

# Ecological Displays for Robot Interaction: A New Perspective

Bob Ricks  
Computer Science Department  
Brigham Young University  
Provo, UT USA  
cyberbob@cs.byu.edu

Curtis W. Nielsen  
Computer Science Department  
Brigham Young University  
Provo, UT USA  
curtisn@cs.byu.edu

Michael A. Goodrich  
Computer Science Department  
Brigham Young University  
Provo, UT USA  
mike@cs.byu.edu

**Abstract**—Most interfaces for robot control have focused on providing users with the most current information and giving status messages about what the robot is doing. While this may work for people that are already experienced in robotics, we need an alternative paradigm for enabling new users to control robots effectively. Instead of approaching the problem as an issue of what information could be useful, the focus should be on presenting essential information in an intuitive way. One way to do this is to leverage perceptual cues that people are accustomed to using. By displaying information in such contexts, people are able to understand and use the interface more effectively. This paper presents interfaces which allow users to navigate in 3-D worlds with integrated range and camera information.

## I. INTRODUCTION

Robot control can be difficult for a variety of reasons. One of the major reasons is that remote operators lack ordinary visual cues that help them navigate and locate things. This manifests itself in reduced ability to a) maintain self-orientation and b) accurately judge distances to objects. Another hindrance to robot operation is communications delay due to the fact that the robot is often at some distance from the operator. Because of communication delays, limited bandwidth and sensor update times, the human may not see the results of commands sent to the robot for some time. Lack of visual cues and delay contribute to loss of situation awareness [1] and mental load on robot operators.

Our informal studies have shown that one of the hardest things for robot operators to do is to keep track of obstacles just outside their camera view. One reason for this is that delays require human operators to remember the commands they have given to the robot until they see the effects of those commands in the interface. Another reason is that sensor information, especially video, is typically only updated a few times per second. This requires the human to mentally connect new information with the commands they have given the robot and their memory of the robot's previous state. The mental load required to keep track of robot pose and compensate for delay adversely affects the operator's ability to effectively control the robot.

One common method for dealing with delay is to use prediction. For example, airplanes use a "tunnel-in-the-sky" to help pilots stay on their flight plan [2]. Another type of predictive display, known as a quickened display,

has also been used for navigation [3]. The difference between quickening and prediction is that prediction shows the current state of the system and a prediction of what will be happening in the future. By contrast, quickened displays only show the predicted future. The reasoning behind leaving out the current state of the system is that "current error contains no information that is useful for correction" [3, Pg. 409].

The most common method for dealing with lack of perspective in video images is to have range sensors give approximate positions of objects in the the area around the robot. Range information is typically in a separate display which the user must integrate with the video for localization purposes. This requires users to divide attention between multiple displays which increases cognitive load and takes time to learn.

This paper proposes an interface which combines prediction with a spatial representation of range information using 3-D graphics. This provides users with an intuitive way of visualizing a robot's position relative to obstacles around it and what will happen as the robot performs actions in the world. In our tests, this interface improved operators' ability to control the robot without adding complexity to the robot or its sensors.

## II. ON IMPROVING TELEOPERATION

There are many reasons to study teleoperation, especially from the standpoint of improving the user interface. Teleoperation can be the most effective way to control mobile robots because it is easy to implement and easy for people to understand. Teleoperation is also a very simple autonomy level that allows us to study the interface itself apart from the intelligence derived from autonomy. Further intelligence can be added to the robot while keeping the benefits of an improved display. Therefore we are using teleoperation as a basis upon which to study human-robot interaction.

Many other methods have been developed to make robots easier to teleoperate. Supervisory control, which involves a human supervising semi-autonomous robots, is one such method. Sheridan's book [4] is a good reference on supervisory control. Many others have worked on supervisory control, safeguarded control [5], [6] and adjustable autonomy [7]. These approaches can actually be used in

conjunction with our interface. Since adding intelligence to the robot makes it harder to model and to study the effects the interface itself has on performance, this paper focuses on simple robots which do not have any autonomy.

Some effort has also gone into improving the visual experience afforded human operators. One method is to use a panspheric camera, which gives a distorted view of the entire region around the robot [8], but can be de-warped to look more natural. This has many advantages, including the ability to visually find and track landmarks. A high-bandwidth communication channel is necessary to allow frequent image updates for the user to maintain continuity between images. In order to limit the hardware requirements and to focus on the effects of the interface on performance, only robots with a single forward-looking camera were used in this paper.

Virtual reality could also be used to control a robot. However, there are two problems we see with this approach. First, virtual reality requires an accurate model of the world in order to work. Second, too much visual information can overload operators with information that is not really important.

This paper takes an approach more along the lines of augmenting virtuality. Instead of adding complexity to the robot or its sensor suite, we simply display rudimentary sensor information in a way that is easy for people to understand. First, we show a representation of the robot in a world of obstacles which represent range data from the sonars and the laser range-finder. This is done in 3-D from a tethered perspective a little above and behind the robot [3]. The second display element is the most recently received image from the robot's camera (see Figure 1). Finally, the display is quickened which allows the operator to see the effects of their actions right away. Quickening is accomplished by moving the camera and the robot in the virtual world. The latest image from the robot also moves to line up with where it would have come from in the robot's current field of view.

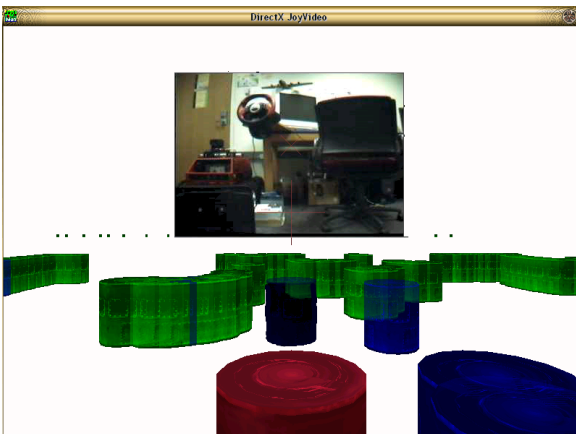


Fig. 1. The Ecological Display.

By integrating the latest images from the robot with a representation of the robot in a field of obstacles, the operator gets a better idea of where obstacles are in the world.

The reason for using a tethered perspective is that the egocentric aspects of the tethered display make it natural to use for navigation, while pulling the viewpoint back enables the operator to integrate spatial information. By quickening the display, the operator is better able to control the robot because they no longer need to remember as much and do as much prediction about where the robot has moved. The display gives users a more intuitive understanding of what is happening in the world by taking advantage of their natural spatial reasoning and 3-D visualization.

### III. HOW THE PREDICTION WORKS

Ten times per second the joystick code sends a joystick movement command to the robot. This command includes a forward velocity, angular velocity and a timestamp. In addition to being sent to the robot, each command is stored in a queue in the interface program. Because of bandwidth constraints, the robot may send image and range data at a different rate. Information packets from the robot include the timestamp from the last joystick command the robot received. Commands in the interface queue with timestamps earlier than the one received in the latest sensor update are discarded because these commands will no longer influence how the robot will move.

New sensor information is quickened by predicting how the robot has moved. Prediction is accomplished by extrapolating where the robot will be after executing the commands currently in the command queue of the interface. As a reasonable approximation, we assume that commands are executed on the robot for the amount of time between when the command was sent from the joystick process and the time the next one was sent. The most recently issued command is handled a little differently. Prediction based on the most recently issued command uses the amount of time since the command was sent to the robot instead of the amount of time we predict it will be processed on the robot. This allows the prediction to be linear instead of jumping to a new position every time we send a new command to the robot.

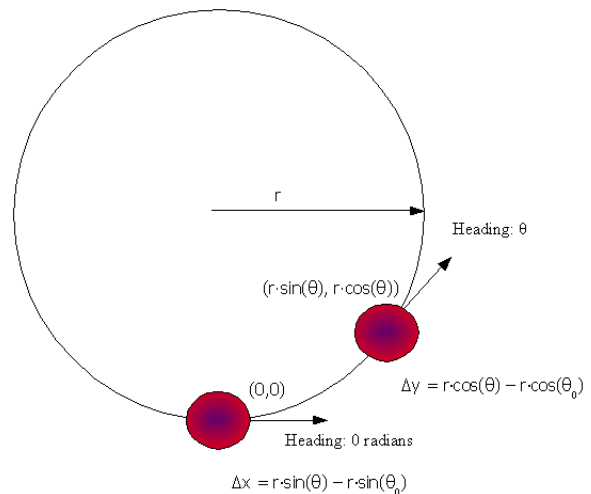


Fig. 2. Ideal Prediction.

Since movement commands sent to the robots consist of a desired translational velocity and a desired angular velocity, dead-reckoning predictions are fairly easy. The robot starts out at the origin, which is defined as the pose of the robot where it collected the latest sensor information. The robots we are using take translational velocity ( $V_x$ ) and angular velocity ( $\omega$ ) command inputs. From this, the change in x position,  $\Delta x$ , change in y position,  $\Delta y$  and change in heading,  $\Delta\theta$ , can be calculated. When given a non-zero angular velocity, the robot will follow a circular course (see Figure 2). If this course were followed for long enough, the robot would make a complete circle. The radius of this circle is proportional to the forward velocity and inversely proportional to the angular velocity. If the radius of the circle is fairly small, we can use the starting position on this circle and the ending position on the circle to calculate change in robot position:

$$\begin{aligned} r &= V_x/\omega \\ \Delta\theta &= \omega \Delta t \\ \Delta x &= r[\sin(\theta_0 + \Delta\theta) - \sin(\theta_0)] \\ \Delta y &= r[\cos(\theta_0 + \Delta\theta) - \cos(\theta_0)]. \end{aligned} \quad (1)$$

Since commands in the command queue could have different desired velocity and angular velocity, each node in the command queue could follow a different size circle. This is acceptable because we can simply append an arc from one circle size onto the arc generated from the last command node. Using this method, we iteratively update  $\Delta\theta$ ,  $\Delta x$  and  $\Delta y$ , using the previous values for  $\theta_0$ ,  $x_0$  and  $y_0$ . Each prediction stage uses the velocity, angular velocity and the amount of time the command was active for  $V_x$ ,  $\omega$  and  $\Delta t$ . The new values of  $\Delta x$ ,  $\Delta y$  and  $\Delta\theta$  are generated from that command and this is repeated for the next command. The most recently issued command is handled in exactly the same way, except we use the amount of time the command has been active for  $\Delta t$ .

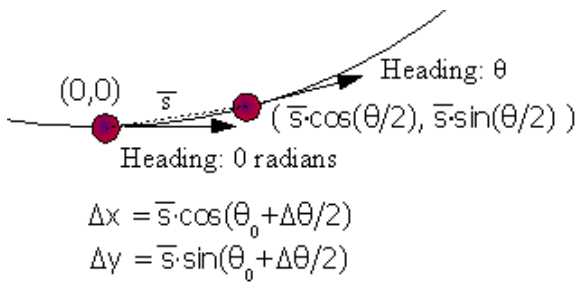


Fig. 3. Straight Line Predictions.

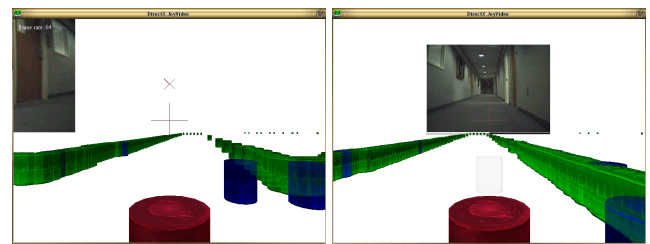
If the robot is not turning very quickly,  $\omega$  will be very small, which can lead to significant floating point error. A simpler formula for dead reckoning is thus used when  $|\omega|$  is small (less than  $\pi/300$ ); these formulas were adapted from simple straight-line formulas [9]. Using these formulas, we first calculate the displacement of the robot,  $\bar{s}$ , and the amount the robot has turned,  $\Delta\theta$  (see Figure 3).  $\Delta y$  and  $\Delta x$  are related to the sine and cosine of the amount the

robot has turned. The straight line approximations are given as follows:

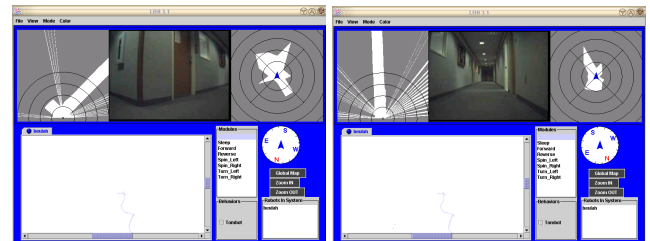
$$\begin{aligned} \bar{s} &= V_x \Delta t \\ \Delta\theta &= \omega \Delta t \\ \Delta x &= \bar{s} \cos[\theta_0 + (\Delta\theta/2)] \\ \Delta y &= \bar{s} \sin[\theta_0 + (\Delta\theta/2)]. \end{aligned} \quad (2)$$

If the robot is traveling on a circular path, the angle from the origin to the robot will be half the change in heading of the robot, as long as the change in heading is less than  $360^\circ$ . For example, if the robot has turned  $90^\circ$  it has gone  $1/4$  of the way around the circle. If the original position was at the origin, facing  $0^\circ$ , the new position would be along the ray  $45^\circ$  from the origin. So, depending on the forward velocity, the new position would be at  $(1,1)$  or  $(\pi, \pi)$ , etc. Of course, there would be significant error in the fact that the robot has taken a curved path, instead of a straight path for the distance it has traveled. This error goes to zero as the change in heading goes to zero, however, which is why we use these formulas when  $|\omega|$  is small.

This amounts to first-order prediction, which is similar to the prediction model of a simple Kalman filter. These models are used extensively in robotics [10], [11]. A more accurate model of robot movement could be obtained by taking into account acceleration and the current velocity of the robot. The subjective difference may be small, however, because there will still be errors determining how long a particular command will run on the robot and how future commands would effect the robot when using second-order prediction. Another issue is that different robots have different acceleration characteristics, so the parameters would need to be adjusted for each new robot.



Ecological Display



Standard Display

Fig. 4. Prediction Helps Robot Turn Corner.

Figure 4 shows how prediction can help users effectively turn corners. The top left picture shows a view from the

ecological display when the past commands have been telling the robot to turn right and the current command is for the robot to stop. The next picture (top right) shows what is in the display after the robot has come to a stop. The robot has turned about  $45^\circ$ , which is approximately the same amount as was predicted. This can be seen in the laser representations of the hallway. The two pictures of a standard display, taken at the same times as the pictures above them, show that a user using this display has to figure out for themselves when they have finished turning the corner and it is time to straighten out.

#### IV. HOW WE DRAW THE 3-D WORLD

The display is built around standard 3-D rendering software using DirectX on a Pentium IV computer with a Radeon 9000 video card. We are using a Pioneer 2 DXe robot which has a laser range finder, sonars and a forward-facing camera. The simulated robots simulate the same sensor suite as the Pioneer robot. The most recently received image from the camera is rendered on a rectangle some distance from the robot. The laser range finder gives us a reading for the 180 degrees in front of the robot, one reading per degree. There are 16 sonars, which give readings for the area surrounding the robot in all directions. These readings are much less accurate than the laser readings, but they are the only sensors which can detect obstacles behind the robot.

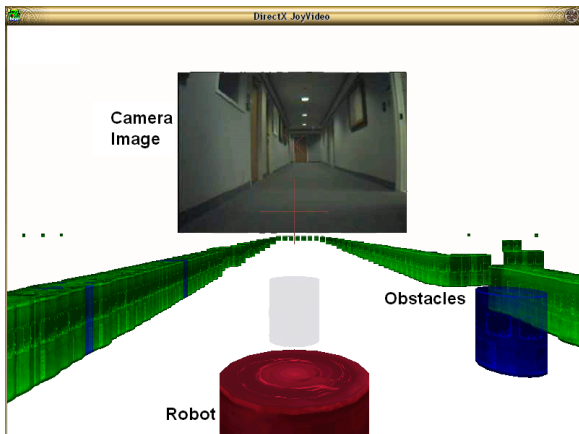


Fig. 5. Robot in Hallway.

In the display (as shown in Figure 5), the robot is rendered as a red cylinder in the bottom center of the display. A green barrel is shown in the display for each laser reading. These barrels are placed in the relative position of the reading from the predicted position of the robot. Blue barrels are placed in locations where the robot found obstacles with sonar. Figure 5 shows what a typical hallway looks like through the interface. Quickening changes the view to reflect the change in position and orientation of the robot. The robot moves with the view, so it should remain stationary at the bottom of the display.

#### V. TESTING

We validated the display with a group of 32 people with varying, but minimal, levels of robot experience.

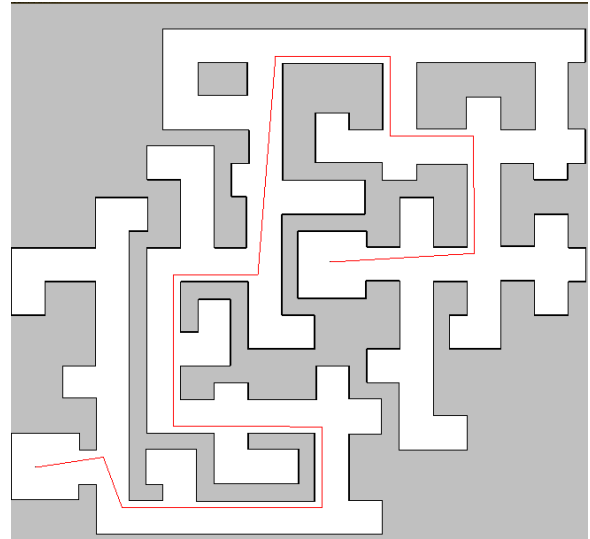


Fig. 6. Example Simulation World.

Their task was to follow simple waypoints from a starting position to an ending position in four worlds, such as the one shown in Figure 6. In all four worlds, the average performance was better with the ecological display than a standard display. Operators using the ecological display finished an average of 17% faster than when they used the standard display. Additionally, there were 5 times as many collisions using the standard interface and people preferred the ecological display 4 to 1. Standard t-tests show that completion time and number of collisions are statistically significant at  $p = 0.01$ . Combining this with the 4 to 1 preference ratio demonstrates that the ecological interface is more acceptable and easier for people to use than the conventional interface.

#### VI. ADJUSTABLE HUMAN-ROBOT INTERACTION

One of the key elements when discussing the interaction between a human and a mobile robot is the frame of reference through which the user views the happenings around the robot [3]. One of the points discussed by Wickens and Hollands is that “Different display formats may be better- or worse-suited for different tasks.” [3]. A challenge that we face in interface design is the adjustable nature of human-robot interactions. If a task is sufficiently difficult, it is feasible that there will be different interaction methods that are better or worse throughout the task. Wickens and Hollands have identified two types of tasks that are typical with human-robot systems: tasks involving navigation and tasks involving understanding [3]. According to the literature, tasks involving navigation are better supported by displays offering more egocentric information [3], [12], [13], while tasks involving understanding of the spatial structure of the environment are better supported by displays offering more exocentric information. [14]–[17]. Overall understanding of the situation the robot is in is important to the user making good decisions [18], [19].

In human-robot systems, it is often necessary to perform tasks that involve both navigation and understanding. The

most effective interaction will be based on the task at hand, the robot autonomy, the workload of the operator, and the number of users operating the system. The challenge, then, is to design an interface that facilitates adjustable levels of human-robot interaction.

## VII. AN EXTENSION TO MULTIPLE PERSPECTIVES

To overcome the challenge of adjustable interaction, we extend the previous display to a virtual 3-D representation where the viewpoint of the environment can be adjusted to fit the current task and/or the needs of the operator. Another change in the interface is that walls and paths can be shown in an *a priori* map in 3-D perspective. This allows operators to see the robot in the context of all available information. The display can further be extended to include multiple robots. Having this sort of display affords users greater situation awareness with respect to the activities and tasks of the robots. Not only can the operator gather information about where robots are in the map, they can zoom in on any of the robots to see what they are seeing and what they may encounter. Figure 7 shows some of the perspectives that are possible with this interface. The first two images show a view that may be useful for teleoperation. The next two images may be useful for a control scheme based on commands such as “take the next left” or “take the next right”. The last two images represent perspectives which could be used for waypoint control.

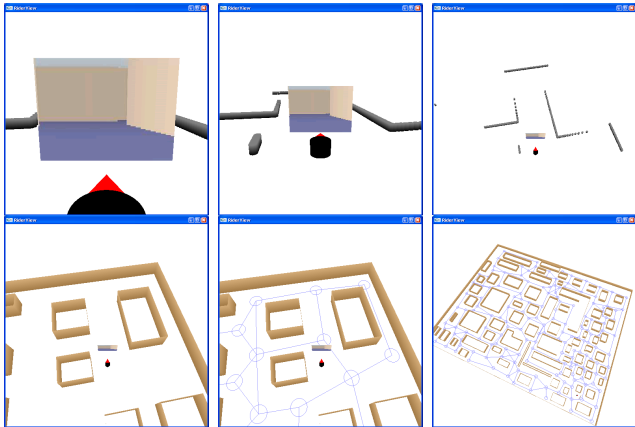


Fig. 7. Views from Different Perspectives.

In order for an interface to be useful in a variety of situations, the interface must meet certain requirements. First, the user should be able to dictate the level of interaction with the robot. Specifically, the user should be able to add and remove information in the display at any time. Second, the user should be able to view the environment and the robot from multiple angles. Third, the display should be able to support multiple users working with multiple robots. Finally, features added to the interface should enhance the user’s experience without overloading them.

As an example, we have added the ability to take snapshots of interesting places in the environment (see Figures 8 and 9). The user simply clicks a button on the

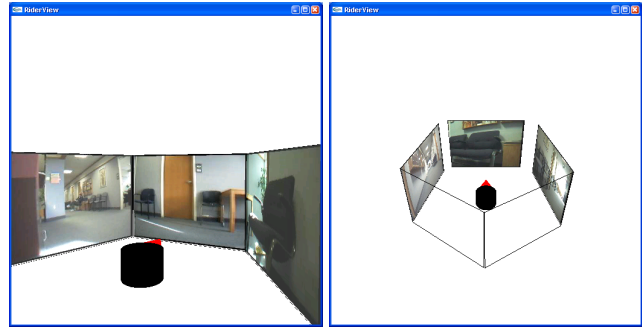


Fig. 8. Panoramic Snapshots

joystick and a snapshot is placed at the robot’s location in the virtual environment. This snapshot ability is useful when doing a task such as identifying objects or places of interest. By allowing the user full control over the viewpoint of the virtual environment, the user can tell the robot to autonomously perform a simple task and then visit snapshots that have been taken by robots in various places throughout the environment. As task complexity increases and more robots and operators are utilized, it will be important to have an interface that supports multiple-robot, multiple-user interactions. As an example, imagine a search and rescue mission over a large area where many vehicles are required for effective searching. The interface we present will allow multiple users to view the environment thereby enabling the integration of modular information from robots and operators into a single useful display that an organizing committee can use to make global decisions.

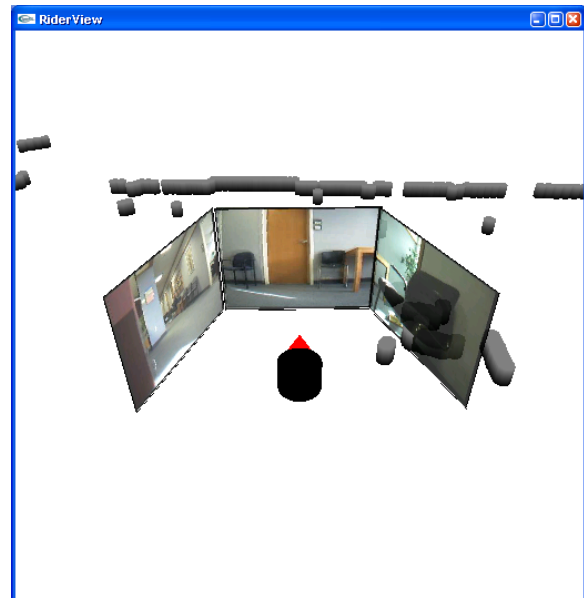


Fig. 9. Snapshots with Depth Information

The interface allows users to do anything from planning courses of action for entire human-robot teams to guiding an individual robot through a cluttered part of an environment. With the addition of snapshots, users can identify

places of interest and share their findings with other users. Not only can the interface display where on the map a feature is located, it also facilitates path planning to the place of interest. Users could traverse such a path in the virtual world to visualize what they might see if they traveled the path in the real world.

### VIII. FUTURE WORK

The prediction algorithm we use is effective, but could be improved by integrating acceleration, velocity and additional timestamp information. Integrating state-of-the-art occupancy grid mapping and localization [20], [21] would also help human-robot teams perform in unfamiliar environments. Topological maps that facilitate path-planning and path-changing algorithms are currently in the works. Along with this, we will implement algorithms for displaying the intentions of a robot so that a user can quickly comprehend what the robot is about to do.

To further validate our research, we will continue user studies in an effort to identify what principles of interface design apply to the field of adjustable interaction. As these principles are identified, we will integrate them into our system. As part of the user studies, we will have users working together with a large team of robots to test the effectiveness of the multiple-user, multiple-robot interactions.

### IX. CONCLUSION

In human-robot systems, it is important to present the information to an operator in a usable manner. In this paper we have presented an interface that supports teleoperation by placing the viewpoint behind the robot such that the robot appears in the user's view of the environment. We have presented results indicating a 17% increase in performance using the new interface using a simulated robot. Operators preferred the ecological display 4 to 1 over a standard interface. Preliminary results with a robot in the real world have yielded similar results. In addition, we have extended the original interface to allow users to view the environment from a continuous range of positions. This enables the user to find a perspective that supports the current task and to switch to other perspectives when their needs change.

### ACKNOWLEDGMENT

This work was partially funded by DARPA under contract #NBCH1020013.

### REFERENCES

- [1] M. R. Endsley, "Automation and situation awareness," in *Automation and human performance: Theory and applications*, R. Parasuraman and M. Mouloua, Eds. Mahwah, NJ: Lawrence Erlbaum, 1996, pp. 163–181.
- [2] M. Mulder, *Cybernetics of Tunnel-In-The-Sky Displays*. Delft University Press, 1999.
- [3] C. D. Wickens and J. G. Hollands, *Engineering Psychology and Human Performance*, 3rd ed. Prentice-Hall, New Jersey, 2000.
- [4] T. B. Sheridan, *Telrobotics, Automation, and Human Supervisory Control*. Massachusetts Institute of Technology, 1992.
- [5] E. Krotkov, R. Simmons, F. Cozman, and S. Koenig, "Safeguarded teleoperation for lunar rovers: From human factors to field trials," in *Proceedings of the IEEE Workshop on Planetary Rover Technology and Systems*, April 1996.
- [6] T. Fong, C. Thorpe, and C. Baur, "A safeguarded teleoperation controller," in *IEEE International Conference on Advanced Robotics 2001*, August 2001.
- [7] G. Goodrich, D. R. O. Jr., J. W. Crandall, and T. Palmer, "Experiments in adjustable autonomy," in *Proceedings of the IJCAI01 Workshop on Autonomy, Delegation, and Control: Interacting with Autonomous Agents*, 2001.
- [8] R. Thomas, W. D. Robinson, and S. Dow, "Improving the visual experience for mobile robotics," in *Seventh Annual Iowa Space Grant Proceedings*. Drake University, November 1997, pp. 10–20.
- [9] G. W. Lucas, "A tutorial and elementary trajectory model for the differential steering system of robot wheel actuators," 2001. [Online]. Available: <http://rossum.sourceforge.net/papers/DiffSteer/DiffSteer.html>
- [10] R. W. Beard, T. W. McLain, M. A. Goodrich, and E. P. Anderson, "Coordinated target assignment and intercept for unmanned air vehicles," in *IEEE Transactions on Robotics and Automation*, vol. 18, no. 6, Dec 2002, pp. 911–922.
- [11] M. M. Veloso, P. Stone, and K. Han, "CMUNITED-97: Robocup-97 small-robot world champion team," in *AI Magazine*, vol. 19, no. 3, 1998, pp. 61–69.
- [12] O. Olmos, C. Wickens, and A. Chundy, "Tactical displays for combat awareness," *International Journal of Aviation Psychology*, 1999.
- [13] C. Wickens and T. Preveet, "Exploring the dimensions of egocentricity in aircraft navigation displays: Influences on local guidance and global situation awareness," *Journal of Experimental Psychology: Applied*, vol. 1, pp. 110–135, 1995.
- [14] A. Aretz, "The design of electronic map displays," *Human Factors*, vol. 33, pp. 85–101, 1991.
- [15] W. Barfield and C. Rosenberg, "Judgments of azimuth and elevation as a function of monoscopic and binocular depth cues using a perspective display," *Human Factors*, vol. 37, pp. 173–181, 1995.
- [16] C. Wickens, C.C.Liang, T. Preveet, and O. Olmos, "Egocentric and exocentric displays for terminal area navigation," *International journal of aviation psychology*, vol. 6, pp. 241–271, 1996.
- [17] C. Wickens, "Frames of reference for navigation," in *Attention and performance (Vol 16)*, D. Gopher and A. Koriat, Eds. Orlando, FL: Academic Press, 1999, pp. 113–144.
- [18] M. R. Endsley, "Designing for situation awareness in complex systems," in *Second International workshop on symbiosis of humans, artifacts, and environment*, Kyoto, Japan, 2001.
- [19] A. A. Nofi, "Defining and measuring shared situation awareness," Center for Naval Analyses, November 2000.
- [20] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millenium*, G. Lakemeyer and B. Nebel, Eds. Morgan Kaufmann, 2002.
- [21] J. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," in *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, 2000.