Experimental Results for a LMI-Based Fuzzy Visual Servoing Controller Applied in Robots Manipulators in 2D

Maximiliano Bueno López[1]
Marco Antonio Arteaga Pérez[2]
Fernando Lizarralde[3]

ABSTRACT
In this paper we consider a LMI-Based Fuzzy Visual Servoing controller for planar robot manipulators in the presence of parametric uncertainties associated with robot dynamics and camera parameters. For tracking purposes, a Takagi-Sugeno controller is used, based on the dynamic model of the robot in image coordinates. The design includes the use of Lyapunov functions as a fuzzy mixture of multiple quadratic functions for stability proof. Experimental results on two different robots (LAW3 SCHUNK and A465 CRS Robotics) show the good performance of the complete system.

Keywords: Visual servoing control, robots manipulators, fuzzy logic.

[1] Doctor en Ingeniería–Universidad Nacional Autónoma de México. Profesor Asociado II, Facultad de Ingeniería, Programa de Ingeniería Eléctrica, Universidad de La Salle. E-mail: maxbueno@unisalle.edu.co.
[2] Doctor en Ingeniería–Universidad Gerhard Mercator de Duisburg, Alemania. Profesor Titular C, Departamento de Control y Robótica, División de Ingeniería Eléctrica de la Facultad de Ingeniería, Universidad Nacional Autónoma de México. E-mail: marteagp@unam.mx
[3] Doctor en Ingeniería Eléctrica–Universidad Federal de Río de Janeiro. Profesor Asociado, Departamento de Ingeniería Eléctrica COPPE/UFRJ, Universidade Federal do Río de Janeiro, Brasil. E-mail: fernando@coep.ufrj.br.

Reception: January 23th, 2014 • Approved: April 1st, 2014

How to cite this article: Bueno López, M., Arteaga Pérez, M. A. & Lizarralde, F. (2014). Experimental Results for a LMI-Based Fuzzy Visual Servoing Controller Applied in Robots Manipulators in 2D. *Épsilon* (22), 151-168.

Resultados experimentales para un controlador servovisual difuso basado en LMI aplicado en robots manipuladores en 2D

RESUMEN

En este artículo se considera un controlador servovisual difuso basado en LMI para robots manipuladores planares en presencia de incertidumbre en los parámetros asociada con la dinámica del robot y los parámetros de la cámara. El diseño incluye el uso de funciones de Lyapunov que son una mezcla de múltiples funciones cuadráticas para la prueba de estabilidad. Resultados experimentales sobre dos diferentes robots (LAW3 SCHUNK y A465 CRS Robotics) muestran el buen comportamiento del sistema completo.

Palabras clave: control servovisual, lógica difusa, robots manipuladores.

Resultados experimentais para um controlador servo visual difuso baseado em LMI aplicado em robôs manipuladores em 2D

RESUMO

Neste artigo se considera um controlador servo visual difuso baseado em LMI para robôs manipuladores planares em presença de incerteza nos parâmetros associada com a dinâmica do robô e os parâmetros da câmara. O desenho inclui o uso de funções de Lyapunov que são uma mescla de múltiplas funções quadráticas para a prova de estabilidade. Resultados experimentais sobre os diferentes robôs (LAW3 SCHUNK e A465 CRS Robotics) mostram o bom comportamento do sistema completo.

Palavras chave: controle servo visual, lógica difusa, robôs manipuladores.

Introduction

The use of cameras in the control of robot manipulators has increased in recent years, not only at the research level, but also in industry. Initially, this strategy was not very interesting for tracking controller due to the possibility of obtaining good results simply by using available encoders at each joint. It has been shown that the use of cameras reduces these errors and improves performance, which is important in some applications that require maximum accuracy. The first visual control structure was introduced in Sanderson and Weiss (1980), and Weiss, Sanderson and Neuman (1985 and 1987) consolidated the difference between the two main control schemes of visual servoing: position-based visual servoing (PBVS) and image-based visual servoing (IBVS) (Hutchinson & Chaumette, 2007). Servovisual systems are limited due to restrictions of the field of view and low sampling rates. One solution includes sensors of different kinds to provide more information. The first robot control systems based on vision were reported in the early 1970's (Shirai & Inoue, 1973). In the 1980's the progress in visual servoing was very slow, but from the early 1990's the development greatly increased. The term visual servoing was introduced in Hill and Park (1979) to replace the term visual feedback. Different techniques, such as PID, adaptive and robust control and artificial intelligence, can be combined with vision systems as seen in Lizarralde, Hsu and Costa (2008); Weng, Hui and Chen (2010); Hsu, Costa and Lizarralde (2007), and Liu, Cheah and Slotine (2006).

There are several works which use fuzzy logic in the control of robot manipulators (Ma, Sun & He, 1998; Korba, Babuska & Verbruggen, 2003). Nevertheless, combining these techniques with visual servoing has been barely studied; only few publications have addressed this issue (Alavandar & Nigan, 2008; Choi, 2007). The main contribution of this paper is to present experimental results using two different robots to validate the fuzzy visual servo control design.

Fuzzy Control Design Applied to Robot Manipulators

During the last decade, fuzzy logic control has attracted great attention both from academic and industrial communities. Many researchers have dedicated a great deal of time and effort to theoretical research and implementation techniques for fuzzy controllers; up to date, fuzzy logic has been suggested as an alternative approach to conventional control techniques for complex systems, being one of the most useful approaches for taking advantage of the qualitative knowledge in controller design. Among the model based fuzzy control approaches, the Takagi Sugeno (T–S) is one of the most effective strategies (Tanaka, Ikeda & Wang, 1996; Wang, Tanaka & Griffin, 1996). Takagi and Sugeno proposed their model in 1985, and it has emerged as one of the most active areas of research. In the design of fuzzy controllers a good option is to use parallel distributed compensation (PDC). PDC offers a procedure to design a fuzzy controller from the T–S model. In contrast to conventional control, fuzzy logic was initially introduced as model free based on a representation of the knowledge of human operators. In de Jong & Pieter (2000), a visual servoing control scheme is proposed for positioning a robot manipulator, divided into two steps: first, a fuzzy logic-based visual servoing controller is used to guide the gripper into the neighborhood of the object, and then a local neural network is applied to set the gripper in a desired position. In Yow Lian, Tu and Liou (2006) and Bernal and Guerra (2010), concepts of stability are applied for fuzzy logic controllers using LMI. Fuzzy logic began to be used for the design of observers not long ago, and it is a strategy that is currently under research (Zhang & Fei, 2006). One of the few works on this subject is presented in Grande-Meza (2003), and different designs applied to several nonlinear fuzzy observers are shown with excellent results.

Preliminaries

The dynamics of a rigid robot arm with revolute joints can adequately be described by using the Euler-Lagrange equations of motion (Sciavicco & Siciliano, 2000), resulting in

$$H(q)\ddot{q}+C(q,\dot{q})\dot{q}+D\dot{q}+g(q)=\tau-\tau_p \tag{1}$$

where $q \in \mathbb{R}^n$ is the vector of generalized joint coordinates, $H(q) \in \mathbb{R}^{nxn}$ is the symmetric positive definite inertia matrix, $C(q,\dot{q})\dot{q} \in \mathbb{R}^n$ is the vector of Coriolis and centrifugal torques, $g(q) \in \mathbb{R}^n$ is the vector of gravitational torques, $D \in \mathbb{R}^{nxn}$ is the positive semidefinite diagonal matrix for joint viscous friction coefficients, $\tau \in \mathbb{R}^n$ is the vector of torques acting at the joints, and $\tau_p \in \mathbb{R}^n$ represents any bounded external perturbation or friction force.

Throughout this paper, we will assume that the robot is a 2DOF planar manipulator, that is, it is n = 2 in (1). The direct kinematics is a differentiable map $f_k(q)$: $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ relating the joint positions $q \in \mathbb{R}^2$ to the Cartesian position $x_R \in \mathbb{R}^2$ of the centroid of a target attached at the arm end effector as

$$x_R = f_k(q) \tag{2}$$

The output of the system is the position $y \in \mathbb{R}^2$ of the image feature, that is, the position of the target in the computer screen. The robot workspace is the *x–y* plane. The camera is not assumed to be completely parallel to this plane, so that there are two angles of rotation $\phi, \psi \in \mathbb{R}$, associated to the map from image to workspace coordinates. In order to write y in terms of x_R, the image feature y can be computed through transformation and perspective projections. By computing the derivative of position we can get the differential perceptual kinematic model given by

$$y = \alpha\lambda R_\phi z_R \tag{3}$$

where λ is the focal length, α is a conversion factor from meters to pixels. It is considered $J(q) = \partial f_k(q)/\partial q$ that it is called geometrical Jacobian matrix of the robot which satisfies (Sciavicco & Siciliano, 2000)

$$\dot{x}_R = J(q)\dot{q} \Rightarrow \dot{q} = J^{-1}(q)\dot{x}_R \tag{4}$$

R_ϕ represents the orientation of the camera frame with respect to the robot frame,

$$R_\phi = \begin{bmatrix} \cos(\phi) & \sin(\phi) \\ \sin(\phi) & -\cos(\phi) \end{bmatrix} \tag{5}$$

where $\phi \in \mathbb{R}$ is the angle of rotation. Note that $R_\phi^{-1} = R_\phi^T = R_\phi$.

From (3)

$$\dot{q} = \frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \dot{y} \tag{6}$$

Equation (6) defines the visual mapping that relates the velocity in Cartesian space with the velocity in image space. The process for obtaining the position of the end effector in image coordinates is shown in detail in Bueno-López and Arteaga Pérez (2013).

To use a visual control algorithm is necessary to rewrite the model of the robot manipulator in terms of image coordinates y. By computing the derivative of (6), we obtain

$$\ddot{q} = \frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \ddot{y} + \frac{1}{\alpha\lambda} \dot{J}^{-1}(q) R_\phi \dot{y} \tag{7}$$

where

$$\dot{J}^{-1}(q) = \frac{d}{dt} J^{-1}(q) \tag{8}$$

We can now substitute these expressions into the manipulator dynamics (1) and pre-multiply by $R_\phi J^{-T}(q)$ to obtain

$$R_\phi J^{-T}(q) H(q)\left[\frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \ddot{y} + \frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \dot{y}\right] + R_\phi J^{-T}(q) C(q,\dot{q})\left[\frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \dot{y}\right] + \tag{9}$$
$$R_\phi J^{-T}(q) D\left[\frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \dot{y}\right] + R_\phi J^{-T}(q) g(q) = R_\phi J^{-T}(q)\left(\tau - \tau_p\right)$$

Finally, we have the equation (1) in function of the image coordinates

$$\bar{H}(q)\ddot{y} + C(q,\dot{q})\dot{y} + \bar{D}\dot{q} + \bar{g}(q) = \tau - \tau_p \tag{10}$$

where

$$\bar{H}(q) = R_\phi J^{-T}(q) H(q)\left[\frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \ddot{y} + \frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \dot{y}\right]$$

$$\bar{C}(q,\dot{q}) = R_\phi J^{-T}(q) C(q,\dot{q})\left[\frac{1}{\alpha\lambda} J^{-1}(q) R_\phi\right]$$

$$\bar{D} = R_\phi J^{-T}(q) D\left[\frac{1}{\alpha\lambda} J^{-1}(q) R_\phi \dot{y}\right]$$

$$\bar{g} = R_\phi J^{-T}(q) g(q)$$

The process shown in this section has been developed for two robot employees described in Section V.

Visual Servoing Fuzzy Controller Design

A two-link robot arm is considered as nonlinear plant. In this part, we are interested in T–S fuzzy models by using parallel distributed compensation (PDC) laws. The PDC approach provides a procedure to design a fuzzy controller from a given T–S fuzzy model. The construction of the continuous T–S fuzzy model is based on rules of the form (Tanaka & Wang, 2001).

$$\begin{aligned} &Rule\ i; If\ z_I\ is\ M_{i1}\ and\ \ldots\ldots\ldots\ and\ z_g\ is\ M_{ig} \\ &Then\ \dot{x} = A_i x + B_i u \\ &y = C_i x, i = 1,\ 2, \ldots\ldots, r \end{aligned} \tag{11}$$

Here M_{ij} (j = 1, 2, . . . , g) are fuzzy sets, *r* is the number of model rules, $x(t) \in \mathbb{R}^n$ is the system state vector, $u(t) \in \mathbb{R}^m$ is the input, $y(t) \in \mathbb{R}^q$ is the output vector, $A_i \in \mathbb{R}^{nxn}$, $B_i \in \mathbb{R}^{nxm}$, $C_i \in \mathbb{R}^{qxn}$, $z(t) = [z_1(t), \ldots, z_g(t)]$ are known premise variables that may be functions of the state variables, external disturbances, and/or time. Given a pair of (x(t),u(t)), a standard fuzzy inference method is used, *i. e.* a singleton fuzzifier, product fuzzy inference, and weighted average defuzzifier. The final state of the fuzzy system is inferred as

$$\dot{x}(t) = \frac{\sum_{i=1}^{r} w_i(z(t))(A_i x(t) + B_i u(t))}{\sum_{i=1}^{r} w_i(z(t))} = \sum_{i=1}^{r} h_i(z(t)(A_i x(t) + B_i u(t)) \tag{12}$$

$$y(t) = \frac{\sum_{i=1}^{r} w_i(z(t)) C_i x(t)}{\sum_{i=1}^{r} w_i(z(t))} = \sum_{i=1}^{r} h_i(z(t)) C_i x(t) \tag{13}$$

Where

$$w_i(z(t)) = \prod_{j=1}^{g} M_{ij}(z_j(t)) \tag{14}$$

$$h_i(z(t)) = \frac{w_i(z(t))}{\sum_{i=1}^{r} w_i(z(t))} \tag{15}$$

$M_{ij}(z(t))$ is the grade of membership of $z_j(t)$ in M_{ij}. It is assumed that

$$w_i(z(t)) \geq 0, i = 1, 2, \ldots, r \tag{16}$$

and

$$\sum_{i=1}^{r} w_i(z(t)) > 0 \tag{17}$$

for all t. Therefore

$$h_i(z(t)) \geq 0, \; i = 1, 2, \ldots, r \tag{18}$$

and

$$\sum_{i=1}^{r} h_i(z(t)) = 1 \tag{19}$$

For the convenience of notation $h_i(z(t)) = h_i$ and $w_i(z(t)) = w_i$. Then the dynamics of the final state of the fuzzy system can be represented as

$$\dot{x}(t) = \sum_{i=1}^{r} h_i A_i x(t) + \sum_{i=1}^{r} h_i B_i u(t) \tag{20}$$

With PDC, we have a simple and natural procedure to handle nonlinear control systems. Other nonlinear control techniques require special and rather involved knowledge. The overall output is given by

$$u = -\sum_{i=1}^{r} h_i F_i x \tag{21}$$

The PDC scheme that stabilizes the T–S fuzzy model was proposed by Tanaka and Wang (2001) as a design process that includes a control algorithm and stability proof by using LMIs constraints. The aim is finding appropriately F_i to ensure closed-loop stability.

Theorem 1. The system $\dot{x}(t) = f(x(t), u(t))$ is quadratically stable if exists a quadratic function $V(x(t)) = x^T(t) Px(t), V(0) = 0$ which satisfies the following conditions

$$V(x(t)) > 0, \forall x(t) \neq 0 \Rightarrow P > 0$$
$$\dot{V}(x(t)) > 0, \forall x(t) \neq 0$$

If V(t) exists, is called Lyapunov Function.

The fuzzy controller design is to determine the local feedback gains F_i for the closed-loop T–S fuzzy system. We define $X_i = P_i^{-1}, F_i = M_i X_1^{-1}, X_i = a_{ij} X_j$ for $i, j = 1, \ldots \ldots, r$, where $a_{ij} \neq 1$ and $a_{ij} > 0$ for $i \neq$ j, and $a_{ij} = 1$ for $i =$ j. By giving $\phi_\rho > 0$ and a_{ij} for $i, j, \rho = 1, \ldots \ldots, r$, we obtain the following LMIs conditions that constitute a stable fuzzy controller design problem:

$$X_i > 0, i = 1, \ldots \ldots, r \tag{22}$$

$$\sum_{\rho=1}^{r} \phi_\rho X_\rho + X_i A_j^T - \alpha_{ij} M_j^T B_j^T + A_j X_i - \alpha_{ij} B_j M_j < 0, i, j = 1, \ldots.., r \tag{23}$$

$$X_i A_j^T - \alpha_{ik} M_k^T B_j^T + X_i A_k^T - \alpha_{ij} M_j^T B_k^T + A_j X_i - \alpha_{ij} B_j M_k + A_k X_i - \alpha_{ij} B_k M_j < 0 \tag{24}$$

for each setting of $i, j, k \in \{1, \ldots, r\}$ such that $j < k$

$$\begin{bmatrix} 1 & x^T(0) \\ x(0) & x_i \end{bmatrix} \geq 0 \ \ for \ i = 1, \ldots, r, \tag{25}$$

$$\begin{bmatrix} \Phi_\rho X_i & W_{ij\rho l}^T \\ W_{ij\rho l} & \Phi_\rho I \end{bmatrix} \geq 0 \tag{26}$$

where $W_{ij\rho l} = \xi_{\rho l}(A_i X_i - a_{ij} B_i M_j)$. Note that from Xi $= a_{ij} X_j$ we have $X_j = \left(\frac{1}{\alpha_{ij}}\right) X_i = \alpha_{ij} X_i$, so that $a_{ij} = 1/a_{ij}$ $\forall i, j \in \{1, \ldots, r\}$, and hence, for given i and j, the relation $a_{ij} a_{ij} = 1$ is used.

The coefficients a_{ij} and Φ_ρ for $i, j, \rho = 1, \dots, r$ and $i \neq 1$ can be chosen heuristically according to the application. In particular, Φ_ρ's are chosen to obtain a fast switching among IF–THEN rules in order to keep the speed of response for a closed–loop system (Tanaka & Wang, 2001). The objective of the tracking control is to make the system outputs follow some predefined trajectories. The inputs are given by the vision system and correspond to the image features, and the output is the torque applied in each joint. Fuzzy controller is summarized as follows:

1) Select the fuzzy plant rules and membership functions for nonlinear system.
2) Calculate the matrices A_i, B_i and C_i by using model.
3) Establish the reference model or path to follow.
5) Solve the LMI to obtain F_i.
6) Solve the LMI to obtain P_i.
7) Construct the fuzzy controller.

The system state is given by

$$x(t) = \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} = \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \end{bmatrix} = \begin{bmatrix} y_1 - y_{d1} \\ y_2 - y_{d2} \end{bmatrix} \quad (27)$$

where (y_1, y_2) represents the robot end effector position in image coordinates, and $(\dot{y}_1, \dot{y}_2)$ are the corresponding velocities. Note that it is possible to establish the following limits for the state due to physical restrictions.

$$x1(\mathrm{t}) \in [x_{1min}, x_{1max}] = [-10, 10][pixels]$$
$$x2(\mathrm{t}) \in [x_{2min}, x_{2max}] = [-10, 10][pixels/s]$$

To minimize the design effort and complexity, we try to use as few rules as possible. The maximum and minimum values for state variables are defined based on acquired images. x_1 and x_3 are measurable through the camera. Based on the dynamic model equation which is a function of image coordinates, matrices A_i, B_i and C_i are calculated. The T–S fuzzy model is given by the following rules whose membership functions are of triangular form.

$$\textit{Rule } 1\text{: } \textit{If } \Delta y_1 \textit{ is about } -10 \textit{ and } \Delta y_2 \text{ is about } 10$$

$$THEN\ \dot{x}(t) = A_1 x(t) + B_1 u(t), y = C_1 x(t)$$

$$\textit{Rule } 2\text{: } \textit{If } \Delta y_1 \textit{ is about } -10 \textit{ and } \Delta y_2 \text{ is about } 0$$

$$THEN\ \dot{x}(t) = A_2 x(t) + B_2 u(t), y = C_2 x(t)$$

$$\textit{Rule } 3\text{: } \textit{If } \Delta y_1 \textit{ is about } -10 \textit{ and } \Delta y_2 \text{ is about } -10$$

$$THEN\ \dot{x}(t) = A_3 x(t) + B_3 u(t), y = C_3 x(t)$$

$$\textit{Rule } 4\text{: } \textit{If } \Delta y_1 \textit{ is about } 0 \textit{ and } \Delta y_2 \text{ is about } -10$$

$$THEN\ \dot{x}(t) = A_4 x(t) + B_4 u(t), y = C_4 x(t)$$

$$\textit{Rule } 5\text{: } \textit{If } \Delta y_1 \textit{ is about } 0 \textit{ and } \Delta y_2 \text{ is about } 0$$

$$THEN\ \dot{x}(t) = A_5 x(t) + B_5 u(t), y = C_5 x(t)$$

$$\textit{Rule } 6\text{: } \textit{If } \Delta y_1 \textit{ is about } 0 \textit{ and } \Delta y_2 \text{ is about } 10$$

$$THEN\ \dot{x}(t) = A_6 x(t) + B_6 u(t), y = C_6 x(t)$$

$$\textit{Rule } 7\text{: } \textit{If } \Delta y_1 \textit{ is about } 10 \textit{ and } \Delta y_2 \text{ is about } -10$$

$$THEN\ \dot{x}(t) = A_7 x(t) + B_7 u(t), y = C_7 x(t)$$

$$\textit{Rule } 8\text{: } \textit{If } \Delta y_1 \textit{ is about } 10 \textit{ and } \Delta y_2 \text{ is about } 0$$

$$THEN\ \dot{x}(t) = A_8 x(t) + B_8 u(t), y = C_8 x(t)$$

$$\textit{Rule } 9\text{: } \textit{If } \Delta y_1 \textit{ is about } 10 \textit{ and } \Delta y_2 \text{ is about } 10$$

$$THEN\ \dot{x}(t) = A_9 x(t) + B_9 u(t), y = C_9 x(t)$$

where x is given by (27) and $u = \tau$. In order to get the matrices (A_i, B_i, C_i), a model of the form (10) is necessary. The amount of 9 rules was gotten by trial and error. Finally, by employing the MATLAB LMI Toolbox, we obtain the corresponding matrix F_i and P_i. Some values used are presented in the Appendix.

Experimental Results

The algorithm was tested in two different robots: Robot A465 CRS Robotics (Figure 1) and Robot LAW3 SCHUNK (Figure 2).

One experiment has been carried out for the control schemes. It consists of following a circle in the $y_1 - y_2$ plane described by

$$y_d = \begin{bmatrix} 60 \times \sin(0.2t) + 650 \\ -60 \times \cos(0.2t) + 450 \end{bmatrix} pixel$$

With raw eye, it was set $\phi \cong 45°$ and $\psi \cong 25°$. The camera is approximately at 1.50 m from the plane of robot movement. In Figures 3 and 4, the desired and actual trajectories in the image plane are shown for the approach.

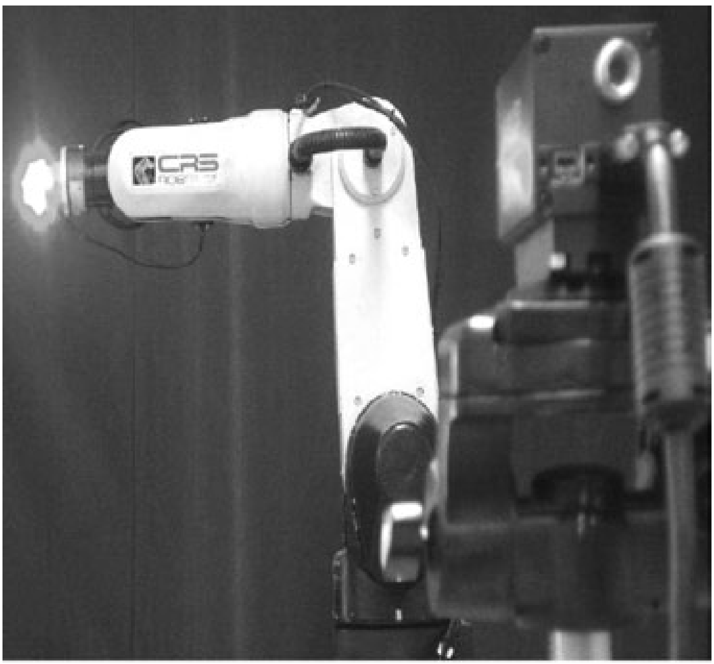

Figure 1. Robot A465 CRS Robotics

The initial position has been chosen to be the same in each case and inside the circle. Note that since the origin (0, 0) is in the upper left corner, a minus sign has been

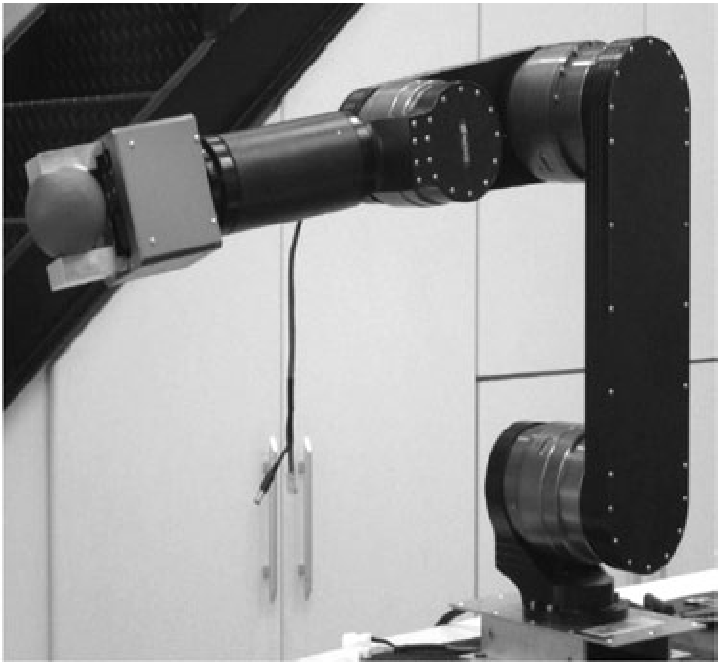

Figure 2. Robot LAW 3 SCHUNK

added to have the y_2 in the right direction, even though actually it is always a positive value. The proposed algorithm achieves zero tracking error (recall that the pixels are integer units); we have computed the Root Mean Square Error (RMSE) as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i}^{n} e_i^2}$$

where i is the current sample number, e_i is the error associated to i, and n is the total number of samples. Table 1 confirms what can be appreciated in the graphics, *i. e.* the good performance of the proposed approach.

Table 1. Root Mean Square Errors

Error	A465 CRS Robotics	LAW3 SCHUNK
Δy1	3.9019	4.0912
Δy2	4.1932	3.1921

Another advantage of fuzzy logic is that it is less sensitive when motor dynamics is not considered, just like in this case. Furthermore, only few laws were necessary for implementation.

In figures 5 and 6, the position error is show for by Robot LAW3 SCHUNK and A465 CRS Robotics.

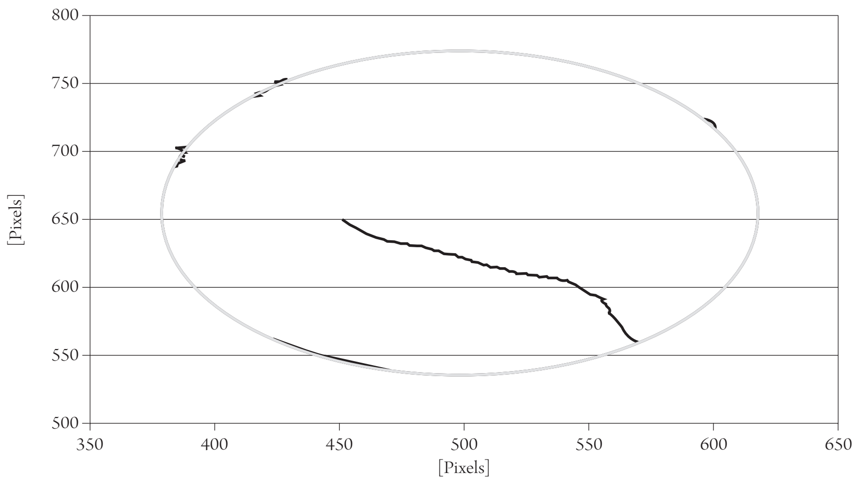

Figure 3. Trajectory for Robot A465 CRS Robotics

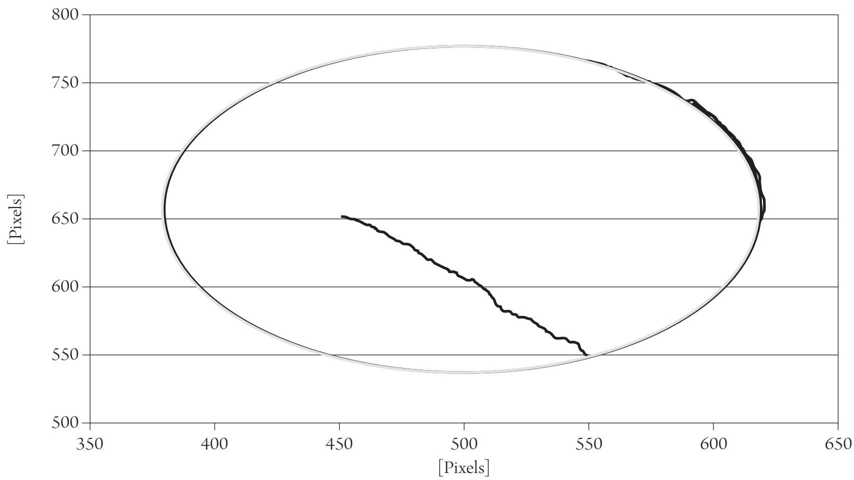

Figure 4. Trajectory for Robot LAW 3 SCHUNK

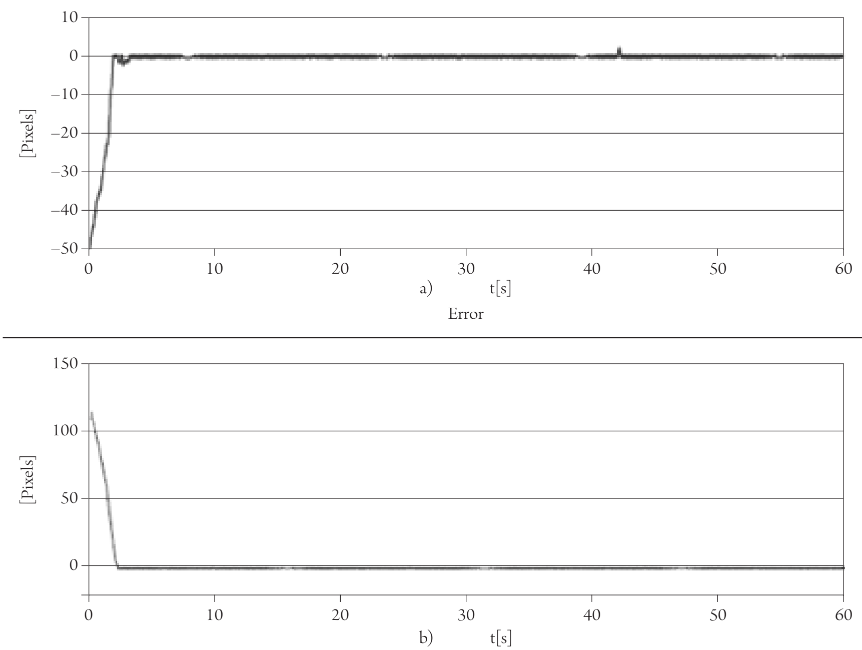

Figure 5. **Position errors for the proposed algorithm in Robot LAW3 SCHUNK**

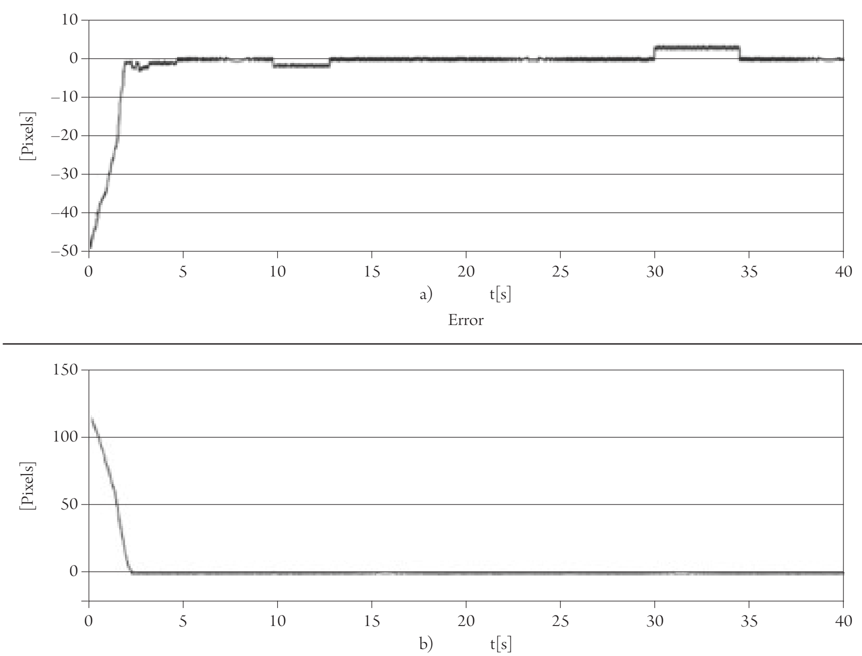

Figure 6. **Position errors for the proposed algorithm in Robot A465 CRS Robotics**

Conclusions

This paper presents experimental results for a fuzzy visual tracking control scheme applied in a robot manipulator. Two different robots were used to show the universality of the proposed algorithm. A fuzzy controller was developed to arbitrarily reduce the tracking errors. With this article we want to show how a control algorithm can be applied, with minor technical adjustments, to operate on any platform, which is a very valuable factor in industrial robotics.

Visual servo control in 2D has multiple developments; the advantages of this proposal are easy design and low errors, as well as serving as a comparative method to other traditional control techniques.

References

Alavandar, S., & Nigan, M. (2008). Adaptive Neuro-Fuzzy Inference System based control of six DOF robot manipulator. *Engineering Science and Technology Review, 1*(2), 106-111.

Bernal, M., & Guerra, T. M. (2010). Generalized Nonquadratic Stability of Continuous-Time Takagi-Sugeno Models. *IEEE Transactions on Fuzzy Systems, 18*(4), 815-882.

Bueno-López, M., & Arteaga Pérez, M. (2013). Fuzzy vs nonfuzzy in 2D visual servoing for robot manipulators. *International Journal of Advanced Robotic Systems, 10*(108), 1-11.

Choi, H. (2007). LMI-Based Nonlinear Fuzzy Observer-Controller Design for Uncertain MIMO Nonlinear Systems. *IEEE Transactions on Fuzzy Systems, 15*(5), 956-971.

De Jong F., & Pieter, P. J. (2000). Visual Servoing in PCB Manufacturing. 6th Annual Conference of the Advanced School for Computing and Imaging (ASCI) (pp. 59-63). Lommel, Belgium.

Grande-Meza, A. (2003). *Observadores difusos y control adaptable difuso basado en observadores* (Master Thesis. Mexico: Instituto Politécnico Nacional).

Hill, J., & Park, W. (1979). Real time control of a robot with a mobile camera. In *Proceedings of the 9th ISIR* (pp. 233-246). Washington D.C.

Hsu, L., Costa, R., & Lizarralde, F. (2007). Lyapunov/Passivity-based adaptive control of relative degree two MIMO systems with an application to visual servoing. *IEEE Transactions on Automatic Control, 52*(2), 364-371.

Hutchinson, S., & Chaumette, F. (2007). Visual servo control. Part II: Advanced approaches. *IEEE Robotics Automation Magazine, 4*(1), 109-118.

Korba, P., Babuska, R., & Verbruggen, B. (2003). Fuzzy gain scheduling: Controller and observer design based on Lyapunov method and convex optimization. *IEEE Transactions on Fuzzy Systems, 11*(3), 285-298.

Liu, C., Cheah, C., & Slotine, J. J. (2006). Adaptive Jacobian Tracking control of rigid-link electrically driven robots based on visual task-space information. *Automatica, 42*(9), 1491-1501.

Lizarralde, F., Hsu, L., & Costa, R. (July, 2008). Adaptive Visual Servoing of Robot Manipulators without Measuring the Image Velocity. In *Proceedings of the 17th World Congress The International Federation of Automatic Control* (pp. 4108-4113). Seoul, Korea.

Ma, X., Sun, Z., & He, Y. (1998). Analysis and Design of Fuzzy Controller and Fuzzy Observer. *IEEE Transactions on Fuzzy Systems, 6*(1998), 41-51.

Sanderson, A., & Weiss, L. (1980). Image-based visual servo control using relational graph error signals. In *Proceedings of the IEEE International Conference on Cybernetics and Society* (pp. 1074-1077). Cambridge, Massachusetts.

Sciavicco, L., & Siciliano, B. (2000) *Modeling and control of robot manipulators.* London: Springer-Verlag.

Shirai, Y., & Inoue, H. (1973). Guiding a robot by visual feedback in assembling tasks. *Pattern Recognition, 5*(1), 99-108.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics, 6*(1), 116-132.

Tanaka, K., & Wang, H. (2001). *Fuzzy control systems design and analysis. A linear matrix inequality approach.* New York: John Wiley and Sons.

Tanaka, K., Ikeda, T., & Wang, H. O. (1996). Robust stabilization of a class of uncertain nonlinear systems via fuzzy control: Quadratic stabilizability, H control theory, and linear matrix inequalities. *IEEE Transactions on Fuzzy Systems, 4*(1), 1-13.

Wang, H., Tanaka, K., & Griffin, M. (1996). An approach to fuzzy control of nonlinear systems: Stability and design issues. *IEEE Transactions on Fuzzy Systems, 4*(1), 14-23.

Weiss, L., Sanderson, A., & Neuman, C. (1985). Dynamic visual servo control of robots: an adaptive image-based approach. In *Proceedings of the IEEE International Conference on Robotics and Automation* (pp. 662-668). Pittsburgh, USA.

Weiss, L., Sanderson, A., & Neuman, C. (1987). Dynamic sensor-based control of robots with visual feedback. *IEEE Journal of Robotics and Automation, 3*(5), 404-417.

Weng, H., Hui Liu, Y., & Chen, W. (2010). Uncalibrated visual tracking control without visual velocity. *IEEE Transactions on Control Systems Technology, 18*(6), 1359-1370.

Yow Lian, K., Tu, H., & Liou, J. (2006). Stability conditions for LMI Based fuzzy control from viewpoint of membership functions. *IEEE Transactions on Fuzzy Systems, vol. 14*(6), 874-884.

Zhang, J., & Fei, M. (2006). Analysis and Design of Robust Fuzzy Controllers and Robust Fuzzy Observers of Nonlinear Systems. In *Proceedings of the 6th World Congress on Intelligent Control and Automation* (pp. 3767-3771). Dalian, China.

Appendix

Due to lack of space, we present only the matrices for the operation point p_1 for Robot A465 CRS Robotics. In this case, we use the next values:

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 3.9 & -0.001 & -0.32 & -8.4x10^{-6} \\ 0 & 0 & 0 & 1 \\ -6.9 & 0.002 & 3.2 & 6,2x10^{-6} \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & 2 \end{bmatrix}^T$$

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$P_1 \begin{bmatrix} 0.0034 & 0.0005 & -0.0004 & 0.0005 \\ 0.0005 & 0.0005 & 0.0000 & 0.0000 \\ -0.0004 & 0.0000 & 0.0190 & 0.0011 \\ 0.0000 & 0.0000 & 0.0011 & 0.0004 \end{bmatrix}$$

$$F_1 = \begin{bmatrix} -263.49 & -100.42 & -143.45 & 54.11 \\ -164.11 & -63.37 & -138.30 & -53.05 \end{bmatrix}$$