



## Faculty Publications

---

2003-10-01

# Seven principles of efficient human robot interaction

Michael A. Goodrich  
mike@cs.byu.edu

Dan R. Olsen Jr.  
dan\_olsen@byu.edu

Follow this and additional works at: <https://scholarsarchive.byu.edu/facpub>



Part of the [Electrical and Computer Engineering Commons](#)

## Original Publication Citation

Goodrich, M. A., and D. R. Olsen Jr. Seven Principles of Efficient Human Robot Interaction. Vol. 4., 23

---

## BYU ScholarsArchive Citation

Goodrich, Michael A. and Olsen, Dan R. Jr., "Seven principles of efficient human robot interaction" (2003). *Faculty Publications*. 1287.  
<https://scholarsarchive.byu.edu/facpub/1287>

This Presentation is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Faculty Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact [ellen\\_amatangelo@byu.edu](mailto:ellen_amatangelo@byu.edu).

# Seven Principles of Efficient Human Robot Interaction\*

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

Dan R. Olsen, Jr.  
Computer Science Department  
Brigham Young University  
Provo, UT, USA  
olsen@cs.byu.edu

**Abstract** – *Advances in robot technology and artificial intelligence have increased the range of robot applications as well as the importance of supporting human interaction with robots and robot teams. Previous work by the authors has highlighted the importance of creating neglect tolerant autonomy and efficient interfaces. In this paper, lessons learned from evaluating neglect tolerance and interface efficiency are compiled into a set of principles for efficient interaction. Emphasis is placed on designing efficient interfaces, but many of the principles require autonomy levels that support the principles. Each principle is illustrated by an example and motivated by citing relevant factors from cognitive information processing.*

## 1 Introduction

Improving robot technology has increased the types of domains where robots can be useful to humans. Many of these domains require that humans interact with the robots to guide the mission, interpret sensor data, or coordinate multiple tasks. In this paper, we restrict attention to circumstances where the robot is remote from the human.

Ideally, the presence of the robot is transparent to the human; the human interacts directly with the world. The human probes the world through some control element and gathers information about the world through some information presentation element.

Unfortunately, a remote human cannot directly interact with the world but must, instead, interact with the world via a robot intermediary. In such a situation, control commands are passed from human to robot, modulated by robot autonomy, and then implemented in the world. On the return path, information is not passed directly to the human, but is instead obtained via a limited set of robot sensors where they are interpreted and, perhaps, used to modulate robot behavior.

In this context, there are two interaction loops: the human interacts with the robot via an interface, and the robot interacts with the world via an autonomy mode. The robot has limitations to what it can do autonomously, how it can act when autonomous, and where it can operate autonomously. The interface likewise has limitations.

Such limitations mean that human intent is transformed and restricted by the available technology. Every such transformation introduces a potential workload bottleneck or potential error condition. The principles that we develop are designed to counteract the effects of these bottlenecks.

## 2 Basis for Principles

### 2.1 Neglect Time

A very important metric in measuring the autonomy of a robot with respect to some task (and corresponding task effectiveness metric) is the robot's *neglect tolerance*. Neglect tolerance is a measure of how the robot's current task effectiveness declines over time when the robot is neglected by the user.

We have conducted a series of experiments that suggest that for a given robot, interface, and problem space there is a characteristic neglect curve such as that shown in Figure 1. These experiments required human subjects to perform secondary tasks that forced them to attend to something other than robot control for selected periods of time. Indeed, as attention is turned elsewhere, average robot performance declines.

These experiments allow us to evaluate the amount of time that a robot can be neglected before average performance drops below a given level of performance. This time is referred to as *neglect time* and is illustrated in Figure 1. Each autonomy mode has a characteristic neglect time which

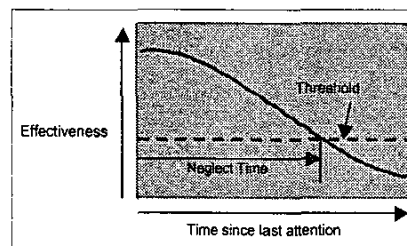


Figure 1: Neglect time is the amount of time before robot performance drops below a threshold.

varies as a function of world complexity.

\*0-7803-7952-7/03/\$17.00 © 2003 IEEE

## 2.2 Interaction Time

In addition to neglect time, the interface between human and robot also constrains performance. Interfaces that efficiently allow a human to control the robot and manipulate the world are desirable.

Experiments were conducted where subjects were required to perform secondary tasks. When attention is turned from the secondary task back to the robot task after a selected period of time, performance begins to increase. The amount of time required before performance rises from threshold to peak performance is referred to as *interaction time*. This concept is illustrated in Figure 2 when performance starts at a minimum.

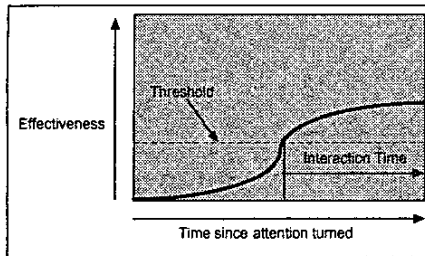


Figure 2: Interaction time is the amount of time before robot performance rises from threshold to peak performance.

Interaction time includes the time required to switch attention from a secondary task to the primary task, time required to establish context, time to plan, and time to communicate the plan to the robot. Interfaces that produce small interaction times are preferred to interfaces that produce large interaction times.

## 2.3 Robot Attention Demand

A key metric of system performance pivots on the question, "How much time does a robot demand?" As shown in Figure 3, interaction time (IT) and neglect time (NT) determine the answer to this question. In a task with consis-

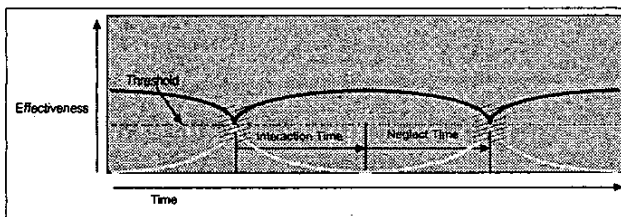


Figure 3: A cycle of interaction and neglect that illustrates RAD. The cross-hatched areas indicate transition regions where transition between neglect and interaction occurs.

istent NT and IT, the fraction of human time dedicated to the robot is given by  $RAD = \frac{IT}{NT+IT}$  where *RAD* denotes **R**obot **A**ttention **D**emand. Although a simple representation of true RAD, the RAD equation can help us determine where design

emphasis should be placed to maximize interaction effectiveness.

## 2.4 Free Time

We now turn attention to evaluating what RAD can tell us about designing systems that support effective interaction. The first metric of an efficacy is *free time*. Free time is a measure of how much time is left over for other tasks and is given by  $FT = 1 - RAD = \frac{NT}{NT+IT}$ .

If we want to learn how to maximize free time, we can generate a plot of FT versus NT and IT. Such a plot is shown in Figure 4. From the plot and from the equation, it is appar-

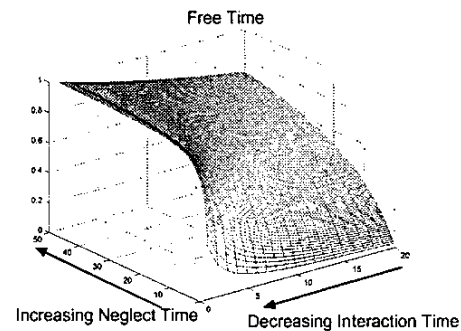


Figure 4: Free time grows most quickly with neglect time.

ent that free time is maximized when neglect time is maximized.

This is a fairly intuitive result; if we want a human to have as much free time as possible, then the robot should be as autonomous as possible. This is, however, only a minimum requirement for effective interaction (one that, for example, is necessary for dealing with communication latency). Because we want people to be able to perform multiple tasks during their free time, free time alone is insufficient to characterize efficient interaction.

## 2.5 Fan Out

Suppose, for the moment, that the free time available to a human will be used to manage a team of several homogeneous robots. *Fan out* is a measure of how many such robots can be effectively controlled by a human. Fan out depends on RAD and is bounded above by  $FO \leq \frac{1}{RAD} = \frac{NT+IT}{IT}$ . This bound occurs because attention is not the only resource consumed in a multi-robot environment. Of particular importance to multi-tasking is the role of working memory, since each task requires a corresponding mental model and some portion of short-term memory. Since working memory has limited capacity, a large number of tasks may saturate working memory and increase the amount of time required to obtain situation awareness.

Fan out can be maximized by minimizing IT. This conclusion is supported by plotting fan-out as a function of IT and NT. Such a plot is shown in Figure 5. From the plot, we see

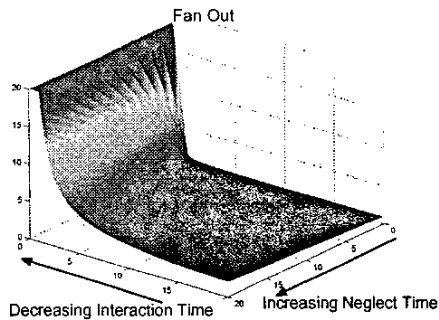


Figure 5: Fan out grows most quickly with interaction time.

that if NT is sufficiently high, IT becomes the critical factor in determining how many robots can be controlled.

This result simply means that if we want to be able to control a lot of robots with our available free time, then we should work on minimizing the amount of time spent with each robot. This can be extended to heterogeneous robots and general multi-tasking domains.

### 3 Making Interactions Efficient

In this section, we present seven principles for making interaction efficient. We also briefly state how these principles are compatible with the role of working memory in generating behavior. The principles are:

1. implicitly switch interfaces and autonomy modes,
2. let the robot use natural human cues,
3. manipulate the world instead of the robot,
4. manipulate the relationship between the robot and world,
5. let people manipulate presented information,
6. externalize memory, and
7. help people manage attention.

We will discuss and illustrate each principle in turn.

#### 3.1 Implicitly Switch Modes

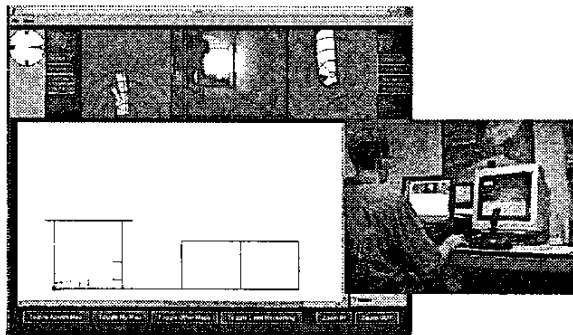


Figure 6: The control element and the region of the interface being manipulated implicitly determine the interaction.

It is often desirable to change the way in which an operator controls a robot and receives information from the robot. Such changes are sometimes mandated by the environment and sometimes made at the discretion of the human; which autonomy mode and interface elements are selected depends on the context established by the environment, communications channel, operator, etc. For example, when precise control is needed and a quality communication link is available, a user may switch from scripted control to pure teleoperation.

Such switches are necessary, but they should not place an undo burden on the operator. If the user wants to switch to teleoperation, then grabbing the joystick is sufficient evidence of the user's intent; the robot and interface should automatically adapt. Cognitive effort and attention should not be required. If the user wants to switch to waypoint planning, then dropping landmarks in a planview map is sufficient evidence of intent; the robot and interface should automatically adapt.

This is illustrated in Figure 6 which shows a typical user workstation and interface. The input device selected by the user (mouse, joystick, or keyboard, for the situation displayed) and the portion of the interface affected (planview, perspective view) determine the robot's autonomy mode. For example, if the user has been using the mouse to give waypoints in a plan view, but the user grabs the joystick, the robot should automatically switch to teleoperation mode. As a side note, if the robot and interface switch modes based on a response to an implicit user command, the interface should confirm the change to help avoid mode confusion.

In terms of cognitive information processing, the user should not need a mental model that tells them the steps required to switch interaction modes; knowing how to act in each mode should be sufficient knowledge.

#### 3.2 Use Natural Cues

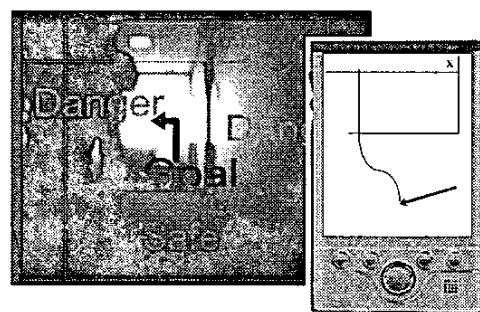


Figure 7: Sketches and labels are examples of interaction modes that are natural for people to use.

People have extensive experience in accomplishing tasks and in interacting with other people. With this experience comes a set of natural expressions. Most often, scientists emphasize the naturalness of speech in supporting natural interactions, but natural language is an elusive goal and many

other forms of natural expression are useful. Skubic has looked at sketch-based maps [12] as a means of expressing intent through natural cuing. Schultz has investigated gestures as an alternative [9].

Figure 7 illustrates both a simple sketch-based interface as well as an image-based interface. In the image-based interface, a user can express danger or affordance by dropping icons in the image, and the robot can then associate these natural notions with image features and other sensor signatures. This is illustrated by labelling walls with danger icons and free space with affordance icons. Such an interface is supported by work in semantic map-building [5] and in image-based identification of regions [4].

The sketch-based interface could support both natural mapping and iconic representation of regions-of-interest/risk. Since people frequently use sketches to give directions (including landmarks), it is possible that such an interface would employ previously existing mental models [12].

In terms of cognitive information processing, “naturalness” means that well-calibrated mental models are available, well-known sensory stimuli receive attention, and well-practiced use of short term memory is employed. Thus, naturalness is compatible with effective interaction because it invokes well-practiced response generation.

### 3.3 Directly Manipulate the World

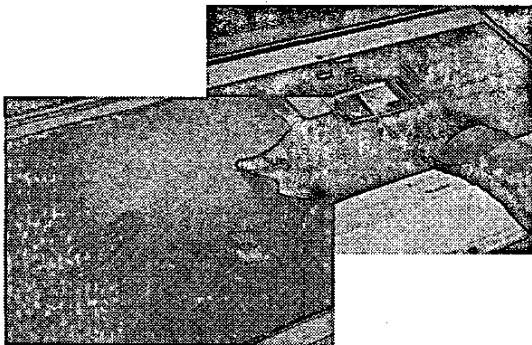


Figure 8: Touching the screen and having the robot automatically go to the location in the image is an example of allowing the user to directly manipulate the world without thinking about the robot.

The purpose of interacting with a remote robot is to accomplish some task in the world. Insofar as possible therefore, robot AI and interfaces should be designed so as to allow the task to be done, rather than drawing attention to the robot and the interface per se.

Some possible ways to accomplish this are illustrated in Figure 8. On the right side of the figure, the user is presented with a first-person perspective of what the robot's camera sees. If the user's goal is to gather information about the world, then it is natural to directly manipulate the image.

The user can touch a point in the image that it wants information about; the interface interprets this action as a command and the robot drives to the spot (via closed-loop feedback from the interface). On the left side of the figure, a user is presented a plan-view perspective of the robot's situation (obtained, for example, via a UAV). If the user's goal is to gather information about a certain region of the world, then it is natural for the user to touch that region and have the robot autonomously go to the location.

In terms of cognitive information processing, interacting with the world requires a mental model, and interacting with the robot requires a separate mental model. If the robot is transparent to the user, than only one mental model is required. This entails that working memory is less likely to be overtaxed with extra data in short-term memory and extra mental models. Thus, transparency is a desired element of efficient interaction.

### 3.4 Manipulate Robot-World Relationship

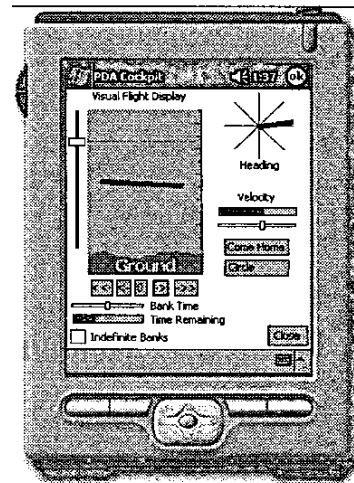


Figure 9: This UAV display presents the relationship between the airplane and the world.

It is sometimes difficult to develop interfaces and autonomy that directly supports world manipulation. Under these circumstances, human attention may need to be drawn to the robot. When attention needs to be drawn to the robot, it is most helpful if this attention remains focused on elements of the world and the task. More precisely, information regarding the status of robot in relation to a goal state or information that relates robot pose to world coordinates is useful.

This is illustrated in Figure 9 which shows a prototype interface developed for operating the BYU UAV via a personal digital assistant (PDA) (see <http://www.ee.byu.edu/ee/robotics/facilities/#uavs>). On the PDA, information about the UAV's altitude, pitch, and direction is presented. The display of this information follows conventional cockpit displays and includes some

very useful information about the relationship between the UAV and the world.

Unfortunately, the prototype interface requires that the user directly specify commands such as roll, pitch, etc. This requires that the user evaluate how these commands translate into resulting consequences in the real world for the task assigned. This means that a mental model which performs such a translation must be resident in working memory; since working memory is limited, this implies that fewer secondary tasks can be accomplished (e.g., fan-out is reduced).

Suppose instead that the user be allowed to grab the UAV's wingtip and pull it down or push it up to induce a bank. This could be displayed on the interface directly as a banking airplane in relation to the ground. Similarly, altitude could be selected by dragging the center of the airplane up or down. Provided that the UAV autonomy supports this level of command, the user burden is eased. This process of allowing the user to manipulate the relationship between the world and the robot rather than just manipulate the robot can be continued. Suppose that rather than requiring the user to specify a bank angle, the user simply dragged the nose of a UAV icon in the compass to a desired direction or location; the interface and UAV would coordinate to orient the UAV in the desired direction.

In terms of cognitive information processing, the relationship between robot and world must be known before a human can plan what the robot should do. Directly presenting information about this relationship allows the human to use only the mental model that generates behavior, rather than using not only this mental model but also the mental model that translates sensor data into such a representation. Furthermore, since the translation of data into a representation imposes a burden on short-term memory, removing this translation frees up short term memory resources.

### 3.5 Information is Meant to be Manipulated

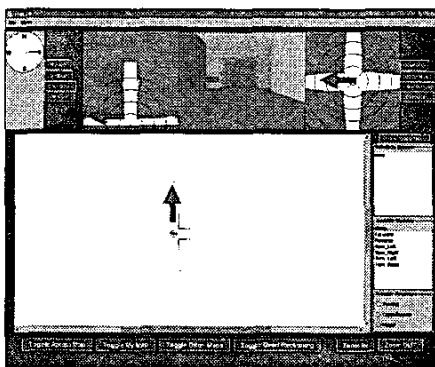


Figure 10: This display presents several representations of a left-turn; the interface should allow any representation to be used to task the robot.

One primary purpose of an interface is to present infor-

mation, primarily about the world, the relationship between the world and the robot, and about the robot. Consider the conventional interface illustrated in Figure 10. In this interface, information is presented about laser range finder and sonar ranges (upper left and right, respectively), global position and orientation (middle topological map and compass), and camera. This display could be augmented with robot status information, teaming information, etc.

When such information is displayed to a user, the purpose of the information is to support decision-making by the user. At perhaps the most fundamental level, this information is used to determine environment affordances (e.g., a "foo" or a path) and dis-affordances (e.g., obstacles or risky zones).

When affordances are presented to users, these (dis-) affordances invite interaction. For example, if an obstacle is presented then the user may wish to guide the robot around it. If an opening in a hallway is presented, then the user may wish to guide the robot toward it.

The interface should support interaction with the information presented. For example, the laser range finder, camera, sonar, and global map all present a perspective on turning left. The prototype interface allowed turns to be specified only through a joystick; or by clicking a mouse in either the camera view or the global view. This prototype did not support turning left between obstacles by allowing the user to click in the laser range finder or sonar.

In general, if information is presented to a user, the user should be able to manipulate this information directly and thereby guide the robot or make progress on a task. In terms of the cognitive information processing model, if information can be manipulated directly, there is no need for a mental model that translates this information into an action that will occur in a different modality.

### 3.6 Externalize Memory

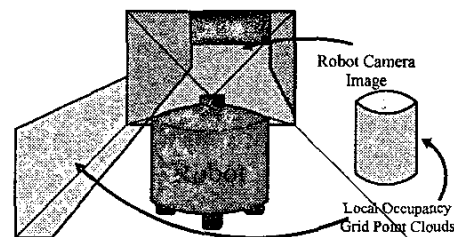


Figure 11: This display shows what obstacles have been seen in the past as well as the relationship between the robot and the world.

One of the difficulties with teleoperating a robot via a camera perspective is that the user cannot see where the "robot's shoulder are." A common occurrence in human machine interaction is when the user projects herself of himself into the machine so that the machine is an extension of the user. This greatly simplifies the task of keeping the correct mental models resident in working memory, but is limited if the sense of proprioception is missing. Without this sense, the user

must either (a) maintain all relevant information in short-term memory and then integrate this information into a mental representation, or (b) consult other sensors and integrate all sensors into a consistent whole. Both are hard to do and both place burdens on short-term memory. This can make the task of guiding a robot all-encompassing. In fact, this may be one reason why a single user cannot both guide a robot through a disaster area and find victims [2].

One way to simplify the cognitive load associated with navigation (and thereby support multi-tasking) is to externalize memory. A notional display that externalizes memory and integrates sensors is presented in Figure 11. In the figure, information about past sensor readings is included as an obstacle density field in the same field of view as the camera. (Additionally, the relationship between obstacles in the world and the robot is presented, thus satisfying the previous principle.) The user need not remember where all obstacles occurred once they are out of the camera's field of view, and the user need not integrate range information with the camera data.

### 3.7 Support Attention Management

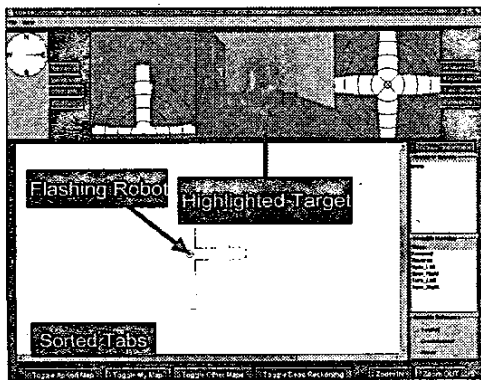


Figure 12: A flashing target, flashing robot, or sorted tab guides attention to the correct location.

Attention appears to be a major bottleneck in cognitive information processing. Even if sufficient information is presented to a user, if their attention is not on this information than incorrect decisions can be made. Thus, it is important for a user to properly manage attention.

A well designed interface and robot autonomy level should support proper attention management. For example, if a user is not attending to a relevant sensor and keeps running into an obstacle, the interface could highlight the information. Also, if one robot in a team needs attention, it could change colors, flash, or pop to the front of the attention queue [10].

In addition to these obvious attentional support devices, it is also important to help schedule attention. Since neglect time is stochastic, a user should schedule attention for the worst case scenario. To facilitate a service schedule based on the average case, an interface and robot autonomy mode

that supports an UNDO would be beneficial. Furthermore, such assistance would help users properly calibrate trust and thereby avoid misuse and abuse [6].

## 4 Summary and Future Work

We have reviewed the notions of neglect time and interaction time, and used these notions to motivate the importance of minimizing interaction time in a multi-tasking context. We then stated seven principles that make interactions more efficient, illustrated these seven principles with an example, and discussed why these principles are justified from an information-processing perspective.

Although it is a lot of fun to speculate on principles of efficient interaction, it is important to validate these principles in future work. The principles that we have chosen to present have arisen partly through trends uncovered through our experiments, and partly through experience in trying to design efficient interfaces. A suite of algorithms is being developed and a series of secondary task experiments is being conducted to validate these principles.

## References

- [1] A. Baddeley. *Working Memory*, volume 11 of *Oxford Psychology Series*. Oxford, 1986.
- [2] J. Casper and R. Murphy. Workflow study on human-robot interaction in "usar". In *Proceedings of ICRA 2002*, pages 1997–2003, 2002. Finalist for the Anton Phillips Best Student Paper Award.
- [3] K. A. Ericsson and P. F. Delaney. Long-term working memory as an alternative to capacity models of working memory in everyday skilled performance. In A. Miyake and P. Shah, editors, *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control*. Cambridge University Press, 1999.
- [4] J. A. Falls and D. R. Olsen Jr. Interactive machine learning. In *Intelligent User Interfaces*, 2002. Outstanding paper award.
- [5] B. Kuipers. The spatial semantic hierarchy. Technical Report AI99-281, 29, 1999.
- [6] J. D. Lee and K. A. See. Trust in automation: Designing for appropriate reliance. *Human Factors*, 2003. To appear.
- [7] A. Miyake and P. Shah, editors. *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control*. Cambridge University Press, 1999.
- [8] H. E. Pashler. *Attention??* The MIT Press, 1997.
- [9] D. Perzanowski, A. Schultz, W. Adams, E. Marsh, and M. Bugajska. Building a multimodal human-robot interface. *Intelligent Systems*, 16(1):16–21, Jan/Feb 2001.
- [10] J. Scholtz. Situational awareness for human-robot interaction. MARS 2020 Presentation, April 2003.
- [11] T. Shallice. *From Neuropsychology to Mental Structure*. Cambridge University Press, 1988.
- [12] M. Skubic, S. Blisard, A. Carle, and P. Matsakis. Hand-drawn maps for robot navigation. In *2002 AAAI Spring Symposium on Sketch Understanding Proceedings*, March 2002.
- [13] C. D. Wickens and J. G. Hollands. *Engineering Psychology and Human Performance*. Prentice Hall, third edition, 2000.