2011 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 25-30, 2011. San Francisco, CA, USA

Two Hands are Better than One: Assisting Users with Multi-Robot Manipulation Tasks

Bennie Lewis and Gita Sukthankar

Abstract— **Multi-robot manipulation, where two or more robots cooperatively grasp and move objects, is extremely challenging due to the necessity of tightly coupled temporal coordination between the robots. Unfortunately, introducing a human operator does not necessarily ameliorate performance due to the complexity of teleoperating mobile robots with high degrees of freedom. The human operator's attention is divided not only among multiple robots but also between controlling a robot arm and its mobile base. This complexity substantially increases the potential neglect time, since the operator's inability to effectively attend to each robot during a critical phase of the task leads to a significant degradation in task performance. In this paper, we propose an approach for semi-autonomously performing multi-robot manipulation tasks and demonstrate how our user interface reduces both task completion time and the number of dropped items over a fully teleoperated robotic system. Propagating the user's commands from the actively-controlled robot to the neglected robot allows the neglected robot to leverage this control information and position itself effectively without direct human supervision.**

I. Introduction

RAP systems combine the unique capabilities of **R**obots, **A**gents, and **P**eople to achieve goals that cannot be completed by any single entity [13]. The effectiveness of a RAP system can be enhanced in a variety of ways including improving task allocation across entities with different capabilities [8], the use of adjustable autonomy to improve the timing of agent intervention [14], [15], and building explicit models of user distraction [1], [10]. However in some cases, tasking (which entity should perform the task) and timing (when should the agents/robots act autonomously) are relatively straightforward to compute. The barrier to adjustable autonomy in RAP system can be selecting effective actions during time segments when the robot is acting autonomously. This is a problem for all autonomous systems operating in uncertain environments, yet RAP systems have options that are unavailable to normal robots. A common solution is to decrease the time period of autonomous operation and increase the amount of user intervention, but in cases where the task is complicated and the user's workload is already high, this approach threatens to degrade the overall system performance. In this paper, we propose an alternate approach in which the agents and robots leverage information about what the user is doing and has recently done to decide their future course of action. We demonstrate our approach on a multi-robot manipulation task that is both difficult to perform autonomously due to sensor limitations and challenging for human teleoperation because of the higher degrees of freedom.

In our multi-robot manipulation task, the user directs a team of two mobile robots to lift objects using an arm and gripper for transport to a goal location. The environment contains a heterogeneous selection of objects, some of which can be transported by a single robot and others that require both robots to lift. Figure 1 shows a picture of the team of robots cooperatively moving an object that cannot be carried by a single robot. This task is a modified version of the multi-robot foraging problem that has been successfully addressed by decentralized task allocation algorithms (e.g., [2]) but with the additional complication that grasp planning and manipulation must be executed by the human due to the sensor limitations of our robotic platform. Like cooperative box pushing [5], multi-robot manipulation requires tight coordination between robots; a poor grasp from either of the robots will often lead to the object being dropped. Our task can be viewed as an abstraction of the type of manipulation task that would be performed by a team of USAR (Urban Search and Rescue) or domestic service robots in which they have to find and move objects in an unstructured environment.

Our mixed-initiative interface provides the user with two important new cooperative functions: 1) autonomous positioning of the second robot (**locate ally**), and 2) a **mirror** mode in which the second robot simultaneously executes a modified version of the commands that the user has issued to the actively controlled robot. When the user requests help to move a large object, these cooperative functions enable the robot to autonomously move to the appropriate location, cooperatively lift the object and drive in tandem to the goal. The locate ally and mirror modes are created through intermittently propagating the user's command history across the robots. The unmanaged robot follows a simple learning-by-demonstration paradigm where it attempts to cooperate with the teleoperated robot, based on the user's current and previous commands.

II. Related Work

Khatib et al. [7] defines the basic manipulation capabilities needed by robots to operate in human-populated environment as being: 1) integrated mobility and manipulation; 2) multi-robot coordination; 3) human-robot interaction; 4) collision-free path planning. Our system focuses on the use of effective human-robot interaction and multi-robot coordination to address deficiencies in the robots' sensing. Note that the human interacts with the robots exclusively through a

B. Lewis and G. Sukthankar are with the Department of EECS, University of Central Florida, 4000 Central Florida Blvd, Orlando, FL 32816-2362 {blewis,gitars}@eecs.ucf.edu

978-1-61284-455-8/11/$26.00 ©2011 IEEE

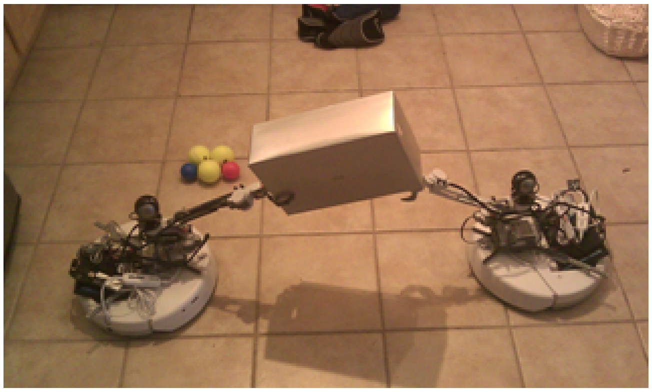


Fig. 1. Two HU-IE robots cooperate to lift an object. One robot is teleoperated while the other moves autonomously to mirror the user's intentions. The user can seamlessly switch robots during such maneuvers.

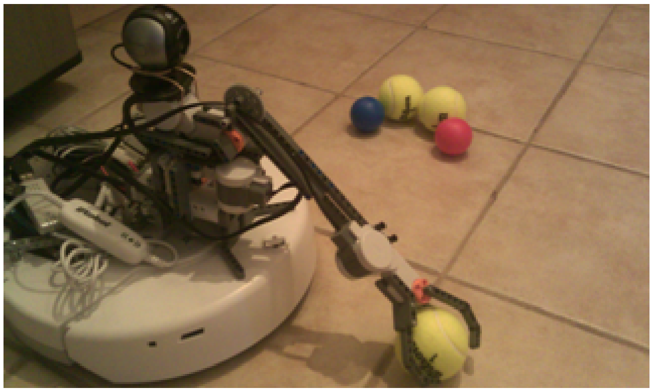


Fig. 2. HU-IE combines a mobile base (3 DOF) with a robotic arm (2 DOF) equipped with a gripper. This enables HU-IE to navigate indoor environments and pick up small objects. The user can wirelessly teleoperate the robot using a webcam.

gamepad controller and graphical user interface rather than cooperatively grasping the object along with the robots as was done in [3]. The mirror mode in our interface effectively functions as a leader-follower architecture similar to [9] with the teleoperated robot serving as the leader.

There are many different ways to structure a RAP (Robots, Agent, People) system. For instance, Kawamura et al. [6] propose an agent-based architecture in which different aspects of the user interface are tasked to separate agents. An alternative approach is to base the human-robot interaction on the same principles that govern effective human-robot teamwork. Hoffman and Breazeal [4] introduce a collaborative interaction architecture to support turn-taking, mutual support, and joint grounding for settings where a human user works alongside an autonomous humanoid robot in a shared workspace. Equipping each of the robots with a separate teamwork proxy that can enable the robots to coordinate among themselves was shown to successfully increase robot autonomy and decrease demands on the human operator in a USAR task [16].

In contrast, our user interface has a single autonomous agent with different modes that can be either executed autonomously or requested by the user; in this paper we demonstrate that injecting these extra modes into the human-robot interaction significantly improves task performance and user satisfaction. Rosenthal et al. [12] propose the notion of a symbiotic human-robot relationship where the user can help the robot overcome its limitations. Similarly, we rely on the human to address some of our robot's deficiencies, particularly in perception. However, our work differs in that the human is not physically present but rather that robot control moves seamlessly between user and the agent, enabling multiple robots to cooperate with a human in-the-loop that drives the system's high-level goals.

III. HU-IE Robot Platform

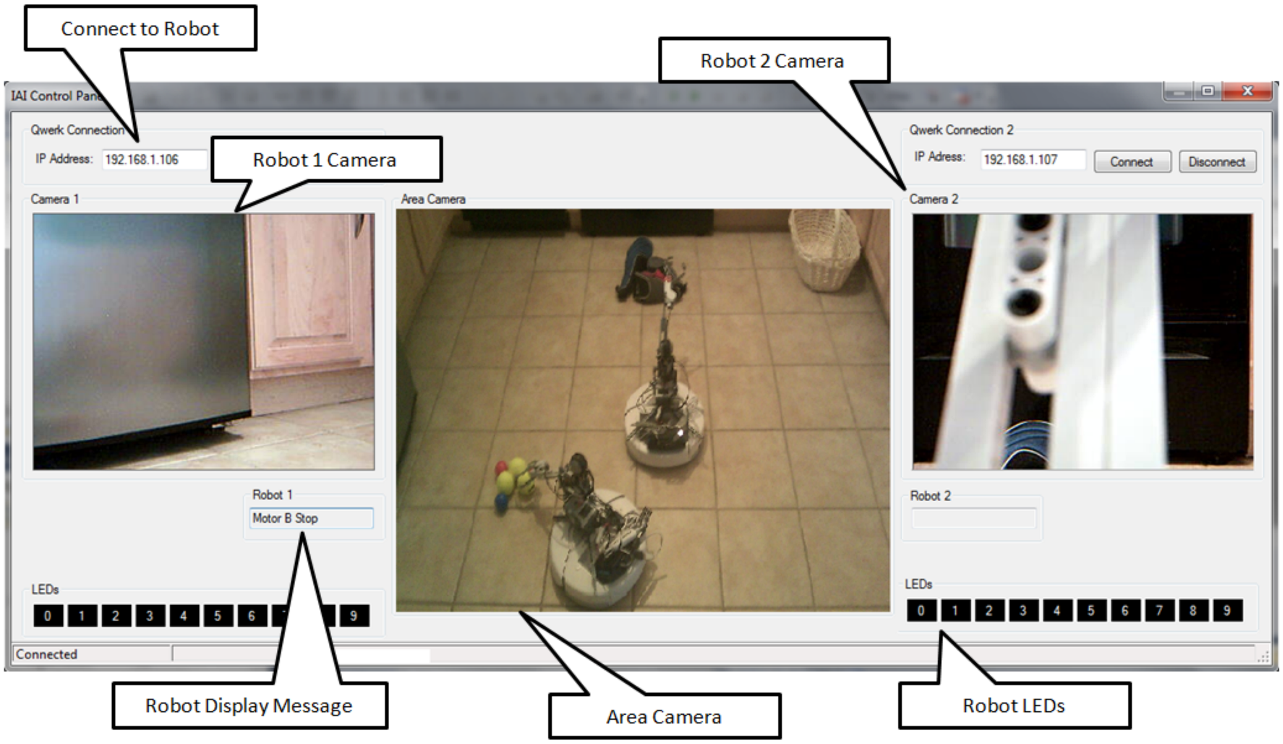
Our robot, the Home and Urban Intelligent Explorer (HU-IE), features a mobile base attached to an arm and gripper (Figure 2). It is designed to be able to retrieve light objects in a household environment with either carpets or hard floors. We constructed our robot using three components: iRobot Create, Charmed Labs' Qwerk board [11], the arm from the NXT 2.0 Robotics Kit, and a Logitech Communicate STX Webcam.

The robotic arm on the HU-IE robot was created using the LEGO NXT Robotic Kit. It is 1.2 feet long and extends 8 inches in front of the robot. The arm is actuated using three motors, can rotate 360° around the robot base and has an operating range of -45°–90° in elevation. At the end of the robotic arm is a four tong claw with rubber grips capable of grasping objects sized for a human hand. An NXT intelligent brick, containing a 32-bit ARM7 microprocessor, functions as the brain of the robotic arm, connecting all the actuators together. Commands from the user interface are sent directly to the robotic arm via Bluetooth, bypassing the Qwerk board. The webcam is mounted on the robotic arm to enable the operator to view the object from the arm's perspective.

IV. INTELLIGENT AGENT INTERFACE

The user views the environment and interacts with the robot team through our user interface (Figure 3), which was designed to minimize teleoperation workload. The operator issues controls to both robots through an Xbox 360 gamepad, using a button to switch between robots. Prior work on human-robot interfaces indicates that gamepad interfaces are generally preferred over keyboards/mice for mobile robot teleoperation [10].

The basic user interface requires the user to fully teleoperate both robots. In this paper, we present and evaluate the Intelligent Agent Interface (IAI) which adjusts its autonomy based on the user's workload. In addition to automatically identifying user distraction, the IAI leverages prior commands that the user has issued to one robot to determine a course of action of the second robot. To enable the human to simultaneously control multiple robots, the interface allows robots to be placed in a **search** mode, where the robot continues moving in the specified direction, while hunting for objects and avoiding obstacles. IAI monitors each of the robots and identifies robots that are ignored by the operator



IP Address: 192.168.1.106

IP Adress: 192.168.1.107

Motor B Stop

Connected

Fig. 3. The intelligent agent interface is designed to enable the user to seamlessly switch teleoperation across multiple robots. The IAI supports a cooperative mode where the agent supports the user's active robot by mirroring its intentions.

through measuring time latencies. It then assumes control of the unattended robot and cedes control if the user sends the robot an explicit teleoperation command.

The IAI provides the user with two important new cooperative functions: 1) autonomous positioning of the second robot (**locate ally**) and 2) a **mirror** mode. In these modes, the unmanaged robot attempts to learn through demonstration the intent of the user by monitoring the command history of the teleoperated robot. The autonomous robot uses a simple set of translation rules to modify the command sequence. In the **locate ally** mode, the robot positions itself on the side adjacent to the teleoperated robot, closest to the nearest pile of objects. For the **mirror** mode, it simply executes the same set of commands, corrected for differences in the robots' orientation.

Robots have the following modes of operation:

Search: the robots wander the area searching for objects.

Help: a robot enters this mode if the human operator calls for help using the gamepad or when the teleoperated robot is near an object too large to be moved by an individual robot.

Pickup: the robot detects an object and requests that the human teleoperate the arm.

Transport: the robot transports an object held by the gripper to the goal.

Locate Ally: the unmanaged robot autonomously moves to a position near the teleoperated robot based on command history.

Mirror: the robot mimics the commands executed by the teleoperated robot to simultaneously lift an object and transport it to the goal location.

In a typical usage scenario, the IAI moves the unattended robot around the environment in search of objects to be moved (clutter). At the start of the mission, the region is roughly partitioned into two areas of responsibility for exploration. Given this partition, each robot independently searches its assigned space. The robot's current state is displayed on the user interface for the benefit of the human operator. When the user needs help manipulating an awkward object, the second robot can be called using the gamepad controller. The **Help** function can also be automatically activated by the IAI system, based on the other robot's proximity to large objects. Once in the **Help** mode, the robot executes the **Locate Ally** behavior. IAI maintains a history of both robots' navigational movements and uses dead reckoning to determine the teleoperated robot's position.[1] Each HU-IE robot has a cliff sensor, which when activated indicates that a robot has been forcibly moved. If that occurs, the IAI system notifies the user to reorient the robot by driving it to its initial starting position. If the user is not actively soliciting help, the unmanaged robot typically moves into the **Search** mode; once the robot detects an object, it notifies the user

[1] In indoor environments, the Create robot base only experiences a little slippage so the robot's position estimates are accurate to within a few cms.

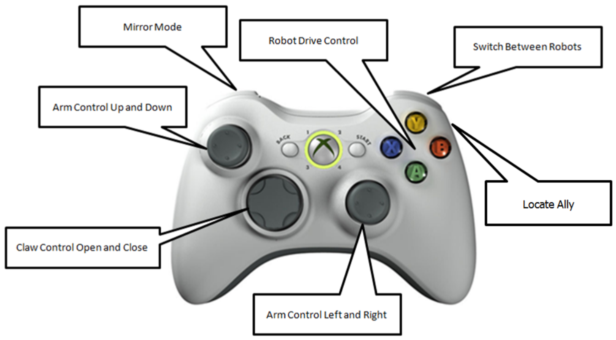


Fig. 4. The physical interface to the IAI is through an Xbox 360 gamepad from which the operator can select a robot and send it teleoperation commands.

that it needs help with manipulation. After the object has been lifted by the user, the robot transports it to the goal. The aim of the IAI system is to smoothly transition between the unmanaged robot *rendering* help to the user and *asking* for help with the manipulation section of the task.

The human operator can also opt to put the other robot into **Mirror** mode. In this mode, the unmanaged robot intercepts the commands given to the teleoperated robot and attempts to duplicate them in its own frame of reference. This mode is essential for reducing the workload of the operator during cooperative manipulation, when two robots are required to lift the object. By combining the **Help**, **Locate Ally**, and **Mirror** modes, the robot can autonomously detect when its help is needed, move to the correct position and copy the teleoperated robot's actions with minimal intervention from the operator.

The operator controls the robots using an Xbox 360 Gamepad controller (Figure 4) as follows. The trigger buttons on the Xbox 360 controller are used to toggle teleoperation between the two robots and to activate the **mirror** mode in the unmanaged robot. The **A**, **B**, **X** and **Y** buttons are used to drive the mobile base. The right button halts the actively managed robot. The left and right analog sticks control the elevation and azimuth, respectively, of the robot arm. The claw grip is controlled by the D-pad on the Xbox 360 controller.

V. EXPERIMENTAL METHODOLOGY

Our user studies are designed to evaluate the performance of the IAI vs. manual teleoperation on a variety of measures, including speed of task completion, number of object drops, and user satisfaction. Three indoor scenarios plus a training scenario were employed in our user study. The user was asked to execute each of the scenarios twice, once using our Intelligent Agent Interface and the other using the basic teleoperation functionality. The scenarios were always presented in ascending order of difficulty, with a randomized ordering of the user interface condition. Participants were allotted 10 minutes of practice time and 15 minutes for each of the scenarios. The scenarios are as follows:

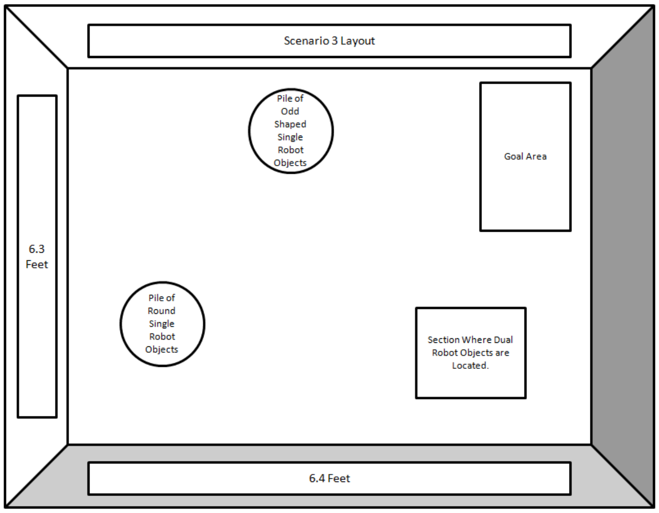

Scenario 3 Layout
Pile of Odd Shaped Single Robot Objects
Goal Area
6.3 Feet
Pile of Round Single Robot Objects
Section Where Dual Robot Objects are Located.
6.4 Feet

Fig. 5. Scenario 3 Layout: the two robots operate in an area of $6.3' \times 6.4'$ and move objects from various piles to the goal area. Objects differ in difficulty and demand different strategies from the robot-user team.

Training: Each participant was given a ten minute training session during which they were able to familiarize themselves with the teleoperation controls and the IAI system. The human operator was allowed to practice picking up objects and transporting them to the goal location.

Scenario 1: For the first task, the participant had to use the two robots to search the area and transport small objects (movable by a single robot) to the appropriate goal. The environment contained three piles with five objects. The first pile had large round objects, the second pile contained oddly-shaped objects, and the third pile contained small round objects.

Scenario 2: In the second scenario, the participant had to locate and retrieve awkward objects that required both robots to simultaneously lift and carry. These objects were grouped in three sections, each containing 3 awkward objects.

Scenario 3: The final mission consisted of a mixture of objects as shown in Figure 5: the first section contained five (small and large) objects; the second had oddly-shaped objects; and the third contained awkward objects that required bimanual manipulation.

The baseline system, designated as manual operation, consisted of a standard teleoperation setup where the human operator controls all aspects of the robot's motion using the Xbox 360 controller for the three scenarios. The user interface is only used to display camera viewpoints, and the robots never attempt to act autonomously. In our proposed approach, the user has access to the additional commands, **help** and **mirror** through the controller. The IAI automatically detects lack of robot activity and triggers the **search**

TABLE I

DEMOGRAPHICS AND EXPERIENCE LEVEL OF THE USER-STUDY PARTICIPANTS.

Age	Gender		PC Game Skill Level		
20–28	Male	Female	Expert	Inter.	Beginner
20	10	10	4	6	10

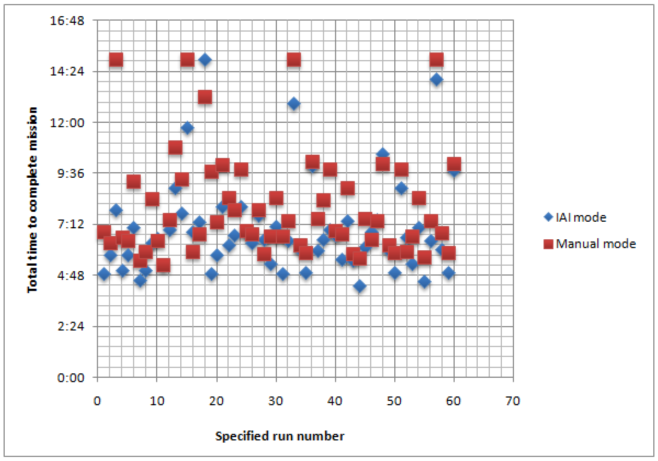


Fig. 6. Scatterplot showing the time required the complete the task in each scenario under the two experimental conditions: IAI (adjustably autonomous) and Manual (pure teleoperation). We observe that in the majority of scenarios, IAI reduces the time required to complete the task.

TABLE II

ANALYSIS OF AVERAGE TASK COMPLETION TIME. IAI RESULTS IN A SIGNIFICANT IMPROVEMENT OVER THE MANUAL MODE IN ALL THREE SCENARIOS.

Scenario	IAI Time $\pm\sigma$ (sec)	Manual Time $\pm\sigma$ (sec)	Significance ($p < 0.05$)
1	361.1 ± 63.2	411.8 ± 81.9	0.017
2	353.6 ± 56.0	407.2 ± 59.4	0.003
3	538.5 ± 157.3	627.8 ± 158.6	0.004

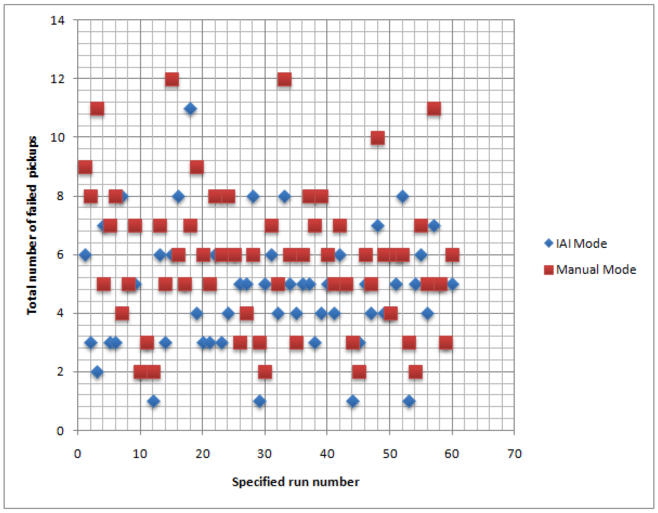


Fig. 7. Scatter plot showing the number of failed pickup attempts for each user in each scenario.

mode to hunt for objects with the unmanaged robot. When an object too large for a single robot is detected, the IAI system autonomously positions the robot and waits for the user to activate the mirror. We measured task performance using two metrics: (1) the time required to complete the task (bounded by 15 minutes); (2) the number of times objects were dropped during the scenario. We present our findings in the next section.

VI. RESULTS

Table I summarizes the demographics of our user study. The majority of the users had limited prior experience with game controller interfaces. Figure 6 presents a scatter plot of the time taken by each user to complete each scenario under the two experimental conditions, pure teleoperation (denoted “Manual mode”) and IAI. We see that in the majority of runs, IAI significantly accelerates the human operator’s progress. We attribute this to the fact that the robot controlled by the IAI continues to assist the human-operator while the teleoperation condition forces the user to inefficiently multi-task between robots.

Table II presents an analysis of the completion time results. We confirm that the improvements in completion time reported by the majority of users are statistically significant under a Student’s one-tailed t-test ($p < 0.05$).

Figure 7 shows the number of failed pickup attempts by the user in each scenario, both under IAI and manual conditions. Table III analyzes these results to test our belief that IAI should result in fewer dropped objects. We see that the number of average drops is lower with IAI in all three scenarios. However, the improvement in Scenario 1 is not shown to be statistically significant on the Student’s t-test ($p < 0.05$). In general, IAI results in fewer drops because the mirror mode enables the user and agent to coordinate grasping and movement, whereas in manual mode the user risks dropping the object as a robot is moved during two-handed manipulation task.

Our user post-questionnaire study indicated a strong preference (90%) for IAI over the manual teleoperation mode; the remaining 10% expressed no preference between the two conditions.

A. User Comments

As part of our post-task questionnaire we asked subjects about their strategies for collecting objects. The comments

TABLE III

AVERAGE TOTAL COMBINED FAILED PICKUPS (OBJECT DROPS). IAI REDUCES THE NUMBER OF DROPS ON AVERAGE; THIS IMPROVEMENT IS STATISTICALLY SIGNIFICANT IN SCENARIOS 2 AND 3.

Scenario	IAI Failed Pickups $\pm\sigma$	Manual Failed Pickups $\pm\sigma$	Significance ($p < 0.05$)
1	5.75 ± 1.55	6.20 ± 1.64	NO
2	3.30 ± 1.21	4.70 ± 1.55	0.002
3	5.00 ± 2.22	6.80 ± 3.31	0.026

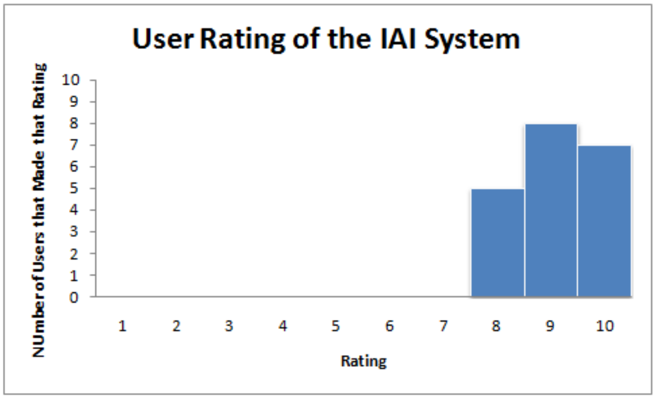


Fig. 8. Histogram of user ratings of the IAI user interface on post-task questionnaires. Users were overwhelmingly positive.

indicate that the users were definitely relying upon IAI for either 1) autonomously transporting objects or 2) coordinated object pickup. On the post-questionnaire, 90% of the participants expressed a preference for IAI when queried about their preferences and all of the participants gave IAI high ratings (Figure 8).

From our own observations, we noted that many of the users spent the entire time controlling a single robot and exclusively relied upon IAI for controlling the second robot's navigation. When the IAI system notified the user to pick up an object, most users switched to that robot, used the arm to lift the object, and then chose to immediately return to controlling the original robot. At that point, IAI would autonomously deliver the object to the goal. Some users chose to call for assistance on each pile of objects, instead of assigning robots to different piles. Many participants experienced some initial difficulty during the training period and first scenario learning how to lift objects with the arm. By the second scenario, most users learned the knack of controlling the arm, resulting in fewer object drops. Users experienced more problems in manual mode while lifting the larger objects as compared to IAI.

VII. CONCLUSION AND FUTURE WORK

Adding manipulation capabilities to a robot team widens its scope of usage tremendously at the cost of increasing the complexity of the planning problem. By offloading the manipulation aspects of the task to the human operator, we can tackle more complicated tasks without adding additional sensors to the robot. In this paper, we demonstrate and evaluate an effective human-robot interface paradigm for multi-robot manipulation tasks. Rather than increasing the workload of the human user, we propose an alternate approach in which the robots leverage information from commands that the user is executing to decide their future course of action. We illustrate how this approach can be used to create cooperative behaviors such as mirroring and locate ally; together the robots can coordinate to lift and transport items that are too awkward to be manipulated by a single robot. In the user study, our mixed-initiative user interface (IAI) shows statistically-significant improvements in the time required to perform foraging scenarios and the number of dropped items. Users were able to master the interface quickly and reported a high amount of user satisfaction. In future work, we plan to generalize our method of learning-by-demonstration. Currently, our system employs a static translation system for converting the users' commands. A more powerful approach would be to use either plan repair or case-based adaptation to modify the user's commands for use by the partner robot.

VIII. ACKNOWLEDGMENTS

This research was supported in part by NSF award IIS-0845159.

REFERENCES

[1] X. Fan and J. Yen. Realistic cognitive load modeling for enhancing shared mental models in human-agent collaboration. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2007.

[2] D. Goldberg, V. Cicirello, M. B. Dias, R. Simmons, S. Smith, and A. T. Stentz. Market-based multi-robot planning in a distributed layered architecture. In *Multi-Robot Systems: From Swarms to Intelligent Automata: Proceedings from the 2003 International Workshop on Multi-Robot Systems*, volume 2, pages 27–38. Kluwer Academic Publishers, 2003.

[3] Y. Hirata, Y. Kume, Z. Wang, and K. Kosuge. Decentralized control of multiple mobile manipulators based on virtual 3-D caster motion for handling an object in cooperation with a human. In *International Conference on Robotics and Automation*, 2003.

[4] G. Hoffman and C. Breazeal. Collaboration in human-robot teams. In *Proceedings of AIAA Intelligent Systems Technical Conference*, 2004.

[5] N. Kalra, D. Ferguson, and A. T. Stentz. Hoplites: A market-based framework for planned tight coordination in multirobot teams. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 1170–1177, April 2005.

[6] K. Kawamura, P. Nilas, K. Muguruma, J. Adams, and C. Zhou. An agent-based architecture for an adaptive human-robot interface. *Hawaii International Conference on System Sciences*, 2002.

[7] O. Khatib, K. Yokoi, O. Brock, K. Chang, and A. Casal. Robots in human environments: Basic autonomous capabilities. *The International Journal of Robotics Research*, 18(7):684–696, 1999.

[8] M. Koes, I. Nourbakhsh, and K. Sycara. Constraint optimization coordination architecture for search and rescue robotics. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) 2006*, pages 3977–3982, May 2006.

[9] K. Kosuge and T. Oosumi. Decentralized control of multiple robots handling an object. In *International Conference on Intelligent Robots and Systems (IROS)*, 1996.

[10] B. Lewis, B. Tastan, and G. Sukthankar. Improving multi-robot teleoperation by inferring operator distraction (extended abstract). In *Proceedings of International Conference on Autonomous Agents and Multi-agent Systems (AAMAS)*, 2010.

[11] Qwerk robot platform. `http://www.charmedlabs.com/`.

[12] S. Rosenthal, J. Biswas, and M. Veloso. An effective personal mobile robot agent through symbiotic human-robot interaction. In *Proceedings of International Conference on Autonomous Agents and Multi-agent Systems (AAMAS)*, 2010.

[13] P. Scerri, L. Johnson, D. Pynadath, P. Rosenbloom, N. Schurr, M. Si, and M. Tambe. Getting robots, agents, and people to cooperate: An initial report. In *AAAI Spring Symposium on Human Interaction with Autonomous Systems in Complex Environments*, 2003.

[14] P. Scerri, D. Pynadath, and M. Tambe. Adjustable autonomy for the real world. In *Agent Autonomy*, pages 163–190. Kluwer, 2003.

[15] M. Sierhuis, J. M. Bradshaw, A. Acquisti, R. van Hoof, R. Jeffers, and A. Uszok. Human-agent teamwork and adjustable autonomy in practice. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2003.

[16] J. Wang, M. Lewis, and P. Scerri. Cooperating robots for search and rescue. In *Proceedings of AAMAS Workshop on Agent Technology for Disaster Management*, 2006.