

Agent-Based Expert Assistance for Visual Problem Solving

Erika Rogers, Robin R. Murphy, and Barb Ericson
Dept. of Computer & Information Science
Clark Atlanta University
Atlanta, GA 30314 USA

Center for Robotics and Intelligent Systems
Colorado School of Mines
Golden, CO 80401-1887 USA

Abstract

This paper presents a domain-independent architecture for facilitating visual problem solving between robots or softbots and humans. The architecture defines virtual and human agents in terms of their inherent cognitive and perceptual abilities, and their weak and strong knowledge sources. It introduces a third agent, an expert assistant called teleVIA, to mediate the flow of information, facilitate cooperation, filter and cache data, and generate the appropriate visual displays. The agent architecture addresses the technical issues of using an expert assistant to coordinate perception, thought, and action, manage sensing and perception, facilitate diagnosis, and to support the collaboration between people and agents in real-time. A proof-of-concept prototype for teleoperation has been developed based on sensor data scenarios extracted from two different mobile robots. The focus of the paper is on details of the system design, which are presented with examples from the prototype implementation.

Introduction

Many applications involve interactions between heterogeneous agents. In semi-autonomous control of robots, for example, human cognitive agents direct physical computational agents (i.e., remote robots) in order to identify and rectify any problems encountered, such as hardware malfunctions, changes in the environment,

or errors in the task specification. Advanced robotics still has a need for keeping the "human in the loop" for two main reasons: 1) to observe the execution and state of the robot, and 2) to compensate for the information and decision-making inadequacies of the robot (Coiffet & Gravez 1991; Pin, Parker, & DePiero 1992). One key advantage of involving humans is to exploit their ability to rapidly "see" the source of a problem just by looking at the relevant data, also known as *visual problem solving*. Visual problem solving assumes that the data is presented in a useful format; however, in applications such as teleoperation, there is often too much data for a human to process without cognitive overload or too little data due to communication bandwidth limitations.

Previous work in the field of diagnostic radiology has led to the development of a cognitive model of visual interaction which attempts to capture how humans focus attention on the relevant information in an image (Rogers 1995a). This model has been used as the basis for an intelligent assistance program called VIA (Visual Interaction Assistant) which supports visual decision-making and actively modifies the image display to enhance relevant features. The blackboard-based architecture of the VIA system was chosen to facilitate the opportunistic problem-solving needed for complex and ill-structured applications. Evaluation of a preliminary prototype for radiologists, VIA-RAD, has demonstrated some promising results, both in performance and in acceptance by the users (Rogers 1995b). We have since combined the intelligent supervisory assistance provided by the VIA system with the semi-autonomous perceptual and motor control abilities of the teleoperation version of the Sensor Fusion Effects (teleSFX) architecture for mobile robots (Murphy 1993) to produce teleVIA. The utility

of teleVIA in assisting human operators in diagnosing sensing failures at the remote site is discussed in (Murphy & Rogers 1996).

The teleVIA system uses a blackboard architecture to observe and manage the information posted independently by the robot and human agents. Blackboards have been previously used successfully for teleoperation in the Ground Vehicle Manager's Associate project (Edwards *et al.* 1994), and for high level programming and control of mobile robots in a hazardous environment (Pang & Shen 1990). However, neither of these systems places an emphasis on the integration of perceptual and problem solving capabilities of the human supervisor, as is done in teleVIA. Other teleassistance systems {e.g., STRIPE (Kay & Thorpe 1993)} have similar concerns for lengthy delays in updating information for local and remote agents, but do not approach the problem from the standpoint of introducing an additional computational agent.

This paper concentrates on the domain-independent architecture used by teleVIA for facilitating visual problem solving between robots or softbots and humans. The architecture defines virtual and human agents in terms of their inherent cognitive and perceptual abilities, and their weak and strong knowledge sources. TeleVIA is the third agent, an expert assistant which mediates the flow of information, facilitates cooperation, filters and caches data, and generates the appropriate visual displays. The architecture relies on the teleVIA blackboard to serve as the central structure for permitting teleVIA to cooperatively assist the remote and human agents. Knowledge sources operating on the blackboard encapsulate the human's innate ability to recognize patterns and reason about anomalies together with the knowledge, image, and graphics processing capabilities of the computer. The agent architecture addresses the technical

issues of using an expert assistant to coordinate perception, thought, and action, manage sensing and perception, facilitate diagnosis, and to support the collaboration between people and agents in real-time.

The architecture and the intelligent assistant are discussed in terms of the teleoperation domain. A typical teleoperation scenario begins when the human cognitive agent specifies a task for the remote, situated agent to accomplish, for example move-to-object x. The remote agent is expected to perform the task autonomously and attempt to overcome any difficulties, within limits. If the robot encounters a sensing failure (e.g., dirt is kicked up on its lens), it attempts to classify and recover from the problem on its own. If it is unable to do so, or if the time involved will exceed limits on the task, it must inform the local supervisor. Rather than have the human repeat the diagnostics already performed by the robot and/or display all the sensory data, the intelligent assistant establishes what the remote agent has accomplished, posts the list (and results of tests) to the human, and then selects the most appropriate data display(s) to bootstrap the visual problem solving process. The human may immediately see the problem and intercede, approve the reallocation of resources to diagnose the problem, or generate and explore new hypotheses and tests. The intelligent assistant manages the hypothesis list, posts reminders as to the necessary tests to confirm a hypothesis, handles the requests for data and new observations from the robot, and generates the appropriate displays. The human agent can at any time override the intelligent assistant and directly select his/her own displays.

The following sections present an overview of the architecture, followed by the details of the system design including domain knowledge, system control, and knowledge sources. A

discussion of how the architecture resolves the technical issues is given next. The paper concludes with a summary and advantages of this system for teleoperations.

[Figure 1]

Agent Architecture

As shown in Fig. 1, the agent architecture consists of three agents: the *remote robot*, the *intelligent assistant*, and the *local* human. These agents can be described in terms of the three fundamental dimensions of multi-agent systems as discussed by Bird (Bird 1993), namely, distribution, heterogeneity, and autonomy. By combining three different types of agents with different levels of autonomy, as shown in Fig. 2, our system can be characterized as a collaborative multi-agent system, according to Bird's taxonomy, and this loosely coupled framework provides a basis for further discussion of the relevant technical issues.

The focus of the system is on the robot, which is expected to be located in physically remote and possibly hazardous environments, while the intelligent assistant resides locally with the human supervisor. The robot's tasks will typically involve activities such as waste cleanup, specimen collection and rescue operations. At this stage, the role of the human supervisor is primarily to monitor the robot's activities, and to make diagnostic and recovery decisions in the event of a problem. The main purpose of the intelligent assistant, therefore, is to support this decision-making process by facilitating the communication between the physically distributed agents, and by bridging the gap in the types of knowledge utilized by each to carry out their tasks.

Remote Robot Agent

The remote robot is a situated agent which is expected to sense and react to changes in the environment in order [Figure 2] to accomplish its task. It bases its decisions and actions on integrating (making "sense" of) the perceptual data obtained from its sensors. Since it is the agent in direct contact with the environment, it provides all the direct observations for use by the other agents. The robot is also expected to attempt to identify and correct any run-time problems such as sensing failures. It has a significant amount of domain-specific knowledge, and is aware of all the diagnostics that can be run on its sensor platform given symptoms of a sensing or task failure. As such, it can diagnose and recover from many instances of failures as shown in (Murphy & Hershberger 1996). However, the robot agent does not contain general problem-solving skills, which are difficult to embed at this level as discussed in (Murphy & Hershberger 1996). Of the three entities in the system, the robot's cognitive knowledge is the weakest and exists only at a low level. The robot is capable of executing and modifying its behaviors to achieve a certain task (e.g., pick up X), but it does not have any knowledge as to why it must perform the task or what any other agents are doing.

The robot may require the human supervisor to aid in diagnosis and recovery. When the robotic agent is attempting to solve its problem autonomously, it generates and tests one or more hypotheses about the current failure. If these hypotheses are denied, and the robot has encountered an anomaly that it cannot resolve, it must seek help from the human supervising agent. In addition, even if a hypothesis is confirmed, the robot may be unable to take corrective action, and therefore must, again, ask for help. Unfortunately, the weakness of the robot's cognitive capabilities introduces additional uncertainty into the failure episode, which must be resolved in order to solve the problem. Challenges

arise when it is not clear whether the outcomes of the robot's own problem solving (or exception handling) are correct or not; this is referred to as an *exception episode*. In the best case, if the robot has correctly confirmed a hypothesis, and just needs to know what action to take, the human supervisor does not need to do any extra problem solving, but can go directly to retrieval or development of a recovery plan or strategy. If the robot, however, has incorrectly confirmed a hypothesis, the local supervisor may waste time determining a recovery procedure, which is, in fact, inappropriate (i.e., go down the garden path). If [Figure 3] the robot has correctly identified and denied a diagnostic hypothesis, this means that the local assistance can focus on a smaller hypothesis search space, and, again, come to a resolution as quickly as possible. If the robot has incorrectly denied a hypothesis (due to insufficient knowledge, or limited inferencing capabilities), the supervisor may overlook the correct solution if too much trust is placed in the robot's assessment.

An intelligent assistant is needed to effectively present to the human supervisor at least two types of information from the robot: the diagnosis "trace" and the perceptual context. In general, the only reliable situation is when the robot has been able to recover autonomously. Otherwise, it is important that the "trace" of the robot's problem solving activity be communicated to the human supervisor, and that some assistance be provided in the presentation of this information which may support a rapid resolution of the problem by the human agent. In addition to its problem-solving context, the robot must also report its perceptual context when its exception-handling mechanisms are unsuccessful. From the robot's perspective, this is centered around the status of its sensors, the associated data produced by the sensors, and the belief values involved in the

failed sensor fusion process which triggered the exception episode. Fig. 3 shows the two major categories of information which must be communicated to the human supervisor through the medium of the intelligent assistant.

Local Human Agent

The human supervisor is responsible for task planning, coordination of multiple remote robotic agents, and general problem solving. It is clearly the superior cognitive agent, encompassing a strong domain theory about problem solving. In our teleoperation domain, the human agent's superior visual problem [Figure 4] solving capabilities are exploited: information presented in diagrammatic or image-based representations is quickly perceived, and critical decisions can be made in a timely manner. Our characterization of the human supervisor as a cognitive agent is based on an information-processing model of human visual interaction previously developed in the domain of diagnostic radiology (Rogers 1995a; 1995b), which describes how perception and problem solving processes might exchange information in working memory.

However, the cognitive abilities of the human supervisor are also limited. One of the results of this work showed that human professionals performing image-based diagnosis may experience three different types of errors or oversights: 1) at the perceptual level, a *detection oversight* may occur when the person does not notice or see the abnormal object or feature at all; ii) at the identification level, a *labeling* error can occur if the person sees the anomaly in question, but labels it incorrectly; and iii) at the problem solving level, *integration* errors occur when people see and label anomalies correctly, but fail to use this information in the generation of diagnostic hypotheses. These results suggest that the design of computerized assistance for such tasks should support the

spectrum of interaction between perception and problem solving by integrating appropriate image enhancements with automated decision aids. In the telerobotics domain, a further technical challenge is to bridge the gap between the robotic-based perceptual knowledge and the human-based cognitive processing of that knowledge.

Intelligent Assistant

The novel aspect of our architecture is the third intelligent assistant agent, teleVIA. The objective of the teleVIA assistant is to transform the robot's exception episode into a form that can be easily understood and analyzed by the human supervisor. However, this is not simply a matter of one-to-one display tactics: teleVIA provides added value to the presentation of the information in the form of perceptual and problem solving enhancements.

These responsibilities are handled by the four components of teleVIA, shown in Fig. 4. This model is based on Riecken's suggestion to integrate several different agents in order to "realize a software assistant capable of performing a broad range of tasks" (Riecken 1994). The *Attention Director* handles the perceptual and attentional aspects of the assistance session. It is concerned with requesting relevant sensor data, displaying that data in the possible formats: raw, default-enhanced, or knowledge-base-enhanced, and providing attentional directives to the human supervisor. The *Hypothesis Manager* is responsible for the presentation and management of diagnostic hypotheses: incorporating the results of the robot's problem solving with the generation and testing of additional hypotheses. The *Strategy Selector* guides the activities of both of these entities, providing high-level rules for conducting the problem solving session. Finally, the *Interface Manager* is used to control the human agent's view of the

interactive session, and to manage the display of the presentation graphics (Rogers & Murphy 1994).

Te1eVIA Implementation

In order to effectively collaborate with the other two agents, the teleVIA assistant must "know" a great deal about the robotic agent, and present this information in a form compatible with the capabilities of the human agent. The main concepts of the domain knowledge, therefore, include facts, data, and hypotheses related to the robot's operation. The teleVIA blackboard is the central data structure of the collaboration among the three agents, and it is divided into logical partitions, called *panels*, which provide groupings of the domain knowledge that are consistent with components of the cognitive model of human visual interaction described in (Rogers 1995a). The four general categories of context, perception, hypotheses and attention are reflected in the *Current Context Panel*, *Interactive Exception Handling Panel*, *Hypotheses Panel* and *Attention Panel*, respectively. The latter two are further subdivided into two distinct regions, and an additional panel, called *Interactive Configuration*, allows the feedback of recovery procedures from the local supervisor to the remote robot. A frame-based representation has been used to capture both the declarative and procedural aspects of the underlying domain concepts, and an example of the sensor frame is shown in

Fig. 5. System Control

In the implemented system, events serve as an interface between blackboard database operations and the [Figure 5] control shell, and are either predefined events that occur when the state of the blackboard database changes (e.g., a frame is instantiated on a panel), or events that occur for user-defined purposes. Events are used to trigger user

interface functions as well as the knowledge sources which perform the cooperative problem solving. The control shell in this version of the prototype is based on a precondition/action model in which the activation of a triggered knowledge source is determined by a precondition function, and the execution order of the resulting action function is controlled by an execution rating returned by the precondition function (GBB 1995).

The system's user interface presents two logical user views: monitor mode, which is used solely for monitoring the robot's normal behavior, and failure (or problem-solving) mode, which appears when a request for assistance is sent by the robot. All panels, when visible, are presented as interactive direct manipulation windows, with a variety of icons, text and buttons used to display the underlying frame-based information. Color-coding is used to indicate correspondences between related data distributed across several panels.

A failure scenario in which we utilize previous data from a robot to demonstrate teleVIA's capabilities, can be broadly described in terms of five distinct steps: i) robot sends failure information; ii) teleVIA generates its own hypotheses; iii) teleVIA requests and posts relevant images; iv) teleVIA tests its hypotheses with the help of the human supervisor; and v) recovery procedures are determined. Fig. 6 shows a sample screen after all these steps have occurred, and illustrates the different panels described below.

In monitor mode, only the Current Context window and a restricted version of the Attention window are visible to the user. This allows the supervisor to request intermittent sensor images from the robot, and to track current environmental expectations. The system startup begins with monitor mode, and once the initial knowledge source has created all the instances

on the current context panel, and the graphical interface has been initialized, the control shell continues to check for a failure event, and monitors the time on task.

In the current implementation, the transition from monitor mode to failure mode is made by selecting a particular scenario from an interactive window. This causes a file to be created which contains the specific information which would be sent by the robot, and which is needed to fill the exception-handling (EHKS) frame described in the next section. The creation of this file signals a failure event, and the user is alerted by the display of a failure button which also sounds an alarm. Acknowledgement of the alert switches teleVIA into failure mode, and this event causes the interactive exception handling, attention and both hypothesis windows to be activated and displayed. The EHKS frame, including subordinate bodies-of-evidence and robot hypothesis frames are instantiated with the corresponding information from the failure data file, and posting of this instance to the blackboard is a trigger event for the knowledge sources which initiate the perceptual and problem solving activities.

Domain Knowledge

The Current *Context Panel* contains information about the current robot and its sensor configuration, the task to be performed, and the known environmental factors and conditions. This information is retrieved from the knowledge base at the beginning of the current mission, and much of it is expected to remain static for the duration of this particular mission.

The *Interactive Exception Handling Panel* is where the perceptual status of the robot is posted when a failure is signaled. This includes the type of failure, currently active sensors, and the sensors' belief values at the time of the sensor fusion attempt. The exception handling knowledge structure (EHKS) frame is a complex knowledge structure which

allows transfer of all the information relevant to a failure situation from the robot to the local system, and is based on the analogous structure produced by the teleSFX exception handling module (Chavez 1994). It provides failure characteristics (failure step and failure type), the sensor data leading to the fusion failure (a bodies-of-evidence frame for each sensor), and a history of the robot's exception handling behavior (one or more robot hypothesis frames). If the robot decides to reacquire sensor data as part of its attempt to recover, additional EHKS instances will be generated, and all of these must be considered by teleVIA's problem solving mechanism as a profile of activity since the original failure. The display of the different portions of the EHKS frame is actually distributed over several panels: the bodies-of-evidence information appears on the Interactive Exception Handling Panel as shown in Fig. 7, whereas the raw image data will be posted to the Attention Panel.

The robot hypotheses are displayed on the *Robot Hypothesis Panel*. [Figure 6] [Figure 7] This panel 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.

The *TeleVIA Hypotheses Panel* maintains the failure hypotheses generated by the knowledge sources of teleVIA. These hypotheses are represented by frames which include the hypothesis name (e.g., ultrasonicsensor-malfunction), symptoms associated with the failure (may be procedures which query the supervisor or the knowledge base), a further checklist for gathering related information (also procedural attachments), a list of possible causes (for explanation purposes), a belief value, and a list of related recovery procedures.

The purpose of the *Attention Panel* is to focus the attention of the human supervisor on relevant data, and therefore, this panel is used to display both raw and enhanced data/images,

attention directives to the user, and the current problem solving steps employed by teleVIA's knowledge sources.

Knowledge Sources

The knowledge sources (KS's) of the teleVIA system are divided into two major categories, depending on whether they are based on perceptual or problem solving schemas. The perceptual KS's, under the control of the *Attention Director*, emphasize the automatic selection, display and enhancement of sensor data which may most effectively assist the local supervisor in identifying and responding to the failure situation. On the other hand, the problem solving KS's (handled by the *Hypothesis Manager*), are concerned with managing the hypotheses about the failure, including calculating teleVIA's beliefs, gathering more evidence when necessary, either from the robot, or from the human user, and providing related recovery strategies. [Figure 8]

The knowledge sources which aid the user's perception have two main goals: **1)** request the most "effective" sensor data to be transmitted from the robot: this involves a tradeoff between choosing sensor data whose information is easily perceived by the human user (e.g., a video image) and data which can be quickly transmitted; and **2)** automatically invoke enhancements (either default or knowledge-based) which can improve the data display even further. Therefore a set of four related knowledge sources, detailed in Fig. 8, has been designed for each sensor. Competition for execution among these KS's can be either inter- or intrasensor, depending on the purpose of the KS.

On the other hand, TeleVIA's problem solving knowledge sources are designed to generate and test failure hypotheses, either automatically, or with the help of the human supervisor. The general form for these KS's is shown in Fig. 9, but, in practice, they must

be hand-crafted to address particular conditions related to the individual sensor types. While beliefs are calculated automatically by the system, and are altered by the system when additional evidence becomes available (typically through interaction with the human supervisor, the human agent always has the capability of manually changing the belief values. Recovery procedures are then posted in order of beliefs in related hypotheses.

Discussion

As seen in the previous sections, teleVIA was explicitly designed to coordinate perception, thought, and action, manage sensing and perception, and facilitate diagnosis. An important issue is whether teleVIA can manage these collaborations between people and agents in real time. The *task* frame includes two deadlines; the impact of these time constraints and how they can be computed are explored further in (Murphy 1995).

The relationship of teleVIA to VIA merits further discussion. VIA consists of two agents: the human cognitive agent (doctor) and the intelligent assistant (VIA). In the medical diagnostic domain used, there was no remote agent able to collect or operate on the medical data autonomously. Therefore teleVIA extends the VIA concept from strictly being an assistant to the cognitive agent, to being an assistant to the remote as well. TeleVIA *was* shown to interpret the EHKS data coming from the remote situated agent and distribute it over different panels to facilitate operator understanding. Likewise, teleVIA converts calls by the human for more data into instantiations of the appropriate knowledge source (e.g., Get Sensor Data).

Although the test bed for our efforts has been teleassistance of mobile robots, we believe this work is generalizable to other applications. The three agent partitioning is applicable to any operation involving a situated agent and a human. We are currently involved in adapting this for the remote control of power stations by the utility industry, where the control software at the station serves as a situated agent.

At this time, one of the challenges in this project is the lack of a strong domain theory, due to individual robot configurations and constraints of the applications. The problem solving knowledge sources, in particular, must be hand-crafted as individual problem solving schemas are identified, and it is often not known in advance how the robot itself will behave under various circumstances. However, the inclusion of the human supervisor's input allows teleVIA to generate hypotheses which are much more tailored than those the robot is capable of generating on its own. Therefore, future work is planned to explore the role of learning *between* teleVIA and the human user, as well as between teleVIA and the remote robot.

Summary

The research presented in this paper links a human cognitive agent, a remote situated agent, and an intelligent (expert) assistant in a collaborative problem-solving system which combines the human's innate ability to recognize patterns and detect anomalies with the knowledge, image and graphics processing capabilities of the computer. It contributes an architecture which supports the collaboration between people and agents. Under the teleVIA system, the three agents cooperate using an asynchronous blackboard. The blackboard is divided into four general categories of domain knowledge: context (hardware, task, known environmental factors), perception (active sensors, belief, failure information), hypotheses

(hypotheses generated by robot, by intelligent assistant, and by human), and attention (raw and enhanced data images, attention directives, current problem solving steps). The knowledge sources operating on the blackboard are either perceptual (supports visual interaction) or problem solving (managing hypotheses about a sensing failure).

The addition of such an intermediate intelligent assistant 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; and 4) to improve efficiency by reducing the need for supervision so that one person could control multiple robots simultaneously. Furthermore, the highly modular and adaptive nature of the systems is expected to support the incremental evolution of telesystems to full autonomy.

Acknowledgements

This research is currently supported by the Army Center of Excellence in Information Science at Clark Atlanta University, ARO Grant # DAAL03-92-G0377, and National Science Foundation Grant # IRI9502289.

References

- Bird, S. D. 1993. Towards a taxonomy of multi-agent systems. *Int. Journal of Man-Machine Studies* 39:689-704.
- Chavez, G. T. 1994. Exception handling for intelligent sensor fusion. Technical Report MS Thesis T-4422, Colorado School of Mines, Golden, CO.
- Coiffet, P., and Graves, P. 1991. Man-robot cooperation: Towards an advanced teleoperation mode. In Tzafestas, S. G., ed., *Intelligent Robotic Systems*, 593-636. New York: Marcel Dekker.
- Edwards, G. R.; Burnard, R. H.; Bewley, W. L.; and Bullock, B. L. 1994. The ground vehicle manager's associate. Technical Report AIA A-94-1249-CP, AIAA.
1995. Gbb concepts, version 1.2. Technical report, Blackboard Technology Group, Inc., Amherst, Mass.

- Kay, J., and Thorpe, C. 1993. Stripe: Supervised telerobotics using incremental polygonal earth geometry. In *Proc. of the Third International Conference on Intelligent Autonomous Systems*, 399-406.
- Murphy, R. R., and Hershberger, D. 1996. Classifying and recovering from sensing failures in autonomous mobile robots. In *Proc. of AAAI-96*, 922-929. Morgan Kaufmann.
- Murphy, R. R., and Rogers, E. 1996. Cooperative assistance for remote robot supervision. *Presence* 5(2):224-240.
- Murphy, R. R. 1993. Robust sensor fusion for teleoperations. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, volume 2, 572-577. IEEE.
- Murphy, R. R. 1995. Estimating time available for sensor fusion exception handling. In *Proc. of SPIE Conf. on Sensor Fusion and Networked Robotics VIII*. SPIE.
- Pang, G. K. H., and Shen, H. C. 1990. Intelligent control of an autonomous mobile robot in a hazardous material spill accident - a blackboard structure approach. *Robotics and Autonomous Systems* 6:351-365.
- Pin, F. G.; Parker, L. E.; and DePiero, F. W. 1992. On the design and development of a human-robot synergistic system. *Robotics and Autonomous Systems* 10:161-184.
- Riecken, D. 1994. M: An architecture of integrated agents. *Communications of the ACM* 37 (7): 107-116.
- Rogers, E., and Murphy, R. R. 1994. Tele-assistance for semi-autonomous robots. In *Proc. of AIAA Conference on Intelligent Robots in Field, Factory, Service and Space*, 500-508. NASA Conference Publication 3251.
- Rogers, E. 1995a. A cognitive theory of visual interaction. In Chandrasekaran, B.; Glasgow, J.; and Narayanan, N. H., eds., *Diagrammatic Reasoning: Computational and Cognitive Perspectives*, 481-500. Menlo Park, CA: AAAI/MIT Press.
- Rogers, E. 1995b. Via-rad: A blackboard-based system for diagnostic radiology. *Artificial Intelligence in Medicine* 7:343-360.

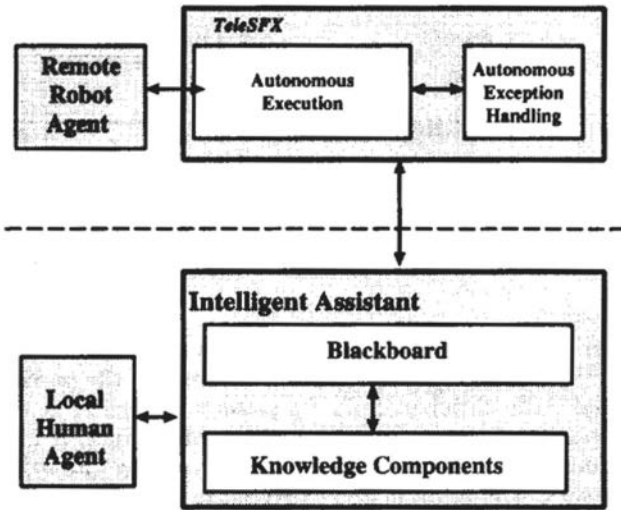


Figure 1: Overview of three types of agents.

Agent	Distribution	Heterogeneity (Agent Type)	Autonomy
<i>Robot</i>	remote	situated computational robotic agent	semi-autonomous
<i>Intelligent Assistant</i>	local	collection of computational software agents	not autonomous
<i>Human Supervisor</i>	local	cognitive agent	autonomous

Figure 2: Dimensions of the Multi-Agent System

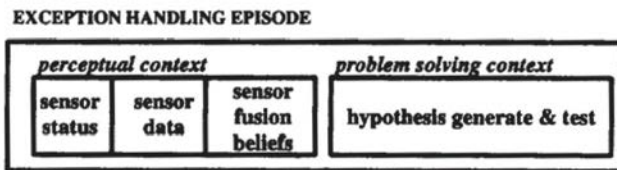


Figure 3: Information Needed for Exception Episode.

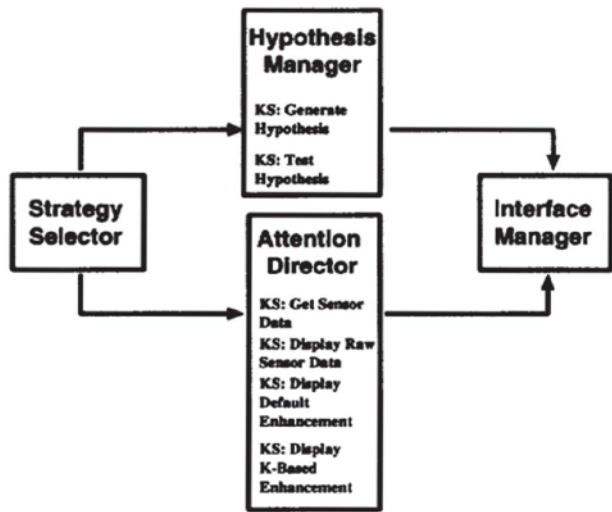


Figure 4: Components of the TeleVIA Intelligent Assistant.

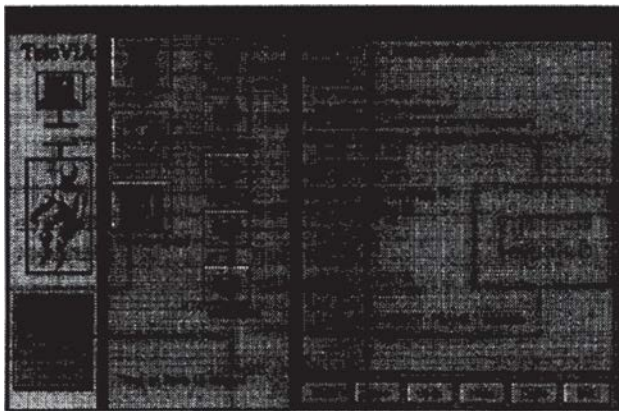


Figure 5: Details from a Sensor Frame.

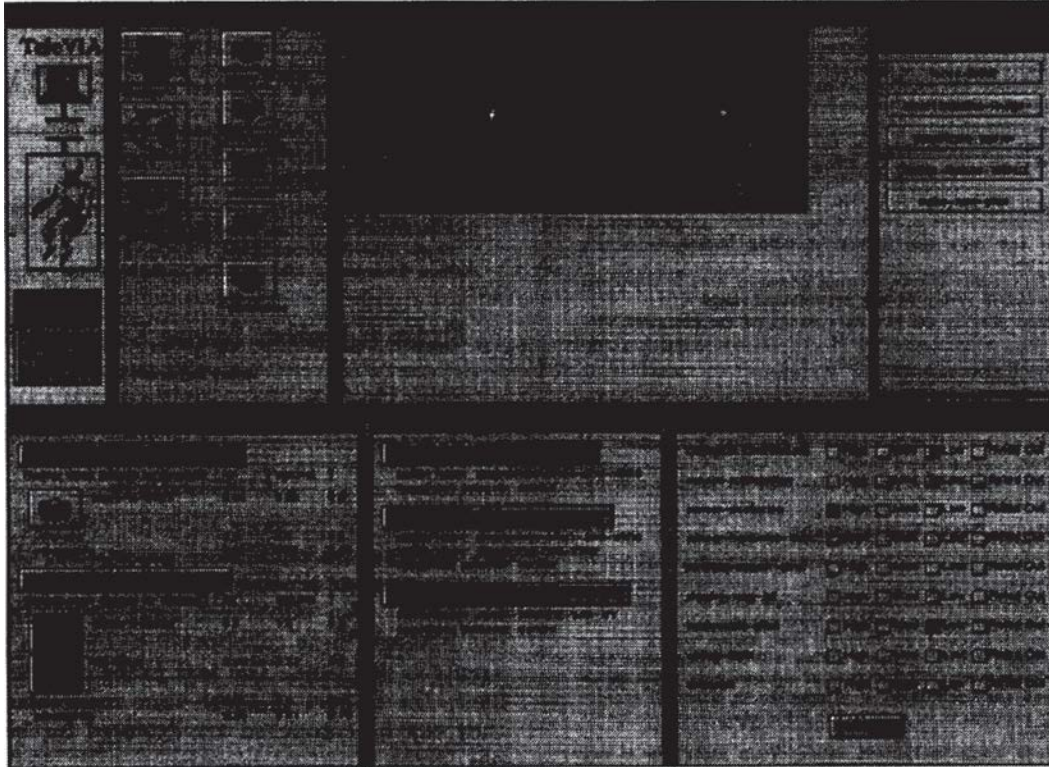


Figure 6: Sample TeIeVIA Screen.

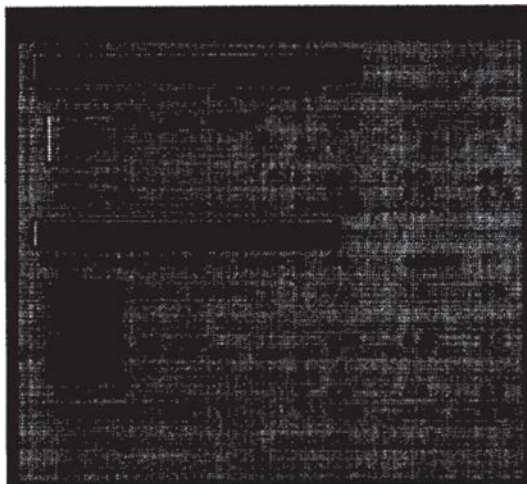


Figure 7: Interactive Exception Handling Panel

1. Get Sensor Data:

Trigger Evan!' creation of BUM frame instance

Precondition Function! check whether the sensor is in the suspect sensor list; calculate the priority rating based on a weighted combination of the perception rating and the time rating slots of the sensor frame.

Action Functions send a message to the robot to transmit the raw data, and return the raw data file name when transmission it complete.

2. Display Raw Sensor Data:

Trigger Events raw data slot in corresponding bodies-of-evidence frame it updated

Precondition Functions if transmission of sensor data is complete, assign the execution rating from a pre-selected range

Action Function: display the raw data image on the attention window.

3. Display Default Enhancement;

Trigger Events raw data slot in corresponding bodies-of-evidence frame is updated

Precondition Functions if transmission of raw data is complete, assign rating from a pre-selected range

Action Function' apply enhancement algorithm to raw data and display resulting image.

4. Display Knowledge Based Enhancement:

Trigger Event' raw data slot in corresponding bodies-of-evidence frame is updated

Precondition Functions if transmission of raw data is complete and specific knowledge needed is available from appropriate blackboard panel (e.g., environmental conditions), then assign rating from a pre-selected range. (Note that this rating will be higher than that of default enhancement; if both types of enhancement KS's are on the queue, and the cycle ends, it is possible to either a) reorder these KS's by priority, or b) remove the lower priority enhancement KS's altogether.)

Action Function; apply specialised enhancement algorithm to raw data and display resulting image.

Figure 8: Perceptual Knowledge Sources.

1. Generate teleVIA hypotheses

Trigger Event: creation of EHKS frame instance

Precondition Function' if the XS is sensor-specific, then check if sensor is in suspect sensor list; check failure step/failure type slot in EHKS frame

Action Functions post teleVIA hypothesis instance to hypothesis panel with initial belief yaks.

2. Test teleVIA hypotheses

Trigger Events creation of teleVIA hypothesis instance Precondition Function' match hypothesis name; check symptoms: check related information

Action Function: raise belief in hypothesis; post yecLidesy strategies.

Figure 9: Problem Solving Knowledge Sources.