URC97004

SENSORY INTERACTIVE TELEOPERATOR ROBOTIC GRASPING

Keli Alark
isis@slider.unm.edu

Dr. Ron Lumia
lumia@louie2.unm.edu

NASA ACE Center for Autonomous Control Engineering (ACE)
Department of Mechanical Engineering, Room 202
The University of New Mexico
Albuquerque, New Mexico 87131

ABSTRACT
As the technological world strives for efficiency, the need for economical equipment that increases operator proficiency in minimal time is fundamental. This system links a CCD camera, a controller and a robotic arm to a computer vision system to provide an alternative method of image analysis. The machine vision system which was employed possesses software tools for acquiring and analyzing images which are received through a CCD camera. After feature extraction on the object in the image was performed, information about the object's location, orientation and distance from the robotic gripper is sent to the robot controller so that the robot can manipulate the object.

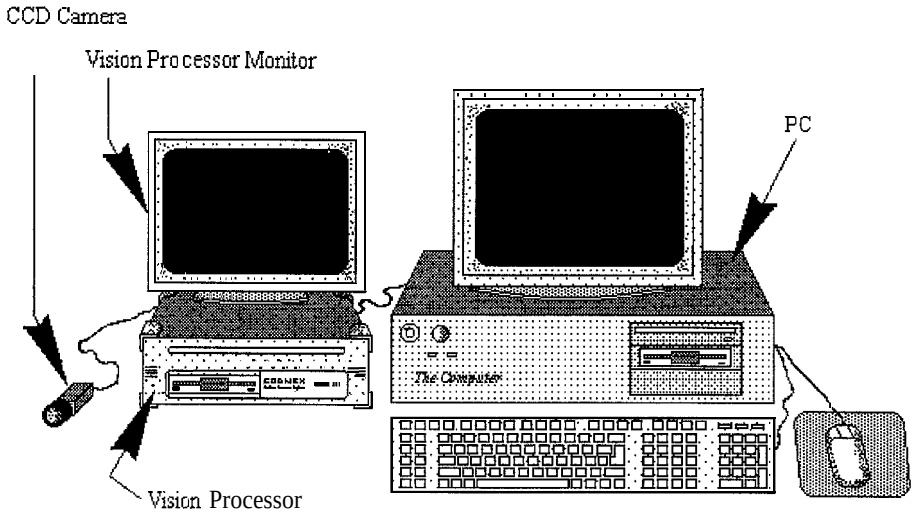
INTRODUCTION

The animal kingdom's vision and cognitive qualities has evolved to a point that sensing and reacting to stimuli in its environment are almost instantaneous [1]. As the environments of robotic systems become more complex and often dangerous, it is desired to create a system that can operate continuously in almost any environment. NASA's future goals includes the development of a permanent manned space station and the technology to insure the availability of affordable and competent satellite communications [2]. This increases the need for low-cost, low-maintenance robot vision systems that are both more mobile and more versatile. Although the processors are becoming increasingly more powerful yet more minuscule, conventional image analysis utilizes equipment that is expensive and difficult to operate. The vision system needs to be simple enough to understand in both the programming and execution phases, with minimum time expended to learn and execute each phase. The system should be compact for portability, yet the reduction in physical size of the equipment should neither increase the cost, nor decrease the capabilities of the system. However, it also needs to be versatile enough that it can be used for a multitude of different environments with minimal human operation and minimal maintenance. As a vision system is typically equipped with a camera and a CPU complete with memory and a processor adequate to analyze the data, it should be possible to locate an object, identify it by feature extraction, move to the object and manipulate it.

PROJECT-ELEMENTS

The project entailed manipulation of a pipe in an semi-random position (pipes could not be viewed on their ends) by sending the information learned from the vision system which pertained to the location, size and orientation of the pipe in real-life coordinates to the controller. Although a current trend in robotic vision is to allow camera motion in a plane or in a 3 dimensional fashion [1,3], the employed system has more thorough integration of human and computer qualities, and reduces cost. The system that was employed in this research combines a computer vision system, a robot controller, and a single CCD camera mounted to a teleoperator manipulator robot [4]. Considering that there are a myriad of vision systems to choose from, finding a system that would perform sufficiently to complete the tasks was not especially difficult. The system that was chosen was the Checkpoint 400 system by Cognex. The Cognex Checkpoint system is used for fabricating and performing vision applications such as assembly-line part inspection. The software provides the user with tools for feature extraction of the object based on the object's physical geometric properties, and its texture created from light reflection. In addition to calibration into real-world units, the software also provides operator-interface tools to facilitate the execution of the program in the workspace. Thus it was decided that although the system was not intended for precisely this type of research, it was relatively easy to program the system to performing as desired.

For the development of the programs, the system requirements include a PC with a monitor, and the CCD camera which are to be interconnected to the Checkpoint CPU Vision Processor and the Vision Processor Monitor. The PC allows the user to create and edit the text of the programs, and then debug and execute the program as necessary. After the program is completed the Checkpoint Vision Processor and the Monitor can be taken to the work-site as a stand-alone system.



There were several systems that were integrated with the Checkpoint system. Cimetrix offers an open architecture robot controller and software that allowed the application to be simulated before it was actually executed. The controller accepts inputs from the Checkpoint system in the form of data strings. Once the information from Checkpoint was sent to the controller, the Cimetrix software simulates the series of motions the robot will undergo and if the simulation proved that there were no errors, the actual robot was deployed.

Barrett Technology created a three fingered dexterous gripper. The design has one opposable finger that acts as a thumb and is un-movable. The other two fingers have a 180° range of motions. Each finger is rated for 19.62 Newtons of force that allows the Barrett Hand to grasp firmly. Its ability to grasp up to 647.46 N (especially near pivot axis of j oints) decreases the possibility that there will be injury caused to the hand or other elements of the system by a large couple-moment. The robot that is used is a Puma 560 by Unimate, which has 6 degrees of freedom.

SYSTEMATIC APPROACH TO THE ALGORITHMS

The camera functions as the eye of the system which is ultimately controlled by the operator, and as the robot moves within a workspace, the camera sends live images of the area, in real-time, back to the Vision Processor Monitor. Once the operator designates an object of interest, a still image of the object is acquired through Checkpoint which appears on the Monitor screen. The operator at the Checkpoint system then begins the algorithm which graphically extracts features of the object. The operator adjusts the Checkpoint vision tools so they highlight the features of the object, and once this is inputted, the Checkpoint computer sends data strings to the robot controller, which deploys the robot to move to the object, grasp the object, and move the object to a new location.

The Cognex Checkpoint system performs visual inspections, usually of identical or similar objects, based upon the features extracted by graphical tools pre-programmed into the Checkpoint software. The following are the steps involved in creating a repetitive recognition program. The test for the system utilized PVC pipes (due to their low density) that were either black or white:

- Define an archetype in an initial program, based on the geometric features of the object. 'This entails positioning the object in the camera's field of view, in-focus and unoccluded. The programmer accepts the acquired still-image of the object.
- The programmer decides what are the best tools to accurately and consistently locate the object and calculate the necessary information,
- With each test, edge tools, which identify edges based on 1 ight and dark pixels, located opposite edges of the pipe and two lines tools were placed down either profile of the pipe. In this application, the

angle between the lines was measured to determine the orientation of the pipe, and as the diameter of the pipe was known. the diameter of the pipe on the screen would vary with distance and angle from the camera.

- Code is added to calculate the location, orientation and position of the object from the robot.

Below are the steps used each time the program is executed:

- The robot is moved in order to allow the camera to scan the area for suitable objects as defined above.
- The executable program prompts the operator to designate an object.
- Checkpoint displays the tools that were used in the program on the Vision Processor Monitor.
- The operator moves only the edge tools defined in the initial program to accommodate the new position of the object.
- The program configures the remaining tools which are dependent on the edge tools and then displays al I the chosen graphical took that were used to extract the features of the prototype on the screen.
- The data from the vision tools are calculated, displayed on the screen, and sent in data strings to the robot controller.
- Cimetrix controller deploys robot to grasp and manipulate object.

CALIBRATION

Once the camera was secured to the robot, initial measurements of the camera were taken. Both the camera and the Checkpoint system were tested to determine the amount of accuracy in the measurements. The camera acquired several pictures of objects of known sizes, at predetermined distances. The lighting of the objects was kept as controlled as possible, as shadows and specular reflections created invalid data. Next it was necessary to determine the position of the center of the field-of-view of the camera so that the object could be placed as close to the center of the field-of-view as possible to avoid spherical aberration.

Then an image of the pipe was acquired. [n an ideal environment, the pipe will have no dimples or marks on it that would prevent its profile from being rectangular. As the length of the pipe increases from the center of vision, the profile of the pipe no longer seems parallel as the lines that define the pipe's profile appear to converge. The angle to which the lines converge depends on the distance that the pipe is from the camera vision sensor, the diameter of the pipe, and the angle that the major axis of the pipe makes with the axis of the center of field-of-view. The diameter of the pipe in the image was determined at two locations across the converging profile lines, the angle of which was determined by utilizing two edge tools and the line tools in Checkpoint.

The relative position between the camera and the robot was determined by a 4 x 4 matrix and a calibration square. The calibration square measured 10 x 10 cm and a picture of the square was analyzed in Checkpoint. Each of the vertices of the calibration plate in terms of the camera were sent to the controller. The controller then moved the robot to and contacted each of the vertices and the robot's joint angles at each point was recorded. All information was then entered into a 4x4 matrix and the relative position between the camera and the robot was thus determined.

ISSUES AND RESOLUTIONS

Certain precautions had to be attended to initially. The camera had to be secured to the robot in a manner so the camera would not be hit during in the extreme ranges-of-motion of the robot gripper. As pipes can vary in length, diameter and density, the BarrettHand three-fingered gripper was chosen to be certain that the robot would have a sufficient grasp on the pipes and not damage any of the equipment by manipulation. In addition, it was necessary to ensure that the pipes were cylindrical tubes with a planer view of a rectangle and the opposite ends circles. To avoid spherical aberration, it is necessary that the image of the calibration square is as close to the center of the field-of-view as possible, without being so small as to cause errors in pixel overlapping.

The camera can be calibrated in real world coordinates utilizing the screen resolution (in pixels per inch) or by using precisely machined calibration plates at known distances. All subsequent measurements (except angle measurements) are given in units chosen by the operator. Angles are measured only in degrees (although trigonometric functions use only radians).

Calibration in general is prone to errors both mechanical and human. One method of avoiding the probable errors is by removing it completely as did Yoshimi and Allen [3]. However this there is still a calibration of the camera to the robot by using the Jacobian. Lighting was an unavoidable and immense task. The information

calculated by the system can vary greatly with disturbances in lighting. Large quantities of concentrated light can create specular reflections on even optically flat surfaces. Excessively little light will produce no image.

CONCLUSIONS

[integration of the systems above have yielded a system that is extremely versatile. First, it decreases the time an operator would invest in learning and executing the programs. Second, it limits the chances for operator error by requiring only minimal inputs from the operator and all calculations are performed within the system. Third, the system is compact and relatively inexpensive. Fourth, the system can be configured for distance learning, for institutions that cannot afford the capital investment. Finally, sensory based robotic systems could be very valuable for space exploration. For example, a Mars rover must have local self-preservation skills or wait the 17-40 seconds for communication from earth.

ACKNOWLEDGEMENTS

This work was supported in part by NASA under contract # NCCW-0087

REFERENCES

[1] J.L. Crowley, 1-1, I. Christensen, Vision as Process, Basic Research on Computer Vision Systems, Springer-Verlag, Berlin, 1991

[2] National Aeronautics and Space Administration, *Job Choices in Science and Engineering: 1997*, National Association of Colleges and Employers, pp. *94-95, 1996*

[3] B. H. Yoshimi, P.K. Allen, "Active, Uncalibrated Visual Servoing". In *Proceedings, IEE international Conference on Robotics and Automation, Volume 1*, pp. 156-161,8-13 May 1994

[4]. W. E. Snyder, Industrial Robots: Computer Interfacing and Control, Prentice Hall, Inc., Englewood Cliffs, NJ 1985