AN ADAPTIVE ROBOTIC TRACKING SYSTEM USING OPTICAL FLOW

Ren C. Luo, Robert E. Mullen Jr., and Daniel E. Wessell

Robotics and Intelligent Systems Laboratory
Department of Electrical and Computer Engineering
North Carolina State University, Raleigh, NC 27695-7911

Abstract

A robotic system that can visually track and intercept an arbitrary object which is travelling at an unknown velocity on a conveyor has been presented.

A fiber-optic eye-in-hand vision system developed by the Robotics and Intelligent Systems Laboratory at NCSU is used as an integral part of the entire tracking system. The eye-in-hand system is employed to characterize the object trajectory in real time via a modified optical flow approach. A control strategy has been developed which utilizes the kinematic data that is extracted by the tracking algorithm to intercept the moving object.

An overall system configuration and its basic principles are described. The demonstration of the initial results is presented.

1. Introduction

The implementation of robotic devices with intelligent sensory capability has been recognized as an essential ingredient towards the success of flexible manufacturing systems. Currently, a typical robot-conveyor system requires a priori knowledge of the conveyor operating speed. The target objects must be placed on the conveyor at defined orientations and intervals. With such a system, difficulties can arise when the conveyor speed varies even slightly or the object is not placed with sufficiently precise orientation. Therefore, a more flexible system which allows the robot to acquire randomly oriented workpieces from a conveyor moving at an arbitrary velocity would be an excellent candidate to solve this problem.

Various approaches have been explored in an attempt to develop a means of accomplishing vision based object tracking. Conveyor tracking using a CMAC based learning control system has been developed by Miller [1]. The major constraints of this approach are the need for numerous trials to actually "learn" the control strategy and the computational complexity that hinder real-time implementation. Many algorithms have been developed which track objects in the image plane. They do not, however, attempt to use this data to perform tracking using the robot. These methods include two that have been developed by Samy [2]. The first performed linear feature matching in corresponding frames using Hough transform techniques. The second approach employed image segmentation techniques based on adaptive statistical clustering and efficient texture measures. Also, an adaptive tracking system using stereoscopic vision has been explored by Rasure [3].

In addition, many papers have been presented in which optical flow is used to estimate rigid body motion [4-8]. Optical flow has also been applied to the passive navigation problem. The major accomplishment of these papers is to demonstrate that motion of a rigid body can be efficiently recovered via optical flow. The system presented in this paper makes use of this result in order to better facilitate realtime robotic tracking.

We have developed a robot conveyor tracking system that incorporates a combination of visual and acoustic sensing. This system avoids the lengthy computational requirements inherent in many other approaches for realtime tracking applications.

A brief description of the overall system configuration is presented in Section 2. Section 3 describes the basic principles used to achieve the tracking of an object moving on a conveyor. The preliminary experimental results are included in Section 4. Section 5 is the conclusion of the paper.

2. Overall System Configuration

Most of the visual sensing systems that have been developed utilize a constant in-time sensing field, i.e. a fixed overhead video camera located above the work area. However, this static arrangement has a major disadvantage in that the field of view of the vision system is obstructed while the robot manipulator is retrieving the object. Furthermore, should one be interested in only a part of the work area, it is not possible to zoom in on that area with increased resolution. To overcome these problems, camera-in-hand systems have been developed [9-12].

The advantages of camera-in-hand systems are two fold: (1) the camera is always stationary relative to the robot hand, (2) the object to be acquired cannot be hidden from the camera by interference from the manipulator. However, the disadvantage is that these cameras are so bulky that even the smallest camera is too large to allow convenient physical integration with the gripper. Thus the maneuverability of the robot system is constrained. The NCSU Robotics and Intelligent Systems Laboratory has developed a fiber-optic based eye-in-hand vision system to improve the effectiveness of our camera in hand system. The eye-in-hand system is shown in Figure 1.

Because this system utilizes fiber optics to allow the remote location of the camera, it thereby offers many advantages over conventional vision systems which include: the physical separation of electronics and optics; the integration of the sensing head (which is lightweight and considerably smaller than most small cameras) into the robot's gripper, thus minimizing the reduction in net payload of the robot caused by the additional weight of the sensor; and improved image quality due to the inherent immunity to electrical noise of fiber optics.

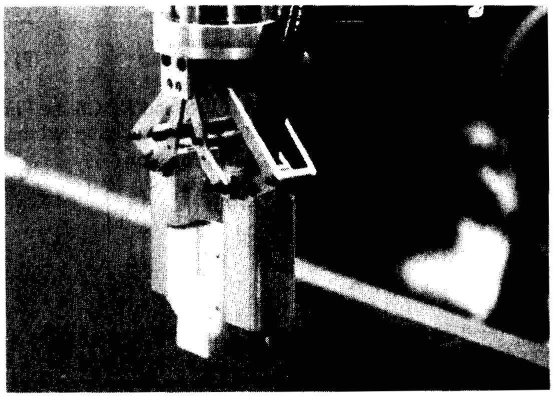


Figure 1.) Eye-in-hand Vision Sensor

The overall system configuration used in the initial experiments is shown in Figure 2. The major components include a PUMA 560 robot with a hybrid force/position servo-controlled parallel jaw gripper, a Trapix real time image processor, a Polariod ultrasonic sensor, a variable speed conveyor, and a MicroVAX II computer. One of the robot gripper fingers houses a fiber optic eye-in-hand vision sensor while the other accomodates a fiber optic light source and an ultrasonic sensor. The eye in hand sensor is connected via a fiber optic bundle to a CCD camera and interfaced to the Trapix real time image processor. The MicroVAX II workstation serves as a supervisory computer for the PUMA controller, the image processor, and the ultrasonic sensor.

A concurrent process monitors each of these external devices to allow parallel processing and transfer of sensory data in order to better achieve real time tracking of the target object. Interprocess comunication is realized through the use of a common block in the host's system library. The flow of information that occurs during the tracking process is shown in Figure 3.

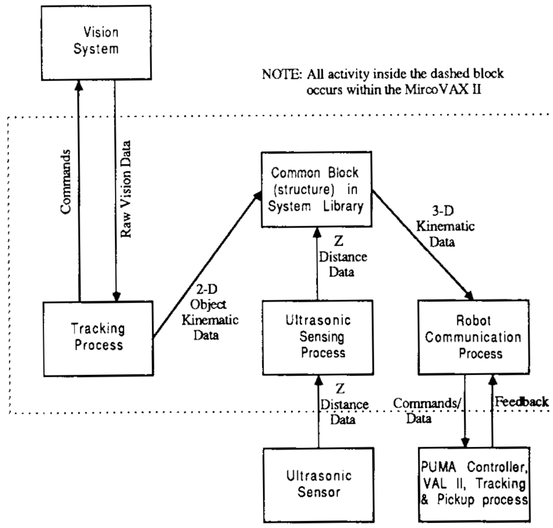


Figure 3.) Tracking System Communication Block Diagram

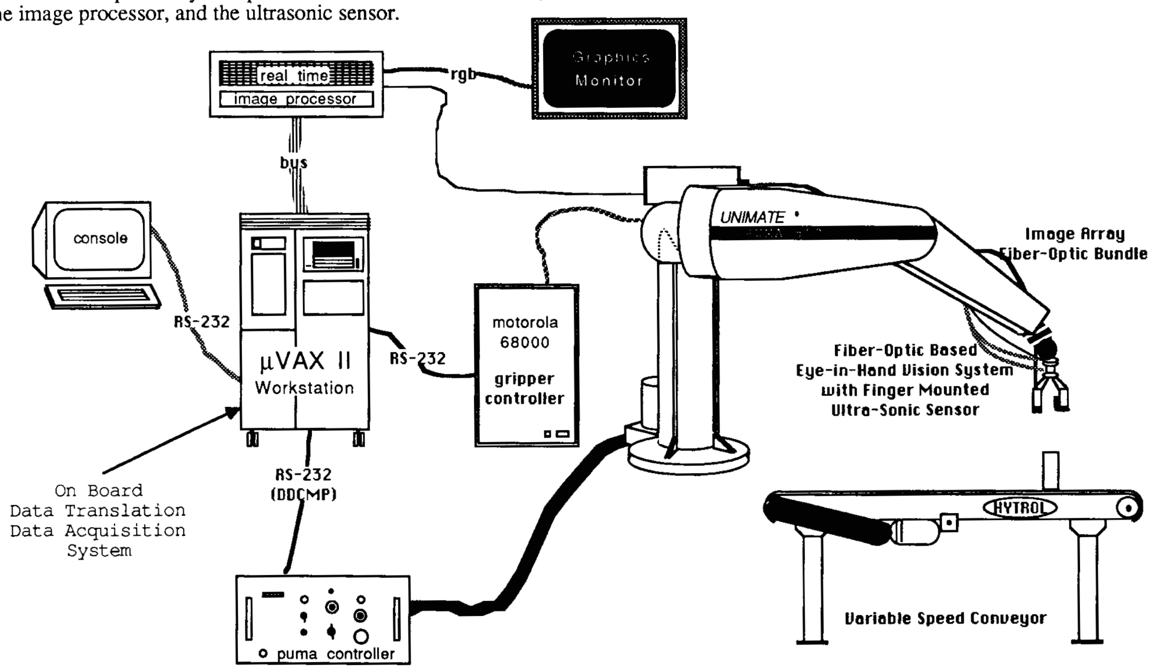


Figure 2.) Overall System Configuration

The purpose of a robot vision system is to provide information describing the task environment in order to control the motion of the robot necessary to perform some desired function. However, most vision systems for such applications can provide only a 2-D description of the scene in the plane of the workspace which is incident to the optical axis of the camera [13]. Vision systems provide good shape, position and orientation data of objects but their range sensing ability is relatively poor [14].

An ultrasonic sensor was incorporated into the tracking system for quick and accurate range data measurement. This data is essential for interpreting the image data correctly, since the distance from the camera to the object plane must be known in order to relate a displacement in image coordinates to its true displacement in world coordinates. This information can also be used to determine true object size from its image plane projection. Thus the primary purpose of the ultrasonic sensor in our task is to provide initial range data on how far the camera is from the object so that all visual information can be interpreted correctly.

3. Combined Visual/Acoustic Approach for Robotic Conveyor Tracking

TRACKING

In this application, the motion of the conveyor is known to be from left to right. In our tracking scheme we have also assumed that only one object is to be tracked at a time, and that the object is undergoing 1-D translation with no change in orientation. These assumptions allow us to track the object in consecutive frames by recognizing the first pixel that is scanned (from bottom to top, left to right) by the robot eye-in-hand vision system, and finding the corresponding point in future frames. This point is to be referred to as the object coordinate system (OCS). This simplification eliminates the need for classical feature extraction in consecutive frames and thus greatly reduces the overall processing time.

The coordinate systems of the robot hand and image plane are defined as follows; the x-axis of the robot hand coordinate system (RHCS) is aligned with the direction in which the conveyor is moving and the RHCS x-axis is aligned in parallel with the x-axis of the image coordinate system (ICS). The RHCS and ICS y-axes are similarly aligned. Consequently, the xy-plane of the ICS and the xy-plane of the RHCS are parallel.

Using the ultrasonic sensor, the distance d from the camera (and the robot hand) to the conveyor can be measured. The dimensions of the visual field frame are proportional to d. Thus, the d value is used to convert ICS locations, which are given in pixels, to true distance quantities (in our case inches). The conversion from pixels to inches for our system is;

$$\begin{aligned} x_I &= 6.2\times10^{-4}\cdot d\cdot x' \text{ inches} \\ y_I &= 6.2\times10^{-4}\cdot d\cdot y' \text{ inches} \end{aligned} \tag{1}$$

where, (x',y') is the pixel location of the object in the image frame (ICS), (x_I,y_I) is the object location in the image frame expressed in inches, d is the distance from the camera to the conveyor, and 6.2×10^{-4} is a proportionality constant which relates d to the image size.

A point ^{I}P in the ICS can be transformed to a point ^{R}P in the RHCS by the multiplication of a homogeneous tranformation matrix $^{R}\mathbf{T}_I$.

$$^{R}P = {}^{P}\mathbf{T}_I \; {}^{I}P \tag{2}$$

Since the x and y axes of the ICS and RHCS are in alignment, $^{P}\mathbf{T}_I$ contains no rotations. We have calibrated the robot arm system in the format shown in equation (3):

$$\begin{bmatrix} x_R \\ y_R \\ z_R \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & (1.25-.079d) \\ 0 & 1 & 0 & -.079d \\ 0 & 0 & 1 & -d \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_P \\ y_P \\ 0 \\ 1 \end{bmatrix} \tag{3}$$

Thus, the transformation from ICS to RHCS is merely a function of d.

Since the conveyor is always known to be travelling parallel to the x-axis of the ICS (also the x-axis of the RHCS), ICS velocities, V_I, are proportional to RHCS velocities, V_R, once ICS velocities have been converted from pixels per second to inches per second. We can specify robot position and velocity relative to the RHCS via software. Thus, x_R, y_R, z_R and V_R can be employed directly to control the robot to track and pick-up an object. V_R is used to track the object and when it is time to pick-up the object, x_R, y_R, z_R are used to calculate the robot hand trajectory necessary to ensure a proper pick-up.

Several criteria can be used to determine when an object can be picked up (e.g. when the relative velocity between the RHCS and the OCS is zero). When the criteria are met, the trajectory necessary to pick up the object within t seconds can be calculated, where t is proportional to V_R . If the robot is moving at a velocity V_R when the pickup criteria are met (call this t_p), the object will travel with a velocity V_R for another t seconds before pickup is made. If at t_p the object is at an arbitrary point in the RHCS, it is necessary to move the robot such that the origins of the RHCS and OCS coincide at time t_p+t . If the robot travels in a straight line the distance s_r it must travel can be computed by:

$$s_r = \sqrt{(x_p+tV_R)^2 + y_p^2 + z_p^2} \tag{4}$$

The tV_R term is added to the x_p component of position since the object will still be moving in the x-direction for t seconds until it is picked up.The robot must travel the additional t seconds along this trajectory. At this moment, the velocity V_{pickup} can be calculated as:

$$V_{pickup} = s_r / t \tag{5}$$

This pickup trajectory was chosen for the simplicity in which it can be implemented in VAL II.

Based on this analysis, it is possible to track and pick up an object from a conveyor by extracting object position and velocity information from a sequence of images. As mentioned previously, the position of the object relative to the RHCS will only be corrected for during the pick-up. The velocity information, however, is used throughout the tracking process in such a manner as to try to achieve and maintain zero relative velocity between the object and the RHCS.

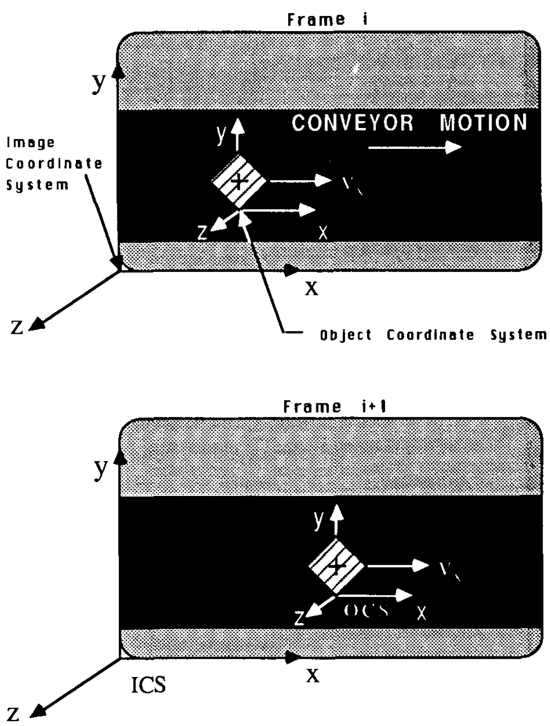
Vision Based Tracking

Given an image of an object there are various approaches for extracting feature information. These methods have been presented elsewhere and are not within the scope of this paper. Given two consecutive image frames it is possible to locate a common point of an object (e.g. a corner or the centroid) so that we can calculate the velocity at which the object is moving.

Since we have initiallly assumed that the x-axis of the ICS and RHCS are aligned and that all conveyor motion is solely along the x-axis, the velocity, V_{Rx} of the object along the x-axis in RHCS is calculated as:

$$V_{Rx} = \frac{\partial}{\partial x}({}^{ICS}P_0) - \frac{\partial}{\partial x}({}^{RHCS}P_{ICS}) \tag{6}$$

where, ${}^{ICS}P_0$ is the position of the object in the ICS and ${}^{RHCS}P_{ICS}$ is the position of the ICS relative to the RHCS. The use of an eye-in-hand vision sensor simplifies the problem of computing the velocity of an object relative to the robot hand because the position of the ICS to the RHCS is known and constant. Thus, the second term on the right hand side of equation (6) becomes zero. The velocity of an object in the RHCS can be determined directly from its position in the ICS. Using a discrete approximation this velocity can be approximated as shown in Figure 4.



$$\vec{V}_{X\,avg} = \frac{X_{i+1} - X_i}{\Delta t}$$

Figure 4.) Object Velocity Estimation

This approach was originally employed to estimate the velocity for use in the tracking algorithm. Initially features from a binary image were used. This method was fairly accurate, however, the time required to choose an appropriate threshhold and binarize the image was relatively lengthy so that the maximum conveyor tracking speed was slow. For this reason we chose to try a tracking algorithm which utilized the initial grey level image, thus saving the time it takes to binarize the image. As a result, this method sampled at roughly twice the speed of the binary method, but it suffered from a noise sensitivity problem.

It was therefore deemed necessary to develop an algorithm which took the dynamic nature of the problem into account. An optical flow approach was then employed to assist in determining the true velocity of the object.

Optical Flow Approach

Since we are concerned with rigid body motion, optical flow corresponds well with the motion field. Let E(x,y,t) is the image irradiance at time t at the point (x,y), and let u(x,y) and v(x,y) be the x and y components of the optical flow at the point (x,y) respectively. Assuming the flow and motion fields are the same, the irradiance should be the same a small time interval, Δt, later at the point $(x+V_x\Delta t, y+V_y\Delta t)$, where V_x is the velocity of the point in the x direction and V_y is the velocity of the point in the y direction. In our case all velocity is in the x direction, thus V_y=0. Hence;

$$E(x+V_x\Delta t, y, t+\Delta t) = E(x, y, t) \tag{7}$$

The motion field of the object is continuous, thus, assuming that brightness is continuous (or at least close to continuous) in x, y, and t the left hand side of equation can be expanded in a first order Taylor series as;

$$E(x,y,t) + \Delta x \frac{\partial E}{\partial x} + \Delta y \frac{\partial E}{\partial y} + \Delta t \frac{\partial E}{\partial t} = E(x,y,t) \tag{8}$$

subtracting the irradiance at the point (x,y,t) from both sides, dividing by Δt and then taking the limit as Δt approaches zero yields:

$$\frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial y}\frac{dy}{dt} + \frac{\partial E}{\partial t} = 0 \tag{9}$$

Since y is constant this equation reduces to:

$$\frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial t} = 0 \tag{10}$$

The partial derivatives in equation (10) are approximated using a discrete technique similar to one used by Horn [8]. This method attemps to minimize the sum of the departure from smoothness and optical flow constraint error. The result is an iterative solution of the form;

$$V_x^{n+1} = V_x^n - \frac{\frac{\partial E}{\partial x}V_x^n + \frac{\partial E}{\partial y}V_y^n + \frac{\partial E}{\partial x}}{1 + \lambda\left[(\frac{\partial E}{\partial x})^2 + (\frac{\partial E}{\partial y})^2\right]} \cdot \frac{\partial E}{\partial x}$$

$$V_y^{n+1} = V_y^n - \frac{\frac{\partial E}{\partial x}V_x^n + \frac{\partial E}{\partial y}V_y^n + \frac{\partial E}{\partial x}}{1 + \lambda\left[(\frac{\partial E}{\partial x})^2 + (\frac{\partial E}{\partial y})^2\right]} \cdot \frac{\partial E}{\partial y} \tag{11}$$

where, $\bar{V}_x$ and $\bar{V}_y$ are local averages of V_x and V_y , and λ is a parameter that weights the error in the image motion equation relative to the departure from smoothness. This implies λ should be small for a noisy image and large for a noise free image.

To solve these iterative equations it is necessary to approximate the first order partial derivatives of E(x,y,z). We can do this by looking at the image brightness at a point (x,y) at time t and again at time t+Δt. The approximations are as follows;

$$\frac{\partial E}{\partial x} \approx \frac{1}{4\Delta x}(E_{x+1,y,t} + E_{x+1,y,t+\Delta t} + E_{x+1,y+1,t} + E_{x+1,y+1,t+\Delta t}) - \frac{1}{4\Delta x}(E_{x,y,t} + E_{x,y,t+\Delta t} + E_{x,y+1,t} + E_{x,y+1,t+\Delta t})$$

$$\frac{\partial E}{\partial y} \approx \frac{1}{4\Delta y}(E_{x,y+1,t} + E_{x,y+1,t+\Delta t} + E_{x+1,y+1,t} + E_{x+1,y+1,t+\Delta t}) - \frac{1}{4\Delta y}(E_{x,y,t} + E_{x,y,t+\Delta t} + E_{x+1,y,t} + E_{x+1,y,t+\Delta t}) \tag{12}$$

$$\frac{\partial E}{\partial t} \approx \frac{1}{4\Delta t}(E_{x,y,t+\Delta t} + E_{x,y+1,t+\Delta t} + E_{x+1,y,t+\Delta t} + E_{x+1,y+1,t+\Delta t}) - \frac{1}{4\Delta t}(E_{x,y,t} + E_{x,y+1,t} + E_{x+1,y,t} + E_{x+1,y+1,t})$$

where $E_{x,y,t}$ is the grey level intensity of the point (x,y) in the image that was taken at time t. For our 1-D case the equation for V_x has been simplified slightly since Vy and $\partial E/\partial y$ are both zero (actually $\partial E/\partial y$ is of the indeterminite form (∞ - ∞), but we shall approximate this as zero). The net effect of these equations is that the velocity at a point is set equal to the average velocity of its neighbours minus an adjustment. This technique requires several iterations to converge and is limited to the measurement of constant velocity fields. Refinement of this technique is required in order to accomplish adaptive robotic tracking because the robot/vision sensor and the object are not travelling at a relative velocity which is truely constant.

The modified optical flow scheme is based upon the assumption that the motion and the flow fields are essentially the same. In addition, the selection of an object moving on the conveyor can be entirely arbitrary provided that the object is a rigid body. Since the camera starts to move once the first velocity is calculated, the velocity of the camera relative to the object changes after each iteration. In this scheme, V_x^{n+1} does not represent the absolute velocity at which the object is travelling. It is instead the relative velocity between the camera and the object. After each iteration the velocity of the robot hand (and camera) is modfied by an amount of V_x^{n+1}. Thus only at the first iteration is V_x^{n+1} aproximately the same as the absolute velocity of the conveyor. At each iteration V_x^{n+1} is calculated as shown in (11), where V_x^n is the average pixel velocity and is calculated as shown in Figure 4 (edge pixels and their neighbors are used for this calculation since the brightness gradient is changing rapidly at these points), and the $\partial E/\partial y$ and V_y^n terms are both zero. Thus ,V_x^{n+1} is a function of λ, the brightness gradient, and the average edge velocity. As the number of iterations, n, increase, V_x will approach zero. The tracking algorithm is shown in Figure 5.

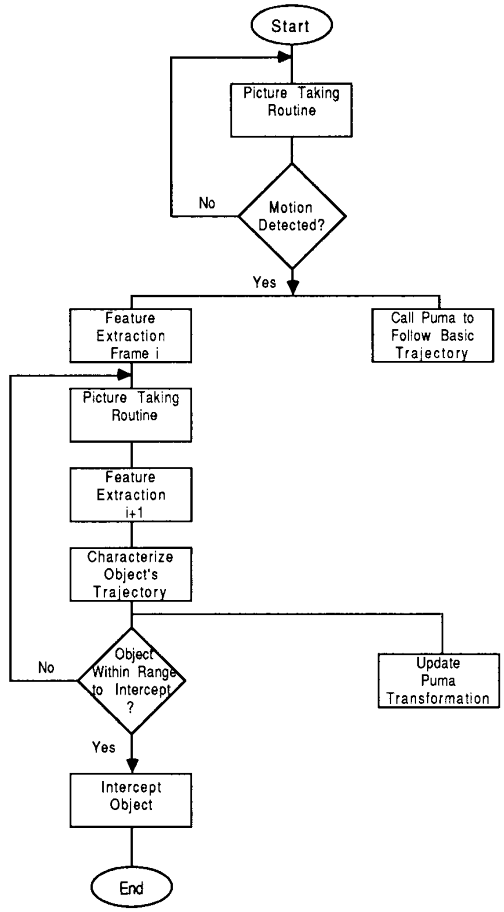


Figure 5.) Tracking Algorithm Flow Chart

Accurate realtime velocity estimation is imperative to ensure accurate tracking through quick response to any relative velocity of the object. The value of λ that optimizes the solution must be determined experimentally. A good selection of λ allows the system to effectively mask out false velocity fluctuations caused by varying lighting conditions and thus make the system better suited for operation in a real world environment where the use of highly structured lighting is impractical. The preliminary experimental results showed this method to be highly accurate.

4. Experimentation and Preliminary Results

The first experiment was the investigation of the accuracy of the vision portion of the tracking system. To perform this experiment the robot arm remained stationary as objects passed through the camera's field of view. The optical-flow algorithm was used to calculate the velocity of each object from the visual information. Three different objects (a sphere, cylinder and prism) were used in this experiment, with the conveyor travelling at speeds ranging from 1.5 to 20 ft/min. (fifteen different speeds in this range). The results showed an average error between actual and measured velocity of approximately 10%.

Another experiment was performed in order to evaluate the overall performance of the tracking system. This experiment involved the use of the tracking system to track and intercept (pickup) five different objects: a brass pipe fitting, golf ball, styrofoam block, aluminum machine part, and a chrome plated padlock. These objects are shown in Figure 6.

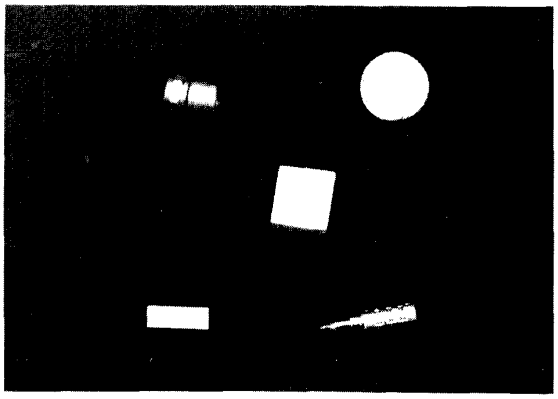


Figure 6.) Tracking Experiment Test Objects

The objects were placed on the conveyor with the tracking system having no a priori knowledge as to which of the objects was present or how fast it was travelling. One hundred trials, twenty per object, were conducted at velocities ranging from 6 to 15 feet per minute. Each object was placed on the conveyor in a random orientation. The lighting conditions were changed at random and the image quality varied accordingly. This experiment yielded an 87% success rate. A successful trial was interpreted as tracking which was accurate enough to allow for intercept (pickup) to occur.

5. Conclusion

A system for accurate 1-D robotic tracking of unknown objects travelling at arbitrary velocities has been presented. This approach is unique in that it uses an optical flow based vision algorithm to control a robotic manipulator such that it can track and acquire a generalized moving object in real time.

The experimental results demonstrate that the system exhibits good repeatability and is accurate for a variety of different objects. Also the system has proven to be robust in an environment without constant highly structured lighting as would be found in a real world application.

Efforts are currently being made to refine the performance of the present 1-D tracking system. The most significant enhancement to be made is to widen the field of view of the camera to allow the tracking of objects at higher velocities. The present window is approximately 4"x 4" at a distance of 2' (using a 25mm lens). With the present window, it has been found to be impossible to track objects traveling faster than 17 feet per minute. Doubling the window current size should allow for tracking at close to 35 feet per minute.

References

[1] W. T. Miller, "Senor-Based Control of Robotic Manipulators Using a General Learning Algorithm," IEEE Journal of Robotics and Automation, Vol. RA-3, No.2, April 1987.

[2] R. Samy "Real-Time Tracking of Target Moving on Natural Textured Background," Computer Vision for Robots Procedings, Cannes, France, December 1985.

[3] J. R. Rasure,"Robot Control Using Stereoscopic Vision Feedback," Recent Trends in Robotics: Modeling, Control, and Education, Elsevier Science Publishing Co. Inc., 1986.

[4] Amar Mitchie, "Computation of Optical Flow and Rigid Motion," Proceedings of the Workshopon Computer Vision Represntation and Control, Ringe, NH, 1982.

[5] J. Fang and T.S. Huang, "Estimating 3-D Movement of a Rigid Object: Experimental Results," in Proc. Int. Joint Conf. Artif. Intell, Karlsruhe, West Germany, 1983, pp. 1035-1037.

[6] Allen Waxman, "An Image Flow Paradigm," Proceedings of the Workshopon Computer Vision Represntation and Control, Ringe, NH, 1982.

[7] Daryl Lawton, "Motion Amalysis Via Local Translational Processing," Proceedings of the Workshopon Computer Vision Represntation and Control, Ringe, NH, 1982.

[8] B.P.K. Horn, "Robot Vision," pp. 48-53, The MIT Press, Cambridge Massachusetts, 1985.

[9] C. Loughlin, "Eye-in-Hand Robot Vision Scores Over Fixed Camera," Sensor Review, Vol. 3, January 1983.

[10] M. Shmerier, et al.,"Visual Feedback for Robot Control," Workshop on Industrial Applications of Machine Vision, Research Triangle Park, NC, 1983.

[11] R. C. Luo, "Microcomputer-Based Robot Dynamic Sensing System for Object Recognition and Manipulation," Journal of Robotics Systems, Vol. 4, No. 1, 1986.

[12] A. Pugh, et al.,"A Research Program in Sensor Guided Assembly," Proc. of the 13th International Symposium on Industrial Robotics; Chicago, IL, April 1983.

[13] D. G. Morin,"Development of a Robot Vision Systems Using Ultrasonics," Design and Synthesis Elsevier Science Publishers B.V. (North-Holland), 1985.

[14] K. Sasaki, et al.,"Development of Ultrasonic Robot Sensor," Design and Synthesis Elsevier Science Publishers B.V. (North-Holland), 1985.