Robust, Occlusion-aware Pose Estimation for Objects Grasped by Adaptive Hands

Bowen Wen, Chaitanya Mitash, Sruthi Soorian, Andrew Kimmel, Avishai Sintov and Kostas E. Bekris

Abstract— **Many manipulation tasks, such as placement or within-hand manipulation, require the object's pose relative to a robot hand. The task is difficult when the hand significantly occludes the object. It is especially hard for adaptive hands, for which it is not easy to detect the finger's configuration. In addition, RGB-only approaches face issues with texture-less objects or when the hand and the object look similar. This paper presents a depth-based framework, which aims for robust pose estimation and short response times. The approach detects the adaptive hand's state via efficient parallel search given the highest overlap between the hand's model and the point cloud. The hand's point cloud is pruned and robust global registration is performed to generate object pose hypotheses, which are clustered. False hypotheses are pruned via physical reasoning. The remaining poses' quality is evaluated given agreement with observed data. Extensive evaluation on synthetic and real data demonstrates the accuracy and computational efficiency of the framework when applied on challenging, highly-occluded scenarios for different object types. An ablation study identifies how the framework's components help in performance. This work also provides a dataset for in-hand 6D object pose estimation. Code and dataset are available at:** `https://github.com/wenbowen123/icra20-hand-object-pose`

I. INTRODUCTION

Robot manipulation often requires recognizing objects and detecting their 6D pose, i.e., position and orientation. Applications include logistics [1], where picking is a frequent task. Once picked, an object may need to be purposefully placed for packaging, sorting or restocking. Depending on the task, regrasping or within-hand manipulation may also be required. These objectives need the object's 6D pose relative to the robot's hand post-grasp. Most existing work in pose estimation is focusing on the pre-grasp case [2], [3], [4], [5], [6], which is not always a good indicator of the post-grasp one due to the effects of contact. This is especially true for adaptive hands, such as underactuated, compliant systems that naturally and safely adapt to an object's shape as in Fig. 1. There are multiple challenges that arise in this context:

- Severe occlusions: The hand often significantly occludes the grasped object. Thus, solutions need to robustly distinguish the target object from the robot's fingers and noisy scene. Small objects further complicate the process as they are mostly covered by the hand from the camera's viewpoint.

- Unpredictable contacts and dynamic tasks: Pre-grasp pose estimation does not suffer as much from occlusions. Recent work for in-hand pose estimation [7] assumes the pose does not change significantly upon grasping and can initialize ICP (Iterative Closest Point). But as the hand grasps the object, the pose changes dynamically. This is also true if regrasping or within-hand manipulation is performed, where it is difficult to account for contacts, especially for compliant and adaptive hands. 6D pose tracking [8], [9], [10], [11] can help but also requires a good initial estimate. If the tracking loses the object, robust pose estimation given a highly-occluded snapshot is still needed.

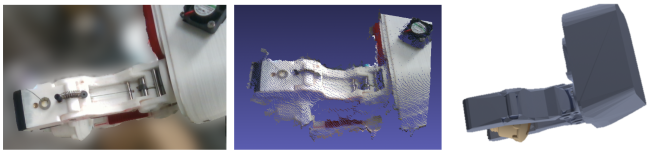

Fig. 1. Left: Original image showing the adaptive hand grasping and severely occluding a texture-less object. Middle: Point-cloud data. Right: Scene reconstruction given the output of the approach.

- Robustness and Generalizability Pose estimation based on color or texture data [7], [12], [13], [14], [15] can be sensitive to lighting conditions, and challenging for texture-less objects or when the object and the robot hand look similar. Extracting local 3D descriptors and finding correspondences [16], [17], [18] may suffer from limited object visibility.

This paper presents a framework for robust, within-hand 6D object pose estimation using a consumer-level depth sensor. It addresses the issues arising from adaptive hands and focuses on the Yale Hand T42 [19], given its use for dexterous manipulation [20], [21]. A key feature is the estimation of the hand's state to help infer the object's region on the image. The method builds a hand-SDF (Signed Distance Field) to regularize the object's pose given physical constraints. This makes the task computationally manageable even under severe occlusions. The proposed framework exhibits these properties:

- High precision; it achieves high accuracy, even for a tight error threshold of 5mm under the `ADI` metric [22].
- Computational efficiency; as it returns the pose of the object and the state of the adaptive hand in 0.5 to 0.7 seconds. The hand's state estimate may also be helpful for control purposes;
- Robustness; as the method works for various objects, including textureless ones, and with a cluttered background, where RGB-based methods would struggle.

This work also contributes a synthetic and a real dataset, where an adaptive hand holds various objects, with RGB-D data and ground truth information, since no related dataset exist in the literature beyond for objects contained in human hands [8]. Experiments on both datasets demonstrate the effectiveness, robustness and efficiency of the proposed system for multiple objects in various scenarios, compared against state-of-the-art methods. An ablation study highlights how the method's critical components help in performance.

The authors are with the Computer Science Dept. of Rutgers Univ. in NJ, USA. This work is supported by NSF awards IIS-1734492, IIS-1723869, CCF-1934924. The opinions & findings in this paper do not necessarily reflect the sponsor's views. Email: {bw344,kostas.bekris}@cs.rutgers.edu.

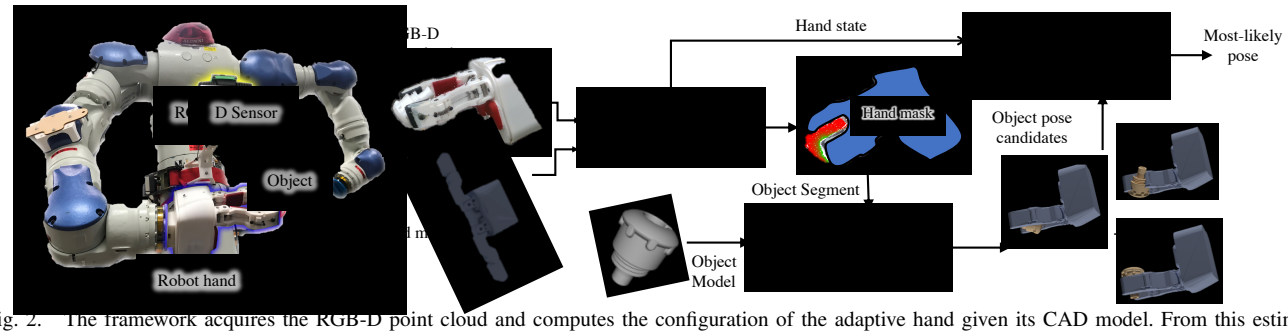

Fig. 2. The framework acquires the RGB-D point cloud and computes the configuration of the adaptive hand given its CAD model. From this estimate, the hand is removed from the point cloud and the object segment is recovered. A set of pose candidates is generated by matching the segment to the object's model. The most likely pose is returned by evaluating the consistency of the interactions between the estimated hand and the in-hand object.

II. RELATED WORK

This section covers different approaches for object pose estimation related to manipulation tasks.

Alternatives to Vision: Various sensors have been used for in-hand pose estimation, such as proprioception [23], [24], and contact/force sensing [25], [26], [27]. Such sensors have also been combined with vision to decrease uncertainty [28], [29], [30], [31], [32], [33], [34], [35]. Nevertheless, these sensing modalities are not always accessible, as they require careful engineering of the hands and increase cost. Under-actuated adaptive hands, for instance, do not often provide information for identifying finger configurations. Thus, a vision-only solution is desirable.

Single Image Object Pose Estimation: Recent advances in object detection [36], [37] and pose estimation [13], [38] have shown promise given access to sufficient labeled data. This allows to project an object's 3D bounding box on the image and solve for a pose using PnP [39], [14]. This is problematic, however, under severe occlusions. Alternatively, direct 6D pose regression has been attempted [13], [40]. Nevertheless, the complexity of SO(3) results in instability in training and prediction. Recent work [41], [42] attempts to jointly estimate a human hand and the in-hand 6D object pose accounting for physical consistency but the resulting precision is not sufficient for manipulation. In contrast, a robotic hand's kinematic information is available, which helps increase precision.

3D Registration Methods: Registration [16] often uses local geometry features followed by voting, which makes them sensitive to point cloud density that is problematic under severe occlusions. Alignment solutions can use gradient descent optimization [43] but again degrade under severe occlusions, when only few features and correspondences can be extracted on the small point cloud segment of the object. Super4PCS has been shown effective in global registration, whereas its RANSAC nature makes it inefficient when large number of outliers exist. This work builds upon prior efforts [43] and achieves higher accuracy with faster speed by introducing heuristics-guided sampling.

Object Pose Tracking: Methods have used a variety of approaches: GPU-accelerated particle filtering with a likelihood estimation based on color, distance and normals [10]; modeling occlusions to eliminate outliers [44]; Gaussian Filtering to track objects using depth [8]. Promising precision is achieved for small errors but tracking loss arises frequently. Recent work [45] formulates the 6D object pose tracking problem in the Rao-Blackwellized particle filtering framework. This method, however, requires a reliable single image pose for re-initialization upon tracking loss. The current work differs from the above in that it achieves fast, high precision estimates from individual high-occlusion snapshots without knowledge about previous frames. It can be integrated with such tracking frameworks to (re-)initialize.

Visual Servoing: A simple solution is to attach fiducial markers [46], [32] on the object [47], [48], [20] but it is not always practical to keep the marker visible, especially during in-hand manipulation. Additionally, complex surfaces make the attachment troublesome. Recent work trained an end-to-end policy network to perform within-hand manipulation while reasoning about object pose [12]. Computational resources, however, prevent it from easy application across conditions, such as objects unseen during training or having less distinctive features. Another effort estimated object pose by first segmenting the robot hand given a Naive Bayes classifier and then performing ICP (Iterative Closest Point) for the object segment, assuming the object does not move much upon grasping [7]. This assumption is often violated when grasping or in-hand manipulation leads to object slippage. The current work does not depend on a pre-grasp estimate.

III. PROBLEM FORMULATION

Given a depth image from camera C, a mesh model M of object O, the goal is to compute O's 6D pose, i.e., the rigid transform T_M^C, where O is grasped by an adaptive hand in C's view. The work considers under-actuated hands (the Yale Hand T42 [19]) for which a CAD model is available. The hand state determined by configuration of the N fingers $x_H = \{q_{F_i}\}_{i=1}^N$ are initially unknown and not available. The camera is calibrated and the transform T_H^C of the hand's wrist frame H to the camera is available.

IV. APPROACH

Fig. 2 outlines the proposed approach with 3 key components: 1) parallel evolutionary optimization to estimate the hand's configuration; 2) heuristics-guided global pointset registration to generate pose hypotheses for the object; 3) scene-level physics reasoning that considers the hand-object interaction to find the most-likely object pose.

A. Hand State Estimation

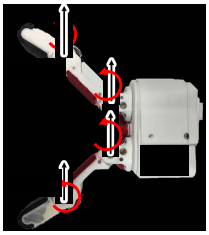

Fig. 3. Adaptive hand with 2 underactuated fingers.

An adaptive hand consists of a wrist and a set of fingers. The fingers are not sensorized to a level that provides reliable state information. Each finger F is treated as an articulated chain and its configuration is the set of all joint angles, i.e., $q_F = \{\theta_{F1}, \theta_{F2}, ..., \theta_{Fn}\}$ (see Fig. 3). A 3D region-of-interest (ROI) is identified that contains the point cloud P_S of the in-hand object and fingers. The ROI is computed based on the wrist's pose T_C^H obtained from forward kinematics and the hand dimensions. `ICP`, performed over the point cloud and the wrist's model, refines T_C^H to compensate for errors in forward kinematics and camera calibration.

The next step aims to find the finger configuration, which minimizes the discrepancy between the robot hand model and the observed depth image given P_S. It is possible to formalize this problem as convex objective optimization and employ gradient descent algorithms to obtain the optimal pose, as in related work [49]. An initial estimation from the previous frame, in the context of a tracking scenario, can be good initialization for the gradient descent to converge. Nevertheless, in single image estimation, such as in this work, no such initial guess is assumed. For this reason, this paper proposes Particle Swarm Optimization (`PSO`) for searching each finger configuration, inspired by prior work on human hand pose tracking [50]. `PSO` is an evolutionary process where particles interact with each other to search the parameter space. In addition to being less sensitive to local optima, it is highly parallelizable and does not require the objective function to be differentiable. This allows to formalize the cost function as minimizing the negative `LCP` (Largest Common Pointset) [51] score computed via an efficient KDTree implementation.

Unlike human hands, the configuration space of robot hands is more constrained. It was empirically observed that instead of estimating the hand state globally in `PSO`, sequentially estimating each finger's configuration leads to more stable solutions and faster convergence (with 15 particles and 3 iterations for each finger). Therefore, `PSO` was applied to each finger separately to estimate its configuration starting from the finger closest to camera. Each `PSO` particle is a vector representing the current finger configuration q_F and the swarm is a collection of particles. Initially, particles are randomly sampled and their velocities are initialized to zero. In each generation, a particles velocity is updated as a randomly weighted summation of its previous velocity, the velocity towards its own best known position, and the velocity towards the best known position of the entire swarm.

The cost function evaluation is given in Alg. 1. The inputs are the finger configuration q_F, which will be evaluated, the hand region point cloud P_S and finger model point cloud P_F. In lines 2 - 5, a penalty is assigned to cases when fingers have collisions. It returns a score that is linearly dependent on the penetration depth d to encourage particles to move to a more promising parameter space that satisfies collision avoidance. The λ_c parameter is a penalization term and is arbitrarily assigned to a very large value. P_S is first transformed into the finger frame using forward kinematics and q_F. A KDTree is built on the transformed P_S to compute the `LCP` score with the finger model cloud efficiently.

Algorithm 1: COST_FUNCTION (q_F, P_S, P_F)

1 $P_S^{finger} \leftarrow$ transform P_S to finger frame using forward kinematics and q_F;
2 **for** *any other finger* Q_F **do**
  `/* collision penetration depth (negative) */`
3 $d \leftarrow collisionCheck(P_F, Q_F)$;
4 **if** $d < \varepsilon$ **then**
5 return $-\lambda_c - \lambda_c d$;
6 $kdtree(P_S) \leftarrow$ build kdtree from P_S^{finger};
7 $LCP \leftarrow 0$;
8 **for** *each* $p_F \in P_F$ **do**
9 $p_{nei} \leftarrow kdtree(P_S).findNearestNeighbor(p_F)$;
10 **if** $||p_{nei} - p_F|| < \varepsilon$ *and* $normal(p_{nei}) \cdot normal(p_F) > \delta$ **then**
11 $LCP \leftarrow LCP + 1$;
12 return $-LCP$;

The single shot hand state estimation is implemented for parallel execution in C++. This component can also be very useful for tracking approaches [49], [52] as initialization or re-initialization.

B. Object Pose Hypotheses Generation and Clustering

Once the full hand state x_H is available, *SDF* (Signed Distance Field) is computed for the hand. All P_S points with signed distance below a threshold $SDF(p, x_H) < \varepsilon$ are eliminated ($\varepsilon = 3$ *mm* in the accompanying experiments). The remaining point cloud P_O is now assigned to the object. The new goal is to register the object mesh M_O against the point cloud P_O, despite the imperfections of P_O due to sensor noise, occlusions or errors in the hand state estimate.

This paper builds upon prior work for hypotheses generation [53], [54]. It samples sets of 4-point, co-planar bases on the object's point cloud (P_O), and searches for congruent sets on the object model (M_O) to provide a pool of rigid alignments (Fig. 4). Bases can be sampled randomly [53] or given the stochastic output of a `CNN` [54]. To limit the number of samples, while maximizing the chances of sampling a valid base (where all points belong to the object), this work proposes sampling heuristics given the hand state.

The base sampling process is given in Alg. 2, where inputs are the object point cloud P_O, heuristics π and a hash map PPF_M of Point Pair Features (PPF) [16] of the model M_O. The hash map PPF_M is precomputed. It counts the number of times a discretized PPF feature appears on M. The PPF for any two points on M_O is given by:

$$PPF(\mathbf{p}_1, \mathbf{p}_2) = (\|\mathbf{p_1 p_2}\|_2, \angle(\mathbf{n}_1, \mathbf{d}), \angle(\mathbf{n}_2, \mathbf{d}), \angle(\mathbf{n}_1, \mathbf{n}_2))$$

where n_1 and n_2 are point normals and d is the distance between the points. This avoids outliers from P_O. For sampling one base, 4 points are sampled incrementally by using a

heuristic score associated with every point on the point cloud P_O. The heuristic score follows an exponential distribution of the *Euclidean Distance Transform* of each point, which is computed from the hand's signed distance field *SDF*:

$$\pi(p_i) \propto 1 - exp(-\lambda SDF(p_i; x_H)).$$

where $\pi(p_i)$ returns a point's probability to be sampled. The probability distribution of all the points on the object cloud P_O are normalized and denoted as π. Points further away from the hand are more likely to belong to the object and are prioritized. To balance exploitation and exploration, a discounting factor $\gamma = 0.5$ decays the heuristic when a point is sampled. The discounting generates more dispersed and promising pose hypotheses.

Algorithm 2: SAMPLE_ONE_BASE (P_O, π, PPF_M)

1 $b_1 \leftarrow$ sample a point from P_O according to π ;
2 $B \leftarrow \{b_1\}$;
3 **for** $p \in P_O$ **do**
4 $\quad f_1 \leftarrow PPF(p, b_1)$;
5 $\quad$ **if** $PPF_M[f_1] == \emptyset$ **then**
6 $\quad\quad \pi(p) \leftarrow 0$;
7 **for** $i \leftarrow 0$ **to** *max_iter* **do**
8 $\quad b_2, b_3 \leftarrow$ sample two different points from P_O according to the updated distribution π ;
9 $\quad \pi(b_2) \leftarrow \gamma\, \pi(b_2)$;
10 $\quad \pi(b_3) \leftarrow \gamma\, \pi(b_3)$;
11 $\quad f_{23} \leftarrow PPF(b_2, b_3)$;
12 $\quad$ **if** $PPF_M[f_{23}] \neq \emptyset$ *and* $\angle(\overrightarrow{b_1 b_2}, \overrightarrow{b_1 b_3}) > \delta$ **then**
13 $\quad\quad B \leftarrow B \cup \{b_2, b_3\}$;
14 $\quad\quad$ break ;
15 **for** $i \leftarrow 0$ **to** *max_iter* **do**
16 $\quad b_4 \leftarrow$ sample a point from P_O according to the updated distribution π ;
17 $\quad \pi(b_4) \leftarrow \gamma\, \pi(b_4)$;
18 $\quad$ **if** $distance(plane(b_1, b_2, b_3), b_4) > \varepsilon$ **then**
19 $\quad\quad$ continue ;
20 $\quad f_{24} \leftarrow PPF(b_2, b_4)$, $f_{34} \leftarrow PPF(b_3, b_4)$;
21 $\quad$ **if** $PPF_M[f_{24}] \neq \emptyset$ *and* $PPF_M[f_{34}] \neq \emptyset$ **then**
22 $\quad\quad B \leftarrow B \cup b_4$;
23 $\quad\quad$ break ;
24 return B;

The sampling ensures that the 4 points are co-planar given a small threshold (Line 18). Base sampling is repeated until a desired number of bases is achieved. Given a base B, its congruent set on the object model is retrieved by hypersphere rasterization [53]. Alignment between the matching bases can be solved in a least square manner [53]. This returns a set of object pose hypotheses along with their LCP score. Base sampling and alignment are executed in parallel.

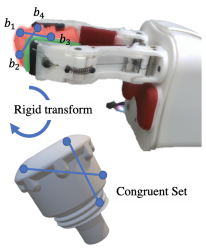

Fig. 4. A 4-point base is heuristically sampled. For a congruent set on the model a candidate transform is defined.

The large number of pose candidates generated often contains many incorrect or redundant poses. Clustering in SE(3) is performed to group together similar poses and reduce the size of the hypotheses set. Similar to prior work [55], a fast and effective technique is adapted for this step: a round of coarse grouping is performed in R^3 via *Euclidean Distance Clustering*. Then, each group is split by clustering according to the minimal geodesic distance along $SO(3)$:

$$d(R_1, R_2) = arccos(\frac{trace(R_1^T R_2) - 1}{2}).$$

Different from prior work [55], however, rather than using K-means, which can be computationally expensive, the new hypotheses are formed by the poses with the highest LCP score per cluster and refined by *Point-to-Plane* ICP [56]. After ICP, some candidates may converge to the same pose and are merged. The top k hypotheses (empirically set to 100) with the highest LCP score are kept to improve computational efficiency.

C. Pose Hypothesis Pruning and Selection

Physical reasoning is leveraged to further prune false hypotheses via collision checking and scene-level occlusion reasoning. Physical consistency is imposed by checking if the object model collides beyond certain depth with the estimated hand state, or if the object is located above certain distance from the hand mesh surface, indicating that the hand is not touching the object. This process can be performed efficiently by utilizing the hand state and its *SDF*.

Ambiguities might still arise due to several pose candidates achieving similar LCP score with the object under high occlusions. Any (non-corrupted) observation of a non-zero depth indicates that there is nothing between the observed point and the camera, up to some noise threshold and barring sensor error [57]. This scene-level reasoning is adapted by comparing the accumulated pixel-wise discrepancy between the observed depth image and the rendered one (computed via *OpenGL* using both the estimated object pose and hand state). Based on this rendering score, the top $1/3rd$ of pose hypotheses are retained. The final optimal pose is selected from this set according to the highest LCP.

V. EXPERIMENTS

This section evaluates the proposed approach and compares against state-of-the-art single-image pose estimation methods on in-hand objects. Note the difference with tracking methods [28], [49], [52], [8], since here the 6D object pose is recovered from a single static image without dependency on previous frames. To the best of the authors' knowledge, there are no relevant datasets in the literature beyond those for objects in human hands [8]. A benchmark dataset is developed that includes both simulated and real world data for in-hand object pose estimation with adaptive hands and will be released publicly.

A. Experimental Setup

The setup consists of a robot manipulator (Yaskawa Motoman) and a Yale T42 adaptive hand (Fig. 3), which was 3D printed based on open-source designs. Objects considered for in-hand manipulation were picked to evaluate the robustness

Method	Avg. Recall (%)
Super4PCS [53] + *HS*	71.58
Super4PCS [53] + *HS* + *ICP*	78.83
DOPE [14]	31.88
DOPE [14]+ *ICP*	35.58
DOPE [14] + *HS* + *ICP*	55.40
AAE* [60]	57.77
AAE* [60] + *ICP*	69.85
AAE* [60] + *HS* + *ICP*	78.06
OURS	**95.33**

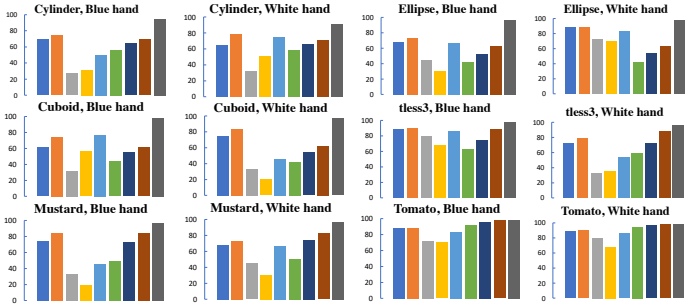

Fig. 5. Comparison on simulation dataset. For the table, *+HS* implies using the proposed `PSO` hand pose estimation to remove the hand related cloud from the scene, *+ICP* implies applying Point-to-Plane ICP for pose refinement.

of estimation. As shown in Fig. 6, the selected set is a mix of objects: with and without texture or geometric features.

Fig. 6. Mesh of objects used: A cylinder with diameter 0.035 *m* and length 0.064 *m*, an ellipsoid with length 0.064 *m*, a cuboid with side length 0.03 *m* and length 0.064 *m*, an industrial object #3 from T-LESS dataset [58], a mustard bottle and tomato soup can from YCB dataset [59]. Right 2 images: Yale T42 adaptive hands painted in blue and white.

All experiments are conducted on a standard desktop with Intel Xeon(R) E5-1660 v3@3.00GHz processor. For the comparison to deep learning methods, neural network inference is performed on a NVIDIA Tesla K40c GPU.

B. Evaluation Metric

The recall for pose estimation is measured based on the error given by the `ADI` metric [22], which measures the average of point distances between poses T_1 and T_2 given an object mesh model M:

$$e_{ADI}(T_1,T_2) = \text{avg}_{p_1 \in M} \min_{p_2 \in M} ||T_1(p_1,M) - T_2(p_2,M)||_2,$$

where $T(p,M)$ corresponds to point p after applying transformation T on M. Given a ground-truth pose T^g, a true positive is a returned pose T that has $e_{ADI}(T,T^g) < \varepsilon$, where ε is a tolerance threshold. ε is set to 5 *mm* in all experiments except in recall curves, to evaluate the applicability of different methods for precise in-hand manipulation scenarios.

C. Simulation Dataset and Results

Simulated RGB-D data were generated by placing a virtual camera at random poses around the model of the hand. Poses are sampled from 648 view points on spheres of radius 0.3 to 0.9 *m* centered at the hand. To generate each data point, an object is placed at a random pose between the fingers. The two articulated-fingers are closed randomly until they touch the object, verified by a collision checker. Physical parameters, such as friction, gravity or any grasping stability metric are deliberately not employed since this work aims at any-time single image 6D object pose estimation during the entire within-hand manipulation process, in which a stable grasp is not always a true assumption. By randomizing the object pose relative to the hand, the dataset is able to cover various in-hand object poses that can occur during an in-hand manipulation process. For the adaptive hand, two colors are chosen. The blue hand differs from any object color used in the experiments whereas the white hand resembles textureless objects and evaluates robustness to lack of texture. In addition to the RGB-D data, ground-truth object pose and semantic segmentation images are also obtained from the simulator. For each combination of the 6 objects and the 2 adaptive hands, 1000 data points are generated, resulting in 12000 test cases.

Fig. 5 reports the recall for pose estimation on the synthetic dataset. When `Super4PCS` is directly applied to the entire point cloud, outlier points that do not belong to the object are often sampled, leading to poor results (5.83 %). On introducing the proposed `PSO` hand state estimation (HS) and thereby eliminating the hand points from the scene, points belonging to the object are more likely to be sampled, which dramatically improves the performance of (Super4PCS+*HS*). Recent state-of-the-art learning-based approaches are also evaluated. DOPE [14] trains a neural network to predict 3D bounding-box vertices projected on the image and recovers 6D pose from them via Perspective-N-Point (PnP), which has shown to outperform PoseCNN [13] on the YCB dataset. To eliminate the domain gap from the scope of evaluation, the training and test data were generated in the same simulator and domain randomization was utilized as suggested [14]. AAE [60] is another learning-based method that trains an autoencoder network to embed object 3D orientation information using extensive data augmentation and domain randomization techniques. It has been shown to be successfull on textureless objects and achieved state-of-art results on the T-LESS dataset [58]. This approach is only able to predict 3D orientation. The translation is based on the output of another object detection network. For the scope of this evaluation *ground-truth* bounding-box were provided as input to AAE [60].

A dramatic performance improvement is observed for all methods, when the proposed `PSO` hand state estimation is utilized to remove the hand related point cloud from the scene. This proves the significance of additionally estimating the robot hand state for in-hand 6D object pose estimation.

D. Real Dataset and Results

The real dataset contains 986 snapshots of 2 Yale T42 hands holding 4 types of objects including cylinder (295), ellipse (239), cuboid (187) and tless3 (265). All the objects and the adaptive hands are 3D printed. Similar to the setting in simulation, the adaptive hands are painted in two colors:

Method	Modality	cylinder	cuboid	ellipse	tless3	Avg.
Super4PCS [53] + *HS*	Depth	52.49	43.85	62.64	62.64	55.41
Super4PCS [53] + *HS+ICP*	Depth	70.51	43.85	54.81	78.49	61.92
*AAE** [60]	RGB	11.19	8.56	15.92	40.38	19.01
*AAE** [60] + *ICP*	RGBD	43.39	22.99	27.35	55.85	37.40
*AAE** [60] + *HS+ICP*	RGBD	41.02	29.41	29.80	81.89	45.53
OURS	Depth	**87.12**	**72.19**	**80.82**	**93.96**	**83.52**

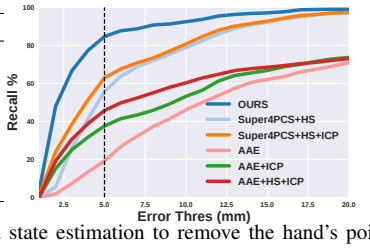

TABLE I: **Left:** Recall percentage ($e_{ADI} < 5mm$) on real data: *+HS* means using the proposed PSO hand state estimation to remove the hand's point cloud, *+ICP* means applying Point-to-Plane ICP at the end for pose refinement. AAE* [60] is provided a ground-truth bounding-box. **Right:** recall-threshold curves of compared methods on real data.

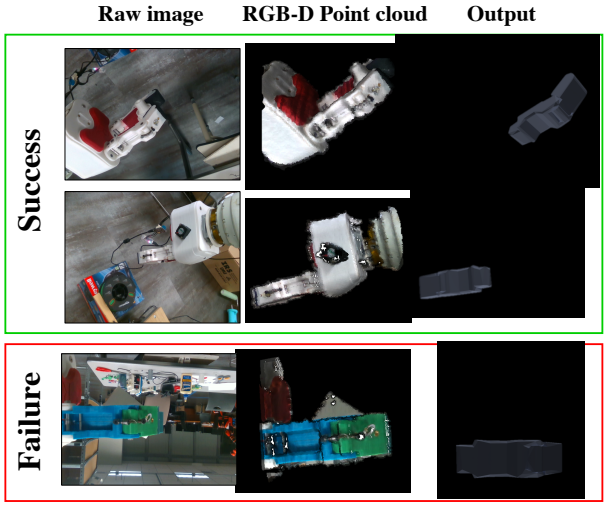

Fig. 7. Qualitative results for the proposed approach showing success and failure cases under challenges like occlusion and symmetry.

blue-green and white. The images are collected with an Intel RealSense SR300 RGB-D camera and the ground-truth poses are manually annotated using a GUI developed by the authors. Before each image is taken, objects are grasped randomly and the adaptive hand performs a random within-hand manipulation. Due to the small size of objects relative to the hand, severe occlusions occur frequently, as exhibited in Fig. 7.

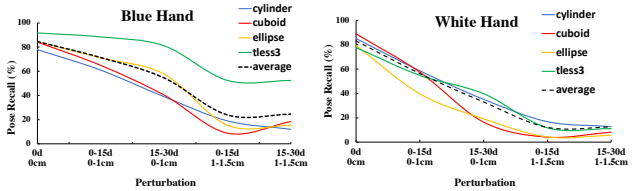

Fig. 8. Pose recall of [7] on the real dataset. As the approach requires initialization, it is evaluated over perturbations on the ground-truth pose.

Table I presents results on real data. Given the large appearance gap between synthetic training data and real scenarios, and the presence of textureless objects, the performance of DOPE does not translate well, and thereby was dropped from the table. AAE [60] was robust to some of these challenges and given the ground-truth bounding-boxes, it could predict the correct rotation in some cases. An additional related work [7] was evaluated on real data. It was developed to perform pose estimation for in-hand objects during robot manipulation. It assumes the initial object pose does not change much upon grasping and serves as an initialization for ICP. To evaluate this approach, pose initialization is provided by perturbing the ground-truth pose. Fig. 8 shows how the performance of this approach varies with the perturbation. Our proposed approach outperforms the best-case (small perturbation) of [7] even though pose initialization is not provided to our system.

Method	Mean recall (%)
Baseline	8.44
(+) `PSO` handpose	61.92
(+) `PPF`-constrained sampling	75.97
(+) Heuristic sampling	79.80
(+) Hypothesis pruning	83.52

TABLE II: Ablation study of critical components in our system. Results are averaged across the entire real dataset. `Baseline` refers to random base sampling on the entire scene cloud.

E. System Analysis

Fig. 7 exhibits examples of the output from the proposed approach on real data where severe occlusions occur and additional challenge is introduced by virtue of the noise in consumer-level depth sensor. Table. II shows the ablation study where the recall percentage for the object pose ($e_{ADI} < 5mm$) is measured by incrementally adding the critical proposed components.

Component	Speed (ms)	
	Mean	Std
Pointcloud processing	66.47	10.63
Hand wrist ICP	9.15	3.46
Hand pose estimation	45.98	3.58
Pose hypothesis generation	90.06	12.00
Pose clustering and ICP	61.41	11.84
Pose Hypothesis pruning	231.47	62.95
Pose selection	73.95	14.05
Misc	38.17	10.60
Total	616.64	64.50

TABLE III: Run-time decomposition of the system on real data.

Table III presents the overall and the decomposition of the running time for each component of the proposed pipeline, when tested on real data. *Misc* includes transformations, building KDTree, etc. Given the parallel implementation, the proposed technique requires a relatively short amount of time to perform a single image pose estimation without any initialization such as in tracking.

VI. CONCLUSIONS

This work presents a framework for fast and robust 6D pose estimation of in-hand objects. Due to the lack of relevant datasets, both real and synthetic data will be released as a benchmark for 6D object pose estimation applied to robot in-hand manipulation. Extensive experiments demonstrate advantages of the proposed method: robustness under severe occlusions and adaptation to different objects while able to run fast as a single-image pose estimation method. Although not real time, it could be integrated with tracking-based methods to provide initialization or recovery from lost tracking.

REFERENCES

[1] N. Correll, K. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Osada, A. Rodriguez, J. Romano, and P. Wurman, "Analysis and Observations From the First Amazon Picking Challenge," *T-ASE*, 2016.

[2] A. Zeng, K. Yu, S. Song, D. Suo, E. Walker, A. Rodriguez, and J. Xiao, "Multiview self-supervised deep learning for 6d pose estimation in the amazon picking challenge," in *ICRA*, 2019.

[3] M. Schwarz, A. Milan, A. S. Periyasamy, and S. Behnke, "Rgb-d object detection and semantic segmentation for autonomous manipulation in clutter," *The International Journal of Robotics Research*, 2018.

[4] M.-Y. Liu, O. Tuzel, A. Veeraraghavan, Y. Taguchi, T. K. Marks, and R. Chellappa, "Fast object localization and pose estimation in heavy clutter for robotic bin picking," *The International Journal of Robotics Research*, 2012.

[5] M. Zhu, K. G. Derpanis, Y. Yang, S. Brahmbhatt, M. Zhang, C. Phillips, M. Lecce, and K. Daniilidis, "Single image 3d object detection and pose estimation for grasping," in *ICRA*, 2014.

[6] C. Mitash, A. Boularias, and K. Bekris, "Physics-based scene-level reasoning for object pose estimation in clutter," *The International Journal of Robotics Research*, 2019.

[7] C. Choi, J. Del Preto, and D. Rus, "Using vision for pre-and post-grasping object localization for soft hands," in *ISER*, 2016.

[8] J. Issac, M. Wüthrich, C. G. Cifuentes, J. Bohg, S. Trimpe, and S. Schaal, "Depth-based object tracking using a robust gaussian filter," in *ICRA*, 2016.

[9] S. Trinh, F. Spindler, E. Marchand, and F. Chaumette, "A modular framework for model-based visual tracking using edge, texture and depth features," in *IROS*, 2018.

[10] C. Choi and H. I. Christensen, "Rgb-d object tracking: A particle filter approach on gpu," in *IROS*, 2013.

[11] M. F. Fallon, H. Johannsson, and J. J. Leonard, "Efficient scene simulation for robust monte carlo localization using an rgb-d camera," in *2012 IEEE international conference on robotics and automation*. IEEE, 2012, pp. 1663–1670.

[12] M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray *et al.*, "Learning dexterous in-hand manipulation," *arXiv preprint arXiv:1808.00177*, 2018.

[13] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," *arXiv preprint arXiv:1711.00199*, 2017.

[14] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects," *arXiv preprint arXiv:1809.10790*, 2018.

[15] M. Kokic, D. Kragic, and J. Bohg, "Learning to estimate pose and shape of hand-held objects from rgb images," *arXiv preprint arXiv:1903.03340*, 2019.

[16] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3d object recognition," in *2010 IEEE computer society conference on computer vision and pattern recognition*. Ieee, 2010, pp. 998–1005.

[17] A. G. Buch, L. Kiforenko, and D. Kraft, "Rotational subgroup voting and pose clustering for robust 3d object recognition," in *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 4137–4145.

[18] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3dmatch: Learning local geometric descriptors from rgb-d reconstructions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1802–1811.

[19] L. U. Odhner, R. R. Ma, and A. M. Dollar, "Open-loop precision grasping with underactuated hands inspired by a human manipulation strategy," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 3, pp. 625–633, 2013.

[20] A. Sintov, A. S. Morgan, A. Kimmel, A. M. Dollar, K. E. Bekris, and A. Boularias, "Learning a state transition model of an underactuated adaptive hand," *IEEE Robotics and Automation Letters*, vol. 4, pp. 1287–1294, 2019.

[21] A. Kimmel, A. Sintov, J. Tan, B. Wen, A. Boularias, and K. E. Bekris, "Belief-space planning using learned models with application to underactuated hands," in *International Symposium on Robotics Research (ISRR)*, 2019.

[22] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian conference on computer vision*. Springer, 2012, pp. 548–562.

[23] M. T. Mason, A. Rodriguez, S. S. Srinivasa, and A. S. Vazquez, "Autonomous manipulation with a general-purpose simple hand," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 688–703, 2012.

[24] B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus, "Haptic identification of objects using a modular soft robotic gripper," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 1698–1705.

[25] S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, and S. Levine, "Manipulation by feel: Touch-based control with deep predictive models," *arXiv preprint arXiv:1903.04128*, 2019.

[26] J. Bimbo, S. Luo, K. Althoefer, and H. Liu, "In-hand object pose estimation using covariance-based tactile to geometry matching," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 570–577, 2016.

[27] K. Aquilina, D. A. Barton, and N. F. Lepora, "Principal components of touch," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8.

[28] T. Schmidt, K. Hertkorn, R. Newcombe, Z. Marton, M. Suppa, and D. Fox, "Depth-based tracking with physical constraints for robot manipulation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 119–126.

[29] P. K. Allen, "Integrating vision and touch for object recognition tasks," *The International Journal of Robotics Research*, vol. 7, no. 6, pp. 15–33, 1988.

[30] P. Hebert, N. Hudson, J. Ma, and J. Burdick, "Fusion of stereo vision, force-torque, and joint sensors for estimation of in-hand object location," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 5935–5941.

[31] L. Zhang and J. C. Trinkle, "The application of particle filtering to grasping acquisition with visual occlusion and tactile sensing," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 3805–3812.

[32] K.-T. Yu and A. Rodriguez, "Realtime state estimation with tactile and visual sensing. application to planar manipulation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7778–7785.

[33] M. Pfanne, M. Chalon, F. Stulp, and A. Albu-Schäffer, "Fusing joint measurements and visual features for in-hand object pose estimation," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3497–3504, 2018.

[34] J. Bimbo, L. D. Seneviratne, K. Althoefer, and H. Liu, "Combining touch and vision for the estimation of an object's pose during manipulation," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4021–4026.

[35] M. Chalon, J. Reinecke, and M. Pfanne, "Online in-hand object localization," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 2977–2984.

[36] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NIPS*, 2015, pp. 91–99.

[37] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016, pp. 779–788.

[38] C. Mitash, B. Wen, K. Bekris, and A. Boularias, "Scene-level pose estimation for multiple instances of densely packed objects," *arXiv preprint arXiv:1910.04953*, 2019.

[39] B. Tekin, S. N. Sinha, and P. Fua, "Real-time seamless single shot 6d object pose prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 292–301.

[40] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," *arXiv preprint arXiv:1901.04780*, 2019.

[41] M. Kokic, D. Kragic, and J. Bohg, "Learning to estimate pose and shape of hand-held objects from rgb images," *ArXiv*, vol. abs/1903.03340, 2019.

[42] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 807–11 816.

[43] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *European Conference on Computer Vision*. Springer, 2016, pp. 766–782.

[44] M. Wüthrich, P. Pastor, M. Kalakrishnan, J. Bohg, and S. Schaal, "Probabilistic object tracking using a range camera," *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3195–3202, 2013.

[45] X. Deng, A. Mousavian, Y. Xiang, F. Xia, T. Bretl, and D. Fox, "Poserbpf: A rao-blackwellized particle filter for 6d object pose tracking," *arXiv preprint arXiv:1905.09304*, 2019.

[46] R. Munoz-Salinas, "Aruco: a minimal library for augmented reality applications based on opencv," *Universidad de Córdoba*, 2012.

[47] B. alli, K. Srinivasan, A. Morgan, and A. M. Dollar, "Learning modes of within-hand manipulation," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3145–3151, 2018.

[48] S. Cruciani, C. Smith, D. Kragic, and K. Hang, "Dexterous manipulation graphs," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2040–2047, 2018.

[49] T. Schmidt, R. A. Newcombe, and D. Fox, "Dart: Dense articulated real-time tracking." in *Robotics: Science and Systems*, vol. 2, no. 1. Berkeley, CA, 2014.

[50] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun, "Realtime and robust hand tracking from depth," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1106–1113.

[51] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust surface registration," *ACM Transactions on Graphics*, vol. 27, no. 3, pp. #85, 1–10, 2008.

[52] C. G. Cifuentes, J. Issac, M. Wüthrich, S. Schaal, and J. Bohg, "Probabilistic articulated real-time tracking for robot manipulation," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 577–584, 2016.

[53] N. Mellado, D. Aiger, and N. J. Mitra, "Super 4pcs fast global pointcloud registration via smart indexing," in *Computer Graphics Forum*, vol. 33, no. 5. Wiley Online Library, 2014, pp. 205–215.

[54] C. Mitash, A. Boularias, and K. Bekris, "Robust 6d object pose estimation with stochastic congruent sets," in *BMVC*, 2018.

[55] C. Mitash, A. Boularias, and K. E. Bekris, "Improving 6D pose estimation of objects in clutter via physics-aware monte carlo tree search," in *ICRA*, 2018.

[56] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and vision computing*, vol. 10, no. 3, pp. 145–155, 1992.

[57] V. Ganapathi, C. Plagemann, D. Koller, and S. Thrun, "Real-time human pose tracking from range data," in *European conference on computer vision*. Springer, 2012, pp. 738–751.

[58] T. Hodaň, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis, "T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects," *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017.

[59] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The ycb object and model set: Towards common benchmarks for manipulation research," in *2015 international conference on advanced robotics (ICAR)*. IEEE, 2015, pp. 510–517.

[60] M. Sundermeyer, Z. Marton, M. Durner, and R. Triebel, "Implicit 3d orientation learning for 6d object detection from rgb images," in *ECCV*, 2018, pp. 699–715.