

Globally Optimal Object Pose Estimation in Point Clouds with Mixed-Integer Programming

Gregory Izatt and Russ Tedrake

CSAIL, Massachusetts Institute of Technology, Cambridge, MA

Email: {gizatt, russt}@mit.edu

Abstract—Motivated by the limitations of local object trackers, we present a formulation of the underlying point-cloud object pose estimation problem as a mixed-integer convex program, which we efficiently solve to optimality with an off-the-shelf branch and bound solver. We show that reasoning about object pose estimation in this way allows natural extension to point-to-mesh correspondence, multiple object pose estimation, and outlier rejection without losing the ability to obtain a globally optimal solution. We probe the extent to which rich problem-specific formulations typically tackled with unreliable nonlinear optimization can be rigorously treated in a global optimization framework to provide reliable initializations for powerful-but-local object trackers.

I. INTRODUCTION

The robotic perception community has placed significant emphasis on designing and improving perception modules for estimating the poses of objects in a scene. These modules have enormous value for autonomous systems, in that they reduce extremely high-dimensional sensor inputs to compact and semantically loaded object state that can be consumed by a broad range of motion planners and robot controllers.

Here, we are concerned with systems for estimating object poses from point cloud information, e.g. from increasingly ubiquitous RGB-D cameras. Myriad techniques perform pose estimation without an initial guess, e.g. via sampling [11, 23, 26], feature extraction [7] [17] [31] [32] [35], template matching [14], shape descriptors [1], and direct machine learning [33]. However, because of the scale of the sensory data and the difficulty of the global optimization, few of these techniques run in real time, and those that do can't make claims concerning reliability and convergence to global optimality. Given a reasonable initial guess, another broad class of pose tracking techniques perform real-time object tracking. These techniques have grown extremely mature, boasting support for multiple articulated objects [28], deformable objects [29], and support for tactile sensing [27] [20] [16] [18].

In this work, we present a study of the core optimization problem that underlies many of these techniques. We show that the pose estimation problem for point clouds can be viewed through the lens of mixed-integer programming, and that doing so leads to a problem formulation permitting optimization to certifiable approximate global optimality. This formulation is written in a general form that is extensible to handle explicit outlier rejection and multiple models, and can consume the output of other local and global pose estimation algorithms as

seeds to accelerate the global search – and in doing so, verify the optimality of the output of those other algorithms.

II. PROBLEM FORMULATION

We will focus on an instance of the *point-cloud object pose estimation problem* which involves finding the best parameters of a rigid body model to explain the data available from a sensed point cloud. In this problem, model parameters are rotations $R \in SO(3)$ and translations $T \in \mathbb{R}^3$. The point cloud sensor samples a set of N_s points $z = \{s_i\}$ from the geometry of the world. Many techniques represent the model as a collection of N_m point features, which leads to an optimization penalizing a norm (shown here in the single-object case):

$$\min_{R, T, C} \sum_{i \in [1, N_s]} \|Rs_i + T - m_{C(i)}\|, \quad (1)$$
$$C(i) = \operatorname{argmin}_{j \in [1, N_m]} \|Rs_i + T - m_j\|,$$

where $C(i)$ corresponds each scene point to the closest model feature according to the desired norm.

Objectives like this one are reflected in many of the pose estimation techniques in the literature. Key differences between these techniques lie in the model representation and distance function used; the method of optimization; and further additions to the objective beyond optimization of just a distance function.

A critical feature of this problem is that the correspondences C and transformation $\{R, T\}$ are each independently sufficient to specify a solution to this problem. Given the correspondences, the optimal transformation can be computed in closed form [9]. Given the transformation, correspondences can be backed out if desired via, e.g., closest point lookups on the model. The famous Iterative Closest Point (ICP) algorithm, from which many object trackers are inspired, performs Expectation Maximization by alternating between solving these two problems [3] [5].

III. RELATED WORK

At a high level, two broad classes of approaches have been used to attempt global optimization of this point cloud pose estimation objective directly on the raw point cloud:

semidefinite programming (SDP) relaxation, and branch and bound search.

SDP relaxation of the rotation and correspondence constraints constrain R to be within a convex hull of $SO(3)$, and allow a continuous relaxation of C [2]. This relaxation transforms the difficult nonlinear problem to a much easier convex one. This technique has proven very powerful for solving the Procrustes Matching (PM) problem, [21], and similar SDP relaxation can be applied to the pose estimation problem with fixed correspondences (i.e. an alignment problem) [4]. Their method boasts tightness up to a quantified noise threshold, and is demonstrated aligning 800 points across 30 overlapping point clouds.

Other methods perform branch and bound over the space of rotations [13, 19, 25], or rotations and translations [34]. The latter – GO-ICP – accomplishes our broad goal of providing globally optimal pose estimates, but it does not explicitly reason about correspondences. This manifests itself most clearly in the handling of outliers: a user of GO-ICP must specify the expected fraction of outliers ahead of time, and setting the parameter incorrectly frequently results in invalid results. Other techniques take direct advantage of the property that it is easy to detect inconsistencies in small sets of correspondences in order to prune branches in the search tree [10, 12].

The transform and correspondence information are tightly coupled in the pose estimation problem. Thus, a formulation that reasons about point correspondences and model transformations *simultaneously* stands to benefit from this interplay. We present such a formulation based on mixed-integer programming [24]. While even restricted class of mixed-integer linear programs is itself NP-hard, MIPs that are convex in their continuous and integer variables are amenable to branch and bound search that can be very efficient, given the right problem structure (e.g. mixed-integer linear programs (MILP)). These algorithms are implemented by powerful off-the-shelf solvers capable of solving problems with millions of variables and constraints [15].

IV. MIXED-INTEGER PROBLEM FORMULATION

Our formulation of this problem uses a generalized mesh model to represent the objects, for the reason that the mesh model is significantly more compact than a traditional sampled point model. Given a model defined by N_m vertices and N_f faces, where each face is defined as an affine combination of a subset of coplanar vertices, as well as

- scene points $S = \{s_i\}, i \in [1, N_s]$,
- model vertices $M = \{m_j\}, j \in [1, N_m]$,
- a binary face membership map $F \in \{0, 1\}^{N_f \times N_m}$,

the generic pose estimation problem is equivalent to finding a rotation matrix R , a translation matrix T , a combination matrix $C \in \mathcal{R}^{N_s \times N_m}$, and a face correspondence matrix $f \in \{0, 1\}^{N_s \times N_f}$ that satisfy the following.

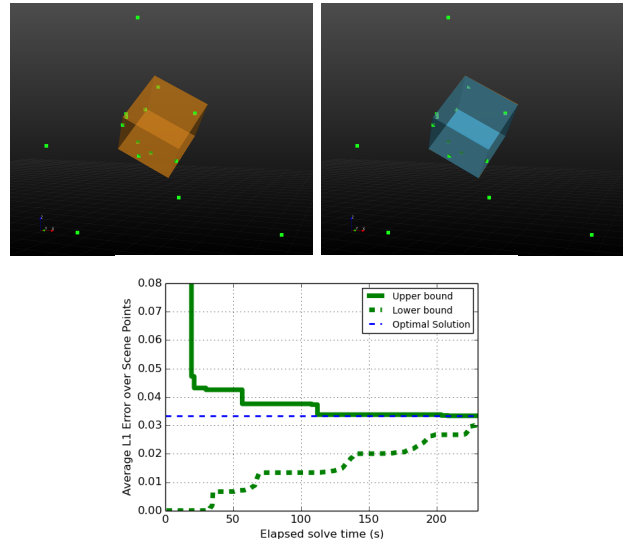


Fig. 1: Pose estimate produced by our MILP mesh model formulation for a cube model of 12 triangular faces, given 15 scene points with 5 outliers. The solution shown here has optimal cost that matches the optimal cost of the ground truth solution. Optimality of this solution was certified to a MIP gap of 5%. **Top left:** Ground truth pose. **Top right:** Pose estimate using our MILP mesh-model formulation. **Bottom:** Convergence time of upper and lower bounds across time for the MILP solution.

$$\begin{aligned}
 & \underset{R, T, C, f}{\text{minimize}} && \frac{1}{N_s} \sum_{i \in [1, N_s]} \|Rs_i + T - MC_{i,:}^T\|_2^2 \\
 & \text{subject to} && R \in SO(3), \\
 & && \sum_{j \in [1, N_m]} C_{i,j} = 1, \forall i, \\
 & && \sum_{k \in [1, N_f]} f_{i,k} = 1, \forall i, \\
 & && 0 \leq C_{i,j} \leq F f_{i,j}, \forall i, j, \\
 & && f_{i,j} \in [0, 1], \forall i, j.
 \end{aligned}$$

The affine combination coefficient for the i^{th} scene point and j^{th} model point $C_{i,j}$ is constrained to be inactive unless one of the faces for model point j is active. Scene points can only correspond to a single model face.

To relax this approximation to a mixed-integer *linear* program, we approximate the $R \in SO(3)$ constraint with a piecewise convex outer approximation in the spirit of the McCormick Envelope [22]. For each member of the rotation matrix $R_{i,j}$, we introduce new binary variables to assign $R_{i,j}$ to one of N_k partitions of $[-1, 1]$. These binary variables are used to activate region-specific constraints approximating $R \in SO(3)$, as presented in [6].

V. EXTENSIONS

A. Handling Outliers

Correct outlier handling is critical for object pose estimation algorithms, as point clouds in the wild invariably include unmodeled points from nearby objects and support surfaces in the scene. We extend this formulation to allow scene points to be explicitly classified as outliers.

We first switch from the L-2 to the L-1 norm in our error metric, so that we can include the distance to each point in the set of linear constraints. We introduce an intermediate variable ϕ_i for each scene point s_i storing the L-1 distance from s_i to the matched point on the model. Additional slack variables $\alpha^{i,l}$ are introduced to implement the $l \in 1 \dots 3$ absolute values within the L-1 norm for each scene index i . We bound ϕ_i with a constant *maximum allowed L-1 distance* ϕ_{max} as a threshold (and penalty) for classifying points as outliers. Finally, we add a new binary variable o_i for each scene point indicating that that scene point is being considered an outlier.

In the mesh-model case, we now solve (for large \mathbb{M}):

$$\begin{aligned}
 & \underset{R,C}{\text{minimize}} && \min \frac{1}{N_s} \sum_{i \in [1, N_s]} \phi_i \\
 & \text{subject to} && \text{Relaxed } R \in SO(3), \\
 & && \phi_i \geq \mathbf{1}^T \alpha_{i,l}, \\
 & && \phi_i \geq \phi_{max} o_i, \\
 & && \alpha_i \geq + (R s_i + T - M C_{i,:}^T) - \mathbb{M} o_i, \quad (2) \\
 & && \alpha_i \geq - (R s_i + T - M C_{i,:}^T) - \mathbb{M} o_i, \quad (3) \\
 & && \sum_{j \in N_m} C_{i,j} + o_i = 1, \\
 & && \sum_{k \in N_f} f_{i,k} + o_i = 1, \\
 & && 0 \leq C_{i,j} \leq F f_i, \\
 & && \phi_i, \alpha_i \geq 0, \\
 & && f_{i,j}, o_i \in \{0, 1\}.
 \end{aligned}$$

B. Handling Multiple Objects

Using similar machinery to that employed to correspond to outliers, we can extend our formulation further to support multiple objects. We can extend the formulation to simultaneously optimize over multiple rotations and translations $\{R_1, T_1\}, \dots, \{R_{N_b}, T_{N_b}\}$ for N_b separate bodies. Given a map $B \in \{0, 1\}^{N_b \times N_f}$, where the $(i, j)^{th}$ entry indicates if face j is a member of body i , we can replace constraints (2) and (3) with the disjunction

$$\begin{aligned}
 & \forall i \in N_m, k \in N_b : \\
 & \alpha_i \geq + (R_k s_i + T - M C_{i,:}^T) - \mathbb{M}(1 - B_{k,:} f_{i,:}^T), \\
 & \alpha_i \geq - (R_k s_i + T - M C_{i,:}^T) - \mathbb{M}(1 - B_{k,:} f_{i,:}^T).
 \end{aligned}$$

where the expression $\mathbb{M}(1 - B_{k,:} f_{i,:}^T)$ deactivates the constraint if the current assignment f does not assign scene point i to a

face on body k .

C. Using Other Pose Estimation Methods as a Heuristic

A benefit of optimizing directly over the fundamental problem addressed by a wide class of pose estimation methods is that we can take advantage of solutions generated by those other methods by consuming them as candidate feasible solutions. The branch and bound algorithm (and solvers that implement it) is able to asynchronously consume feasible solution guesses as nodes in the search tree. These new feasible solutions provide upper bounds on the global optimal cost, which are used to prune bad nodes. Because a significant amount of search time is spent finding better feasible solutions (as can be seen in the results in e.g. Figure 1), getting better feasible solutions from faster but less-consistent pose estimation methods can improve the runtime of the global optimization. This ability also means that this formulation can be used to post-process the output of notoriously unpredictable methods, like neural networks, in order to guarantee stable results without completely discarding the efficiency of the original method.

Given the mesh model MILP formulation described above and a candidate pose $\{R_0, T_0\}$ generated by any method, one can extract $C, f, \phi,$ and α via closest-point queries against the mesh models. The means to extract the variables in the rotation approximation vary, but in this case of the piecewise-linear convex approximations, the value of R directly determines which binary variables should be active.

VI. RESULTS

We executed several experiments to verify our formulation on synthetic data. To perform these experiments, we implemented both formulations in both C++ and Julia, relying on the Drake [30] and JuMP [8] symbolic optimization libraries respectively. We used Gurobi 7.0.2 [15] as a backend to solve the resulting mixed-integer programs.

A. General Performance

We generated a synthetic point cloud from a cube model with a side length of 1 unit. We generated 15 scene points, with 10 sampled randomly from the surface of the cube at its ground truth pose, and 5 more generated randomly in the area around the cube. We included outliers in this test case to illuminate how the solver performs in terms of progress of the upper *and lower* bounds – without outliers, the optimal error would be close to the trivial lower bound of 0. An optimal fit in this configuration has an optimal average saturated L-1 error of 0.033: the $\frac{2}{3}$ of points that are inliers have L-1 error of 0, and the $\frac{1}{3}$ of points that are outliers have L-1 error of $\geq \phi_{max} = 0.1$ by construction.

The MILP mesh model formulation converged to the optimal solution and certified its global optimality to within a MIP gap of 5% (Figure 1). This desired MIP gap is tunable, and trades off with runtime. This gap of 5% was chosen arbitrarily, and corresponds to an optimality gap of $0.033 \times 0.05 = 0.00165$ average L-1 error over the scene

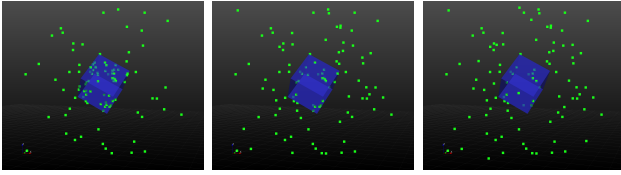


Fig. 2: Pose estimates produced by our MILP mesh model formulation for a cube model of 12 triangular faces, given 100 scene points, with a varying number of them being outliers: **Left:** 50% outliers, converged in 50s. **Middle:** 80% outliers, converged in 400s. **Right:** 90% outliers, converged in 2000s. Rotations were frozen to the ground truth rotation in order to produce these solutions in reasonable time. All solutions shown here have optimal cost that matches the optimal cost of the ground truth solution and align with the ground truth pose. Optimality of these solutions were certified to a MIP gap of 5%.

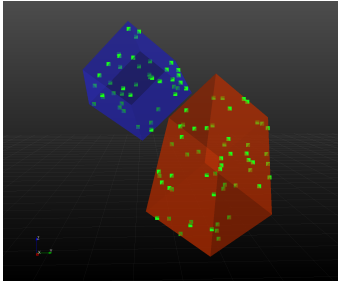


Fig. 3: Pose estimates produced by our MILP mesh model formulation simultaneously fitting two box models to 100 scene points with no outliers. Rotations were frozen to the ground truth rotations in order to produce these solutions in reasonable time. Convergence took 1100s.

points. We used 4 binary variables per element of R . The largest elementwise infeasibility of $R^T R = I$ being 0.020, and $\det(R)$ was 1.002. These values reflect that the approximation of $R \in SO(3)$ is reasonably tight.

B. Outlier Rejection and Multiple Models

To highlight the extensions of our formulation, we generated similar synthetic point clouds to test the outlier rejection and multiple model cases, with results shown in Figures 2 and 3. To avoid unreasonably long runtimes, we had to constrain rotations and limit the search to be over translations and correspondences. R was thus constrained to take the value of the ground truth rotation.

C. Upper Bounds from ICP

To demonstrate that solutions generated from other efficient but non-global methods can be leveraged to make our global optimization faster, we implemented an ICP-based heuristic for generating candidate feasible solutions online during the optimization. This procedure is directly inspired by GO-ICP [34]. Our solver maintains a queue of feasible solutions found by the branch and bound algorithm, and runs point-to-plane ICP

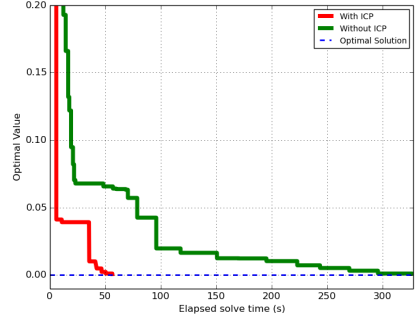


Fig. 4: Comparison of the upper bound convergence behavior of the MILP mesh model formulation with 50 scene points and 0 outliers fitting a box model, with and without an ICP algorithm generating novel feasible solutions in parallel. The lower bound is omitted, as it is trivially 0 for the 0 outlier case.

with proportional outlier rejection on each feasible solution in a parallel thread alongside the global optimization solver. If the ICP produces a solution better than the best currently held by the solver, the ICP solution is handed to the solver as a heuristically-derived feasible solution. This procedure significantly improves runtime, as is shown in Figure 4.

VII. CONCLUSION

The formulations we present can be used to find certifiably globally optimal solutions for small numbers of scene points and outliers, even in the face of combinatorial complexity. The solver is capable of finding and certifying the right solution, even in very high outlier ratios, and can optimize with multiple objects seamlessly. However, this technique requires further work to scale to practical point clouds from experimental data. Careful analysis of the time scaling of the algorithm with respect to scene and model complexity is merited; preliminary results support that this scaling is polynomial but potentially high order. Another critical future direction is to compare the outlier handling capability of this technique to other global pose estimation algorithms via experiments on real data.

That this technique can so easily incorporate hypotheses from other methods makes it a candidate for being an offline *verification* technique for the results from other efficient but inconsistent pose estimation methods. This functionality is critical when considering the kinds of highly ambiguous point clouds that result from highly cluttered scenes, and from tactile sensing. Further, the formulation and its branch and bound solution has the significant advantage that, by examining partial relaxations of the problem, it either verifies that a solution is globally optimal, or provides a search region that may contain a better solution. Our formulation provides a direction to extend that powerful search strategy to a more general and extensible formulation of the point cloud pose estimation problem.

REFERENCES

- [1] Ceyhan Burak Akgül, Bülent Sankur, Yücel Yemez, and Francis Schmitt. 3d model retrieval using probability density-based shape descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):1117–1133, 2009.
- [2] Mica Arie-Nachimson, Shahar Z Kovalsky, Ira Kemelmacher-Shlizerman, Amit Singer, and Ronen Basri. Global motion estimation from point matches. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*, pages 81–88. IEEE, 2012.
- [3] Paul J. Besl and Neil D. McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.
- [4] Kunal N Chaudhury, Yuehaw Khoo, and Amit Singer. Global registration of multiple point clouds using semidefinite programming. *SIAM Journal on Optimization*, 25(1):468–501, 2015.
- [5] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and vision computing*, 10(3):145–155, 1992.
- [6] Hongkai Dai and Russ Tedrake. Global inverse kinematics via mixed integer convex optimization. 2017.
- [7] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 998–1005. Ieee, 2010.
- [8] Iain Dunning, Joey Huchette, and Miles Lubin. Jump: A modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320, 2017. doi: 10.1137/15M1020575.
- [9] D. W. Eggert, A. Lorusso, and R. B. Fisher. Estimating 3-d rigid body transformations: A comparison of four major algorithms. *Mach. Vision Appl.*, 9(5-6):272–290, March 1997. ISSN 0932-8092. doi: 10.1007/s001380050048. URL <http://dx.doi.org/10.1007/s001380050048>.
- [10] Olof Enqvist, Klas Josephson, and Fredrik Kahl. Optimal correspondences from pairwise constraints. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1295–1302. IEEE, 2009.
- [11] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [12] Natasha Gelfand, Niloy J Mitra, Leonidas J Guibas, and Helmut Pottmann. Robust global registration. In *Symposium on geometry processing*, volume 2, page 5, 2005.
- [13] Richard I Hartley and Fredrik Kahl. Global optimization through rotation space search. *International Journal of Computer Vision*, 82(1):64–79, 2009.
- [14] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision*, pages 548–562. Springer, 2012.
- [15] Gurobi Optimization Inc. Gurobi optimizer reference manual, 2016. URL <http://www.gurobi.com>.
- [16] Gregory Izatt, Geronimo Mirano, Edward Adelson, and Russ Tedrake. Tracking objects with point clouds from vision and touch. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017.
- [17] Andrew E. Johnson and Martial Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on pattern analysis and machine intelligence*, 21(5):433–449, 1999.
- [18] Matthew Klingensmith, Michael C Koval, Siddhartha S Srinivasa, Nancy S Pollard, and Michael Kaess. The manifold particle filter for state estimation on high-dimensional implicit manifolds. *arXiv preprint arXiv:1604.07224*, 2016.
- [19] Hongdong Li and Richard Hartley. The 3d-3d registration problem revisited. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.
- [20] Shuai Li, Siwei Lyu, and Jeff Trinkle. State estimation for dynamic systems with intermittent contact. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3709–3715. IEEE, 2015.
- [21] Haggai Maron, Nadav Dym, Itay Kezurer, Shahar Kovalsky, and Yaron Lipman. Point registration via efficient convex relaxation. *ACM Transactions on Graphics (TOG)*, 35(4):73, 2016.
- [22] Garth P McCormick. Computability of global solutions to factorable nonconvex programs: Part iconvex underestimating problems. *Mathematical programming*, 10(1): 147–175, 1976.
- [23] Nicolas Mellado, Dror Aiger, and Niloy J Mitra. Super 4pcs fast global pointcloud registration via smart indexing. In *Computer Graphics Forum*, volume 33, pages 205–215. Wiley Online Library, 2014.
- [24] George L Nemhauser and Laurence A Wolsey. Integer programming and combinatorial optimization. *Wiley, Chichester. GL Nemhauser, MWP Savelsbergh, GS Sigismondi (1992). Constraint Classification for Mixed Integer Programming Formulations. COAL Bulletin*, 20: 8–12, 1988.
- [25] Carl Olsson, Fredrik Kahl, and Magnus Oskarsson. Branch-and-bound methods for euclidean registration problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):783–794, 2009.
- [26] Chavdar Papazov and Darius Burschka. An efficient ransac for 3d object recognition in noisy and occluded scenes. *Computer Vision-ACCV 2010*, pages 135–148, 2011.
- [27] Tanner Schmidt, Katharina Hertkorn, Richard Newcombe, Zoltan Marton, Michael Suppa, and Dieter Fox. Depth-based tracking with physical constraints for robot

- manipulation. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 119–126. IEEE, 2015.
- [28] Tanner Schmidt, Richard Newcombe, and Dieter Fox. Dart: dense articulated real-time tracking with consumer depth cameras. *Autonomous Robots*, 39(3):239–258, 2015.
- [29] John Schulman, Alex Lee, Jonathan Ho, and Pieter Abbeel. Tracking deformable objects with point clouds. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1130–1137. IEEE, 2013.
- [30] Russ Tedrake and the Drake Development Team. Drake: A planning, control, and analysis toolbox for nonlinear dynamical systems, 2016. URL <http://drake.mit.edu>.
- [31] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique signatures of histograms for local surface description. In *European Conference on Computer Vision*, pages 356–369. Springer, 2010.
- [32] Paul Wohlhart and Vincent Lepetit. Learning descriptors for object recognition and 3d pose estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
- [33] Jay M Wong, Vincent Kee, Tiffany Le, Syler Wagner, Gian-Luca Mariottini, Abraham Schneider, Lei Hamilton, Rahul Chipalkatty, Mitchell Hebert, David Johnson, et al. Segicp: Integrated deep semantic segmentation and pose estimation. *arXiv preprint arXiv:1703.01661*, 2017.
- [34] Jiaolong Yang, Hongdong Li, Dylan Campbell, and Yunde Jia. Go-icp: a globally optimal solution to 3d icp point-set registration. *IEEE transactions on pattern analysis and machine intelligence*, 38(11):2241–2254, 2016.
- [35] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *European Conference on Computer Vision*, pages 766–782. Springer, 2016.