Active Recognition and Manipulation of Objects Based on Shape Primitives

Dirk Holz, Matthias Nieuwenhuisen, David Droeschel, Jörg Stückler, and Sven Behnke[1]
Alexander Berner and Reinhard Klein[2]
[1]Autonomous Intelligent Systems, [2]Computer Graphics
Computer Science Institute, University of Bonn, Germany

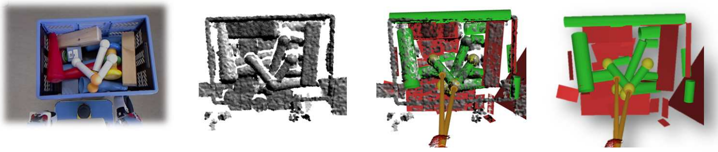
universitätbonn

1. Introduction

Framework for recognizing and grasping objects based on shape primitives:

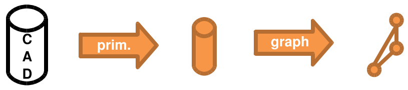
- Efficient recognition of objects composed of shape primitives
- Grasp planning using shape graphs
- View planning to improve recognition and grasping performance

2. Object Recognition

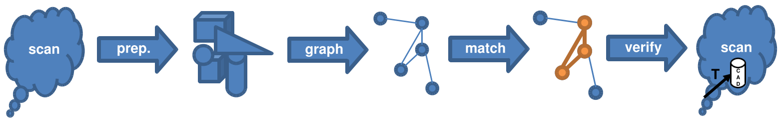


example scene | 3D scan | detected object | shape primitives

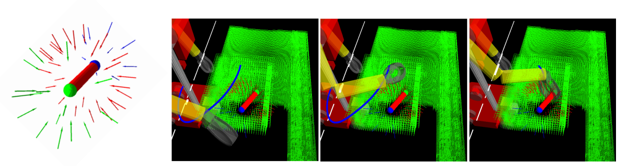
Object Model Acquisition:
We extract shape primitives from CAD models using an efficient RANSAC algorithm [1]. We extend the shape detection method to incorporate 2D primitives such as circles. A graph captures spatial relations between shapes.



Object Recognition:
In a 3D scan, we detect shape primitives and establish a spatial relation graph. We apply probabilistic sub-graph matching [2] to find partial overlaps of the model graph with the scene graph. For the best matches, we extract object hypotheses and poses. We verify the hypotheses: Each object is required to have sufficient overlap in the scan and must not occlude measured points.

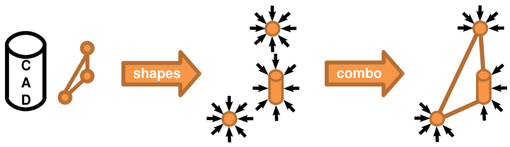


3. Grasp and Motion Planning

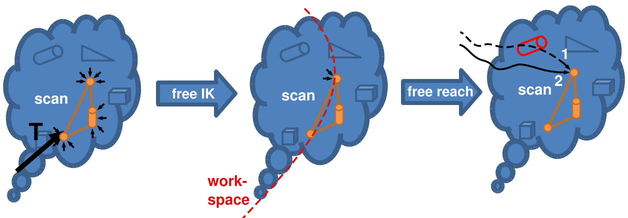


sampled grasps on shape primitive | reaching motion | grasp

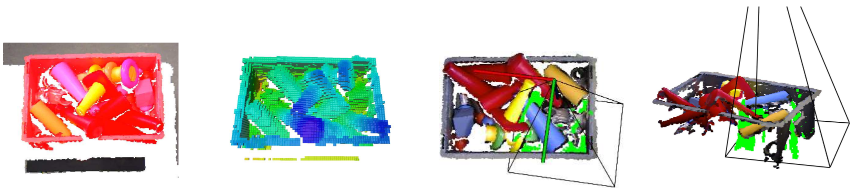
Offline Phase:
We precompute the set of grasps on an object that are feasible irrespective of its context in a scene. For each shape primitive in the object compound, we uniformly sample a set of grasps that approach the shape in normal direction onto its surface. We then consider collisions of grasps with all shapes in the object.



Online Phase:
In the actual scene context, we first find all grasps among the precomputed set that are reachable within the workspace of the robot. We also check for collisions with other objects in the scan. We rank the remaining grasps according to simple reachability criteria such as the height of the grasp. In the final stage, we successively evaluate the reachability of the best grasp by planning and executing reaching motions until a reachable grasp has been found.



4. Active Recognition



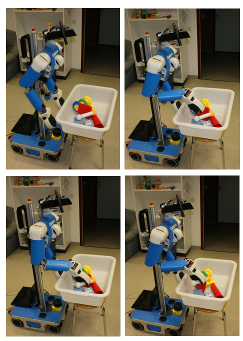
point cloud | voxel map | next-best-view (green: discovered volume)

In a bin picking scenario, e.g., many occlusions may hinder the recognition of objects due to dense packing and limitations of the field-of-view. We develop view planning to fuse information from multiple views. In a next-best-view approach, we favor view poses that discover unknown space. In addition, we will incorporate recognition value provided by object recognition into the utility measure. We represent the volume of interest in a voxel map in which we determine unknown space by ray-casting.

5. Results

We evaluate our method for object recognition and grasping in simulation and with a real robot.

In a reproducible simulation setup, an object is placed in random poses in a transport box. In ten trials, our approach requires in avg. 14.9 sec to choose a non-colliding grasp in the robot's workspace (on Intel Core i7-Q720). Until an executable grasp has been found, the planning time of reaching motions takes in avg. 2.45 sec. Planning a single reaching motion requires 0.45 sec in average. In all ten test runs, the robot succeeded to grasp the object.



References

[1] R. Schnabel, R. Wahl, and R. Klein. Efficient RANSAC for point-cloud shape detection. In Computer Graphics Forum, 2007.

[2] R. Schnabel, R. Wessel, R. Wahl, and R. Klein. Shape recognition in 3D point-clouds. In Proc. of the 16th Int. Conf. in Central Europe on Computer Graphics, Visualization, and Computer Vision, 2008.

This research has been funded by the FP7 ICT-2007.2.2 project ECHORD (grant agreement 231143) experiment ActReMa.