Protection from Human Error

Guarded Motion Methodologies for Mobile Robots

By Kevin S. Pratt and Robin R. Murphy

© ISTOCKPHOTO.COM/KOYA79

Industrial manipulators and unmanned systems often address a large number of tasks with some type of human-in-the-loop method. In these systems, the robot is given responsibility for some portion of the control tasks, but the human has some role for a variety of reasons; for example, the current technology may not be sufficient for the robot to complete the entire task: there may be safety, liability, or regulatory constraints, or the economics favor a human-in-the-loop process. An example of where human-in-the-loop control is of increasing interest is for telecommuting by health-care providers [1] and the general public [2] and for data gathering for disaster response [3]. These remote presence applications allow humans to perceive and act from a distance through a mobile-robot. Remote presence is more challenging than telesurgery and space telepresence from an interface perspective, as the operators are not expected to be highly trained on robots and will be working in dynamic or unpredictable environments.

To arbitrate potential conflicts between the human and robot controller, systems often implement a guarded motion functionality to integrate the competing commands. For the purposes of this article, guarded motion is defined as a method for monitoring and addressing safety constraints for human-directed operations when the human has inferior knowledge (compared to the robot) of the robot's pose and relation to its environment. In essence, guarded motion is a type of human-in-the-loop control where the robot guards itself from unintended consequences of human directives (collision avoidance, unstable configurations, excessive force, unnecessary power consumption, etc.).

Digital Object Identifier 10.1109/MRA.2012.2220500

Date of publication: 6 December 2012

1070-9932/12/$31.00©2012IEEE

This article reviews the literature on guarded motion for mobile-robots in order to capture the state of the practice and to identify open research issues. Understanding guarded motion is important and timely because unmanned systems for remote presence applications in business, military, medicine, law enforcement, and emergency response are moving from manual teleoperation toward increased shared autonomy. These robots are becoming equipped with manipulators and are working at ever larger distances from the operator. While progress is being made in autonomy, guarded motion is also needed as an intermediate step for fully autonomous systems.

This article surveys 32 systems using a novel taxonomy created to capture both control elements (autonomy intervention criteria, command integration method, monitored condition) as well as the interface characteristics (interface modality, display preprocessing). A taxonomy for guarded motion does not appear to exist, instead the variations of guarded motion are loosely grouped together under the category of supervisory control. As Sheridan describes in [4], manual control covers methods with no autonomy, autonomous control covers methods with trivial or no human intervention, and supervisory control covers all methodologies in between. Sheridan's levels of control have been refined by the levels of automation scale presented by Parasuraman et al. in [5]. In the levels of automation classification, guarded motion systems would then occupy levels 4–7; from "[The computer] suggests one alternative" to "[The computer] executes [the suggestion] automatically, then necessarily informs the human" [5]. This framework positions guarded motion as a middle ground where human and computer generated commands must be integrated, but does not provide specific guidance on what methods to use under what conditions.

History of Guarded Motion

Because guarded motion involves both manual and autonomous control schemes, it may seem a more complex control methodology and hence something that was developed later in the evolution of robotic systems. In fact, the concept of guarded motion has been around nearly as long as researchers have been working with robots. The idea of guarded motion was first proposed by Will and Grossman in 1975 as a *guarded move*, which they define as "a move until some expected sensory event occurs" [6]. In [6], they were constructing a robot arm for automated assembly, so guarded moves were most commonly employed to allow an arm to detect when it had made contact with a surface and to stop its commanded motion path.

As with many concepts in robotics, much of the early work was done with robotic arms and other types of telemanipulators. In 1980, Bejczy presented a summary of telemanipulator work being conducted at Jet Propulsion Laboratories (JPL), California Institute of Technology, Pasadena [7]. Though he did not use the term *guarded motion*, his description of shared control systems, where ". . . the computer is in series with the operator and transforms or modulates the operator's functional commands" [7], is one of the earliest explorations of how the concept of guarded motion might be best utilized. In his work, Bejczy discusses proximity sensors being developed for the Shuttle Remote Manipulator System (SRMS), force-torque sensors, and slip sensors, and presents results that show that the shared control systems being developed decreased movement time and increased placement accuracy when compared with strict manual control. Following Bejczy, there were contributions by Lee et al. in 1985 with the JPL general bilateral manipulator control [8], Backes and Tso in 1990 [9] and Backes et al. in 1991 [10] (the first to deal explicitly with the time delays introduced by teleoperating space-based systems) with the JPL UMI telemanipulator, and Hirzinger et al. in 1993 who described the ROTEX experiment that had flown on STS-55 [11].

It is not until 1996 that guarded motion concepts (most notably both Krotkov et al. and Simmons et al.'s work on a lunar rover testbed system [12], [13]) appeared to be applied to mobile-robots. Some of the concepts originally presented in both of these early papers have evolved and are still used by the latest guarded motion developments deployed on the Mars Exploration Rover (MER), as discussed separately by Baumgartner [14], Biesiadecki and Miamone [15], Trebi-Ollennu et al. [16], and Wright et al. [17].

While some works on guarded motion, notably Krotkov et al. [12] and Simmons et al. [13], preferred the term *safe-guarded teleoperation*, this article will follow the principle of priority from paleontological taxonomy, and give preference to the term first used, guarded motion, as presented in [6]. None of these sources give a clear definition of the term *safe-guarded teleoperation*, but the implicit definition shows that the two terms should be viewed as functionally synonymous, or at least as referring to the same core concept.

Taxonomy

The process of examining and classifying guarded motion implementations is fundamentally a study of how the system handles conflicts (axes 1 and 2), which variables the system monitors (axis 3), and how this information is presented to the operator (axes 4 and 5). Proceeding from the definition of guarded motion posited above, a five-axis system has been devised which accounts for these questions. These axes are

1) **Autonomy Intervention Criteria:** details when the system intervenes with the commands issued by the operator.
2) **Command Integration Method:** covers how the system integrates its commands with the user's commands.
3) **Monitored Condition:** describes which variables and environmental conditions the system monitors and accounts for.
4) **Interface Modality:** accounts for how the system presents information to the operator.
5) **Display Preprocessing:** discusses how much processing is performed on sensor data before it is presented to the operator.

Table 1. Control elements of guarded motion: autonomy intervention criteria, command integration method, and monitored condition.

	Autonomy Intervention Criteria		Command Integration Method		Monitored Condition				
System	Exception	Continuous	Traded	Blended	Obstacles	Pose	Energy	Health	Effector Force
Bologna Haptic Pioneer [18]		✓		✓	✓				
Bremen Autonomous Wheelchair [19]	✓	✓	✓	✓	✓				
CMU Pioneer [20]–[22]	✓	✓	✓	✓	✓	✓	✓	✓	
Dante II [23]	✓		✓		✓	✓			
ERA [24], [25]	✓		✓			✓			✓
ESA ROTEX [11]	✓			✓	✓	✓			✓
IBM Automated Assembler [6]	✓		✓			✓			✓
INL ATRV [26]–[28]		✓		✓	✓				
INL Packbot [29]		✓		✓	✓				
INL Pioneer [29]	✓		✓		✓				
JEMRMS-MA [24], [25]	✓		✓			✓			✓
JEMRMS-SFA [24], [25]	✓		✓			✓			✓
JPL General Bilateral Manipulator [8]	✓		✓			✓			
JPL Shared Space Telerobot [30]	✓		✓			✓			✓
JPL UMI Telerobot [9], [10]	✓		✓						✓
Lowell ATRV Jr. [31], [28]	✓			✓	✓				
MER [14]–[17]	✓		✓		✓	✓	✓	✓	
NASA 3T System [32]–[35]	✓		✓			✓			✓
NASA Ames K10 [36]	✓		✓		✓	✓	✓	✓	
NASA MESUR/Pathfinder [37]–[39]	✓		✓		✓	✓	✓	✓	
NASDA ETS-VII [40]–[42]	✓	✓	✓	✓		✓			✓
Postech pioneer [43]		✓		✓	✓				
Ratler [12], [13]	✓		✓		✓				
Robonaut [44]		✓		✓		✓			✓
Stanford Haptic Gripper [45]	✓		✓						✓
SPDM [46], [24], [25]	✓		✓			✓			✓
SRMS [33], [24]	✓		✓			✓			✓
SSRMS [24], [25]	✓		✓			✓			✓
Strathclyde ROV [47]		✓		✓	✓				
UofM NavChair [48], [49]		✓		✓	✓				
UrBot [50]	✓		✓		✓				
Wheelesley [51]	✓			✓	✓				

Table 2. Two interface components of guarded motion: modality and preprocessing.

System	Modality Count	Modality Type			Display Preprocessing		
		Visible	Audible	Tactile	Direct	Augmented	Virtual
Bologna Haptic Pioneer [18]	2			✓		✓	
Bremen Autonomous Wheelchair [19]	1			✓		✓	
CMU Pioneer [20]–[22]	3	✓	✓		✓		
Dante II [23]	1	✓					✓
ERA [24], [25]	1	✓			✓		
ESA ROTEX [11]	1	✓					✓
IBM Automated Assembler [6]	1	✓			✓		
INL ATRV [26]–[28]	2	✓		✓		✓	
INL Packbot [29]	2	✓		✓		✓	
INL Pioneer [29]	1	✓				✓	
JEMRMS-MA [24], [25]	1	✓			✓		
JEMRMS-SFA [24], [25]	1	✓			✓		
JPL General Bilateral Manipulator [8]	3	✓	✓	✓	✓		
JPL Shared Space Telerobot [30]	2	✓		✓			✓
JPL UMI Telerobot [9], [10]	2	✓		✓		✓	✓
Lowell ATRV Jr. [31], [28]	1	✓			✓		
MER [14]–[17]	1	✓					✓
NASA 3T System [32]–[35]	1	✓			✓	✓	✓
NASA Ames K10 [36]	1	✓					✓
NASA MESUR/Pathfinder [37]–[39]	1	✓					✓
NASDA ETS-VII [40]–[42]	1	✓					✓
Postech Pioneer [43]	1			✓		✓	
Ratler [12], [13]	1	✓					✓
Robonaut [44]	3	✓	✓	✓		✓	
Stanford Haptic Gripper [45]	2	✓	✓	✓		✓	
SPDM [46], [24], [25]	1	✓			✓		
SRMS [33], [24]	1	✓			✓		
SSRMS [24], [25]	1	✓			✓		
Strathclyde ROV [47]	1	✓					✓
UofM NavChair [48], [49]	1			✓		✓	
UrBot [50]	1	✓			✓		
Wheelesley [51]	1			✓	✓		

Table 1 describes the control element axes (1–3) and Table 2 presents the interface axes (4 and 5), with each table listing a range of robotic systems and noting how they are classified under this taxonomy.

Autonomy Intervention Criteria

The first of the differentiation axes, autonomy intervention criteria, describe when the system modifies the set of commands issued by the operator. The two potential values on this axis are exception and online. While the system is of course constantly monitoring in both cases, in the exception case, the system becomes fully activated if the parameter in question violates the set constraint condition, while in the online condition, the system is constantly adjusting the operator command input before it is sent to the effectors.

A common example of the exception-type intervention criteria is an emergency stop system. When the constraint is violated (the robot goes out of a GPS bounding box, it gets too close to an obstacle, or is about to enter an unsafe pose), the guarded motion behavior intervenes and stops the system to prevent it from causing any harm. Within the surveyed literature, 21 examples of exception-based intervention criteria were found. One clear example of an error condition update system, the MER, actually presents two distinct examples: the MER rover mobility and instrument deployment device (IDD) motion control software. Baumgartner [14], Biesiadecki and Maimone [15], and Wright et al. [17] all discuss different aspects of the MER mobility system, while Trebi-Ollennu et al. [16] discuss the IDD. In both cases, the system will execute a stream of commands at a given autonomy level, but if any of a myriad of safety conditions are violated, the system is halted, and operators on earth are alerted. Another example of an error condition system is presented by Griffin et al. in their discussion of a system for helping operators maintain appropriate grip force with a remote robotic hand through visual, auditory, and haptic alerts on over and under gripping conditions [45].

For online systems, the most direct example is a potential field system that is repulsed by obstacles. As an object comes closer and closer in the view of a range sensor, the system adds a proportionally increasing control vector in the direction opposite the detected obstacle. Even at longer distances where there is no danger to the robot, the system is modifying the user commands (though much less significantly) to help the operator avoid the obstacle. One example of an online system is the wheelchair with assistive navigation by Bell et al, which, in the presence of obstacles, commands the wheelchair in the unobstructed direction nearest to the direction commanded by the user and then reduces the chair's speed based on the amount of this deviation [48]. A second wheelchair by Rofer and Lankenau also uses an online methodology, but only to reduce the vehicle's speed as objects approach [19]. A final example of such an online system is presented by Diolaiti and Melchiorri [18]. Their system uses a virtual mass and spring model to apply control inputs to the rover as it approaches an obstacle.

Command Integration Method

The next axis of differentiation, command integration method, covers how the system behaves in the event of a violation of safety constraints. The systems handle this condition in one of two ways: either by assuming total control for the subsystem in question or by modifying the operator's commands to help avoid the hazard.

The logical example of traded control would be unmanned ground vehicles (UGVs) stopping during a drive action when they detect an obstacle in the directed path. Conversely, a simple example of a blended control response would be if this same UGV instead modeled the detected obstacle with a repulsive potential field and relied on the p-field methodology to appropriately divert the vehicle around the obstacle. As with the values on the autonomy intervention criteria axis, these are simply some straightforward examples, these values can be applied to vehicle mobility, manipulators, payload sensors, and across all available robot domains.

It is worth noting that traded control in this case does not fully follow the definition given by Sheridan. The given definition for traded and shared control is:

> The human may remain as a supervisor or may, from time to time, assume direct control (this is called traded control), or may act as supervisor with respect to control of some variables and direct controller with respect to other variables (shared control). [4]

Traded control in Sheridan's definition means that the human steps in and takes control for a period of time and then passes control back to the robot when they are done. In guarded motion, just the inverse of this happens. Control is still traded back and forth between man and machine, but in this case, the robot makes the decision to assume control from the operator and then relinquishes control to the human again once the given safety conditions are satisfied. The definition of blended control used here is also similar to the definition of shared control presented above, but instead of the human controlling some elements directly and the robot other elements, blended control has them both simultaneously controlling the same element and blending (through command summation or otherwise) the two sets of commands.

As previously seen with the autonomy intervention criteria, one clear example of traded control on a detected safety condition violation is the MER mobility and IDD software. In both systems, the command stream is executed until the goal condition is reached, or an error condition is reached. In the mobility software, a wide range, of conditions are monitored to make sure it is safe to drive, including sensors and vehicle health, vehicle pose, sufficient northerly tilt to maintain solar charge, time of day, and obstacles [14], [15], [17]. On the other hand, the IDD monitoring focuses on the joint sensors and the contact switch on the end of the arm (incidentally, this contact switch is also used to verify successful completion of an arm move when it makes contact in the expected position) [16]. While it could be tempting to say that simple traded control demonstrates a lower level of system sophistication or intelligence (such as when the robot is over 50 million km away), there are conditions when the absolute safety of the system is paramount, and thus the better system appears more cautious and less advanced than others. Krotkov et al. and Simmons et al. detail a similar planetary rover system in their initial development tests for a lunar rover system [12], [13]. A second example of traded control is the family of robotic arms used on the International Space Station and other spacecraft. This set of remote manipulators includes the SRMS, Space Station Remote Manipulator System (SSRMS), Special Purpose Dextrous Manipulator (SPDM), ETS-VII, Japanese Experiment Module Remote Manipulator System–Main Arm (JEMRMS-MA), Japanese Experiment Module Remote Manipulator System–Small Fine Arm (JEMRMS-SFA), ROTEX, and the European Robotic Arm (ERA). While each of these systems has a unique control system, they are all direct descendents of the original remote manipulator system, the Space Shuttle's SRMS, and all employ a traded control scheme which will stop all movement operations if a joint angle, force sensor, or other limitation is violated [33], [46], [24], [11], [40]–[42], [25].

On the other hand, an example of the blended control system is the assistive wheelchair navigation system presented by Bell et al.: user commands given to the wheelchair are modified with a potential field's representation of detected obstacles to drive the wheelchair in the safe direction closest to the intended user command. Similar wheelchair control methodologies can be seen in Rofer and Lankenau [19], Simpson and Levine [49], and Yanco [51], and this blended control can also be seen applied to UGVs in Bruemmer et al. [27], Goodrich et al. [52], Krotkov et al. [12], and Simmons et al. [13]. Griffin et al. illustrate a blended control system with a robotic manipulator assisting the operator in maintaining appropriate gripping force on an object [45].

Monitored Condition

The third axis, monitored condition, describes which variables and environmental conditions the system monitors and uses to reject or modify an operator directive. As noted in the definition of guarded motion,while it is tempting to constrain guarded motion simply to obstacle avoidance, guarded motion behavior can be used to monitor a much wider set of conditions than just obstacles. Unlike some of the other axes, the potential values on this axis are best represented by gradient or range of values, rather than distinct positions. In addition to obstacle avoidance, guarded motion methodologies can also be used to track vehicle pose, end-effector force, energy state and capacity, vehicle state, longitudinal system health, or any combination of these values (or numerous potential other conditions); due to space limitations, a discussion of only the most illustrative situations where the robot overrides the human is presented below.

Several examples of straightforward obstacle avoidance guarded motion have been presented in both the assistive navigation wheelchair systems as well as the UGV systems. The wheelchair systems tend to use sonar detection systems [48], [19], [49], while UGV systems see a mix between sonar, scanning laser ranger, and stereo (or other) vision systems [27]–[29], [31].

Regarding the other monitored conditions, Fong et al. demonstrate a controller that monitors not only obstacles but also pose (in the form of Euler angles to monitor for rollover) and system health (as energy state, overtemperature, motor stall, and controller failure) [20]. The other system of note for monitoring multiple data points is the MER system. The system not only watches for obstacles but also senses six degrees of freedom pose, suspension rocker bogie angles, energy state, charge capability (angle toward the sun), and system and component health (among others) [14], [15], [17].

With the early telemanipulator [8]–[10], [30], [6] and the remote manipulator systems operating in space [33], [46], [24], [11], [40]–[42], [25] it is more common for the system to monitor pose and effector force rather than monitor the overall operational envelope for obstacle avoidance.

Interface Modality

The next axis of classification, interface modality, encompasses the system interface and how the system presents information to the operator. Along this axis, systems tend to have elements from the three primary modalities (visual, audible, tactile), but they may use these elements in any of the possible combinations and permutations.

As prototypical examples, within monomodal systems, the large majority are visual systems only (as Table 2 shows, 17 visual-only systems versus zero audible-only, and four tactile-only). In simple cases, this can be only a video feed from the robot, but also includes video streams presented contemporaneously with other visual information such as numerical readouts, status messages, or video overlays. An example multimodal system might then add tactile feedback through a force-feedback controller and audible alerts to grab the operators' attention when they become intently focused on the visual display.

The field of interface design and modality is by its own right a well-established research area, readers are therefore referred to human–robot interaction (HRI) and human–computer interaction (HCI) research for numerous additional examples of monomodal, visual-based design systems (not to imply that HRI/HCI is only concerned with this type of interface, but simply that they have been well studied in this context).

However, as Table 2 indicates, monomodal interfaces are not strictly limited to the visual channel. The assistive wheelchair navigation systems already discussed all present their feedback as haptic information, either through a vibrotactile or force-feedback controller, or through changes to the chair's velocity, a haptic feedback sensed by the user's vestibular system rather than the traditional touch-based systems [48], [19], [49].

A majority of the newer UGV interfaces all discuss multimodal interfaces, either as one of the independent variables tested, or as simply the methodology chosen to develop the interface. The methods discussed include fully trimodal systems (visual, auditory, and tactile) as well as all possible bimodal configurations [27], [29], [18], [43], [53]. The robotic manipulator tests presented by Griffin et al. also evaluated the full gamut of multimodal systems with tactile, auditory, and visual alerts for gripper force [45]. This full range of possibilities is also discussed in Fong and Thorpe's survey of teleoperation interfaces (as indeed, modality applies to all such interfaces, not just guarded motion systems) [21].

Display Preprocessing

The fifth and final axis to consider, display preprocessing, is similarly concerned with how the data are presented to the operator, but from the perspective of how much and what type of processing is done to the data before they are presented to the user. The three potential values along our preprocessing axis are a direct feed, augmented reality, and virtual reality, using the most frequent occurrence of terminology from the papers themselves. As can be inferred, the direct feed simply presents the data collected by the robot directly to the user with a minimum of additional processing. *Augmented reality* then refers to a state where this basic feed has been overlaid or enhanced with additional elements of data or alternate representations of the given data. At its base, this augmented reality is still looking at the world perceived by the robot, which is in contrast to the virtual reality system where the underlying representation is a reconstruction of the environment; something not directly perceived by the robot (sensor readings are of course used in this virtual reality world, but to construct and modify the model presented to the user). We note that the boundary between augmented reality and virtual reality in older mobile-robot systems is becoming blurred, and newer mobile-robot systems may be better described as using *augmented virtuality* [54].

It should be noted that in their taxonomy of HRI, Yanco and Drury [55] identify sensor preprocessing as one of their classifiers as well. As used in the guarded motion taxonomy, we use a somewhat broader definition than the presentation in the HRI taxonomy. Yanco and Drury present two examples in their taxonomy (a sonar used to build a map, and a video with certain regions highlighted), which, in our taxonomy, would both be considered part of the middle, augmented, level, as some processing is performed on the sensor data in both cases, but the base of the display is still the direct sensor readings.

The clear example for all three of these cases is with a video stream presented to the user. In the direct-feed case, the video from the robot's camera is simply shown to the operator. The augmented reality then enhances that video stream with an overlay such as detected obstacles, navigation and goal position markers, or numerical readouts of pose, energy management, and other vehicle state information. The virtual reality interface would instead use the vehicle's detected position and relation to obstacles to construct a three-dimensional (3-D) model of the operating environment, place the robot in the environment, and then present a view of this constructed environment to the operator such that the operator is controlling the vehicle in this virtual environment, and the physical agent then is given commands to mimic the actions of its virtual counterpart.

While there are undoubtedly a large number of examples throughout the teleoperation and HRI literature, the most elementary and easiest to understand of these interface types is the direct feed. Two examples of this type are the summary of visual robotic interface elements compiled by Ellis [56], and the survey of teleoperation interfaces compiled by Fong and Thorpe [21]. Ellis discusses a broad range of elements used in many of the early visual teleoperation interfaces, while Fong and Thorpe discuss teleoperation interfaces as a whole, breaking their discussion out into direct rate controller interfaces, multimodal/multisensor systems, supervisory control, and novel controller systems.

Alternately, examples of the augmented display style can be seen in many of the UGV interfaces: particularly, clear examples are available in Fong et al. [22] and Bruemmer et al. [27].

An example of the third and final option, virtual reality is the rover sequencing and visualization program (RSVP) used by JPL to control the MER rovers. Using data collected by the stereo imagers and the HazCams on the rovers, a local 3-D model of the rover's environment is generated, and this model is then shown to the operator through RSVP. The operators then drive the rover's path for the next socially optimized learning in the virtual environment of RSVP. Based on the operator's commands to the virtual rover, a command sequence is then generated which is then fine-tuned and sent to the rover for execution. In the case of the MERs, this virtual reality capability is necessary to overcome the time delay to and from Mars; if the robot is not in a given location to take video, this view must be generated in the virtual reality space [14], [15], [17].

Such methodologies of controlling a virtual robot through a virtual environment can also be used in real time, when the video is insufficiently clear to operate from, or temporal data (such as past map elements) is difficult for operators to recall quickly and accurately [20], [27]–[29], [31]. Virtual reality can also be used not only when video is degraded but also unavailable. Lin and Kuo describe a system for operating a remotely operated vehicle (ROV) near the foundation structure of an offshore oil rig. In such a subsea environment, with high turbidity and absorptivity, insufficient light means video is not a viable sensor, except immediately in front of the vehicle where local light systems can overcome these problems. Sonar, on the other hand, can map the structure of the oil rig viewed from the ROV and then be used to register the vehicle in the water to its location to be updated within the virtual model [47].

Discussion

In addition to the taxonomy and synthesis of design heuristics, the survey of the literature provides three general observations about the history and general nature of guarded motion and identifies four open research questions for applying guarded motion to mobile-robots.

Observations

Examining the taxonomical results above, we identify three quantitative points that emerge from the evaluated literature. These observations are listed with a brief note discussing the point.

All taxonomical options had been tried at least once by 1990, except for energy and health management: Within the classifications, we looked for any natural groupings or clusters that appeared along the time axis. Rather than finding that guarded motion strategies had evolved from one type to another or that there was a clear order of development, the data revealed that, except for two particular cases, all of the potential classifications had been investigated at least once by 1990. These two outstanding cases are the energy and health-monitored conditions, which appear in the literature by 1995. Both of these conditions are particularly relevant to mobile-robot systems and systems designed for extended field service. As many of the earlier systems reviewed were either manipulator systems (where power is continuously available), or proof-of-concept vehicles designed for lab use (where it can be expected that repairs and overall system health monitoring will be performed by the researchers), it is little surprise that the first discussion of monitoring these items appears with the Mars Pathfinder mission and the Sojourner rover [37]–[39].

Obstacle avoidance, pose, and effector force are the most commonly monitored conditions: As Table I illustrates, within the surveyed literature, there were 16 examples of a guarded motion behavior being used for obstacle avoidance, 16 occurrences of it monitoring actuator/effector pose, and 14 examples where guarded motion behavior was used to monitor effector force. While there were some cases of overlap into the obstacle avoidance category (the ESA ROTEX arm [11] and the MER IDD [16] both had obstacle detection capabilities), these three common categories were primarily aligned with the type of robot: mobile systems used obstacle avoidance, while manipulators tended to monitor pose and effector force. While crossover is indeed possible, as noted above, such distinct differentiation indicates a tacit agreement by researchers on the ecological needs each type of robot encounters. Mobile-robotics typically operate in an at least partially unmodeled environment and must therefore react to objects and obstacles in the environment, while manipulator systems are typically used in a more engineered environment and are more concerned with how they interact with objects in their environment (touch or grasp an object without either slipping or crushing it).

More examples of guarded motion were found in mobile-robot systems than manipulator robots, but several examples of each exist: While not explicitly called out, Table 1 contains 20 mobile-robot systems and 13 manipulator systems that were found to employ guarded motion in fashion. While we do not claim that this table contains all systems which have ever used guarded motion, this sampling certainly shows that guarded motion can be, and has been, used on both types of robots. As the previous paragraph notes, it may be employed to monitor different conditions and perform a different function within the overall architecture, but guarded motion is still a valuable design tool for engineers constructing either types of system.

Open Research Questions

In addition to the above-noted findings, reviewing the classified guarded motion systems also raised four questions for discussion and potential future research. These discussion points are that there is no consistent method or metrics used in evaluating guarded motion systems, that visual feedback has been used almost exclusively as the primary display modality, it was unclear what the role of guarded motion would be within some of the autonomous systems currently under development, and finally, if there were conditions where guarded motion techniques would not be effective and they should not be employed. These questions will be addressed as follows:

No Consistent Methods/Metrics, Most Systems Are Proof-of-Concept

The obvious question to ask when classifying different guarded motion techniques is "which technique is best?" Discounting for the moment the fact that the answer to that question is highly task-dependent and that there, indeed, may be no distinct answer, it may be all but impossible to evaluate that question with post hoc analysis given the technique employed throughout this literature. The guarded motion systems surveyed were predominantly proof-of-concept or demonstrator systems (the exception being the fielded research platforms such as Pathfinder or MER, but these were still descriptions of systems designed to accomplish one particular task or set of tasks), and they did not employ any consistent metrics or design methods for evaluating and selecting a guarded motion methodology. Developing such a common metric for evaluating a guarded motion design or a methodology for selecting one particular design over another would be a valuable future research topic for engineers to make more informed decisions about the system as it is being designed. Considering operator/pilot situation-awareness testing as an example, while they certainly do not answer all questions, the development of situation- awareness metrics such as situation-awareness global assessment technique (SAGAT) [57] or situation-awareness rating technique (SART) [58] has allowed developers and, later, readers of the literature, to make more meaningful comparisons across systems even when there are disparities between the overall designs.

Predominance of Vision Interface

Except for the semiautonomous wheelchairs and two mobile-robots explicitly to test haptics, the primary display modality for all other systems was visual. While there were no other nonvisual systems to corroborate this conjecture, the collaborative agreement appears to be that the visual modality should be the primary interface modality, except when the operator is colocated with the robot and can directly observe the environment without computer mediation. While there appears to be no reason to contest the visual modality as the optimal channel to present guarded motion information to an operator, there also appears to be no experiments evaluating this assumption. While we would not expect the hypothesis to be rejected, we do note it as a potential area for future research.

Conditions Where Guarded Motion Does Not Work Well or Should Not Be Used

An area of significant concern is: Are there conditions where guarded motion does not work? Many of the surveyed systems included multiple levels of control, one of which was commonly a direct teleoperation interface (used for diagnostics and system checkout). While they could be controlled through direct teleoperation, or have individual actuators manipulated directly as described for Dante II [23], this was not the default operating method, but was used only under special conditions. This raises the question however: Are there any tasks or conditions where designers should not include a guarded motion type capability at all? Returning to the definition of guarded motion presented earlier, the only time guarded motion would not be useful would be when it was instead the robot that had inferior understanding and the human who had the superior knowledge of the robot's pose and relation to the environment. Given the known limitations of mediated perception and other detrimental factors, this is likely only a theoretical case, but nonetheless could certainly be investigated in the future.

Role of Guarded Motion in Autonomous Systems
Another question that should be raised in relation to guarded motion systems is: What is the role of guarded motion in relation to autonomous systems? Given that guarded motion was first explored in the late 1970s and early 1980s, and that there has been an increasing focus in the last decade on more fully autonomous systems, is guarded motion still a relevant technique? How does it fit in with these systems? However, Murphy and Burke [3] identify a large class of search and observation tasks, called remote presence applications, that will likely always be conducted as human-in-the-loop, and it is expected that these tasks would always benefit from a guarded motion capability. Given the presence of remote presence applications and the surprising historical pervasiveness of guarded motion seen in the section describing the history of guarded motion, it would be incorrect to assume that guarded motion is simply an old technique and will no longer be useful in the future.

Conclusion
This survey of the literature on guarded motion for mobile-robots showed that there are over 40 successful systems where the robot overrides the human directives to maintain safety, especially to avoid collisions. Given that almost every combination of intervention mechanism, command integration method, monitored conditions, interface modality, and display preprocessing were incorporated into a workable system, it was difficult to determine what works best. Instead, conclusions were divided into 1) heuristics for design that were directly supported by the studies reviewed in this article and 2) additional, more speculative design guidelines organized around the five axes in the taxonomy.

Heuristics for Design
An examination of the guarded motion systems identified in the above taxonomy yields three general findings about such systems that can help inform the incorporation of guarded motion into a robot system.

Time delay in the command loop suggested that exception-based autonomy intervention criteria be used as the control element and predictive simulation as the display element. In 1986, Sheridan clearly showed the benefits of using predictive simulation to generate operator displays when teleoperating a robot with a time delay [59]. All of the guarded motion systems surveyed that dealt with a time delay heeded Sheridan's findings and used some form of predictive simulation in their operator displays. But these time-delayed systems also had another feature in common: exception-based autonomy intervention criteria. When considering a time-delayed guarded motion system, it is understandable why exception-based autonomy intervention has been consistently selected. In a guarded motion system, autonomy intervention occurs when the operator's understanding of the robot's environment has degraded to such a degree that a safety constraint has been violated. Regardless of why this misunderstanding has occurred, the best case response will be the round-trip signal time (the return trip to alert the operator of the problem, and the outbound trip to send the correction), for any delay that is long enough to be called a delay, this is enough time for a robot to go from violating a safety constraint to causing a safety incident. To ensure safety in such a system, the robot must assume full authority when a constraint violation occurs and must allow the operators to correct their understanding of the situation so that operations can continue within the performance envelope.

Ranged exteroception is necessary for continuous autonomy intervention case. Examining the systems that used continuous autonomy intervention revealed that these systems shared a second commonality as well. All of the surveyed guarded motion systems with continuous intervention systems also employed ranged exteroceptors. This finding makes intuitive sense as well: proprioception provides binary (yes/no) indication of constraint violation, while ranged exteroception gives continuous look-ahead capability so the guarded motion system can adapt before a constraint violation occurs. Thus, we found that the surveyed guarded motion systems indicate that ranged exteroception is a necessary capability for a continuous intervention system.

Clusters of applications exist which provide *de facto* design recommendations. In examining the classified systems, we identified three consistent clusters within the available configuration space: while they did not partition the whole space, they did provide a design recommendation for the specific task that each cluster represented. The three identified tasks with clustered designs are: robot manipulators in space, rovers, and wheelchairs. For robotic manipulators in space, all of the systems surveyed (ERA [24], [25], ESA ROTEX [11], JEMRMS-MA and JEMRMS-SFA [24], [25], SPDM [46], [24], [25], SRMS [33], [24], and SSRMS [24], [25]) used an exception-type autonomy intervention criteria, a traded command integration method, monitored pose, and effector force, with a visual interface rendering a virtual representation of the robot and the task. Rovers on the other hand (such as the Pathfinder/Sojourner [37]–[39], the twin MERs [14]–[17], and the NASA Ames K10 [36]) all used exception-type autonomy intervention criteria, traded command integration, monitored obstacles, pose, energy, and system health for constraint violation, and presented the operator with a visual interface based on a virtual representation of the rover's state. While these first two systems lay somewhat close within the configuration space, the third category, semiautonomous wheelchairs, occupied a very different region of the design space (as one can imagine, the design requirements for a wheelchair and a remote scientific rover are themselves quite different). The wheelchair systems (the Bremen Autonomous Wheelchair, the University of Maryland NavChair, and Wheelesley) were all continuous intervention systems with blended command integration monitoring obstacles and presenting the operator information through a tactile interface based on a direct representation of the robots environment. The one exception to this wheelchair design heuristic, Wheelesley (which used an exception-type system with traded command integration), proves the rule and makes the overall point of these clustered designs. Given the strong convergence of each of these design types, future designers of such systems should consider these archetypal systems as a baseline or reference design for their systems. These systems should serve as a design inspiration but not a limitation. Future engineers should also identify where their specific design requirements diverge from the established patterns and identify how those differences should influence the guarded motion configuration they employ. Returning to the Wheelesley example, this wheelchair was designed for users with cognitive and fine motor impairments, something that had not been a requirement for the other systems. Given these additional obstacles, these users would have a significantly more difficult time to smoothly and quickly adapting their inputs based on the obstacles the chair had detected, suggesting a different type of autonomy and an alternate command integration method.

Additional Guidelines by Axis

The three heuristics captured the overall system-wide trends; however, a mobile-robot designer may construct a system as a series of design tradeoffs from within each of the five taxonomic axes in the section on taxonomy. The predictors of success for each element in the taxonomy are less clear that the three heuristics, but this section speculated on the issues and choices associated with each control or display element.

The surveyed studies showed that exception-based autonomy intervention is essential for applications with long-time delays, but the choice is less clear for applications where latency is not a problem. The choice of exception-based or continuous autonomy intervention appears to be a fundamental design decision. Is the robot fully autonomous except when an exceptional event occurs, and then control must shift back to the human, or are the robot and human continuously working together? The studies offer little insight into the answer, but continuous autonomy intervention appears attractive for applications where the human is actively involved and, is working in a complex, open environment where reaction is more important (or possible) than accurate preplanning.

Rather than choosing a single type of command integration, it may be desirable for a mobile-robot to have both traded and blended methods. The regime should always have at least one instance of traded control, where the robot overrides the human directive to perform another action, if only for an emergency stop or staying within a bounding box. Blended control, where the robot modifies or adapts human directives (such as avoids obstacles while moving in the intended direction), should be considered as the default teleoperation mode. Even if a robot calls an emergency stop and the human begins to directly control the robot, the human may not have a useful perceptual vantage point or may be encountering a significant time lag; thus the robot may be able to better sense and react to the environment.

The choice of monitored conditions for the robot to guard against depends on the task. The majority of cases where the robot took over or adapted human commands involved mobility conditions, but internal and indirect conditions are also a concern. For example, the robot might shut down rather than allow a human to burn up a motor while spinning in place. The studies surveyed in this article showed no consensus on which conditions to monitor but there was a sense that the more robust designs had a larger number of monitored external and internal conditions.

While the choice of interface modality is dependent on the specific tasks, in practice, designers have relied almost exclusively on visual displays. This practice appears to stem from convenience rather than a conscious design decision. Designers are encouraged to conduct HRI analyses to determine the division of functions between human and robot and then to apply good human–computer interface design principles so as to enable the human to realize those functions. In general, there will be at least two functions of a guarded motion interface, one is to display the nominal information needed for the nominal control regime (e.g., blended control, supervisory control) and the other is to determine why the robot has assumed control (e.g., why an emergency stop was issued). Force feedback joysticks appear appropriate for blended navigational control, and auditory or tactile alerting may be useful for signaling exceptions.

Display preprocessing is a must for human understanding of applications involving large time delays or when unusual sensors or perceptual representations are used. The appropriate amount of simulation or virtuality needed to understand why a robot has assumed control or to be confident in traded or blended control depends on the task ecology.

Summary

This article surveyed 32 manipulator and mobile-robot systems that rely on guarded motion. Guarded motion provides a methodology for the robot to track and monitor safety conditions it is better suited to observe and then integrate its findings with the command sequence provided by the human operator. It has been used in some form since 1975 and will likely play an important role in future systems for military, medicine, law enforcement, and emergency response applications as they shift from teleoperation to increased autonomy, where the advances in autonomy are not sufficient or mission, safety, regulatory, or economic concerns require human participation. To classify and analyze the surveyed systems, a novel five-axis taxonomy for guarded motion was created. The axes are autonomy intervention criteria, command integration method, interface modality, display preprocessing, and monitored condition. These differentiators cover when and how the system intervenes, how this information is presented to the operator, and finally, what conditions are monitored by the system. The taxonomy is expected to be capable of accounting for all of the systemic differentiations between any future guarded motion implementations and serves as formal basis for comparing designs. The article also contributed a set of design heuristics on what control and display elements to use for certain conditions and identified several open research questions. The survey, taxonomy, heuristics, and discussion are expected not only to add to the fundamental theory of autonomy and HRI but also serve as a practical guide to implementing a guarded motion system.

Acknowledgment

This work was supported in part through Lockheed Martin and by NSF grants IIS-1143713 and CNS-0923203. The authors would like to thank R. Grant for his encouragement.

References

[1] J. M. Beer and L. Takayama, "Mobile remote presence systems for older adults: Acceptance, benefits, and concerns," in *Proc. IEEE/ACM Int. Conf. Human-Robot Interaction,* 2011, pp. 19–26.

[2] K. M. Tsui, M. Desai, H. A. Yanco, and C. Uhlik, "Exploring use cases for telepresence robots," in *Proc. IEEE/ACM Int. Conf. Human-Robot Interaction*, 2011, pp. 11–18.

[3] R. Murphy and J. Burke, "From remote tool to shared roles," *IEEE Robot. Autom. Mag.*, vol. 15, no. 4, pp. 39–49, 2008.

[4] T. B. Sheridan, *Telerobotics, Automation, and Human Supervisory Control.* Cambridge, MA: MIT Press, 1992.

[5] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," *IEEE Trans. Syst., Man Cybern., Part A: Syst. Humans,* ., vol. 30, no. 3, pp. 286–297, May 2000.

[6] P. M. Will and D. D. Grossman, "An experimental system for computer controlled mechanical assembly," *IEEE Trans. Comput*., vol. C-24, no. 9, pp. 879–888, Sept. 1975.

[7] A. K. Bejczy, "Sensors, controls, and man-machine interface for advanced teleoperation," *Science*, vol. 208, no. 4450, pp. 1327–1335, 1980.

[8] S. Lee, G. Bekey, and A. Bejczy, "Computer control of space-borne teleoperators with sensory feedback," in *Proc. 1985 IEEE Int. Conf. Robotics Automation*, Mar., vol. 2, pp. 205–214.

[9] P. G. Backes and K. S. Tso, "UMI: An interactive supervisory and shared control system for telerobotics," in *Proc. 1990 IEEE Int. Conf. Robotics Automation*, May, vol. 2, pp. 1096–1101 .

[10] P. G. Backes, K. S. Tso, T. S. Lee, and S. Hayati, "A local-remote telerobot system for time-delayed traded and shared control," in *Proc. 5th Int. Conf. Advanced Robotics, Robots Unstructured Environments*, June 1991, vol. 1, pp. 243–248.

[11] G. Hirzinger, B. Brunner, J. Dietrich, and J. Heindl, "Sensor-based space robotics-rotex and its telerobotic features," *IEEE Trans. Robot. Autom*., vol. 9, no. 5, pp. 649–663, Oct. 1993.

[12] E. Krotkov, R. Simmons, F. Cozman, and S. Koenig, "Safeguarded teleoperation for lunar rovers: From human factors to field trials," in *Proc. IEEE Planetary Rover Technology Systems Workshop*, 1996.

[13] R. Simmons, L. Henriksen, L. Chrisman, and G. Whelan, "Obstacle avoidance and safeguarding for a lunar rover," in *Proc. AIAA Forum Advanced Developments Space Robotics*, 1996, pp. 1–9.

[14] E. T. Baumgartner, "Motion planning technologies for planetary rovers and manipulators," in *Proc. Int Workshop Motion Planning Virtual Environments*, 2005.

[15] J. J. Biesiadecki and M. W. Maimone, "The mars exploration rover surface mobility flight software driving ambition," in *Proc. 2006 IEEE Aerospace Conf.*, Mar., pp. 4–11.

[16] A. Trebi-Ollennu, E. T. Baumgartner, P. C. Leger, and R. G. Bonitz, "Robotic arm in-situ operations for the mars exploration rovers surface mission," in *Proc. 2005 IEEE Int. Conf. Systems, Man Cybernetics*, Oct., vol. 2, pp. 1799–1806.

[17] J. Wright, F. Hartman, B. Cooper, S. Maxwell, J. Yen, and J. Morrison, "Driving on mars with rsvp," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 37–45, June 2006.

[18] N. Diolaiti and C. Melchiorri, "Teleoperation of a mobile robot through haptic feedback," in *Proc. IEEE Int. Workshop Haptic Virtual Environments Their Applications*, 2002, pp. 67–72.

[19] T. Rofer and A. Lankenau, "Ensuring safe obstacle avoidance in a shared-control system," in *Proc. 7th IEEE Int. Conf. Emerging Technologies Factory Automation*, 1999, vol. 2, pp. 1405–1414.

[20] T. W. Fong, C. Thorpe, and C. Baur, "A safeguarded teleoperation controller," in *Proc. IEEE Int. Conf. Advanced Robotics*, Aug. 2001, pp. 1–6.

[21] T. Fong and C. Thorpe, "Vehicle teleoperation interfaces," *Autonom. Robot.*, vol. 11, no. 1, pp. 9–18, July 2001.

[22] T. Fong, C. Thorpe, and C. Baur, "Advanced interfaces for vehicle teleoperation: Collaborative control, sensor fusion displays, and remote driving tools," *Autonom. Robot.*, vol. 11, no. 1, pp. 77–85, 2001.

[23] T. W. Fong, H. Pangels, D. Wettergreen, E. Nygren, B. Hine, P. Hontalas, and C. Fedor, "Operator interfaces and network-based participation for dante ii," in *Proc. SAE 25th Int. Conf. Environmental Systems*, July 1995, pp. 1–12.

[24] N. J. Currie and B. Peacock, "International space station robotic systems operations a human factors perspective," *Human Factors Ergon. Soc. Ann. Meeting Process.*, vol. 46, no. 5, pp. 26–30, 2002.

[25] P. Laryssa, E. Lindsay, O. Layi, O. Marius, K. Nara, L. Aris, and T. Ed, "International space station robotics: A comparitive study of ERA, JEMRMS, and MSS," in *Proc. 7th ESA Workshop Advanced Space Technologies Robotics Automation*, Nov. 2002.

[26] M. Baker, R. Casey, B. Keyes, and H. A. Yanco, "Improved interfaces for human-robot interaction in urban search and rescue," in *Proc. 2004 IEEE Int. Conf. Systems Man Cybernetics*, Oct., vol. 3, pp. 2960–2965.

[27] D. J. Bruemmer, D. A. Few, and C. W. Nielsen, "Spatial reasoning for human-robot teams," in *Emerging Spatial Information Systems Applications*, B. Hilton, Ed. Hershey, PA: Idea Group Inc., 2006, pp. 350–372.

[28] H. A. Yanco, B. Keyes, J. L. Drury, C. W. Nielsen, D. A. Few, and D. J. Bruemmer, "Evolving interface design for robot search tasks: Research articles," *J. Field Robot*., vol. 24, nos. 8–9, pp. 779–799, 2007.

[29] D. J. Bruemmer, C. W. Nielsen, and D. I. Gertman, "How training and experience affect the benefits of autonomy in a dirty-bomb experiment," in *Proc. 3rd 2008 ACM/IEEE Int. Conf. Human Robot Interaction*, New York, NY, pp. 161–168.

[30] S. Hayati and S. Venkataraman, "Design and implementation of a robot control system with traded and shared control capability," in *Proc. 1989 IEEE Int. Conf. Robotics Automation*, May, vol. 3, pp. 1310–1315.

[31] H. A. Yanco, M. Baker, R. Casey, B. Keyes, P. Thoren, J. L. Drury, D. A. Few, C. W. Nielsen, and D. J. Bruemmer, "Analysis of humanrobot interaction for urban search and rescue," in *Proc. IEEE Int. Workshop Safety, Security, Rescue Robotics*, Aug. 2006, pp. 1–7.

[32] R. P. Bonasso, D. Kortenkamp, D. P. Miller, and M. Slack, "Experiences with an architecture for intelligent, reactive agents," in *Intelligent Agents II Agent Theories, Architectures, Languag°es* (Lecture Notes Computer Science), vol. 1037. Berlin, Germany: Springer-Verlag, 1996, pp. 187–202.

[33] R. Bonasso, R. Kerri, K. Jenks, and G. Johnson, "Using the 3T architecture for tracking shuttle rms procedures," in *Proc. 1998 IEEE Int. Joint Symp. Intelligence Systems*, May, pp. 180–187.

[34] G. A. Dorais, R. P. Bonasso, D. Kortenkamp, B. Pell, and D. Schreckenghost, "Adjustable autonomy for human-centered autonomous systems," in *Proc. 16th Int. Joint Conf. Artificial Intelligence Workshop Adjustable Autonomy Systems*, 1999, pp. 16–35.

[35] L. Sim, M. Cummings, and C. A. Smith, "Past, present and future implications of human supervisory control in space missions," *Acta Astronaut.*, vol. 62, nos. 10–11, pp. 648–655, 2008.
[36] T. W. Fong, M. Allan, X. Bouyssounouse, M. G. Bualat, M. Deans, L. Edwards, L. Flükiger, L. Keely, S. Y. Lee, D. Lees, V. To, and H. Utz, "Robotic site survey at Haughton Crater," in *Proc. 9th Int. Symp. Artificial Intelligence, Robotics, Automation Space*, Feb. 2008.
[37] D. Shirley and J. Matijevic, "Mars pathfinder microrover," *Auton. Robot.*, vol. 2, no. 4, pp. 283–289, 1995.
[38] H. W. Stone, "Design of the MESUR/pathfinder microrover," in *Proc. NASA Johnson Space Center, 7th Annu. Workshop Space Operations Applications Research*, Jan. 1994, vol. 1, pp. 2–10.
[39] H. Stone, "Mars pathfinder microrover: A small, low-cost, low-power spacecraft," in *Proc. AIAA Forum Advanced Developments Space Robotics*, Aug. 1996, pp. 1–9.
[40] T. Kasai, M. Oda, and T. Suzuki, "Results of the ETS-7 mission: Rendezvous docking and space robotics experiments," in *Proc. 5th Int. Symp. Artificial Intelligence, Robotics, Automation Space*, June 1999, pp. 299–306.
[41] M. Oda, "System engineering approach in designing the teleoperation system of the ETS-VII robot experiment satellite," in *Proc. 1997 IEEE Int. Conf. Robotics Automation*, Apr., vol. 4, pp. 3054–3061.
[42] W.-K. Yoon, T. Goshozono, H. Kawabe, M. Kinami, Y. Tsumaki, M. Uchiyama, M. Oda, and T. Doi, "Model-based space robot teleoperation of ETS-VII manipulator," *IEEE Trans. Robot. Autom.*, vol. 20, no. 3, pp. 602–612, June 2004.
[43] S. Lee, G. S. Sukhatme, G. J. Kim, and C.-M. Park, "Haptic control of a mobile robot: A user study," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots System*, 2002, vol. 3, pp. 2867–2874.
[44] M. A. Diftler, C. J. Culbert, R. O. Ambrose, J. R. Platt, and W. J. Bluethmann, "Evolution of the NASA/DARPA robonaut control system," in *Proc. IEEE Int. Conf. Robotics Automation*, Sept. 2003, vol. 2, pp. 2543–2548.
[45] W. B. Griffin, W. R. Provancher, and M. R. Cutkosky, "Feedback strategies for telemanipulation with shared control of object handling forces," *Presence*, vol. 14, no. 6, pp. 720–731, Dec. 2005.
[46] E. Coleshill, L. Oshinowo, R. Rembala, B. Bina, D. Rey, and S. Sindelar, "Dextre: Improving maintenance operations on the international space station," *Acta Astronaut.*, vol. 64, nos. 9–10, pp. 869–874, 2009.
[47] Q. Lin and C. Kuo, "Virtual tele-operation of underwater robots," in *Proc. IEEE Int. Conf. Robotics Automation*, Apr. 1997, vol. 2, pp. 1022–1027.
[48] D. A. Bell, J. Borenstein, S. P. Levine, Y. Koren, and J. Jaros, "An assistive navigation system for wheelchairs based upon mobilerobot obstacle avoidance," in *Proc. IEEE Int. Conf. Robotics Automation*, May 1994, pp. 2018–2022.
[49] R. C. Simpson and S. P. Levine, "Adaptive shared control of a smart wheelchair operated by voicecontrol," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots Systems*, Sept. 1997, pp. 622–626.
[50] L. Matthies, Y. Xiong, R. Hogg, D. Zhu, B. Kennedy, M. Hebert, R. Maclachlan, C. Won, T. Frost, G. S. Sukhatme, M. C. McHenry, and S. Goldberg, "A portable, autonomous, urban reconnaissance robot," in *Proc. IEEE Int. Conf. Robotics Automation*, July 2000, pp. 1842–1847
[51] H. A. Yanco, "A robotic wheelchair system: Indoor navigation and user interface," in *Lecture Notes Artificial Intelligence: Assistive Technology Artificial Intelligence*, V. O. Mittal, H. A. Yanco, J. Aronis, and R. Simpson, Eds. New York: Springer-Verlag, 1998, pp. 256–268.
[52] M. A. Goodrich, D. R. Olsen, J. W. Crandall, and T. J. Palmer, "Experiments in adjustable autonomy," in *Proc. IJCAI-Workshop Autonomy, Delegation, Control: Interaction Autonomous Agents*, 2001, pp. 1624–1629.
[53] S. Lee, G. J. Kim, G. S. Sukhatme, and C.-M. Park, "Effects of haptic feedback on telepresence and navigational performance," in *Proc. Int. Conf. Artificial Reality Telexistence*, 2004, pp. 1–8.
[54] D. Johansson and L. de Vin, "Omnidirectional robotic telepresence through augmented virtuality for increased situation awareness in hazardous environments," in *Proc. IEEE Int. Conf. Systems, Man Cybernetics*, Oct. 2009, pp. 6–11.
[55] H. A. Yanco and J. Drury, "Classifying human-robot interaction: An updated taxonomy," in *Proc. 2004 IEEE Int. Conf. Systems, Man Cybernetics*, Oct., vol. 3, pp. 2841–2846.
[56] S. Ellis, *Pictorial Communication in Virtual and Real Environments*. Washington, DC: Taylor Francis, 1993.
[57] M. R. Endsley, *Direct Measurement of Situation Awareness: Validity and Use of SAGAT*, 1st ed. Mahwah, NJ: Lawrence Erlbaum Associates, 2000, ch. 7, pp. 147–174.
[58] R. Taylor, "Situational awareness rating technique(sart): The development of a tool for aircrew systems design," in *Proc. AGARD, Situational Awareness Aerospace Operations*, 1990, vol. 17, pp. 23–53.
[59] T. B. Sheridan, "Human supervisory control of robot systems," in *Proc. 1986 IEEE Int. Conf. Robotics Automation*, Apr., vol. 3, pp. 808–812.
[60] T. Fong, C. Kunz, L. M. Hiatt, and M. Bugajska, "The human-robot interaction operating system," in *Proc. 1st ACM SIGCHI/SIGART Conf. Human-Robot Interaction*, 2006, pp. 41–48.
[61] V. Groom and C. Nass, "Can robots be teammates: Benchmarks in humanrobot teams," *Interact. Stud.*, vol. 8, no. 18, pp. 483–500, Oct. 2007.
[62] J. M. Bradshaw, P. J. Feltovich, M. Johnson, M. R. Breedy, L. Bunch, T. C. Eskridge, H. Jung, J. Lott, A. Uszok, and J. van Diggelen, "From tools to teammates: Joint activity in human-agent-robot teams." in *HCI 10* (Lecture Notes Computer Science), vol. 5619, M. Kurosu, Ed. New York: Springer-Verlag, 2009, pp. 935–944.
[63] M. T. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Trans. Syst., Man Cybern.,* vol. 11, no. 6, pp. 418–432, June 1981.
[64] M. K. O'Malley, A. Gupta, M. Gen, and Y. Li, "Shared control in haptic systems for performance enhancement and training," *J. Dynamic Syst., Meas., Control*, vol. 128, no. 1, pp. 75–85, 2006.
[65] T. B. Sheridan, "Teleoperation, telerobotics and telepresence: A progress report," *Control Eng. Pract.*, vol. 3, no. 2, pp. 205–214, Feb. 1995.
[66] B. Ricks, C. W. Nielsen, and M. A. Goodrich, "Ecological displays for robot interaction: A new perspective," in *Proc. 2004 IEEE/RSJ Int. Conf. Intelligent Robots Systems,* Sept./Oct. 2004, vol. 3, pp. 2855–2860.
[67] J. L. Marble, D. J. Bruemmer, D. A. Few, and D. D. Dudenhoeffer, "Evaluation of supervisory vs. peer-peer interaction with human-robot teams," in *Proc. 37th Annu. Int. Conf. System Sciences*, Jan. 2004, pp. 1–4.
[68] G. C. Burdea, "Invited review: The synergy between virtual reality and robotics," *IEEE Trans. Robot. Autom.*, vol. 15, no. 3, pp. 400–410, June 1999.
[69] A. Monferrer and D. Bonyuet, "Cooperative robot teleoperation through virtual reality interfaces," in *Proc. 6th Int. Conf. Information Visualisation*, 2002, pp. 243–248.
[70] L. A. Nguyen, M. Bualat, L. J. Edwards, L. Fl¨uckiger, C. Neveu, K. Schwehr, M. D. Wagner, and E. Zbinden, "Virtual reality interfaces for visualization and control of remote vehicles," *Auton. Robot.*, vol. 11, no. 1, pp. 59–68, 2001.
[71] J. L. Marble, D. J. Bruemmer, and D. A. Few, "Lessons learned from usability tests with a collaborative cognitive workspace for human-robot teams," in *Proc. 2003 IEEE Int. Conf. Systems, Man Cybernetics*, Oct. 2003, vol. 1, pp. 448–453.
[72] E. Rogers, R. R. Murphy, A. Stewart, and N. Warsi, "Cooperative assistance for remote robot supervision," in *Proc. 21st Century IEEE Int. Conf. Systems, Man Cybernetics, Intelligent Systems*, Oct. 1995, vol. 5, pp. 4581–4586

Kevin S. Pratt, Texas A&M, College Station, Texas, USA. E-mail: kpratt@gmail.com.

Robin R. Murphy, Texas A&M, College Station, Texas, USA. E-mail: murphy@cse.tamu.edu.