Evaluation of an Assistive Manipulation System

Abraham Shultz, Andreas Ten Pas, Robert Platt, Holly Yanco

Abstract— **This paper describes a complete system, composed of a robot arm mounted to an assistive mobility device, a control system for that arm, and a proposed user interface for selecting objects for the arm to grasp for the user. The ability of the arm to grasp various objects is assessed, and problems with manipulating novel objects are discussed.**

I. Introduction

The system described in this paper integrates a grasp planning system, a user interface for designating objects, and a manipulator to produce a system that can grab user-specified objects.

Overall, the hardware and task approach for this system is similar to the work of Kemp *et al.* [**?**]. However, that work assumed a user with normal upper body control, and focused on the perception and object retrieval elements of the problem. The system described in this paper is intended to be used in experiments with a population including people with cerebral palsy, spinal trauma, and similar conditions. The purpose of these experiments is to guide the construction of user interfaces for assistive robots that permit people with motor disabilities to control the system.

People with motor disabilities will be solicited to use the arm and provide feedback on its utility for activities of daily living (ADLs). These users have limited arm mobility, and so limited ability to control the laser pointer. The UI for moving the laser pointer can be varied to permit people with different disabilities to aim the laser. For example, if the laser is mounted to a pan-tilt unit, a person with limited upper limb mobility could use a joystick to move the laser, and a user with quadriplegia could use a sip-and-puff switch to control the movement of the laser.

To assess the performance of such a system, it would be desirable to compare the system to other visually-guided grasping systems. Further, the eventual goal of the system is as a testbed for human-robot interaction (HRI) for assistive robotics. As a consequence, the metrics and results used in assessing it should be transferable to performing assistive operations, such as retrieving objects for a user. For the purposes of human-robot interaction, the emphasis from the user is more heavily on task performance than on motion accuracy. If the robot is off by several millimeters when grasping something, it only matters if the discrepancy causes the robot to be unable to perform the desired task.

Some proposed benchmarks are overly broad. The RoboCup@Home tasks have been proposed as a benchmark for the performance of assistive robots [**?**]. The RoboCup@Home tasks include tracking a person, reacting to verbal commands, and fetching a specified object. As a superset of grasping, these tasks provide more benchmark information than is required for comparing grasp planning and execution on a user-specified object. At the same time, the RoboCup@Home tasks are too specific to be applicable to our system. For example, objects to retrieve and actions to perform are specified to the robot using speech input, and so any robot which does not use speech input cannot perform RoboCup@Home tasks.

More specific proposals for standards have the problem that they do not cover the complete space of grasping systems, and so a novel system may find no applicable standard. Benchmarks have been proposed for grasping based on the smallest and largest enveloping grasps and grasp resistance to force [**?**]. These are benchmarks for a hand only, not for a grasping system, and only apply to planar grasping, not to 3D object grasping.

Other proposed standards or benchmarks make assumptions that are not applicable to our system. Solving a Rubik's Cube has also been proposed as a benchmark for assistive robotic systems, because the solution requires both visual perception of the cube's state and dexterous manipulation to solve it [**?**]. However, solving a Rubik's Cube is a two-handed manipulation problem. For a single manipulator, the Rubik's Cube is difficult to manipulate, and so this benchmark is only applicable to dual-gripper systems or those capable of reasoning about which objects in the environment can be used as fixtures.

One method of checking grasp quality evaluates grasps with a cost function that accounts for how likely they are to cause a collision between the gripper and other objects in the area [**?**]. However, this method requires a model of the objects, which is not available for novel objects or deformable objects. In the absence of a full model of the object, another system uses partial shape information registered to a large set of shapes with known grasps to find grasps with partial sensor data [**?**]. The database used is the Princeton Shape Benchmark (PSB), which contains 1814 models of objects [**?**]. By *matching* the shape of objects rather than *recognizing* specific objects, the grasp planner attempts to generalize from the grasps known for objects from the PSB to novel objects. In simulation, the system finds force closure grasps on those models from the PSB that are amenable to analysis of the grasp quality. As the authors point out, the analysis used is not a physically realistic simulation. It also depends on the presence of models roughly corresponding to the partially-sensed object. Deformable objects are not as amenable to model matching, and so cannot be evaluated by this system.

The Yale-CMU-Berkley (YCB) Object and Model set [**?**] also provides a large set of models of objects. The YCB objects were chosen to be widely available, and provide a variety of shapes, sizes, weights, textures, and rigidities for grasp testing. The YCB object set also includes items from a variety of tests proposed in the literature on upper body thereapy and rehabilitation for humans, with the hope that the existing protocols for evaluating human ability could be applied to robots as well.

However, despite the intention of providing objects with varying properties, all but two of the objects in the data set are rigid. The exceptions are a bundle of nylon rope, and a plastic chain, which is rigid at the scale of an individual link. Rather than providing a protocol for testing grasping systems, the YCB provides a format for developing and sharing protocols. The methods section of this paper is structured according to the YCB guidelines.

The OpenGRASP Benchmark [?] combines simulated scenarios and robots with models of realistic objects and interface code that permits testing of grasping and grasp planning algorithims in simulation. It appears that OpenRAVE, the robot simulation platform that underlies OpenGrasp, has the ability to simulate sensors, and so provide simulated sensing data from within the simulated environment. Our system is intended for tests with human research subjects from who may have cognitive difficulties, and so not understand that the simulation is intended as a model of the real world.

Two benchmarks have been proposed for the combination of vision systems and manipulators, but are not applicable to the work in this paper because they both assume that objects are rigid [?], [?]. These approaches score grasps by detecting the amount that the grasped object moves relative to the gripper, because a solid grasp does not permit the object to slip. Measuring relative motion only makes sense with rigid objects, as e.g. a stuffed toy could completely change shape when lifted, despite being firmly grasped. One of the proposed benchmarks ([?]) also requires stereo pairs of images, rather than point clouds, and so is bound to a specific form of visual input. When the Visgrab benchmark was proposed, this was a reasonable assumption, because point cloud cameras were expensive and rare. Our system uses commonly-available RGB-D cameras, and so works with point clouds rather than stereo images.

II. Method

A. Task Description

The task is lifting objects under the direction of a user and moving them to a fixed location. It is intended to serve as a precursor to retrieving objects that the user wants to move to a specific location, such as a shopping basket while using the system in a supermarket.

B. Setup Description

add measurements of objects

The object set consists of a selection of household objects. The primary criterion in selecting objects is that at least one of their dimensions is small enough to fit within the gripper of the arm when it is open, and be grasped by it when it is closed. The gripper configuration of the Baxter arm can be changed to have variable grip widths, but for this work, it was set to a maximum open size of 7cm and a closed size of 3cm. Any object smaller than 3cm in every dimension could not be picked up, and so was not tested. Similarly, objects wider than 7cm in every dimension were not tested.

Because the goal of the system is to facilitate user testing for robotic assistance with activities of daily living (ADLs), the objects are common household objects, rather than those that might be more commonly found in a laboratory setting. Objects commonly associated with ADLs have been suggested [?], [?]. Our object set contains objects from many of the proposed classes of ADLs, especially food preparation and housekeeping.

The objects are a stuffed toy lobster, a rocket-shaped air bulb duster, a can of black pepper, a container of white pepper, a container of lavender, a blue foam ball, a white spray bottle, a blue spray bottle, a container of coffee creamer, a stuffed toy drill, a stuffed toy screw, a box of coffee stirrers, a plastic packet of agar flakes, a bar of soap in a box, a computer mouse, a nozzle from a vacuum cleaner, and a pocket-sized packet of tissues.

The tissues, lobster, rocket, toy drill, toy screw, and packet of agar are all deformable to some degree. The box of coffee stirrers and blue ball are not as soft as the other objects, but do flex when pinched by the gripper. The containers of spices, spray bottles, container of coffee creamer, soap box, mouse, and vacuum-cleaner nozzle are rigid against the forces applied by the gripper.

The objects were placed 75cm from the camera, on a table 46cm from the ground. The arm base is 69cm from the ground as mounted on the scooter. Each object was placed in a specific orientation for testing, as listed in table **??** and shown in figure **??**.

Fig. 1: The objects as they were oriented for testing

C. Robot Description

The system consists of a single robotic arm attached to a mobility scooter designed for use by people with partial mobility disabilities. The arm is one of a pair that were originally attached to a Rethink Robotics Baxter robot. It is a 7-DOF arm with a 1-DOF (parallel) gripper.

The arm is mounted to a Golden Avenger mobility scooter. The scooter is designed for users with full mobility, but limited strength and endurance. The arm and the computer controlling it are powered from the scooter batteries, so the system as a whole is mobile.

The image stream and point clouds used for perception in the system are provided by a Primesense Carmine RGB-D camera, mounted near the base of the arm. It is expected that addition of a second RGB-D camera will increase the ability of the system to detect grasps, by providing a more complete point cloud, so a mount point for a second camera is available. However, for the work described in this paper, only a single camera is used.

Fig. 2: View of the front of the scooter, showing the arm and camera.

All of the software developed for the system uses the ROS framework [?]. For the test described in this paper, three computers were used. The computer mounted to the scooter runs a modified version of the Baxter Research and Education SDK. The modifications are needed to allow the system to run without the torso, head, or right arm of the Baxter robot. A second computer ran the perception and grasp selection nodes. The third computer was used to control the pan-tilt unit to move the targeting laser. For mobile use, the second and third computers will be replaced by a laptop that will be mounted to the scooter.

The pan-tilt unit for the targeting laser stands in for the human user in this test. The pan-tilt unit moves the laser point to select the target object. For each object, the laser was scanned over its surface to target different points on the object. Each scan location was approximately 1.5cm from the previous one, but curvature in the surface of the objects could contribute to increasing or decreasing that distance.

The laser point is detected on the surface of the object by comparing successive RGB image frames from the RGB-D camera to detect motion, and confirming that the moving area is the correct color and size for the laser pointer dot. Because the pan-tilt unit can hold the laser more still than a human, the laser blinks on and off to increase its visibility to the laser detection ROS node.

The point cloud is then segmented using Locally Convex Connected Patch (LCCP) segmentation, and each point is labeled with an identifier for the segment that it belongs to [?]. In order to restrict the grasp discovery process to only find grasps on the desired object, the laser pointer dot is located in the image, and the corresponding point is found in the point cloud. The label for the point that the laser is targeting is used to select those points in the same LCCP segment as the laser dot, and those points are presented to the grasp planner as the object to grasp [?]. All of the other points in a volume of space around the selected segment are also provided to the grasp planner, to permit the planner to find grasps that do not collide with nearby objects.

In order to locate potential grasping locations in the point cloud, an approach based on the geometry of the Baxter gripper is used [?]. Because the gripper used is a parallel gripper, good locations to grasp are restricted by certain constraints:

1) The gripper and the robot's forearm must not collide with any point in the point cloud.
2) The grasp point must be in the plane between the robot's fingers.
3) The plane orthogonal to the direction of minimum curvature of the surface to be grasped must be parallel to the plane between the robot's fingers.

For a sampling of the points in the point cloud for the target object, these criteria are assessed. If the point matches the criteria, then it is considered a potential grasp location. The potential grasps are then checked to determine if they are antipodal. An antipodal grasp is one where the line between to contact points on the objects surface remains within the friction cones of the contact points. The friction cone is defined by the forces, normal and frictional, that a point contact can apply to a surface. If the line between two contact points remains within the friction cone of both points, then forces can be applied in opposition from those points, pinching the object.

However, since the point cloud is likely partial, due to occlusion, the relationship of the gripper to the grasped surface may be unknown, and so it is impossible to determine if the grasp is antipodal. Instead, a support vector machine (SVM) trained on partial point clouds is used to assess the grasps. The SVM data was labeled by determining if a given grasp was antipodal using a point cloud from multiple sensors, but trained using the data available from only one sensor at a time. As a consequence, the SVM was trained to determine if a grasp was antipodal using only a partial point cloud, effectively learning whether the hidden side of the object permits an antipodal grasp.

After the possible grasps are found, they are scored by a selection algorithm that takes into account how easy to reach the grasps are, how easy it is to perform the grasp using the gripper, and how far the arm has to travel to reach the grasp location. Grasps that are unreachable are rejected, leaving only grasps that are collision-free, antipodal, and reachable by the arm.

The chosen grasps are then sorted by angle off of vertical and then by relative height of the grasp point, with higher locations being preferred. The intent of this sorting is to have the arm approach objects from directly above the object. Since the next step of moving the object is to lift it up, approaching from above causes the motion towards the object and the lifting motion to occur along the same path. If the desired object had an obstacle above it, preventing more vertical approaches, the most vertical grasp could still be from the side.

It had been hoped that because approaching from the top matches the way humans grab many objects, the resulting motion of the robot would be more understandable to human observers. Unfortunately, this is not the case, and the arm still engages in visually striking contortions to reach some grasps. Constraining the inverse kinematic (IK) solver to prevent these motions is beyond the scope of this work, but is an active area of research [?].

The arm then moves to a position such that it can approach the object with the fingers of the gripper parallel to the arm's motion, and moves in to grasp the object and closes the gripper. After grasping the object, the arm moves 2cm vertically and re-closes the gripper, which is to say, sends a second close command to the gripper without opening it. If the object slipped as the arm started to lift it, this re-gripping can tighten the robot's grip while the object is still partly supported by the surface it was resting on.

After regripping the object, the arm moves 10cm vertically.

If the object is still in the gripper when this move is complete, the grasp is considered successful. For objects with variable centers of gravity, such as a bottle with water in it, the grasp may fail later in the process due to the center of gravity moving. The overall score for an object is the number of successful grasps divided by the number of attempts. Finally, the arm returns to the starting position, carrying the object.

D. Procedure

The execution of the test is largely automated, but progress from stage to stage while grasping the object is controlled by a human operator. The operator's primary responsibility is to control the robot through a test script and reset the object after each attempted grasp.

The operator places the object to be grasped, and aims the laser pointer at a location on the object. The motion of the laser point is controlled by a pair of Dynamixel actuators arranged as a pan-tilt platform. The purpose of the pan-tilt platform is to allow the operator to scan the laser over the surface of the object in a regular fashion, targeting different points on the surface of the object. The operator then confirms to the test script that the object is ready, which commands the robot to attempt to grasp the object. The robot attempts the grasp, and the test script then prompts the operator to indicate if the grasp is a success or failure. If the grasp attempt fails, the operator notes the cause of the failure, and resets the object. For the purposes of this study, errors were in one of five categories.

1) Bad points: The point on the object was incorrectly detected as being part of a different area of the point cloud, or was not detected.
2) Narrow grasps: The grasp was planned for an area of the object that was too narrow for the robot to grasp.
3) Knocked object: The robot knocked over the object while positioning itself for the grasp.

If the grasp attempt succeeds, the operator confirms the success and resets the object. The operator then moves the laser point to the next location on the object and continues the test. After and attempt has been made for all laser points on the object, the operator quits the test script. The test script reports the number of attempts, the number of successes, and the percentage of the attempts that were successful.

E. Execution Constraints

Because the system operates on novel objects, and does not model or recognize objects, there are no constraints that apply to specific objects in the way that e.g. orientation constraints would be applied to an open-topped container.

III. Results

The system was able to pick up all but one of the objects at least once. On average, it succeeded on 46.7% of the grasp attempts, but the level of success varied heavily with the object, rather than remaining consistent across objects. This indicates that the problems arose as a result of qualities of specific objects, rather than systemic problems that affect all objects.

It had been theorized that deformable objects would be easier to grasp, due to their conformation to the gripper. In practice, while many of the deformable objects were more easily graspable, the lobster and agar packet were the most difficult.

The most common problems were caused by perceptual difficulties. The white bottle presented a challenge to the laser detection system. The surface of the bottle was consistently overexposed in the camera image, and so the laser did not have sufficient contrast to be detected at some locations on the bottle. If the laser is not detected in a frame, the id of the segment associated with the laser point is set to 0. The 0th segment is the set of all points that LCCP segmentation did not assign to any other segment. When this condition is detected, the system cannot determine which object should be grasped, and so does not attempt to find grasps.

The water bottle was largely transparent. Sections of it were frequently missing from the point cloud, and so were not considered as grasp locations. Those areas were also not considered "occupied," and so were likely to be hit by the arm as it attempted to move to make a grasp. The clear areas of the water bottle also caused it to appear to be thinner than it was, likely due to reflection of the IR pattern in some areas. Some grasps on the illusory thin areas were still successful, but they were unreliable when the arm moved. Transparent areas also allowed the laser to pass without illuminating the object, increasing the number of bad points.

Imperfections in the registration of the color image, which is used to detect the laser, and the point cloud, which is used to find grasps, meant that some areas near the edges of the object were actually considered to be part of some other region of the point cloud. These points are referred to as "Bad Points" in table **??**. Bad points were most frequently located along the top edge of objects, and resulted in the table behind them being selected as the target object to grasp. Due to its thin shape and reflectivity, the agar flake packet had more bad points than possible valid grasp points. Specular reflection off of some objects contributed to bad points as well. The reflected laser light would illuminate the surface of the table, causing the system to regard the tabletop as the selected object. In an environment with multiple objects, this effect could result in accidental targeting of undesired objects.

With the black pepper container, the failures were consistently due to attempting to grasp the front corner of the box. The top of the box was not present in the point cloud due to the position of the box relative to the camera, and so the closed box could not be distinguished from a box with an open top, and the depth of the box was not available to the system. When the gripper approached, the lower finger of the gripper would contact the front of the box, and the upper finger would contact the top of the box, pushing the box backwards. When the gripper closed, the box would be knocked over. The white bottle had a similar problem with its upper area, where the spray nozzle was located. The nozzle was too narrow to grasp, but because the side of the nozzle faced the camera, the thickness of the nozzle was hidden.

Object	Attempts	Successes	Percent Success	Bad Points	Narrow	Knocked	Attempts	Successes	Percent Success	Bad Points	Narrow	Knocked
Rocket	15	14	93.3%	1	8	3						
Blue Ball	15	11	73.3%	2		2						
Spice Jar	15	11	73.3%	1		2						
Soapbox	14	10	71.4%									
Vacuum Nozzle	13	8	61.5%	2		1						
Stuffed Drill	15	9	60.0%	2								
Tissue Packet	12	7	58.3%			1						
Stuffed Toy Screw	9	5	55.6%		2							
Creamer	16	8	50.0%	1		3						
White Pepper	9	4	44.4%	3								
Black Pepper	22	9	40.9%	2		9						
Water Bottle	22	7	31.8%	5		7						
Blue Bottle	33	12	36.4%		4	8						
Coffee Stirrers	28	8	28.6%			9						
White Bottle	47	11	23.4%	1	2	13						
Mouse	13	3	23.1%	2	1							
Lobster	13	2	15.4%	2	4							
Agar Flake Packet	7	0	0.0%	11								

TABLE I: Manipulation success rates and causes of error for various objects.

Another possible failure is that, due to the segmentation failing to separate e.g. an object and the table it is resting on, all of the points in the volume of space around the selected segment are considered part of the target object. Combining the table and an object results in an illusory object that is sufficiently large that the arm could not likely manipulate it, and so is considered an error. If more than 90% of the available points are considered part of the object, the system does not use the LCCP segmentation and instead considers all points within an 8cm radius of the laser to be part of the target object. This heuristic allows the system to function despite inconsistent segmentation. However, it can result in the system missing good grasp locations that are located more than 8cm from the laser on the target object.

IV. Conclusions

The grasp selection algorithm described in this paper can achieve 85% success grasping single objects with the point cloud from a single RGB-D camera [?]. As described above, certain kinds of perceptual problems resulted in difficulties finding a valid grasp. However, all of these problems have potential solutions.

Transparent objects are a known problem for RGB-D sensors. Because transparent objects may not be fully transparent at other wavelengths, it is possible to detect them under specialized illumination [?]. Such an approach requires controlled illumination, but can reconstruct unknown objects. In the Microsoft Kinect, the area occupied by a transparent object appears as an invalid area of the image, but the shape of the invalid area can be matched to views of modeled objects to estimate the shape of the transparent object that created the invalid area [?]. Unfortunately, this approach requires the use of model objects, and so is likely to have difficulties with novel transparent objects. Training with a larger set of objects, such as the PSB or YCB object collections, may alleviate this problem sufficiently to permit the system to operate on transparent objects likely to be encountered in ADLs.

Primary cause of failures to grasp objects is not the grasp selection software. Segmentation of the object is a problem. When the segmentation node fails to select the object, it usually selects the table under the object, or merges the table and the object into one segment, rather than making them separate segments.

Baxter robots have two arms, and the system presented in this work only uses one of them. The other arm will be mounted to a rolling cart, for use by wheelchair users, as with the Bath University "Wessex" robot [?]. This will facilitate testing and user interface development with a larger population of users, by including people who cannot use a mobility scooter.

References

[1] J. Stuckler, D. Holz, and S. Behnke, "Demonstrating everyday manipulation skills in robocup@ home," *IEEE Robotics and Automation Magazine*, pp. 34–42, 2012.

[2] G. Kragten, C. Meijneke, and J. Herder, "A proposal for benchmark tests for underactuated or compliant hands," *Mechanical Sciences, 1 (1), 2010*, 2010.

[3] C. Zielinski, W. Szynkiewicz, T. Winiarski, and M. Staniak, "Rubiks cube puzzle as a benchmark for service robots," in *Proceedings of the 12th IEEE International Conference on Methods and Models in Automation and Robotics, MMAR*. Citeseer, 2006, pp. 579–84.

[4] D. Berenson and S. S. Srinivasa, "Grasp synthesis in cluttered environments for dexterous hands," in *Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on*. IEEE, 2008, pp. 189–196.

[5] C. Goldfeder, M. Ciocarlie, J. Peretzman, H. Dang, and P. K. Allen, "Data-driven grasping with partial sensor data," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 1278–1283.

[6] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark," in *Shape modeling applications, 2004. Proceedings*. IEEE, 2004, pp. 167–178.

[7] M. Popović, G. Kootstra, J. A. Jørgensen, D. Kragic, and N. Krüger, "Grasping unknown objects using an early cognitive vision system for general scene understanding," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 987–994.

[8] G. Kootstra, M. Popović, J. A. Jørgensen, D. Kragic, H. G. Petersen, and N. Krüger, "Visgrab: A benchmark for vision-based grasping," *Paladyn, Journal of Behavioral Robotics*, vol. 3, no. 2, pp. 54–62, 2012.

[9] A. ten Pas and R. Platt, “Localizing antipodal grasps in point clouds,” *CoRR*, vol. abs/1501.03100, 2015. [Online]. Available: http://arxiv.org/abs/1501.03100

[10] S. C. Stein, M. Schoeler, J. Papon, and F. Worgotter, “Object partitioning using local convexity,” in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 304–311.