Mobile Robot Control Using a Small Display

William A. Hoff, Joshua C. Lisle
whoff@mines.edu, jcl@profitool.com
Colorado School of Mines
Golden, Colorado
USA

Abstract

This paper addresses the problem of controlling a mobile robot using a small, portable user interface. In some applications, including security and military, a user might command a mobile robot in the field using a wearable computer. The wearable computer may have a small display that would make information delivery difficult. Also, full video feedback from the robot may not be available due to radio bandwidth limitations.

One solution is to have the user interact with a virtual model of the robot environment (available from other sources). We designed an interface to interact with such a model, using a small display. The user controls the robot by gesturing with a single hand. The gestures are sensed using an instrumented glove and a 6 DOF (degree of freedom) magnetic tracker mounted on the user's hand. The interface was tested using two different tasks: a 6 DOF manipulation task and a visualization task which was primarily 3 DOF. For comparison, a simple, 2 DOF input device was also implemented. It was found that the 6 DOF input device was clearly superior for the manipulation task, but results were inconclusive for the visualization task.

1 INTRODUCTION

Many mobile robot applications, including military and security, require a person to remotely operate a robot in the field, using a small, portable interface. For example, the Tactical Mobile Robotics (TMR) program, sponsored by the US Defense Advanced Research Projects Agency (DARPA), developed small mobile robots to provide foot soldiers with information on his or her surroundings [1]. The goal was to make the robots semiautonomous, and supervised by a soldier using a small wearable computer.

A head mounted display could be used for a user interface, but it may obscure the user's vision. Another option, which we used, is to mount a small display on the user's wrist (Figure 1). This allows the user to detach from the system quickly, cognitively and physically, which may be important in some applications.

To communicate with the robot, a radio frequency (RF) communications link may be used. However, an RF link may not have enough bandwidth for full video streams, especially when operating over long distances and passing through obstacles such as buildings.

Figure 1 User interface with wrist-mounted display and data glove.

An alternative approach is to have the user control the robot using a virtual model of the robot and its environment [2, 3]. The robot could create a model of its environment from sensor data. Then, only occasional updates to the model need to be transmitted from the robot to the user. Creating a model from sensor data is difficult and the subject of much research. In our work, we assume that such a model is available (although it may be incomplete and contain errors).

To interact with the virtual model, our system allows the user's hand to control the pose of a three dimensional (3D) "cursor" [4], which is displayed as a "virtual" hand in the scene. This device is used to manipulate the virtual robot model as well as change the user's viewpoint.

Using a very small display for visualization presents difficulties in visualizing and manipulating the virtual world [5-7]. The objective of our work was to develop an interface that allowed efficient manipulation and visualization on a small display. Our approach, described in detail in Section 3, was to allow the user to freely move about the virtual world to aid in visualization.

In our application, we were interested in not only visualizing the scene, but in performing a manipulation task with the robot. Our virtual robot had a 6 degree of freedom (DOF) manipulator arm that we had to control with the interface. Because we had to control multiple degrees of freedom simultaneously, we chose the user's hand as an input device. The user's hand is tracked by a magnetic tracker (Figure 1), and hand gestures are sensed with an instrumented glove, or "data glove". This has the additional benefit of allowing the user to hold another object with their other hand, which may be important in some applications.

Our interface is novel in that it uses a very small display for robot control, and allows natural control of the user's viewpoint as well as control of a manipulator, through the user's hand. We also performed a series of experiments to evaluate the interface and compare it to an alternative, more conventional, input device.

2 RELATED WORK

Visualizing 3D information on small 2D displays is difficult due to the small angular field of view that the display presents to the user's eye. As a result, the user may have difficulty discerning the depth of 3D objects, and keeping track of the relative location of objects that are not simultaneously displayed.

Depth cues in rendered scenes can be created using standard computer graphics techniques such as perspective display, shading and occlusion. Also, stereo glasses can reveal the depth of objects through binocular disparity. Hu, et al, [8] found that stereoscopic viewing and shadows had statistically significant effects on depth perception, using a high resolution head mounted display. However, stereoscopic viewing requires additional hardware. Also, since we wanted to use a small wrist mounted display, and it was not clear if the range of disparity on a small display could provide adequate depth perception.

On the other hand, allowing the user to move through the virtual world improves depth perception with small displays [6]. However, there is a cost in the increased time needed for exploration. Allowing motion also improves the user's sense of engagement [9]. This is convenient because our application required the user to move through the virtual world anyway.

With respect to input devices, magnetic trackers have proven useful for determining the 6 DOF pose of an object (*e.g.*, the hand) with respect to some other reference frame (*e.g.*, a sensor located on the body). Although nearby metallic objects will disrupt the magnetic field and produce erroneous measurements, this may be acceptable for qualitative motions.

Two DOF input devices such as a joystick or mouse may be easier and cheaper to implement, but they require mapping the 2 DOF commands to 3 DOF or 6 DOF space [10]. This is not very intuitive and requires some cognitive effort and mode switching on the part of the user. Our experiments, described in Section 4, evaluated the tradeoffs between 2 DOF and 6 DOF input devices.

Sayers and Paul [2] note that the interactions with a virtual robot arm are more effective if the arm's motions, as seen by the viewpoint, match the user's hand motions. This correspondence, called isomorphism, reduces cognitive effort on the part of the operator. Zhai *et al* [4] argue that the mathematical complexity of the transformation from the controller space to the virtual space determines the difficulty for the user.

Methods for viewpoint control include gaze-directed steering and hand-directed steering. Gaze directed steering is not appropriate for small displays because the narrow field of view greatly constrains the direction of the user's gaze. For hand-directed steering, metaphors for setting the viewpoint include "scene-in-hand" and "flying vehicle". The scene-in-hand metaphor allows the user to externally view a scene he holds and moves in a virtual hand. This is well suited for viewing and manipulating closed objects but poor for inspecting interiors of objects. The flying vehicle metaphor involves the user flying a virtual vehicle through a virtual world. This metaphor seemed most appropriate for our application.

3 SYSTEM DESCRIPTION

This section briefly describes the design of the system (complete details are in [11]). For development and testing, we used a desktop PC to display graphics on a 4" color LCD monitor (VGA resolution); although in an actual application the desktop PC could be replaced with a wearable computer.

A Polhemus Isotrak Magnetic Tracker was used to track the user's hand. The data glove (MindTel, Inc.) had resistive bend sensors to sense finger flexion (Figure 2, left). In our experiments, we sensed only the thumb, index and fore fingers.

A simple button type input device (Gravis game pad) was also implemented as a comparison (Figure 2, right). It had a two DOF toggle and four buttons. Two of the buttons were utilized as a third degree of freedom. The remaining two buttons were for mode switching.

Figure 2 Alternative input devices: data glove (left) and game pad (right).

The user interface was written in Java3D, an object-oriented toolkit built upon OpenGL. Java3D is based on the concept of a scene graph. This is a treelike data structure used to store, organize and render 3D scene information. The nodes in the graph represent objects to be displayed, aspects of the environment of the virtual world or coordinate transformations. Thus, the scene graph represents the 3D model of the robot (which is known) and the robot's environment (which we assume has been constructed from sensor data).

To test the interface, two sample environments were created – one for a manipulation task and the other for a visualization task.

3.1 Manipulation Task

The manipulation task required the user to pick up a dumbbell-shaped object, using the robot's manipulator arm and place it in a box (Figure 3).

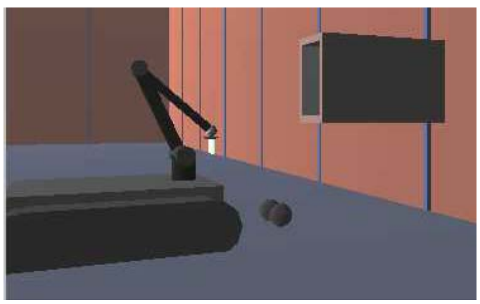

Figure 3 Manipulation task.

The user's hand, tracked by the magnetic sensor, controls the pose of a 3D cursor (a virtual hand) in the scene at a rate of about 30 Hz. When the game pad is used instead, it manipulates the pose of the cursor.

To control the robot manipulator, the user moves the virtual hand until it is close to the robot gripper, or end effector. A "behavior" node in the scene graph continually monitors the distance of the hand from the end effector. When the hand is close enough to "grab" the effector, this behavior changes the color of the effector and the hand (Figure 4).

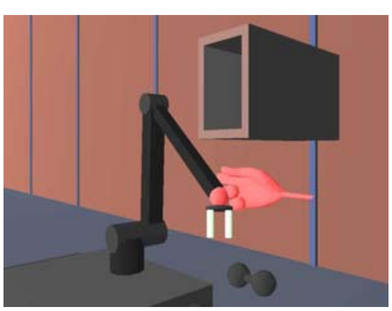

Figure 4 When the hand is close enough to grab the effector, both are illuminated.

While in this "grab" condition, another behavior detects a particular gesture from the data glove[1]. This behavior will cause the hand to disappear, and the end effector of the robot arm will subsequently follow the movement of the user's hand in an isomorphic fashion. Another behavior calculates inverse kinematics and changes the arm's joint angles in order to achieve the desired pose of the end effector.

The user may release control of the end effector by making a different gesture. This is useful when the user's hand has reached its physical limits – the user can release the effector, move his hand to a more comfortable pose, and then re-grab the effector.

When the end effector gets close enough to the dumbbell to pick it up (and its prongs are oriented correctly), a behavior illuminates the dumbbell, indicating the user may grab it (Figure 5). The user makes another gesture with the data glove, which causes the dumbbell to move with the end effector.

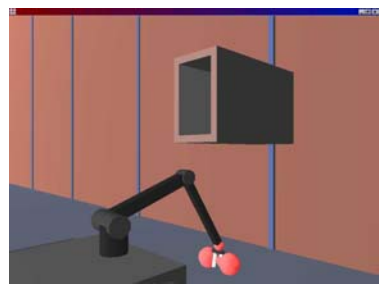

Figure 5 The dumbbell is illuminated when the end effector is close enough to grab.

The arm behavior also detects collisions between the dumbbell and the walls of the virtual box. If a collision is detected, the arm behavior stops the motion of the dumbbell into the box, illuminates the sides of the box that are in collision (Figure 6) and continuously rings a bell while in collision. To start the arm moving again, the user must move the effector in a direction that will take the dumbbell out of collision with the box.

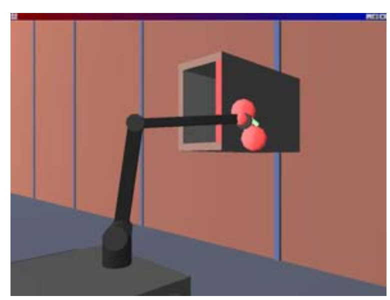

Figure 6 A collision with the side of the box.

[1] When the gamepad is used as the input device instead of the dataglove, gestures are replaced by button presses.

The goal of the manipulation task is to place the dumbbell completely inside the box. When the dumbbell is completely inside, we signify this by stopping the arm motion, releasing the hand from the end effector, and changing the color of the box to green (Figure 7).

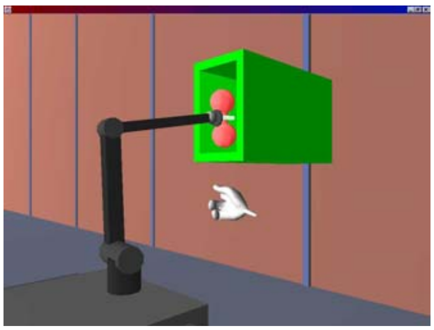

Figure 7 Task completion signified by green box.

3.2 Visualization Task

The second task was a visualization task. The scene consisted of two mobile robots, three large drums and a target on one of the walls of a room (Figure 8).

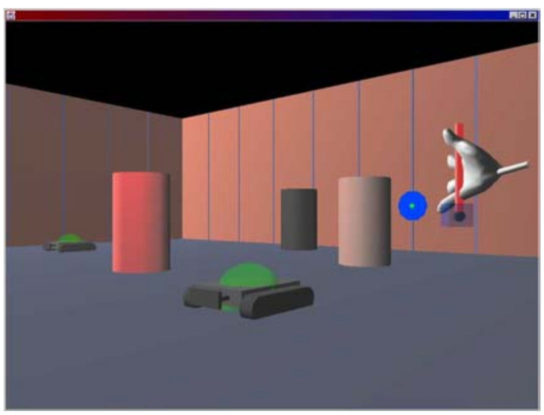

Figure 8 Visualization task.

The task required the user to determine whether a line-of-sight existed between each robot and the target (or whether it was blocked by one of the drums). To do this, the user had to move the viewpoint so that it was located at each robot, and then see if the target was visible.

To control the viewpoint, the interface has a control stick which gives the user the ability to fly through the scene. The user must first grab the stick with the virtual hand. Similar to the manipulation task, a behavior monitors the pose of the hand with respect to the stick. When the hand has the proper orientation and is close enough, the stick changes color (Figure 9, left). The user makes a gesture to grab the stick, which causes the appearance of the hand to change, and the stick subsequently follows the motion of the hand (Figure 9, right).

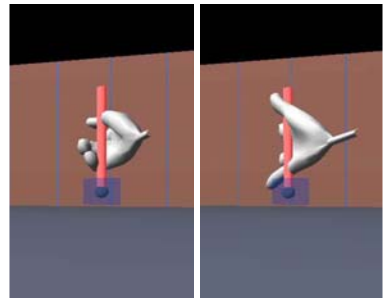

Figure 9 (Left) Just prior to acquisition of the virtual stick. (Right) After acquisition.

The displacement of the stick from its "null" position controls the velocity of the camera in the scene, isomorphically with respect to the viewpoint. To avoid excessive velocity, the magnitude of the translation of the stick is limited. For rotation, yaw was unlimited, pitch was limited to ±10º and no roll was allowed (Figure 10).

There is a "dead zone" around the null pose of the stick, so that small accidental displacements of the stick do not cause unwanted "creep". The magnitude of the deflection past the edge of the null region determines the translation rate (Figure 10).

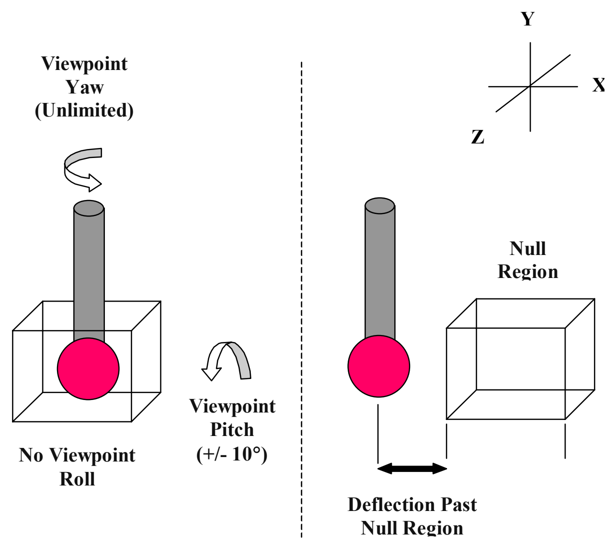

Figure 10 Virtual stick operation.

Our visualization task required the user to position the viewpoint on top of each robot. To aid in this task, we display a translucent green dome on top of the robot (Figure 11, left). Once the viewpoint is within this dome, the dome disappears and the top of the robot turns bright green (Figure 11, right). In addition, the "dead zone" around the stick is made larger so that the user does not accidentally drift off of the robot.

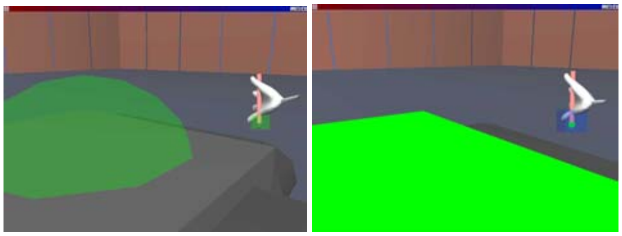

Figure 11 (Left) Approaching robot. (Right) Arrived.

Once at the first robot, the task requires that the user rotate in place and look for a target on the room's wall. The target may or may not be obscured by a drum. Figure 12 shows a view of the target from the robot.

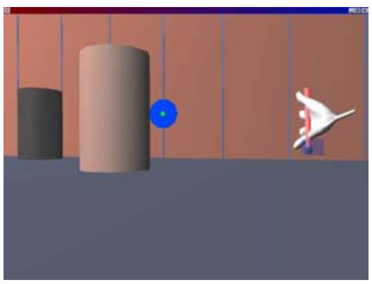

Figure 12 Target is visible from robot viewpoint.

Once the user has scanned the first wall, he states whether the target is visible. He then continues the turn and looks for the second robot, and states whether the second robot is visible or not from the current position. He then proceeds toward the second robot, and once there, determines whether the target is visible from that robot.

4 EXPERIMENT DESCRIPTION

Three right-handed test subjects were used in the experiments. Since the entire testing phase would take several hours, the subjects were seated in an office chair for the experiment (Figure 13), with the display on a tripod. The subjects were seated as close as possible to the 4" monitor without it interfering with their arm motion. The test conductor watched the experiment in progress on a larger monitor.

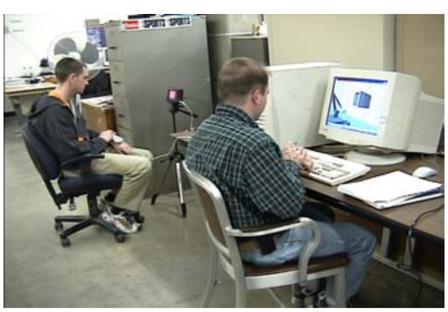

Figure 13 Test subject (left) and conductor (right).

For the manipulation task, a total of 36 scenes were created. Six scenes were used for training, and the rest for testing. The scenes differed from each other with respect to several variables: viewpoint, robot distance from wall, box height from floor and robot offset from the centerline of the box. These variables were constrained so that the robot arm could easily grab the dumbbell and easily place it within the box. Each scene was performed with the 6 DOF magnetic tracker and with the game pad.

The resulting task times are shown in Figure 14. Error bars indicate the 95% confidence level. The results clearly show that for each subject, the task times were significantly less using the 6 DOF hand tracker input device (TR) than using the 2 DOF game pad input device (GP).

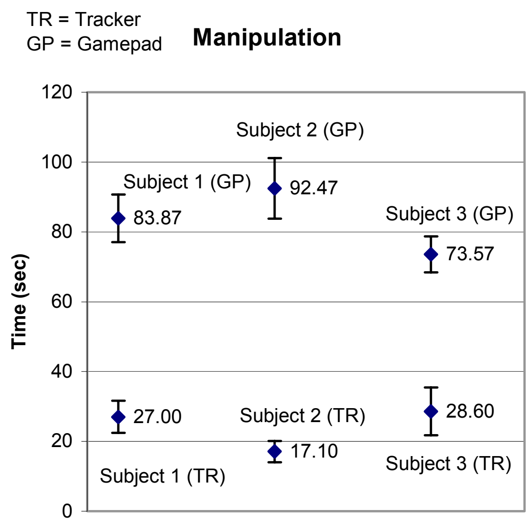

Figure 14 Task times for manipulation task.

For the visualization task, a total of 36 scenes were also used. The scenes differed from each other in several respects. The distances between vantage points changed from scene to scene, but the total distance "flown" was constant. The direction of the initial turn and its magnitude varied from scene to scene, but the total rotation per scene was constant and the number of left and right turns was equal.

As Figure 15 shows, the results were mixed. In all three subjects, the trial confidence intervals did not overlap. However, subjects 1 and 3 had lower tracker times (TR) but subject 2 was more proficient with the game pad (GP). It was interesting to note that subject 2 was the most proficient with the tracker and least proficient with the game pad in the manipulation exercise. More testing would be necessary to make any conclusions.

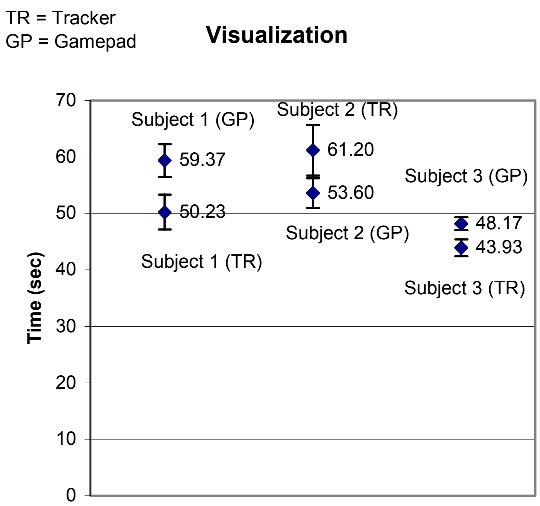

Figure 15 Task times for visualization task.

5 CONCLUSIONS

This paper described a novel interface for manipulating a virtual model of a robot and its environment, using a very small display. The user's hand, tracked by a magnetic tracker, was used to manipulate objects in the virtual world. A data glove and various gestures allowed the user to grasp and release objects in the model. An alternative 2 DOF input device was also implemented as a comparison. To determine the interface's ease of use, the primary and alternative input devices were tested in two dissimilar tasks.

The manipulation task was primarily a 6 DOF task. The visualization task was primarily a 3 DOF task. Experiments showed that the tracker input device was significantly faster than the game pad for the manipulation task, although not for the visualization task.

We believe that the 6 DOF control afforded by the tracker was well suited for the manipulation task. Using the game pad for this task, by virtue of having translation-only or rotation-only modes available, was challenging. A large amount of time was spent switching between these degrees of freedom.

The visualization trials showed less disparity between the two input devices. This task was essentially planar. By carefully alternating between rotation and translation, the test subjects could efficiently use the game pad to complete the task. If rotations were precise, mode switching (the biggest time consumer in the manipulation task) could be minimized. Also, limited DOF actually helped the game pad. When using the tracker to rotate in place on the robots, the subjects often inadvertently translated off and had to reposition themselves. The game pad, by virtue of only being able to rotate or translate at one time, had no such issues.

6 ACKNOWLEDGEMENTS

We thank the support of SAIC Corp. and the participation of the test subjects. The magnetic tracker was loaned by Wendell Chun of Lockheed-Martin Corp.

7 REFERENCES

[1] E. Krotkov and J. Blitch, "Defense Advanced Research Projects Agency (DARPA) Tactical Mobile Robotics Program," *International Journal of Robotics Research*, vol. 18, pp. 769-776, 1999.

[2] C. P. Sayers and R. P. Paul, "An Operator Interface for Teleprogramming Employing Synthetic Fixtures," *Presence*, vol. 3, pp. 309-320, 1994.

[3] T. Blackmon and L. Stark, "Model-based Supervisory Control in Telerobotics," *Presence*, vol. 5, pp. 205-223, 1996.

[4] S. Zhai, et al., "Anisotropic Human Performance in Six Degree-of-Freedom Tracking: An evaluation of Three-Dimensional Display and Control Interfaces," *IEEE Transactions On Systems, Man, and Cybernetics - Part A: Systems And Humans*, vol. 27, pp. 518-528, 1997.

[5] S. Feiner and A. Shamash, "Hybrid User Interfaces: Breeding Virtually Bigger Interfaces for Physically Smaller Computers," *Proc. of ACM Symp. on User Interface Software and Technology*, pp. 9-17, 1991.

[6] G. W. Fitzmaurice, et al., "Virtual Reality for Palmtop Computers," *ACM Transactions on Information Systems*, vol. 11, pp. 197-218, 1993.

[7] H. Noma, et al., "A Palmtop Display for Dextrous Manipulation with Haptic Sensation," *Proc. of Human Factors in Computing Systems*, pp. 126-133, 1996.

[8] H. H. Hu, et al., "Visual Cues for Perceiving Distances from Objects to Surfaces," *Presence*, vol. 11, pp. 652 - 664, 2002.

[9] R. Balakrishnan and G. Kurtenbach, "Exploring Bimanual Camera Control and Object Manipulation in 3D Graphics Interfaces," *Proc. of Human Factors in Computing Systems*, pp. 56-63, 1999.

[10] C. B. Phillips, et al., "Automatic Viewing Control for 3D Direct Manipulation," *Proc. of Symposium on Interactive 3D graphics*, Cambridge, MA, 1992.

[11] J. C. Lisle, "Design And Evaluation Of A Virtual Reality Interface For Mobile Robot Control Using A Small Display," in *Engineering Division*. Golden, CO: Colorado School of Mines, 2002, pp. 72.