Deliberative Perception for Warehouse Automation

Venkatraman Narayanan Maxim Likhachev

***Abstract*— We describe the application of a recently developed *deliberative* perception framework to the task of multi-object instance recognition in warehouse environments. Traditional object recognition pipelines based exclusively on discriminative feature-matching and/or statistical learners are often sensitive to inter-object occlusions and the training data used. Deliberative approaches such as PERCH treat multi-object pose estimation as a generative global optimization over possible configurations of objects, thereby predicting and accounting for occlusions. Further, D2P—an extension of PERCH, leverages guidance from modern learning-based techniques to combine the efficiency of discriminative approaches with the robustness provided by global reasoning. We conclude with a discussion of how these approaches were used by Carnegie Mellon University's team in the 2016 Amazon Picking Challenge, and their role in the upcoming 2017 Amazon Robotics Challenge.**

I. Introduction

Warehouse environments pose a significant challenge for object recognition and localization. Densely packed shelves with extreme amounts of occlusion, deformable objects, and sensor-unfriendly (e.g. specular) objects are typical causes of perception brittleness [1]. At the same time, the semi-structured nature of warehouses, i.e, prior knowledge of environment, knowledge of shelf contents, and sensor pose provide opportunities to design algorithms that are more robust compared to general computer vision techniques. In this abstract, we discuss how a recently developed *deliberative* perception framework is well-suited to the class of problems arising in warehouse perception.

Traditional methods for object instance detection have relied on matching hand-coded feature descriptors between the observed scene and 3D models, with recent data-driven methods permitting automated learning of those feature descriptors. While these methods, broadly classified as *discriminative*, provide attractive test-time speeds, they remain brittle in practice despite numerous variants that have shown promise. As an example, consider the scene in Fig. 1, where we need to identify and localize the Elmers' glue bottle, which is almost completely occluded by the shelf. Methods that employ feature correspondence matching fare poorly as key feature descriptors could be lost due to occlusion by the shelf, whereas learning-based methods could suffer as they might have not seen a similar training instance where only such a small portion of the object is visible. However, one could jointly reason about the occlusion caused by the shelf and the positions of the other objects to infer the exact pose of the Elmer's glue bottle. This kind of global reasoning forms the basis of deliberative perception algorithms such as PERCH [2].

The Robotics Institute, Carnegie Mellon University, PA, USA {venkatraman,maxim} at cs.cmu.edu. This research was sponsored by ARL, under the Robotics CTA program grant W911NF-10-2-0016.

(a) (b)

Fig. 1: A typical scene from a warehouse picking task. Here, we have access to the list of objects in the bin, their 3D models and that of the shelf as well. (a) An image of the shelf-bin from which we are required to identify and localize the Elmer's glue bottle, marked by a red bounding box. (b) An image of the Elmer's glue bottle that needs to be localized.

II. Related Work

Discriminative Approaches. Typical approaches for object *instance* detection in point clouds employ local or global 3D feature descriptors. Approaches that use local descriptors follow a two step procedure: i) compute and find correspondences between a set of local shape-preserving 3D feature descriptors on the model and the observed scene and ii) estimate the rigid transform between a set of geometrically consistent correspondences. Examples of local 3D feature descriptors include Spin Images [3], Fast Point Feature Histograms (FPFH) [4], Signature of Histograms of Orientations (SHOT) [5] etc. Approaches that use global descriptors (e.g., VFH [6], CVFH [7], OUR-CVFH [8], GRSD [9] etc.) follow a three step procedure: i) build a database of 3D global descriptors on renderings corresponding to different viewpoints of each object during the training phase, ii) extract clusters belonging to individual objects in the test scene, and iii) match each cluster's 3D global descriptor to one in the database to obtain both identity and pose together. In both approaches, a final local optimization step such as Iterative Closest Point (ICP) [10] is often used to fine-tune the pose estimates. A comprehensive survey of descriptor-based methods is presented in [11].

Other discriminative approaches for object instance detection are based on template matching [12, 13], Hough forests [14] and deep neural networks trained on colorized versions of synthetically-generated or real depth images of object instances [15, 16].

Generative Approaches. Despite their speed and

prevalence, a primary limitation of descriptor-based and discriminatively-trained methods is their brittleness to occlusions and other variations not captured during the training phase. Further, they are ill-suited for *multi-object* instance detection and pose estimation since the training data needs to capture the combinatorics of the problem (i.e, the features learnt must be capable of predicting inter-object occlusions for arbitrary combinations of objects).

Generative approaches on the other hand treat multi-object pose estimation as an optimization or filtering problem over possible renderings of the scene [2, 17–19]. This allows them to inherently account for inter-object occlusions. Further, they do not require any semantic grouping/segmentation of points into "objects" as required by global descriptor approaches. In what follows, we will summarize our prior works under this category—Perception via Search (PERCH) [2] and D2P [19].

III. Technical Details

PERCH addresses the task of identifying and localizing multiple objects with known 3D models in a static depth image or point cloud. Formally, we are given the 3D models of N unique objects, an input point cloud I (which can also be generated from a depth image) containing $K \geq N$ objects, some of which may be duplicates of an unique instance, and the 6 DoF pose of the camera sensor along with its intrinsics. The number (K) and type of objects in the scene are assumed to be known a priori, but no "clustering" is required (i.e, the algorithm looks at the scene as a whole, rather than identifying and estimating the pose of each cluster in the scene). Objects are assumed to vary only in 3 DoF pose (x, y, yaw) with respect to their 3D model coordinate axes. In practice, we construct multiple 3D models of an object corresponding to canonical poses (stable configurations in the absence of other objects), and treat each of those as distinct objects. While this assumption is reasonable in most cases, we are currently investigating the extension of PERCH to tractably handle full 6 DoF pose.

Optimization Formulation. PERCH formulates the problem of identifying and obtaining the poses of objects $O_1, O_2, \ldots, O_K$ as that of finding the minimizer of an "explanation" cost function which captures how well the rendered scene matches the input scenes, paying due attention both-ways—i.e, points in the input cloud should have an associated point in the rendered cloud, and vice-versa. Formally,

$$J(O_{1:K}) = \underbrace{\sum_{\boldsymbol{p} \in \boldsymbol{I}} \textsc{Outlier}(\boldsymbol{p}|\boldsymbol{R_K})}_{J_{observed}(O_{1:K}) \text{ or } J_o} + \underbrace{\sum_{\boldsymbol{p} \in \boldsymbol{R_K}} \textsc{Outlier}(\boldsymbol{p}|\boldsymbol{I})}_{J_{rendered}(O_{1:K}) \text{ or } J_r} \tag{1}$$

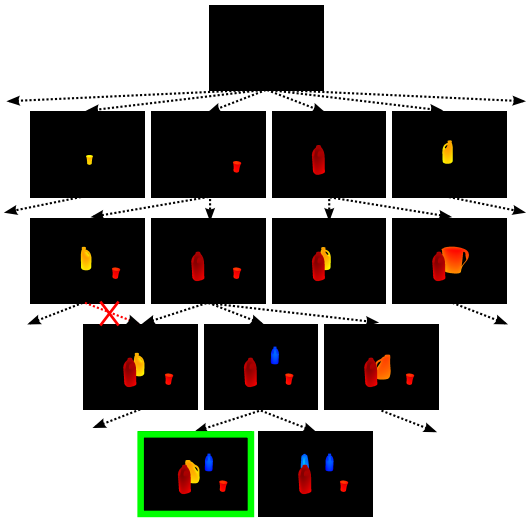


Fig. 2: Portion of a Monotone Scene Generation Tree (MSGT): the root of the tree is the empty scene, and new objects are added progressively as we traverse down the tree. Notice how child states never introduce an object that occludes objects already in the parent state. A counter-example (marked by the red cross) is also shown. Any state on the K^{th} level of the tree is a goal state, and the task is to find the one that has the lowest cost path from the root—marked by a green bounding box in this example.

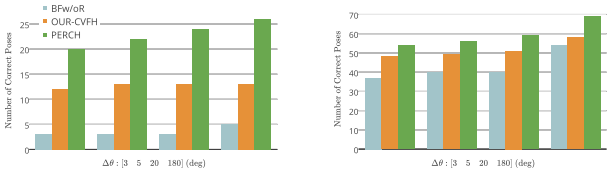
in which $\textsc{Outlier}(\boldsymbol{p}|\boldsymbol{P})$ for a point cloud $\boldsymbol{P}$ and point $\boldsymbol{p}$ is defined as follows:

$$\textsc{Outlier}(\boldsymbol{p}|\boldsymbol{P}) = \begin{cases} 1 & \text{if } \min_{\boldsymbol{p'} \in \boldsymbol{P}} \|\boldsymbol{p'} - \boldsymbol{p}\|_2 > \delta \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where δ represents the sensor noise resolution.

While this optimization problem looks completely intractable at the outset due to the combinatorially large search space (joint poses of all objects), the following insight allows us to circumvent exhaustive search: by enforcing a specific ordering in which object poses are assigned, the explanation cost can be decomposed into a sum of per-object costs, thereby allowing for a smarter search scheme. Specifically, the constraint on the ordering is that every time an object is added to the scene, it does not occlude any of the existing objects. In other words, the number of points in the rendered point cloud should be monotonically non-decreasing. This constraint on the ordering and the decomposition of the cost function results in the minimization problem being reduced to a tree search problem on what we call the Monotone Scene Generation Tree (MSGT). Specifically, the minimization problem is now equivalent to finding the shortest-cost path from the root node (empty scene) to a goal node (state with all object poses assigned) in the MSGT (Fig. 2).

Leveraging Discriminative Guidance. While tree-search is much more tractable than exhaustive search over the joint object poses, the branching factor is still large and could result in prohibitively large run times. In D2P [19], PERCH was extended to leverage arbitrary discriminative techniques as heuristics to guide the tree search. Specifically, by adopting a multi-heuristic graph search algorithm which supports inadmissible heuristics [20], D2P incorporates learning-based



(a) $\Delta t = 0.01m$ (b) $\Delta t = 0.1m$

Fig. 3: Number of objects whose poses were correctly classified by the baseline methods (BFw/oR, OUR-CVFH) and PERCH, for different definitions of 'correct pose'. Results reproduced from [2].

methods such as deep neural networks to focus the search on promising solutions, while preserving theoretical guarantees on the solution quality. In the context of the 2016 Amazon Picking Challenge, we used a superpixel-based convolutional neural network trained on RGB images to constrain the search space for PERCH. Our current approach to the 2017 Amazon Robotics Challenge is based on using a fully convolutional network to obtain pixelwise labels, which in turn will serve as a heuristic for PERCH.

IV. Experimental Results.

To evaluate the performance of PERCH for multi-object recognition and pose estimation in challenging scenarios where objects could be occluding each other, we pick the occlusion dataset described by Aldoma et al. [11] that contains objects partially touching and occluding each other. The dataset contains 3D CAD models of 36 common household objects, and 22 RGB-D tabletop scenes with 80 object instances that vary only in yaw and translation. We compared PERCH with two baselines: the first is the OUR-CVFH descriptor [8] that was designed to be robust to occlusions. We trained the OUR-CVFH pipeline by rendering 642 views of every 3D CAD model from viewpoints sampled around the object. Our second baseline is an ICP-based optimization one, which we refer to as Brute Force without Rendering (BFw/oR). Here, we slide the 3D model of every object in the scene over the observed point cloud, and perform a local ICP-alignment at every step. The set of object poses that have the best total fitness score is taken as the solution.

Figure. 3 provides a quantitative comparison with the baselines. The latter shows the number of correct poses produced by each of the methods (out of 80 objects), for the following definition of 'correct pose': a predicted pose (x, y, θ) for an object is considered correct if $\|(x, y) - (x_{\text{true}}, y_{\text{true}})\|_2 < \Delta t$ and $\text{ShortestAngularDifference}(\theta, \theta_{\text{true}}) < \Delta\theta$. We see that PERCH consistently dominates the baselines for different definitions of correct pose, with the improvements being most significant when very accurate poses are desired (translation error under 1 cm). While PERCH without using any discriminative guidance is computationally expensive (on the order of minutes for a scene), using it in conjunction with discriminative heuristics reduced computation to under a minute for the 2016 Amazon Picking Challenge. We also note that PERCH does not require any training time unlike other discriminative methods, making it suitable for cases when there is not enough training time available.

References

[1] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman, "Lessons from the Amazon Picking Challenge," *arXiv preprint arXiv:1601.05484*, 2016.

[2] V. Narayanan and M. Likhachev, "PERCH: Perception via Search for Multi-Object Recognition and Localization," in *ICRA*. IEEE, 2016.

[3] A. E. Johnson and M. Hebert, "Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes," *PAMI*, vol. 21, no. 5, pp. 433–449, 1999.

[4] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D Registration," in *ICRA*. IEEE, 2009.

[5] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *European conference on computer vision*. Springer, 2010.

[6] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3D Recognition and Pose Using the Viewpoint Feature Histogram," in *IROS*. IEEE, 2010.

[7] A. Aldoma, M. Vincze, N. Blodow, D. Gossow, S. Gedikli, R. B. Rusu, and G. Bradski, "CAD-model Recognition and 6DOF Pose Estimation using 3D Cues," in *ICCV Workshops*. IEEE, 2011.

[8] A. Aldoma, F. Tombari, R. B. Rusu, and M. Vincze, "OUR-CVFH–Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation," in *DAGM*, 2012.

[9] Z.-C. Marton, D. Pangercic, N. Blodow, and M. Beetz, "Combined 2D–3D Categorization and Classification for Multimodal Perception Systems," *IJRR*, 2011.

[10] Y. Chen and G. Medioni, "Object Modeling by Registration of Multiple Range Images," in *ICRA*, 1991.

[11] A. Aldoma, Z.-C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. B. Rusu, S. Gedikli, and M. Vincze, "Point Cloud Library," *IEEE Robotics & Automation Magazine*, vol. 1070, no. 9932/12, 2012.

[12] 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 *ACCV*, 2013, pp. 548–562.

[13] P. Wohlhart and V. Lepetit, "Learning Descriptors for Object Recognition and 3D Pose Estimation," in *CVPR*, 2015.

[14] A. Tejani, D. Tang, R. Kouskouridas, and T.-K. Kim, "Latent-class Hough Forests for 3D Object Detection and Pose Estimation," in *ECCV*, 2014, pp. 462–477.

[15] M. Schwarz, H. Schulz, and S. Behnke, "RGB-D Object Recognition and Pose Estimation Based on Pre-trained Convolutional Neural Network Features," in *ICRA*. IEEE, 2015.

[16] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal Deep Learning for Robust RGB-D Object Recognition," in *IROS*, 2015.

[17] M. R. Stevens and J. R. Beveridge, "Localized Scene Interpretation from 3D Models, Range, and Optical Data," *Computer Vision and Image Understanding*, 2000.

[18] Z. Sui, O. C. Jenkins, and K. Desingh, "Axiomatic particle filtering for goal-directed robotic manipulation," in *IROS*.

[19] V. Narayanan and M. Likhachev, "Discriminatively-guided Deliberative Perception for Pose Estimation of Multiple 3D Object Instances," in *Robotics: Science and Systems*, 2016.

[20] V. Narayanan, S. Aine, and M. Likhachev, "Improved Multi-Heuristic A* for Searching with Uncalibrated Heuristics," in *Eighth Annual Symposium on Combinatorial Search (SoCS)*, 2015.