VISION BASED ROBOT CONTROL USING GENERALIZED PREDICTIVE CONTROL

Raşit Köker[1] *e-mail: rkoker@sakarya.edu.tr*

Cemil Öz[2] *e-mail: coz@sakarya.edu.tr*

Recep Kazan[3] *e-mail: kazan@sakarya.edu.tr*

[1] *Sakarya University, Faculty of Technical Education, Department of Electronics & Computer Education*
[2] Sakarya University, Faculty of Engineering, Department of Computer Engineering
[3] *Sakarya University, Faculty of Engineering, Department of Mechanical Engineering,*
Sakarya, Turkey

Key words: Generalized predictive control, Robot control,Vision-based control

ABSTRACT

In this study, an efficient application of a vision based position control of a robotic manipulator using Generalized Predictive Control (GPC) is presented. The system consists of a camera, a capture card and a software that includes all dynamics and kinematics of the manipulator. The results of the simulations and algorithms used in this study are given.

I. INTRODUCTION

Generalized Predictive Control (GPC), introduced by Clarke and his coworkers in 1987, belongs to a class of digital control methods called Model-Based Predictive Control (MBPC). MBPC techniques have been analyzed and implemented successfully in process control industries since the end of the 1970's and continue to be used because they can systematically take into account real plant constraints in real-time. GPC is known to control non-minimum phase plants, open-loop unstable plants and plants with variable or unknown dead time. It is also robust with respect to modelling errors, over and under parameterization, and sensor noise. GPC has been originally developed with linear plant predictor models which leads to a formulation that can be solved analytically, but has also been proposed to be extended for the control of non-linear systems [1].

In recent years, due to the cost of growing raw materials, workmanship, energy and growing up competition environment, much effort has been directed to automating various production activities in industry by applying machine vision technology. The machine vision technology has begun to be developed rapidly, parallel to the development on the computer technology. Robots has also begun to be used with vision systems, such as an object picking up system using a manipulator and a vision system including a camera, a capture card and a software.

In this study, fistly the images are captured from the work area. These images are processed to seperate the object image from background and, then computed the centroid of the object. This position information is used in the simulations as a target position. The simulation software includes all dynamics and kinematics model of robotic manipulators. GPC algorithm, image processing algorithms and simulation results are presented in this paper.

II. GENERALIZED PREDICTIVE CONTROL ALGORITHM

Basically, GPC is the algorithm predicting y(t) output throughout finite horizon and, control signals string u (t) is computed using the minimization of a quadratic performans criterion according to the some assumptions interested in the control signals that implements approprate response. To design the controller, the discrete time CARIMA model is used in (1).

$$A(z^{-1}).y(t) = B(z^{-1}).u(t-1).e(t) \qquad (1)$$

The predictor: To make calculations with GPC, it is needed to have predictors throughout the chosen horizon ($t+1, t+N_2$). N_2 is known as maximum prediction horizon. The predictor for j step later may be given as,

$$y(t+j) = G_j.\Delta u.(t+j-1) + F_j.y(t) + E_j.\zeta.(t+j)$$
$$[I] = E_j(z^{-1}).A.(z^{-1}).\Delta + z^{-j}.F_j.(z^{-j})$$
$$E_j.(z^{-1}).B(z^{-1}) = G_j.(z^{-1}) \qquad (2)$$

Here, [I] is the definition matrix in the appropriate dimension. To express the predictions in more closed forms, the vectors below may be defined,

$$\hat{y} = [y(t+1)^T, ..., y(t+N_2)^T]^T$$

$$\tilde{u} = \left[\Delta u.(t)^T, \ldots, \Delta u.(t+N_2-1)^T\right]^T$$

$$f = \left[f(t+1)^T, \ldots, f(t+N_2)^T\right]^T \quad (3)$$

f (t+j), the component which is composed of the known signals of y (t+j) in an instant t. For instance;

$$f(t+1) = \left[G_1.(z^{-1}) - g_{10}.\Delta u(t) + F_1.y(t)\right] \quad (4)$$

where, $G_i(z) = g_{i0} + g_{i1}^{-1} + \ldots$

Using the information given above, the predictions may be shown in closed form as,

$$\hat{y} = G\hat{u} + f \quad (5)$$

here, $$G = \begin{bmatrix} G_0 & 0 & 0 & . & 0 \\ G_i & G_0 & . & . & 0 \\ . & . & . & . & \\ G_{N_2-2} & . & . & . & 0 \\ G_{N_2-1} & G_{N_2}-1 & . & . & G_2 \end{bmatrix}$$

The controller: The performance index of GPC is given below,

$$J = \sum_{j=1}^{N_2}\left\{[y(t+j) - w(t+j)]^T[y(t+j) - w(t+j)]\right\} + \sum_{j=1}^{N_2}\Delta u.(t+j-1).\Lambda(j).\Delta u.(t+j-1) \quad (6)$$

where, $\Lambda(j)$: The weight vector in control increases.
w : The reference trajectory.

Using the equations (5), the control law that minimizes the equation (6) may be obtained as,

$$\Delta u(t) = [I \quad 0 \quad 0 \quad .. \quad 0].(G.G^T + \Lambda)^{-1}.G^T(w - f) \quad (7)$$

$$\Lambda = diag[\Lambda(1), \Lambda(2), \ldots, \Lambda(N_2)]^T \quad (8)$$

The most strong way of GPC is the acceptances on the future control trajectories. To increase the stability feature of the control and to increase the computational complexity, the acceptance below is done (9) . Here, instead of making control increments free, after the chosen control horizon N_u , the control increments are accepted 0 [2]. Thus, the G matrix is converted to G_u as,

$$G_u = \begin{bmatrix} G_0 & 0 & 0 & 0 & . & . & . & 0 \\ . & . & . & . & . & . & . & 0 \\ . & . & . & . & . & . & . & G_0 \\ . & . & . & . & . & . & . & G_1 \\ . & . & . & . & . & . & . & . \\ G_{N_2-1} & . & . & . & . & . & . & G_{N_2}-N_u \end{bmatrix} \quad (9)$$

III. IMAGE PROCESSING

The captured real word images are processed by a few image processing algorithms before the calculation of the centroid. These processes which have been implemented by using Delphi Programming Language are given below.

NOISE REDUCTION

Noise reduction, in other words, filtering or smoothing is one of the most important processes in image processing systems. Images are often corrupted by positive and negative impulses stemming from decoding errors or noisy channels. An image can be degradated because of the undesirable effects due to illumination and other objects in the environment. Median filter is widely used for smoothing and restoring images corrupted by noise. It is a non-linear process useful especially in reducing impulsive or salt-and-pepper noise. In a median filter, a window slides along the image, and the median intensity of the pixels within the window becomes the output intensity of the pixel being processed. Different from linear filters such as the mean filter, median filter has attractive properties for supressing impulse noise while preserving edges. Median filter is used in this study due to the edge preserving feature of it [3][4][5][6]

THRESHOLDING

Many applications require the differentiation of the objects from background in the captured image data. Thresholding is easily appliable method for this purpose. Thresholding is choosing a threshold value and assigning 0 to the pixels with values smaller or equal to T and 1 to the ones with greater values than T. We are interested in dark objects on a light background, the parameter T, called the brightness threshold, is choosen and applied to the image f (x,y) as follows:

If $f(x,y) \geq T \quad f(x,y) = \text{background}$
Else $\quad f(x,y) = \text{object}$

As seen above, thresholding is a technique widely used in image segmentation [7][8].

EDGE DETECTION

Edge detection is a segmentation technique based on the detection of discontinuity. An edge is the place where there is a more or less abrupt change in gray level or color. In an image most of the information lies on the boundaries between different regions. Detecting edges is very usefull in a number of image processing contexts. Using edge map of images decreases the processing time in image processing systems. In many image understanding task such as our study described as object recognition , an essential step is to segment image into different two regions corresponding to object and background and then to find the edge map of the image to compute moment invariant feature vector set [3][8].

There are many edge detection algorithms used in the literature. They are generally based on laplacian or gradient. An application of Sobel algorithm that is based on gradient is presented below with its 3x3 masks.

Let $f(x,y)$ be the gray level of a pixel at (x,y) in the observed image. The gradient of $f(x,y)$ is given by

$$\nabla f(x,y) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} \tag{10}$$

where,

$$g_x = \sum_j \sum_i f(x+i, y+j).w_x(i,j), \tag{11}$$

$$g_y = \sum_j \sum_i f(x+i, y+j).w_y(i,j) \tag{12}$$

The horizantal and vertical Sobel edge operators $W_x(i,j)$ and $W_y(i,j)$, $-1 \le i,j \le 1$, are given in figure 1. The magnitude of the gradient is defined by

$$g(x,y) = |\nabla f(x,y)| = \left[g_x^2 + g_y^2\right]^{1/2}. \tag{13}$$

$g(x,y)$ forms the gradient image of the observed image $f(x,y)$. In figure 2, the image after the implementation of Sobel algorithm is given [9].

-1	-2	-1
0	0	0
1	2	1

(a) $W_x(i,j)$

-1	0	1
-2	0	2
-1	0	1

(b) $W_y(i,j)$

Figure:1 The horizantal (a) and vertical (b) of Sobel edge detectors.

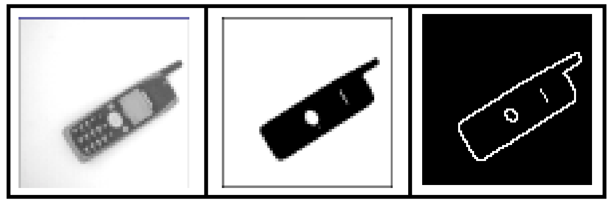

Figure: 2 Samples of processed images

CALCULATION OF THE CENTROID

The applications of moments provides a method of describing the properties of an object in terms of its area, position, orientation and other precisely defined parameters. The basic equation defining the moment of an object is given as below.

$$m_{ij} = \sum_x \sum_y x^i.y^j.f(x,y) \tag{14}$$

Where,

x ,y : Pixel coordinates
f(x,y) : Pixel brightness

Zero- and first order moments can be given as;

$$m_{00} = \sum_x \sum_y f(x,y) \tag{15}$$

$$m_{10} = \sum_x \sum_y x.f(x,y) \tag{16}$$

$$m_{01} = \sum_x \sum_y y.f(x,y) \tag{17}$$

The ‘centroid’ is a good parameter for specifying the location of an object. It is the point having coordinates x', y' such that the sum of the square of the distance from it to all other points within the object is a minimum[10][11]. The centroid can be expressed in terms of moments as,

$$x' = \frac{m_{10}}{m_{00}} \tag{18}$$

$$y' = \frac{m_{01}}{m_{00}} \tag{19}$$

IV. EXPERIMENTAL SIMULATIONS

The parameters of manipulator used in the simulations are shown in table 2, and its model is given in figure 4. The simulations were done using computed object centroids (x,y) via image processing algoritms. The third dimension z is accepted known for the object. Obtained (x,y,z) coordinates is the final position of end effector.

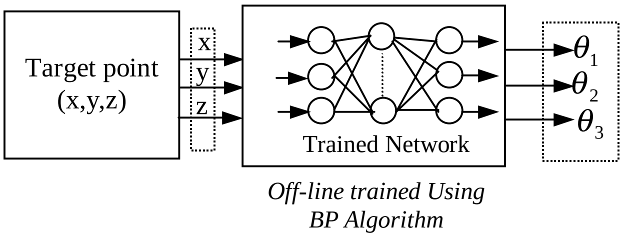

Figure: 3 The inverse kinematics solution scheme.

The object position is used as target position. GPC is applied to take the end effector to target position. The (x,y,z) point is firstly transformed to the angular position (θ_1, θ_2, θ_3). This is known as inverse kinematics solution. We have done inverse kinematics solution using neural network, and the scheme for inverse kinematics solution is given in figure 3 [14].

i	joint1	joint2	joint3	units
m_i	13.1339	10.3320	6.4443	kg.
a_i	0.1588	0.445	0.10	m.
α_i	$\pi/2$	0.0	$\pi/2$	Radyan
X^*_i	-0.0493	-0.1618	0.0	m.
z^*_i	0.0	0.0	0.2718	m.
k_{i11}	5.6064	3.929	82.0644	m^2x10^{-3}
k_{i22}	8.9196	47.8064	81.9353	m^2x10^{-3}
k_{i33}	13.2387	45.4838	1.400	m^2x10^{-3}

Table: 2 The parameters of the used robotic manipulator

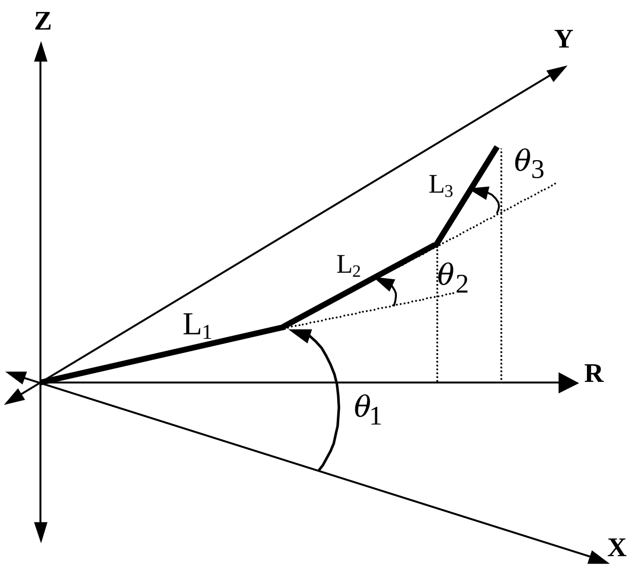

Figure: 4 The manipulator model used in this study

V. RESULTS AND DISCUSSION

The simulation results for each joint are given below. In figure 5, the position results for each joint are given. In figure 6, the results for speed are also given.

As a result of these simulation studies, the curves refer to joint position and speed; in spite of the disturbances and diffucult trajectories the position control is implemented perfectly. The observed error at the position result of joint 2 is acceptable.

In this study, illumination is also important due to the reflectance and shadow. The illumination is done using two light sources to obtain image without shadow and reflactance. It is observed that noise reduction removed some of these factors.

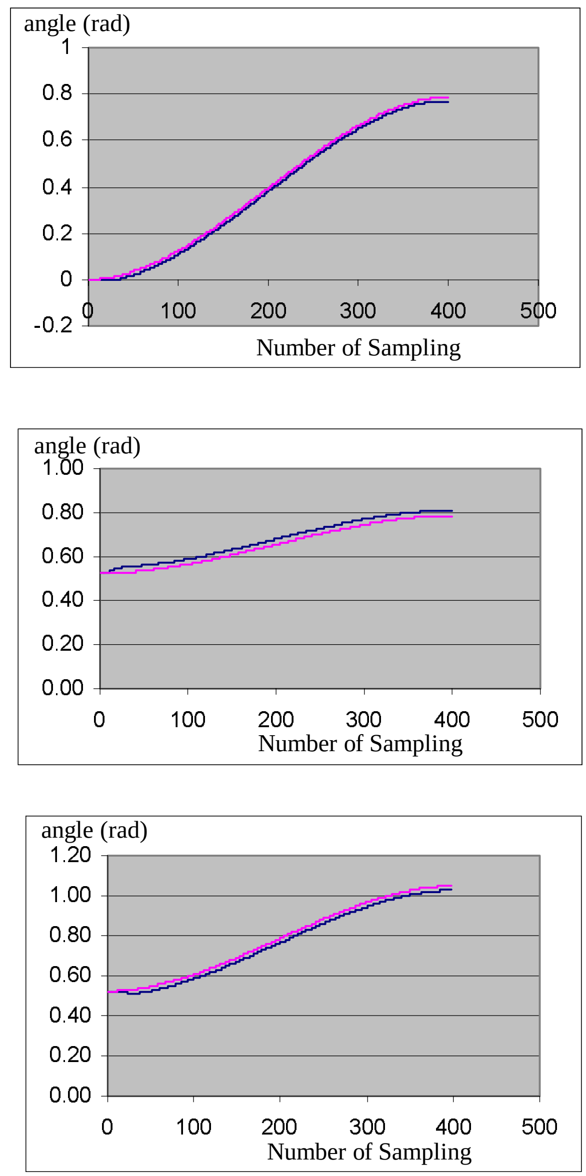

Figure: 5 The results of position for each joint.

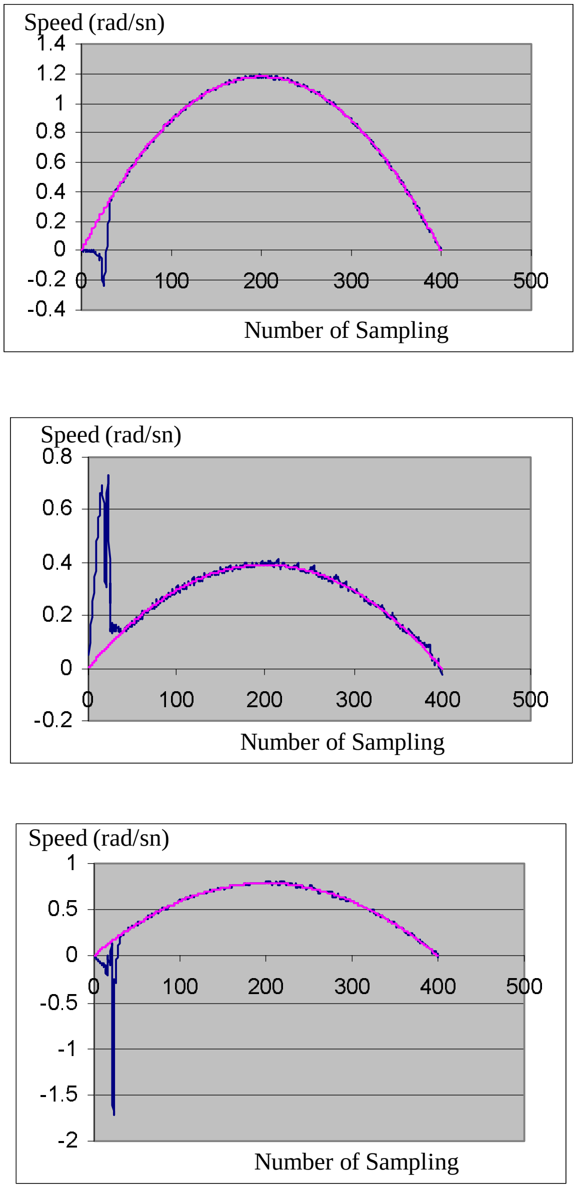

Figure: 6 The results of speed for each joint.

VI. CONCLUSION

This paper has given an efficient application of GPC algorithm applied to the vision based robotic manipulator control. The simulation results showed the GPC algorithm is good at control of robotic manipulator.

REFERENCES

1. D. Soloway, P. Haley, Neural/Generalized Predictive Control,a Newton-Raphson Implementation, Proc. of The Eleven IEEE Int. Symp. on Intel. Cont., pp. 277-282, 1996.
2. R. Kazan, C. ÖZ, Altı Eklemli Bir Robotik Manipülatörün (MIMO) Genelleştirilmiş öngörülü Kontrol Algoritması ile Konum Kontrolü, III. Balıkesir Mühendislik Mimarlık Sempozyumu, pp.121-130, 1992.
3. J.S., Lim, Two-Dimensional and Image Processing, Prentice-Hall, 1990.
4. X. Yang, P.S. Toh, Adaptive Fuzzy Multilevel Median Filter, IEEE Transaction on Image Processing, Vol.4, No.5, pp.680-682, May 1995.
5. H. Hwang, R.A. Haddad, Adaptive Median Filters: New Algorithm and Results, Transactions on Image Processing, Vol. 4, No. 4, pp.499-505, April 1995
6. A. Rosenfeld, Digital Picture Processing, Academic Press Inc., 1982.
7. U.M. Erdem, 2D Object Recognition In Manufacturing Environment Using Implicit Polynomials and Algebraic Invariants, Master Thesis, Boğaziçi University, 1997.
8. K.S. Fu, J.K. Mui, A Survey On Image Segmentation, Pattern Recognition, Vol.13, pp.3-16, Pergamon Press, 1981.
9. D.M. Tsai, C.T. Lin, A Moment Preserving Approach For Depth From Defocus, Pattern Recognition, Vol.31, No.5, pp.551-560, 1998.
10. G. J. Awcock, R. Thomas, *Applied Image Processing*, McGraw Hill, Inc., 1996, p. 148.
11. G. I. Chicu, J. Hwang, A Neural Network-Based Stochastic Active Contour Model (NNS-SNAKE) for Contour Finding of Distinct Features, IEEE Transactions on Image Processing, Vol.4, No.10, October 1995, pp. 1407-1416.
12. R. Köker, C. Öz, A. Ferikoğlu, Development of a Vision Based Object Classification System For an Industrial Robotic Manipulator, The 8th IEEE Intern. Conf. on Electronics, Circuits and Systems, Malta, 2001.
13. R. Köker, C. Öz, A. Ferikoğlu, Object Recognition Based on Moment Invariants Using Neural Network, 3rd Intern. Symp. on Intelligent Manufacturing Systems, IMS 2001, Sakarya, Turkey, accepted paper.
14. R. Köker, C. Öz, T. Çakar, A Study of Neural Network Based Inverse Kinematics Solution For 3-Joint Robot, 3rd Intern. Symp. on Intelligent Manufacturing Systems, IMS 2001, Sakarya, Turkey, accepted paper.