Towards Reliable Grasping and Manipulation in Household Environments

Matei Ciocarlie, Kaijen Hsiao, E. Gil Jones, Sachin Chitta, Radu Bogdan Rusu and Ioan A. Şucan

Abstract We present a complete software architecture for reliable grasping of household objects. Our work combines aspects such as scene interpretation from 3D range data, grasp planning, motion planning, and grasp failure identification and recovery using tactile sensors. We build upon, and add several new contributions to the significant prior work in these areas. A salient feature of our work is the tight coupling between perception (both visual and tactile) and manipulation, aiming to address the uncertainty due to sensor and execution errors. This integration effort has revealed new challenges, some of which can be addressed through system and software engineering, and some of which present opportunities for future research. Our approach is aimed at typical indoor environments, and is validated by long running experiments where the PR2 robotic platform was able to consistently grasp a large variety of known and unknown objects. The set of tools and algorithms for object grasping presented here have been integrated into the open-source Robot Operating System (ROS).

1 Introduction and Related Work

As algorithms for autonomous operation are constantly evolving, complete robotic platforms with the ability to combine perception and action are starting to explore the rich set of applications available in unstructured environments. As part of this effort, we present an approach to reliable grasping and manipulation of household objects. With the directional goal of enabling autonomous applications in human settings, we have focused on the following aspects:

Matei Ciocarlie, Kaijen Hsiao, E. Gil Jones, Sachin Chitta and Radu Bogdan Rusu
Willow Garage Inc., Menlo Park, CA, USA.
e-mail: {matei,hsiao,gjones,sachinc,rusu}@willowgarage.com

Ioan A. Şucan
Rice University, Houston, TX, USA. e-mail: isucan@rice.edu

- the ability to grasp and manipulate both known and unknown objects in a reliable and repeatable manner. The combination of object recognition algorithms and extensive pre-computed knowledge bases has the potential to extend a robot's capabilities and the range of achievable tasks. However, a robot operating in a human environment is likely to also be faced with situations or objects never encountered before.
- reliable operation in a wide range of scenarios, requiring robustness to real-world problems such as imperfect calibration or trajectory following.
- safe operation in a wide variety settings. In particular, a manipulation task should be collision free for both the robot itself and the object that it is manipulating.

Achieving this type of functionality has required the integration of multiple modules, each charged with its own subtask, such as:

- scene segmentation and object recognition;
- collision environment acquisition and maintenance;
- grasp planning for both known and unknown objects;
- collision-free arm motion planning;
- tactile sensing for error correction during grasp execution.

It is important to note that each of these goals can be considered a research area in its own right. Furthermore, in addition to the technical challenges posed by each sub-task, their integration reveals the interplay and reciprocal constraints between them. One of the main features of this study is that it reports on an integrated system, allowing us to share the lessons learned regarding the importance of each component as well as the potential pitfalls of combining them into a complete platform.

The integration of the multiple modules presented in this study was done using the Robot Operating System (ROS). In addition, the complete architecture is included in the current ROS distribution[1]. We hope that it will prove a useful tool both to researchers aiming to improve manipulation capabilities (who can focus on one or more particular components of our architecture) and those attempting to build towards more complex applications (who can use the complete system as a building block).

There are a number of complete robot platforms that have demonstrated combined perception and action to manipulate objects autonomously in human environments, such as [7, 14, 13, 6, 9, 15, 1]. Preliminary results based on our approach were also presented in [11]. In this study we expand on our previous efforts by adding a number of components, such as grasp planning for a wide variety of objects, tactile feedback during task execution, *etc*.

[1] The complete codebase used for achieving the results presented in this paper is available as part of the ROS C Turtle distribution. See `http://www.ros.org` for general ROS information and `http://www.ros.org/wiki/pr2_tabletop_manipulation_apps` for documentation of the relevant code packages.

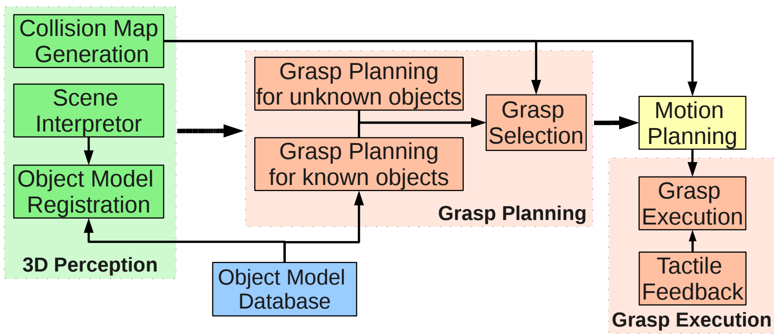

Fig. 1 Top: The PR2 robot platform. Bottom: Our system architecture.

2 Technical approach

The overall structure of our system is shown in Fig. 1; in this section we will provide additional details on each individual component. The hardware used for implementation is the PR2 personal robot, which has an omni-directional base and two 7-DOF arms. It is also equipped with a tilting laser scanner mounted on the head, two stereo cameras, a fixed laser scanner mounted on the base, and a body-mounted IMU. Encoders provide position information for each joint. The end-effector is a parallel jaw gripper equipped with fingertip capacitive sensor arrays, each consisting of 22 individual cells.

2.1 Semantic Perception and Object Segmentation

The sensory input to our system is in the form of 3D point cloud data that (on the PR2 robot) comes from laser range sensors and stereo cameras. The first step consists of processing this data to obtain semantic information, with the goal of segmenting a complete image of the environment into individual graspable objects. As household objects in domestic environments are usually found on flat planar surfaces, we exploit this structure and obtain additional semantic information by computing a planar fit of the surface that provides support for the objects. Euclidean clustering on the points above the planar support provides the list of graspable objects in the scene.

In addition, our system attempts to match each segmented object against a database of known 3D models, using an iterative technique similar to the ICP algorithm [2]. Our current matching algorithm operates in a 2-DOF space, and can therefore be applied for situations where partial pose information is known (*e.g.* rotationally symmetrical objects such as cups, glasses or bowls resting on a flat surface). If the match between a segmented point cloud and a model in the database exceeds a certain quality threshold, the object is assumed to be recognized. Fig. 2 shows an example of a complete scene with semantic information, including the table plane and both recognized and unknown objects.

2.2 Collision Environment

In order to operate the robot safely in the environment, the system depends on a comprehensive view of possible collisions. The semantic perception block provides information about recognized objects and the table plane while data from a wider view sensor, like the tilting laser on the PR2, is used to generate a binary 3D occupancy grid in the arm's workspace. The occupied cells near recognized objects are filtered to take advantage of the higher resolution information available from the semantic perception component. The combined collision environment consists of oriented bounding boxes for occupied cells, box primitives for the dominant table plane and for bounding boxes around unrecognized point clusters, and triangle meshes for the robot's links and any recognized objects (Fig. 2). The collision environment is used in grasp selection, to perform collision-aware inverse kinematics, as well as in motion planning, to check arm trajectories against possible collisions.

2.3 Grasp Planning and Selection

The goal of the grasp planning component is, for every object segmented from the environment, to generate a list of possible grasps, each consisting of a gripper pose relative to the object (we note that, for more dexterous hands, a grasp would also have to contain information regarding finger posture). The current version of our grasping planning component provides separate methods for creating such a list, based on whether the object is recognized as one of the models in our database or treated as an unkown point cluster.

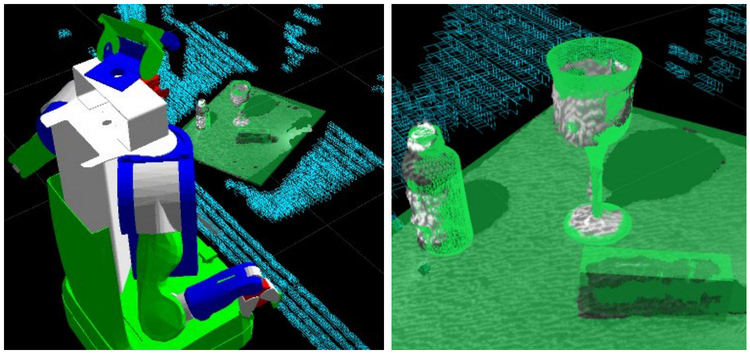

Fig. 2 Scene perception result. Note unknown obstacles (blue) and obstacles with semantic information, such as the table and the objects (green). Recognized objects have complete 3D meshes superimposed.

All the known objects in our model database are annotated with large sets of stable grasp points, pre-computed using the *GraspIt!* simulator [8]. In our current release, the definition of a stable grasp is specific to the gripper of the PR2 robot, requiring both finger pads to be aligned with the surface of the object and further rewarding postures where the palm of the gripper is close to the object as well. Our grasp planning tool uses a simulated annealing optimization, performed in simulation, to search for gripper poses relative to the object that satisfy this quality metric. For each object, this optimization was allowed to run over 4 hours, resulting in an average of 600 grasp points for each object. An example of this process is shown in Fig. 3.

Grasps for unrecognized objects are computed at run-time from 3D sensor data, using heuristics based on both the overall shape of the object and its local features. The intuition behind this approach is that many human-designed objects can be grasped by aligning the hand with the object principal axes, starting from either above or to the side of the object, and trying to grasp it around the center. If the center is not graspable, any other part that fits inside the hand can be attempted, along similar guidelines. Grasps found according to these principles are then ranked using a small set of simple feature weights, including the number of sensed object points that fit inside the gripper, distance from object center, *etc*. A number of examples are shown in Fig. 3, and additional information about this component can be found in [5].

Once the list of possible grasps has been populated, execution proceeds in a similar fashion regardless of which grasp planner was used. Each of the grasps in the list is tested for feasibility in the current environment; this includes collision checks for both the gripper and the arm against potential obstacles, as well as generation of a collision-free arm motion plan for placing the gripper in the desired pose.

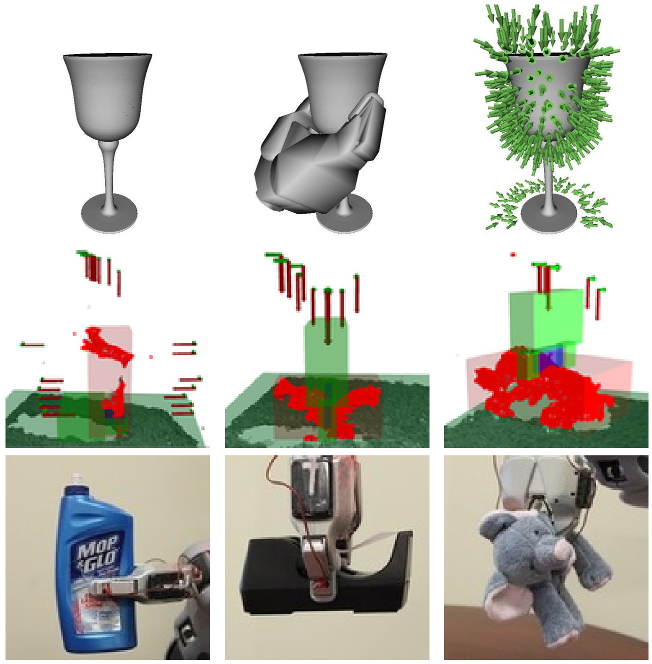

Fig. 3 Grasp Planning. Top row: grasp planning in a simulated environment for a known object (the object model, a simulated grasp and the complete set of pre-computed grasps). Middle row: grasp planning from 3D sensor data for novel objects. Bottom row: grasp execution for novel objects.

2.4 Motion Planning

Sampling-based planning combined with collision-aware inverse kinematics is used to plan motions to the desired poses for grasping, lifting and placing objects. Paths are planned to a *pre-grasp* location that is offset from the desired grasp location, followed by a straight line path in Cartesian space to the grasp pose. Objects that have been grasped are attached to the robot model to avoid collisions with the environment during transport. These models are also padded to provide a buffer zone while executing motion plans. The planner generates joint-space paths that are further rocessed using *short-cutting* techniques to obtain smooth, collision-free spline trajectories conforming to joint velocity and acceleration limits.

2.5 Trajectory Execution and Reactive Grasping

The resulting trajectories are executed by a controller that continuously monitors, and aborts the execution if there is a possibility of collisions at any point, forcing the system to re-plan a path to the desired goal. During the final stages of the grasp, reactive behaviors are used to achieve the desired result despite small errors in object localization and perceived shape. The first reactive behavior uses information from the tactile sensors on the fingertips to maneuver around the object when unexpected contacts are detected during the approach. The second behavior accounts for cases where one fingertip comes into contact with the object earlier than the other, by executing a *compliant grasp* that coordinates the motion of the arm and the gripper so that the object is not pushed away while the gripper is closed. The final behavior accounts for grasps that are likely to be unstable by adjusting the position of the end-effector to achieve a grasp where contacts are seen at the centers of the fingertip sensor arrays.

3 Experiments and Results

We validated our approach by carrying out experiments for grasping and placing objects found in typical household environments. Our model database contains 3D shape information for a subset of the objects we used (*e.g.*, bowls, cups, cans, *etc.*). The rest of the objects (*e.g.*, stuffed toy animal, stapler, tape dispenser, wire coil, *etc.*) are not part of the dataset, and were therefore treated as unknown models.

Our first set of experiments focused on grasping 30 novel objects, using the online grasp planner working directly on 3D sensor data. We attempted a set of 68 grasping tasks distributed over the objects in our set; Fig. 3 shows a number of examples from this set. When using open-loop grasp execution, the object was successfully grasped and lifted from the table 60 out of 68 times. When tactile-based reactive grasp execution was used, the success rate increased to 66 out of 68. More details on the grasping experiments for unknown objects, showing grasp planning results for each object and including an analysis of the effect of tactile-based error correction, can be found in [5].

Our second set of experiments were intended as a comprehensive test of the complete system. Here, the task was to continuously move objects between different locations on a table, in a fully autonomous loop. The objects were presented to the robot in groups of three, in order to maintain the level of euclidean separation required by our semantic perception algorithms. The robot would then select an object, pick it up from the table and place it down in a new location, while avoiding collisions with the environment. The presence of additional vertical obstacles, like the one shown in Fig. 2, considerably limited the free workspace available to the robot. We note that, during execution, the location of the objects on the table was often not preset by the user, but was instead the result of previous pick-and-place operations. Tactile-reactive behaviors were used on grasps of unknown objects, but not on the stored grasps of known objects.

During this experiment, the robot was to perform 2 pick-and-place operations on each of 15 objects, for a total of 30 operations. 29 of 30 operations succeeded; one object was inadvertently collided with while executing a different task. 3 of the operations required 2 grasp attempts each, for a total of 32 grasp attempts (29+3).

10 of the 15 objects were in the model database; in 15 of 20 detections they were correctly recognized. In addition, 2 unknown objects were mistakenly classified as database models. However, of these 7 recognition errors, only 2 resulted in grasp failures: 3 objects were grasped successfully even though they were not recognized (being treated as novel objects), and 2 were recognized as models that were close enough in shape to the true object to allow task completion. Finally, one object was dropped despite correct recognition. Overall, 25 of 32 detections were correct, and 29 of 32 grasps succeeded.

This behavior was also demonstrated in a live environment at the 2010 Intl. Conf. on Robotics and Automation. Over three days of execution, the system often operated continuously without grasp failures for periods of time ranging from 30 minutes to 1 hour, and successfully grasped novel objects (such as shoes, apples, keys, or hats) supplied by spectators.

4 Conclusions and Future Directions

In this work, our goal was to achieve reliable grasping of typical household objects. The approach we present was successful in performing pick and place tasks on a wide set of objects and our quantitative results show that consistent performance can be achieved from a set of basic behaviors. At its core, our approach was to break down a complex task into manageable components for which we designed simple but reliable solutions. Another key to the success of our approach is the tight integration of information from a variety of sensors, in a manner that exploits the particular strengths of each sensor to create a consistent view of the environment. The use of local reactive behaviors to quickly correct globally-planned motions has also proven critical to designing a fast, robust system.

An interesting question regards the combination of semantic perception with a pre-computed knowledge base for manipulation tasks. The grasp planning component presented in this study is split into two modules, with one assuming no prior knowledge about the grasped object, and the other relying on a correct result from the recognition and pose detection module. Instead of choosing between these options, we are currently developing a probabilistic framework that combines them, aiming to find consensus on how the object should be grasped by using the information from multiple recognition results, each with an associated confidence level, as well as raw sensor data.

Our system is currently object-centric, requiring explicit reasoning about the target object. In contrast, other grasp planning algorithms bypass object segmentation altogether and operate on a complete image of the environment [13]. Other examples of grasp planning based on approximate object recognition from partial sensor data can also be found in [4]. We will continue to explore such options, as we believe that a robot operating in an unstructured environment should be able to handle unknown scenarios while still exploiting high-level perception results and prior knowledge when these are available.

Our results using the architecture presented in this study underline the importance of tactile sensing for manipulation tasks. We have already discussed the reactive grasping module, which uses tactile information to correct for errors during grasp execution. In addition, we are currently integrating a novel grasp controller that uses information from the tactile sensors together with other types of proprioceptive data to adapt the grasping force to the material properties of the grasped object. A complete study describing this component is currently under review [10].

The results presented in this study mainly focus on the task of grasping an object. For higher level applications, reliable and collision-free object transport are just as important. For example, a grasp that can successfully lift the object from a table might fail when faced with different disturbances during transport. To alleviate this risk, we are currently integrating constraint-based motion planning aiming to preserve the orientation of the grasped object during arm movement; this component will also be required for transporting objects under imposed orientations, as in the case of liquid-filled open containers. We are also integrating active monitoring methods, using the stereo cameras mounted on the robot head to track the path of the arm and detect potential fast-moving obstacles.

Our current system does not support highly cluttered scenes that prevent euclidean object segmentation, or unknown objects requiring complex or situation-specific grasps (*e.g.*, a pitcher to be grasped by the handle). The registration-based recognition module is limited to 2-DOF pose detection; we expect to obtain better results by using feature-based recognition [12] and combining 3D point clouds with intensity images. The motion planning component would also benefit from algorithms that compute a path through free space all the way into gripper contact with the target object [3]. These aspects will be the focus of future work.

References

1. Balasubramanian, R., Xu, L., Brook, P., Smith, J., Y., M.: Human-guided grasp measures improve grasp robustness on physical robot. In: IEEE Intl. Conf. on Robotics and Automation (2010)
2. Besl, P., McKay, N.D.: A method for registration of 3-D shapes. IEEE Trans. on Pattern Analysis and Machine Intelligence **14**(2) (1992)
3. Şucan, I., Kalakrishnan, M., Chitta, S.: Combining planning techniques for manipulation using realtime perception. In: IEEE Intl. Conf. on Robotics and Automation (2010)
4. Goldfeder, C., Ciocarlie, M., Peretzman, J., Dang, H., Allen, P.: Data-driven grasping with partial sensor data. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (2009)
5. Hsiao, K., Chitta, S., Ciocarlie, M., Jones, E.G.: Contact-reactive grasping of objects with partial shape information. In: Workshop on Mobile Manipulation, IEEE Intl. Conf. on Robotics and Automation (2010)
6. Inamura, T., Okada, K., Tokutsu, S., Hatao, N., Inaba, M., Inoue, H.: HRP-2W: A humanoid platform for research on support behavior in daily life environments. Robotics and Autonomous Systems (2009)
7. Jain, A., Kemp, C.: EL-E: an assistive mobile manipulator that autonomously fetches objects from flat surfaces. Autonomous Robots (2010)
8. Miller, A., Allen, P.K.: GraspIt!: a versatile simulator for robotic grasping. IEEE Robotics and Automation Magazine **11**(4) (2004)

9. Partlitz, C., Hagele, M., Klein, P., Seifert, J., Dautenhahn, K.: Care-o-bot 3 - rationale for human-robot interaction design. In: Intl. Symposium on Robotics (2008)
10. Romano, J.M., Hsiao, K., Niemeyer, G., Chitta, S., Kuchenbecker, K.J.: Human-inspired robotic grasp control with tactile sensing. Under Review (2011)
11. Rusu, R., Sucan, I., Gerkey, B., Chitta, S., Beetz, M., Kavraki, L.: Real-time perception-guided motion planning for a personal robot. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (2009)
12. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (FPFH) for 3d registration. In: IEEE Intl. Conf. on Robotics and Automation (2009)
13. Saxena, A., Wong, L., Ng, A.: Learning grasp strategies with partial shape information. In: AAAI Conf. on Artificial Intelligence (2008)
14. Srinivasa, S., Ferguson, D., Weghe, M., Diankov, R., Berenson, D., Helfrich, C., Strasdat, H.: The robotic busboy: Steps towards developing a mobile robotic home assistant. In: Intl. Conf. on Intelligent Autonomous Systems (2008)
15. Stuckler, J., Schreiber, M., Behnke, S.: Dynamaid, an anthropomorphic robot for research on domestic service applications. In: European Conf. on Mobile Robots (2009)