An Auto-Tuning Fuzzy Rule-Based Visual Servoing Algorithm for a Slave Arm

Joo Gon Kim*, Dong Hyuk Cha*, Hyung Suck Cho* and Seung Ho Kim**

*Dept. of Mechanical Engineering, Korea Advanced Institute of Science and Technology
373-1 Kusongdong, Yusonggu, Taejon, Korea

**Korea Atomic Energy Research Institute
150 Dukjindong, Yusonggu, Taejon, Korea

Abstract

In telerobot systems, visual servoing of a task object for a slave arm with an eye-in-hand camera has drawn an interesting attention. As such a task is generally conducted in an unstructured environment, it is very difficult to define the inverse feature Jacobian matrix. To overcome this difficulty, this paper proposes an auto-tuning fuzzy rule-based visual servo algorithm. In this algorithm, a visual servo controller composed of fuzzy rules, receives feature errors as inputs and generates the change of slave position as outputs. The fuzzy rules are tuned by using steepest gradient method of the cost function, which is defined as a quadratic function of feature errors. Since the fuzzy rules are tuned automatically, this method can be applied to the visual servoing of a slave arm in real time. The effectiveness of the proposed algorithm is verified through a series of simulations and experiments.

1. Introduction

The technology of teleoperation applies to various operations in hostile environments where human can hardly work.A teleoperation system is generally composed of a master arm which is controlled by a human operator and a slave arm which performs actual operator and a slave arm which performs actual operations by duplicating the motions of the master arm.

Recently, owing to the brilliant advance of the robot manipulator technology, many researchers have been studied to assign autonomous working ability for the slave arm in order to improve the task performance. Among the many sensor information, such as visual information, force, tactile, voice, etc., visual information can be effectively used for the autonomous teleoperation. The method of using the visual information in the telerobot system can be devided into two catagories. The first one is to construct a three-dimensional image using the visual information obtained from the camera mounted at the site of the slave arm. Then the three-dimensional image is fed back to the operator[1]. The second one is to move the slave arm autonomously without aid of the operator by using the visual servo method[2].

In the field of robotics the visual servo method can be classified into a position-based visual servoing, and a feature-based visual servoing method. In the position-based visual servoing, 3-D position of target objects is calculated from the visual information, and then, the robot moved based upon the calculated position.

In the feature-based visual servoing, the robot moves based on the simple image feature obtained from the camera image. As this method does not require calculation of the position of target, servoing can be conducted in real time. Thus, many researchers have been studied this method such as: the research of choosing the method of model reference adaptive control by Sanderson[3], of generating the visual servo system with information from the accurate CAD model of the target object[4], of the visual servo system using the uncalibrated camera[5], of visual servo using normalized feature matrix[6].

In the above methods, it is required to obtain the feature Jacobian matrix, which is a linearized relationship between the change of image features and the corresponding motion of the robot manipulator. However, it is very difficult to obtain the feature

0-7803-2722-5/95 $4.00 © 1995 IEEE

Jacobian precisely. Moreover, as the Jacobian is obtained via a linearization, it contains linearization error. To overcome these problems a visual servo algorithm which does not use the feature Jacobian is proposed in this paper. In this algorithm, a visual servo controller composed of fuzzy rules receives feature errors as inputs and generates the change of slave position as outputs. The fuzzy rules are tuned by using steepest gradient method of the cost function, which is defined as a quadratic function of feature errors.

The effectiveness of the proposed algorithm is verified through a series of simulations and experiments.

This paper is composed as follows: A telerobot system used in the study is descried in section 2, an auto-tuning fuzzy rule-based visual servo algorithm is described in section 3, experimental results are in section 4. Finally some conclusions are made in section 5.

2 Description of a telerobot system

Fig. 1 shows the configuration of a telerobot system developed at the Laboratory for Control system and Automation in the KAIST[8]. The telerobot system consists of a force reflective master arm, a slave arm, a visual sensor, an image processing system, and a system controller. The master arm has a vertical articulated structure with 3 degree of freedom(DOF), the slave arm is an industrial robot(Samsung, FARA A1-U). The visual sensor is a CCD camera (SONY, XC-77RR) and a CCTV camera. The image process system is an industrial vision board(Samsung, MVB-02) with a DSP chip .

The whole teleoperation procedure is divided into two phases. In the first phase, the operator moves the master arm while monitoring the working environments with the aid of a CCTV camera. Then the slave arm moves near the object point, the second phase starts. In the second phase, the slave arm does not duplicate the motion of the master arm, but moves automatically according to the proposed servoing algorithm.

3 The auto tuning fuzzy rule-based Visual servoing algorithm

To track a target object in real time, a visual algorithm using an auto-tuning fuzzy rule is presented in this paper.

3.1 Inputs and outputs of the fuzzy logic controller

The fuzzy controller shown in Fig. 2 receives the feature error vector as inputs, and generates the displacements for the slave arm to moves outputs. Let the desired image feature vector, $\vec{F}_{ref}$, be represented by

$$\vec{F}_{ref} = (f_{1,ref}, f_{2,ref}, f_{3,ref})^T, \qquad (1)$$

where $f_{i,ref}$(i=1, 2, 3) is reference image feature. Also let the current image vector $\vec{F}_{act}$ be represented by

$$\vec{F}_{act} = (f_{1,act}, f_{2,act}, f_{3,act})^T \qquad (2)$$

where $f_{i,act}$(i=1, 2, 3) is actual image feature. As shown in Fig. 2, a feature error, $\vec{E}_F$, that represents the difference between the two vectors in (1) and (2) is utilized as the input to the controller. The $\vec{E}_F$ is expressed by

$$\vec{E}_F = \vec{F}_{ref} - \vec{F}_{act} \qquad (3)$$

The output of controller is represented by a vector, $\Delta\vec{X}$, which is displacements for the slave arm to move, and expressed by

$$\Delta\vec{X} = [\Delta x_1, \Delta x_2, \Delta x_3]^T \qquad (4)$$

where $\Delta\vec{X}$ is denoted the displacements by which the slave arm is to be move.

3.2 Rule base

Let a fuzzy value for the feature error, e_{f_i}, be denoted as $\tilde{e}_i$. The $\tilde{e}_i$ can be expressed by

$$\tilde{e}_i = g_i \times e_{f_i} \qquad (5)$$

where, e_{f_i} denotes the feature errors, and g_i is a scale factor between the feature error and fuzzy rule input($\tilde{e}_i$).

The rule base is composed of the rule that define the relationship of the feature error and the corresponding motion of the robot manipulator.

$$\begin{aligned} &RULE\ \text{k}: IF\ \tilde{e}_1\ is\ \mathrm{E}_1^k,\ \tilde{e}_2\ \ is\ \mathrm{E}_2^k, \tilde{e}_3\ \ is\ \mathrm{E}_3^k \\ &\qquad THEN\ u_1\ is\ \mathrm{U}_1^k\ u_2\ is\ \mathrm{U}_2^k, u_3\ is\ \mathrm{U}_3^k \\ &\qquad (\,\mathrm{k}=1,2,\cdots,l) \end{aligned} \tag{6}$$

where, l is the number of rules. $\tilde{e}_i$ (i=1,2,3) is a feature error, E_i^k is the fuzzy value for the $\tilde{e}_i$, u_i is the output variable, and U_i^k is the fuzzy value for each u_i.

3.3 Fuzzy inference and defuzzification

The product sum method[9] is used as the fuzzy inference in this study. This method has advantages of both relatively short execution and simple learning. As a defuzzification, the center of gravity method is adapted. The displace Δx_i (i=1,2,3) resulting from the inference and the fuzzification can be expressed by

$$\Delta x_i = \frac{h_1 \mathrm{U}_i^1 + h_2 \mathrm{U}_i^2 + \cdots + h_l \mathrm{U}_i^l}{h_1 + h_2 + \cdots + h_l} \tag{7}$$

where, i denotes the direction of reference coordinates, and h_k denotes the product between each degree of fitness when fuzzy input is $\tilde{e}_i$ (i=1, 2, 3), is expressed by

$$h_k = \mu_{E_1^k}(\tilde{e}_1) \cdot \mu_{E_2^k}(\tilde{e}_2) \cdot \mu_{E_3^k}(\tilde{e}_3) \tag{8}$$

where $\mu_{E_i^k}(\tilde{e}_i)$ (i=1,2,3) is the degree of fitness when fuzzy input is $\tilde{e}_i$.

3.4 The auto-tuning algorithm

To adjust the membership value of output variables, an auto-tuning algorithm is used. In this algorithm, rule is tuned such that a cost function is minimized by the steepest gradient method. The cost function, J, is defined by

$$J = \frac{1}{2}[\alpha_1 (e_1)^2 + \alpha_2 (e_2)^2 + \alpha_3 (e_3)^2] \tag{9}$$

where α_i (i=1,2,3) is the weighting factor for each $\tilde{e}_i$. To minimize the J, the output fuzzy value is updated as follow:

$$U_i^k(t+1) = U_i^k(t) - \eta \frac{\partial J(t)}{\partial U_i^k(t)} \tag{10}$$

where η denotes positive learning rate, and t is sampling time. The term $\frac{\partial J(t)}{\partial U_i^k(t)}$ in the above equation can be obtained by a chain rule, i.e.,

$$\frac{\partial J(t)}{\partial U_i^k(t)} = \frac{\partial J(t)}{\partial e_1(t)} \cdot \frac{\partial e_1(t)}{\partial U_i^k(t)} + \frac{\partial J(t)}{\partial e_2(t)} \cdot \frac{\partial e_2(t)}{\partial U_i^k(t)} + \frac{\partial J(t)}{\partial e_3(t)} \cdot \frac{\partial e_3(t)}{\partial U_i^k(t)} \tag{11}$$

where the term $\frac{\partial e_j(t)}{\partial U_i^k(t)}$ is expressed by

$$\frac{\partial e_j(t)}{\partial U_i^k(t)} = \frac{\partial e_j(t)}{\partial \Delta x_1(t)} \cdot \frac{\partial \Delta x_1(t)}{\partial U_i^k(t)} + \frac{\partial e_j(t)}{\partial \Delta x_2(t)} \cdot \frac{\partial \Delta x_2(t)}{\partial U_i^k(t)} + \frac{\partial e_j(t)}{\partial \Delta x_3(t)} \cdot \frac{\partial \Delta x_3(t)}{\partial U_i^k(t)} \tag{12}$$

using the equation (6), the term $\frac{\partial \Delta x_i(t)}{\partial U_i^k(t)}$ (i=1, 2, 3) obtained by

$$\frac{\partial \Delta x_i(t)}{\partial U_i^k(t)} = \frac{h_k}{h_1 + h_2 + \cdots + h_l} \tag{13}$$

By using equations(6), (11),(12) and (13), the output fuzzy value U_i^k in equation (10) is updated. Through this procedure, the fuzzy output variable is tuned to decrease the feature error. In the next section, this proposed algorithm is applied to the prescribed telerobot system.

4 Results and discussions

The proposed visual servoing algorithm is applied to track a cubic objective. To show the efficiency of the proposed algorithm, the results are compared with those of the inverse-feature Jacobian method.

4.1 The experimental conditions

Fig. 3 shows the reference coordinate and the camera coordinate. The robot moves with respect to x, y, z-coordinate at a fixed pose. Fig. 4 shows the image features. The condition of experiment 1 is shown in table. The fuzzy input membership function is shown in Fig. 5. and the total number of the rules is 343. The initial values of fuzzy output variable are randomly chosen to be a value between -1 and 1.

4.2 The experimental results

To learn the fuzzy rule, the 50 trials of tracking a target were carried out at the same initial position. At the first learning, the change in the feature error is shown in Fig. 6(a)~(c). and the tracking path is shown Fig. 6(d)~(f). It can be seen from the figures that the feature error does not change not converge to zero and the tracking path is not smooth. At the 20th learning, the change in the features is shown in Fig. 7(a)~(c), and tracking path is shown in Fig. 7(d)~(f). At the 50th learning, the change in the feature errors is shown in Fig. 8(a)~(c), and the tracking path is shown in Fig. 8(d)~(f). It can be seen from figures 7 and 8 that the feature error converges to zero at the 20th and 50th learning steps. In the experiment, it is observed that, form the initial to the final point of moving, 48 sec. and 8 sec are required at the first and 50th learning step, respectively. Furthermore, as the learning step increases, the moving path becomes straight.

4.3 The comparison with a conventional method

The results of visual tracking using the feature Jacobian matrix are employed Fig. 9. The path is not straight line. The reason is that the model of the pin hole camera doesn't coincide with the real system. When the result compares with that in Fig. 6, the method of auto-tuning fuzzy rule have the shorten path.

5 Conclusions

In this paper, the visual servo algorithm using the auto-tuning fuzzy rule is proposed. The proposed algorithm is applied to track the target object in space and from the results some conclusions can be made as follows:

1) It takes much times to track the target object with the initial rule, but with the learned rule tracking does successfully in real time.
2) The error of measurement in the calibration and the error generating when the feature Jacobian matrix is calculated don't take place.

References

[1] S. H. Lee, "Intelligent Sensing and Control for Advanced Teleoperation", Proc. IEEE Int. Sym. on Int. Con., pp19-28, June, 1993.

[2] Hasegawa, Suehiro, Ogasawara, "An Integrated Tele-Robotics System With a Geometric Environment Model and Manipulation Skills", IEEE Int. Work. on Int. R. and Sys., pp335-341.

[3] E. Lee, Sanderson and P. Charles, " Dynamic Sensor-Based Control of Robot with Visual Feedback", IEEE Journal of Robotics and Automation, vol, RA-3, NO.5, pp. 404-417, Oct, 1987.

[4] Feddema, T. John, and O. R. Mitchell, "Vision-Guided Servoing with Feature-Based Trajectory Generation", IEEE Tr. on Robotics and Automation, vol, no.5, pp. 691-700, Oct., 1989.

[5] B. H. Yoshimi, and P. K. Allen, "Active, Uncalibrated Visual Servoing", Proc. 1994 IEEE Int. Conf. on Robotics and Automation, pp. 156-161, IEEE, May, 1994.

[6] W. Jang, "Visual Servoing of an Eye-in-Hand Robot based on Features", Ph. D dissertation, KAIST, 1991.

[7] J. G. Kim, D. H. Cha, J. S. Kim, H. S. Cho, S. H. Kim, "Visual Feedback Control of a Telerobot System with an Uncalibrated Camera", Proceedings of the 9th Korea Automatic Control Conference, pp473~478, 1994.

[8] C. O. Lee, H. S. Cho, and P. H. Chang, et.al. "A Study on the manipulation of Teleoperation System Using Redundant Robot", The Development of Advanced Robotics for the Nuclear Industry, 1994.

[9] M. Mizumoto, "Realization of PID Controls by Fuzzy Control Methods", IEEE Int. Conf. on Fuzzy System, pp.709-715, 1992.

Table 1 The condition of the experiment

Area of target object(Area)	640 mm^2 (experimentation)
focal length	12mm
desired feature	$\vec{F}_{ref} = (\ 0,\ 0,\ 55\ )^T$

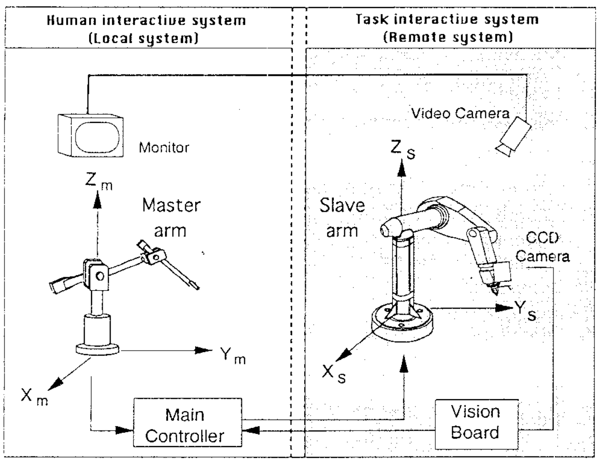

Fig. 1 The configuration of the telerobot system

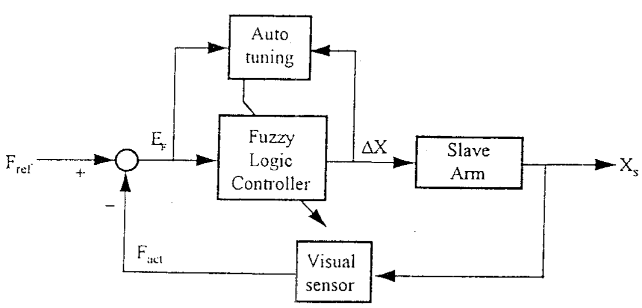

Fig. 2 The block diagram of an auto-tuning fuzzy rule-based visual servo system

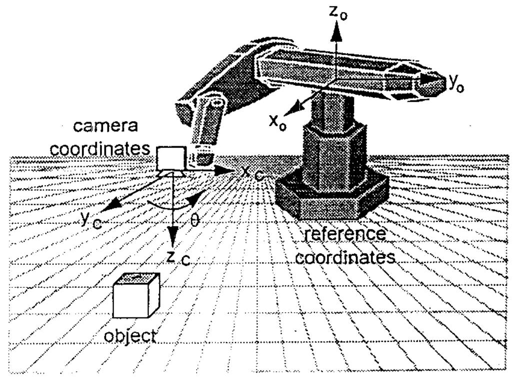

Fig. 3 The reference coordinates and the camera coordinates

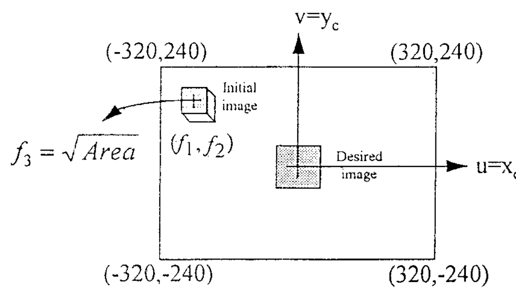

Fig. 4 features of the object image

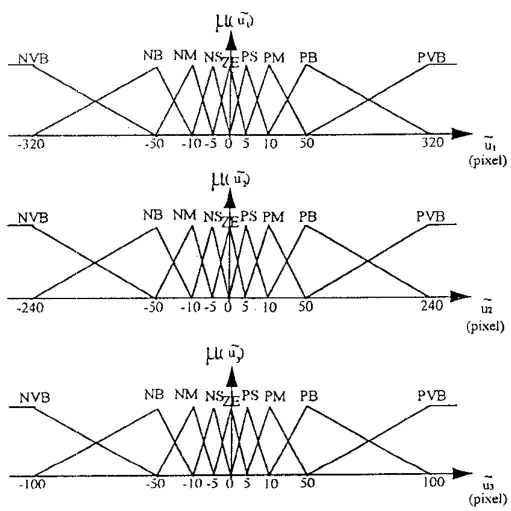

Fig. 5 Fuzzy input menbership functions

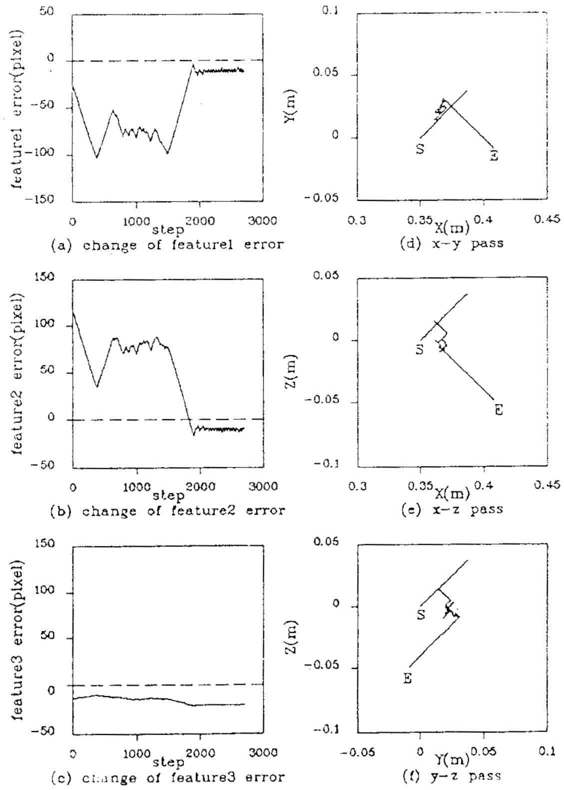

Fig. 6 The change of the feature errors and the corresponding path of the slave arm (number of learn =0)

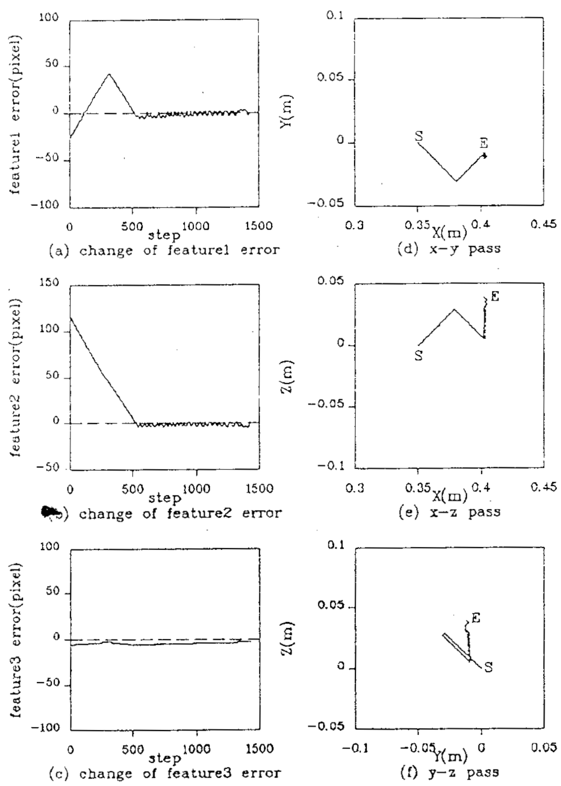

Fig. 7 The change of the feature errors and the corresponding path of the slave arm (number of learn =20)

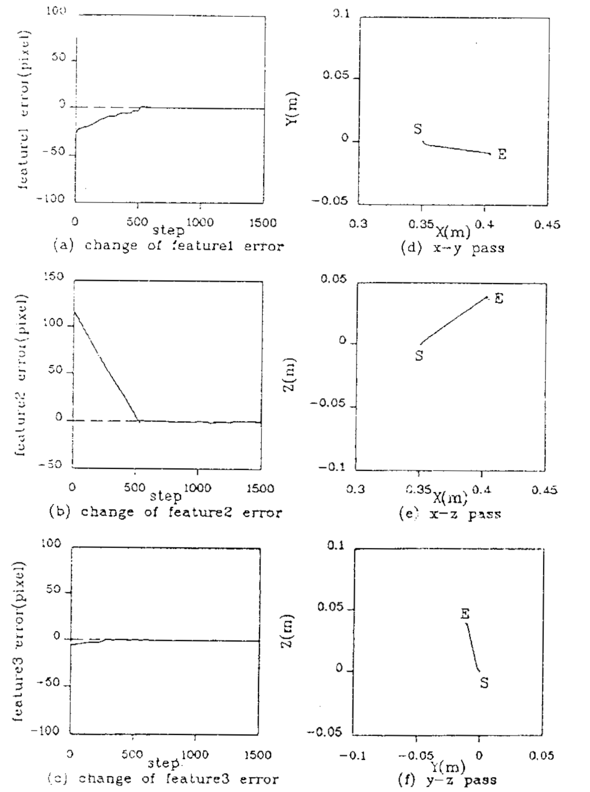

Fig. 8 The change of the feature errors and the corresponding path of the slave arm (number of learn =50)

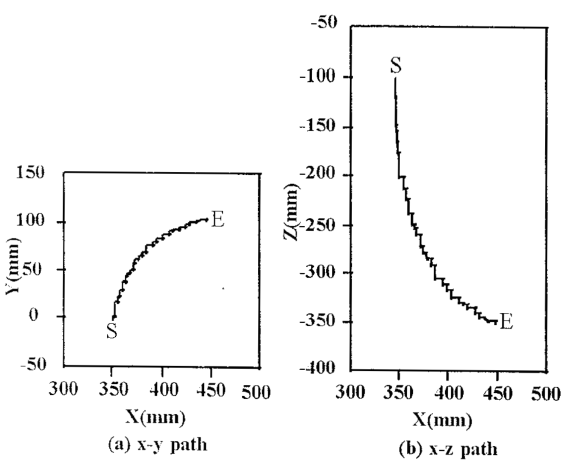

Fig. 9 The experiment results of the slave arm when Jacobian matrix is used