

Fabrication and Control of 4-DOF, Autonomous Robotic Arm Using Low Cost AVR Controller

Javaid Iqbal, Nida Hassan, Atif Waqar, Sophia Bano, Rashid Ilyas
Department of Mechatronics, College of E&ME, National University of
Sciences and Technology Rawalpindi, Pakistan,
jiqbal-EME@nust.edu.pk, nidahas@hotmail.com, atif1982@yahoo.com,
sophiabano@hotmail.com, pearlsatdawn@yahoo.com

ABSTRACT

This paper summarizes the design and fabrication of a four degree of freedom autonomous robotic arm using Digital image processing. The robotic arm boasts a high performance, RISC architecture based microcontroller manufactured by Atmel, the AtMega16. Control of the robotic arm has been achieved successfully using four servo motors, each of which incorporates high resolution quadrature encoders. The robotic arm is equipped with a pair of high resolution digital cameras. These cameras transmit real time video to a Pentium based processor. The image processing module searches the video stream for predefined templates and calculates the coordinates of the object to grasp and transmits them serially to the microcontroller. The microcontroller implements inverse kinematics algorithms and then actuates the respective motors through angles so as to reach the coordinates desired in the most efficient manner. The motors are actuated using the internal PWM channels of the microcontroller and power amplifiers using efficient Mosfets. The robotic arm also has the provision of being controlled without the cameras relying on user input for the starting and terminating points. These coordinates are fed to the controller directly using a keypad and LCD module eliminating the Pentium processor. Thus even without a vision system the robotic arm may be defined as being autonomous. The end effector is a two finger gripper. The robotic arm has a load bearing capacity of upto 500gms and a maximum workspace of about 650mm of a meter.

1. Introduction

As manpower becomes more expensive and lives of people more precious, the need to find cheaper, faster and safer ways of performing tasks grows exponentially. This has led to research and growth in the field of robotics. The growing need in factory automation for an expanding involvement of robotic workstations in industrial applications increases the demand for not just efficient but also light weight robotic arms [1]. However in this era of fast growing technology there is an imminent need for the manipulator to be cost effective. Many ways of controlling the arm remotely have been developed, using the microcomputers [2, 3]. However in order for robots to become more useful, they must be able to function in more demanding environments than typically found in most factories. Thus if the system is autonomous, it is both time saving and cuts down on the labor cost.

Computer vision [4] is a fundamental requirement for autonomous machines. Methods of acquiring information using computer vision vary according to requirements. A pair of audio

video cameras has been used to acquire stereoscopic video and the code has been developed on assumption of uniform background using edge detection [5] and pair of cameras.

The robotic arm developed is a prototype, a light weight and low powered version of an industrial robotic arm. The goal is to implement all the control concepts and prove them using this scaled down version. A servo system has been used for the control of robotic arm [6, 7]. The robotic arm is powered by four DC servo motors. The entire control of the robotic arm has been achieved using one RISC based microcontroller, the Atmel AtMega16 [8]. This controller is not capable of image processing hence the need of a Pentium based processor aroused.

To achieve precise, accurate and stable system, PI control has been implemented on each motor [9]. The control terms must be individually adjusted or “tuned” for optimal performance. A simple PI algorithm has been developed so as to reduce the response time of the system. PI controller provides a robust closed loop control capable of implementing position control.

A pair of audio video cameras acquire real time video stream and an algorithm searches the video for predefined objects to calculate their location relative to reference axes. Once the position coordinates have been calculated, joint angles for each respective motor are determined using the algorithm based on the inverse kinematics [10].

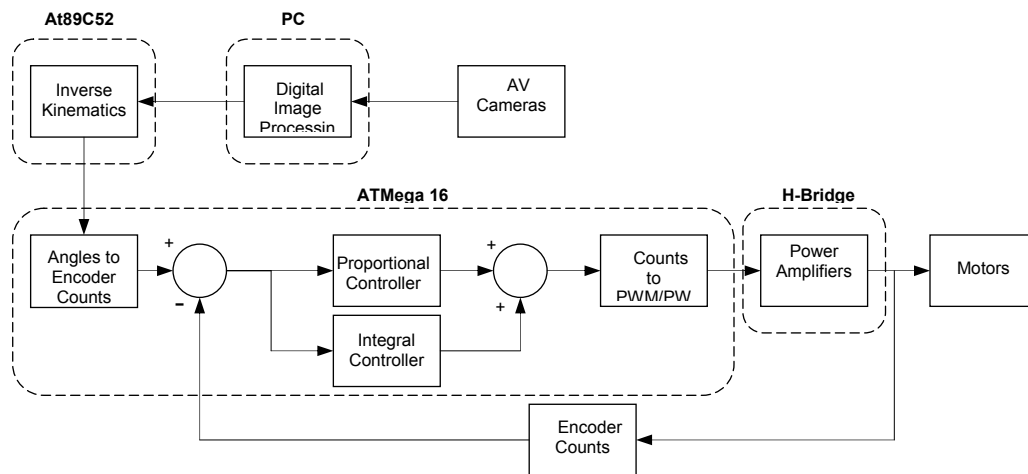


Figure 1 A Block Diagram of the System

2. The Mechanical Structure

The mechanical structure has been designed with the aim of demonstrating the concept of implementing digital image processing, inverse kinematics and closed loop controls. With this purpose in mind, the manipulator has been limited to four degrees of freedom. Being a concept demonstrator, the weight handling capability and workspace has been limited. Aluminum has been used due to its light weight, ease of machinability and low cost.

All the four joints are revolute, three of which form a planar manipulator and the fourth is the waist for the planar arm. The global workspace is the set of points (x, y, Φ_3) (where Φ_3 is the angle of the end effector with the horizontal) that can be reached by the end effector for all specified orientations Φ_3 of the output link r_3 (where r_3 is the length of the last link).

The manipulator consists of 3 links of lengths 380mm, 250mm and 210mm respectively and a turn table having a diameter of 500mm. It features a two finger hybrid gripper as the end effector. The gripper can grasp regular shaped objects like cylinders with convenience. Four 24V, 3200 rpm servo motors have been used to drive each of the four joints. The reduction ratio of the gearboxes for the planar robotic arm is 130:1. The speed of the waist has been reduced by a ratio

of 500:1. The entire mechanical structure has been modeled in ProE and a stress analysis of every individual component has been carried out using Ansys 9.0.

Limit switches are also mounted on the links. Their purpose is to ensure that no physical motion goes out of range. Thus limit switches here act as override controls for the arm segments.



Figure 2 The 4 DOF Manipulator



Figure 3 Camera Used

3. Vision System

Computer vision is a fundamental requirement for autonomous machines. Methods of acquiring information using computer vision depend upon the specific application. Typical techniques use camera calibration for machine vision. In contrast, our technique doesn't require calibration of cameras. Object detection is implemented using edge detection and a pair of cameras assuming uniform background. For the calculations we use distance between optical centers of cameras, the angles which the cameras make with a predefined line and field of view [11].

For range and offset, horizontal field of view is utilized. Law of sine is used in calculation.

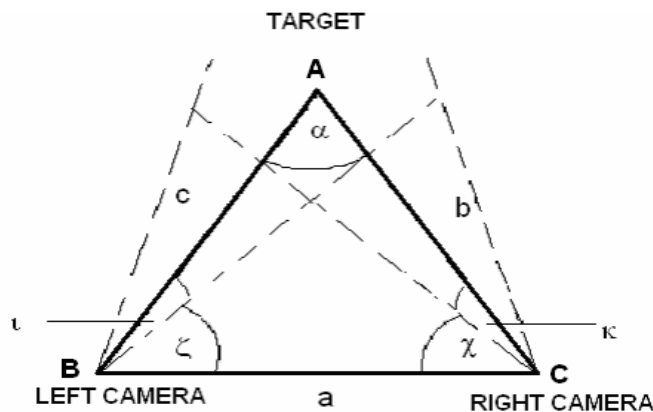


Figure 4 Principle Arrangement of Cameras

In figure 4,

$$\beta = \zeta + \iota$$

$$\gamma = \chi + \kappa$$

where, ζ and χ are angles which each camera makes with the line BC and ι , κ are angles calculated from image. Solving above triangle give us range and offset of target from our desired camera.

For height calculation, actual height of camera lens is measured and is not disturbed thereafter. The perpendicular imaginary line is drawn from camera to reference height and its corresponding pixel (cp) is found. Now utilizing triangulation, angle for target point is found. The

side opposite to target angle is calculated. If pixel corresponding to point of target is less than cp then the side calculated is added to referenced height and vice versa.

4. Kinematics

To reach the desired position and orientation motors, the joint angles are needed. These joint angles are calculated using the inverse kinematics and the motors are then actuated accordingly.

An inverse kinematics code developed runs from the microcontroller At89C52 calculates the joint angles for the respective motors. The code has been optimally written as the controller has only 256 bytes of on chip RAM. The algorithm is based on the numerical approach. Due to the mechanical limitations, only elbow up solution is used. The joint angles are governed by the following equations:

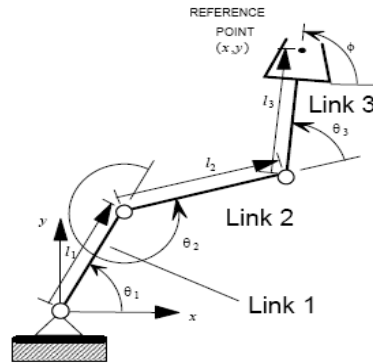


Figure 5 Kinematic Model

$$x = l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3) \quad (3)$$

$$y = l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3) \quad (4)$$

$$\Phi = \theta_1 + \theta_2 + \theta_3 \quad (5)$$

There is a second mode of operation also. This provides the user with a manual override, a hex keypad and LCD has been interfaced with At89C52. Instead of using the coordinates from the vision module, the controller prompts the user for the start and end point of the end effector. These parameters are then used to calculate the joint angles. These angles are then serially transmitted to the AtMega16 for implementing closed loop controls.

5. Controller

Atmel AtMega16 is a high performance, low power, 8 bit microcontroller. It features an on chip in system programmable flash memory of 16 KB and 1024 bytes of SRAM, 4 PWM channels all of which are used for motor control and 3 external interrupts which are used for reading the encoders. The controller has two modes of serial communication, the standard USART and the SPI protocol for master slave configuration. AtMega16 has a maximum throughput of 16 MIPS. Based on the RISC architecture, most of the commands are executed on single clock cycles. The combination of these features makes the controller suitable for control applications.

The joint angles from the inverse kinematics module are received by the AtMega16. Each of the motors is then rotated by their respective angles. This is done by implementing closed loop controls on each of the motor. The closed loop controls are implemented using the PI compensators.

5.1 PI Compensator

Each of the motors used has built in quadrature encoders. The encoders generate square waves at a frequency proportional to the angular velocity of the motor shaft. There is a phase difference of 90° between the waveforms of each of the two encoders.

The joint angles calculated by the inverse kinematics algorithm are translated into the encoder counts using the following equation.

$$r(t) = (\Phi_d / 360) * (ec * gr) \quad (6)$$

Where $r(t)$ =total counts

Φ_d =desired angle

ec =encoder counts per revolution

gr =gear ratio

The number of counts thus calculated is treated as the error. At time $t=0$ the error is maximum and is equal to the calculated counts. The proportional part of the controller ensures that the motor gets an output voltage proportional to the error. Greater the error, the greater is the applied voltage. As the motor rotates in the desired direction the error starts decreasing and hence the output voltage to the motors. During this motion the encoders generate the pulses which are counted. This is the real time position of the shaft and is represented by $y(t)$.

$$y_p(t) = K_p * \{r(t) - y(t)\} \quad (7)$$

Where K_p is the proportional gain

Since under loaded conditions the error can never be zero, the integral controller is implemented to eliminate this steady state error. The integral error is defined as:

And the output of the integral controller is

$$y_i(t) = K_i * \{e(t) + e(t-1)\} \quad (8)$$

The two expressions for the output are superposed to get a function that eliminates the error most efficiently.

The new output function is

$$y(t) = K_p * e(t) + K_i * \{e(t) + e(t-1)\} \quad (9)$$

6. Results

The design, fabrication and control of an autonomous robotic arm is a challenging task. The interfacing of a vision system makes the task more complicated. Results showed that the light-weight mechanical arm performed well when a PI controller is used. The efficient use of an isotropic material, such as aluminum, results in improved rigidity. The reduction of the link masses enables actuators to accelerate the joints very quickly. The image processing algorithm developed is fast enough to generate the coordinates of the object so that the time response of the system is not compromised. The errors in the calculated coordinates are within specified tolerances. The graph below shows the error trend in the calculated depth and the actual depth.

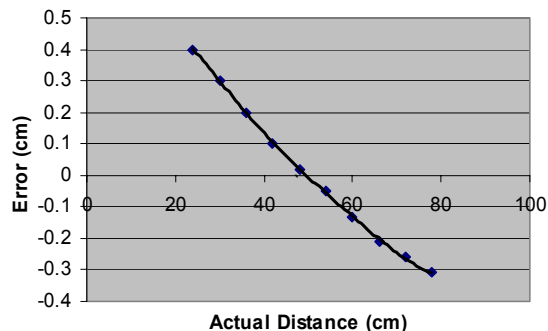


Figure 5 Error Trend in Range

Error can be reduced by increasing resolution, by decreasing viewing angle of cameras and by increasing distance between optical centers of cameras. The inverse kinematics code is efficient so as to ensure that the RAM of the microcontroller is not exceeded; the results have been verified by both algebraic and geometric methods. The PWM channels and the interrupts of AtMega16 made the implementation of PI control easy. The entire system performed as desired, the accuracy of the robotic arm was less than 0.5cm along each axis.

7. Conclusions

The control of light weight 4 DOF autonomous robotic arm has been described. To improve performance, the arm is made of aluminum. The coordinates of the object are generated by an algorithm using a vision system based on a pair of audio video cameras. These coordinates are then used to calculate the joint angles using an inverse kinematics code embedded in microcontroller At89C52. Position control has been implemented on these angles using a PI controller through the microcontroller AtMega16.

References

- [1] Michael P. Hennessey, Joseph A. Priebe, Paul C. Huang and Richard J. Grommes, "Design of a light weight robotic arm and controller", Proc of IEEE Work on Robotics Systems, pp. 779- 784, 1987.
- [2] John N. Lygouras, Basil G. Mertzios and Nicholas C. Voulgaris, Design and construction of a microcomputer-controlled light-weight robot arm, Robotics and Autonomous Systems 7, 269-283, 1991.
- [3] N. Ahuja, U. S. Banerjee, V. K. Darbhe, T. N. Mapara, A. D. Matkar, R. K. Nirmal, S. Balagopalan, "Computer Controlled Robotic Arm", Proceedings of the 16th IEEE Symposium on Computer-Based Medical Systems (CBMS'03), 2003.
- [4] Mana Saedan and Marcelo H. Ang Jr., '3d Vision Based Control on An Industrial Robot", Iasted International Conference On Robotics And Applications, Nov 19-22, 2001, Florida, USA.
- [5] Digital Image Processing By Rafeel C. Gonzalez, Richard E. Woods, 2nd Edition ,Pearson,2002
- [6] B.P Lathi, "Signals, Systems and Controls". Harper and Row, New York, 1974.
- [7] B.C Kuo, "Automatic Control Systems", Prentice-Hall, Eaglewood Cliffs, N.J .1982.4th Edition
- [8] www.Atmel.com last accessed on March 2005
- [9] www.jashaw.com last accessed on January 2005
- [10] John. J. Craig, "Introduction to Robotics", Mechanics and Control, 2nd Edition, Addison-Wesley, 1989.
- [11] www.dr-lex.34sp.com/qtvr/viewangle.html