Coordinating a Multi-agent Team Using A Multiple Perspective Interface Paradigm

J. Alan Atherton, Benjamin C. Hardin, Michael A. Goodrich
Brigham Young University
Computer Science Department
Provo, UT 84602
jaa39@email.byu.edu, bch36@email.byu.edu, mike@cs.byu.edu

Abstract

A mission to Mars will be composed of several groups that need to interact effectively. Scientists on Earth, supervisors in a habitat on Mars, and a surface exploration team on Mars will all require different views of information. This position paper proposes a multiple perspective interface paradigm which implements augmented virtuality and multiple camera perspectives to clearly present these views.

Overview

Envision a mission to Mars, comprised of a multi-agent team of humans, Unmanned Ground Vehicles (UGV), and Unmanned Air Vehicles (UAV). A habitat on Mars serves as a central base for this team, with one or more supervisors remaining at the base while multi-agent sub-teams leave the habitat to explore, collect samples, or fulfill other tasks. The supervisor has the job of assigning those tasks in the larger context of the overall mission, determining team composition, and remaining in communication with the field team to provide task clarification, give additional instructions, and receive gathered information.

An Earth-based control center oversees the mission. Because of the time delay involved in Earth to Mars communication, scientists on Earth make only high-level decisions. They receive data from the Mars habitat, analyze it, and update mission parameters if necessary. A similar mission scenario is outlined in more detail by (Mendell 1991). Figure 1 depicts this hypothetical team structure.

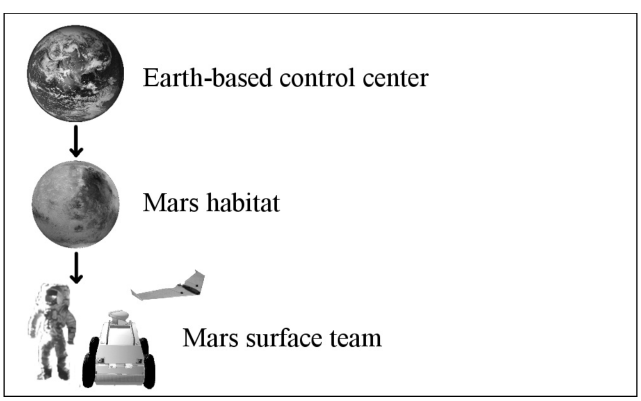

Figure 1: Hypothetical team structure for Mars exploration.

With such a variety of roles, the problem of presenting information to each human agent in a coherent manner becomes significant. A surface team may require topographical, thermal, or weather data, while the supervisor in the habitat may want astronaut locations, intentions, and the condition of robotic agents. Earth-based scientists may be interested in scientific data that has been gathered, task progress, and overall astronaut well-being. In addition to the problem of information presentation, there are inherent difficulties in coordination between a human and a robotic or other human team member; these difficulties are compounded by time delays in communication.

In this paper we propose a multiple perspective interface paradigm that grounds team communication and promotes team situation awareness by granting each operator/user a task-appropriate view of the data. In this paradigm data is shared, but the interface displays only data pertinent to the role of the human accessing it (Scholtz 2002) (Parasuraman, Sheridan, and Wickens 2000). Because the data is shared, agents have the ability to view it from the perspective of any other agent, or even from synthesized camera views such as third-person or top-down.

To accomplish the multiple perspective display we use augmented virtuality to increase user comprehension, safety, and decision-making abilities. For example, live video can be embedded directly in the interface, allowing a supervisor to see actual images from the perspective of an individual agent, while maintaining an overall situation awareness. Multiple views can be layered over each other, showing relationships between different layers while only showing pertinent data. Agents, structures, or other objects of interest can be shown iconically or with increased detail depending on distance, whether a user indicates an interest, or some other metric. Failure indications can be combined with a decision support system, and peril or no-fly zones are clearly represented. These features support team situation awareness by allowing team members to know the location of other team members and share points of interest.

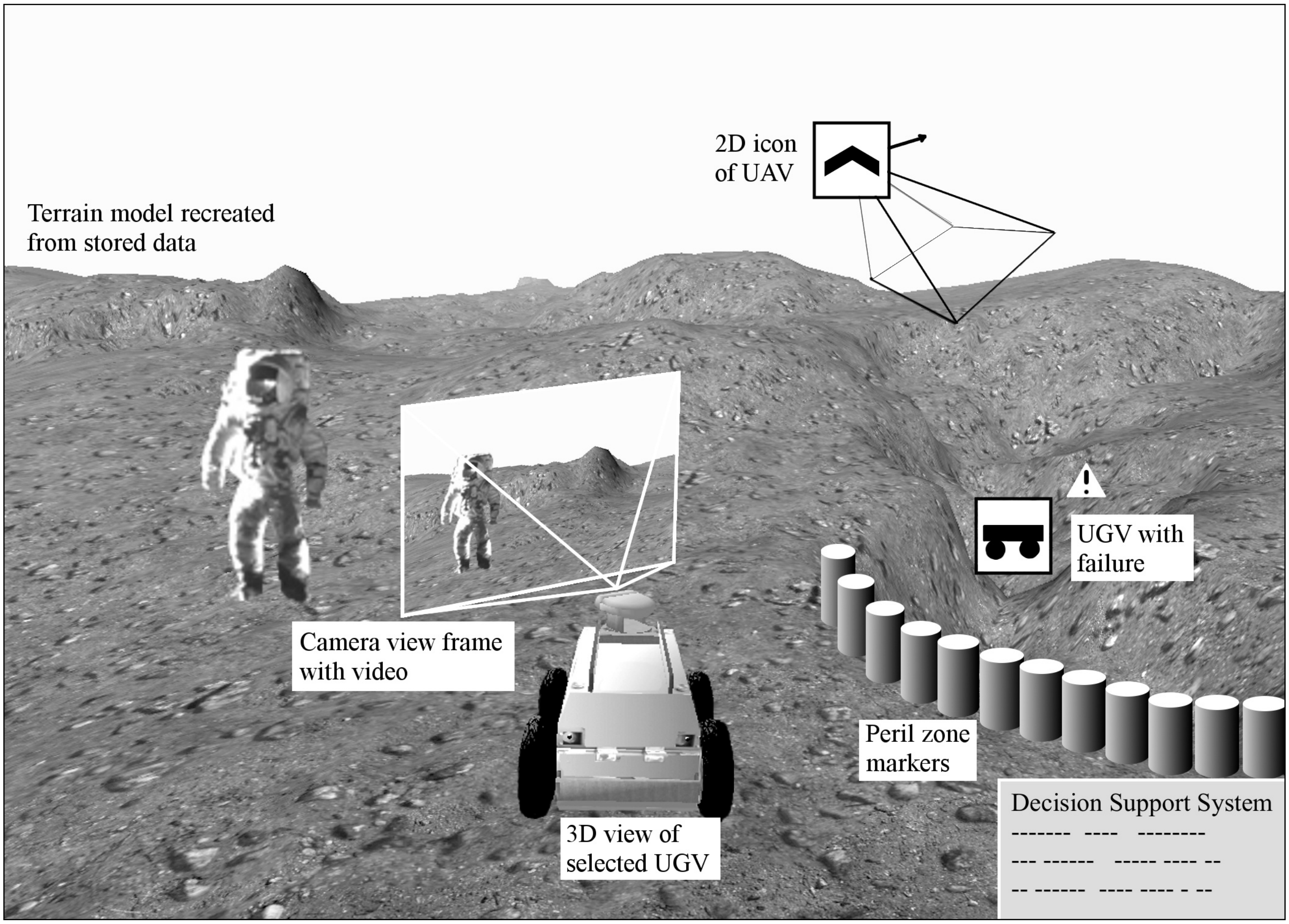

Figure 2: Interface mock-up displaying elements of augmented virtuality.

Augmented Virtuality

Augmented virtuality is a way to represent real-world information in a virtual environment (Drascic and Milgram 1996). This approach allows an agent to view sensor data, for example, in context with other elements of the world instead of as a seemingly disjoint set of information. The following list explains some elements of a multiple perspective interface which uses augmented virtuality; see Figure 2.

- **Camera view frames** depict the wireframe view frustum of a camera. Live video can be shown directly in the view frame. This allows other agents or supervisors to see what a camera is looking at in context. This is useful in supporting task switching and acquisition of situation awareness in the new task. The operator of a team of robots can quickly determine which robot's camera is aimed at a desired target, then show the live video feed for the desired camera. In Figure 2, a wireframe from the UAV outlines the camera's view.

- **Quickening** (not shown in figure) is a technique to cope with delayed information by predicting the future state of an agent. For example, when an operator sends a camera movement command (e.g. pan left), the interface immediately updates the camera view frame to show the expected movement of the camera. Once the camera has completed the command and transmitted its actual orientation, the interface adjusts the view frame if necessary (Ricks, Nielsen, and Goodrich 2004).

- A **decision support system** shows data relevant to an event. For example, when an agent enters a failure mode, an alert icon will appear next to the agent in the interface. In addition, the decision support system will present data about the failure as well as potential courses of action. See for example (Pinson, Louca, and Moraitis 1997). In Figure 2, the decision support system is shows data relevant to the selected UGV.

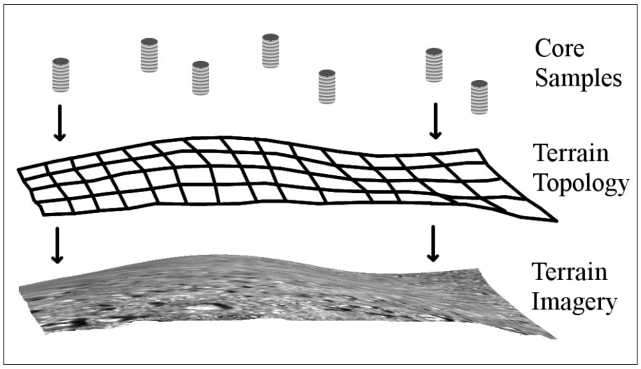

Figure 3: View of three possible data layers.

- **Multiple data layers** allow the view to be customized for a particular task. In Figure 3, for example, core sample data is layered on top of the terrain topology and imagery to show their relationship. Other possible layers include: agents' intended paths, no-fly zones, altitude, and agent icons. These layers can be stacked or hidden.

- **Agent icons** represent agents in the interface. These icons have varying levels of detail depending on factors such as the user's role, proximity, or explicit request. The lower levels of detail use a 2D icon to depict information such as agent type, location, and heading (e.g., the UAV and failed UGV in Figure 2). Closer views use a 3D model to show agent pose and other details (e.g., the astronaut and selected UGV in Figure 2) (Smallman and St. John 2005).

Multiple Perspectives

Augmented virtuality makes it possible for an interface paradigm to include a synthesized camera view that can be positioned anywhere in the environment. We propose that a multiple virtual camera perspective interface allows agents to more easily understand the situation and better support team situation awareness. Overhead views assist a supervisor in gaining a general idea of the situation, while close-up views show more detail for a robot operator. This interface paradigm enables an agent to view the situation from virtually any perspective. As illustrated in Figure 4, we will discuss three of the possible classes of camera perspectives: overhead, third-person, and first-person.

The overhead camera perspective (Figure 4 left) shows an overview of the entire mission area, much like a satellite view. This perspective would likely be frequently used by an Earth-based team to show general mission progress. For example, if one of the mission tasks is to obtain additional imagery from a particular region, the distant perspective can show which parts of the region have been photographed. Upon request, the actual imagery corresponding to the region can be viewed.

For a human operator controlling several robots, the third-person perspective is useful to assess the needs of each of the robots (Figure 4 center). The operator can quickly see the relative position and orientation of all of the robots at once. The third-person perspective can also follow a particular agent for an "over-the-shoulder" view. We propose that the task of piloting a robot becomes easier when this perspective is combined with augmented virtuality features such as peril zones, paths, and camera view frames (Ricks, Nielsen, and Goodrich 2004).

When an agent needs to see exactly what a robot's camera has in view, the first-person perspective (Figure 4 right) shows the live video feed of the camera at close range. This is useful when the operator needs to see greater detail in the video coming from a camera.

Model

Augmented virtuality uses a model as the basis for building a view and creating a synthesized camera perspective. The model is loosely defined as the collection of all the data that has been gathered. The model functions like a centralized system even though data can be buffered on a robot before sending it to the shared storage on Mars, and similarly before sending the shared storage to Earth.
For example, a top-down perspective of a building under construction can be recreated from the model, even if no camera actually recorded that view. Similarly, rather than displaying a simple view with current data, the model

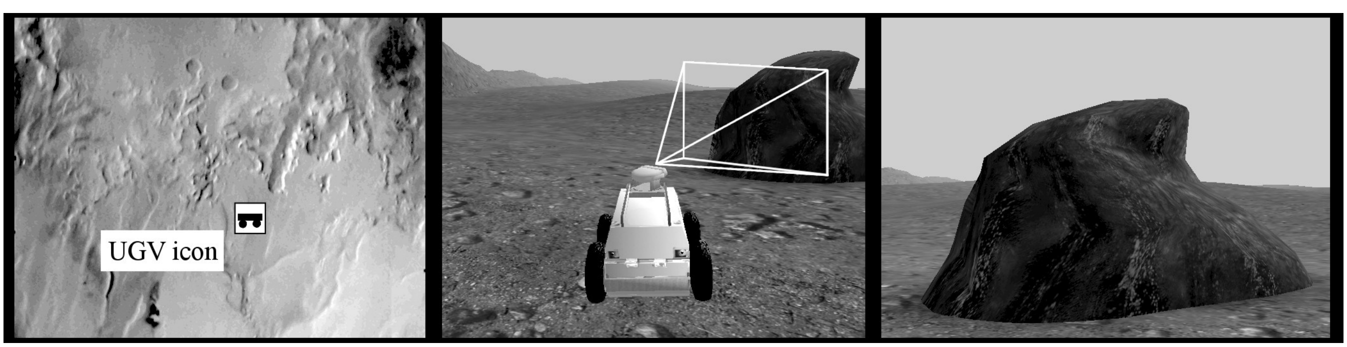

Figure 4: Three synthesized camera perspectives. Overhead (left), third-person (middle), and first-person (right). Left image taken from http://nssdc.gsfc.nasa.gov/image/planetary/mars/f225b69.jpg

stores historical, extrapolated, and even manually entered data to create an augmented virtuality.

One of the main requirements of the model is the ability to store any granularity of data, and display it at the level of detail requested by the user. For example, initial satellite imagery can be stored in the model, and as exploring UAVs and UGVs take more detailed pictures, the additional imagery can be made available. An agent may collect or generate information that is normally irrelevant, or that is not possible to keep locally on Mars due to limited storage resources. Additional information can be transmitted to the Earth station for storage, but even that is limited by available bandwidth.

The decision of what information to store locally on Mars and what information to transmit to Earth can be made in two main ways: an autonomous decision by the interface software, or an explicit request from the user. For example, the autonomous decision to store a higher granularity of data is triggered by a failure situation in some aspect of the mission. Information about the failure takes precedence in bandwidth or storage allocation, which may result in loss of other information, but facilitates error detection, correction, and future prevention.

An explicit request by a user is straightforward. For example, something may catch a supervisor's or scientist's interest, and they want additional information about it. An experiment might be taking place that a supervisor wants detailed data collected about it. It could also be as simple as a supervisor selecting an agent in the interface, which indicates to the model that telemetry from that agent should take precedence (Nielsen, Ricks, Goodrich, Bruemmer, et al 2004).

By allowing user requests to prioritize what data to collect and transmit, multiple camera perspectives and augmented virtuality remain possible while still taking into account limited resources.

Case Study

To illustrate the concepts outlined above, consider a case study involving the same team configuration outlined at the beginning of the paper: a surface exploration team of UGVs, UAVs, and humans that are managed by a supervisor in a habitat on Mars, while Earth-based scientists oversee the project.

While viewing the satellite imagery data layer in his interface, a scientist notices an abnormal discoloration on the terrain. Finding that there is no additional information about that region, he uses his interface to send a request to the team supervisor on Mars to explore the area in more detail.

On Mars, the supervisor receives the request in her own interface, and arranges an exploration mission. To begin the mission, a UAV is dispatched to collect additional imagery and topography data, while a UGV with various sensors and specialized instruments follows on the ground.

En route to the region of interest, the UGV becomes stuck in the sand. Using internal sensors to determine that it is in a failure situation, it transmits buffered state information for the time leading up to the failure, including data on position, orientation, wheel alignment, etc. The supervisor's interface receives the data, and indicates visually that a failure has occurred. Using the decision support system, the supervisor realizes that the problem involves the wheel of the UGV, and decides to get a closer look using the UGV's camera. Quickening and camera view frames assist in properly orienting the camera, and the supervisor obtains a view of the wheel.

Realizing that the UGV is unable to free itself, the supervisor decides to send a team of humans out as part of a surface exploration mission to assess the situation. Using her interface, the supervisor selects the nearest human agents and reassigns them to rescue the UGV. The team uses the view from the UAV to get a better idea of the situation, and plan a path to the trapped UGV.

Arriving at the UGV, the human team frees it from the sand, then uses their interface to make sure the robot has returned to an operational state. They then use their interface to tell the UAV and UGV to resume their mission.

An Approach to Experimental Validation of Multiple Perspective Interface Concepts

A multiple perspective interface paradigm such as this will need extensive validation, not only of the interface as a whole but of individual features. For example, consider the case where we wish to test the effectiveness of camera view frames in helping a user regain situation awareness when switching from one agent's view to another. An experimental validation may be as follows.

First, we establish a scenario where a user may need to switch views. Take, for instance, a potential situation on Mars, where a supervisor sends a UGV to collect rock samples. The UGV has a first-person camera, and a camera mounted on a collection arm. Once the robot has arrived at the collection location, the supervisor uses the first-person view to move from rock to rock, and the arm-camera view to look at individual rocks close-up before selecting candidates for collection.

Second, to determine whether camera view frames help the supervisor maintain situation awareness while switching between first-person to arm cameras, we create two instances of the interface, one with the view frame feature and one without. We establish relevant metrics to measure the results, in this example possible metrics include the amount of time needed to complete the task, the number of errors committed, and a more subjective measure of workload on the operator.

After running the scenario, we compare the results of using the interface feature against the results of not using it, and determine whether the feature is detrimental, insignificant, or beneficial. Based on the results, we can either remove or refine the feature further.

This example of validating camera view frames is typical of the process used to validate any potential feature. The elements of this process include establishing a scenario, measuring the relative benefits of the feature, then refining the design.

References

Drascic, David and Paul Milgram. 1996. Perceptual Issues in Augmented Reality. *SPIE: Stereoscopic displays and Virtual Reality Systems III*, 2653.

Mendell, W. 1991. A Mission Design for International Manned Mars Mission. http://ares.jsc.nasa.gov/HumanExplore/Exploration/EXLibrary/DOCS/EIC036.HTML

Nielsen, C. W., B. Ricks, M. A. Goodrich, D. Bruemmer, D. Few, and M. Walton. 2004. Snapshots for semantic maps. In *Proceedings of the 2004 IEEE Conference on Systems, Man, and Cybernetics.*.

Parasuraman, R., T. Sheridan, and C. Wickens. 2000. A model for Types and Levels of Human Interaction with Automation. In *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 30(3):286-297.

Pinson, S.D., J.A. Louca, and P. Moraitis. 1997. A distributed decision support system for strategic planning. *Decision Support Systems*, Volume 20, Number 1, May 1997, pp. 35-51(17)

Ricks, B., C. Nielsen, and M. Goodrich. 2004. Ecological Displays for Robot Interaction: A New Perspective. *Proceedings of IROS 2004.* Sendai, Japan.

Scholtz, J. 2003. Theory and evaluation of human robot interactions. Hawaii International Conference on System Sciences (HICSS-36), Big Island, Hawaii.

Smallman, H. S. and M. St. John. 2005. Naïve realism: Misplaced faith in the utility of realistic displays. *Ergonomics In Design*. In press.