Proceedings, 5th World Multiconference on Systemics, Cybernetics and Informatics, July 22-25, 2001, Orlando, Florida, USA, Volume VII, pp. 455-460.

Pan-tilt-verge stereo heads for mobile robots: The use of software components in driver design

Kevin Nickels and Daniel DiPaolo
{knickels,ddipaolo}@engr.trinity.edu
Department of Engineering Science
Trinity University
San Antonio TX 78212
Ph: 210-999-7543 Fax: 210-999-8037

Abstract

Mobile robots that undergo significant field-testing often have to deal with hardware and software component failures. One approach to dealing with these failures is to have multiple redundant systems available for the robot to use. By creating a common Applications Programming Interface (API) for hardware components of similar but nonidentical functionality, we have applied this approach to the test-bed for the ERA Project (Extra-Vehicular Activity Robotic Assistant) at NASA/JSC (National Aeronautics and Space Administration's Johnson Space Center).

Keywords: Mobile Robotics, Application Programming Interface, CORBA, Stereo Vision Heads, Pan-Tilt-Verge Units

1 Introduction

Mobile robots that undergo significant field-testing often have to deal with hardware and software component failures. One approach to dealing with these failures is to have multiple redundant systems available for the robot to use. By creating a common Application Programming Interface (API) for hardware components of similar but nonidentical functionality, we have applied this approach to the test-bed for the Extra-Vehicular Activity Robotic Assistant (ERA) Project at NASA/JSC (National Aeronautics and Space Administration's Johnson Space Center).

The Robotic Systems Technology Branch of NASA/JSC is currently working on the development of an ERA under the sponsorship of the Surface Systems Thrust of the NASA Cross Enterprise Technology Development Program. This will be a mobile robot that can follow field geologists during planetary surface exploration, carry their tools and the samples that they collect, and provide video coverage of their activity.

In this paper, we provide a high-level overview of the ERA system and explain in more detail the two PTV (pan-tilt-verge) drivers that were developed. These drivers expose a common generic PTV interface to the other software, and control a Metrica TracLabs Biclops PTV head and a Helpmate Zebra Vergence PTV head. The ERA utilizes the common object request broker architecture (CORBA) to modularize and compartmentalize functions. We have created two *components* (the term for software modules with a CORBA interface) to control stereo PTV units. These components share a common interface and therefore can be swapped in and out of the robotic test-bed without modification of any other components that are, for example, requesting movements of the PTV unit [6, 3]. We also compare and contrast the physical hardware and programming interfaces of the two units, and provide some rough in-situ performance comparisons.

Both the Biclops and Zebra units support two cameras and provide three degrees of freedom: pan, tilt, and verge. In the Biclops, the axes are all decoupled while in the Zebra the pan and tilt axes are coupled, requiring two motor movements for a pure pan or tilt move. The Zebra provides much larger motors and therefore higher maximum acceleration, while the Biclops provides more elegant on-board computation and a smaller, lighter, and less power-hungry overall package.

The PTV interface defines generic commands and status requests for PTV units. Relative and absolute moves in units of angular degrees, angular radians, and Cartesian meters (here, the location of the object of interest is specified) are all provided. Reasonable defaults and the ability to change the minimum and maximum velocity and acceleration are also provided. Along with initialization and shutdown procedures and status functions, this has been shown to provide a functional set of operations for the target application.

For the purpose of stereo tracking of a moving object (in this case, a suited astronaut) the generic PTV interface has been shown to be sufficient. Preliminary performance measurements indicate that for normal tracking applications (15-30 Hz frame rate, rover speed of less than approximately 1 m/s, and ground-based obstacles on the order of 20 cm) either hardware unit provides adequate performance. By utilizing the generic interface in all components, the PTV hardware can be replaced on failure *without any changes* in other components in the system.

2 Rover Hardware

As one end goal of the ERA Project is a rover that can follow a field geologist during planetary surface exploration and provide video coverage of his activity during movement over the planetary surface, an active vision system is required. This section describes the hardware infrastructure of the rover that serves as the base and computing environment for the PTV head.

2.1 Robot Base

The robot used as a base for the project is a modified ATRV Jr., from RWI. The wheels have been mounted on rigid extensions to provide adequate ground clearance, and a tower has been added to the top for the stereo vision hardware. Figure 1 shows a cartoon of the rover.

Besides the PTV head(s), the rover has a magnetic compass, an Inertial Measurement Unit (IMU) that includes accelerometers and gyro-

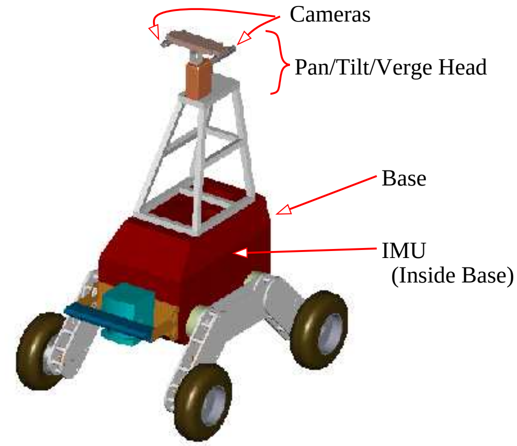


Figure 1: The EVA Robotic Assistant testbed

scopes for each axis, two frame-grabbers, two cameras, a global positioning system receiver, a sound input/output card, and a wireless Ethernet connection.

2.2 Computing Environment

There are three computers on-board the ERA base: one dual-processor Pentium II running Linux tasked to handle the vision system, one single-processor Pentium II running Linux tasked as an executive, and one single-processor Pentium Pro running Windows NT tasked to image processing. The long-term plan is to eliminate the Pentium Pro and migrate the image processing code to the dual-processor Linux machine.

2.3 The Biclops PTV head

One ERA team member at NASA is also a developer of the Biclops PTV head. As a result, the Biclops head used for these tests was one of the first functional prototypes. This hardware has been since upgraded to a production unit.

The Biclops used for this project is one of the first functional prototypes for the design. The Biclops is a much smaller head than the Zebra, having dimensions of 155mm high by 160mm wide by 101mm long, and having a mass of 1.1kg without cameras attached. It uses a relatively standard cabling interface. The input to the Biclops PTV is a simple RS-232 serial port. The motors run off a 24V power, while the internal logic requires

7.5V. The 24V supply is reduced internally, necessitating only a single 24V external supply. Biclops may draw as much as 1A (ampere), with the motor pulling a maximum of 750mA and the logic circuitry pulling a constant 300mA.

The pan and tilt axes on the Biclops are driven by belts that are independently controlled by an servo motor. The verge axis uses a metal spring which is drawn in and out by a motor-driven screw threaded through it. Each motor has an accompanying encoder that it uses to determine its position, velocity, and acceleration. The motors are independently controlled, with each having a PIC Servo Controller dedicated to it. The PIC controllers are in turn commanded by Biclops firmware that accepts commands received on the built-in RS-232 serial port.

2.4 The Zebra PTV head

The Helpmate Zebra PTV head had been used by the stereo image tracking team in previous projects, and because of the team's familiarity with the head and all of its capabilities (as well as the several Zebra heads available for this project), this PTV head was also selected as a candidate for the ERA project. This head is physically much larger than the Biclops head, and it is also shaped much differently. The Zebra PTV head consists of a tall base topped with a wide head. The base dimensions are: 3.875in. long by 2.25in. wide by 9.5in. high. The head itself is 8.75in. wide, however, greatly adding to the overall dimensions of this PTV head.

The communications and power scheme is similar to the Biclops. The Zebra communicates via a standard RS-232 serial port and requires special power conversions to supply both logic power and motor power to the PTV head. Instead of a small inline connector adapter like the Biclops, the Zebra has an additional controller box provides a manual motor power-on control, an external kill switch, and divides 24V input power to 5V logic power. The current requirement is a bit steeper for the Zebra PTV head, 1.5A for the logic power and up to 6A for motor power.

The motor configuration of the Zebra leads to a pair of coupled axes - the pan and tilt axes. This means that any motion in the pan axis will cause a corresponding motion on the tilt axis. This has to be compensated for in the control of the PTV head.

Table 1: Specification Summary

	Biclops	Zebra
Range of motion		
Pan	$\pm 165^o$	continuous
Tilt	$\pm 60^o$	90^o
Verge	0 to 11^o	0 to 26
Max velocities		
Pan	120^o/sec	360^o/sec
Tilt	120^o/sec	270^o/sec
Verge	10^o/sec	35^o/sec
Max accelerations		
Pan	$300^o/\text{sec}^2$	$8264^o/\text{sec}^2$
Tilt	$300^o/\text{sec}^2$	$8264^o/\text{sec}^2$
Verge	$50^o/\text{sec}^2$	$17611^o/\text{sec}^2$

3 Hardware Comparison

The basic tradeoff between these two heads is power and weight versus torque and thus acceleration. The Zebra has significantly more mass and volume than the Biclops, particularly if the external controller box is included. However, it also has significantly higher maximum acceleration.

4 Tasks for the Vision System

The stiff suspension and skid steering that characterize the base tend to exacerbate the vibration seen in the image that is induced by ground-based obstacles. However, it is not necessary for the PTV head to correct for all vibration. In this section, we review the major tasks that have been proposed for the vision system.

4.1 Object Tracking

The primary task of the stereo vision system is to track a suited astronaut to provide video coverage of his or her activities. In order to track the image of the astronaut, the vision system needs to be able to identify ego-motion (the motion in the image induced by the movement of the camera in the world) as well as object motion (the motion of the tracked object in the world). The PTV head, however, does not need to be able to perfectly cancel out all image motion. It has been shown to be sufficient that the relevant image motion is known [5].

4.2 Gesture Recognition

Given that the astronaut is being tracked by the cameras attached to the PTV head, software has been developed that will allow gestures to be used to communicate with the robot [4]. While this was fully implemented in the PRISM system, this functionality has not to date been ported to the current EVB testbed.

4.3 An Aid to Navigation

A secondary task that has been proposed for the vision system is to be able to track stationary objects in the scene. This would allow the rover to determine (independent of motion of the astronaut) the movement of the cameras in the world, and therefore of the rover itself [1].

5 The PTV Implementations

The ERA project requires that many other software modules of the project be able to communicate with the PTV head as well as one another. To do such things, it was decided that using CORBA (Common Object Request Broker Architecture, a communications protocol) [2] was to be the way of interfacing all of the components together. So, it was necessary to design drivers for the heads that would be able to receive CORBA calls, interpret them appropriately and then issue the necessary calls to the PTV head in the head's own language via Linux serial port commands. One advantage to a standard PTV CORBA API is that modules can communicate with either of the PTV heads in the same fashion. So, to any CORBA clients, it is transparent as to which PTV head you are actually communicating. Figure 2 illustrates this concept.

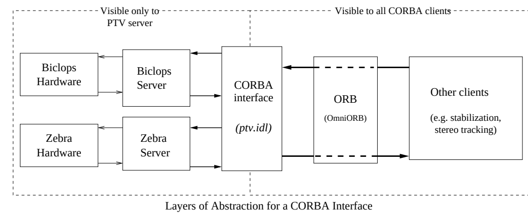


Figure 2: Interaction of PTV with other clients

5.1 CORBA Basics

CORBA is the proverbial "glue" that unifies all of the modules of this project into a cohesive unit. Using CORBA and its services, the ERA project can successfully run a number of different servers that interact with one another on different computers running different operating systems. CORBA is a layer of abstraction that sits on top of other layers that handle things normally important to servers: initialization, requests, exceptions, etc. A centralized broker (the ORB, or Object Request Broker) receives requests for service, identifies a server that can fulfill the request, and sets up communication between the server and requesting client. After this initialization, the ORB does not interfere with the actual communication.

A CORBA server is defined by the IDL (Interface Definition Language) file of that interface. To establish the transparency described above in the case of the multiple PTV servers, each server simply specifies the same IDL to advertise its interface.

5.2 The Biclops Implementation

The Biclops PTV has an on-board Z180 based processor that directly controls the servos. An API is exposed to the user for access through a standard RS232 serial port. This API has predefined interfaces for status, mode changes (coordinated move, uncoordinated move, or set velocity, or stop), reset and initialization routines.

In addition to translating PTV motion requests and formatting them to pass them along to the Biclops, the PTV CORBA server also provides some additional functionality, such as a "look at" (move to a position such that the cameras will point to a position specified in Cartesian coordinates), moves specified in either radians or degrees, and moves relative to the last commanded move. All additional functions are implemented in software and issued as a series of calls native to the Biclops PTV.

5.3 The Zebra Implementation

The on-board processor of the Zebra is an LM629, a much simpler processor than the Z180 used above. The native functionality is similarly reduced. The native API includes functions to set the home position, set trajectories and gains, and begin coordinated moves. The PTV CORBA server for the zebra also implemented in software extensions to the native API.

5.4 Comparison of PTV Servers

With the exception of the "look at" command, the majority of the functions provided by the generic PTV server are subsets of or simple translations to the vendor-specific native API. The advantage of isolating the client issuing the motion commands comes when there is a failure in the hardware or software, and the PTV heads need to be swapped out. In this event, the clients not only don't need recompiling, they can actually remain running.

6 Performance Evaluation

Once the drivers for each of the heads was completed and ready to be used on the ERA robot, the performance of the hardware in conjunction with the CORBA server was evaluated. Keeping in mind that the purpose of these PTV heads is to be used for the stereo tracking software that is currently being developed for this project, certain things are of greater importance in terms of performance. The main traits that were tested were latency and smoothness of move.

6.1 Latency

An important characteristic of any proposal to use a networked client/server implementation is the latency, specifically that caused by the added overhead of the system over, for example, single multi-threaded application with one thread issuing move commands and another issuing serial commands to the (local) PTV head.

The hardware configuration for these tests included the PTV head mounted on the rover with the appropriate PTV server running on the rover computer. The test client that issued the commands was running remotely, across one wireless hop. No explicit record of the latency between the client issuance of a request and the server's receipt of that request was made. This latency would be primarily due to the wireless Ethernet.

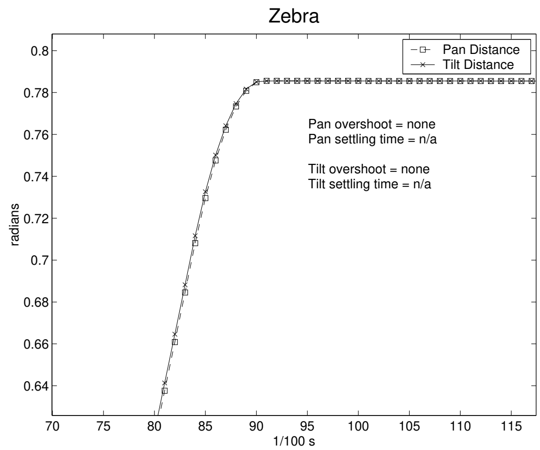
Due to software development issues, timely status information from the PTV was not available for this test. Therefore an external trigger was used, with a "human in the loop." That is, a person watched the head and pressed a key when the PTV head began moving. Even with a wireless network connection setup, the heads predictably respond much more quickly than a person can react and press a key. The fastest response that was measured was approximately 200ms for each head.

6.2 Smoothness of move

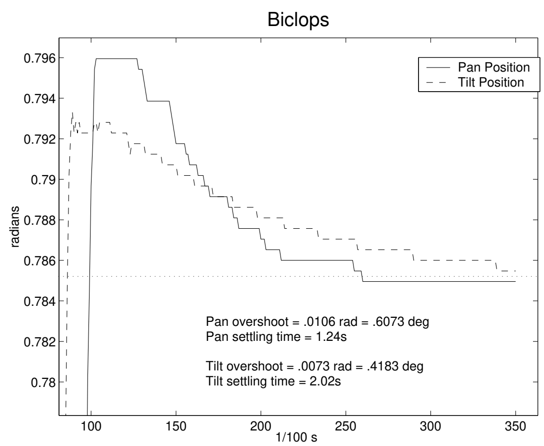
For the purpose of this project, a smooth move is a very big plus. With smoother moves, tracking is much easier and much more efficient. The test designed to measure the smoothness of each head's moves was that each head was to be set to move at a fairly high speed, and that it would make a significant move at that speed while we log the position of the head and then analyze it - computing overshoot and settling time for each, and then comparing them.

The setup for these tests was identical to the latency tests above. As mentioned in the description of the test, this test was supposed to use a high-speed move. However, the Zebra server was not to the point where the velocities were easily modified, so the default (i.e. lower) velocities were used. For this test, only the pan and tilt axes were measured for overshoot on a move from home to pan and tilt angles of 45 degrees.

As seen in Figure 3, there was no measurable overshoot for the Zebra due to the slower maximum velocities. The 0.6^o pan overshoot and 0.4^o tilt overshoot translates to 3.2cm horizontal shift and a 2.2cm vertical shift for an object 3m from the camera (a typical value for our application.)



(a) Biclops PTV



(b) Zebra PTV

Figure 3: Smoothness of Move Tests

7 Conclusions

We have motivated and presented the design of two server components for control of PTV stereo vision heads. These components are implemented in CORBA and have been shown to acceptably perform their task of isolating the task of controlling the PTV heads from a client. There are significant tradeoffs between the Biclops and Zebra Vergence hardware. The Zebra is heavier and requires more power to run, but can theoretically yield much faster acceleration and velocity. The Biclops is a lighter and less power-hungry unit, but has lower performance numbers. It has been seen that either head is acceptable for the proposed uses, and the software presented make it quick and easy to change between the two hardware units.

References

[1] D. Ballard and C. Brown. *Computer Vision*. Prentice-Hall, 1982.

[2] M. Henning and S. Vinoski. *Advanced CORBA Programming with C++*. Addison-Wesley-Longman Inc, 1999.

[3] E. Huber and D. Kortenkamp. Behavior-based approach to active stereo vision for mobile robots. *Engineering Applications of Artificial Intelligence*, 11(2):229–243, April 1998.

[4] D. Kortenkamp, E. Huber, and R. Bonasso. Recognizing and interpreting gestures on a mobile robot. In *Proceedings of the National Conference on Artificial Intelligence*, volume 2, pages 915–921, August 1996.

[5] K. Nickels and E. Huber. Inertially assisted stereo tracking for an outdoor rover. In *Proc. IEEE International Conference Robotics and Automation*, 2001.

[6] G. Wasson, D. Kortenkamp, and E. Huber. Integrating active perception with an autonomous robot architecture. *Robotics and Automation Journal*, 29:175–186, 1999.