Sloan_2002

A MODULAR MULTI-FUNCTION ROVER AND CONTROL SYSTEM FOR EVA*

Stephen McGuire, Joel Richter, Kevin Sloan
The Penn State Mars Society
A Chapter of The Mars Society at
The Pennsylvania State University – University Park
http://www.clubs.psu.edu/mars

INTRODUCTION

As technology and computing power increase at extraordinary rates, our ability to effectively explore our solar system increases to new levels. The immediate future will see the continual development of robotic exploration as our primary means of exploring other planets, and more specifically, Mars. Once the time does arise for mankind to again push his frontiers to new limits, our definition of space exploration will be completely redefined. However, human exploration of our solar system cannot happen without the assistance of our robotic counterparts which helped blaze the trail into space. This transition to human exploration will see astronauts virtually isolated by lengthy communication delays as long as forty minutes, and consequently being required to maintain an extremely high level of self sufficiency. Expectations of such a high profile mission will also mandate that copious amounts of field work and related studies be conducted over its duration. Due to their circumstances, early Martian explorers will be forced to work with equipment that was at one time controlled by large teams of scientists and engineers with immediate access to significant computing resources. In order for these astronauts to be able to maximize their time, and produce extraordinary quantities of data, new methods for human-rover interaction in planetary exploration must be developed.

One scenario that shows a strong need for more research into new methods is a small team of astronauts (or possibly even a single astronaut) on the surface of Mars conducting field work out of the immediate reaches of their base. This seemingly commonplace situation will find team members out facing the elements while conducting research. It seems only natural for research rovers to accompany the team into the field. However, having to deal with robotic equipment while in a space suit presents several issues, the primary one being control. By incorporating virtual reality glove technology into the astronaut's gloves, his hands become a quick, easy and effective input device. A simple hand command can activate the gloves, and the rover begins to respond to hand gestures, which are interpreted as commands. Our isolated astronaut now has complete flexibility in the control over all of the different robotic equipment and machines that will be in the field with him.

All plausible mission outlines for the first manned missions to Mars entail the crew collecting extraordinary amounts of data in a plethora of different areas. Most of the field work would be conducted with research rovers such as those described above, each specializing in a different area. The amount of rovers to be sent will quickly add up. A much smaller fleet of modular rovers allows for mission flexibility while significantly cutting back on overall mission cost and weight. These modular rover bases, which will accept a wide variety of scientific units, will operate on a very standard platform. Because all of the equipment will have interchangeable components, the astronauts will be able to troubleshoot most basic problems that could arise in the field. This modularity, coupled with the simplicity of the glove input, tackles many of the difficulties that an astronaut would face in the field that would otherwise severely hinder his productivity.

GESTURE BASED CONTROL SYSTEM

Motivation

The primary objective in developing a control system suited for the aforementioned situation is to compensate for the drawbacks that would inherently arise with the use of standard input devices in rather unconventional environments. Limitations will be imposed by gloves that will naturally be somewhat bulky and cumbersome, thus limiting finger dexterity as well as the ability to accurately press a small button. Compared to other commercially available input devices, we feel that a virtual reality glove is the best candidate to be adapted to use during EVA in a pressure suit. In order to use a keyboard, the individual keys would need to be large enough to be reached without trouble from pressure gloves. If the key size were to be scaled appropriately, the overall size of the keyboard would be ungainly. A traditional joystick is limited by the number of input dimensions; generally, only two degrees of freedom are present through the manipulation of the stalk, with additional degrees provided by trigger buttons or other small actuators. Even if the additional buttons were to be scaled to be usable from a pressure glove, there simply are insufficient degrees of freedom available for general-purpose tasks.

Glove and Gesture Recognition Software

Current Work

We begin with a description of the input devices with which we are working. For this project, we have obtained two 5DT Technologies, Inc. virtual reality gloves, each with a serial interface to the host machine. These gloves have five finger sensors, as well as an auxiliary roll and pitch sensor. Each finger sensor consists of an optical fiber that wraps around the length of the digit, such that the flexion of a finger bends the fiber. In an electronics package attached to the body of the glove, a light source emits light, and photosensors observe the transmission through the fiber. The glove is calibrated based on the principle that as an optical fiber flexes the intensity of a transmitted light will vary as a linear function of the flexion. This flexion is represented as a single 8-bit byte value to the host. The roll and pitch sensors also produce 8-bit accurate results. After opening each device, we are ready to sample data.

In order to submit a sample to the processing pipeline, we must first read the data stream coming from the glove(s), and then assign the sample to the processing pipeline. To interpret the raw glove data, we utilize a vendor-supplied library function that returns the actual value measured by the glove hardware. Since our project was designed to have the option of using multiple gloves, we keep track of the mapping between samples and gloves. From the input module's point of view, there is no more work to be done, and so the next sample is obtained.

Fundamentally, the system developed to interpret gestures has three major components: input filtering, gesture recognition, and device-specific output. Each of these components runs in a separate thread of execution, exchanging data through buffer queues. By multithreading the processing jobs, the overall program can process data without a dependence on the complexity of individual components.

Once a sample has been provided to the processing pipeline, the next stage is a simple exponential filter that serves to regulate noisy input data from the gloves. This stage was added after initial testing indicated that users have slightly shaky hands; after filtering, the data is much smoother and appropriate to use in a decision process.

After data filtering, logical gestures may be interpreted out of the physical data. We define a gesture to be a region of flexion for each digit. In practice, we have found that the output of the gloves can be divided into only three or four "zones", due to the fact that a human cannot repeat gestures with exact precision. Assuming that each finger was capable of producing each position independently, there are a maximum of 1024 gestures; this number is entirely too optimistic. For example, as a limitation of the design of the individual gloves that we are using, the thumb measurement only has two zones; we have also found that users are not as comfortable with intermediate positions of the thumb as with positions of the fingers. Secondly, most people cannot move their pinky finger without incurring some movement in the ring finger; in the same vein, the ring finger generally implies a movement in the middle finger. Truly independent movement is only possible for the index and middle fingers. We divide the limitations into two categories: extrinsic for those limitations such as the thumb movement that are a result of the manufacture of the glove, and intrinsic for those limitations such as the non-independent movement of the pinky finger. Extrinsic constraints may be mitigated by investing in higher-quality gear; for our purposes, however, the performance of our gloves is adequate.

Since the virtual reality gloves are measuring one of the user's primary world interaction mechanisms, there may be situations where a user does not want his gestures to be interpreted. Similarly, a user may direct his gestures toward different targets. To accommodate these requirements, we represent the gesture recognition engine as a Mealy finite state machine, with glove data driving both transitions and outputs. Glove data is monitored for transition events, and then transformed into an output value appropriate for the device. We envision a future system in which there exists a hierarchy of states such that an initial gesture selects a device to control, and then subsequent gestures navigate the state space for a given device. In our current testbed, we have only one controllable device with one interpretation of glove data; thus, we have the two states illustrated in Figure 1. This single interpretation is a "direct-drive" state; the user's hand movements are directly interpreted into motion of the target. In more advanced control layouts, this direct drive state would be a child state of a general device selection state. For our purposes, this representation is appropriate for our prototype.

If any device output is necessary, the gesture engine passes a request off to an appropriate output processing thread for the device at hand. Since both of our devices are locomotion devices, we have only one output thread. Depending on the capabilities of the controlled devices, different formats are supported. For our prototype device, we chose a basic serial format that can accommodate translational and rotational movement commands.

To facilitate the independent development of software from the underlying hardware, we opted to use a network robot hardware simulator. Called "Player/Stage" [1], this simulator is designed to allow a controller to be developed under simulation, and then use the same binary code on the real hardware. One of our group members is employed at a mobile robotics laboratory on campus, and his managers have graciously allowed us to test the gesture control system on real robots. These robots are ActivMedia Pioneer 2-AT class devices; one under glove control is pictured in Figure 2. This image is a clip from a movie that shows the range of motion of the glove control system; the rover is put through a series of maneuvers combining forward and reverse rotational and translational movement.

Future Work & Extensions

In our current prototype, there are many more degrees of freedom in the controller than the device. Since one of the future goals of the project is to have glove control over several devices, we expect to take advantage of the excess freedom. However, a complicated control system will lead to human confusion and error. To make the gesture language easier to learn and apply, we will be adding a second glove to the system. In this extension, a gesture state transition can use one or both gloves for data input.

The use of two hands will make a system easier to interact with, but also raises the concern that a user will not be able to carry or hold anything while performing a two-handed gesture. In response to this issue, a complete backup gesture system could be implemented such that no command is impossible to perform without two fully functional gloves. The two-handed state transition could serve merely as a convenient shortcut to the same destination state as a series of single-handed transitions.

To experiment with two-handed control, a rudimentary case in which the second glove controls an independent device has been implemented. This independent device is the gripper/lift combination on the front of the Pioneer robots; this device is only used to test out the capabilities of the gesture recognition system while our final project is under construction.

In the discussion regarding the hardware system of the virtual reality glove hardware, it was noted that each glove requires a serial port for communications. With our available computer resources, we were unable to run both gloves using the same host; the solution to this problem came through the previously mentioned Player robot server. Since both glove-host machines were on the same network as the robot, each could control its device independently. The point of this extension was to show that the same binary code could interpret gestures independently, as opposed to having a left-hand controller and a right-hand controller.

Although this scheme works as a proof of concept, we expect to use a host that is capable of driving both gloves simultaneously. The use of a robot server does point out a possible source of redundancy; if the gesture-based primary control system were to fail in the field, less efficient but functional alternatives could be engaged to complete the mission.

At this point it would seem prudent to discuss the field practicality of our glove control system. As noted previously, the gloves are hard wired to the serial input on a computer. This limitation is one that we are facing due to the model of gloves that we have been provided. Wireless gloves are commercially available; however they naturally are more expensive. Despite the need for the gloves to be physically connected to a computer, this does not limit their application to a laboratory setting. Our implementation of this control system will eventually entail a small, lightweight laptop that can be worn in a backpack. The glove-end laptop will communicate to the rover via wireless Ethernet. This modification will allow us to effectively simulate an actual glove control system.

MODULAR ROVER

Motivation

The rover which our glove control system will be controlling highlights another unique feature of our project. One issue that mission planners will surely face with a manned mission (as well as they do for all missions, human or robotic) is the trade-off of reducing overall cost and weight, while still sending ample equipment. Most rovers and other robotic equipment sent will be optimized for one specific portion of the mission, and will consequently lay idle for lengthy periods of time. To solve this issue, we are designing a modular research rover, which will maximize the versatility of the available equipment.

Modularity

An issue being addressed in this rover design is long term usefulness and flexibility. Extremely specialized rovers are the most logical solution when sending single unmanned missions that will only last a period of weeks or months. When humans travel to Mars, they will most likely take numerous rovers with them, as well as spare parts to help ensure their longevity. With this being the case, it only seems natural to maximize each rover to its fullest potential. The balance between specialization for a specific goal and long term usefulness leads to one conclusion.

Rovers that are sent to Mars on a human expedition will almost certainly be modular. This will allow the rovers to serve several purposes throughout the duration of the mission. The idea is that a rover chassis can be built to accept payloads with standardized connectors and control implementations. The modular system will consist of multiple bays atop of the rover into which scientific instruments will dock. There will be a connection for power, data transfer to the rover computer, and mechanical connections to secure it in place. By simply plugging in the module and connecting the latches, the new module will be ready to use. The rover chassis will provide every module with locomotion, communication back to the astronaut and habitat, and a computer to process the data collected by the module.

This modularity allows for more types of science packages than there are rovers, increasing the mission capability at a lower cost than specialized rovers. Redundancy is also increased, because a failure will most likely take place on the rover chassis and not the module, due to the complexity of the drive system and the harsh Martian environment. If the chassis fails, the module can be placed in another rover with no loss of functionality.

The key to modularity is a set of standards on which all electronic equipment will operate on. While this is not a new idea, it has not been widely employed. The need to adopt a standard is clear considering the time required to rewrite drivers and software to transfer hardware from one rover to another. One member of the Penn State Mars Society experienced this hassle while working at NASA Ames. While this leads to wasted time on Earth, astronauts will not necessarily have the luxury of time to reprogram equipment, leading to equipment laying dormant if there is a problem on the rover carrying it unless a standard is adopted. The benefits will naturally carry over to Earth when time is spent transferring hardware from rover to rover.

Rover Overview

When designing the overall rover layout and major systems, one of the most heavily influencing factors is intended usage. A smaller rover proves ideal for investigating where a human cannot, namely small crevasses and caves. On the other hand a larger rover provides more independence and flexibility as it can traverse longer distances at higher speeds, and carry a rather high amount of scientific packages. Obstacles would also prove less difficult to avoid with a larger rover. However, for the purposes of this project, we soon realized that a compromise would suit us best. A mid-sized rover can hold all of the major equipment that we need it to and accept multiple modules at once, but at the same time be small enough to necessitate modularity. Furthermore, a rover or this magnitude is the most feasible to construct, both from the point of view of cost and general ease of construction. Microrovers require working with tiny devices which are very expensive, in addition to requiring significantly more precision on the fabrication end. Large scale rovers begin to involve components from larger vehicles, such as all-terrain vehicles and even standard automobiles. Although these parts are easily accessible, they are somewhat expensive and also become very cumbersome when producing a research rover.

The rover we are constructing (sketches for which may be seen in Figure 3) is 24” long, 16” wide and the main compartment is 10” high. This compartment will sit about 8” off the ground suspended by shock absorbers on a bottom plate containing the motors and axles. With the separate bottom plate, containing the motors and axles, we are intending to isolate minor vibrations that occur as a result of driving on mildly rugged terrain. This suspension system is designed to compensate only for small jolts, and will not allow the rover to become an aggressive off-road vehicle. The wheels chosen will be 10” in diameter. While the ultimate objective of our design will be the utilization of treads, wheels may need to suffice for short-term demonstration purposes. The reason for this decision is one of availability. Extensive searching for treads has turned up only a limited number of options. Snowmobile treads are the most readily available, but their excessive width and robust construction make them impractical for our purposes. Snowblower treads are of a more reasonable scale in comparison, but tend to be largely unavailable on the internet and in large home improvement stores.

An on-board laptop will provide both computational resources and communications via wireless Ethernet. The primary function of the computer will be to receive input data from the glove-end computer and transmit vital information to microcontrollers which will directly interface with most of the hardware. Two firewire cameras mounted on the front of the rover will provide several opportunities for stereo vision applications. Perhaps the most obvious application would be for autonomous navigation. The rover would gain increased self-sufficiency with its ability to detect hazards and locate targets. The stereo vision would also provide useful in terrain mapping and related scouting missions. Once modules for the rover are developed and we begin to perform field tests, the laptop on board will serve to establish the rover as a mobile laboratory. Any samples that are collected could easily be analyzed in the field for quicker results.

PROJECT TIMEFRAME AND FUTURE WORK

Project Timeframe & Possible Extensions

By the end of 2002 designs will be finalized for the rover base, as well as plans for extensions of the glove control system. Construction will begin early in 2003 and will focus primarily on the rover base. A fully functioning mobile base, complete with on-board computing systems will be running by May of that year. In addition, this stage will see basic stereo vision applications implemented. This includes simpler tasks such as locating, and driving to, an astronaut in the field. Glove control will also be expanded to a two-glove input system that will begin to allow us increased flexibility in the control of the rover. Over the course of the 2003-2004 academic year the rover base will be completely finalized and fine-tuned, and we will begin to develop different modules that can be used in field tests.

Because this project has been built with modularity in mind, there are inevitably countless extensions which could easily be explored and implemented. The stereo vision system can be expanded to perform more advanced autonomous navigation tasks. This includes hazard detection and obstacle avoidance, two very crucial abilities. Beyond even the rover we will have built, we can begin to explore team robotics to tackle even more varied situations. One such example would be a separate module which serves simply to deploy a microrover. Microrovers would be designed for the sole purpose of going where a larger rover simply cannot access. Since the glove control system is designed with a universal language, controlling an entire team of rovers would be a natural extension of the basic command language. While a plethora of possibilities exist, this outlines just a few of the possible extensions for this project.

Field Testing / Mars Desert Research Station

Once the mobile rover base has been completed, we will begin with very basic field tests. By analyzing its response to different situations and testing environments we may gain added insight that can be applied towards latter systems on the rover, including the individual modules. As we develop the rover and glove control system into a fully mature state field testing will obviously become a very logical step in this project. Because the focus of this project isn't so much the actual technology as it is the implementation of that technology, most of the knowledge to be gained will come from these experiments. An ideal testing situation would be at The Mars Society's Mars Desert Research Station (MDRS) in southern Utah. At MDRS, Mars Society members conduct studies in manned Mars missions from an operations standpoint [2]. More specifically, they look at many of the human factors of a manned Mars mission, including how work will be performed. A project like ours, which deals with human-rover interaction, would be an ideal test subject at the station due to its investigation of research methods. A field test at MDRS would naturally be a beneficial partnership for both parties.

REFERENCES

[1] Brian P. Gerkey, Richard T. Vaughan, Kasper Støy, Andrew Howard, Gaurav S. Sukhatme, and Maja J Mataric; "Most Valuable Player: A Robot Device Server for Distributed Control;" *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001)*, pages 1226-1231, Wailea, Hawaii, October 29 - November 3, 2001.

[2] The Mars Society; "The Mars Society: Mars Desert Research Station;" http://www.marssociety.org/mdrs/index.asp

ACKNOWLEDGMENTS

We would like to thank our sponsors and supporting groups:
The Pennsylvania Space Grant Consortium
The Pennsylvania State University College of Engineering
The Schreyer Honors College at the Pennsylvania State University
The Department of Computer Science and Engineering at the Pennsylvania State University
The Applied Research Laboratory at the Pennsylvania State University

FIGURES

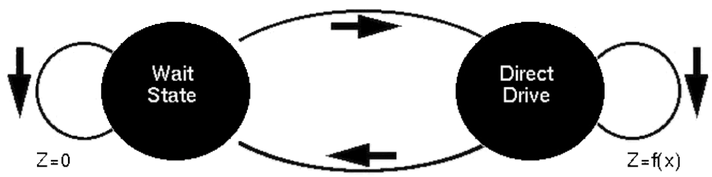

Figure 1: Gesture State Diagram

Figure 2: Pioneer Robot and Glove

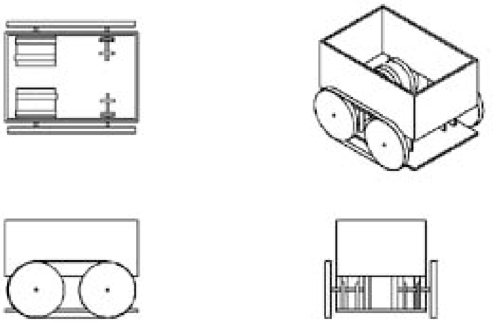

Figure 3: Rover Overview Sketches