hand shuffled the cups intense occlusions occurred that did not prevent our framework from maintaining plausible hypotheses about the 3D position and orientation of the fully/partially occluded hand, cups and of the truly invisible ball.

A final experiment regarded object stacking. Concave objects were stacked, one by one, forming a pyramid (5^{th} column of Fig. 5). This scenario challenged both the dynamics modelling and the optimization module, because stacking of generic geometry is indeed a difficult problem for dynamics simulators to handle stably, which in turn yields an erratic behaviour in the objective function. However, PSO, overcoming the problem, yielded plausible hypotheses and thus the pyramid was tracked well.

5. Summary

In this work we enabled the efficient 3D tracking of complex scenes by exploiting the *single actor hypothesis*. To achieve this, we proposed the use of a dynamics model (physics simulator) and a appearance model (3D rendering) as a powerful, combined forward model, that is turned into an inference mechanism by means of black-box optimization. The gains from considering this inference framework include: (a) the successful, principled and uniform treatment of several tracking scenarios that pertain to challenging observability issues, (b) scene tracking solutions that are, by construction, physically plausible, (c) problem dimensionality that does not increase with the number of scene objects involved. This has been demonstrated in a series of qualitative and quantitative experiments on challenging scenarios, exhibiting complex interactions of a human hand with several objects.

By introducing physics in 3D tracking of multi-entity scenes, the proposed approach, establishes the foundations upon which extensions can be built and can lead to even more effective solutions. A natural extension of this work would be to consider a wider observation horizon in order to tackle cases where the hand is not constantly observed to manipulate objects but only initiates motion by passing kinetic energy. For such cases, more observations of the object are required in order to better estimate a velocity profile. Other extensions might consider additional/adjusted

priors that would regard more specific tracking scenarios. The access that is provided to dynamic aspects of motion can be used to fuel higher level inference. Notably, the *single actor hypothesis* does not constrain the actor to be single but only that all source of state change is directly and efficiently modelled: it can also regard the extension to two active hands, an active body or even active objects, *etc*. Interestingly, testing different/larger forward models is made easy due to the decoupling of modelling from optimization.

Acknowledgments

This work was partially supported by the IST-FP7-IP-288533 project RoboHow.Cog. The contribution of Nikolaos Pateromichelakis, member of the CVRL laboratory, is gratefully acknowledged.

References

[1] L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, 2012.

[2] K. Bhat, S. Seitz, J. Popović, and P. Khosla. Computing the physical parameters of rigid-body motion from video. In *ECCV*, 2002.

[3] M. Brand. Physics-Based Visual Understanding* 1. *CVIU*, 65(2), 1997.

[4] M.A. Brubaker, D.J. Fleet, and A. Hertzmann. Physics-based person tracking using the anthropomorphic walker. *IJCV*, 87(1), 2010.

[5] M.A. Brubaker, L. Sigal, and D.J. Fleet. Estimating contact dynamics. In *ICCV*, 2009.

[6] Erwin Coumans. Bullet game physics simulation, 2011.

[7] Q. Delamarre and O. Faugeras. 3D Articulated Models and Multiview Tracking with Physical Forces* 1. *CVIU*, 81(3), 2001.

[8] D.J. Duff, J. Wyatt, and R. Stolkin. Motion estimation using physical simulation. In *ICRA*, 2010.

[9] A. Gupta, A. Efros, and M. Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *ECCV*, 2010.

[10] N. Hansen, A. Auger, R. Ros, S. Finck, and P. Pošík. Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009. In *GECCO*. ACM, 2010.

[11] J. Kennedy, R.C. Eberhart, and Yuhui. Shi. *Swarm intelligence*. Morgan Kaufmann Publ., 2001.

[12] H. Kjellstrom, D. Kragic, and M. Black. Tracking people interacting with objects. In *CVPR*, 2010.

[13] N. Kyriazis, I. Oikonomidis, and A. Argyros. Binding vision to physics based simulation: The case study of a bouncing ball. In *BMVC*. BMVA, 2011.

[14] R. Mann, A. Jepson, and J. Siskind. The computational perception of scene dynamics. *CVIU*, 65(2), 1997.

[15] D. Metaxas and D. Terzopoulos. Shape and non-rigid motion estimation through physics-based synthesis. *PAMI*, 15(6), 1993.

[16] I. Oikonomidis, N. Kyriazis, and A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *BMVC*. BMVA, 2011.

[17] I. Oikonomidis, N. Kyriazis, and A. Argyros. Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *ICCV*, 2011.

[18] I. Oikonomidis, N. Kyriazis, and A.A. Argyros. Tracking the articulated motion of two strongly interacting hands. In *CVPR*, 2012.

[19] V. Papadourakis and A. Argyros. Multiple objects tracking in the presence of long-term occlusions. *CVIU*, 114(7), 2010.

[20] C. Papazov and D. Burschka. Stochastic global optimization for robust point set registration. *CVIU*, 2011.

[21] Z. Popović and A. Witkin. Physically based motion transformation. In *Conf. on Computer Graphics and Interactive Techniques*. ACM, 1999.

[22] M. Salzmann and R. Urtasun. Physically-based motion models for 3d tracking: A convex formulation. In *ICCV*, 2011.

[23] R. Urtasun, D.J. Fleet, and P. Fua. Monocular 3d tracking of the golf swing. In *CVPR*, 2005.

[24] M. Vondrak, L. Sigal, and OC Jenkins. Physical simulation for probabilistic motion tracking. In *CVPR*, 2008.

[25] J.M. Wang, D.J. Fleet, and A. Hertzmann. Optimizing walking controllers. *ACM Trans. on Graphics (TOG)*, 28(5), 2009.

[26] Y. Ye and C.K. Liu. Synthesis of detailed hand manipulations using contact sampling. *ACM Transactions on Graphics (TOG)*, 31(4), 2012.

[27] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *ICPR*, 2004.