Using prioritized relaxations to locate objects in points clouds for manipulation

Robert Truax, Robert Platt, John Leonard

Abstract— **This paper considers the problem of identifying objects of interest in laser range point clouds for the purposes of manipulation. One of the characteristics of perception for manipulation is that while it is unnecessary to label *all* objects in the scene, it may be very important to maximize the likelihood of correctly locating a *desired* object. This paper leverages this and proposes an approach for locating the most likely object configurations given an object parameterization and a point cloud. While many other approaches to object localization need to explicitly associate points with hypothesized objects, our proposed method avoids this by optimizing relaxations of the likelihood function rather than the exact likelihood. The result is a simple, efficient, and robust method for locating objects that makes few assumptions beyond the desired object parameterization and with few parameters that require tuning.**

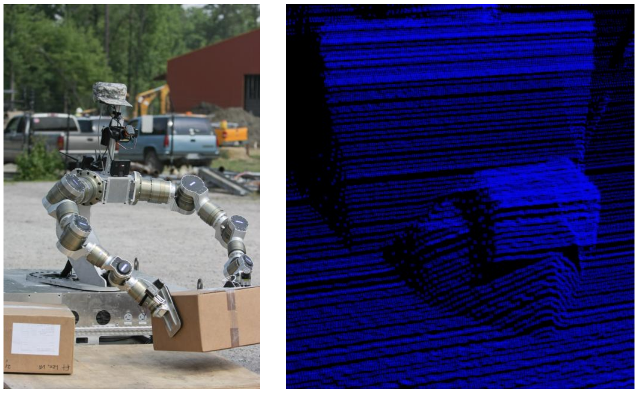


Fig. 1: Scenario in which a robot must estimate the pose and shape of modeled objects with high confidence.

I. Introduction

Range sensing has recently become the sensor modality of choice for robot manipulation and assembly systems. Popular choices include time-of-flight systems, “nodding” LIDAR systems, or stereo depthmap systems that produce an accurate and dense array of points in space. In the context of robot manipulation, the goal of perception is to locate and characterize a candidate object for manipulation. This is a bit different than the standard objective of perception. Our objective is not to label the entire image or range scan, but rather to identify one or more occurrences of an object of interest in the input data, and to verify that the objects that are localized do in fact correspond to actual occurrences of the objects of interest. An additional consideration is the need to be robust to noise and occlusion. For example, in the box stacking task (shown in Figure 1), we are less interested in locating all boxes in the stack as we are in being sure that a box identified and localized in the input data really corresponds to a box in the world. In addition, we want an approach that does not need to be adjusted depending upon the expected amount of noise in the data.

Finding objects in 3D range data is a fundamental problem in robotics and computer vision, and accordingly, the literature on the 3D object recognition problem is vast. Examples of representative seminal works include (1) Besl and Jain [1], [2], who developed techniques for object recognition from range images, creating a system that analyzed surface curvature and classified critical points; (2) Viola and Wells, who developed gradient descent techniques for alignment of 3D models with 2D and 3D data sets based on maximizing the mutual information; and (3) Besl and McKay, who developed a method for registration of 3-D shapes based on iterative closest point (ICP) matching. Our work in this paper differs from these works, and from the majority of mainstream object recognition research, in that our approach is situated in the task of mobile manipulation in semi-structured environments. Specifically, our goal is to enable a mobile manipulator to identify and locate parameterized objects with a minimum set of assumptions about the environment to enable subsequent manipulation (e.g., loading boxes of various shapes and sizes onto pallets). In this context, recent work aimed to support mobile manipulation is most relevant.

A common approach for finding objects in 3D range data is to use a bottom-up procedure where planes or curves are located in the data first and then the planes or curves are fit to known models. For example, Vosselman *et al.* adopt this approach for reconstructing ground plans based on aerial point cloud data. After extracting roof planes using a combination of a 3-D Hough transform and least squares plane estimation, the resulting planes are fit to *a priori* roof models [3], [4]. Liu *et al.* take a similar approach to model the objects encountered by a mobile robot: EM is used to locate planes in the scene and the planes are mapped onto a known set of polygons [5]. Rusu *et al.* apply a similar approach for the task of locating objects in a household kitchen environment. Planes in the scene are found using MLESAC [6]. Plane parameters are estimated using regression, neighboring planes are merged, and the resulting planes are matched to a given polyhedral model [7].

Rather than matching point data to models known *a priori*, an alternative approach is to reconstruct object models directly from point data. If only primitive planes or curves are of interest, then RANSAC-based methods have been demonstrated to be very effective [6], [8]. If more complex objects are to be modeled, RANSAC methods can be used

The authors are with the Computer Science and Artificial Intelligence Laboratory at MIT. Email: {robtruax,rplatt,jleonard}@csail.mit.edu

to find the subset of points that match a desired object. Given these points, a superquadric can be used to model the geometry of the object surface [9]. Alternatively, Marton *et al.* propose using RANSAC methods to identify shape primitives and then modeling objects as arbitrary constellations of primitives [10].

A key difficulty with the compositional methods described above is that while shape primitives are used to locate complex objects, there is no flow of information in the other direction: object information does not inform the search for primitives. In manipulation, we are often interested in deciding which grasp action has the greatest chance of succeeding. If object models are given, then we must decide which modeled object in the scene is *most likely*: our objective is to find the object model(s) that maximize the likelihood of the point cloud data. However, in order to maximize the likelihood of matching points in the cloud to object surfaces, it is necessary to associate each hypothesized object with a subset of points. For example, Liu *et al.* solve the data association and the optimization problems simultaneously using expectation maximization (EM) [5]. The need to associate points with objects makes object localization more complex because a bad association will cause localization to fail.

Instead, this paper explores relaxations of the exact likelihood function. Although optimizing the exact likelihood function requires an association between points and hypothesized objects, it turns out that relaxations of the likelihood function exist that do not require this data association. This paper considers two relaxations that have joint optima near optima of the exact likelihood. The first requires that no point in the cloud be contained within the region occluded by an object (given the location of the ranging system). The second relaxation counts the expected number of points on the object surface. Both of these relaxations may be optimized for an arbitrary number of objects by optimizing for each individual object independently over all points in the cloud. In an approach reminiscent of Blake and Zisserman's graduated non-convexity algorithm [11], both relaxations are optimized simultaneously by ascending the gradient of the second relaxation in the tangent space (the null space) of the gradient of the first relaxation. The result is that we optimize for the number of points on the surface of objects *subject to* keeping the correct occlusion relationships. The paper develops this approach and illustrates its performance in the context of a robotic box lifting task.

II. Problem statement

Our basic problem is to locate objects in the environment based on a set of range measurements. The range measurements are produced by a sensor such as a "nodding" LIDAR, stereo cameras, or time-of-flight ranging systems. The range measurements are typically expressed as a collection of spatially-located points known as the "point cloud." Let P be the domain of the points in the point cloud. The range sensor locates a collection of points, $Z \subseteq P$, that can be assumed to lie on the surfaces of objects in the environment. Our objective is to identify those parts of the point cloud where certain modeled target objects are located.

Let X be the configuration space of an object. X may include position, orientation, and shape parameters. For example, a cuboid in space has a nine-dimensional parameterization: the position in $\mathbb{R}^3$, the orientation in $SO(3)$, and the face extents in $\mathbb{R}^3$. The possibility of different types of objects can be accommodated by optimizing for different object parameterizations at once. One way to view the object localization problem is in terms of maximizing the likelihood of a function defined over possible object configurations. Assume that all points in the point cloud lie on the surface of the object in question. Then the maximum likelihood configuration of the object is:

$$x_{max} = \arg\max_{x \in X} P(Z \subseteq \mathscr{B}(x)), \tag{1}$$

where $\mathscr{B}(x) \subseteq P$ denotes the surface (boundary) of the object, x.

Unfortunately, since the point cloud will typically include points that lie on the surface of many objects and not just on the surface of the object in question, using Equation 1 requires a preprocessing step where points belonging to the object are first segmented from the rest of the points in the cloud. As an alternative, one might create a hypothesis space that describes *all* objects that affect the point cloud. For example, let

$$H = \{X^k : 1 \leq k \leq K\}$$

be the set of all multi-object hypotheses (up to a maximum number of objects, K), where for simplicity we assume that all objects in the cloud belong to the class described by X. The localization problem is now expressed by:

$$h_{max} = \arg\max_{h \in H} P(Z \subseteq \mathscr{B}(h)), \tag{2}$$

where $\mathscr{B}(h) = \cup_{x \in h} \mathscr{B}(x)$ describes the surfaces of all objects in h. Since h is a hypothesis that accounts for *all* points in the cloud, it is no longer necessary to segment the cloud.

Unfortunately, the large dimensionality of H suggests that the maximization problem will be computationally intractable unless some structure can be found in $P(Z \subseteq \mathscr{B}(h))$. For example, one might express $P(Z \subseteq \mathscr{B}(h))$ in terms of a single object, $P(Z \subseteq \mathscr{B}(x))$, and attempt to optimize the likelihood of all objects independently. Assuming that each point in the point cloud is independent of every other point, we have:

$$\begin{aligned} P(Z \subseteq \mathscr{B}(h)) &= \prod_{i=1}^{|Z|} P(z_i \in \mathscr{B}(h)) \\ &= \prod_{i=1}^{|Z|} [1 - P(z_i \notin \mathscr{B}(h))] \\ &= \prod_{i=1}^{|Z|} \left[1 - \prod_{x \in h} P(z_i \notin \mathscr{B}(x)) \right], \end{aligned} \tag{3}$$

Unfortunately, this equation shows that the likelihood does not factor into object likelihoods that may be optimized independently. For example, it is impossible to optimize this

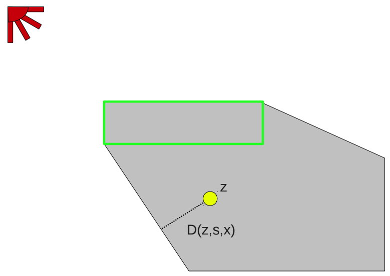


Fig. 2: Calculation of the distance, $D(z_i, s, x)$, of point z from the region not occluded by the object. The object hypothesis is illustrated in green; the region occluded by the object with respect to the source is shaded.

function by independently ascending gradients, $\frac{\partial P(Z \subseteq \mathscr{B}(h))}{\partial x}$, associated with each object, $x \in h$, because the gradients depend on the parameters associated with all objects, not just object x.

III. Relaxations of the likelihood function

Rather than attempting to optimize Equation 3 directly, this paper proposes a combination of two relaxations. The relaxations have a structure that makes the likelihood optimization more tractable.

A. Primary relaxation: occlusion regions

Our first relaxation of Equation 3 is based on the following: a necessary condition for all points to be on the surface of hypothesized objects is that there are no points that are contained in a region occluded by hypothesized objects. Let $\mathscr{O}(s,h) \subseteq P$ be the region of point space that is occluded by objects contained in hypothesis h when the range sensor source is located at s (see Figure 2). Because of the "necessary" aspect of the above condition, we have:

$$\begin{aligned} P(Z \subseteq \mathscr{B}(h)) &= P(Z \subseteq \mathscr{B}(h)) \\ &\leq P(Z \cap \mathscr{O}(s,h) = \emptyset). \end{aligned} \tag{4}$$

This relaxation can be factored according to the set of objects, $x \in h$. Assume that the extent of the objects comprising h do not overlap so that the occlusion or non-occlusion of a point by a particular object has no bearing on the likelihood of its occlusion by another object. In this case, we have:

$$P(Z \cap \mathscr{O}(s,h) = \emptyset) = \prod_{x \in h} P(Z \cap \mathscr{O}(s,x) = \emptyset). \tag{5}$$

Since Equation 5 can be expressed as a product of independent likelihoods between zero and one, each factor can be optimized independently. That is: if each factor, $P(Z \cap \mathscr{O}(s,x) = \emptyset)$, in Equation 5 is minimized, then $P(Z \cap \mathscr{O}(s,h) = \emptyset)$ must be at a minimum as well.

Once $P(Z \cap \mathscr{O}(s,h) = \emptyset)$ has been factored, calculating each $P(Z \cap \mathscr{O}(s,x) = \emptyset)$ is straightforward. Assuming independent observations, we have:

$$P(Z \cap \mathscr{O}(s,x) = \emptyset) = \prod_{i=1}^{|Z|} P(z_i \notin \mathscr{O}(s,x)). \tag{6}$$

$P(z_i \notin \mathscr{O}(s,x))$ is the likelihood that point z_i is *not* included in the region occluded by x. Let $D(z_i, s, x)$ be the distance of point z_i to the region not occluded by the object (see Figure 2). Then, the likelihood that a point, z_i, is occluded by object x is proportional to:

$$P(z_i \notin \mathscr{O}(s,x)) \sim \mathscr{N}\left(D(z_i,s,x), W_z\right), \tag{7}$$

where $\mathscr{N}(x-\mu, \Sigma)$ evaluates the PDF of the Gaussian with a deviation from the mean of $x-\mu$ and covariance Σ. In the next section, we will optimize Equation 6 using gradient ascent. The gradient of Equation 6 is:

$$\begin{aligned} &\frac{\partial P(Z \cap \mathscr{O}(s,x) = \emptyset)}{\partial x} \\ &= \sum_{z_i \in Z} \frac{\partial \mathscr{N}\left(D(z_i,s,x), W_z\right)}{\partial x} \prod_{z_j \in (Z - z_i)} \mathscr{N}\left(D(z_j,s,x), W_z\right). \end{aligned} \tag{8}$$

B. Subordinate relaxation: point percentage

The relaxation described in Equation 4 is clearly insufficient to completely localize objects in the point cloud. Our approach is to introduce a subordinate relaxation that operates in the regions where the first relaxation is optimized. Essentially, a new likelihood function is defined implicitly by the intersection of the first and second relaxations.

The subordinate relaxation simply counts the expected percentage of the points in the cloud on the surfaces of the hypothesized objects,

$$\begin{aligned} P(Z \subseteq \mathscr{B}(h)) &= \prod_{i=1}^{|Z|} P(z_i \in \mathscr{B}(h)) \\ &\leq \frac{1}{|Z|} \sum_{i=1}^{|Z|} P(z_i \in \mathscr{B}(h)), \end{aligned} \tag{9}$$

(one may verify that this inequality is correct). Since the expected number of points on all hypothesized objects is just the sum of the expected number of points on each object, we have:

$$\sum_{i=1}^{|Z|} P(z_i \in \mathscr{B}(h)) = \sum_{x \in h} \left[\sum_{z_i \in Z} P(z_i \in \mathscr{B}(x)) \right]. \tag{10}$$

The form of Equation 10 demonstrates that we can optimize this relaxation by optimizing the term in brackets for each object, $x \in h$. Let $B(z_i, x)$ denote the distance between the non-occluded surfaces of x and the point z_i. Then, we have:

$$P(z_i \in \mathscr{B}(x)) = \mathscr{N}(B(z_i,x), W_z),$$

where we have again assumed that point in the cloud have a covariance of W_z caused by measurement noise. The gradient of the term in brackets in Equation 10 is:

$$\frac{\partial \sum_{z_i \in Z} P(z_i \in \mathscr{B}(x))}{\partial x} = \sum_{z_i \in Z} \frac{\partial \mathscr{N}(B(z_i,x), W_z)}{\partial x}. \tag{11}$$

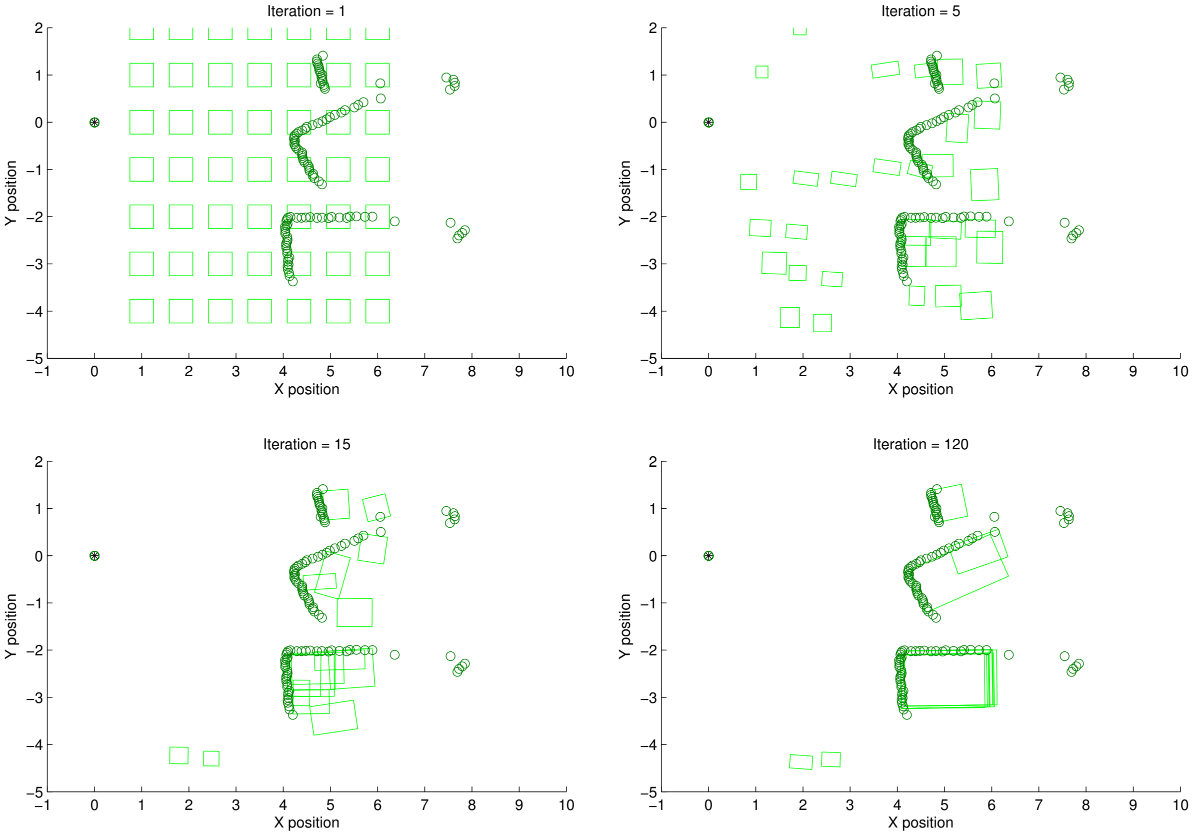


Fig. 3: Illustration of localization in two dimensions for a collection of rectangles. Green boxes represent hypotheses, laser return points are circles, and the laser source is located at the star in the upper left corner.

While neither Equations 5 nor 10 are sufficient on their own to localize objects well, the intersection is. Our approach is to use constrained gradient ascent where the second-priority relaxation is optimized in the tangent space of the gradient of the primary relaxation. Let $\frac{\partial R_1}{\partial x}$ be the gradient in Equation 8 and let $\frac{\partial R_2}{\partial x}$ be the gradient in Equation 11. We ascend the gradient of relaxation 2 in the tangent space of the gradient of relaxation 1:

$$\frac{\partial R_{12}}{\partial x} = \frac{\partial R_1}{\partial x} + \frac{\partial R_2}{\partial x} N_1, \tag{12}$$

where

$$N_1 = \frac{\frac{\partial R_1}{\partial x}^T \frac{\partial R_1}{\partial x}}{\frac{\partial R_1}{\partial x} \frac{\partial R_1}{\partial x}^T}$$

is the tangent space projection matrix for $\frac{\partial R_1}{\partial x}$.

C. Planar illustration

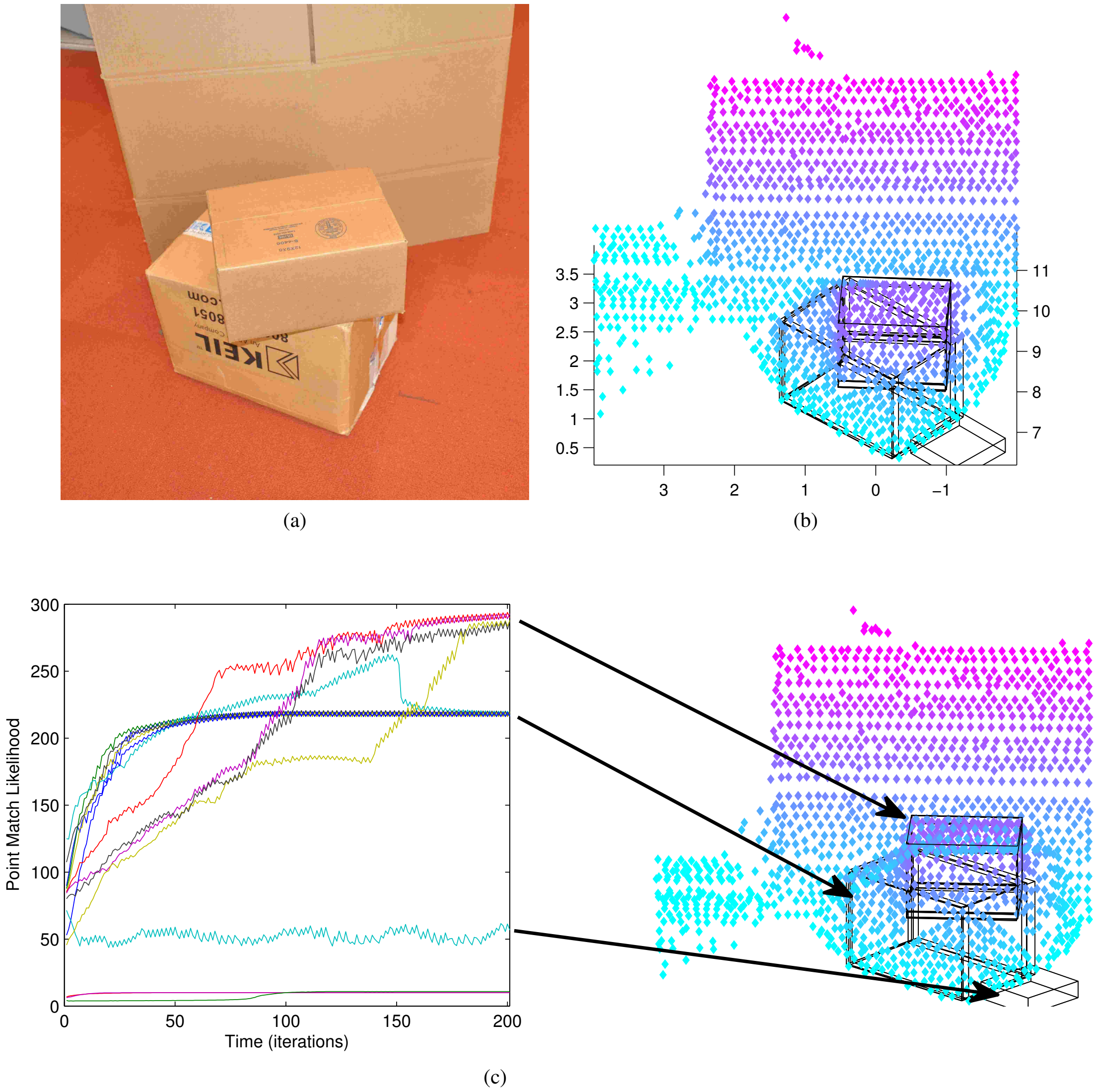
Figure 3 illustrates the localization process in the plane. The point cloud (the green circles in Figure 3) was collected indoors using a Hokuyo UTM laser scanner mounted in a fixed configuration in front of a collection of three rectangular boxes. The objective is to locate a maximum likelihood rectangular objects in the scene. Since this is a planar problem, the hypothesis space for a single rectangle is five dimensional (position in $\mathbb{R}^2$, orientation in $SO(2)$, and the extents of the two sides in $\mathbb{R}^2$),

$$x = (p_x, p_y, d_x, d_y, \theta)^T,$$

where p_x and p_y are the coordinates of the center of the rectangle, d_x and d_y are the extents of the rectangle sides, and θ is the orientation of the rectangle. As explained in Section III-A, the distance function, $D(z_i, s, x)$ in Equation 7, is the nearest distance to a region not occluded by the object hypothesis. $D(z_i, s, x)$ is calculated with respect to the sides of the rectangle hypothesis visible to the laser source and the occlusion boundary lines (see Figure 2).

Figure 3 shows four frames from the planar localization example. The starting sampling of hypothesis assumes nothing about the actual locations of the objects. It begins with uniformly spaced samples of identical small size. As the iterations progress, relaxation 1 (the high-priority relaxation) causes the hypotheses to migrate toward regions within the occluded regions in the point cloud. Hypotheses that do not migrate quickly enough shrink because the decreasing side extents decreases the number of points in the hypotheses occlusion regions. Hypotheses are removed if their likelihoods fall below a threshold. Once within the occluded regions in the point cloud, hypotheses slowly migrate toward configurations that maximize the number of points on their sides.

Preprint submitted to 2011 IEEE International Conference on Robotics and Automation. Received September 15, 2010.



(a)

(b)

(c)

Fig. 4: For a scene (a) a final state of the hypotheses (b) is illustrated. The time history of the point match likelihoods for for each hypothesis are plotted in (c) with arrows to their corresponding locations in space. Notice a spurious maxima below the floor but with low likelihood when compared to other hypotheses.

Notice that some of the hypotheses converge to maxima that do not correspond to object locations. This reflects the fact that optimizations of the relaxed likelihood functions can find *any* local minimum. Not all optima of the combined relaxations correspond to objects in the scene. However, maxima that do not match the measurements well are given a low likelihood and can be removed.

IV. Experiments

The approach proposed in this paper was used to locate maximum likelihood rectangular boxes in a manipulation scenario.

A. Platform

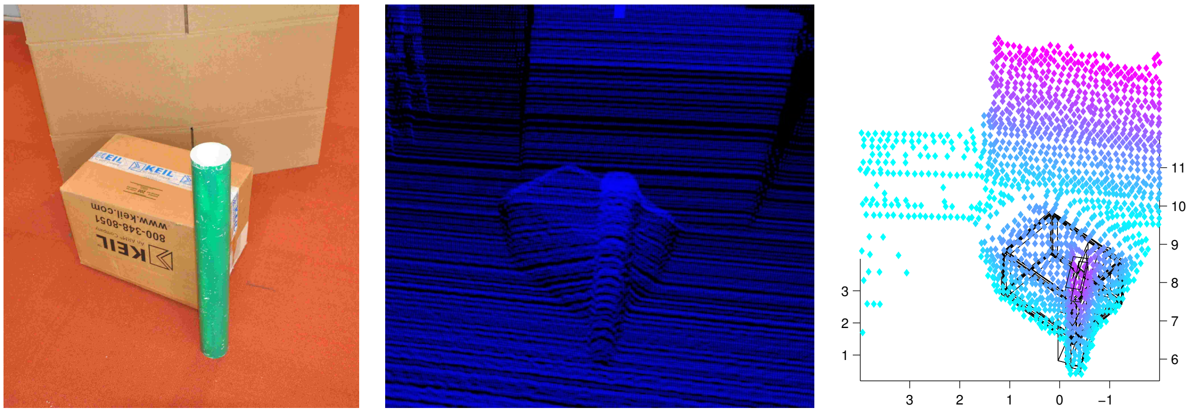
A nodding Hokuyo UTM laser was mounted to the “head” of the manipulator (see Figure 1). Each “sweep” of the nodding head took approximately five seconds and collected approximately 100*k* points. After restricting points to a bounding box around the work area, including removing many of the ground plane points, and subsampling so that no two points were closer than about a centimeter, the point cloud was reduced to about 1800 points. While our algorithm should be unaffected by including more points, this greatly reduced the computational load.

B. Spatial localization

In general, the configuration space for a cuboid in three dimensions is nine dimensional: position of the cuboid center in $\mathbb{R}^3$, orientation of the cuboid in $SO(3)$, and side extents in $\mathbb{R}^3$. We simplified the math in our experiments by constraining orientation to rotation about the vertical axis. The resulting free degrees of freedom of the cuboid are:

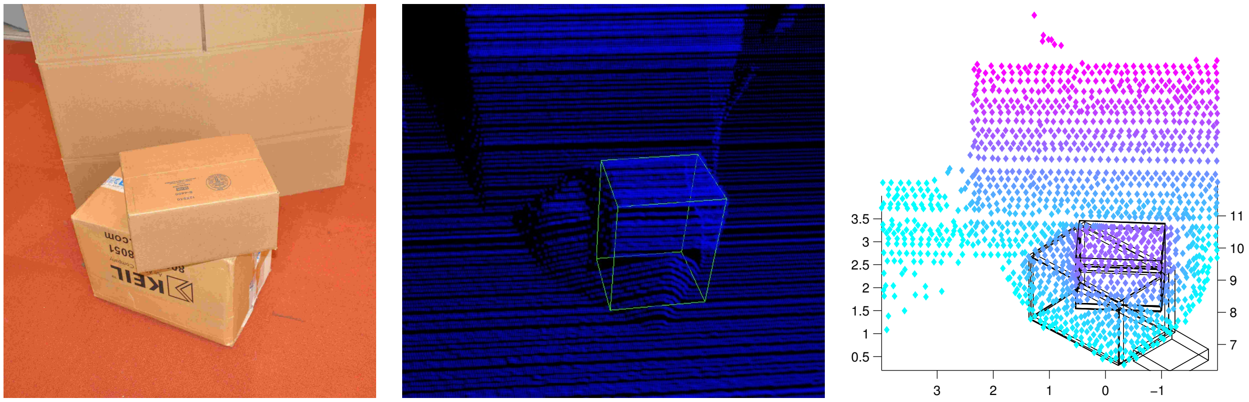
$$x = (p_x, p_y, p_z, d_x, d_y, d_z, \theta)^T,$$

where p_x, p_y, and p_z, are the coordinates of the position of the center of the cuboid, d_x, d_y, and d_z, are the extents of



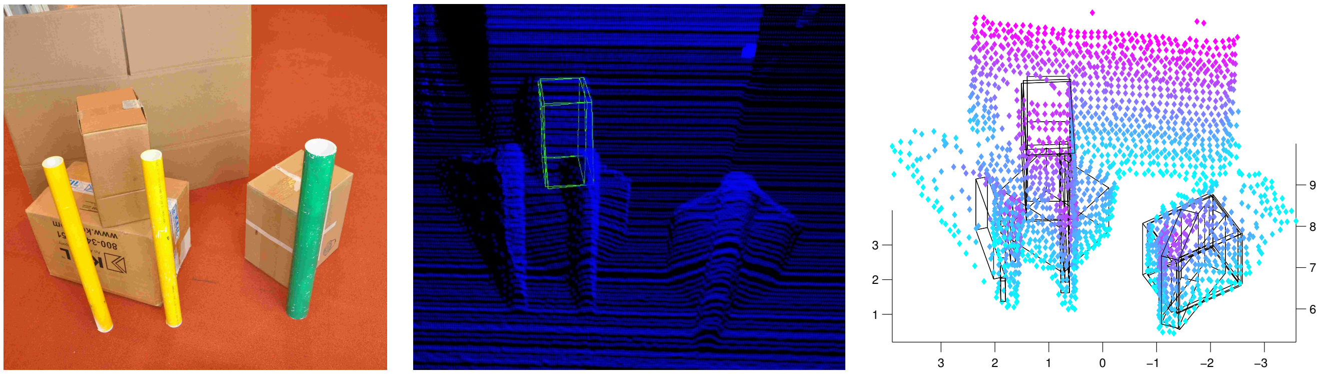
(a) (b) (c)

Fig. 5: A single cylindrical obstacle in front of a box (a) causes a failure for the RANSAC method (b) but is located in our method (c).



(a) (b) (c)

Fig. 6: Two boxes where one edge aligns with the face of another (a) causes a bad hypothesis with the RANSAC method (b) but proper dimensions are found in our method (c).



(a) (b) (c)

Fig. 7: A crowded scene with many boxes and obstacles (a) interferes with all but one box in the RANSAC method (b) but poses minimal challenge to our method (c).

the sides, and θ is the orientation about the vertical axis.

Figure 4 illustrates the process of localizing a pair of stacked boxes (shown in Figure 4(a)) in the seven dimensional object configuration space. The process begins with 15 uniformly seeded object hypotheses. Figure 4(b) illustrates the hypotheses (shown in black) in their final configuration after they have converged to the maxima of the relaxed likelihood functions. Notice that the hypotheses cluster around the true locations of the two boxes. Figure 4(c) illustrates the optimization process in terms of hypothesis likelihood over time. Initially, all hypotheses have similarly low likelihoods. Over time, the likelihood of the boxes rises and ultimately clusters in two high-likelihood configurations.

C. Comparison with a compositional method

As discussed in the introduction, the primary localization methods used in robotic manipulation scenarios are compositional methods where modeled objects are located by first identifying shape primitives using methods such as RANSAC, Hough transforms, EM, *etc.* and then identifying the objects in terms of those primitives. Since our implementation of the robotic porter (Figure 1) was based on the Robot Operating System (ROS) [12], maintained by Willow Garage, we compared our performance with what we were able to achieve using compositional methods found in the Point Cloud Library (PCL) [13]. The general approach is related to what is proposed proposed by Rusu [7]: a variant of RANSAC is used to find planes in the image, the planes are merged and the overall parameters of the plane are estimated using regression, and boxes are hypothesized to exist in places where the appropriate combination of planes is found.

We presented both systems with localization scenarios that were difficult because of obstacles or confusing combinations of planes. Figures 5, 6, and 7 illustrates the comparison. The three figures show an image of the scenario on the left, the box located by our compositional method in the center (the green outline superimposed on the point cloud), and the objects found by our method on the right. In Figure 5, the compositional method is confounded by the cylinder that prevents the robot from finding planes sufficiently nearby to be considered a box. In Figure 6, the compositional method finds only a single box that extends to the floor because it overestimates the size of the planes. Finally, in Figure 7, compositional methods fail to find enough planes to locate all boxes.

One way to improve performance of the compositional method on the above might be to modify the plane-finding parameters so that a larger number of planes are found. However, this would risk finding boxes where none exist. In our experimental use of the compositional method during manipulation, we found it unsafe to raise these thresholds too high for fear that the robot would find and attempt to grasp objects were none really existed.

One potential failure of our approach in the scenarios above is that the occluding poles in the image are misinterpreted as boxes. This is a consequence of the optimization process. Since all points in the cloud are assumed to belong to a modeled object, the poles are interpreted as boxes. Of course, since the poles are not really boxes, they are given a low likelihood and can be identified as poor box candidates.

D. Effect of seeding on optimization

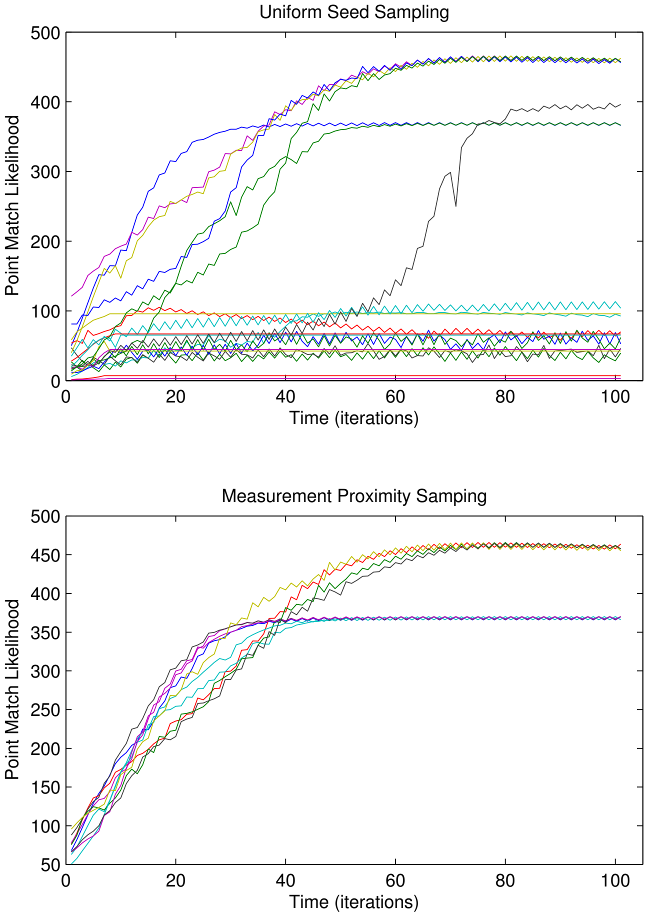


Fig. 8: A comparison of the two types of seeding of hypotheses used for experiments. A uniform sampling produces many more samples who find spurious local maxima, but still slowly find the correct box locations. The measurement proximity seeding produces fewer spurious local maxima, and requires fewer total hypotheses to characterize the environment.

The fact that our optimization process begins with randomly chosen hypotheses suggests that performance can be improved by starting with more intelligent estimates. A number of heuristics are possible: hypotheses may be initially seeded more densely in regions where the point cloud is more dense; hypotheses may be seeded more densely close to the laser source; shape primitives (such as those localized by RANSAC) could be used to inform the set of initial hypotheses. This work explores the possibility of the first and second options above: seeding hypotheses more densely in dense regions of the cloud and giving preference to hypotheses close to the laser source. Figure 8 illustrates the comparison. Both seeding approaches find the two primary maxima (where boxes are located). However, the seeding heuristic focuses the search on the primary maxima and avoids spurious maxima.

V. Conclusion

This paper has presented a novel method to determine the pose of objects of interest in a point cloud. A hierarchy of relaxations uses knowledge of objects to inform how to process raw data. This approach is robust to occlusions, and requires fewer heuristic and parameter tuning necessary in RANSAC based processes. Apart from a parameterized model of the object of interest, our algorithm only requires two variables: the standard deviations of expected noise in each relaxation function. Though boxes were used throughout the examples shown, the algorithm is generalizable to any sufficiently parameterized object.

VI. Future Work

Although our method has proven to be very successful in the box manipulation scenarios explored in this paper, it needs to be demonstrated in the context of more complex object localization tasks. We expect that the approach will generalize well across large classes of objects. For example, we expect that this will be an effective way of locating all chairs or tables in a room based on a roughly parameterized geometry. Another area for future exploration is to move from a fixed-step gradient ascent algorithm to more powerful optimization techniques.

References

[1] P. Besl and R. Jain, “Three-dimensional object recognition,” *ACM Computing Surveys (CSUR)*, vol. 17, no. 1, p. 145, 1985.

[2] P. Besl and R. Jain, “Invariant surface characteristics for 3D object recognition in range images* 1,” *Computer vision, graphics, and image processing*, vol. 33, no. 1, pp. 33–80, 1986.

[3] G. Vosselman, B. Gorte, and G. Sithole, “Recognising structure in laser scanner point clouds,” *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 46, no. 8/W2, pp. 33–38, 2004.

[4] G. Vosselman and S. Dijkman, “3D building model reconstruction from point clouds and ground plans,” *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, no. 3/W4, pp. 37–43, 2001.

[5] Y. Liu, R. Emery, D. Chakrabarti, W. Burgard, and S. Thrun, “Using EM to learn 3D models of indoor environments with mobile robots,” in *Intl. Conf. on Machine Learning (ICML)*, 2001.

[6] P. Torr and A. Zisserman, “MLESAC: a new robust estimator with application to estimating image geomety,” *Computer Vision and Image Understanding*, vol. 78, pp. 138–156, 2000.

[7] R. Rusu, Z. Marton, N. Blowdow, M. Dolha, and M. Beetz, “Towards 3d point cloud based object maps for household environments,” *Robotics and Autonomous Systems*, vol. 56, pp. 927–941, 2008.

[8] R. Schnabel, R. Wahl, and R. Klein, “Efficient ransac for point-cloud shape detection,” *Computer Graphics Forum*, vol. 26, no. 2, pp. 214–226, 2007.

[9] G. Biegelbauer and M. Vincze, “Efficient 3D object detection by fitting superquadrics to range image data for robots object manipulation,” in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2007.

[10] Z. Marton, L. Goron, R. Rusu, and M. Beetz, “Reconstruction and verification of 3d object models for grasping,” in *Proc. of the 14th Intl. Symposium on Robotics Research*, 2009.

[11] A. Blake and A. Zisserman, *Visual Reconstruction*. MIT Press, 1987.

[12] Willow Garage, “ROS.” http://www.willowgarage.com/pages/software/ros-platform, Sept. 2010.

[13] R. B. Rusu, “PCL :: Point cloud library.” http://rbrusu.com/pcl-point-cloud-library.html, Sept. 2010.