Cooperative Assistance for Remote Robot Supervision

Robin R. Murphy and Erika Rogers †

School of Mathematical and Computer Sciences

Colorado School of Mines

Golden, CO 80401-1887

rmurphy@mines.colorado.edu

†Dept. of Computer and Information Science

Clark Atlanta University

223 James P. Brawley Dr., S.W.

Atlanta, GA 30314

erika@pravda.gatech.edu

Abstract

This paper describes current work on a cooperative tele-assistance system for semi-autonomous control of mobile robots. This system combines a robot architecture for limited autonomous perceptual and motor control with a knowledge-based operator assistant which provides strategic selection and enhancement of relevant data. It extends recent developments in artificial intelligence in modeling the role of visual interactions in problem solving for application to an interface permitting the human and remote to cooperate in cognitively demanding tasks such as recovering from execution failures, mission planning, and learning. The design of the system is presented, together with a number of exception-handling scenarios that were constructed as a result of experiments with actual sensor data collected from two mobile robots.

Introduction

The study of vision and motion in both man and machines is of particular importance in the arena of remote robot operations. In such cases, the robot must "see" and "move" to

perform tasks in environments where it is deemed too costly or too dangerous for actual human presence. However, since the current state of technology has not yet produced a fully autonomous robot which can be sent on such missions, there is still a strong need for human intervention. The interaction between human and robot is managed in a variety of ways collectively referred to as *telesystems*. Telesystems have long been recognized as a key technology for space exploration [3, 12, 13, 14, 24, 25, 29], and they are becoming increasingly integral to a variety of terrestrial applications including the decontamination and decommissioning of nuclear processing plants [30], rescue, fire-fighting, intervention operations in hazardous environments [9, 10, 18, 28], and security [8]. Unfortunately telesystems, in general, have three drawbacks. First, most systems require a prohibitively high communication bandwidth in order for the human to perceive the environment and make corrections in the remote's action quickly enough [11]. Even with adequate communication bandwidth, the operator may experience cognitive fatigue due to the repetitive nature of many tasks, poor displays [27], and the demands of too much data and too many simultaneous activities to monitor [7]. Furthermore, telesystems are inefficient in that the operator generally handles only one robot and that interaction leads to reduction of work efficiency by factors of five to eight [20]. As robots use more sensors, the amount of data to be processed by the operator will increase, exacerbating the communication and fatigue problems and leading to less efficiency.

The addition of artificial intelligence at the remote is one solution to these shortcomings. Indeed, the intelligence involved in the operation of a mobile robot can be viewed as encompassing a continuous spectrum from master-slave teleoperation through full autonomy [10, 14]. The question that remains is how to add intelligence so as to move the telesystem forward on the spectrum. The standard evolutionary path has been to organize some aspect of human intelligence into a module that can run unaided on the remote after being initiated by the operator. An alternative approach to compartmentalizing intelligence at either the local or the remote is to *distribute* levels of intelligence between them. The teleSFX architecture [15] is one example of a distribution of intelligence for telesystems. It was designed to support intervention and recovery in the case of execution failures (e.g., sensor malfunctions, faulty plans). Intervention and recovery typically requires problem solving abilities which, along with mission planning, have been resistant to automation [4]. In teleSFX the problem solving activity of identifying the cause of the execution failure and determining the appropriate response may be shared by the remote and the human. The remote attempts to first classify and recover from an execution failure using local knowledge. If the remote is unable to classify or construct a proper response, it alerts the operator and posts the results of its unsuccessful attempt. This is information that the operator can use in conjunction with his/her own expertise in solving the problem.

While distributed systems allow the introduction of more intelligence at the remote, they introduce a new concern: how will the disparate intelligences cooperate? More specifically, how can the perceptual and problem solving capabilities of each intelligent entity be ex-

ploited to solve the task at hand as effectively as possible? One approach to this problem is to introduce an *intelligent assistant*, which contains knowledge not only about the computational side of the system, but also about models of human visual problem solving [21]. This approach has been used to assist radiologists in medical diagnosis, by selectively focusing attention on relevant aspects of the image, automatically enhancing the image according to the current needs of the problem solving activity, and interactively assisting the decision-making process by managing hyptheses [22]. In the case of telesystems, the intelligent assistant works closely with the human supervisor to cooperate and coordinate activities with the remote semi-autonomous robot.

The purpose of this paper is to describe this particular concept of cooperative assistance for telesystems, and to propose a basic cooperative assistance architecture which permits the human and remote to interact and recover from execution failures. Successful cooperative assistance is expected to have the following advantages: 1) to improve both the speed and quality of the supervisor's problem solving performance; 2) to reduce cognitive fatigue by managing the presentation of information; 3) to maintain low communication bandwidths associated with semi-autonomous control by requesting only the relevant sensory data from the remote robot; and 4) to improve efficiency by reducing the need for direct operation so that a supervisor could control multiple robots simultaneously. Furthermore, the approach is highly modular and adaptive, supporting the incremental evolution of telesystems to full autonomy. The architecture is general and can be applied to telesystems in space and on earth.

The paper begins with a review of telesystems, showing the need for cooperative assistance and the related efforts in achieving it. The overall approach of cooperative assistance for telesystems is discussed next, followed by details of the architecture. The description of the architecture focuses on how the system supports the recovery from execution failures; however, it should be emphasized that intelligent assistance is appropriate for other tasks such as mission planning and learning. The feasibility of the architecture as a working system is demonstrated through a number of proof of principle experimental scenarios. Current work in implementing the architecture and refining the role of cooperative assistance is summarized.

Background and Related Work

The need for varying levels of human involvement in the robot's operations has resulted in a number of different approaches to the interaction between humans and machines. Lumia and Albus [14] talk about the continuous spectrum of activities between teleoperation and autonomy, while Giralt et al divide this spectrum into four different operational modes [10]:

1. *Teleoperation* or continuous supervision. In this case, the human is directly in control at servo-process level in a master-slave mode.

2. *Advanced teleoperation* (telepresence, telerobotics, teleprogramming, semi-autonomous robotics). Here the approaches range from a human in the programming loop but not at the servo-process level, to a global, virtual sensory reflexive teleprogramming architecture.

3. *Autonomous but purely reactive.* The robots in these systems operate according to modalities specified at design stage. In this case, there is no operational control.

4. *Autonomous task level programmable robots.* In these systems, the operator has task control over the machine which interprets the program and executes it autonomously according to its perception of the context.

Teleoperation

Traditional teleoperations systems place the operator in a direct control loop with the remote robot, thereby welding the natural intelligence and perceptual abilities of the operator to the robot hardware. As robots have been placed in more demanding situations and with an ever increasing array of sensors, this direct coupling has proven unsatisfactory. Many tasks are boring and repetitive, leading to cognitive fatigue in the operator and subsequent poor performance [9]. Master-slave control also may increase cognitive fatigue by forcing the operator to think entirely in terms of a single robot coordinate frame of reference, instead of using task-specific frames (e.g., object-centric). The amount of data that has to be transferred is high and for master-slave control to be successful, it must be updated frequently. This places practical limits on the sensors and control that can be accommodated by the communication link. Finally, master-slave control precludes the operator supervising multiple robots at one time.

Advanced Teleoperation

The second operational mode of Giralt's taxonomy represents two distinct categories of efforts in addressing the drawbacks of teleoperation: telepresence and semi-autonomous control. In telepresence, the aim is to improve human control and lessen cognitive fatigue by providing sensor feedback to the point that the operator feels present in the remote robot's environment. This is accomplished by projecting the human operator into the work space, so that the human may assume the robot's presence in order to perform the task. This means a) seeing through the robot's "eyes" by using strategically placed cameras, as well as three-dimensional modeling techniques, and b) moving the robot itself and/or its appendages by manipulating effectors through tactile feedback mechanisms. Telepresence gives the operator a more natural egocentric representation of the environment, allowing the task to be represented in any coordinate frame. Unfortunately, telepresence is still primarily a master-slave paradigm and suffers from the same problems. Continuous telepresent supervision may

be impractical for applications where the communication bandwidth acts as a bottleneck, and/or transmission time delays make high dexterity control difficult or impossible [11]. Further difficulties include the introduction of new types of operator fatigue and cognitive overload due to increased environmental complexity or data displayed from non-intuitive sensing modalities such as SAR.

Semi-autonomous control systems advance teleoperation by increasing the intelligence on the part of the robot, thereby reducing the amount of supervision by the operator. The human is involved in some aspects of the remote's operation such as task specification, but routine or "safe" portions of accomplishing the task are handled autonomously. Motivation for semi-autonomous control is described as follows:

> In the solutions that stem from the initial teleoperation concept, both human intelligence and "machine intelligence" are concentrated and cooperate at the operator station level. Another different stream of solutions have been proposed that stem from the autonomous robot concept, i.e., the solutions that are somewhat the offspring of the old-time SHAKEY project. Here the basic objective is to have on-board, in-built intelligence at machine level so that it adapts its actions autonomously to the task conditions...Hence we contend that these requirements point to a functional architecture that should provide machine intelligence aids to the operator at programming level as well as the on-board machine attributes of an autonomous intelligent robot [10].

Semi-autonomous control efforts can be loosely classified as advocating either *shared*, *traded*, or *supervisory* control schemes. In shared control [12], the human initiates the actions the remote robot will use to accomplish the task, monitors the its progress, interacts with the robot by adding perceptual inputs, and interrupts execution as needed. The operator provides low-frequency supervision, essentially periodically "looking over the shoulder" of the remote and adjusting its behavior. Shared control frees the operator's attention from directly controlling nominal activities while allowing direct control during more perceptually intensive activities such as direct manipulation of parts [13]. It also provides the possibility of the remote learning new behaviors by observing the operator's mapping of non-nominal sensory patterns into appropriate control commands [12].

In traded control, the remote and local systems exchange control of the robot based on the demands of the task and the constraints of the environment [1, 12, 19]. In systems such as [19], the remote operates autonomously for tasks in which its performance is known to be better and under direct supervision if the remote fails. However, traded control can go the other way as seen in the KRITIC architecture [2], where the remote may override an instruction from the operator (turn left) if deemed necessary (there is an obstacle on the left). Traded control schemes are particularly advantageous for operating robots in unknown environments. There, the time-delay may cause the operator to issue a directive

whose consequences cannot be perceived fast enough to be revoked. Which aspects of the control to trade and when are still open issues.

Supervisory control relies on the operator to initiate and terminate the action and to respond to emergency situations. As a result, it gives quicker task completion by avoiding commmunication time delays and eliminates the need for continuous attention [26]. Supervisory control may be the most likely control scheme to allow an operator to handle more than one remote robot at a time. However, as noted in [5], a major limitation with either supervisory or fully autonomous control is the development of self-monitoring routines for the robot so that it can alert the operator that an anomalous situation has arisen.

Tele-assistance

Semi-autonomous control schemes increase the artificial intelligence residing at the remote in order to reduce both the amount of communication between local and remote, and the demands on the operator. However, there is still a need for human problem solving capabilities, particularly to configure the remote for new tasks and to respond to unanticipated situations. In order to support the interaction between the different intelligent capabilities at the remote and local, the teleoperations community is becoming increasingly interested in computerized assistance for telesystems (tele-assistance), both for the effective filtering and display of pertinent information or data, and also for the decision-making task itself.

Three systems, in addition to the one presented in this paper, have dealt with some of the issues pertaining to tele-assistance. O'Connor and Bohling [17] address the need for an interactive interface which provides an image analyst with supplemental contextual information. This system does not use any underlying intelligence, but does illustrate the advantages of moving away from the limitations of conventional image processing and traditional interfaces.

Coiffet and Gravez [6] describe a cooperative system in which strategic assistance to the operator should consist of "the selection and processing of relevant data (sensor outputs and execution reports), and the filtering of operator commands." This approach to cooperation results in a dialogue between the human and the robot which involves task-oriented diagnosis, and the proposal of pertinent solutions.

Edwards et al [7] have developed a "manager's associate" interface for mission planning, mission management, and vehicle teleoperation and survey activities on a mobile robot. The associate system uses models of the task and the user to provide advanced user support, including workload management, error recognition and correction, display management, and selective task automation. The primary differences between the manager associate and the cooperative assistance approach taken in this paper is the use of broader artificial intelligence methods such as visual interaction models for problem solving rather than rule-based models, and the encapsulation of the assistant as a separate computational agent.

Approach

The goal of the research described in this paper is a cooperative tele-assistance architecture. Our approach treats the remote and human as computational agents possessing unique knowledge and intelligence. It relies on a third computational agent called the intelligent assistant to act as an intermediary between the human and the robot. This agent resides on the local system; it doesn't move and it doesn't perceive. Rather, it supports the perception and problem solving capabilities of the human and the robot by selectively filtering and enhancing perceptual data obtained from the robot, as well as generating hypotheses about execution failures which cannot be solved by the remote.

The intelligent assistant uses a blackboard architecture to observe and manage the information posted independently by the remote and human intelligences. Blackboards have been previously used successfully for teleoperation by Edwards et al [7] in the Ground Vehicle Manager's Associate project and by Pang and Shen [18] for the high level programming and control of mobile robots to assist the operation of the emergency response team involved in a hazardous material spill. In our application of the blackboard, the robot, the supervisor, and the assistant are considered independent intelligent agents as shown in Figure 1.

<div align="center">

****

Figure 1 about here

****

</div>

Each of the computationals agents has internal routines called *knowledge sources* which read and post information to a global, asynchrononous data structure called the *blackboard*. The knowledge sources at the remote post their information about the status of the robot. The human supervisor, by definition a knowledge source, communicates with the intelligent assistant and the remote robot via a graphical interface managed by the assistant. The interface supports learning new configurations and associates responses to extraordinary events.

A description of the basic operation of the intelligent assistant is given in the following example. If the remote detects an anomalous situation that it cannot fix itself, it posts the nature of the alert and what progress it has made in classification and/or recovery. The intelligent assistant whose knowledge sources monitor the blackboard is alerted by this posting. The intelligent assistant responds to the alert by attempting to assess the nature of the problem, and then uses the principles of visual interaction in problem solving in conjunction with task-dependent models to determine what information, sensor data, and associated levels of enhancement to display to the supervisor. The supervisor, in turn, interprets the display, posts hypotheses and may request additional information and sensor data from the remote. The intelligent assistant manages the hypotheses, reminds the supervisor of appropriate diagnostic procedures, requests sensor data from the remote, and then enhances

<div align="center">

7

</div>

it to highlight the attributes needed to confirm the current hypothesis. The assistant also coordinates the display of relevant contextual information such as terrain or cartographic data, imagery-related data (weather conditions, etc.), and general information (intended task, etc.); the importance of ancillary information was established in [17].

Cooperative Tele-Assistance Architecture

To achieve this goal of cooperative tele-assistance, two major software systems have been joined together within the blackboard paradigm and modified appropriately for this application domain. The first is the Sensor Fusion Effects (SFX) architecture [16], which utilizes state-based sensor fusion to support the motor behavior of a fully autonomous robot. If an execution failure is detected, fusion is suspended and control is passed to an exception-handling mechanism. The exception handler attempts to identify the problem and either repair or replace the sensing configuration and its associated state. The SFX architecture is the basis for the remote roboticagent. The second system, called VIA (Visual Interaction Assistant), is designed to cooperatively assist human perception and problem solving in a diagnostic visual reasoning task [23]. It is the foundation for the intelligent assistant agent and controls the interface to the human agent. VIA is a blackboard-style system which utilizes knowledge-based techniques to focus the user's attention on relevant parts of the image, automatically enhancing the image according to the needs of the user's problem solving process. It further manages diagnostic hypotheses, maintaining beliefs according to current evidence, and assists the user to converge opportunistically on a solution where possible. This system was originally developed in the domain of diagnostic radiology, and a small prototype system was built which demonstrated some of these capabilities, and received favorable response when tested with a number of radiology residents [22].

One argument for combining the SFX and VIA systems is their common emphasis on perception: SFX concentrates on robotic perception, while VIA works with human perception. The intelligent agent allows a computational medium for symbolic communication of what the robot perceives and what the human interprets. These two systems have been adapted to work together in the context of tele-assistance, and the modified systems have been named teleSFX and teleVIA, respectively. A practical advantage of linking these two systems is that under teleSFX, the robot has already attempted a certain amount of trouble-shooting itself. Thus information about what has been tried, the robot's own conclusions, and the relevant sensor images can all contribute to the decision-making process of the local supervisor. In order to achieve this, the teleSFX system includes an interactive exception handling component, which allows the robot to call for help in the event that its own exception handling capabilities could not resolve the problem. An overview of the entire system is shown in Figure 2, and further details are provided in the following subsections.

In this diagram, it can be seen how the interactive configuration and interactive exception handling components of the teleSFX architecture are merged with the intelligent assistance provided by teleVIA, through the panels of the blackboard. The emphasis in this paper is on the interactive exception handling aspects of this design.

TeleSFX

The remote agent is implemented following the teleSFX architecture. In [15], the teleSFX control scheme was introduced, emphasizing the intelligent exception handling mechanism at the remote. Unlike configuration, exception handling must be done in real-time (for example, a robot may be moving when a sensor malfunctions). As shown in [5], autonomous exception handling is difficult because it involves domain and hardware specific information which may not always be available or correct.

TeleSFX uses a three part strategy for exception handling: *detection, classification, and recovery.* The first step, detection, determines that a "sensing failure" has occurred. Sensing failures are any anomalous or suspect conditions that have been previously defined by the knowledge engineer. Sensor malfunctions are one type of failure. Many sensor malfunctions manifest themselves via explicit hardware errors communicated to the controlling process (e.g., bus errors, frame grabber errors) and tend to be straightforward to classify and recover from (e.g., reset the system, request a retry). Another class of sensing failures is due to unanticipated changes in the sensing environment which degrade the performance of one or more sensors (e.g., the lights are turned off, high concentration of dust). The third and final class of failures stem from errant expectations, where the robot is interpreting the observations according to a model. If for some reason the robot has selected the wrong model at the wrong time (e.g., for mechanical reasons, the robot did not rotate fully to the intended viewpoint), the sensor observations are unlikely to agree.

Failures in the latter two classes are difficult to detect because the sensors are operating "correctly" but their data can no longer be interpreted without accounting for the changed context. Therefore teleSFX is sensitive to inconsistencies in the evidence contributed by different sensors for a particular task. The knowledge engineer defines a set of failure conditions representing these inconsistencies for the particular implementation. Each perceptual process may have a different set of thresholds for those failure conditions, given the unique interactions between sensors.

In the classification step, the remote robot attempts to autonomously identify a sensing failure, and adapt the sensing configuration. This involves hypothesis generation, testing and response heuristics at the remote site, and several experiments have been described in

[5] which demonstrate this capability. However, the success of the classification step depends on the expert understanding of the domain and the sensors. This domain-dependence means that classification by the robot is brittle and will not always be successful. Therefore, if the remote system cannot resolve the difficulty, teleSFX must post the request for help to the blackboard, together with immediately relevant data such as current sensor data and a log of the remote's hypothesis analysis. This signals the teleVIA system to activate its knowledge sources in order to request and present further data, as well as to perform further hypothesis analysis.

Figure 3 shows the details of the control system for the remote site.

****
Figure 3 about here
****

The local supervisor is involved primarily in interactive configuration, and general monitoring, until the interactive exception handling is triggered by the remote system. At that point, teleVIA takes over from teleSFX until the repair is communicated.

TeleVIA

In Figure 4 are shown the components of the cooperative system which assists the human supervisory activities at the local site.

****
Figure 4 about here
****

TeleVIA consists of the blackboard data structure, which is organized into five major panels, together with four main control modules: Hypothesis Manager, Strategy Selector, Attention Director and User Interface. These modules interact with a knowledge base which serves as the repository of long-term information in the system.

TeleVIA Blackboard

The blackboard is the heart of the cooperative intelligent assistant. It is where the evolutionary results of the problem solving effort are captured. The logical partitioning of the blackboard is based on components of a cognitive model of visual interaction described in [21], and illustrated in Figure 5.

```
**********************************
              Figure 5 about here
**********************************
```

This organization was designed to facilitate transfer of information between human perception and problem solving during a visual reasoning task. In the domain of tele-assistance, it is seen that, with one exception, the same logical partitions or panels may be used. The additional information which is contributed by the remote robotic system is accommodated in the subpanels as shown in Figure 6.

```
**********************************
              Figure 6 about here
**********************************
```

Current Context Panel

In the general VIA design, this area contains information that is known about the overall problem context. In the teleVIA mode, the Current Context Panel is used to monitor the robot's (or robots') current activities. It is active at all times during the mission, and contains information about the task underway, the known environmental factors and conditions, which sensors are active and working, and intermittent video images from the robot reinforcing the operator's knowledge of the context within which the robot is currently functioning. This panel is visible when the system is in "monitoring" mode, as well as in "failure" mode.

Interactive Exception Handling

This panel reflects the state of the robot's perception. In particular, when a sensor fusion failure occurs and cannot be resolved by the robot, the signal for help is sent here, together with the type of failure, currently active sensors, and the belief table for those sensors. This tells the local supervisor what the perceptual status of the robot is at the time of failure, and provides initial information for teleVIA to begin formulating hypotheses, and requesting further information.

Interactive Configuration

This additional panel will allow the local operator to select appropriate sensors, and to communicate sensing and backup plans to the robot. It is provided to permit direct human-robot communication, and has no counterpart in the original cognitive model.

<u>Hypothesis Panel</u>

This panel contains the current hypotheses that constitute the partial (or complete) solutions that are evolving as a result of the problem solving activity. It is divided into two subpanels:

1. The *Robot Hypotheses* area contains the hypotheses generated by the teleSFX system at the remote site, and reflects the diagnostic and problem solving activities carried out autonomously by the exception handling mechanism of the robot. These must be communicated to the teleVIA system in the event of a failure, so that teleVIA can take advantage of what the robot itself has already tried.

2. The subpanel containing *TeleVIA Hypotheses* consists of hypotheses generated by the knowledge sources of teleVIA, based on the information posted by the remote system in combination with more extensive knowledge retrieved from the teleVIA knowledge base.

<u>Attention Panel</u>

This panel is the locus of the visual focus-of-attention mechanism. It is also partitioned into two parts:

1. *Attention Directives* are issued by the teleVIA system in order to assist the local supervisor's perception of relevant data. To accomplish this, teleVIA may request particular images to be transmitted by the robot. In this way, delays due to transmission of unnecessary and/or extraneous data may be avoided. Furthermore, since the images are selected by teleVIA's knowledge sources according to the current problem, they are more likely to be pertinent and useful. The directives issued to the human supervisor are then aimed at guiding him/her to look at particular aspects of the data provided by the remote system.

2. The second area of the Attention Panel consists of one or more images, obtained from the robot by the teleVIA system. Depending on the sensory modality of the displayed images and/or data (e.g., video vs. infra-red vs. ultrasonics), teleVIA will also automatically execute appropriate image enhancements designed to facilitate the supervisor's perception of the feature(s) in question. In this manner, the superior perceptual capabilities of the human can be exploited in order to diagnose the problem more quickly.

TeleVIA Control

The four control modules of teleVIA are also based on aspects of the cognitive model of visual interaction referenced previously in [21]. The Hypothesis Manager is primarily concerned

with the problem solving aspect of the task, while the Attention Director deals with the perceptual side. The Strategy Selector allows the program to decided which of these aspects to consider next, and also which approach to take. The graphical user interface implements the human-machine communication mechanism between teleVIA and the human supervisor.

The Strategy Selector is used to pass control from the Hypothesis Manager to the Attention Director, since the way in which attention is focused may depend on the strategy used for reducing the list of active hypotheses. The Attention Director is concerned with focusing attention by presenting and enhancing images as well as suggestions to the operator of what to look at next. The User Interface is the component through which the human operator communicates with the teleVIA system.

Hypothesis Manager

The Hypothesis Manager impacts the blackboard through the activities of hypothesis-related knowledge sources. Each knowledge source has a set of preconditions that must be satisfied by information at a particular level of the blackboard. It then performs a transformation of the information at one or more levels. Some examples of knowledge sources which are activated by the type of sensor involved in the failure are illustrated in the following tables.

K-S 1
Precondition:
Infra-red is posted on suspect sensor list.
Action(s):
Request latest image from robot.
Post image to raw data slot of EHKS-frame.

K-S 2
Precondition:
Infra-red is suspect and raw data is available.
Action(s):
Run default false-color enhancement.
Display enhanced image.

K-S 3
Precondition: Infra-red is suspect and raw data is available.
Action(s):   Retrieve environmental knowledge from current context.   If knowledge = available then   invoke knowledge-based-false-color enhancement   Display enhanced image.

In this example, the latter two knowledge sources compete with each other, and will be prioritized, depending on the availability of current context information.

Strategy Selector

This module is invoked by the Hypothesis Manager when a knowledge source needs further information before proceeding. It examines the current blackboard configuration in order to determine an appropriate strategy for the next step in the problem solving process. A High Level Plan is then generated to carry out the selected strategy, and is passed to the Attention Director for refinement and execution.

The strategies can be classified as either perceptual strategies or problem solving strategies. Since images must be requested from the robot and sometimes modified before display, the perceptual strategies include a) selection/display and b) enhancement. In a), the system may decide to consider all of the constraints of the sensors' data (transmission time, resolution quality, perceptual information, etc.) together with the current failure information. On the other hand, mission time restrictions may require a strategy of obtaining relevant images at a lower resolution than normal. In all of these cases, the ideal goal is to choose the image(s) which would be most likely to allow the local supervisor to immediately detect the problem by just looking at the image. In b), again time or other constraints might affect the appropriateness or desirability of invoking different types of image enhancemnts.

The problem solving strategies, on the other hand, provide guidance on how to handle the diagnostice hypotheses. The initial strategies we have identified come from the radiology domain, where there is often a very large set of potential candidates for diagnostic hypotheses. These strategies include: a) rule-out - the system looks for evidence which allows it to reduce the number of candidate hypotheses; b) support - the system seeks supporting evidence for the strongest candidate hypotheses of a relatively small set; c) not-enough-information - the system has not been able to generate any hypotheses, and therefore requires more evidence (particularly of a percpetual nature) in order to begin problem solving. Although initial work in the tele-assistance domain has revealed a paucity of failure-related knowledge, it is expected that as the the domain theory grows, these types of strategies will prove to be more effective.

14

The Attention Director module takes the High Level Plan produced by the Strategy Selector, and constructs an Attention Plan that contains detailed instructions for focusing attention. The steps of the Attention Plan are based on the particular type of evidence that is needed to fulfill the mandate of the Strategy Selector. These steps are expanded with image enhancement procedures where appropriate, and are executed. Control is then passed to the operator for feedback. In this way, the system presents information, makes suggestions, and enhances the image(s) in such a way as to influence the direction of the operator's problem solving.

User Interface

The User Interface is divided into two parts: the *Logical User View*, which controls how much of the blackboard is visible to the user, and the *Presentation Manager*, which controls the form of the interface as it is presented to the user. The Logical User View component of the user interface allows the system to be adapted for various purposes without compromising its basic problem solving approach. For example, when the operator is simply monitoring the robot, and performing interactive configuration, the panels involved in exception handling should be hidden from view. There may also be a certain amount of data posted to the blackboard, which is utilized by teleVIA in its hypothesis management, but which should not necessarily be visible to the operator. On the other hand, the Presentation Manager provides the actual human-machine interface of the system through a displayed representation of the Logical User View. This may take a number of forms including menus, icons, graphics, and/or direct manipulation windows, and may even extend to audio as well as visual mechanisms.

Experiments

The experiments described here use data for scene recognition which was collected from two sources. Scenarios 1, 2, 3 and 5 are based on sensor observations collected from the Denning DRV mobile robot, George, at the Georgia Institute of Technology. Scenario 4 is based on sensor data from Clementine, the Colorado School of Mines' Denning MRV-3 mobile robot.

Five different types of sensors (an Inframetrics true infrared camera, a black and white video camera, a Hi8 color camcorder, a UV camera and ultrasonics) provided observations from George. Clementine supplied data sets from three sensors (a black and white video camera, a color camcorder, and ultrasonics). Both robots simulated security guards, where the task was to determine whether a student desk area of a cluttered room had changed since the last visit. In the following scenarios the focus is on the activities of the teleVIA

system in response to the request for help from the remote system.

Scenario 1

In the first experiment, the robot collected data for the desk scene while facing a different part of the room. An example of this is shown in Figure 7 (this figure shows a "before" and "after" image from the black and white camera).

****
Figure 7 about here
****

The effect of this was that while the infrared sensor correctly believed that the scene had not changed, the black and white camera mistakenly thought that the scene had changed. This resulted in a "high conflict" type of state failure during fusion. The robot then generated a hypothesis of sensor malfunction, and attempted to run diagnostics on the two conflicting sensors. These diagnostics, however, showed that both sensors were working correctly. At this point, the robot could not proceed further, and a signal for assistance was simulated.

A request is posted to the interactive exception handling panel of the blackboard, indicating the type of failure it has encountered, and including the beliefs which led to this failed fusion. In this initial version of teleVIA, the images from both of the suspect sensors are requested and displayed for comparison. The possible causes of failure which are known at this time include: wrong input, sensor malfunction, sensor occlusion, sensor hardware error (missing data, self-diagnostic error), multiple sensor errors and electromagnetic interference. In this case, the human can easily detect the mistake (wrong input) by simply comparing the two images. Since the purpose of the system is to provide assistance as quickly as possible, an assumption is made that, if applicable, images which are most easily perceived by humans are given priority, so that the most effective solution to the problem can be supported. Once the supervisor indicates a high belief in a diagnosis, a list of repair possibilities will be posted to the interactive configuration panel for implementation.

Scenario 2

In the second experiment, a transmission failure was simulated, distorting the image. The distortion led the fusion process to declare a "below minimum" type of failure, and, as a result, the exception handling mechanism generated a first hypothesis of "inadequate sensing plan". A backup plan was then implemented, and sensor data was reacquired accordingly. The new plan added a color camera to the sensing suite, and subsequently a fusion failure of "high conflict" was encountered between the black and white camera and the color camera.

As in the previous experiment, teleSFX then generated the hypothesis of sensor malfunction, performed diagnostics which denied this hypothesis, and then called for assistance.

In this case, both of the failures are posted to the blackboard, together with the beliefs generated for each attempt. Once again, the primary troubled sensor is the black and white camera; however, since the second attempt introduced the conflicting image from the color camera, both the black and white and the color video images are displayed by teleVIA for the operator to examine first. In this case as well, the operator should be able to determine the problem fairly quickly by simply comparing the black and white video image with that of the color camera. Note that this is how the system would respond to problems due to external factors such as dirt on the lens, high dust content in the atmosphere, etc.

Scenario 3

In the third experiment, the lights were turned off during data collection to simulate an unforeseen change in environment. In this case the exception handling mechanism of the robot arrived at a correct conclusion of "environmental change" by testing the visible light information. However, for this type of problem, operator assistance is still needed for recovery, and therefore a message is posted to the interactive configuration portion of the blackboard requesting intervention. The beliefs leading to the original state failure, together with the hypotheses generated and tested by the robot, are posted to the blackboard, while images and data from the relevant sensors (black and white camera, and UV sensor) are also displayed. This enables the supervisor to determine what type of environmental change may have occurred.

In each of these experiments, the primary sensor involved in the problem was the black and white camera. Since these experiments were originally designed to test the autonomous exception handling capabilities of the teleSFX system, the results, when extended to the teleVIA component are somewhat artificial. However, they serve the purpose to establish the type of information which must be communicated between the remote and the local systems in even such elementary scenarios. This allows us to determine the types of knowledge sources which may be activated, the different types of hypotheses which may be needed, and how to present this information effectively using the blackboard mechanism.

Further experiments are underway which emphasize sensor data which is not as easily perceived by the human supervisor, and which may require enhancement before conclusions may be drawn. In these cases, teleVIA knowledge sources are activated according to the type of sensor(s) involved in the state failure. This is then combined with knowledge of the current context to select appropriate enhancements, and display the information through the graphical user interface. The following scenarios were constructed using images acquired by the robot for a drill press scene.

Scenario 4

In this example, it is assumed that the ultrasonics are contributing primarily to the fusion failure. In this experiment, one out of the 24 ultrasonics transducers mounted in a ring began to report widely fluctuating readings. A sensing failure of "highly uncertain" evidence was reported, but the responsible sensor could not be isolated, thereby necessitating aid from the local system. The raw ultrasonic readings that come from a Denning mobile robot are just numbers, which represent measurements in feet. However, when this data is represented as a polar plot as in Figure 8, it is much easier to notice if one or more of the sensors is giving erroneous readings.

<div align="center">
****

Figure 8 about here

****
</div>

This is further reinforced if the numerical data are examined in the light of knowledge about the current context, for example, that a room (or mine shaft) is thought to have certain dimensions. A further enhancement of the data which can aid the local operator is an occupancy grid, which presents a bird's eye view of what the robot has sensed so far. The robot builds up this grid or map as it processes ultrasonics data. With both of these types of displays, the operator is more likely to diagnose the failure of an ultrasonic transducer or board, or to detect an erroneous reading.

Scenario 5

When the sensor in question during exception handling is the infra-red camera, enhancements are once again needed to assist the operator's perception of the information in the image. In this case, the untouched true infra-red image is typically gray scale, and there is often not a great deal of discernible contrast in the image. It is common practise to add false color to such an image to show the heat distribution. However, certain choices of false color maps still do not enhance the image, and may obscure the details even further. In the drill-press example, dividing the grayscale into 6 equal bands of color leads to a primarily yellow image, due to the extreme heat of the drill press. However, utilizing model-specific information about the drill press, for example, can result in a more appropriately enhanced image, making it easier for the operator to see the heat profile represented as blue, green, red, yellow, and white bands. A grey scale rendition of the raw image and the two enhancement variations is shown in Figure 9.

<div align="center">
****
</div>

****

Ongoing Work

Current work is concentrating on constructing experiments in real-time where an operator at Clark Atlanta will interact with the remote robot (Clementine) at Colorado School of Mines. An important issue which has not been addressed in this work so far is that of learning. The robot will typically be working in hazardous and/or remote environments about which little may be known, and therefore it is difficult to anticipate the types of problems which may arise. Not only would it be desirable to increase the autonomy of the individual robot wherever possible, but the knowledge gained from solving these problems could be disseminated to other robots in the field. Furthermore, if the teleVIA system could "remember" certain interactions, these could immediately be retrieved from memory, rather than having to generate the same session over again. The technique of case-based reasoning is a natural candidate for this type of learning. Each interactive exception handling session may be captured as a case, which would be indexed on features such as particular configurations of sensors and failure types. Such a case could also include relevant images, or at least image types and enhancements used, so that teleVIA would simply use a case retrieval mechanism rather than a potentially complicated reasoning strategy. Certain aspects of the exception handling and recovery procedures might also then be communicated to the robot itself, to extend its autonomous capabilities, especially for recurrent problems.

Summary and Conclusions

This paper presents an approach to tele-assistance for semi-autonomous mobile robots, which reduces the level of human supervision and provides intelligent assistance for problem solving. The approach partitions problem solving responsibilities between the remote and the local machines. The remote system monitors its sensing for anomalies, called sensing failures, using teleSFX. If a failure occurs, it attempts to classify the source of the problem using a generate and test methodology. If it is successful in identifying the source, it then attempts to autonomously recover (e.g., go to back up sensors, change parameters). Otherwise, if the source cannot be classified, or if no recovery strategy is available, the local machine must provide the exception handling. Exception handling at the local is done by the supervisor, with the help of teleVIA. TeleVIA uses a common blackboard to cooperatively assist the supervisor by posting what has been done by the remote robot, displaying and enhancing sensor data needed in ascertaining the problem, and managing diagnostic hypotheses and beliefs. Experimental scenarios using data collected from mobile robots illustrate the operation of the system.

Cooperative assistance is expected to improve both the speed and quality of the supervisor's problem solving performance by providing an intelligent interface which manages the presentation of data and guides the problem solving process using task models. It is also expected to reduce cognitive fatigue for the same reasons. The assistant maintains a low communication bandwidth by requesting only data which is believed pertinent to the current cognitive task, rather than post all information to the supervisor. The overall work efficiency is likely to increase as the assistant frees the human to supervise multiple remotes. Overall, the approach supports the incremental addition of artificial intelligence as more progress is made in learning and planning.

References

[1] Boissiere, P.T. and Harrigan, R.W., "Telerobotic Operation of Conventional Robot Manipulators", *Proc. 1988 IEEE International Conference on Robotics and Automation*, 1988.

[2] Boissiere, P.T. and Harrigan, R.W., "An Alternative Control Structure for Telerobotics", *Proc. NASA Conference on Space Telerobotics*, Pasadena, CA, 1989.

[3] Brooks, T.L. and Ince, I., "Operator Vision Aids for Telerobotic Assembly and Servicing in Space". *Proc. 1992 IEEE International Conference on Robotics and Automation*, 1992, 886-891.

[4] Causse, O. and Crowley, J.L., "A Man Machine Interface for a Mobile Robot". *Proc. of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1993, pp. 327-335.

[5] Chavez, G. T., and Murphy, R.R., 1993. "Exception Handling for Sensor Fusion" SPIE Sensor Fusion VI, Boston, MA, Sept. 7-10, 1993, pp. 142-153.

[6] Coiffet, P. and Gravez, P., 1991. "Man-Robot Cooperation: Towards an Advanced Teleoperation Mode". In S. G. Tzafestas (Ed.), *Intelligent Robotic Systems*. Marcel Dekker: New York, pp. 593-636.

[7] Edwards, G.R., Burnard, R.H., Bewley, W.L., and Bullock, B.L., "The Ground Vehicle Manager's Associate", *Tech Report AIAA-94-1249-CP*, 1994, pp. 520-526.

[8] Everett, H.R. and Laird, R.T., "Reflexive Teleoperated Control". *Proc. AUVS?*, 1990, no page numbers.

[9] Fogle, R.F., "The Use of Teleoperators in Hostile Environment Applications". *Proc. 1992 IEEE International Conference on Robotics and Automation*, 1992, 61-66.

[10] Giralt, G., Chatila, R. and Alami, R. "Remote Intervention, Robot Autonomy, and Teleprogramming: Generic Concepts and Real-World Application Cases". *Proc. of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1993, pp. 314-320.

[11] Held, R. and Durlach, N., "Telepresence, Time Delay, and Adaptation". (?) pp. 28-1 - 28-16.

[12] Hirzinger, G., "Multisensory shared autonomy and tele-sensor programming - Key issues in space robotics". *Robotics and Autonomous Systems 11*, 1993, pp. 141-162.

[13] Kan, E.P. and Austin, E., "The JPL Telerobot Teleoperation System". *International Journal of Robotics and Automation, Vol. 5*, No. 1, 1990, 27-31.

[14] Lumia, R. and Albus, J.S., "Teleoperation and Autonomy for Space Robotics". *Robotics 4*, 1988, pp. 27-33.

[15] Murphy, R.R., 1993. "Robust Sensor Fusion for Teleoperations", 1993 IEEE International Conference on Robotics and Automation, invited special session on Multisensor Fusion, Atlanta, GA, May 2-6, 1993, vol. 2, pp. 572–577.

[16] Murphy, R.R. 1992. *An Architecture for Intelligent Robotic Sensor Fusion.* Technical Report No. GIT-ICS-92/42, College of Computing, Georgia Institute of Technology, Atlanta GA 30332.

[17] O'Connor, R.P. and Bohling, E.H., "User interface development for semi-automated imagery exploitation". *SPIE Vol. 1472 Image Understanding and the Man-Machine Interface III*, 1991, 26-37.

[18] Pang, G.K.H. and Shen, H.C., "Intelligent Control of an autonomous mobile robot in a hazardous material spill accident - a blackboard structure approach". *Robotics and Autonomous Systems 6*, 1990, pp. 351-365.

[19] Papanikolopoulos, N.P. and Khosla, P.K., "Shared and Traded Telerobotic Visual Control". *Proc. 1992 IEEE International Conference on Robotics and Automation*, 1992, 878-885.

[20] Pin, F.G., Parker, L.E. and DePiero, F.W., "On the design and development of a human-robot synergistic system". *Robotics and Autonomous Systems 10*, 1992, pp. 161-184.

[21] Rogers, E., "A Cognitive Theory of Visual Interaction". In B. Chandrasekaran, J. Glasgow and N.H. Narayanan (Eds.), *Diagrammatic Reasoning: Computational and Cognitive Perspectives.* AAAI/MIT Press: Menlo Park, CA, 1995, 481-500.

[22] Rogers, E., "VIA-RAD: A blackboard-based system for diagnostic radiology". *Artificial Intelligence in Medicine 7*, 1995, 343-360.

[23] Rogers, E., "Cognitive Cooperation Through Visual Interaction". *Knowledge-Based Systems 8*, Nos. 2-3, April-June 1995, 117-125.

[24] Schenker, P.S., "Intelligent Robotics for Space Applications". In S.G. Tzafestas (Ed.), *Intelligent Robotic Systems*. Marcel Dekker Inc.: New York, 1991, 545-589.

[25] Schroer, B.J., "Telerobotics Issues in Space Application". *Robotics 4*, 1988, pp. 233-244.

[26] Sheridan, T.B., "Human Supervisory Control of Robot Systems". *Proc. 1986 IEEE International Conference on Robotics and Automation, Vol. 2*, 1986, 808-812.

[27] Stark, L., Mills, B., Nguyen, A.H. and Ngo, H.X., "Instrumentation and Robotic Image Processing Using Top-Down Model Control". *Proc. of 2nd International Symposium Robotics and Mfg Research, Education and Applications*, 1988, no page numbers.

[28] Stone, H.W. and Edmonds, G., "Hazbot: A Hazardous Materials Emergency Response Mobile Robot". *Proc. 1992 IEEE International Conference on Robotics and Automation*, 1992, 67-73.

[29] Tendick, F., Voichick, J., Tharp, G. and Stark, L., "A Supervisory Telerobotic Control System Using Model-Based Vision Feedback". *Proc. 1991 IEEE International Conference on Robotics and Automation*, 1992, 67-73.

[30] Thayer, S., Gourley, C., Butler, P., Costello, H., Trivedi, M., Chen, C. and Marapane, S., "Three-Dimensional Sensing, Graphics and Interactive Control in a Human-Machine system for Decontamination and Decommissioning Applications". *SPIE Vol. 1828 Sensor Fusion V*, 1992, 74-85.
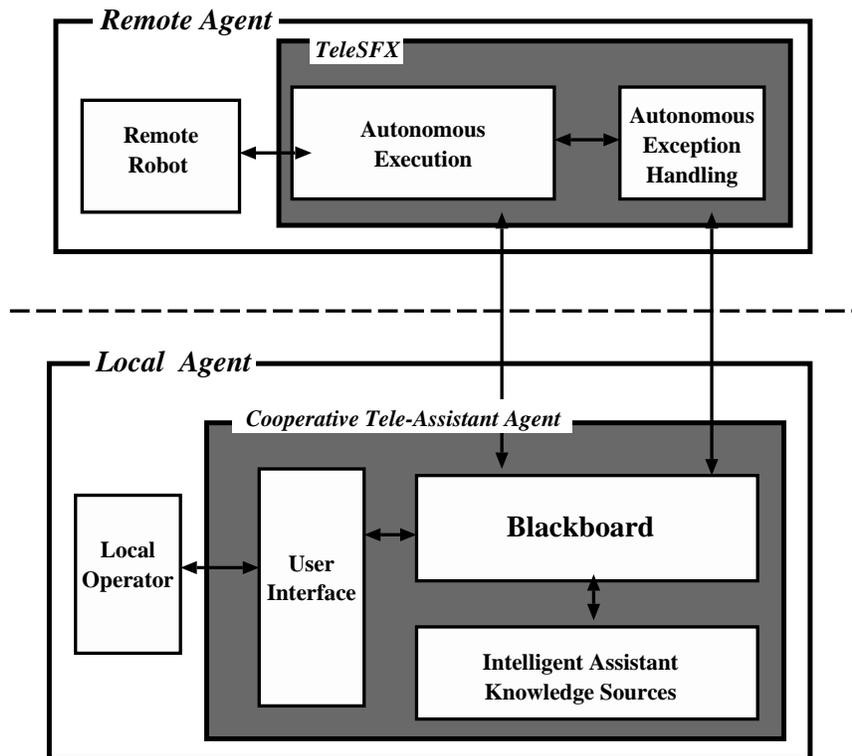
Figure 1: Cooperative Agent Interaction for Tele-Assistance.
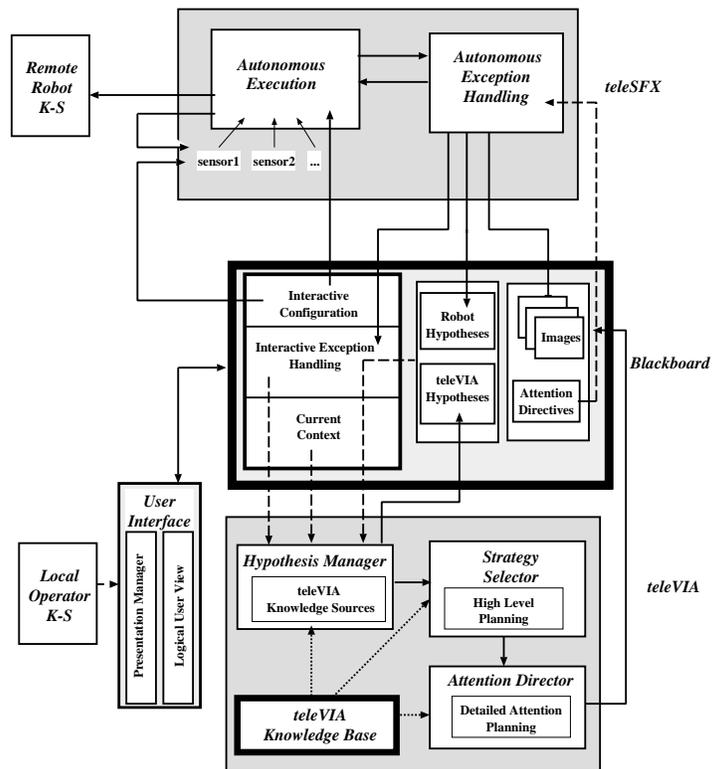
Figure 2: Cooperative Tele-Assistance System Design.
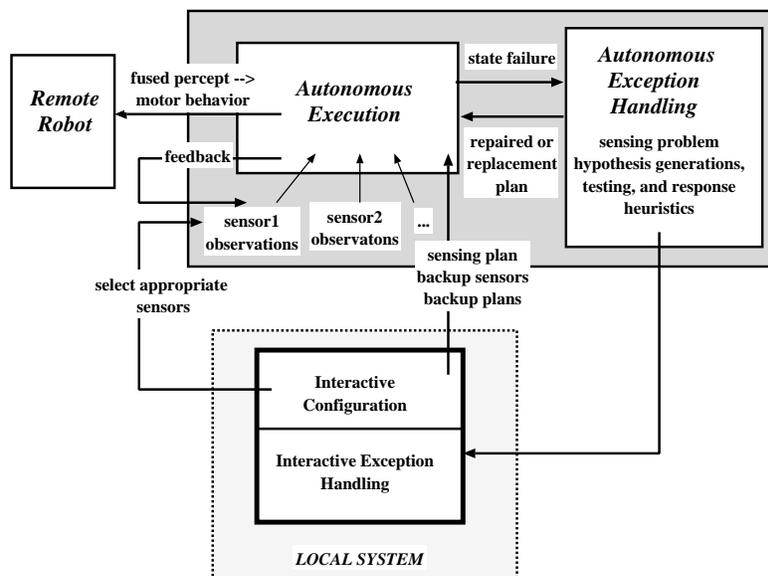
Figure 3: Overview of TeleSFX.
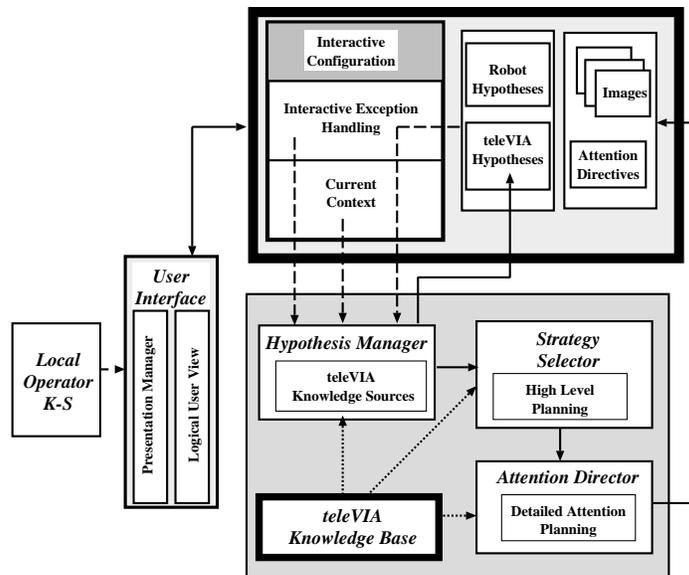
Figure 4: Overview of TeleVIA.

Working Memory
Components of
Cognitive Model

Visual Context   Percepts   Hypothesis Report   High Level Plan / Attention Plan

VIA Blackboard

Context Panel   Percep-tual Panel   Hypothesis Panel (Visual, Reasoning)   Attention Panel (Images, Directives)

VIA-RAD Blackboard

Landmarks Panel   Features Panel   Hypothesis Panel (Findings, Diagnoses)   Attention Panel (Images, Directives)

teleVIA Blackboard

Current Context / Interactive Exception Handling / Interactive Configuration   Hypothesis Panel (Robot, teleVIA)   Attention Panel (Images, Directives)

Figure 5: Blackboard Panel Organization.
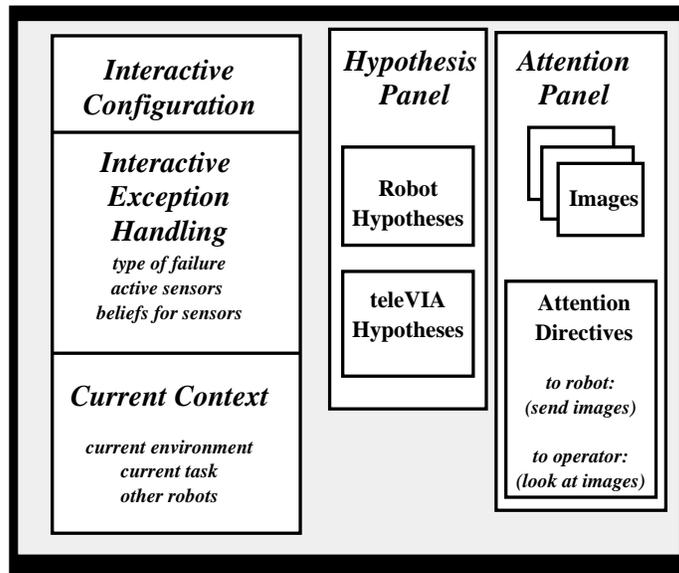
Figure 6: Tele-VIA Blackboard.
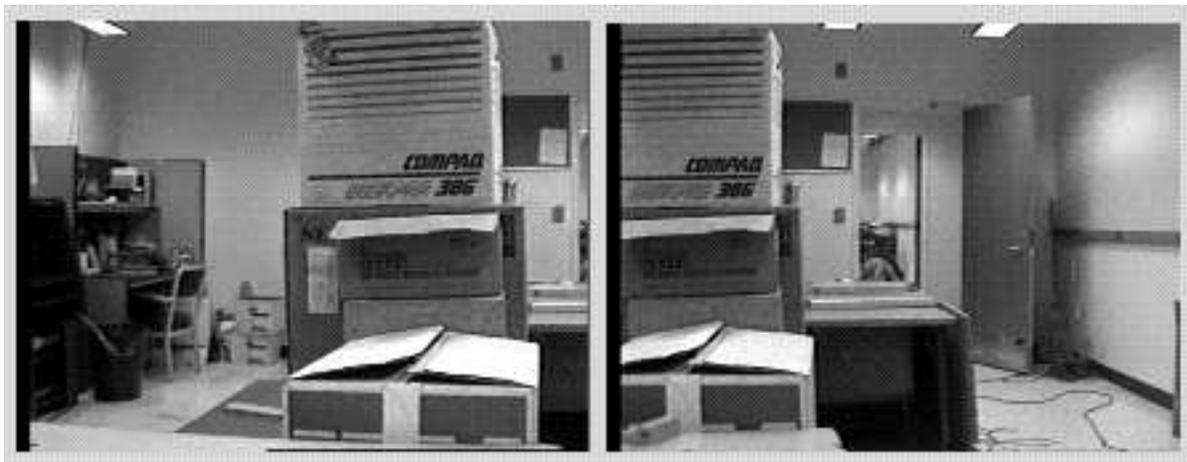
Figure 7: Example for Scenario 1.
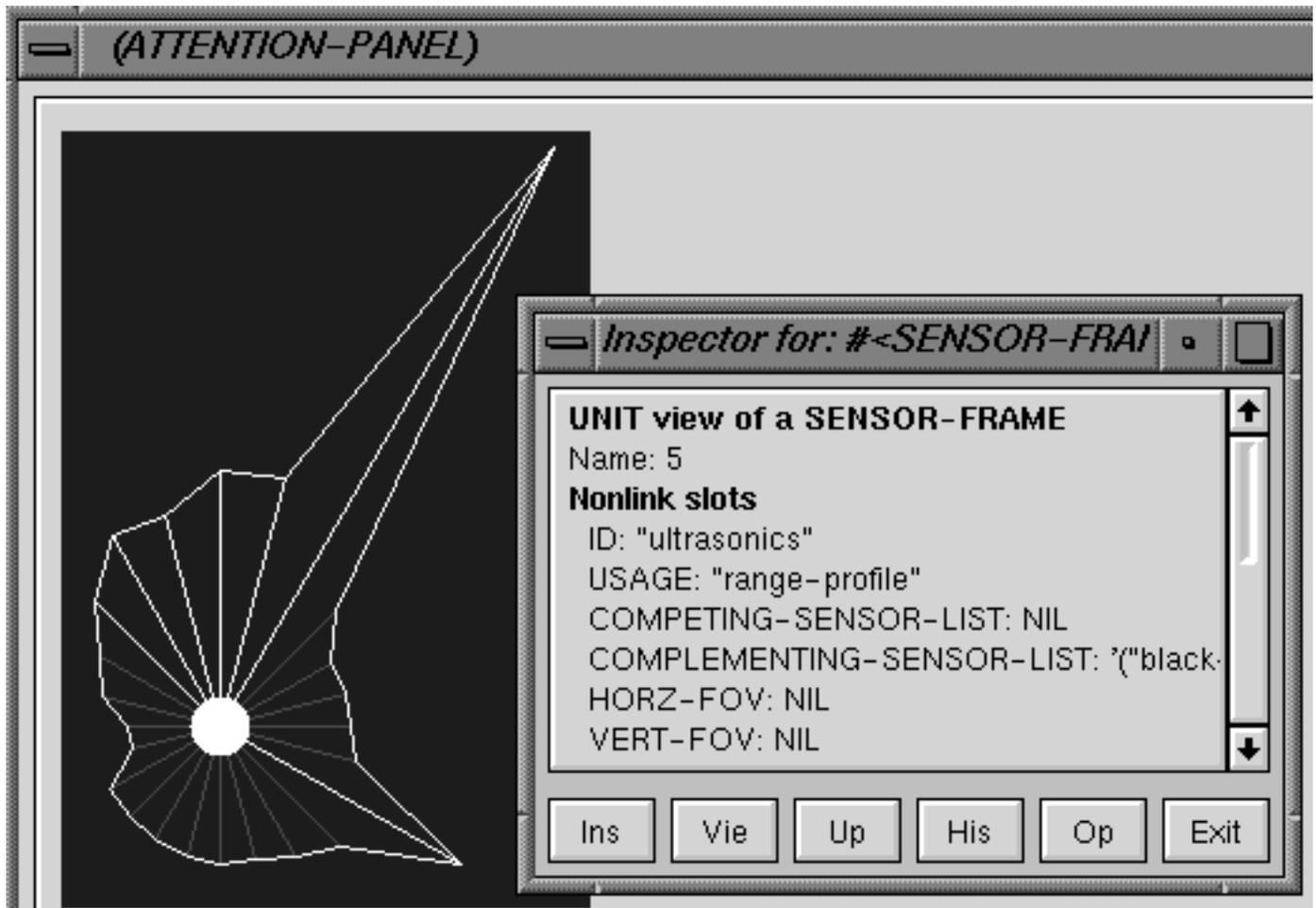
Figure 8: Example Ultrasonics Sensor Frame and Polar Plot.
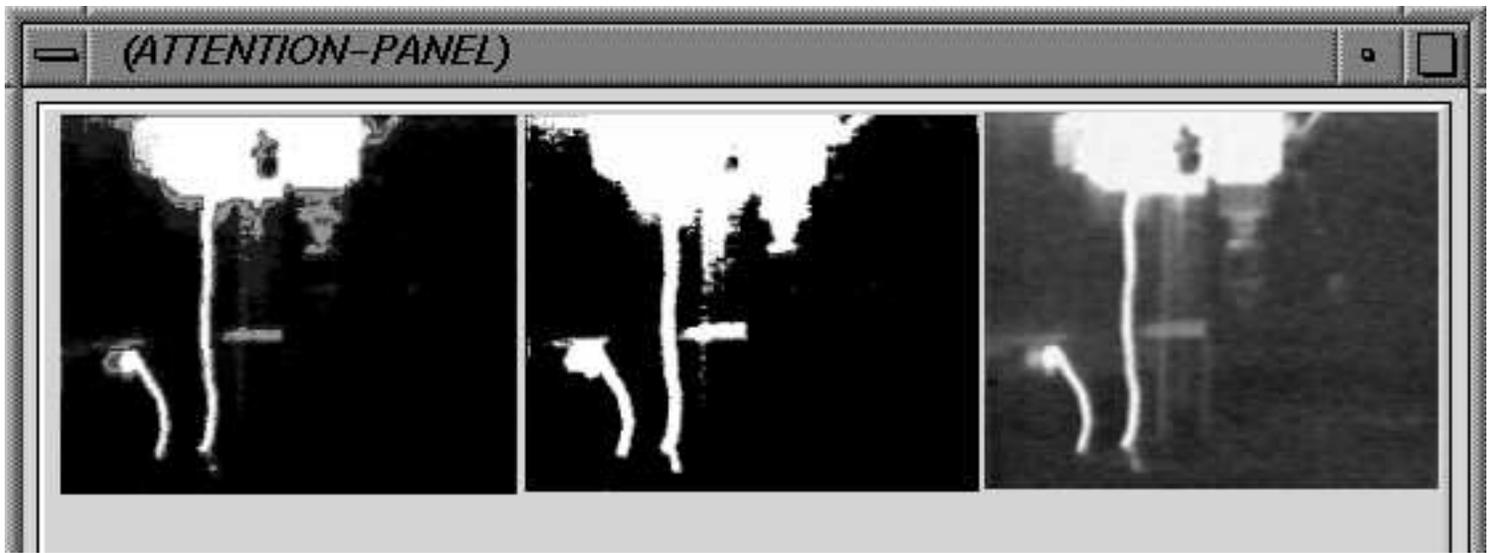
Figure 9: False Color Enhancements and Raw Image.