Attitude and Position Control Using Real-Time Color Tracking*

David P. Miller, Anne Wright and Randy Sargent
KISS Institute for Practical Robotics
Reston, Virginia USA
Voice: 703/620-0551 FAX: 703/860-1802
dmiller@kipr.org, anarch@kipr.org, rsargent@kipr.org

Rob Cohen and Teresa Hunt
Space Systems Laboratory, UMD College Park
College Park, MD USA
rcohen@ssl.umd.edu, tahunt@ssl.umd.edu

Abstract

A variety of sensors and positioning methods have been developed over the years. Most methods rely on active sensors (such as sonars or lasers) which have range and power restrictions, or rely on computationally complex (and often slow) methods such as recovering position from stereo or optical flow. This paper describes a system that can determine a robot's position and orientation, in all six degrees of freedom, relative to a simple passive target. The system can determine its position relative to the target from a single image frame, and process the image and calculate the position faster than a camera's frame rate (60Hz). It uses a standard, uncalibrated color video camera as its sensor. The system runs on an inexpensive microprocessor (a Motorola 68332) leaving many cycles left over. By careful design of the target and the vision processing hardware and software, we are able to selectively perceive only the relevant parts of the target, greatly simplifying and speeding all the necessary computations. The system is being developed for autonomous spacecraft station keeping and docking.

1 Introduction

There are many applications where it is useful to have a robot that is capable of positioning itself in a specific place relative to something in the environment. When talking with someone, you want to be next to and in front of and facing that person; when picking something up, you need to be able to go to that object and approach it from a graspable direction; when leaving a room we prefer that our robots find the doorway rather than heading through the walls at an arbitrary location. For some of these applications, such as conversing with a person, the robot needs to be able to recognize a complex (and not easily modelled in detail) object in the environment. For other applications, such as exiting through a doorway, the object in question is usually quite stylized. In many instances, the doorway is marked with a distinctive colored *EXIT* sign placed nearby. Here the trick is picking the stylized object out of a cluttered environment.

The domain for this work is similar in many ways to finding a doorway in a room. The problem we are trying to solve is to be able to position a free flying robot anywhere in the workspace around a target vehicle on which it is performing a maintenance or docking task. We have some leeway in our ability to mark the target object and the sensing system we use, however both the sensing system and any target markings we add should work in a vacuum or underwater, be temperature invariant, require no power, occupy no space, have no mass, and must not interfere with the dataflow or communications system of either the robot or the target.[1]

The robot in question is a teleoperated free flying robot. Our eventual goal is to provide automated docking, and stationkeeping for the Ranger telerobot. The experiments described below were performed on SCAMP [1, 4], a "free flying" neutrally buoyant robot. Unlike Ranger, SCAMP has no manipulators or docking mechanism. It is instead a free-flying camera

[1] These are the standard requirements for all spacecraft subsystems.

platform. Its task was to maneuver to a specified position and maintain that position and pose, even when peturbed. The results of these experiments are described in section 6.

The remainder of this paper describes the target system used, the SCAMP robot, the color tracking hardware and software, and the experiments performed.

2 Target Design

A docking target ideally can orient a robot in all six degrees of freedom. This provides the robot with the feedback needed to completely and unambiguously plan its trajectory. In practice, precisely calculating the complete position and orientation of the robot is not necessary. Range information need be known only approximately, as some contact speed is usually needed to engage the docking mechanism, and docking speeds are normally quite low. Depending on the specifics of the docking mechanism, some error in rotation and translation can also be tolerated, as can errors in pitch, roll and yaw. What is most important is that the errors be reduced to acceptable limits as the robot comes closer to the final docking position. The docking target and the target sensor must work together in this way.

For this project, the primary sensor is a color tracking system (described in Section 4). This system can track the positions of three color blobs, and returns the image plane coordinates of the centroids of those blocks [7]. To reduce dependence on getting precise position and orientation information out of the sensor, the sensor is part of a reactive feedback loop that controls the robot's motion [3, 6].

The target system we are using consists of three colored 'spheres' (L, C, & R) [5].[2] L & R are separated by a distance of two meters. The middle sphere is mounted on a post with a length of one meter. The post is mounted at the center of the line segment defined by the other two spheres (see Figure 1). There are no requirements on the background behind the target except to avoid the target colors.

This arrangement makes it trivial to roughly calculate the robots position and orientation relative to the target. If the target coordinate system is set up with L & R at $(0, -1, 0)$ and $(0, 1, 0)$ respectively then the robots roll is found by calculating the slope (in the Y-Z plane) of line $\overline{LR}$. Assuming the robot is on or near the X axis means that pitch and yaw can be determined by the position of sphere C relative to the

[2]The objects in the actual target are more cylindrical than spherical. The target uses colored plastic beach pails as the colored 'spheres'.

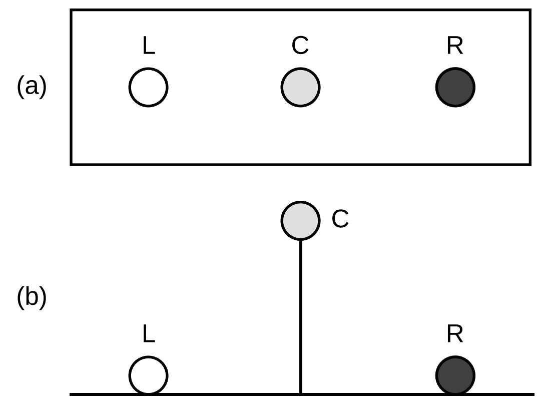

Figure 1: Target Arrangement as Seen From Different Views

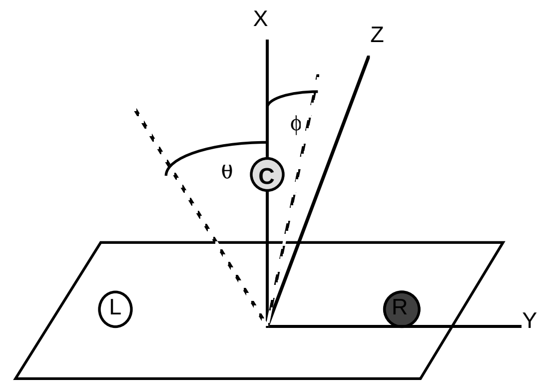

Figure 2: Definitions of θ and ϕ

center of the image. The robot's angular displacement off the X axis, above or below the X-Y plane can be approximated by

$$\phi = \sin^{-1} \frac{2\overline{cC}}{\overline{LR}}$$

where $\overline{cC}$ is the distance in image coordinates from ball C to the line defined by L & R and $\overline{LR}$ is the distance in image coordinates separating L & R (see Figure 2). Displacement off the X-Z plane to the left or right of the Z axis can be determined by comparing the lengths $\overline{LC}$ and $\overline{CR}$. If the angle is greater then $\pi/4$ then the balls will appear out of order. If the displacement is less then $\pi/4$, the angle can be calculated as:

$$\theta = \sin^{-1} \frac{\overline{CR} - \overline{LC}}{\overline{LR}}.$$

An approximation of the distance from the target can be gathered by comparing the apparent separation of L & R with the separation found in a test image at a known distance. The exact function of image plane separation as a function of distance is dependent on the particular optics of the camera, but the function is roughly inversely proportional. Of course this measurement is only accurate when the robot is on or near the XZ plane. Thus if $\theta < \frac{\pi}{4}$ then the distance r can be calculated as:

$$r = k_1 + \frac{k_2 \cos\theta}{\overline{LR}}$$

.

3 SCAMP

The Space Systems Laboratory (SSL) at the University of Maryland studies how people perform useful work in weightlessness, how machines operate in weightlessness, and how the two can work together. The Supplemental Camera and Maneuvering Platform (SCAMP) was designed and built by the SSL, with support from the NASA Telerobotics program, to study further the use of robotic machines in space. This telerobotic research uses the neutral buoyancy environment to simulate the weightlessness of space.

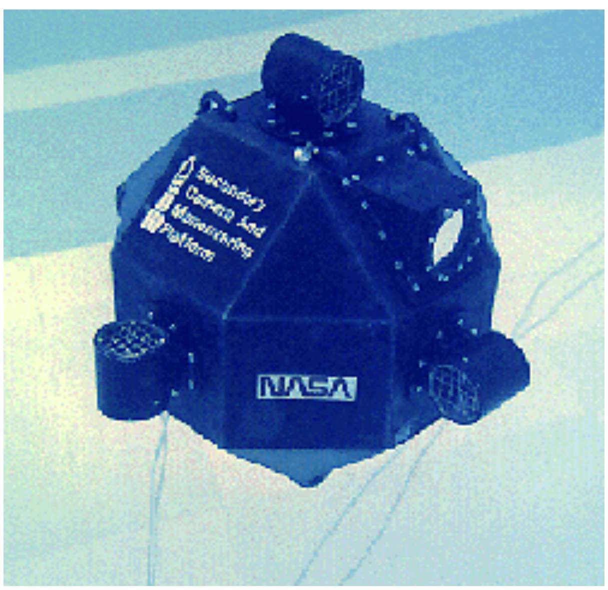

Figure 3: The SCAMP Robot

SCAMP's original design goal was to provide a free flying, independently positionable video source and to function as a supplement to fixed video sources during telerobotic operations. SCAMP's free flying capabilities provide an operator with views of a worksite unattainable by fixed base cameras. As SCAMP was used in day to day operations, its role expanded to include structural inspection as well as documentation. During these operations, the operator was responsible for positioning SCAMP to keep objects of interest in view. This imposed a considerable burden on the operator, especially if the object of interest was in motion. Previous experiments with SCAMP [1, 4] have shown the need for more autonomous behavior. This would reduce operator workload and allow the operator more freedom to perform other tasks. Since the primary data returned from SCAMP is video, closing a control loop around this data stream would help reduce the operator's workload.

3.1 SCAMP System Overview

SCAMP, shown in Figures 3 and 4, is a twenty-six sided solid. SCAMP measures 28 inches (71.1 cm) in diameter and weighs 167 pounds (75.8 kg) in air. Six ducted fan propellers, or thrusters, each located on a primary face, provide propulsion. There is a dedicated, closed loop motor controller for each thruster connected to the on-board computer, a Motorola 68HC11. This computer communicates with the control station via a fiber-optic link using a message-based, serial protocol developed by the SSL

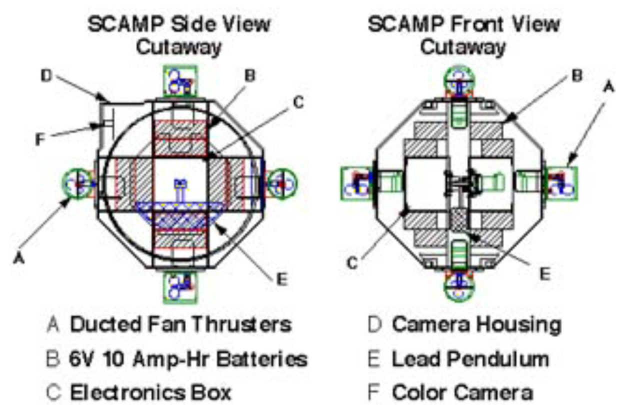

Figure 4: Cutaway view of the Supplemental Camera and Maneuvering Platform

The control station for SCAMP is built around a Macintosh computer. This computer is responsible for accepting operator input, displaying data from SCAMP and communicating with the vehicle. The operator receives feedback from SCAMP and sends commands to SCAMP through the standard Macintosh user interface. The primary form of feedback from SCAMP, however, is video.

The control station has two, 3 degree of freedom (DOF) hand controllers that allow the operator to position SCAMP. One hand controller controls translation and the other controls rotation. The hand controller signals are transmitted to a very simple, yet robust control system installed on SCAMP that commands the thrusters to execute a trajectory as specified by the operator.

The pitch control system on SCAMP consists of a lead weight on the end of an arm (a pendulum) connected to a drive train. Driving the pendulum to a desired position causes the vehicle to rotate, thus changing the pitch angle. This controls the center of gravity (cg) offset about the pitch axis. This cg offset precludes rotating SCAMP about the roll axis. An optical encoder located on the pendulum drive motor measures the pitch angle. Thus, the angle of the pendulum, relative to the vehicle body, is known. Since the pendulum always points down, the pitch angle is known in the neutral buoyancy tank coordinate frame also.

Along with the encoder on the pendulum motor, there are two other sensors on board that allow some closed-loop control to be implemented. A depth sensor provides the capability of closed-loop depth hold, while a rate gyro generates feedback about the pitch rotation rates. These two sensors, in addition to the pendulum provide position feedback relative to the tank frame but SCAMP cannot fully determine its position in the tank. Information regarding X, Y, and yaw is unattainable, given the current sensor package.

SCAMP's on-board sensors are necessary for it to operate effectively in the neutral buoyancy environment, but the real payload is the camera. The on board camera is a fixed-focus, fixed-zoom color camera. The output from the camera is a National Television Standards Committee (NTSC) composite video signal which is converted to a fiber optic signal, transmitted to the surface and converted back to an electrical signal and distributed.

4 Vision Processing

In order for SCAMP to align itself to the target described in Section 2, SCAMP must be able to quickly track the positions of three colored balls. To accomplish this, we use the Cognachrome video processing system [7] (shown in Figure 5).

The Cognachrome vision processor is a tiny, lightweight, low-power system optimized for high-speed, low-latency tracking of colored objects. Because it is specifically designed for this purpose, it is able to outperform more general systems and use less hardware on simple tracking tasks; as are needed in spacecraft docking [9]. For this application, we have artificially slowed the Cognachrome's frame rate to avoid stressing the robot's thrusters with high speed oscillations.

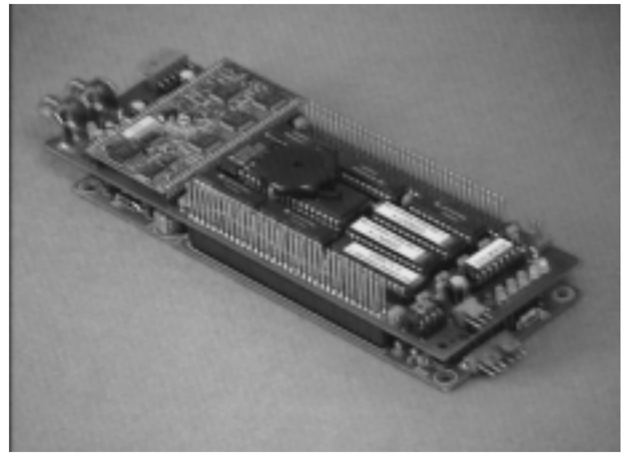

Figure 5: The Cognachrome Hardware

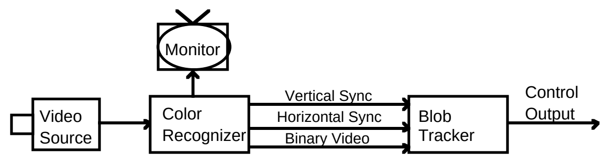

Figure 6: Block Diagram of Cognachrome 2000 organization

4.1 System Organization

The Cognachrome consists of two logically and physically distinct units – the color recognizer and the blob tracker. Figure 6 shows the block diagram for the system. The color recognizer inputs standard color NTSC video, and outputs a corresponding binary image and synchronization signals. The blob tracker captures these signals as timing information, decomposes the image into a number of spatially distinct "blobs," computes tracking information for the blobs, and takes appropriate action, such as transmitting navigation instructions over a serial port.

4.2 Performance Specifications

The primary performance specifications for this system are resolution, frame rate, and latency. The spatial resolution of the tracking data is 256 rows by 250 columns. The frame rate of NTSC video is 30 or 60 Hertz depending on whether interlace is used or not. Most video cameras do not actually use interlace, and therefore provide frames at 60 Hertz, or 16.7 ms/frame. The frame rate of the system depends on how long end of frame processing takes. Converting from endpoint to blob data typically takes 4 ms per frame and happens in parallel with frame acquisition as data is copied from the TPU (timer processing unit) RAM to the processor RAM. Once the end of the frame is reached any remaining data conversion is completed, the blob data is processed, and control action is taken. If this end of frame processing takes 12 ms or less, the effective frame rate will be 60 Hz. If the processing takes longer, frames will be missed and the frame rate will decrease. The processing time available is 28 ms for 30 Hz, 44 ms for 15 Hz, etc. The latency calculations are very similar. The length of time from the beginning of the frame to the completion of the control task is 16.7 ms plus the end of frame processing time. The processing tasks attempted so far have typically fallen within the 12 ms processing window, and therefore have a 60 Hz frame rate and have 17 to 29 ms of latency. The frame rate is artificially slowedto avoid high speed oscillation of the thrusters.

The Cognachrome boards stack together for a total size of $2.5'' \times 6.25'' \times 1.25''$. The primary means of communicating with the processor is over an asynchronous serial port. The processor runs a commercial real-time operating system, ARC [8] for embedded controllers, and provides the control signals for the color recognizer as well as processing the images and using the processed data to provide control output.

5 Vision-Based Positioning for SCAMP

As described in section 3, SCAMP has 6-DoF and uses thrusters for all except pitch. A motorized pendulum is used to control pitch.

Because of SCAMP's design, roll and pitch are linked. It is not possible to maintain a roll angle because of the pitch pendulum. For similar reasons, a pitch error cannot be maintained unless the pendulum is moved. For these reasons, SCAMP really only needs to be controlled in 4-DoF. Because of these self-correcting properties in roll and pitch there is never any left right ambiguity regarding the target. To simplify our task we made the left and right target colors identical.

Coordinate systems are defined as follows:

- The target defines an absolute coordinate system with the base post of the center sub-target at the origin. The X-Y plane is defined by the three sub-targets. The X-axis is defined by the center target mounting post and is positive in front of the target. The Y-axis is defined by the line connecting L & R and is positive moving from L towards R. Z is positive going up.
- The vehicle coordinate system has x moving forward, y to the right, z down; positive yaw is a clockwise rotation when viewed from above.

For these experiments we only needed to control 4-DoF on SCAMP. The **only** sensor that was used was the visual tracking of the target. Therefore the most important step in having SCAMP maintain or move into position was to first visually acquire the target. We used a standard behavior approach, which has worked well with underwater vehicles in the past (e.g., [2]). We controlled SCAMP's movements using the following behaviors:

1. If the center target was in view, yaw was adjusted to center the target and z was servoed at all times to keep the target in frame.
2. If the center target is not in view, then Yaw and Z are powered open-loop along a vector opposite to the vector of the target as it was last viewed in frame.
3. If center and side targets are visible, adjust Z, Y and X to desired positions (using the z,y, and x thrusters) where:

$$r \approx k_1 + \frac{k_2 \cos\theta}{\overline{LR}}$$

$$Y \approx r \sin\theta$$

$$Z \approx r \sin\phi$$

$$X \approx \sqrt{r^2 - Y^2 - Z^2}$$

   k_1 and k_2 correspond to the slope and intercept of the function plotting separation of the L and R sub-targets in image coordinates against distance.

4. If side targets are not visible and the blob size of the center sub-target is 'large', move the robot away from the target by driving it -x while adjusting z and yaw to keep the target in view.

6 Experiments

The experiments with SCAMP were performed in the University of Maryland Neutral Buoyancy Test Facility – a water tank fifty feet in diameter and twenty-five feet deep.

For the purposes of these experiments, the Cognachrome hardware is located on the deck where it receives the video signal from SCAMP. The serial line from the Cognachrome is input into the control station where it issues the same command packets as are normally generated by the control station. A software switch on the control station can switch control between the joysticks on the control station and the Cognachrome.

The target simplification (described in section 5) was useful because it brought down the target colors from three to two. Doing underwater tests, it was difficult to find distinct colors that were not radically effected by the changing 'bluing' effect of the varying water column between SCAMP and the target.

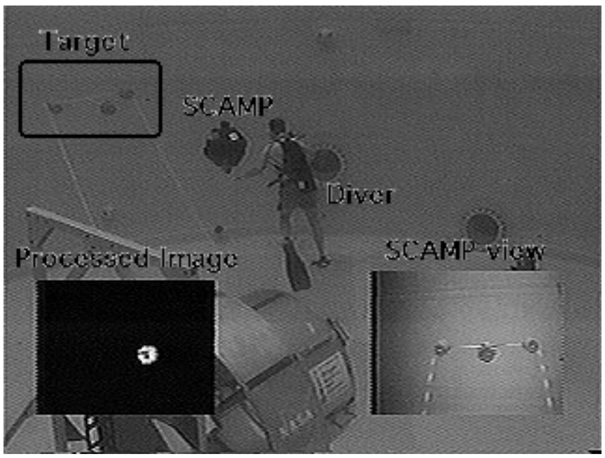

Figure 7: Experiment Setup with Display of 'Pink' Processing

After a series of tests, we found that both bright pink and bright yellow could be tracked across the entire testing area. These colors were not normally found in the test tank (or in space). Pink was used for the center color and yellow was used for the left and right 'spheres'.

Figure 7 shows the experimental setup. The Cognachrome system can track up to three distinct colors simultaneously, however, the debugging video output only displays the tracking data from a single color at a time. We have set the debugging display to show the center target on one frame and the left and right targets on the next frame. The debugging display thus alternates between displaying the two colors it is tracking at 60Hz giving a good visual image of both. The frame shots here, however, show one color at a time. The display showing the left and right targets is shown in Figure 8.

During these tests SCAMP was directed to a variety of [X,Y,Z] positions in front of the target. Once it had achieved the desired position, a diver would perturb the robot and the control system would attempt to regain the position. In the most recent set of tests SCAMP was peturbed 24 times, 14 of which caused the target to go out of view of SCAMP's camera. Twenty-two times SCAMP recovered sucessfully. The

Figure 8: SCAMP with 'Yellow' Processing of Image

two failures both had significant perturbation vectors added after the target was lost from view (e.g., the diver would rotate SCAMP and then push it towards the bottom of the tank). In some instances, The visual system was able to recover from these multiple pertubation tests, but that was pure luck. Since all of the position and orientation information comes from viewing the target, the robot's control system has no way to know that a peturbation vector, that is added after the robot has lost sight of the target, has occurred.

Future tests will be performed using the Ranger NBV vehicle which has rotation rate sensors. On Ranger, these secondary perturbations should not be a problem.

6.1 Conclusions and Future Work

Using color tracking for target docking has certain advantages over more traditional space tracking systems. Since the system is passive, it is much lower power then a ranging system. The targets are simple and inexpensive. The tracking system is also inexpensive and has a minimum of impact on the other spacecraft systems.

The experiments described above have shown that a complex vehicle can be controlled using a simple selective vision system. The color tracking has proven robust even when faced with selective frequency absorption from the water in the tank.

In the next year this system will be ported onto the Ranger NBV robot and we will use it to guide the robot through docking maneuvers. We hope this will demonstrate the system's applicability for an actual flight mission.

Acknowledgements

The authors would like to thank Dave Lavery, Dave Akin, and the staff of the SSL for their support of this work and assistance in conducting the tests. Cathryne Stein provided useful feedback on earlier drafts of this document. This work was supported by a University of Maryland contract #Z628901.

References

[1] D.E. Anderson, C.A. Buck, and R. Cohen. Applications of Free-Flying Cameras for Space Based Operations. *SAE/AIAA International Conference on Environmental Systems* Friedrichshafen, Germany, June 20-24, 1994.

[2] Peter Bonasso. Underwater Experiments Using a Reactive System for Autonomous Vehicles. *Proceedings of the 1991 National Conference on Artificial Intelligence*, AAAI pgs 794-800, July 1991.

[3] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, March 1986.

[4] R. Cohen and D. Akin. The Role of Supplemental Camera Views in Space Teleoperation. *Proc. Teleoperation '93*, 1993.

[5] Miller, D.P. and Wright. A., Autonomous Spacecraft Docking Using Multi-Color Targets. *Proceedings of the 6th Topical Meeting on Robotics*, Monterey, CA, February 1995.

[6] D. P. Miller. Rover navigation through behavior modification. In *Proceedings of the NASA Conference on Spacecraft Operations Automation and Robotics*, July 1990.

[7] Randy Sargent, Carl Witty, and Anne Wright. *Cognachrome Vision System User's Guide*. Newton Research Labs, 1.1 edition, December 1995.

[8] Anne Wright, Randy Sargeant, and Carl Witty. *ARC Development System User's Guide*. Newton Research Labs, 0.3 edition, November 1995.

[9] A. Wright. A High-speed Low-latency Portable Visual Sensing System. *Proceedings of the SPIE symposium on optical tools for manufacturing and advanced automation*, September 1993.