REAL TIME CONTROL OF A ROBOT WITH A MOBILE CAMERA

Technical Note 179

February 1979

by: Gerald J. Agin, Senior Computer Scientist

Artificial Intelligence Center

SRI Project 4391

The work reported herein was supported by the National Science Foundation under Grant APR75-13074, and by sixteen industrial affiliates.

ABSTRACT

In this paper we describe and analyze a control system for a Unimate robot that derives its control information from a small, solid-state TV camera attached to its end-effector. Visual input may be in either of two modes: using conventional lighting or using projected patterns of light to give distance by triangulation. The number of degrees of freedom that may be controlled and their attainable accuracy are dependent on the mode of illumination, the geometric form of the image, the resolution of the camera, and the relationships between the camera, target, and projector (if used). A dynamic analysis of the system accounts for discrete delays in the control loop as well as the transfer function of the robot itself. The system has been demonstrated in several modes simulating manufacturing operations in static and moving coordinate systems.
CONTENTS

ABSTRACT .................................................. ii
LIST OF ILLUSTRATIONS ................................. iv
I  INTRODUCTION ........................................... 1
II  STATIC ANALYSIS ....................................... 3
   A. Depth Measurement by Perspective ............. 3
   B. Depth Measurement by Triangulation .......... 6
   C. The Tool ............................................. 8
III  DYNAMIC ANALYSIS ................................... 10
IV  EXPERIMENTAL RESULTS ............................. 16
REFERENCES .............................................. 20

iii
ILLUSTRATIONS

1  RELATIONSHIP OF DEPTH TO PERCEIVED SPACING . . . . . . . . . . . 4
2  DEPTH MEASUREMENT WITH PROJECTOR . . . . . . . . . . . . . . . . . 7
3  BLOCK DIAGRAM OF VISUAL-SERVO CONTROL SYSTEM . . . . . . . . . . 10
4  TRACKING AN OBJECT ON A MOVING BELT . . . . . . . . . . . . . . . 17
5  SIMULATED SPOT WELDING ON A MOVING LINE . . . . . . . . . . . . . 17
6  PATH FOLLOWING IN THREE DIMENSIONS . . . . . . . . . . . . . . . . . 19
I INTRODUCTION

Existing robot technology is clearly in need of sensory feedback to extend its limited capabilities. Visual feedback can minimize the need for jigs and fixtures and ease the required tolerance of parts. Visual feedback controlling a manipulator in real time can allow it to work on a moving line without requiring precise control of the line.

For quite some time we at SRI have been interested in the use of visual feedback to control a manipulator in real time. Our approach has been to place a small solid-state TV camera in the manipulator end-effector and use its visual feedback to guide the hand to reach a given target. We call this method "visual servoing." Visual servoing could be applied to a large variety of tasks in material handling (moving parts from place to place), inspection (for finding the object to be inspected), and assembly (aligning parts with respect to one another).

There has been little prior effort in visual servoing of a manipulator. A few experiments have been performed in the "blocks world" [1-4] where a fixed TV camera observes a robot at work. In these experiments attempts were made to place one block on top of another, or to insert a peg in an oversize hole, by carefully observing an area of the scene where the mating takes place. The real-time control aspect of visual servoing has been absent in these experiments, in which the basic method was to repeatedly alternate between taking pictures and moving.

For industrial application the approach has generally been to move the TV camera with respect to the workpiece. One experiment of this nature is the work of Hagibara et al. [5], in which the camera was rotated and translated until the perceived image was properly aligned and centered. A more recent publication by Kashioka et al. [6] describes visual servoing with both fixed and manipulator-mounted cameras.
This paper discusses our approach to visual servoing—an approach we hope is sufficiently general to allow a wide range of applications. As control is in real time, a key point in our approach is to make use of binary images to achieve fast and reliable image processing. The constraint of binary image processing forces special consideration to be given to lighting and contrast in the image, but the reward for this is fast operation. In some applications we use projected light patterns to obtain information about range or depth. The real-time nature of our servo system requires consideration of the dynamics of mechanical components and leads to questions of stability and speed of response.
II STATIC ANALYSIS

The function of computer vision in a visual servoing application is to determine the spatial relationships that exist between the camera, the tool, and the workpiece. In this section we discuss some of the possible configurations that may be useful for servoing, as well as the constraints that govern possible placement of the camera with respect to the tool and the workpiece.

A. Depth Measurement by Perspective

A camera maps a three-dimensional world into a two-dimensional image. There is inevitably some information loss inherent in this imaging process. It is impossible to determine the coordinates of a single point in three dimensions from its image; two coordinates may be measured that constrain the point’s position to a ray in space, but the position along that ray cannot be inferred from those coordinates.

For convenience in exposition we will establish some conventions with respect to positions and orientations. Let us arbitrarily affix a Cartesian coordinate system to the lens of our TV camera, with the x-axis pointing toward the camera’s right, the y-axis pointing upward, and the z-axis directed along the camera’s principal ray. From the image of a single point we may establish its x and y coordinates (given some assumption about depth), but the z-coordinate is indeterminate.

When we allow images to consist of many points, some of the information lost in imaging may be recovered. Various components of the scene may give perspective cues that permit estimation of depth—the third dimension z.

The size of any solid object's image is related to its distance from the camera, but size is also dependent on other factors such as
focus and the setting of the binary threshold. To escape this dependency on unrelated factors, we prefer to measure the distance between two points instead of the size of a single object. This distance should be measured between two points whose image locations are not dependent on threshold or focus. To a first approximation the centers of stripes, spots, holes, or similar markings satisfy such a requirement.

![Diagram](https://via.placeholder.com/150)

**Figure 1 RELATIONSHIP OF DEPTH TO PERCEIVED SPACING**

It is useful to examine the resolution and accuracy of this method to measure depth. Sensitivity to a change in $z$ is dependent upon the sensitivity of the camera/image processing system to changes in $x$ and $y$. Consider the situation illustrated in Figure 1. A pair of marks is observed at a known depth $Z_1$, resulting in some perceived spacing between the marks in the image. Since the value of $Z_1$ is known, we can make use of calibration information to calculate the actual separation $W_1$. Now suppose the marks are moved to a new, unknown depth $Z_2$. The spacing of the marks in the image will change, so that a new apparent spacing $W_2$ may be calculated at the original depth $Z_1$. We are interested in the ratio
\[ \frac{\Delta W}{\Delta Z} = \frac{W_2 - W_1}{Z_2 - Z_1}, \]

which relates sensitivity in depth to sensitivity in the other two dimensions. \( \Delta W / \Delta Z \) is related to the angle \( \phi \) as shown in the figure. In the symmetrical case shown,

\[ \frac{\Delta W}{\Delta Z} = -2 \tan \left( \frac{1}{2} \phi \right), \]

in the limit as \( \Delta Z \) approaches 0. For small \( \phi \), we may substitute the approximate relationship

\[ \frac{\Delta W}{\Delta Z} \approx -\tan \phi. \]

\( \Delta W / \Delta Z \) relates resolution or sensitivity in depth to resolution in the other two dimensions. If \( \phi \) is small, it takes a large change in \( z \) to produce a small change in \( w \).

Suppose we attempt to improve the depth resolution by use of a longer-focal-length camera lens. That would improve the resolution in \( x \), but it would also narrow the field of view, so that we would be forced to bring our two marks closer together and proportionately narrow the angle \( \phi \). The resulting depth resolution would remain the same.

A typical case is as follows: a camera with a 25 mm lens 60 cm from its target scans a scene about 15 cm square. If the two reference marks are 10 cm apart, then

\[ \frac{\Delta W}{\Delta Z} = 2 \tan \left( \frac{1}{2} \phi \right) = \frac{10}{60} = 0.166. \]

This means that depth resolution is only 1/6 as fine as lateral resolution. With a 15 cm field of view and a 128x128-element camera, the standard error of position measurement \( [7] \) is

\[ \frac{\Delta W}{\sqrt{24}} = \frac{15 \text{ cm}}{128} = 0.24 \text{ mm}. \]

and the standard error of depth measurement is

\[ \frac{\Delta Z}{0.166} = \frac{0.24 \text{ mm}}{1.4 \text{ mm}}. \]

5
Rotations of rigid bodies may be inferred from the image as well, with rotation about the z-axis the most directly measurable of all. Rotations about the x-axis or the y-axis also produce effects in the image, but these effects are more subtle. To actually measure these tilts involves measuring depth at several locations in the image.

B. Depth Measurement by Triangulation

An alternate method of obtaining information about the third dimension is to use triangulation. Triangulation entails the use of two or more distinct points of view. The use of two separate cameras constitutes stereo. Comparing pictures taken sequentially from a single camera whose position has changed with respect to the object is referred to as motion parallax.

An alternate way of achieving triangulation is to place a camera at one point of view, and at the other point of view a projector that beams a particular pattern of light upon the object to be sensed. The geometric considerations are similar to those for stereo, but the image processing is different in that we analyze only the pattern of projected light. The use of patterned light offers some unique benefits not obtainable with conventional imagery.

The simplest light pattern we can project is a single point. The point pattern can be generated by a laser beam or by the use of conventional optics to image a pinhole with a light source behind it. For explanatory purposes we will assume that the laser or projector is to the left of the camera, on the minus x axis.

The geometric relationships of the projector, camera, and workpiece are illustrated schematically in Figure 2. We derive

\[ \frac{\Delta X}{\Delta Z} = \frac{X_2 - X_1}{Z_2 - Z_1} = \tan \phi \]

in the limit as \( \Delta Z \) approaches 0. The situation is similar to that of Figure 1, except that in this case the angle \( \phi \) is independent of the camera's field of view. A convenient value for \( \phi \) is 30 degrees, so that
\[ \tan \phi = 0.5. \]

Using the same conditions we used to estimate depth resolution previously, the standard error in measuring depth is 0.23 mm. Resolution in depth may be made arbitrarily fine (if we disregard the effects of a finite spot size) by using camera lenses that give high magnification.

If instead of a spot we project a vertical bar of light, we can obtain additional information about the scene. The vertical bar of light may be generated by passing a laser beam through a cylindrical lens or by using conventional optics to image a slit with a light source behind it. Analysis of a vertical bar is similar to scanning a single point up and down and measuring \( z \) at several different \( y \)-values, except that all the points are read with a single picture.

Note that a horizontal bar of light would give us no information at all. If a single beam were scanned from side to side and many images of the spot were input, we would be able to measure depth at many points along a horizontal line. But, in reading all the points simultaneously
in a single image, we lose the knowledge of which point in the image corresponded to which position of the beam. The same situation arises whenever the lens center of the camera lies in the plane of light generating the bar.

If the target is a flat surface, the camera will perceive a straight line. The horizontal position of the line in the image is a function of the z-coordinate or depth of the surface. Any tilt of the line away from the vertical corresponds to a rotation of the plane about the x-axis. If the target, instead of being a featureless flat surface, has a recognizable shape, other information can be inferred. Specifically, if there is a horizontal edge or corner, the image of the bar of light will have a discontinuity, the position of which is a function of the object's y-coordinate.

Other patterns may be imaged upon the workpiece to give additional information. For example, two parallel vertical stripes can give information regarding rotation of a plane about the y-axis. But the possibilities are too numerous for exhaustive analysis here.

C. The Tool

The object of visual servoing is usually to apply a tool to a workpiece. Frequently, however, the tool occludes part of the scene. Ideally, the camera should measure the position of the tool relative to the workpiece but with real-time binary image processing systems, this is not usually possible. If we cannot sense the target with the tool intervening, we must rigidly attach the tool to the camera in a known relative position.

We recognize two distinct modes of visual servoing. In the first mode, which we call "point mode," servoing is used to bring a tool to some specific location—for instance, to insert a bolt into a hole. If the target is in motion the servo system should track it, so that the relative velocity of the camera and tool, with respect to the workpiece, is zero. In the second mode of servo operation "line mode," the objective is to follow a path at some specific nonzero velocity—for example in sealing, gluing, or seam following.
When the camera moves with respect to the target in line mode, additional geometric information about the position and orientation of the object may be obtained from that motion. Successive images of a groove can (with knowledge of how the camera moves) give the orientation of the groove where a single image cannot.

Once visual servoing attains the desired relationship between the camera and the workpiece, we can move a fixed distance to place the tool in the same relationship. In point-mode servoing this involves a separate motion. In line-mode servoing the tool can simply follow the camera.
III DYNAMIC ANALYSIS

The camera and its controlling computer are only one element in a system that tries to maintain a specified relationship between the camera and the workpiece. The other major component is the Unimate robot, considered in combination with its controlling LSI-11 computer. The vision subsystem and the manipulation subsystem interact with each other in unusual and sometimes unpredictable ways. We are interested in stability and speed of response as well as accuracy. An understanding of the interaction between subsystems is essential to obtain the best possible performance.

Figure 3 BLOCK DIAGRAM OF VISUAL-SERVO CONTROL SYSTEM

A simplified block diagram of the control system is shown in Figure 3.
Image processing and overall control take place in the PDP-11/40 minicomputer. Visual analysis of a scene takes place in two phases: picture taking and image processing. Picture taking involves formation of the image on the camera diode array chip and reading of the signal into an image buffer in the PDP-11/40. This may take between 10 and 66 milliseconds. Image processing is slower, requiring from 100 to 500 milliseconds. Analysis of the image data produces an "error signal" that drives the rest of the servo system. This error is a vector quantity, consisting of position in one, two, or three dimensions, and may also include orientation. The control algorithms (also residing in the PDP 11/40) convert the error signal(s) to commands to the manipulation subsystem. Commands are transmitted through a DR-11 parallel interface.

The programs to control the Unimate robot that run in the LSI-11 microcomputer and the communication protocols have been previously described [8,9]. We make use of this facility to specify both positions and velocities.

If we know the visual target is stationary, only positions need be commanded. The PDP-11/40 derives position information in the camera's own coordinate frame, and the LSI-11 microcomputer performs the necessary coordinate transformations to move in the appropriate direction.

In following a path at fixed velocity we make use of the ability of the LSI-11 software to deal with moving coordinate systems. The Unimate control is commanded to maintain a fixed velocity (again relative to the camera's coordinate frame); thereafter we can command position changes relative to the moving coordinate system.

In maintaining position with respect to an object that moves at an unknown velocity, we attempt to control position and velocity simultaneously. This is possible because velocity applies to the moving coordinate system, whereas position applies to the manipulator with respect to that coordinate system.
Twenty times per second the LSI-11 computer transforms the instantaneous commanded position to a set of joint positions (encoder set point values) that are passed on to the Unimate hardware control. A complete and rather complicated servo system moves the robot hydraulically until the actual readings of the joint encoders agree with the commanded set points. The response is nonlinear, nonisotropic, and generally unpredictable. The robot will achieve its commanded position in a reasonable time; but some joints move faster than others, and the speed of response of some joints may depend on the positions of other joints.

As the Unimate moves the TV camera relative to the workpiece, it causes a change in the perceived scene, thereby closing the control loop.

The simplest way to visually servo a manipulator is to take a single picture, estimate the error, command a new position calculated to correct the error, wait a sufficient time for motion to terminate, and then repeat the process. When the target is stationary and speed of response is not critical, this approach can give adequate results. This is the control algorithm used for the bolt-insertion experiments described elsewhere [9,10].

When a faster servo response is desired or when the target may be in motion, it is desirable to take pictures as often as possible. If, for each picture, an incremental move were commanded to precisely cancel the error signal, the delays in the system would quickly cause a highly unstable response. A way to defeat instability is to command smaller moves. The correction applied to the Unimate will be the product of the error estimate and a constant we call beta. Beta is always less than 1 and may be adjusted to give overdamped, underdamped, or critical response to a step function in the target's position.

* Consider the hypothetical case in which a picture is taken and subsequent picture processing reveals that the arm is one cm to the right of its target. If we command a one-cm move to the left and immediately take another picture, the image will be very similar to the first one, because motion of the arm will have barely begun when the scan of the TV camera takes place. Commanding an additional one-cm movement to the left would be counterproductive.
If the target is in motion, the same error signal will control the velocity of the moving coordinate system. Each time an error is sensed along the +x direction for example, we will add the product of that error and another constant, which we call gamma, to the coordinate system velocity. Gamma is generally independent of beta and should be adjusted for critically damped response.

We have determined experimentally that the value of beta that achieves critical damping is linearly related to the time interval between successive pictures. A simple scene with minimal picture processing may be analyzed in as little as 150 milliseconds; however, with increasing sophistication of the processing used, with a more complex scene, and with extraneous noise in the image, processing time may increase to as much as 500 milliseconds. To achieve the best response, beta must be smaller for the shorter cycle times.

To take advantage of this relationship we have implemented a semiautomatic adjustment algorithm for beta, in which the time between pictures is measured using the PDP-11's real-time clock. Then, instead of using a constant beta, we multiply the error estimate by the ratio of elapsed time to a hypothetical "Unimate positional response time." (If that ratio turns out to be greater than unity, we let beta=1.0.) We have found that a value of 500 milliseconds works best for the response time. A similar relationship appears to hold for gamma, but further study is needed.

The proportional control algorithm is deficient in that no account is taken of arm motion during picture processing. There are basically two ways to take accomplish this. The first is to use some mathematical model of the arm. The other way is to use the Unimate control program in the LSI-11 to measure the actual position of the arm.

We have achieved a somewhat smoother response by using the following simple predictive arm model: we keep track of the most recent incremental motion command sent to the robot and assume that during picture processing the arm has moved some proportion of that commanded distance. We assume that the arm is at a different position from where
it was when the picture was actually taken, and compensate for this in estimating the position error. We find that if we "cancel out" 20 to 30 percent of the previous motion command in this way, the response curves show fewer small random excursions.

It is sometimes better to actually measure position than to merely estimate it. We have the capability of interrogating the Unimate's LSI-11 computer to find out the precise, actual position of the Unimate's end-effector at any given time by reading the joint encoders and converting them to Cartesian values in the camera's coordinate system. Furthermore, we can obtain this information at approximately the very time the picture is taken. Thus, at some instant we know both the absolute position of the camera and its visually measured position error. Regardless of processing delays, the Unimate is commanded to a new absolute position relative to the measured one.

The distinction between position estimating and position measuring may be made clearer by the following comparisons: in position-estimating mode, moves are made relative to where the LSI-11 microcomputer has last commanded the arm to move to. In position-measuring mode, moves are made relative to a measured position, regardless of commanded positions. In position-measuring mode, the arm is continually commanded to move all the way to its target position, not just to approach it. With proportional correction the slowest joint of the Unimate dictates a maximum value of beta. By making use of position measuring, each joint can proceed to its target position at its own maximum speed.

There are a number of problems associated with servoing in position-measuring mode that need to be solved before it can be used generally. One problem is that small orientation errors tend to grow and cause drift. Since we are moving relative to measured positions, any tilt in the hand's axis as it moves becomes incorporated into the new target position and, if the error is systematic, it builds up. The details of moving in several degrees of freedom relative to an actual position, while constraining other degrees of freedom, have not yet been resolved.
Another problem stems from the delays in communication between the PDP-11/40 and the LSI-11. At present excessively long, they are caused by the scheduling program in the LSI and the large number of messages that need to be exchanged between the two computers. This problem can be solved by improving the scheduling algorithm or by doing more of the calculations in the LSI-11.
IV EXPERIMENTAL RESULTS

We have demonstrated various experiments in which visual servoing was applied. These experiments are discussed below.

The use of visual servoing for insertion of bolts has already been documented [9]. In addition the assembly operation described in reference [10] demonstrates a related mode, wherein a movable table is positioned relative to a fixed camera. In all the bolting experiments we have tried so far, the target has been stationary and the system has achieved a placement accuracy of better than 1.2 mm (0.050 inch) by taking one or two pictures.

One of the first demonstrations of velocity tracking was to have the Unimate arm, with the camera in its hand, follow an object on a moving belt. A calibration procedure measured the height of the belt relative to the Unimate's coordinate system, but no presumptions were made about the speed or direction of the belt. The usual mode of operation was to place an object on the belt under the camera, commence the servoing action, then turn on the belt drive motor. The Unimate would take several seconds to accelerate to the belt speed. When three consecutive TV pictures showed a position error of 1.2 mm (0.050 inch) or less, the arm would move down and over, relative to its moving coordinate system, to grasp the object. Figure 4 shows the configuration we used for this experiment.

An adaptation of the previous experiment was employed for simulation of spot welding on a moving line. Figure 5 shows the Unimate set up to perform this experiment. The target was a piece of 1/4 inch plywood, supported vertically on the moving belt. A slit projector cast a bar of light upon the edge of the plywood. The servo control used the image of this bar to control both the position and the orientation of the Unimate, relative to the edge. Servoing was
Figure 4 TRACKING AN OBJECT ON A MOVING BELT

Figure 5 SIMULATED SPOT WELDING ON A MOVING LINE
constrained to a fixed height above the belt, with the camera's principal axis pointing horizontally. In addition to the camera, the Unimate's end-effector included a simulated spot welding gun. After positioning itself relative to the edge, the arm moved by dead reckoning to place the gap in the tool over the plywood. The size of the gap relative to the plywood required that the maximum error be 6 mm. The Unimate could achieve this accuracy about 90% of the time. It is unclear how much of the overall inaccuracy is attributable to servoing inaccuracy, to inaccuracy of the dead reckoning move, to nonrigidity of the tool, to variation in belt speed, or to rocking of the plywood on the belt.

In another experiment visual servoing was applied to following a curved path in three dimensions at constant velocity. Such a mode of operation could apply to gluing, sealing, and seam following. The path to follow was created by taping together pieces of cardboard to form a large "tub" with a groove curving in three dimensions. Figure 6 shows the tub with the Unimate in servoing position. The slit projector was directed to generate a V-shaped pattern on the groove. This image was used to directly control position of the camera in two dimensions and to establish an orientation in which the axis of the camera bisected the "V". As the camera and simulated tool moved forward, successive corrections were made to reorient the direction of motion (in two degrees of freedom) to remain parallel to the groove. Thus, in this demonstration, all six degrees of freedom of the Unimate were being controlled by the servo system.

Several difficulties prevented this demonstration from being completely successful. A space intrusion problem was caused by the size and bulk of the hardware needed to support the camera, the projector, and the tool. There were places in the model tub that the tool could not reach because of interference by the end effector. Another problem was a generally jerky response, principally as a result of the considerable distance between the Unimate wrist and the end of the simulated tool. The Unimate LSI computer took into account the length
of the tool, so that changes in orientation would be made around the tip of the tool rather than around the wrist. Consequently, when orientation changes were commanded, a rather large motion of the wrist was required to keep the position of the tool tip constant. To achieve the desired change in orientation slowly and smoothly, further investigation will be necessary.
REFERENCES


With three servo seeks of about one second each and a bolting operation that with lateral, downward, and upward motions requires about two seconds, we find that about five seconds is required to reliably insert a bolt in a hole whose position is known only approximately.

Some very straightforward modifications should reduce the overall time considerably. Three consecutive seek operations are unnecessary—this should be reduced to two. Visual processing of the second image should be overlapped in time with the preparatory motion of the hand so that the second fine correction modifies the final approach. Many hundreds of milliseconds could be saved by optimization of the arm trajectory and the use of sensors to detect contact of the bolt end with the hole and final tightening of the bolt. Thus without extensive modifications to our hardware or methods, the bolting time could be reduced to two seconds.
REFERENCES


