Interface Evaluation for Mobile Robot Teleoperation

Roberto Olivares
Vanderbilt University
Nashville, TN 37235
roberto.olivares@vanderbilt.edu

Chen Zhou
Vanderbilt University
Nashville, TN 37235
chen.zhou@vanderbilt.edu

Bobby Bodenheimer
Vanderbilt University
Nashville, TN 37235
bobby.bodenheimer@vanderbilt.edu

Julie A. Adams
Vanderbilt University
Nashville, TN 37235
julie.a.adams@vanderbilt.edu

ABSTRACT

Controlling mobile robots through teleoperation is a challenging task that demands a flexible and efficient user interface. Mobile robots are often equipped with numerous sensors (proximity sensors, system status sensors, positioning and heading devices, multiple cameras, etc.) that provide a high volume of data to the user. Because the amount of data is vastly larger than what can fit on the screen, and because the needed subset of data can change rapidly and unpredictably depending on events in the robot's environment, modern teleoperation interfaces often display user-selected data with a windowing paradigm that facilitates quick display modification. In this paper, we examine the possibility that too much fine-grained control over window positioning and sizing could hamper user performance by interfering with display modification. To test our hypothesis, three human-computer interfaces for a mobile robot were designed and then evaluated through performance studies consisting of 12 expert and 24 novice participants. The first two interface designs followed the standard Microsoft Windows graphical user interface design paradigm and provided participants with fine-grained control over the position and sizing of sensor displays. The third interface was designed using principles from cockpit and human-factors research, and provided participants with limited control over display position and sizing. User performance for each interface was assessed on a display reconfiguration task consisting of adding, removing, positioning, and resizing sensor data windows, a common task in robot navigation and situational assessment. In general, expert participants preferred the interface with limited control over position and sizing, while novice participants preferred the more traditional windows, icons, menus, pointing device interfaces. No single interface accurately captured all user preferences, but the feature-specific results found provide direction for future designs.

Keywords

Human-Robot Interaction, Human-Computer Interaction, user-interface design, human factors, teleoperation, mobile robotics.

1. INTRODUCTION

Robots are versatile tools useful in a wide variety of situations. When humans and robots interact, the interaction may be proximate and direct, e.g., the physical contact of a haptic interface, or it may take place over time and distance. The latter case is called robotic teleoperation, and requires a user interface to translate operator commands to the robot and provide feedback from the robot to the operator. Teleoperated mobile robots are increasingly used in situations where it is impossible or very difficult for humans to be present, e.g., in underwater exploration or in earthquake-damaged buildings.

Successful operation of the robot, however, depends critically on how well information from the robot is presented to the user. Mobile robots typically contain a variety of sensor modalities, including video, sonar, lidar (light detection and ranging), infrared, and GPS positioning. The presentation of such an abundance of sensory information coupled with the difficulty in interpreting such data often leads to a high cognitive load on the operator. This load can often overwhelm the operator and result in suboptimal performance due to a failure to understand the current state of the robot. Work has been done attempting to filter [3] and partially automate [11, 12] the display of this information, but this automation approach has not been adopted extensively due to the unpredictable demands of real-world environments. Because this automation approach is still unreliable, the task of robotic display management still falls into the hands of the human operator. In this situation it becomes critical that the interface between human and robot be as simple and easy to use as possible. Poorly designed interfaces, on the other hand, can result in spatial disorientation, attentional bottlenecks, lack of situational awareness, confusion, and frustration. This paper describes a set of experiments that evaluate what types of window displays and controls are most helpful in a teleoperation environment, and provides guidelines for the construction of human-robot teleoperation interfaces.

2. DESCRIPTION OF INTERFACES

The baseline user interface for the mobile robotics laboratory at Vanderbilt University was developed by Nilas and is described in [3, 4]. This interface and control system has been used to control several mobile robots, including an ATRV-JR [2] and a Pioneer2-AT [1]. The ATRV-JR is equipped with a pan-tilt vision system, sonar, lidar, and digital GPS. The Pioneer2-AT is equipped with sonar and a 360-degree camera array. Each sensor on the mobile robot collects data in real time from the environment; the data is then transmitted back to the operator and displayed. Operator commands from the interface are likewise transmitted to the robot. In this study, three interfaces were examined.

ACMSE '03, March 7-8, 2003, Savannah, GA.

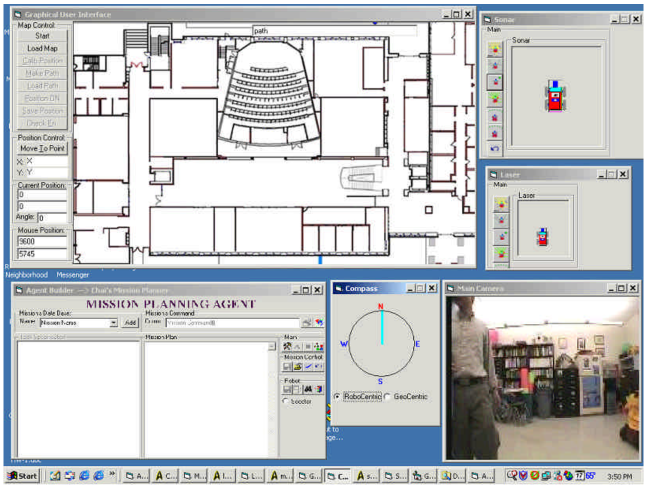


Figure 1. The Nilas interface.

The Nilas interface [3, 4] is composed of four sensory data windows and two command windows. Each window is opened through a separate server program which the user is responsible for manually starting and configuring. After opening the windows required for the task, the user positions them at the proper locations on the screen so that they do not obstruct one another. The user resizes opened windows depending on task requirements, e.g., the user may need a larger map window to navigate the robot precisely. Thus the map window usually is larger than the other windows. Figure 1 provides a snapshot of the Nilas interface.

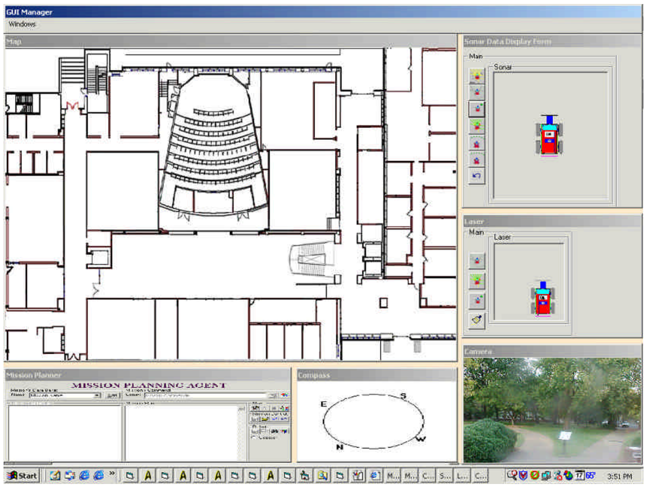
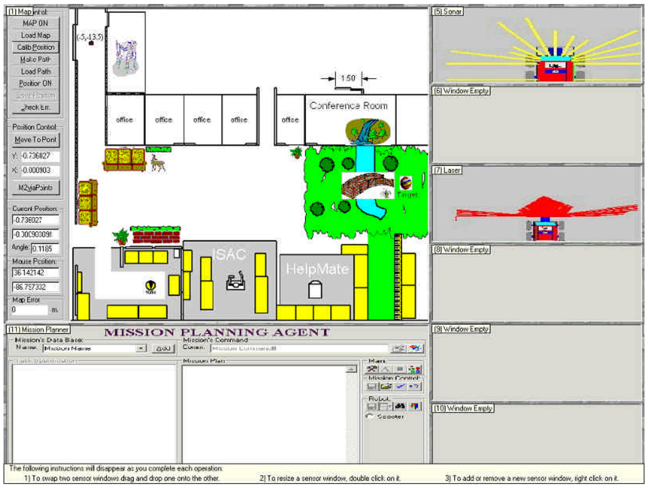


Figure 2. The Express interface.

The second interface, named Express, was designed and developed based on Nilas' interface. Express combines all windows into one and hides the operation of opening server windows for each data and command window. The idea behind the Express interface is to maximize screen real estate with no overlapping windows. By selecting menu choices, the user selects the windows to be displayed. Windows are aligned with respect to one another on the screen and share a common boundary between neighboring windows. The windows are dependent on one another, and can be finely resized. Express organizes the windows in a way that aligns three windows with the same width to the right of the screen and places one large window above two smaller windows on the left of the screen. Figure 2 provides an example configuration of Express windows.

The windows to be displayed are selected by the user via a main menu. If a window is closed, its neighboring windows resize to absorb the free space. When new windows are added, the neighboring windows automatically shrink to provide space for the new windows. Each window has an assigned location and remains in that location throughout the system use. Participants cannot change the window positions, and therefore can always expect to find windows at the predefined locations. When the system starts up, all the windows open at their designated locations with a default size. The user can immediately begin operating the robot with the default settings or change the configurations to match their preferences.



The following instructions will disappear as you complete each operation:
1) To swap two sensor windows drag and drop one onto the other.
2) To resize a sensor window, double click on it.
3) To add or remove a new sensor window, right click on it.

Figure 3. The Cockpit interface.

The final interface is called the Cockpit interface. The Cockpit interface was designed to minimize the significant amount of time participants spend resizing and repositioning windows. To accomplish this task, the interface follows airplane [9] and helicopter [10] cockpit designs and removes fine-grained control over sensor display position and sizing. Instead, sizing is accomplished by either swapping displays into larger pre-assigned windows (drag and drop), or through toggling the window size with a zoom operation (double click). Both of these operations are accomplished rapidly through the use of gross motor movements and single mouse clicks, as opposed to fine-grained mouse positioning and menu navigation. Many of the Windows interface design guidelines are expressly ignored for the Cockpit interface, as they adhere to a standard that does not accurately reflect teleoperation tasks or operators. Figure 3 provides an example sensor display configuration in the Cockpit Interface[1].

---

[1] Sensor windows have not been zoomed to allow readers to see the empty panels used to swap display size and position. Normally, more of the screen space would be occupied.

3. EXPERIMENTAL DESIGN

3.1 Participants

Two groups of participants, novices and experts, were employed to evaluate the three user interfaces. The novice participant group consisted of 24 people with no prior experience working with mobile robots, although they were familiar with the operation of a computer. The expert participant group consisted of 12 people who had considerable experience working with mobile robots. Some members of the expert group had prior experience with the Nilas interface. All participants had normal color vision. All participants were right-handed and performed the tasks with their preferred hands. All but five participants had post high school education while five had high school education. All participants completed a background questionnaire regarding their computer usage and mobile robot teleoperation experience. Most participants had experience using computers, but only expert participants had experience using mobile robots.

3.2 Tasks

A set of tasks was designed for the experiment that participants completed on all three interfaces. Help text was provided in the Express interface and the Cockpit interface for reference by participants. The tasks focused on testing participant's preferences of fine-controlled mouse movement vs. coarse-controlled mouse movement. The task list included adding a new window to the screen, rearranging windows so they were all completely visible, resizing a window to a preferred size, swapping the positions of two windows, removing a window from the screen, and restoring a window to its previous position.

3.3 Procedures

The experiments assumed the focal hypothesis that the Express and Cockpit interfaces were superior to the Nilas interface. Participants were shown how to perform the first three tasks using the Nilas interface and then asked to complete the rest of the tasks on the list. In the case of Express and Cockpit, participants were asked to complete the task list without receiving prior training. One-half of the participants in each group completed their task set with the Express interface first, and the other half were given the Cockpit interface to use first. After completing each task, participants completed a post-task questionnaire consisting of eighteen five-point Likert scale questions. The average task completion time was approximately 15 minutes. The tasks participants were asked to perform are commonly found in teleoperation. These tasks include adding new sensor windows, rearranging sensor windows as new data comes available, and removing sensor windows that are inactive.

After performing the task on all three interfaces, participants completed a comprehensive post-examination questionnaire. These post-examination questions consisted of seven general feature questions (five-point Likert scale), six interface preference questions (multiple choice), five short-answer questions, and one overall interface ranking question. To aid participants in distinguishing the three interfaces for this questionnaire, they were provided with a grayscale screenshot of each interface for reference. The average completion time for the entire experiment was 52 minutes.

4. RESULTS

4.1 Analysis

A series of one-way ANOVAs were performed on post-task Likert scale questions. Individual t-tests were evaluated to differentiate each of the three interfaces for all significant one-way ANOVA results. Results are considered significant if $p < 0.05$. For post-examination Likert scale questions, response means and standard deviations were calculated instead of ANOVA comparisons[2]. An additional one-way ANOVA was performed on these questions to compare expert and novice data sets. Post-examination feature preferences were binned by interface, then converted to percentages. Post-examination interface rankings were binned by rank: 1st choice (most preferred), 2nd choice, 3rd choice (least preferred) then converted to percentages.

4.2 Post-Task Questions

Of the eighteen post-task questions administered, twelve yielded significant differences across at least two of the interfaces. Four of these twelve questions yielded significant results in both the novice and expert data sets simultaneously and are described first.

Both participant groups found significant differences in the appearance of the user interface, finding that both Express and Cockpit were more organized than the Nilas interface ($F(2, 69) = 4.313$, $p = 0.017$ for novices and $F(2, 33) = 14.854$, $p < 10^{-4}$ for experts). Both groups found that adding sensor displays was both faster and easier in the Express and Cockpit interfaces than in the Nilas interface ($F(2, 69) = 3.467$, $p = 0.037$ for novices and $F(2, 33) = 7.867$, $p = 0.002$ for experts). Novice participants also found that that the easiest interaction with the mouse occurred in the Nilas interface ($F(2, 69) = 3.378$, $p = 0.039$) but found the most pleasant mouse interaction occurred with the Cockpit interface ($F(2, 69) = 3.698$, $p = 0.030$). Experts differed on this subject, finding that both the easiest and most pleasant mouse interactions occurred with the Cockpit interface ($F(2, 33) = 20.044$, $p < 10^{-5}$, and $F(2, 33) = 18.348$, $p < 10^{-5}$, respectively). The final significant result for the novice group was their preference for the Express method of adding sensor displays over that of the Cockpit and Nilas interfaces ($F(2, 69) = 5.406$, $p = 0.007$). Experts found no significant differences across interfaces for adding sensor displays.

The expert group contained eight additional significant differences across interfaces that were not found in the novice data set. The first three of these expert differences were only significant between the Cockpit and Express interfaces. These included that Cockpit was considered significantly more pleasant ($F(2,33) = 5.802$, $p = 0.007$) and significantly more efficient ($F(2,33) = 3.371$, $p = 0.047$) to use than Express. Experts also found that resizing sensor displays in Cockpit was significantly faster than in Express ($F(2, 33) = 6.304$, $p = 0.005$). Additionally, the experts felt that Express was significantly more irritating ($F(2,33) = 5.489$, $p = 0.009$) and made resizing displays significantly more difficult ($F(2,33) = 6.683$, $p = 0.004$) than both the Cockpit and Nilas interfaces. These results, while somewhat counterintui-

---

[2] Comparisons across interfaces were not possible for post-examination questions since they were administered after all three interfaces had been used.

tive, were anticipated and are discussed in Section 5. In contrast, experts rated Cockpit significantly more pleasant than the Nilas interface for sensor display position and size ($F(2,33) = 3.518$, $p = 0.041$), as well as significantly better in a rush than both other interfaces ($F(2,33) = 6.507$, $p = 0.004$). The most statistically significant post-task results, however, were those in the expert group concerning sensory display resizing speed, pleasantness of mouse interaction, and difficulty of mouse interaction (see above paragraphs). In each of these cases, p was less than 2.9×10^{-5} and provided the most valuable information for future robot interface designs.

In addition to the significant results found, there was also evidence suggesting that novice participants considered the Cockpit interface the most pleasant to use ($F(2, 69) = 2.98$, $p = 0.058$) and that novice participants found the number of buttons and menus on the Express interface more useful than the number found on the Nilas and Cockpit interfaces ($F(2,69) = 2.95$, $P = 0.059$). The last two findings are only marginally significant ($0.15 > p > 0.05$).

While our post-task questionnaires were designed to extract information about features (i.e. mouse interaction, display addition, display resizing) as they were implemented in each interface, our post-examination questionnaire was intended to identify how participants felt about each feature in general. The next set of results were obtained from the general feature question section of this post-examination questionnaire. While the findings from these questions did not differ significantly across expert and novice groups, certain trends were evident. In general, participants found that the three interfaces were substantively different from one another. Novices found that automatic resizing of windows was somewhat useful but experts felt that this feature was not very useful, and both groups found it disorienting. Both groups found that controlling the sensor display size was useful, and preferred fine-grained control over coarser control. Finally, both groups found sensor display titles and the ability to rearrange the sensor displays useful.

The next section of the post-examination questionnaire focused on which interface participants preferred for each feature. When the interface votes were binned and ranked by feature, 75% of novices and 75% of experts preferred the mouse interaction method of either the Express or Cockpit interface. Experts showed a slight preference for the Cockpit interface (42%) over the Express interface (33%), while novices showed a strong preference for Cockpit (67%). However, when participants ranked individual aspects of mouse interaction, these preferences diverged. For sensor display selection and positioning, both novices and experts preferred the Express interface. Experts and novices also showed a strong preference for the Nilas and Express interfaces (92%) over the Cockpit interface (18%) for sensor display resizing. In the case of sensor display addition, however, this interface preference was reversed, with novices (92%) and experts (100%) preferring the sensor display addition method of the Express and Cockpit over the Nilas interface. Both experts (84%) and novices (88%) preferred the sensor display removal method of the Express and Cockpit interfaces.

The last portion of the post-examination questionnaire asked participants to rank the three interfaces in order from their favorite to least favorite. The votes for each interface were then binned by rank, yielding the percentage of participants voting each interface at a particular rank. These results are shown in Tables 1 and 2. Also included is a column containing the sum of the 1st choice and 2nd choice percentages, which provides a better understanding of how the interfaces compare to each other.

Overall, these results strongly indicate that both experts and novices preferred the Cockpit interface, as 100% of them chose Cockpit as their first or second choice. Put another way, not a single expert or novice participant ranked the Cockpit interface as their least favorite choice. Additionally, experts preferred the Nilas interface slightly more (by a 16% margin) than the Express interface. Curiously, in the novice group this preference was reversed, showing an 18% margin in favor of the Express interface compared to the Nilas interface.

Table 1. Expert Interface Rankings.

	1st or 2nd	1st Choice	2nd Choice	3rd Choice
Nilas	58%	33%	25%	42%
Express	42%	17%	25%	58%
Cockpit	100%	50%	50%	0%

Table 2. Novice Interface Rankings.[3]

	1st or 2nd	1st Choice	2nd Choice	3rd Choice
Nilas	41%	24%	17%	58%
Express	59%	38%	21%	42%
Cockpit	100%	38%	63%	0%

As a group, the experts provided more polarized and opinionated responses than the novices. Experts also tended to arrive at a consensus answer more frequently than novices. Of the 18 post-task questions, 61% of the questions provided significant results in the expert group, whereas only 28% provided significant results in the novice group. Even among non-significant questions, experts tended to arrive closer to consensus with their answers than novices. 86% of non-significant expert answers were considered near-significant, while only 15% of non-significant novice answers were near-significant. Among the expert group, the only question with a strongly insignificant result ($p > 0.15$) was whether size and positioning of displays on each interface was considered useful (means = {1.50, 1.92, 2.17}, standard deviations = {1.18, 1.36, 0.88}).

5. DISCUSSION AND FUTURE WORK

A common problem in human-robot interaction is that there exists a plethora of sensor information that must be integrated into a coherent command and sensor display. The presentation of such a large amount of sensor information has distracted designers from employing human factors considerations into the interface design. This paper investigated whether human factors work in cockpit design was preferable in the real-time task of controlling a mobile robot. Perhaps unsurprisingly, expert participants showed a strong preference for such designs, and novice partici-

---

[3] Column captions indicate rank for a given interface: 1st choice = favorite interface, 3rd choice = least favorite interface. Due to row and column-wise rounding constraints, these values may not sum to exactly 100%.

pants showed a slighter, but still significant preference for them as well.

The core design philosophy for the Express and Cockpit interfaces was speed of task completion, where the tasks were sensor display addition, removal, positioning, and resizing. To optimize this interface aspect, the Express interface applied HCI techniques to the existing Nilas interface, and the Cockpit interface was built from scratch and based on work in human factors and HCI. In both cases, the number of mouse clicks, time spent finely moving the mouse, and time spent having to adjust multiple windows was reduced with the new designs. However, our preference analysis indicated that participants still preferred the traditional Windows interface for positioning and resizing, while simultaneously acknowledging that it was detrimental under time constraints. Because time-critical robot tasks such as navigation and fault diagnosis involve rapidly searching through sensor information, the conflict between aesthetics and efficiency is an important issue for robotic interface design.

We are unsure if this conflict is due simply to participants having habituated to the Windows interface style, or if it is due to an underlying user need to have sensor information presented in a specific size. Our data does not seem to support a user need for fine sizing, as both novice and expert participants ranked the ability to finely resize sensor displays as only “somewhat useful” to “useful.” Additionally, assessment of common expert and novice interface configurations shows that participants tend to size windows in a few common sizes, rather than in a wide variety of finely-controlled sizes. More importantly, both expert and novice participants gave the Cockpit interface the highest overall rankings, despite the lack of a fine sizing feature. Taken together, these findings indicate that control over fine position and sizing is not usually exploited, and therefore not crucial under this scenario. Another dimension to the debate over fine information sizing and positioning is that the required size of any individual sensor window is also task dependent. When asked to rate the usefulness of size and positioning of sensor displays on each interface, no significant result was found in either the novice or expert population. We believe that for novices this finding is due to a misunderstanding of the questions, while for experts it indicates an awareness that making an accurate assessment of usefulness requires a task context. Because of factors such as these, more research is needed to firmly conclude whether fine size variations in sensor displays affect the task performance or cognitive load of the user, and whether they justify including fine sizing control over information displays.

To address the aesthetics issue we need to find common ground that will allow us to provide an interface that enables the efficient repositioning and resizing of sensor displays but that participants also find pleasant to use. Alternative methods, such as employing a weaning approach from fine motor methods to gross motor methods, could also be used. While we feel more investigation into the nature of these preferences is needed, we do not necessarily need to discount the Cockpit and Express interfaces. First and foremost, the cockpit-style interface was directly aimed at experts, and its design assumed familiarity with the interface. Since participants were not trained on it for prolonged periods of time, some acclimatization to the interface was to be expected—especially given its radical departure from what participants were accustomed to. We hypothesize that soldiers or professional participants would gradually come to find the cockpit interface pleasant to use once they more accurately see the performance gains in more complicated tasks. Naturally, more detailed expert user studies will be necessary to corroborate this hypothesis, and the same can be said for certain features of the Express interface.

The data and feedback acquired from participants has, however, given us immediate and significant direction in interface design. By having identified the features of each interface that participants preferred, we can attempt the design of a hybrid interface following the basic Cockpit style, but with additional Express/Nilas interface features. When considering new design approaches, we found it important to note that no individual interface was ranked the best in every category, and that even the Nilas interface was found to excel in a few categories over the newer interfaces. Thus, we will attempt to integrate participant feedback and provide resizing/repositioning schemes that allow participants to take advantage of both gross motor movements and fine motor movements. Other features, such as the single-mouse click limit for core graphical user interface (GUI) tasks will be kept in place. We also feel that a naive combination of the strongest features of the Nilas, Cockpit, and Express interfaces would not yield a superior interface. We have identified certain fundamental Windows GUI principles that do not successfully translate under human-factors and task-specific considerations. These include drop-down menus, window border sizing, window positioning, multiple dialogs, and the application title bar. These features tend to be at odds with the cockpit design, and thereby oppose a simple linear feature combination.

In addition to the statistical data, a common participant request was for better online help for each of the interfaces. This feature was originally left out to avoid biasing the experiment with varying levels of documentation. Another important conclusion drawn from participant comments was that the Express interface contained a feature that many participants did not appreciate: click-less resizing of windows. Participants found this feature unpredictable and difficult to control, therefore we are planning a revision that would maintain the general panel-like (affecting multiple sensor display) sizing approach but remove the click-less aspects. Given user rankings of the three interfaces, we feel that looking to human factors optimization for existing robot interfaces is the correct avenue to pursue. This is especially evident when taking into consideration the expert consensus that conventional window sizing and positioning schemes, while pleasant to use, are insufficient.

The task that novice and expert participants were asked to perform consisted of relatively short, general interface actions for display management. The participants were not interacting with active robots or receiving live sensory feedback that would naturally constrain user behavior. Given this stated limitation, we are unable to generalize temporal or cognitive load results to these situations. Future studies under specific teleoperation scenarios would be required to gather such results.

Our results, however, are important for the burgeoning field of human-robot interaction, because they imply that considerable knowledge from HCI can be leveraged in designing better interfaces for human-robot interaction. We are currently investigating integrating more powerful features into our human-robotic interface design. However, critical to this integration is the realization that the robots themselves possess cognitive abilities. As robots become more cognitively powerful, the interface should provide more information about the robot's assessment of its state. This type of interface will eventually lead to true human-robot collaboration. The interface must take into account that the end system is a complex and dynamic system with cognitive and autonomous behaviors.

6. ACKNOWLEDGMENTS

All authors would like to thank the novice participants and researchers at Vanderbilt University that volunteered their time for this user study. The second author gratefully acknowledges support from DARPA (DASG60-01-1-0001-SMPC). Additionally, the authors would like to thank the anonymous reviewers for their insightful comments and suggestions.

7. REFERENCES

[1] ActivMedia Robotics. http://www.activrobots.com

[2] IRobot Corporation. http://www.irobot.com

[3] Kawamura, K., Nilas, P., Muguruma, K., Adams, J. A., and Zhou, C. An agent-based architecture for an adaptive human-robot interface. In *36th Hawaii International Conference on System Sciences*, January 2003.

[4] Nilas, P. A Multi-agent-based architecture for an adaptive human-robot interface. PhD thesis, Vanderbilt University, expected May 2003.

[5] Smith, M.W., Sharit, J., and Czaja, S.J. Aging, Motor Control, and the Performance of Computer Mouse Tasks. *Human Factors Journal*, 41 (1999), 389-396.

[6] Miller, C.A., Bridging the Information Transfer Gap: Measuring Goodness of Information Fit. *Journal of Visual Languages and Computing*, 10 (1999), 523-558.

[7] Rosson, M.B., and Carrol, J.M., *Usability Engineering: Scenario-based Development of Human-Computer Interaction.* Morgan Kaufman, 2002.

[8] Rubin, J. *Handbook of Usability Testing.* Wiley, 1994.

[9] Honeywell Integrated Avionics Systems. http://www.myflite.com/myflite/products/ias/epic/11Introduction.jsp.

[10] Richards, R.A. Principle Hierarchy Based Intelligent Tutoring System for Common Cockpit Helicopter Training. *Intelligent Tutoring Systems: Proceedings of ITS 2002*, (June 2002), 473-483.

[11] Mitchell, C., and Saisi, D. Use of model-based qualitative icons and adaptive windows in workstations for supervisory control systems. *IEEE Transactions on Systems, Man, and Cybernetics.* 17 (1987), 594-607.

[12] Miller, C., Shalin, V., Geddes, N., and Hoshstrasser, B. Plan-based information requirements: automated knowledge acquisition to support information management in an intelligent pilot-vehicle interface. *Proceedings of the Digital Avionics Systems Conference,* (1992), 428-433.

[13] Microsoft Corporation. *Microsoft Windows Interface Design Guide.* Microsoft Press, 1996.