Real-Time Joint Tracking of a Hand Manipulating an Object from RGB-D Input

Srinath Sridhar[1], Franziska Mueller[1], Michael Zollhöfer[1], Dan Casas[1], Antti Oulasvirta[2], and Christian Theobalt[1(✉)]

[1] Max Planck Institute for Informatics, Saarbrücken, Germany
{ssridhar,frmueller,mzollhoef,dcasas,theobalt}@mpi-inf.mpg.de
[2] Aalto University, Espoo, Finland
antti.oulasvirta@aalto.fi

Abstract. Real-time simultaneous tracking of hands manipulating and interacting with external objects has many potential applications in augmented reality, tangible computing, and wearable computing. However, due to difficult occlusions, fast motions, and uniform hand appearance, jointly tracking hand and object pose is more challenging than tracking either of the two separately. Many previous approaches resort to complex multi-camera setups to remedy the occlusion problem and often employ expensive segmentation and optimization steps which makes real-time tracking impossible. In this paper, we propose a real-time solution that uses a single commodity RGB-D camera. The core of our approach is a 3D articulated Gaussian mixture alignment strategy tailored to hand-object tracking that allows fast pose optimization. The alignment energy uses novel regularizers to address occlusions and hand-object contacts. For added robustness, we guide the optimization with discriminative part classification of the hand and segmentation of the object. We conducted extensive experiments on several existing datasets and introduce a new annotated hand-object dataset. Quantitative and qualitative results show the key advantages of our method: speed, accuracy, and robustness.

1 Introduction

The human hand exhibits incredible capacity for manipulating external objects via gripping, grasping, touching, pointing, caging, and throwing. We can use our hands with apparent ease, even for subtle and complex motions, and with remarkable speed and accuracy. However, this dexterity also makes it hard to track a hand in close interaction with objects. While a lot of research has explored real-time tracking of hands or objects in isolation, real-time hand-object tracking

Electronic supplementary material The online version of this chapter (doi:10.1007/978-3-319-46475-6_19) contains supplementary material, which is available to authorized users.

B. Leibe et al. (Eds.): ECCV 2016, Part II, LNCS 9906, pp. 294–310, 2016.
DOI: 10.1007/978-3-319-46475-6_19

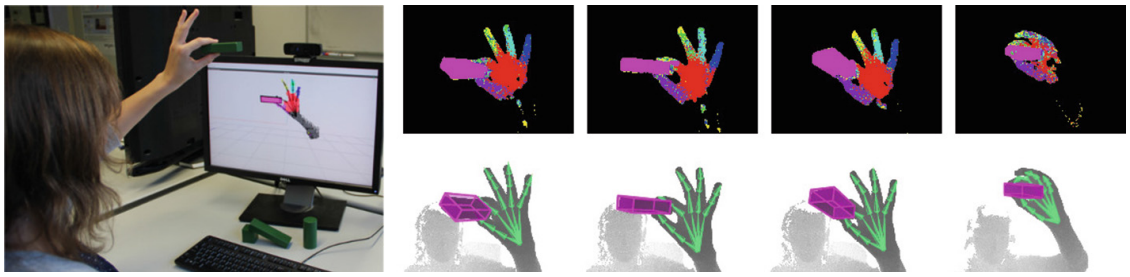


Fig. 1. Proposed real-time hand-object tracking approach: we use a single commodity depth camera (*left*) to classify (*top*) and track the articulation of a hand and the rigid body motion of a manipulated object (*bottom*)

remains unsolved. It is inherently more challenging due to the higher dimensionality of the problem, additional occlusions, and difficulty in disambiguating hand from object. A fast, accurate, and robust solution based on a minimal camera setup is a precondition for many new and important applications in vision-based input to computers, including virtual and augmented reality, teleoperation, tangible computing, and wearable computing. In this paper, we present a **real-time** method to **simultaneously track** a hand and the manipulated object. We support tracking objects of **different shapes, sizes**, and **colors**. Previous work has employed setups with multiple cameras [5,17] to limit the influence of occlusions which restricts use to highly controlled setups. Many methods that exploit dense depth and color measurements from commodity RGB-D cameras [8,13,14] have been proposed. However, these methods use expensive segmentation and optimization steps that make interactive performance hard to attain. At the other end of the spectrum, discriminative one-shot methods (for tracking only hands) often suffer from temporal instability [11,33,43]. Such approaches have also been applied to estimate hand pose under object occlusion [24], but the object is not tracked simultaneously. In contrast, the approach proposed here is the first to track hand and object motion simultaneously at real-time rates using only a single commodity RGB-D camera (see Fig. 1). Building on recent work in single hand tracking and 3D pointset registration, we propose a 3D articulated Gaussian mixture alignment strategy tailored to hand-object tracking. Gaussian mixture alignment aligns two Gaussian mixtures and has been successfully used in 3D pointset registration [10]. It can be interpreted as a generalization of ICP and does not require explicit, error-prone, and computationally expensive correspondence search [7]. Previous methods have used articulated 2.5D Gaussian mixture alignment formulations [27] that are discontinuous. This leads to tracking instabilities because 3D spatial proximity is not considered. We also introduce additional novel regularizers that consider occlusions and enforce contact points between fingers and objects analytically. Our combined energy has a closed form gradient and allows for fast and accurate tracking. For an overview of our approach see Fig. 2. To further increase robustness and allow for recovery of the generative tracker, we guide the optimization using a multi-layer random forest hand part classifier. We use a variational optimization strategy that optimizes

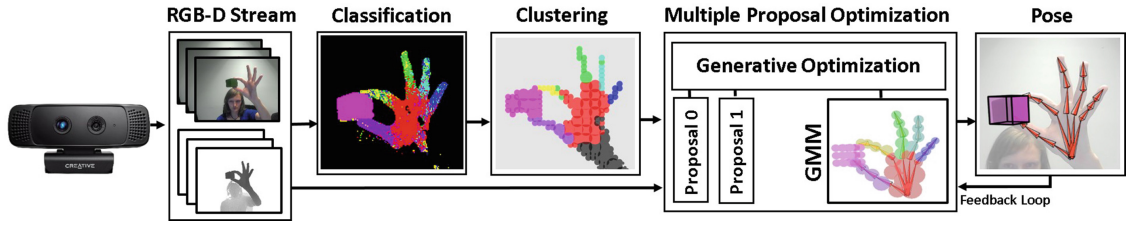


Fig. 2. We perform classification of the input into object and hand parts. The hand and object are tracked using 3D articulated Gaussian mixture alignment

two different hand-object tracking energies simultaneously (multiple proposals) and then selects the better solution. The main contributions are:

- A 3D articulated Gaussian mixture alignment approach for jointly tracking hand and object accurately.
- Novel contact point and occlusion objective terms that were motivated by the physics of grasps, and can handle difficult hand-object interactions.
- A multi-layered classification architecture to segment hand and object, and classify hand parts in RGB-D sequences.
- An extensive evaluation on public datasets as well as a new, fully annotated dataset consisting of diverse hand-object interactions.

2 Related Work

Single Hand Tracking. Single hand tracking has received a lot of attention in recent years with discriminative and generative methods being the two main classes of methods. Discriminative methods for monocular RGB tracking index into a large database of poses or learn a mapping from image to pose space [3,42]. However, accuracy and temporal stability of these methods are limited. Monocular generative methods optimize pose of more sophisticated 3D or 2.5D hand models by optimizing an alignment energy [6,9,15]. Occlusions and appearance ambiguities are less problematic with multi-camera setups [5]. [41] use a physics-based approach to optimize the pose of a hand using silhouette and color constraints at slow non-interactive frame rates. [28] use multiple RGB cameras and a single depth camera to track single hand poses in near real-time by combining generative tracking and finger tip detection. More lightweight setups with a single depth camera are preferred for many interactive applications. Among single camera methods, examples of discriminative methods are based on decision forests for hand part labeling [11], on a latent regression forest in combination with a coarse-to-fine search [33], fast hierarchical pose regression [31], or Hough voting [43]. Real-time performance is feasible, but temporal instability remains an issue. [19] generatively track a hand by optimizing a depth and appearance-based alignment metric with particle swarm optimization (PSO). A real-time generative tracking method with a physics-based solver was proposed in [16]. The stabilization of real-time articulated ICP based on a learned subspace prior

on hand poses was used in [32]. Template-based non-rigid deformation tracking of arbitrary objects in real-time from RGB-D was shown in [45], very simple unoccluded hand poses can be tracked. Combining generative and discriminative tracking enables recovery from some tracking failures [25,28,39]. [27] showed real-time single hand tracking from depth using generative pose optimization under detection constraints. Similarly, reinitialization of generative estimates via finger tip detection [23], multi-layer discriminative reinitialization [25], or joints detected with convolutional networks is feasible [36]. [34] employ hierarchical sampling from partial pose distributions and a final hypothesis selection based on a generative energy. None of the above methods is able to track interacting hands and objects simultaneously and in non-trivial poses in real-time.

Tracking Hands in Interaction. Tracking two interacting hands, or a hand and a manipulated object, is a much harder problem. The straightforward combination of methods for object tracking, e.g. [4,35], and hand tracking does not lead to satisfactory solutions, as only a combined formulation can methodically exploit mutual constraints between object and hand. [40] track two well-separated hands from stereo by efficient pose retrieval and IK refinement. In [18] two hands in interaction are tracked at 4 Hz with an RGB-D camera by optimizing a generative depth and image alignment measure. Tracking of interacting hands from multi-view video at slow non-interactive runtimes was shown in [5]. They use generative pose optimization supported by salient point detection. The method in [32] can track very simple two hand interactions with little occlusion. Commercial solutions, e.g. Leap Motion [1] and NimbleVR [2], fail if two hands interact closely or interact with an object. In [17], a marker-less method based on a generative pose optimization of a combined hand-object model is proposed. They explicitly model collisions, but need multiple RGB cameras. In [8] the most likely pose is found through belief propagation using part-based trackers. This method is robust under occlusions, but does not explicitly track the object. A temporally coherent nearest neighbor search tracks the hand manipulating an object in [24], but not the object, in real time. Results are prone to temporal jitter. [13] perform frame-to-frame tracking of hand and objects from RGB-D using physics-based optimization. This approach has a slow non-interactive runtime. An ensemble of Collaborative Trackers (ECT) for RGB-D based multi-object and multiple hand tracking is used in [14]. Their accuracy is high, but runtime is far from real-time. [21] infer contact forces from a tracked hand interacting with an object at slow non-interactive runtimes. [20,38] propose methods for in-hand RGB-D object scanning. Both methods use known generative methods to track finger contact points to support ICP-like shape scanning. Recently, [37] introduced a method for tracking hand-only, hand-hand, and hand-object (we include a comparison with this method). None of the above methods can track the hand and the manipulated object in *real-time* in non-trivial poses from a *single depth camera* view, which is what our approach achieves.

Model-Based Tracking Approaches. A common representation for model tracking are meshes [5,32]. Other approaches use primitives [14,23], quadrics [29], 2.5D Gaussians [27], or Gaussian mixtures [10]. Gaussian mixture alignment has

been successfully used in rigid pointset registration [10]. In contrast, we propose a 3D *articulated* Gaussian mixture alignment strategy. [44] relate template and data via a probabilistic formulation and use EM to compute the best fit. Different from our approach, they only model the template as a Gaussian mixture.

3 Discriminative Hand Part Classification

As a preprocessing step, we classify depth pixels as hand or object, and further into hand parts. The obtained labeling is later used to guide the generative pose optimization. Our part classification strategy is based on a two-layer random forest that takes occlusions into account. Classification is based on a three step pipeline (see Fig. 3). Input is the color $\mathcal{C}_t$ and depth $\mathcal{D}_t$ frames captured by the RGB-D sensor. We first perform hand-object segmentation based on color cues to remove the object from the depth map. Afterwards, we select a suitable two-layer random forest to obtain the classification. The final output per pixel is a part probability histogram that encodes the class likelihoods. Note, object pixel histograms are set to an object class probability of 1. The forests are trained based on a set of training images that consists of real hand motions re-targeted to a virtual hand model to generate synthetic data from multiple viewpoints. A virtual object is automatically inserted in the scene to simulate occlusions. To this end, we randomly sample uniform object positions between the thumb and one other finger and prune implausible poses based on intersection tests.

Viewpoint Selection. We trained two-layer forests for hand part classification from different viewpoints. Four cases are distinguished: observing the hand from the front, back, thumb and little finger sides. We select the forest that best matches the hand orientation computed in the last frame. The selected two-layer forest is then used for hand part classification.

Color-Based Object Segmentation. As a first step, we segment out the object from the captured depth map $\mathcal{D}_t$. Similar to many previous hand-object tracking approaches [19], we use the color image $\mathcal{C}_t$ in combination with an HSV color segmentation strategy. As we show in the results, we are able to support objects with different colors. Object pixels are removed to obtain a new depth map $\hat{\mathcal{D}}_t$, which we then feed to the next processing stage.

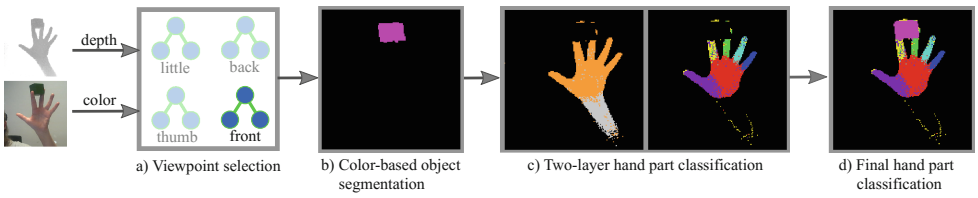


Fig. 3. Three stage hand part classification: Stage 1: Viewpoint selection, stage 2: color-based object segmentation, and stage 3: two-layer hand part classification (Color figure online)

Two-Layer Hand Part Classification. We use a two-layer random forest for hand part classification. The first layer classifies hand and arm pixels while the second layer uses the hand pixels and further classifies them into one of several distinct hand parts. Both layers are per-pixel classification forests [26]. The hand-arm classification forest is trained on $N = 100k$ images with diverse hand-object poses. For each of the four viewpoints a random forest is trained on $N = 38k$ images. The random forests are based on three trees, each trained on a random distinct subset. In each image, 2000 example foreground pixels are chosen. Split decisions at nodes are based on 100 random feature offsets and 40 thresholds. Candidate features are a uniform mix of unary and binary depth difference features [26]. Nodes are split as long as the information gain is sufficient and the maximum tree depth of 19 (21 for hand-arm forest) has not been reached. On the first layer, we use 3 part labels: 1 for hand, 1 for arm and 1 to represent the background. On the second layer, classification is based on 7 part labels: 6 for the hand parts, and 1 for the background. We use one label for each finger and one for the palm, see Fig. 3c. We use a cross-validation procedure to find the best hyperparameters. On the disjoint test set, the hand-arm forest has a classification accuracy of 65.2 %. The forests for the four camera views had accuracies of 59.8 % (front), 64.7 % (back), 60.9 % (little), and 53.5 % (thumb).

4 Gaussian Mixture Model Representation

Joint hand-object tracking requires a representation that allows for accurate tracking, is robust to outliers, and enables fast pose optimization. Gaussian mixture alignment, initially proposed for rigid pointset alignment (e.g. [10]), satisfies all these requirements. It features the advantages of ICP-like methods, without requiring a costly, error-prone correspondence search. We extend this approach to 3D articulated Gaussian mixture alignment tailored to hand-object tracking. Compared to our 3D formulation, 2.5D [27] approaches are discontinuous. This causes instabilities, since the spatial proximity between model and data is not fully considered. We quantitatively show this for hand-only tracking (Sect. 8).

5 Unified Density Representation

We parameterize the articulated motion of the human hand using a kinematic skeleton with $|\mathcal{X}_h| = 26$ degrees of freedom (DOF). Non-rigid hand motion is expressed based on 20 joint angles in twist representation. The remaining 6 DOFs specify the global rigid transform of the hand with respect to the root joint. The manipulated object is assumed to be rigid and its motion is parameterized using $|\mathcal{X}_o| = 6$ DOFs. In the following, we deal with the hand and object in a unified way. To this end, we refer to the vector of all unknowns as $\mathcal{X}$. For pose optimization, both the input depth as well as the scene (hand and object) are expressed as 3D Gaussian Mixture Models (GMMs). This allows for fast and analytical pose optimization. We first define the following generic probability density distribution $\mathcal{M}(\mathbf{x}) = \sum_{i=1}^{K} w_i \mathcal{G}_i(\mathbf{x}|\boldsymbol{\mu}_i, \sigma_i)$ at each point $\mathbf{x} \in \mathbb{R}^3$ in space.

This mixture contains K unnormalized, isotropic Gaussian functions $\mathcal{G}_i$ with mean $\boldsymbol{\mu}_i \in \mathbb{R}^3$ and variance $\sigma_i^2 \in \mathbb{R}$. In the case of the model distribution, the positions of the Gaussians are parameterized by the unknowns $\mathcal{X}$. For the hand, this means each Gaussian is being rigidly rigged to one bone of the hand. The probability density is defined and non-vanishing over the whole domain $\mathbb{R}^3$.

Hand and Object Model. The three-dimensional shape of the hand and object is represented in a similar fashion as probability density distributions $\mathcal{M}_h$ and $\mathcal{M}_o$, respectively. We manually attach $N_h = 30$ Gaussian functions to the kinematic chain of the hand to model its volumetric extent. Standard deviations are set such that they roughly correspond to the distance to the actual surface. The object is represented by automatically fitting a predefined number N_o of Gaussians to its spatial extent, such that the one standard deviation spheres model the objects volumetric extent. N_o is a user defined parameter which can be used to control the trade-off between tracking accuracy and runtime performance. We found that $N_o \in [12, 64]$ provides a good trade-off between speed and accuracy for the objects used in our experiments. We refer to the combined hand-object distribution as $\mathcal{M}_s$, with $N_s = N_h + N_o$ Gaussians. Each Gaussian is assigned to a class label l_i based on its semantic location in the scene. Note, the input GMM is only a model of the visible surface of the hand/object. Therefore, we incorporate a visibility factor $f_i \in [0, 1]$ (0 completely occluded, 1 completely visible) per Gaussian. This factor is approximated by rendering an occlusion map with each Gaussian as a circle (radius equal to its standard deviation). The GMM is restricted to the visible surface by setting $w_i = f_i$ in the mixture. These operations are performed based on the solution of the previous frame $\mathcal{X}_{old}$.

Input Depth Data. We first perform bottom-up hierarchical quadtree clustering of adjacent pixels with similar depth to convert the input to the density based representation. We cluster at most $(2^{(4-1)})^2 = 64$ pixels, which corresponds to a maximum tree depth of 4. Clustering is performed as long as the depth variance in the corresponding subdomain is smaller than $\epsilon_{cluster} = 30\,\text{mm}$. Each leaf node is represented as a Gaussian function $\mathcal{G}_i$ with $\boldsymbol{\mu}_i$ corresponding to the 3D center of gravity of the quad and $\sigma_i^2 = (\frac{a}{2})^2$, where a is the backprojected side length of the quad. Note, the mean $\boldsymbol{\mu}_i \in \mathbb{R}^3$ is obtained by backprojecting the 2D center of gravity of the quad based on the computed average depth and displacing by a in camera viewing direction to obtain a representation that matches the model of the scene. In addition, each $\mathcal{G}_i$ stores the probability p_i and index l_i of the best associated semantic label. We obtain the best label and its probability by summing over all corresponding per-pixel histograms obtained in the classification stage. Based on this data, we define the input depth distribution $\mathcal{M}_{d_h}(\mathbf{x})$ for the hand and $\mathcal{M}_{d_o}(\mathbf{x})$ for the object. The combined input distribution $\mathcal{M}_d(\mathbf{x})$ has $N_d = N_{d_o} + N_{d_h}$ Gaussians. We set uniform weights $w_i = 1$ based on the assumption of equal contribution. N_d is much smaller than the number of pixels leading to real-time hand-object tracking.

6 Multiple Proposal Optimization

We optimize for the best pose $\mathcal{X}^*$ using two proposals $\mathcal{X}_i^*,\ i \in \{0,1\}$ that are computed by minimizing two distinct hand-object tracking energies:

$$\mathcal{X}_0^* = \underset{\mathcal{X}}{\operatorname{argmin}}\, E_{align}(\mathcal{X}),\ \ \mathcal{X}_1^* = \underset{\mathcal{X}}{\operatorname{argmin}}\, E_{label}(\mathcal{X}). \tag{1}$$

E_{align} leverages the depth observations and the second energy E_{label} incorporates the discriminative hand part classification results. In contrast to the optimization of the sum of the two objectives, this avoids failure due to bad classification and ensures fast recovery. For optimization, we use analytical gradient descent (10 iterations per proposal, adaptive step length) [30]. We initialize based on the solution of the previous frame $\mathcal{X}_{old}$. Finally, $\mathcal{X}^*$ is selected as given below, where we slightly favor ($\lambda = 1.003$) the label proposal to facilitate fast pose recovery:

$$\mathcal{X}^* = \begin{cases} \mathcal{X}_1^* & \text{if } \left(E_{val}(\mathcal{X}_1^*) < \lambda E_{val}(\mathcal{X}_0^*)\right) \\ \mathcal{X}_0^* & \text{otherwise} \end{cases}. \tag{2}$$

The energy $E_{val}(\mathcal{X}) = E_a(\mathcal{X}) + w_p E_p(\mathcal{X})$ is designed to select the proposal that best explains the input, while being anatomically correct. Therefore, it considers spatial alignment to the input depth map E_a and models anatomical joint angle limits E_p, see Sect. 7. In the following, we describe the used energies in detail.

7 Hand-Object Tracking Objectives

Given the input depth distribution $\mathcal{M}_d$, we want to find the 3D model $\mathcal{M}_s$ that best explains the observations by varying the corresponding parameters $\mathcal{X}$. We take inspiration from methods with slow non-interactive runtimes that used related 3D implicit shape models for full-body pose tracking [12,22], but propose a new efficient tracking objective tailored for real-time hand-object tracking. In contrast to previous methods, our objective operates in 3D (generalization of ICP), features an improved way of incorporating the discriminative classification results, and incorporates two novel regularization terms. Together, this provides for a better, yet compact, representation that allows for fast analytic pose optimization on the CPU. To this end, we define the following two objective functions. The first energy E_{align} measures the alignment with the input:

$$E_{align}(\mathcal{X}) = E_a + w_p E_p + w_t E_t + w_c E_c + w_o E_o. \tag{3}$$

The second energy E_{label} incorporates the classification results:

$$E_{label}(\mathcal{X}) = E_a + w_s E_s + w_p E_p. \tag{4}$$

The energy terms consider spatial alignment E_a, semantic alignment E_s, anatomical plausibility E_p, temporal smoothness E_t, contact points E_c, and object-hand occlusions E_o, respectively. The priors in the energies are chosen such that they

do not hinder the respective alignment objectives. All parameters $w_p = 0.1$, $w_t = 0.1$, $w_s = 3 \cdot 10^{-7}$, $w_c = 5 \cdot 10^{-7}$ and $w_o = 1.0$ have been empirically determined and stay fixed for all experiments. We optimize both energies simultaneously using a multiple proposal based optimization strategy and employ a winner-takes-all strategy (see Sect. 6). We found empirically that using two energy functions resulted in better pose estimation and recovery from failures than using a single energy with all terms. In the following, we give more details on the individual components.

Spatial Alignment. We measure the alignment of the input density function $\mathcal{M}_d$ and our scene model $\mathcal{M}_s$ based on the following alignment energy:

$$E_a(\mathcal{X}) = \int_{\Omega} \left[\left(\mathcal{M}_{d_h}(\mathbf{x}) - \mathcal{M}_h(\mathbf{x}) \right)^2 + \left(\mathcal{M}_{d_o}(\mathbf{x}) - \mathcal{M}_o(\mathbf{x}) \right)^2 \right] d\mathbf{x}. \tag{5}$$

It measures the alignment between the two input and two model density distributions at every point in space $\mathbf{x} \in \Omega$. Note, this 3D formulation leads to higher accuracy results (see Sect. 8) than a 2.5D [27] formulation.

Semantic Alignment. In addition to the alignment of the distributions, we also incorporate semantic information in the label energy E_{label}. In contrast to [27], we incorporate uncertainty based on the best class probability. We use the following least-squares objective to enforce semantic alignment:

$$E_s(\mathcal{X}) = \sum_{i=1}^{N_s} \sum_{j=1}^{N_d} \alpha_{i,j} \cdot ||\boldsymbol{\mu}_i - \boldsymbol{\mu}_j||_2^2. \tag{6}$$

Here, $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$ are the mean of the i^{th} model and the j^{th} image Gaussian, respectively. The weights $\alpha_{i,j}$ switch attraction forces between similar parts on and between different parts off:

$$\alpha_{i,j} = \begin{cases} 0 & \text{if } (l_i \neq l_j) \text{ or } (d_{i,j} > r_{max}) \\ (1 - \frac{d_{i,j}}{r_{max}}) \cdot p_i & \text{else} \end{cases}. \tag{7}$$

Here, $d_{i,j} = ||\boldsymbol{\mu}_i - \boldsymbol{\mu}_j||_2$ is the distance between the means. l_i is the part label of the most likely class, p_i its probability and r_{max} a cutoff value. We set r_{max} to 30cm. l_i can be one of 8 labels: 6 for the hand parts, 1 for object and 1 for background. We consider all model Gaussians, independent of their occlusion weight, to facilitate fast pose recovery of previously occluded regions.

Anatomical Plausibility. The articulated motion of the hand is subject to anatomical constraints. We account for this by enforcing soft-constraints on the joint angles $\mathcal{X}_h$ of the hand:

$$E_p(\mathcal{X}) = \sum_{x_i \in \mathcal{X}_h} \begin{cases} 0 & \text{if } x_i^l \leq x_i \leq x_i^u \\ ||x_i - x_i^l||^2 & \text{if } x_i < x_i^l \\ ||x_i^u - x_i||^2 & \text{if } x_i > x_i^u \end{cases}. \tag{8}$$

Here, $\mathcal{X}_h$ are the DOFs corresponding to the hand, and x_i^l and x_i^u are the lower and upper joint limit that corresponds to the i^{th} DOF of the kinematic chain.

Temporal Smoothness. We further improve the smoothness of our tracking results by incorporating a temporal prior into the energy. To this end, we include a soft constraint on parameter change to enforce constant speed:

$$E_t(\mathcal{X}) = ||\nabla\mathcal{X} - \nabla\mathcal{X}^{(t-1)}||_2^2. \tag{9}$$

Here, $\nabla\mathcal{X}^{(t-1)}$ is the gradient of parameter change at the previous time step.

Contact Points. We propose a novel contact point objective, specific to the hand-object tracking scenario:

$$E_c(\mathcal{X}) = \sum_{(k,l,t_d)\in\mathcal{T}} \left(||\boldsymbol{\mu}_k - \boldsymbol{\mu}_l||^2 - t_d^2\right)^2. \tag{10}$$

Here, $(k, l, t_d) \in \mathcal{T}$ is a detected touch constraint. It encodes that the fingertip Gaussian with index k should have a distance of t_d to the object Gaussian with index l. We detect the set of all touch constraints $\mathcal{T}$ based on the last pose $\mathcal{X}_{old}$. A new touch constraint is added if a fingertip Gaussian is closer to an object Gaussian than the sum of their standard deviations. We then set t_d to this sum. This couples hand pose and object tracking leading to more stable results. A contact point is active until the distance between the two Gaussians exceeds the release threshold δ_R. Usually $\delta_R > t_d$ to avoid flickering.

Occlusion Handling. No measurements are available in occluded hand regions. We stabilize the hand movement in such regions using a novel occlusion prior:

$$E_o(\mathcal{X}) = \sum_{i=0}^{N_h} \sum_{j\in\mathcal{H}_i} (1 - \hat{f}_i) \cdot ||x_j - x_j^{old}||_2^2. \tag{11}$$

Here, $\mathcal{H}_i$ is the set of all DOFs that are influenced by the i-th Gaussian. The global rotation and translation is not included. The occlusion weights $\hat{f}_i \in [0, 1]$ are computed similar to f_i (0 occluded, 1 visible). This prior is based on the assumption that occuded regions move consistently with the rest of the hand.

8 Experiments and Results

We evaluate and compare our method on more than **15 sequences** spanning 3 public datasets, which have been recorded with 3 different RBG-D cameras (see Fig. 7). Additional live sequences (see Fig. 8 and supplementary materials) show that our method handles fast object and finger motion, difficult occlusions and fares well even if two hands are present in the scene. Our method supports commodity RGB-D sensors like the *Creative Senz3D*, *Intel RealSense F200*, and

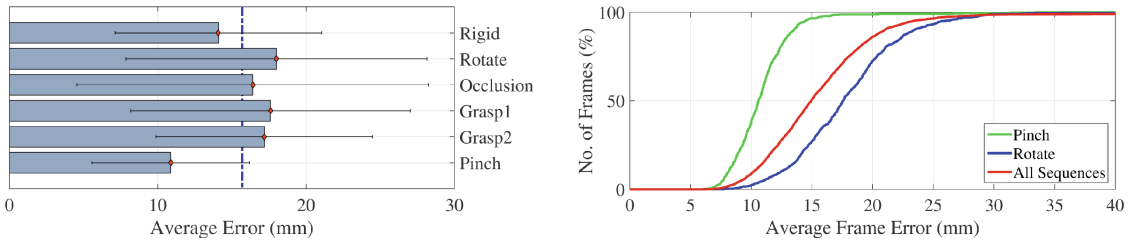
Primesense Carmine. We rescale depth and color to resolutions of 320×240 and 640×480 respectively, and capture at 30 Hz. Furthermore, we introduce a new hand-object tracking benchmark dataset with ground truth fingertip and object annotations.

Comparison to the State-of-the-Art. We quantitatively and qualitatively evaluate on two publicly available hand-object datasets [37,38] (see Fig. 8 and also supplementary material). Only one dataset (IJCV [37]) contains ground truth joint annotations. We test on 5 rigid object sequences from IJCV. We track the right hand only, but our method works even when multiple hands are present. Ground truth annotations are provided for 2D joint positions, but not object pose. Our method achieves a fingertip pixel error of **8.6 px**, which is comparable (difference of only 2 px) to that reported for the slower method of [37]. This small difference is well within the uncertainty of manual annotation and sensor noise. Note, our approach runs over 60 times faster, while producing visual results that are on par (see Fig. 8). We also track the dataset of [38] (see also Fig. 8). While they solve a different problem (offline in-hand scanning), it shows that our real-time method copes well with different shaped objects (e.g. bowling pin, bottle, etc.) under occlusion.

New Benchmark Dataset. With the aforementioned datasets, evaluation of object pose is impossible due to missing object annotations. We therefore introduce, to our knowledge, the first dataset[1] that contains ground truth for **both** fingertip positions and object pose. It contains 6 sequences of a hand manipulating a cuboid (2 different sizes) in different hand-object configurations and grasps. We manually annotated pixels on the depth image to mark 5 fingertip positions, and 3 cuboid corners. In total, we provide 3014 frames with ground truth annotations. As is common in the literature [23,25,27,32,33], we use the average 3D Euclidean distance E between estimated and ground truth positions as the error measure (see supplementary document for details). Occluded fingertips are excluded on a per-frame basis from the error computation. If one of the annotated corners of the cuboid is occluded, we exclude it from that frame. In Fig. 4a we plot the average error over all frames of the 6 sequences. Our method has an average error (for both hand and object) of **15.7 mm**. Over all sequences, the average error is always lower than 20 mm with standard deviations under 12 mm. Average error is an indicator of overall performance, but does not indicate how consistent the tracker performs. Figure 4b shows that our method tracks almost all frames with less than 30 mm error. *Rotate* has the highest error, while *Pinch* performs best with almost all frames below 20 mm. Table 1 shows the errors for hand and object separately. Both are in the same order of magnitude.

Ablative Analysis. Firstly, we show that the articulated 3D Gaussian mixture alignment formulation is superior (even for tracking only hand) to the 2.5D formulation of [27]. On the Dexter dataset [28], [27] report an average fingertip error of **19.6 mm**. In contrast, our method (**without** any hand-object specific

[1] http://handtracker.mpi-inf.mpg.de/projects/RealtimeHO/.



(a) We achieve low errors on each of the 6 sequences in our new benchmark dataset

(b) Tracking consistency of the best, worst and average case

Fig. 4. Quantitative hand-object tracking evaluation on ground truth data. The object contributes a higher error

Table 1. Average error (mm) for hand and object tracking in our dataset

	Rigid	*Rotate*	*Occlusion*	*Grasp1*	*Grasp2*	*Pinch*	Overall (mm)
Fingertips	14.2	16.3	17.5	18.1	17.5	10.3	**15.6**
Object	13.5	26.8	11.9	15.3	15.7	13.9	**16.2**
Combined (E)	14.1	18.0	16.4	17.6	17.2	10.9	**15.7**

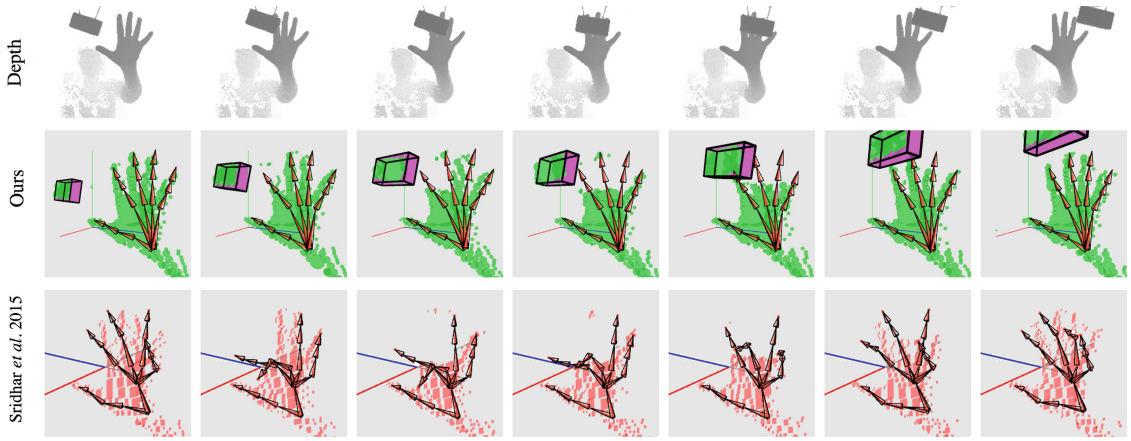


Fig. 5. *Top row:* Input depth, an object occludes the hand. *Middle row:* Result of our approach (different viewpoint). Our approach succesfully tracks the hand under heavy occlusion. *Bottom row:* Result of [27] shows catastrophic failure (object pixels were removed for fairness)

terms) is consistently better with an average of **17.2 mm** (maximum improvement is **5 mm** on 2 sequences). This is a result of the continuous articulated 3D Gaussian mixture alignment energy, a generalization of ICP, which considers 3D spatial proximity between Gaussians.

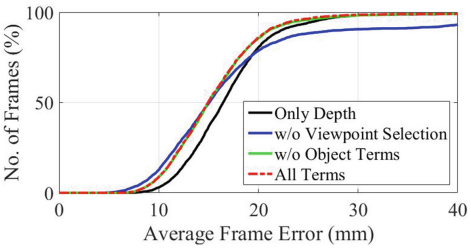


Fig. 6. Ablative analysis

Secondly, we show that the average error on our hand-object dataset is worse without viewpoint selection, semantic alignment, occlusion handling, and contact points term. Figure 6 shows a consistency plot with different components of the energy disabled. Using only the data term often results in large errors. The errors are even larger without viewpoint selection. The semantic alignment, occlusion handling, and contact points help improve **robustness** of tracking results and **recovery** from failures. Figure 5 shows that [27] clearly fails when fingers are occluded. Our hand-object specific terms are more robust to these difficult occlusion cases while achieving real-time performance.

Runtime Performance. All experiments were performed on an Intel Xeon E5-1620 CPU with 16 GB memory and an NVIDIA GTX 980 Ti. The stages of our approach take on average: 4 ms for preprocessing, 4 ms for part classification, 2 ms for depth clustering, and 20–30 ms for pose optimization using two proposals. We achieve real-time performance of 25–30 Hz. Multi-layer random forests ran on the GPU while all other algorithm parts ran multithreaded on a CPU.

Limitations. Although we demonstrated robustness against reasonable occlusions, situations where a high fraction of the hand is occluded for a long period are still challenging. This is mostly due to degraded classification performance under such occlusions. Misalignments can appear if the underlying assumption of the occlusion heuristic is violated, i. e. occluded parts do not move rigidly. Fortunately, our discriminative classification strategy enables the pose optimization to recover once previously occluded regions become visible again as shown in Fig. 9. Further research has to focus on better priors for occluded regions, for example grasp and interaction priors learned from data. Also improvements to hand part classification using different learning approaches or the regression of dense correspondences are interesting topics for future work. Another source of error are very fast motions. While the current implementation achieves 30 Hz, higher frame rate sensors in combination with a faster pose optimization will lead to higher robustness due to improved temporal coherence. We show diverse object shapes being tracked. However, increasing object complexity (shape and color) affects runtime performance. We would like to further explore how multiple complex objects and hands can be tracked.

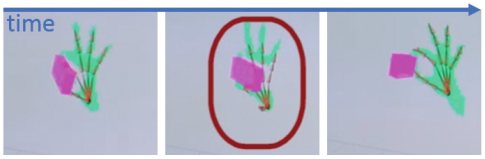
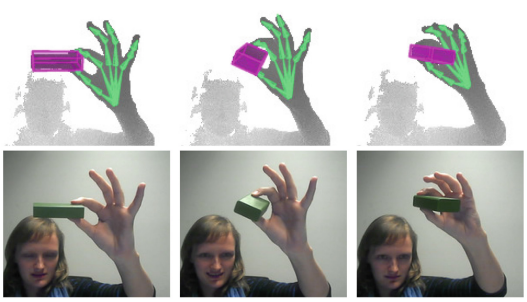
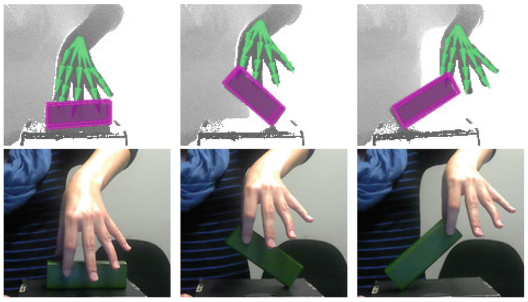


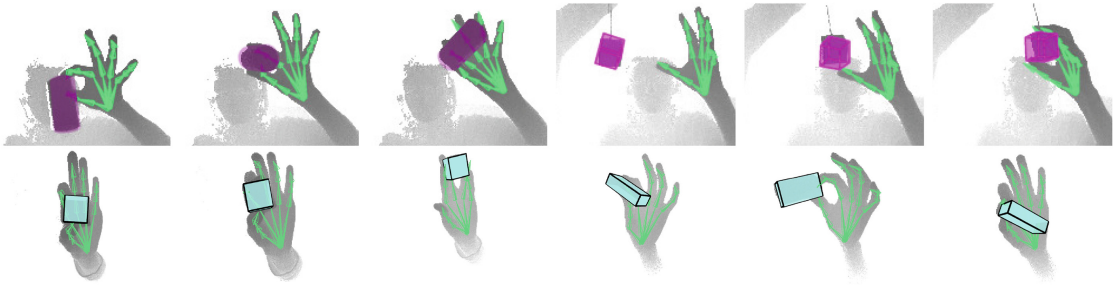
Fig. 9. Occlusion error and recovery



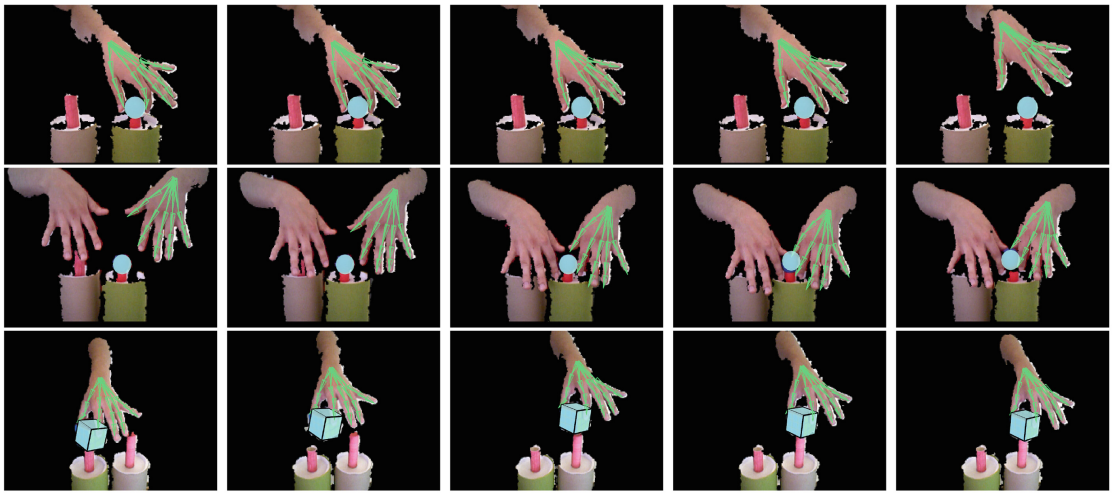
(a) *Rotate* sequence from our dataset



(b) *Grasp2* sequence from our dataset



(c) Real-time tracking results with various object shapes and different users



(d) Results on the IJCV dataset [37]. Notice how our method tracks the hand even if multiple hands are in view. Tracked skeleton in green and object in light blue

Fig. 7. (a, b) show tracking results on our dataset. (c) Shows real-time results with different object shapes and colors. (d) Shows results on a public dataset (Color figure online)

Fig. 8. Subset of tracked frames on the dataset of [38]. Our method can handle objects with **varying sizes, colors, and different hand dimensions**. Here we show how even a complex shape like a bowling pin can be approximated using only a few tens of Gaussians (Color figure online)

9 Conclusion

We have presented the first real-time approach for simultaneous hand-object tracking based on a single commodity depth sensor. Our approach combines the strengths of discriminative classification and generative pose optimization. Classification is based on a multi-layer forest architecture with viewpoint selection. We use 3D articulated Gaussian mixture alignment tailored for hand-object tracking along with novel analytic occlusion and contact handling constraints that enable successful tracking of challenging hand-object interactions based on multiple proposals. Our qualitative and quantitative results demonstrate that our approach is both accurate and robust. Additionally, we have captured a new benchmark dataset (with hand and object annotations) and make it publicly available. We believe that future research will significantly benefit from this.

Acknowledgments. This research was funded by the ERC Starting Grant projects CapReal (335545) and COMPUTED (637991), and the Academy of Finland. We would like to thank Christian Richardt.

References

1. Leap Motion. https://www.leapmotion.com/
2. NimbleVR. http://nimblevr.com/
3. Athitsos, V., Sclaroff, S.: Estimating 3D hand pose from a cluttered image. In: Proceedings of IEEE CVPR, pp. 432–442 (2003)
4. Badami, I., Stckler, J., Behnke, S.: Depth-enhanced hough forests for object-class detection and continuous pose estimation. In: Workshop on Semantic Perception, Mapping and Exploration (SPME) (2013)
5. Ballan, L., Taneja, A., Gall, J., Gool, L., Pollefeys, M.: Motion capture of hands in action using discriminative salient points. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 640–653. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33783-3_46
6. Bray, M., Koller-Meier, E., Van Gool, L.: Smart particle filtering for 3D hand tracking. In: Proceedings of the International Conference on Automatic Face and Gesture Recognition, pp. 675–680 (2004)
7. Campbell, D., Petersson, L.: Gogma: globally-optimal Gaussian mixture alignment (2016). arXiv preprint arXiv:1603.00150
8. Hamer, H., Schindler, K., Koller-Meier, E., Van Gool, L.: Tracking a hand manipulating an object. In: Proceedings of IEEE ICCV, pp. 1475–1482 (2009)
9. Heap, T., Hogg, D.: Towards 3D hand tracking using a deformable model. In: Proceedings of the International Conference on Automatic Face and Gesture Recognition, pp. 140–145, October 1996
10. Jian, B., Vemuri, B.C.: Robust point set registration using Gaussian mixture models. IEEE Trans. Pattern Anal. Mach. Intell. **33**(8), 1633–1645 (2011)
11. Keskin, C., Kira, F., Kara, Y.E., Akarun, L.: Real time hand pose estimation using depth sensors. In: ICCV Workshops, pp. 1228–1234. IEEE (2011). http://dblp.uni-trier.de/db/conf/iccvw/iccvw2011.html#KeskinKKA11

12. Kurmankhojayev, D., Hasler, N., Theobalt, C.: Monocular pose capture with a depth camera using a sums-of-Gaussians body model. In: Weickert, J., Hein, M., Schiele, B. (eds.) GCPR 2013. LNCS, vol. 8142, pp. 415–424. Springer, Heidelberg (2013)
13. Kyriazis, N., Argyros, A.: Physically plausible 3D scene tracking: the single actor hypothesis. In: Proceedings of IEEE CVPR, pp. 9–16 (2013)
14. Kyriazis, N., Argyros, A.: Scalable 3D tracking of multiple interacting objects. In: Proceedings of IEEE CVPR, pp. 3430–3437, June 2014
15. de La Gorce, M., Fleet, D., Paragios, N.: Model-based 3D hand pose estimation from monocular video. IEEE TPAMI **33**(9), 1793–1805 (2011)
16. Melax, S., Keselman, L., Orsten, S.: Dynamics based 3D skeletal hand tracking. In: Proceedings of GI, pp. 63–70 (2013)
17. Oikonomidis, I., Kyriazis, N., Argyros, A.: Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In: Proceedings of IEEE ICCV, pp. 2088–2095 (2011)
18. Oikonomidis, I., Kyriazis, N., Argyros, A.: Tracking the articulated motion of two strongly interacting hands. In: Proceedings of IEEE CVPR, pp. 1862–1869 (2012)
19. Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Efficient model-based 3D tracking of hand articulations using kinect. In: Proceedings of BMVC, pp. 1–11 (2011)
20. Panteleris, P., Kyriazis, N., Argyros, A.A.: 3D tracking of human hands in interaction with unknown objects. In: Proceedings of BMVC (2015). https://dx.doi.org/10.5244/C.29.123
21. Pham, T.H., Kheddar, A., Qammaz, A., Argyros, A.A.: Towards force sensing from vision: observing hand-object interactions to infer manipulation forces. In: Proceedings of IEEE CVPR (2015)
22. Plankers, R., Fua, P.: Articulated soft objects for multiview shape and motion capture. IEEE TPAMI **25**(9), 1182–1187 (2003). http://dx.doi.org/10.1109/TPAMI.2003.1227995
23. Qian, C., Sun, X., Wei, Y., Tang, X., Sun, J.: Realtime and robust hand tracking from depth. In: Proceedings of IEEE CVPR (2014)
24. Romero, J., Kjellstrom, H., Kragic, D.: Hands in action: real-time 3D reconstruction of hands in interaction with objects. In: Proceedings of ICRA, pp. 458–463 (2010)
25. Sharp, T., Keskin, C., Robertson, D., Taylor, J., Shotton, J., Kim, D., Rhemann, C., Leichter, I., Vinnikov, A., Wei, Y., Freedman, D., Kohli, P., Krupka, E., Fitzgibbon, A., Izadi, S.: Accurate, robust, and flexible real-time hand tracking. In: Proceedings of ACM CHI (2015)
26. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: Proceedings of IEEE CVPR, pp. 1297–1304 (2011). http://dx.doi.org/10.1109/CVPR.2011.5995316
27. Sridhar, S., Mueller, F., Oulasvirta, A., Theobalt, C.: Fast and robust hand tracking using detection-guided optimization. In: Proceedings IEEE CVPR (2015). http://handtracker.mpi-inf.mpg.de/projects/FastHandTracker/
28. Sridhar, S., Oulasvirta, A., Theobalt, C.: Interactive markerless articulated hand motion tracking using RGB and depth data. In: Proceedings of IEEE ICCV (2013)
29. Stenger, B., Mendonça, P.R., Cipolla, R.: Model-based 3D tracking of an articulated hand. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 2, pp. II-310. IEEE (2001)

30. Stoll, C., Hasler, N., Gall, J., Seidel, H., Theobalt, C.: Fast articulated motion tracking using a sums of Gaussians body model. In: Proceedings of IEEE ICCV, pp. 951–958 (2011)
31. Sun, X., Wei, Y., Liang, S., Tang, X., Sun, J.: Cascaded hand pose regression. In: Proceedings of IEEE CVPR (2015)
32. Tagliasacchi, A., Schröder, M., Tkach, A., Bouaziz, S., Botsch, M., Pauly, M.: Robust articulated-ICP for real-time hand tracking. In: Computer Graphics Forum (Proceedings of SGP), vol. 34, no. 5 (2015)
33. Tang, D., Chang, H.J., Tejani, A., Kim, T.: Latent regression forest: structured estimation of 3D articulated hand posture. In: Proceedings of IEEE CVPR, pp. 3786–3793 (2014). http://dx.doi.org/10.1109/CVPR.2014.490
34. Tang, D., Taylor, J., Kim, T.K.: Opening the black box: hierarchical sampling optimization for estimating human hand pose. In: Proceedings of IEEE ICCV (2015)
35. Tejani, A., Tang, D., Kouskouridas, R., Kim, T.-K.: Latent-class hough forests for 3D object detection and pose estimation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 462–477. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10599-4_30
36. Tompson, J., Stein, M., Lecun, Y., Perlin, K.: Real-time continuous pose recovery of human hands using convolutional networks. ACM TOG **33**(5), 169:1–169:10 (2014)
37. Tzionas, D., Ballan, L., Srikantha, A., Aponte, P., Pollefeys, M., Gall, J.: Capturing hands in action using discriminative salient points and physics simulation. IJCV **118**, 172–193 (2016)
38. Tzionas, D., Gall, J.: 3D object reconstruction from hand-object interactions. In: Proceedings of IEEE ICCV (2015)
39. Tzionas, D., Srikantha, A., Aponte, P., Gall, J.: Capturing hand motion with an RGB-D sensor, fusing a generative model with salient points. In: Jiang, X., Hornegger, J., Koch, R. (eds.) GCPR 2014. LNCS, vol. 8753, pp. 277–289. Springer, Heidelberg (2014). doi:10.1007/978-3-319-11752-2_22
40. Wang, R., Paris, S., Popović, J.: 6D hands: markerless hand-tracking for computer aided design. In: Proceedings of ACM UIST, pp. 549–558 (2011)
41. Wang, Y., Min, J., Zhang, J., Liu, Y., Xu, F., Dai, Q., Chai, J.: Video-based hand manipulation capture through composite motion control. ACM TOG **32**(4), 43:1–43:14 (2013)
42. Wu, Y., Huang, T.: View-independent recognition of hand postures. In: Proceedings of IEEE CVPR, pp. 88–94 (2000)
43. Xu, C., Cheng, L.: Efficient hand pose estimation from a single depth image. In: Proceedings of IEEE ICCV (2013)
44. Ye, M., Yang, R.: Real-time simultaneous pose and shape estimation for articulated objects using a single depth camera. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2353–2360, June 2014
45. Zollhöfer, M., Nießner, M., Izadi, S., Rehmann, C., Zach, C., Fisher, M., Wu, C., Fitzgibbon, A., Loop, C., Theobalt, C., Stamminger, M.: Real-time non-rigid reconstruction using an RGB-D camera. ACM TOG **33**(4), 156 (2014)