Mobile Manipulation Through An Assistive Home Robot

Matei Ciocarlie[1], Kaijen Hsiao[1], Adam Leeper[2] and David Gossow[3]

***Abstract*— We present a mobile manipulation platform operated by a motor-impaired person using input from a head-tracker, single-button mouse. The platform is used to perform varied and unscripted manipulation tasks in a real home, combining navigation, perception and manipulation. The operator can make use of a wide range of interaction methods and tools, from direct tele-operation of the gripper or mobile base to autonomous sub-modules performing collision-free base navigation or arm motion planning. We describe the complete set of tools that enable the execution of complex tasks, and share the lessons learned from testing them in a real user's home. In the context of grasping, we show how the use of autonomous sub-modules improves performance in complex, cluttered environments, and compare the results to those obtained by novice, able-bodied users operating the same system.**

I. INTRODUCTION

Independence, and a sense of control and freedom, are some of the key factors positively correlated with life satisfaction and health (both psychological and physical) for older adults and people with motor impairments. Confidence in the ability to undertake various tasks is core to one's psychological functioning [9], and a greater sense of control over life is positively correlated with better health [21] and a reduced mortality rate [22].

The vision for this study is that of a mobile manipulation platform sharing a living environment and operating side-by-side with its user, increasing independence and facilitating activities of daily living. In particular, we focus on the set of Instrumental Activities of Daily Living (IADLs) [23], which require manipulating the environment (e.g., performing housework) away from the user's body. A mobile robot could provide assistance with a variety of IADLs, operating in a large workspace without encumbering the user.

A key step for achieving this vision is enabling mobile robots to handle the complexity and variability inherent in real living environments. Despite impressive advances over the past decades, these factors have so far prevented versatile manipulators from achieving the level of reliability needed for long term deployment in real human settings.

We posit that the difficulties associated with the design of fully-autonomous systems could be mitigated by involving the care receiver in the loop, as a user and operator of the robot. The user's cognitive abilities can be tapped to deal with conditions that have proven difficult for autonomous systems to deal with. Autonomy would still play a vital role in this shared framework, but the human operator would provide the information that robots are incapable of deriving themselves. As additional autonomous components mature, they can be incorporated into the system to help reduce the load on the operator. Such a system could be reliable and robust enough for deployment in the near future.

Fig. 1. A mobile robot, operated by a motor-impaired user, performing a manipulation task in a real home environment.

With this directional goal in mind, we have developed a system that enables a motor-impaired user to command a robot in performing manipulation tasks in complex environments. The operator can make use of a wide range of interaction methods and tools, from direct teleoperation of the gripper or mobile base to autonomous sub-modules performing collision-free base navigation or arm motion planning. The operator receives feedback from the robot via a computer screen and provides input using only a mouse. In particular, this enables our system to be used by operators with severe motor impairments who can nonetheless still control a mouse cursor via a head tracking device.

This paper's contributions are as follows. We introduce what is, to the best of our knowledge, the first example of a mobile manipulation platform operated by a motor impaired person using only a head-tracker single-button mouse as an input device, and demonstrated for both varied and unscripted manipulation tasks in a real home and limited

---

*This work was partially supported by the NSF under Award IIP-1142743. A. Leeper is partially supported by an NSF GRFP Fellowship.

[1]Willow Garage Inc., Menlo Park, CA. Email:
`{matei,hsiao}@willowgarage.com`
[2]Stanford University, Stanford, CA. `aleeper@stanford.edu`
[3]Technical Univ. of Munich, Germany. `dgossow@cs.tum.edu`

forms of social interaction. We describe the set of tools that enable the execution of complex tasks, their interplay, and their effect on the overall system. We share the lessons learned from deploying a real robot in a real user's home and attempting to manipulate the environment. In the context of grasping, we show how the use of autonomous sub-modules improves performance in complex, cluttered environments, and compare the results to those obtained by novice, able-bodied users operating the same system.

A variety of assistive robot systems have been tested in the past, including desktop workstations with robotic arms, wheelchair-mounted robotic arms, powered orthotic and prosthetic arms, and mobile manipulators[4], [7]. An example of an assistive robot tested for remote manipulation in real homes is shown in [15]. Recent assistive mobile manipulation systems with a number of important capabilities include Care-o-bot 2[6], SAM[16], and El-E[10]. In addition to showing and quantifying individual capabilities such as object grasping, in this paper we demonstrate operation in a real home for a complete mobile manipulation task, as well as limited social interaction through manipulation.

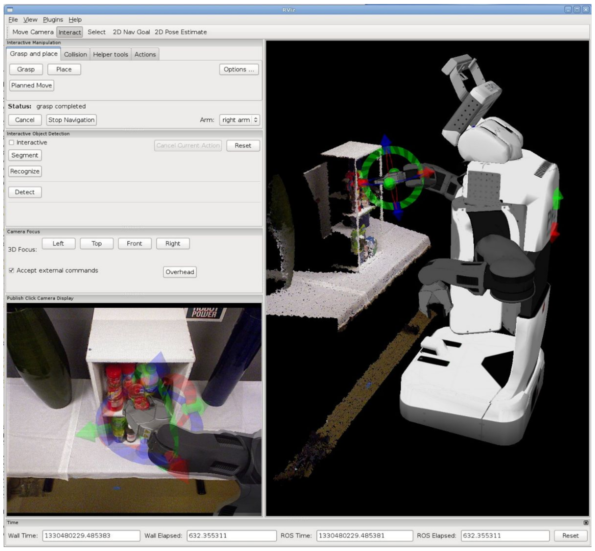
Despite previous advances in assistive mobile manipulators, none has been widely adopted to date. Part of the reason is their cost, as many of these robots, including the PR2 which we use here, are not suitable for widespread commercial adoption. Nonetheless, we believe that creating and demonstrating a system such as the one presented here, with a comprehensive suite of tools with varying levels of autonomous assistance for perception, navigation, and manipulation, can enable future flexible and competent platforms with widespread adoption

II. System Overview

Our system has two main components: the robot itself, a two-armed mobile manipulation platform, and the interface used by the operator to command and receive feedback from the robot. The interface can run on a commodity desktop or laptop computer. The system was designed for remote operation, with the goal of enabling the operator to perform tasks through the robot even from another room, or in other situations when there is no direct visual or audio contact.

A. Hardware Platform

The hardware we used was the PR2 personal robot [24]. The PR2 has two compliant, backdriveable 7-DOF arms with parallel-jaw grippers. We used two range sensors: a widely available Microsoft Kinect$^{\text{TM}}$ mounted on the head of the robot (providing both range and color images), and a tilting laser rangefinder mounted on the chest (used for autonomous collision avoidance). The PR2 can communicate with the computer running the teleoperation interface via a commodity wireless network; we expect that a mobile robot in real households will have to be untethered to perform a large number of tasks. The PR2's form factor was designed for enabling operation in typical human environments. With arms folded in, it has a similar footprint (square, 668mm on each side) to a wheelchair, and can thus navigate in ADA-compliant spaces. It has a telescopic spine, allowing it to touch the floor with its arms, operate low cabinets, and manipulate objects on table- and counter-tops.



Status: grasp completed

Arm: right arm

☐ Interactive

☑ Accept external commands

Wall Time: 1330480229.485383 Wall Elapsed: 632.355311 ROS Time: 1330480229.485381 ROS Elapsed: 632.355311

Fig. 2. Overview of user interface.

B. User Interface

We developed a "point-and-click" Graphical User Interface based on `rviz` [20], a 3D robot visualization and interaction environment that is part of the Robot Operating System (ROS) [14]. The choice of a point-and-click interface was motivated by accessibility: while higher-dimensionality input methods such as trackballs and haptic devices would provide additional benefits for teleoperation, a major advantage of a simple cursor-driven interface is the widespread accessibility of devices that provide cursor control, including, for example, head trackers for motion-impaired users.

Our user interface is shown in Fig. 2. It presents the user with two main displays: on the bottom left, a live image from the Kinect camera mounted atop the PR2's head; on the right, a rendered image of the PR2 in its current posture (using joint encoders for proprioception), along with a 3D point cloud showing the world as seen by the Kinect.

The user can point the robot's head by left-clicking anywhere in the camera view, which centers the robot's camera on the clicked point. Because the right image is only a rendering, its virtual camera can be dragged to any position by rotating, translating, and zooming the scene in and out, which is useful for seeing and understanding the 3D scene by viewing it in motion and from multiple angles.

We have found this dual visual feedback, one from a real 2D camera and one from an artificial rendering, to be highly useful for remote operation. The real image is easy to interpret for human operators, but the limited ability to change the viewpoint makes it difficult to rely on exclusively for tasks requiring depth perception. Conversely, the rendered image can be seen from any viewpoint, but must rely on 3D sensors for data acquisition, and requires the user to be able to manipulate a virtual camera around a rendered scene.

Our user interface allows the user to send commands to the robot in two main ways. The first one is through 3D widgets that are added to the rendered 3D world that the user can click on, drag, translate, rotate, etc. These are implemented using the `interactive_markers` ROS framework [5]. The second interaction type is through more conventional dialog windows. Fig. 2 shows examples of both interaction methods, with a 3D clickable control on the robot's gripper, and a set of dialogs in the upper left part of the window.

III. SHARED AUTONOMY TOOLS FOR MOBILE MANIPULATION

When an assistive robot is enabling a motor-impaired user to physically interact with the world, the human operator can use her cognitive skills to handle complex and unexpected situations, as long as she can receive relevant feedback from and send appropriate commands to the robot. However, a framework in which a teleoperator is always in charge of every aspect of the robot's behavior can be cumbersome.

We propose to address this problem by using a Human-in-the-Loop (HitL) framework that uses both low-level, direct teleoperation (when needed) and autonomous modules for completing sub-tasks (when possible). Low-level command tools are available for complex situations, requiring direct intervention by the operator. Autonomous capabilities can play the important roles of reducing the load on the operator and increasing efficiency for sub-tasks that can be performed reliably, or that require operator input in a form that is relatively effortless to provide. In this section, we introduce the components of our interface built along these directions, focusing on three main components of a mobile manipulation platform: perception, manipulation, and navigation.

A. *Looking Around and Perceiving*

Situational awareness is crucial for any teleoperation interface. Our framework has a number of tools to allow the user to both directly visualize the environment, and also help the robot in better understanding its world.

The user can click anywhere in the streaming camera image feed to point the head, and 3D point clouds from the Kinect allow the user to see the 3D world from any camera angle in the virtual camera view. If compressed, point cloud data can also be streamed in real time. However, it is often the case that crucial parts of the environment will be occluded by the robot. For instance, while manipulating, the robot's arms and grippers typically occlude the object being manipulated. Users can thus take a point cloud snapshot, which stays in the rendered image and is only refreshed when requested. Such a cloud is shown on the right side of Fig. 2.

When the robot is looking at objects of interest for manipulation, the user can ask it to autonomously segment flat surfaces and well-separated ($\geq$3 cm) objects on the surfaces, (Fig. 5, top left). In more complex scenes that autonomous segmentation cannot handle, the user can interactively aid the robot by drawing boxes around objects to segment. Currently there are two interactive segmentation tools available for our interface, one that uses a graph-cut algorithm as described in [13], and one that uses the algorithm from [2].

The user can also ask the robot to autonomously recognize specific object models stored in a database. Currently, we provide a 2-D, ICP-like algorithm that can be used on previously segmented objects [3], as well a textured object detection algorithm that operates on general scenes [19]. Three of the objects in Fig. 5, top left have been autonomously recognized, and their object model meshes are shown in the appropriate poses. If needed, the user can correct the robot's recognition results, by clicking through a set of possible object detections returned by the object recognition algorithms, or by rejecting all returned detections.

The aforementioned features all use the head-mounted Kinect camera. The robot also has a base laser, used for localization and obstacle avoidance while navigating, and a tilting laser rangefinder used for both navigation and collision-free arm motion planning. These sensors build what we refer to as collision maps, or occupancy grids showing the obstacles in the environment. As we describe below, autonomous motion planning for both the base and the arms can be extremely useful for moving the robot. However, moving obstacles or sensor noise can clutter collision maps and leave them unusable. Our interface therefore enables the user to visualize, clear or regenerate the robot's collision maps, or even ignore them altogether and move open-loop.

B. *Manipulating the Environment*

Manipulating objects presents a very diverse set of challenges, such as the high dimensionality of the movement space, the non-anthropomorphic characteristics of the arms, and the complex and cluttered scenes encountered in real homes. Our system combines direct, manual teleoperation controls with controls that offer autonomous functionality in order to give the user efficient and flexible ways to accomplish a variety of tasks.

1) Manual Teleoperation: Despite the large body of research in autonomous manipulation, there will always be tasks that are not anticipated or well-handled by any set of autonomous modules, especially in complex, unstructured settings like a home. In situations where human ingenuity is needed to complete a task, low-level control of the robot's arms can provide the necessary means.

End-Effector Pose: The 7-DOF arms of the PR2 present an interesting challenge for 2D cursor control. For most tasks it is the 6D pose of the end-effector that is of interest. Our gripper control consists of a set of rings and arrows that allow the user to instantly move the gripper, along one dimension at a time. Dragging on a ring will rotate the gripper, while dragging an arrow will translate it (Fig. 3).

Elbow Posture: The PR2 arm has 1 degree of redundancy with respect to the pose of the end-effector. Intuitively, this redundancy allows the elbow to float at different "heights" while keeping the gripper in the same pose. The user may not care where the elbow is during a task, so long as the gripper can reach the desired workspace. However, if the posture of the arm causes unwanted collisions, we provide a 1D ring

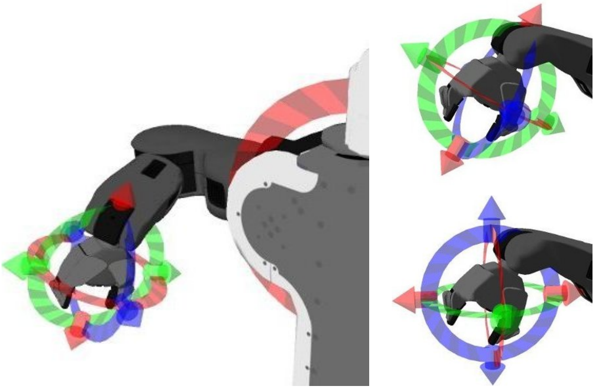


Fig. 3. Gripper control (6D) and shoulder ring (1D). Right images show a gripper-aligned control (top) and a world-aligned control (bottom).

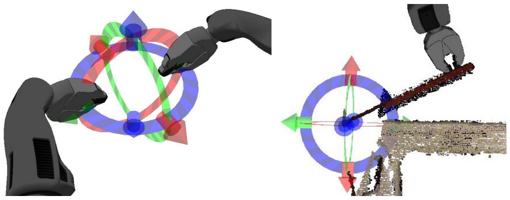


Fig. 4. Left: 2-armed Cartesian control. Right: Control frame moved to cabinet hinge for opening.

(Fig. 3) giving the user a measure of control of the elbow height, mapped loosely to the ring's rotary motion.

Changing the reference frame: Another useful feature of our low-level Cartsian control interface is the ability to move the reference frame for the Cartesian controller. Two common options, available through direct shortcuts, are to use either a gripper-aligned coordinate frame (e.g., for moving the fingertips directly into or out of grasps), or a world-aligned coordinate frame (e.g., for moving grasped objects directly up). Both types are shown in Fig. 3.

The user can also move the control reference frame to an arbitrary pose relative to the gripper; this is useful for tasks such as opening cabinets or using tools. Moving the control frame to the hinge of a cabinet (Fig. 4) allows a gripper grasping the cabinet handle to smoothly move in an arc around the hinge simply by rotating one control ring.

Our framework also allows both grippers to be moved at once by switching to a two-armed control mode, in which the reference frame is set to be halfway between the current poses for the two grippers; this mode is useful for moving around objects grasped by both grippers at once.

2) Autonomous Modules and Tools: For a certain class of well-defined and extensively-studied tasks, such as picking up an object, it can be more efficient to allow autonomous modules to handle most of the process.

Autonomous grasping: For objects that have been recognized, the user can right-click on the overlaid model and ask the robot to pick up the object; the robot then pulls a list of precomputed grasps for that object from its database and uses the first feasible one, as described in [3]. The user can also ask the robot to directly pick up unrecognized but segmented objects. In this case, the robot will compute grasps using the segmented object's point cloud, with the algorithm described in [8]. Both cases are illustrated in Fig. 5. In either case, upon successfully grasping an object, a collision model in the form of the object's bounding box will be attached to the robot's gripper, so that future planned motions avoid hitting the environment with the grasped object.

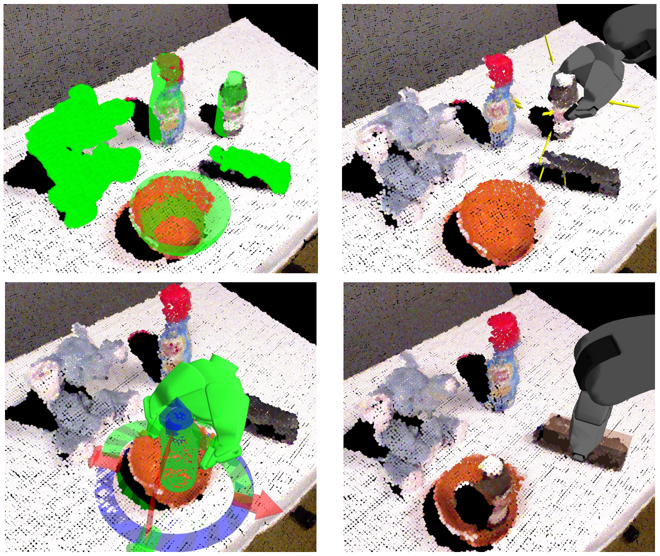


Fig. 5. Autonomous grasping for segmented and/or recognized objects. Top left: objects have been segmented, and three have been recognized (shown by superimposed meshes). Top right: autonomous grasping for a recognized object (peroxide bottle) using pre-computed database grasps. Bottom left: choosing a placing location for the peroxide bottle; the 3D mesh is available to the robot as the object had been previously recognized. Bottom right: Autonomous grasping for a segmented but unrecognized object (stapler).

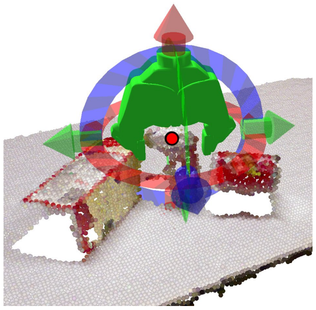


Fig. 6. The user indicates a desired grasp by placing a virtual gripper in the scene, and letting the robot plan a collision free execution path for it.

Collision free grasp execution: In situations where the robot fails at autonomously segmenting or recognizing the desired object, or where a particular type of grasp is desired, a different interface allows the user to select just the final grasping pose, while still taking advantage of autonomous, collision-free motion planning. The user first clicks on a desired point in the 3D environment snapshot. A virtual gripper model is displayed at the clicked location; the operator can use a rings-and-arrows control to adjust its pose as desired (Fig. 6). As the user is adjusting the virtual gripper indicating the desired grasp, the robot continuously computes whether the grasp is feasible through collision-free motion planning. If it is, the virtual gripper control turns green indicating to the user that the grasp can be executed. Once the user is satisfied

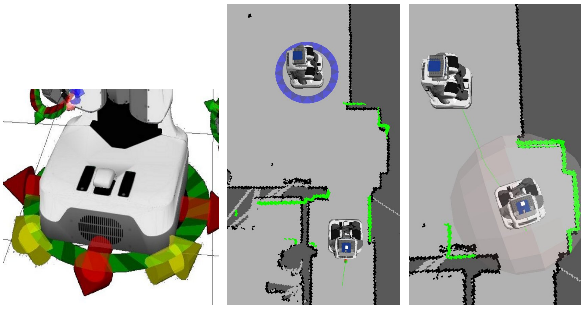


Fig. 7. Moving the base by clicking on rate-control arrows (left), or passing a goal to an autonomous navigation module (middle and right).

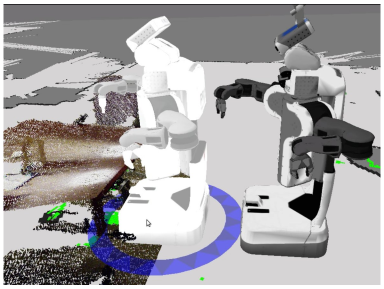


Fig. 8. Using an open-loop navigation goal allows the robot to get close to obstacles such as tables, or to push moveable obstacles with the base.

with a feasible grasp, the robot executes it autonomously.

Placing: After grasping an object, the user can ask the robot to place it at a desired location in the environment, also selected through a virtual gripper rings-and-arrows control. If a model of the object is available, it will be shown along with the virtual gripper, so that the user can visualize the object at the desired place location (Fig. 5, bottom left). When the user is satisfied with the pose, the robot will autonomously plan a collision-free path to it and place the object.

Advanced options allow the user to further customize autonomous grasp and place execution with features such as reactive grasping [8], reactive placement, or slip detection/grasp force adjustment [18].

Planned moves: For other tasks, the operator can ask the motion planner to compute a collision free path to a desired pose in free space by using a similar virtual gripper control.

C. Moving Around: Base Movement and Navigation

For base movement and navigation, our framework provides tools for autonomous, collision-free navigation, as well as open-loop movement for small adjustments near or even into contact with obstacles. The user can ask the robot to navigate through free space by dragging a virtual robot model to a desired position and orientation in a scene. (Fig. 7, middle and right). If a map is available, it can be used as a reference when selecting the goal. 2-D map-making, global path planning, local/reactive path planning, and collision avoidance are provided by the PR2's navigation stack[12].

A similar control is used to perform small, precise, open-loop movements that 2-D autonomous navigation is incapable of or unwilling to execute (Fig. 8), such as moving the base under a table. Because the control takes on the current shape of the PR2 according to the robot's proprioception, the user can precisely position the goal relative to a 3D snapshot of the local environment. For both autonomous and open-loop goals, the robot can be stopped at any time by clicking on a translucent bubble that appears around the robot while it is driving (Fig. 7, right).

Fig. 9. Shelf environment used in the grasping study.

For less-precise but faster small adjustments, the user can also drive the base directly (both strafing and rotating) using rate-controlled arrows (Fig. 7, left).

IV. DEMONSTRATIONS AND RESULTS

The cursor-based assistive teleoperation system described in this study was developed and tested in collaboration with a pilot user named Henry Evans. Henry is quadriplegic and mute due to a brainstem stroke, and can control a computer mouse via a headtracker and also issue single-button click commands with his thumb. Henry performed evaluations and proof-of-concept demonstrations of the system, and piloted its use in several real-life situations.

In this section, we present results from Henry operating the robot in three different contexts. The first attempts to quantify the performance of the manipulation tools for grasping in very cluttered environments. The second one is an example of object pick-and-place enabling social interaction, in the context of giving candy to Halloween trick-or-treaters. In the third demonstration, Henry uses the robot in his own home to retrieve a towel from the kitchen, combining navigation, door and drawer opening and closing, and object grasping.

A. Grasping in a Cluttered Environment

In this experiment, we quantified the ability of our system to execute grasping tasks, a key prerequisite for many manipulation tasks involving object acquisition or transport. Given the goal of operating in real users' homes, we focused on never-seen-before objects sitting in highly cluttered and constrained environments. As shown in Fig. 9, our test environment consisted of a small two-tiered shelf containing a large number of tightly packed objects.

The high degree of clutter and occlusion prevented the use of fully autonomous grasping, based on either object

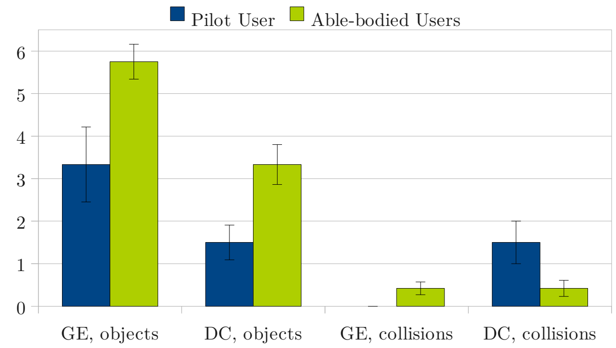


Fig. 10. Results for study on object grasping in cluttered environments for Grasp Execution (GE) and Direct Control (DC) strategies, showing the number of objects grasped and the number of collisions incurred.

segmentation or recognition. We therefore quantified the performance of two grasping strategies:

- **Direct Control** (DC): this strategy involved the operator using the gripper Cartesian control tool in order to bring the gripper into a desired pose relative to a target object, before closing the gripper. When using this method for grasping, the user must essentially "drag" the gripper all the way from its starting position to the desired grasp location, while avoiding collisions along the way.
- **Grasp Execution**(GE): for this strategy, the operator used the final grasp point selection tool described in Sec. III-B. Once a desired grasp pose was confirmed, autonomous components were responsible for planning appropriate arm joint trajectories and avoiding unwanted collisions.

The operator's goal was to securely grasp and lift as many objects from the shelf as possible in a limited amount of time (10 minutes per run), while avoiding collisions with the environment (the shelf or the table). Collisions were marked if unwanted contact was strong enough to displace the shelf or potentially cause damage to either the robot or its surroundings, as opposed to consequence-free occurences such as lightly brushing shelf walls.

With Henry Evans as our pilot user, we performed 3 trials using the Grasp Execution strategy, and 2 trials using the Direct Control strategy. Each trial was defined as a 10 minute run, over which we counted the number of objects grasped and collisions incurred. The results showed that our system can indeed enable an operator with motor impairments to execute grasps even in highly challenging environments. The robot successfully grasped 5, 3 and 2 objects respectively over the 3 trials using Grasp Execution. It also grasped 1 and 2 objects in the Direct Control trials. No collisions were encountered when using Grasp Execution; 1 and 2 collisions occured when using Direct Control.

Fig. 10 shows the complete results of our trials. For reference, we place them in the context of a previous user study [11] where we quantified the robot's performance at the same task, but when operated by able-bodied novice users. However, the small sample size of our current Pilot User study limits the usefulness of quantitative comparisons performed between the two data sets.

Fig. 11. Henry (in the bottom right, using robot interface running on laptop) giving Halloween candy to children through the PR2 robot.

B. Trick-or-Treat: Social Interaction through a Robot

An important category of activities of daily living include social engagement and interaction with other people [1], [17]. In this study, we demonstrate a particular case where the ability to manipulate objects in the environment, and perform relatively simple pick-and-place operations, can serve as an enabler for social interaction. In the activity commonly referred to as trick-or-treating, occurring on the yearly Halloween holiday, children dress up in costumes and walk through the local community receiving candy from neighbors. In personal communication, Henry Evans described a desire to interact with trick-or-treaters and hand out candy, through the intermediary of the PR2 robot.

As a proof of concept implementation of this task, we set up a semi-structured environment designed for this type of interaction. The event took place on Halloween at a local mall, open to any children and their families present in the mall at that time. Henry and the robot sat behind a large table separating them from the public's space. In addition, a small table with candy bars was placed to the robot's side. The setup is shown in Fig. 11.

As a child would approach the robot and hold out his or her candy bag over the separating table, Henry would command the robot to pick up a candy from the side table and then place it inside the child's bag. For picking up the candy, Henry used either the autonomous grasping tool based on object segmentation, or the Grasp Execution strategy described in the previous section, where he indicated the desired grasp point and the robot completed the grasp. For placing the candy in the bag, Henry moved the gripper to a pre-defined pose in front of the robot (either open-loop or using collision-free motion planning), then performed fine adjustments using direct Cartesian control in order to drop the candy inside the bag.

Over the course of one hour, Henry successfully handed out candy to more than a dozen local children. Occasional incidents included the robot dropping the candy bar on the way to a child's bag and having to re-grasp, or missing the bag on the first attempt to drop the candy inside and having to readjust. However, no major failures or disruptions occurred, and all interactions were completed successfully.

C. Mobile Manipulation in a Real Home

The vision driving our assistive robotics project is that of a robot operating in its user's home for indefinite periods of time. The variability and complexity encountered in real homes and the robustness needed for continuous operation over weeks and months, will be the ultimate reference criteria for such a system. As a step in this direction, Henry performed a number of tests with our system in his home. The environment was completely novel to the robot, with the single exception of a 2D floorplan of the house, acquired off-line, and used for localization.

In this setting, Henry demonstrated the ability of the system to perform a complete a task combining navigation, perception, and both prehensile and non-prehensile manipulation. Execution is illustrated in Fig. 12 and summarized below, along with the tools used for each component and the approximate time taken to execute:

- drive to from living room to kitchen: autonomous navigation combined with open-loop base movement (21 min).
- open kitchen cabinet door in order to inspect is contents: Grasp Execution tool for grasping handle, Cartesian end-effector control for opening door (6 min).
- close kitchen cabinet door: Cartesian end-effector control, used for pushing the door shut with the forearm (11 min).
- open kitchen drawer: Grasp Execution for grasping handle, Cartesian end-effector control for pulling drawer (6 min).
- grasping towel from drawer: Grasp Execution tool (3 min).
- bring towel to Henry's wheelchair in the living room: autonomous navigation combined with open-loop base movement (7 min).

The complete task was executed in a single continuous run (54 min), and succeeded on the first attempt.

V. LESSONS AND LIMITATIONS

While the ability to successfully perform a complete mobile manipulation task, on the first attempt and in a complex novel environment, is highly encouraging, our results also highlight numerous potential areas of improvement. Focusing on the time taken to perform the task, we note that:

- Due to a synchronization error between the robot's laser sensor and its computers, the autonomous navigation algorithm failed to process most goals received from the operator, who had to use manual base movement instead. As a result, navigation time accounted for more than half of the total task. This occurrence illustrates both the importance of having fallback mechanisms or multiple ways of achieving the same goal, and the potential efficiency gain obtained when more autonomous tools can be used.
- Cartesian end-effector control along with appropriate 3D perception can enable unforeseen tasks; for example, it enabled pushing a door shut with the forearm, a task that no module in our codebase was explicitly designed for. However, using it for more than small pose adjustments or short movements, especially relative to obstacles in the environment, can be laborious and time-consuming. Sub-tasks that could take advantage of autonomy (such as grasping the door knob, the cabinet handle, or the towel) were executed more efficiently.
- For grasping objects, the Grasp Execution strategy, where the operator only has to select the final gripper pose, proved more effective than Direct Control, as it leverages autonomous motion planning and collision avoidance. However, additional information from the robot can further increase efficiency. A significant part of the user's effort involves modifying a desired gripper pose so that the motion planner considers it reachable. We are currently augmenting our system to provide nearby, feasible suggestions in the event of an infeasible selected pose.

In addition, we can draw a number of conclusions regarding mobile manipulators intended for in-home use, where few strong assumptions can be held. For example, a motion planning module for semi-structured settings could always expect the robot to be at a safe distance from any obstacle, and simply return an error if this assumption is violated. This, however, would greatly limit its usefulness in an unstructured setting, where unexpected contacts can and will occur.

The operator must also be equipped with tools to correct errors in the robot's view of the world. For example, in order to take advantage of the autonomous motion planner in as many cases as possible, the operator must be able to interact with the robot's representation of the world, adding obstacles that would otherwise be invisible to the robot's sensors (e.g. shiny or transparent objects), and removing non-existent obstacles that are the result of sensor noise.

The complexity of a real home can simply be beyond the capabilities of any autonomous algorithm. An illustrative example encountered in our pilot tests involved curtains billowing due to the air current from the robot's fan, and registering as obstacles in the robot's navigation map. Manual annotations, heuristic behaviors, or some level of altering the environment to suit the robot might be the only solutions for such extreme corner cases.

Finally, we have found that a simulated environment used for training was a key enabler to the successful execution of complex tasks. Even though a simulator can not accurately replicate all the complex interactions with real-life objects, it can still help the operator become familiar with the interface and robot, with no risk of injury or damage. We believe that appropriate training mechanisms will prove central to the effort of enabling non-roboticists to effectively use the widely-deployed assistive robots of the future.

VI. CONCLUSIONS

In this paper we have shown how an assistive mobile robot, operated by a pilot user, can perform mobile manipulation tasks in rich, unstructured environments. The operator sends commands to, and receives feedback from the robot through an interface running on a commodity desktop or laptop computer, using only a head-tracker cursor as an input device. The available command tools range from low-level Cartesian

Fig. 12. PR2 robot, operated by a pilot user, performing a mobile manipulation task in the user's home. From left to right: grasping a cabinet handle, pushing open a cabinet door, opening a drawer, grasping a towel inside the drawer, and navigating to desired drop-off location.

movement commands for the base or gripper to autonomous modules for collision-free navigation or grasping.

Our pilot studies showed that a motor-impaired operator can command the robot to successfully grasp objects even amidst clutter and in constrained settings. We have also tested our system in a real home environment, where a pilot user completed a mobile manipulation task involving both prehensile and non-prehensile manipulation, as well as perception and navigation. The task was successfully completed on the first attempt, although we found that a number of components would greatly benefit from increased performance and faster execution time. Our experience performing these studies suggests that mobile manipulators have the potential to enable motor impaired users to perform a wide range of activities of daily living. Through a combination of human control and autonomous algorithms, assistive robots could one day gain the versatility and robustness needed for long term operation in real homes.

ACKNOWLEDGMENTS

This work would not have been possible without Henry Evans, who we thank for inspiring and motivating the project, as well as for his active participation in development and testing. This work is part of a collaborative project between Willow Garage, the Georgia Tech Healthcare Robotics Lab led by Prof. Charlie Kemp, and the Media and Machines Laboratory at Oregon State led by Prof. William Smart; we gratefully acknowledge their contributions.

REFERENCES

[1] J. M. Beer, A. Prakash, T. L. Mitzner, and W. A. Rogers. Understanding robot acceptance. Technical Report HFA-TR-1103, Georgia Institute of Technology: Human Factors and Aging Laboratory, 2011.

[2] M. Bjorkman and D. Kragic. Active 3d scene segmentation and detection of unknown objects. In *IEEE Intl. Conf. on Rob. and Automation*, 2010.

[3] M. Ciocarlie, K. Hsiao, E. G. Jones, S. Chitta, R. B. Rusu, and I. A. Sucan. Towards reliable grasping and manipulation in household environments. In *Intl. Symp. on Exp. Robotics*, 2010.

[4] J.L. Dallaway, R.D. Jackson, and P.H.A. Timmers. Rehabilitation robotics in europe. *IEEE Trans. on Rehabilitation Eng.*, 3(1), 1995.

[5] D. Gossow, A. Leeper, Hershberger D., and M. Ciocarlie. Interactive markers: 3-D user interfaces for ROS applications (ROS Topics column). *IEEE Robotics & Automation Magazine*, 18(4), 2011.

[6] B. Graf, M. Hans, and R.D. Schraft. Care-O-bot II: development of a next generation robotic home assistant. *Autonomous robots*, 16(2):193–205, 2004.

[7] K.Z. Haigh and H.A. Yanco. Automation as caregiver: A survey of issues and technologies. *AAAI Workshop on Automation as Caregiver*, pages 39–53, 2002.

[8] K. Hsiao, S. Chitta, M. Ciocarlie, and E. G. Jones. Contact-reactive grasping of objects with partial shape information. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, 2010.

[9] S. E. Iso-Ahola. Social psychological foundations of leisure and resultant implications for leisure counselling. In E.T. Dowd, editor, *Leisure counselling: Concepts and applications*, pages 97–125. Springfield, IL: Charles C. Thomas, 1984.

[10] A. Jain and C.C. Kemp. El-e: An assistive mobile manipulator that autonomously fetches objects from flat surfaces. *Autonomous Robots*, 28(1):45–64, January 2010.

[11] A. Leeper, K. Hsiao, M. Ciocarlie, L. Takayama, and D. Gossow. Strategies for human-in-the-loop robotic grasping. In *Proc. of Human-Robot Interaction (HRI)*, 2012.

[12] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige. The office marathon: Robust navigation in an indoor office environment. In *Intl. Conf. on Robotics and Automation*, 2010.

[13] B. Pitzer, M. Styer, C. Bersch, C. DuHadway, and J. Becker. J. becker, towards perceptual shared autonomy for robotic mobile manipulation. In *IEEE Intl. Conf. on Rob. and Automation*, 2011.

[14] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng. ROS : an open-source Robot Operating System. In *IEEE Intl. Conf. on Rob. and Automation*, 2009.

[15] M.A. Regalbuto, T.A. Krouskop, and J.B. Cheatham. Toward a practical mobile robotic aid system for people with severe physical disabilities. *J. of rehabilitation research and development*, 29(1), 1992.

[16] A. Remazeilles, C. Leroux, and G. Chalubert. Sam: A robotic butler for handicapped people. In *IEEE Intl. Symp. on Robot and Human Interactive Communication*, 2008.

[17] W. A. Rogers, B. Meyer, N. Walker, and A. D. Fisk. Functional limitations to daily living tasks in the aged: A focus group analysis. *Human Factors*, 1:111–125, 40.

[18] J. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K. Kuchenbecker. Human-inspired robotic grasp control with tactile sensing. *IEEE Trans. on Robotics*, 27, 2011.

[19] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *Intl. Conf. on Computer Vision*, 2011.

[20] Rviz documentation page. http://www.ros.org/wiki/rviz.

[21] R. Schulz. Effects of control and predictability on the physical and psychological well-being of the institutionalized. *Journal of Personality and Social Psychology*, 33:563–573, 1976.

[22] M.S. Searle, M.J. Mahon, S.E. Iso-Ahola, H.A. Sdrolias, and J. Van Dyck. Enhancing a sense of independence and psychological well-being among the elderly: A field experiment. *Journal of Leisure Research*, 27:107–107, 1995.

[23] J.M. Wiener, R.J. Hanley, R. Clark, and J.F. Van Nostrand. Measuring the activities of daily living: Comparisons across national surveys. *Journal of Gerontology: SOCIAL SCIENCES*, 45(6):S229–237, 1990.

[24] Willow Garage, Inc. The PR2 Robot. http://www.willowgarage.com/pages/pr2/overview.