

Fuzzy vs Nonfuzzy in 2D Visual Servoing for Robot Manipulators

Regular Paper

Maximiliano Bueno-López¹ and Marco A. Arteaga-Pérez^{1,*}

¹ Departamento de Control y Robótica, División de Ingeniería Eléctrica de la Facultad de Ingeniería, Universidad Nacional Autónoma de México, México, D. F., México.

* Corresponding author E-mail: marteagp@unam.mx

Received 4 Sep 2012; Accepted 20 Dec 2012

DOI: 10.5772/55593

© 2013 Bueno-López and Arteaga-Pérez; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract In this paper we consider the tracking control problem of planar robots manipulators via visual servoing in the presence of parametric uncertainties associated with the robot dynamics and the camera parameters. An image-based visual servo control is developed using a Takagi-Sugeno (T-S) fuzzy model. The design includes an image coordinate velocity observer. To the best knowledge of the authors such a proposal remains hitherto unpublished. The new scheme is compared experimentally with two different non-fuzzy algorithms showing the good performance and advantages of the complete system.

Keywords Fuzzy Tracking Control, Fuzzy Observer Design, Stability Analysis, Visual Servoing

1. Introduction

Among the model-based fuzzy control approaches, the Takagi-Sugeno (T-S) is one of the most effective strategies. They proposed their model in 1985 Takagi and Sugeno (1985) and it has emerged as one of the most active areas of research. Basically, a complex dynamic model can be composed in a set of local linear subsystems via fuzzy inference. Hence, it is not surprising that

practical applications for control started to appear very quickly after the method had been introduced in different works. On the other hand, in the design of fuzzy controllers a good option is to use parallel distributed compensation (PDC), which offers a procedure to compute a control law from the T-S model. In Kim et al. (2001) a solution to target identification and grasping using a real platform in combination with a vision system is proposed. The task for the robot end-effector is to approach a randomly placed spherical object of known size. The control algorithm is approximated by using an approach based on fuzzy rules that are built in a supervised way, after studying the system behavior. In Wai and Yang (2008) 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 Fares et al. (2007) a fuzzy logic system and a genetic algorithm are integrated for adaptive visual servoing. Genetic algorithms are employed to optimize the internal parameters of membership functions. In Goncalves et al. (2008) a new uncalibrated eye-to-hand visual servoing based on inverse fuzzy modeling to obtain an inverse

relationship of the mapping between image feature variations and joint velocities is proposed. This approach is independent of the robot's kinematic model and camera calibration and avoids the necessity of inverting the manipulator Jacobian online. Different methods can be employed to estimate and identify the internal states of a system when only information from input and output are available, *e. g.* observer design Tanaka et al. (1998). 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 and Fei (2006). In Meza (2003) one of the few works on this subject is presented and different designs applied to several nonlinear fuzzy observers are shown with excellent results. In this paper, a fuzzy visual servoing approach for trajectory tracking of planar robot manipulators in the absence of camera parameters, robot dynamics and velocity measurements is introduced. However, on the contrary to the work in Abdelmalek et al. (2007), where a Takagi–Sugeno model based on joint coordinates is employed, in this work we present the model in image coordinates. This strategy is not yet published according to the literature review. Furthermore, an experimental comparison with other well-known algorithms is carried out.

The paper is organized as follows: Section 2 presents the mathematical preliminaries. Section 3 gives the basic concepts about parallel distributed compensation for the Takagi–Sugeno Fuzzy Model and observer design. The control–observer approach is given in Section 4, while the

experimental results are presented in Section 5. Finally, some conclusions are given in Section 6.

2. Preliminaries

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

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

where $q \in \mathbb{R}^n$ is the vector of generalized joint coordinates, $H(q) \in \mathbb{R}^{n \times n}$ 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, $D \in \mathbb{R}^{n \times n}$ is the positive semi-definite diagonal matrix accounting 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, *i. e.* 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 $x_R \in f_k(q)$. In this work we used the camera fixed configuration.

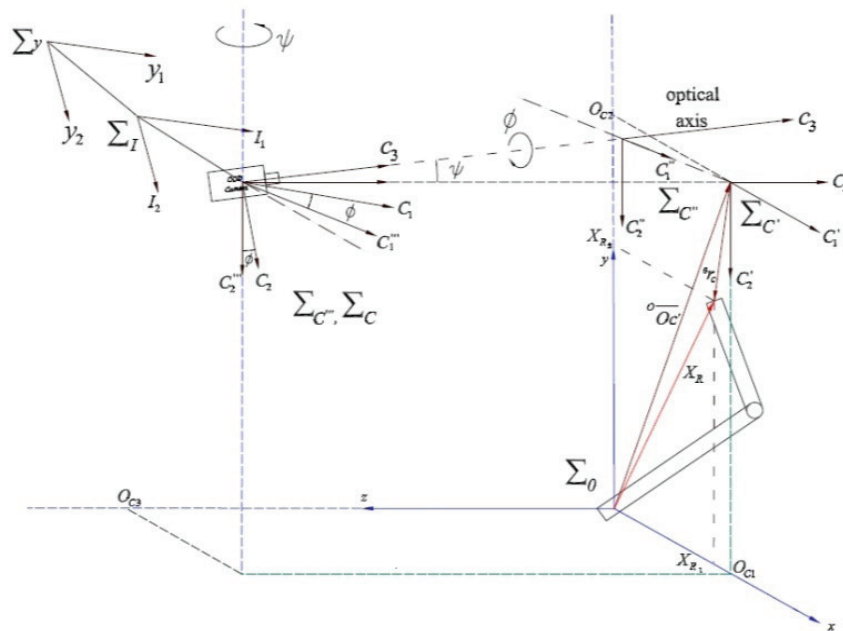


Figure 1. Planar robot system with not parallel camera.

The output of the system is the position $y \in \mathbb{R}^2$ of the image feature, *i. e.* the position of the target in the

computer screen. As shown in Figure 1 the robot workspace is in the x – y plane. The camera is not assumed

to be completely parallel to this plane, so that there are two angles of rotation $\varphi, \psi \in \mathbb{R}$ 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 a perspective projections. As shown in the figure, the following coordinates systems are defined: $\Sigma_i, \Sigma_c, \Sigma_e, \Sigma_c, \Sigma_e, \Sigma_i$ and Σ_r . The position of the origin of the camera frame Σ_c with respect to the robot frame is given by $o_c = [o_{c1} o_{c2} o_{c3}]^T$. Then, similar to the procedure explained in Kelly and Reyes (2000), by assuming $\cos \psi \approx 1$, $\sin \psi \approx \psi$, $\tan \psi \approx \psi$, $\psi^2 \approx 0$ one gets after some manipulation

$$y = \alpha_\lambda R_\varphi \left\{ x_R - \begin{bmatrix} 0_{c1} \\ 0_{c2} \end{bmatrix} \right\} + \begin{bmatrix} u_0 + \alpha_\lambda \cos(\varphi) 0_{c3} \psi \\ v_0 + \alpha_\lambda \sin(\varphi) 0_{c3} \psi \end{bmatrix},$$

where λ is the focal length, α is a conversion factor from meters to pixels, $[u_0 v_0]^T$ is the center offset and

$$R_\varphi = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) \\ \sin(\varphi) & -\cos(\varphi) \end{bmatrix} \quad (2)$$

3. Parallel distributed compensation for Takagi–Sugeno Fuzzy Model

The parallel distributed compensation (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 Takagi and Sugeno (1985)

$$\begin{aligned} \text{Rule } i: & \text{If } z_1 \text{ is } M_{i1} \text{ and ...and } z_g \text{ is } M_{ig} \\ \text{Then } x &= A_i x + B_i u \\ y &= C_i x, \quad i = 1, 2, \dots, r. \end{aligned} \quad (3)$$

Here $M_{ij} (j=1, 2, \dots, g)$ 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, $A_i \in \mathbb{R}^{n \times n}$, $B_i \in \mathbb{R}^{n \times m}$, $C_i \in \mathbb{R}^{q \times n}$, $z(t) = [z_1(t), \dots, z_g(t)]$ 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

$$\dot{x} = \frac{\sum_{i=1}^r w_i(z(t))(A_i x(t))}{\sum_{i=1}^r w_i(z(t))} = \sum_{i=1}^r h_i(z(t))(A_i x(t) + B_i u(t)) \quad (4)$$

$$y = \frac{\sum_{i=1}^r w_i(z) C_i x}{\sum_{i=1}^r w_i(z)} = \sum_{i=1}^r h_i(z) C_i x \quad (5)$$

where

$$w_i(z(t)) = \prod_{j=1}^g M_{ij}(z_j(t)) \quad (6)$$

$$h_i(z(t)) = \frac{w_i(z(t))}{\sum_{i=1}^r w_i(z(t))} \quad (7)$$

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

$$w_i(z(t)) \geq 0, i = 1, 2, \dots, r \quad (8)$$

and

$$\sum_{i=1}^r w_i(z(t)) > 0, \quad (9)$$

for all t . Therefore

$$h_i(z(t)) \geq 0, i = 1, 2, \dots, r \quad (10)$$

and

$$\sum_{i=1}^r h_i(z(t)) = 1. \quad (11)$$

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

$$\dot{x}(t) = \sum_{i=1}^r h_i A_i x(t) + \sum_{i=1}^r h_i B_i u(t). \quad (12)$$

The main goal is to determine the local feedback gain F_i in the consequent parts. 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

$$u = - \sum_{i=1}^r h_i F_i x. \quad (13)$$

The PDC scheme that stabilizes the T–S fuzzy model was proposed by Wang *et al.* Liu and Lam (2006) as a design framework comprising a control algorithm and a stability test using optimization involving LMI constraints.

3.1 Quadratic Stability Conditions

Consider the following definition.

Definition 3.1. Abdelmalek et al. (2007) The system $\dot{x}(t) = f(x(t), u(t))$ is said quadratically stable if there exists a quadratic function $V(x(t)) = x^T(t) P x(t)$,

$V(0)=0$, called Lyapunov function, satisfying the following conditions

$$V(x(t)) > 0, \forall x(t) \neq 0 \Leftrightarrow P > 0, \quad (14)$$

$$V(x(t)) < 0, \forall x(t) \neq 0 \quad (15)$$

By substituting (13) in (4), we obtain the Takagi-Sugeno closed loop fuzzy systems as follows Abdelmalek et al. (2007)

$$\dot{x} = \sum_{i=1}^r \sum_{j=1}^r h_i(z) h_j(z) (A_i - B_i F_j) x, \quad (16)$$

which can be rewritten as

$$\dot{x} = \sum_{i=1}^r h_i(z) h_i(z) G_{ii} x + 2 \sum_{i=1}^r \sum_{i < j} h_i(z) h_j(z) \left(\frac{G_{ij} + G_{ji}}{2} \right) x, \quad (17)$$

where $G_{ij} = A_i - B_i F_j$ and $G_{ii} = A_i - B_i F_i$. The stabilization of a feedback system containing a state feedback fuzzy controller has been extensively considered. The stability conditions corresponding to a quadratic Lyapunov function were derived by Tanaka and Sugeno in Tanaka and Sugeno (1992). The approach requires to find a common positive definite matrix for the r subsystems, which makes it very conservative.

3.2 Stabilization Approach

The stabilization approach used in this paper was introduced by Tanaka *et al.* in Tanaka et al. (2003), where a fuzzy Lyapunov function is proposed. A candidate Lyapunov function is of the form

$$V(x(t)) = \sum_{i=1}^r h_i(z(t)) x^T(t) P_i x(t), \quad (18)$$

where P_i is a positive definite matrix and $h_i(z(t)) \geq 0$, $\sum_{i=1}^r h_i(z(t)) = 1$. An interesting advantage is that this form of Lyapunov function takes into account the speed variation of the decision variables, what allows more relaxed stability conditions since this type of functions reduces the global stability of the nonlinear system to the analysis of the local stability of each local linear model (sub-model) separately. The candidate Lyapunov function (18) satisfies the following conditions:

$$\dot{V}(x) = x^T \left[\sum_{i=1}^r \sum_{j=1}^r h_i(z) h_j(z) (G_{ij}^T P_i + P_i G_{ij}) + \sum_{\rho=1}^r \dot{h}_{\rho}(z) P_{\rho} + \sum_{i=1}^r \sum_{j=1}^r \sum_{j < k} h_i(z) h_j(z) h_k(z) \left(\left(\frac{G_{jk} + G_{kj}}{2} \right)^T P_i + P_i \left(\frac{G_{jk} + G_{kj}}{2} \right) \right) \right] x \quad (22)$$

Under Assumption 3.1, $V(x(t))$ in (22) can be rewritten as follows

$$\dot{V}(x) \leq x^T \left[\sum_{i=1}^r \sum_{j=1}^r h_i h_j (G_{ij}^T P_i + P_i G_{ij}) + \sum_{\rho=1}^r \varphi_{\rho} P_{\rho} + \sum_{i=1}^r \sum_{j=1}^r \sum_{j < k} h_i h_j h_k \left(\left(\frac{G_{jk} + G_{kj}}{2} \right)^T P_i + P_i \left(\frac{G_{jk} + G_{kj}}{2} \right) \right) \right] x \quad (23)$$

1. V is C^1
2. $V(0) = 0$ and $V(x) > 0$ for $x(t) \neq 0$
3. $\|x(t)\| \rightarrow \infty \Rightarrow V(x(t)) \rightarrow \infty$

Equation (18) is said to be a fuzzy Lyapunov function for the T-S fuzzy system if the time derivative of $V(x(t))$ is always negative at $x(t) \neq 0$, where P_i is a positive definite matrix. Based on the fuzzy Lyapunov function, we use an approach that gives less conservative stability conditions Abdelmalek et al. (2007), the key assumptions are:

Assumption 3.1. The time derivative of the premise membership function is upper bounded and such that $\dot{h}_i(z(t)) \leq \varphi_i$, where φ_i are given positive constants with $i = 1, \dots, r$.

Assumption 3.2. The local quadratic Lyapunov functions $x^T(t) P_i x(t)$, $i = 1, \dots, r$ are proportionally related such that $P_j = \alpha_{ij} P_i$ for $i, j = 1, \dots, r$, where $\alpha_{ij} \neq 1$ and $\alpha_{ij} > 0$ for $i \neq j$, and $\alpha_{ij} = 1$ for $i = j$.

Now consider now the following theorem.

Theorem 3.1. Under Assumptions 3.1 and 3.2, the fuzzy system (16) is stable by using the PDC fuzzy controller (13) if there exist φ_{ρ} , α_{ij} for $i, j, \rho = 1, \dots, r$, positive definite matrices P_1, P_2, \dots, P_r and matrices F_1, F_2, \dots, F_r such that for all $i, j, k \in \{1, \dots, r\}$

$$P_i > 0 \quad (19)$$

$$\sum_{\rho=1}^r \varphi_{\rho} P_{\rho} + (G_{ij}^T P_i + P_i G_{ij}) < 0 \quad (20)$$

$$\left(\frac{G_{jk} + G_{kj}}{2} \right)^T P_i + P_i \left(\frac{G_{jk} + G_{kj}}{2} \right) < 0 \text{ (for } j < k), \quad (21)$$

where $G_{jk} = A_j - B_j F_k$ and $G_{ji} = A_j - B_j F_i$.

Proof:

The derivative of $V(x(t))$ in (18) along (17) can be computed after a rather direct but long manipulation as

If equations (19), (20) and (21) hold, the time derivative of the fuzzy Lyapunov function is negative, *i. e.* $\dot{V}(x) \leq 0$, and the closed loop fuzzy system (16) is stable.

Theorem 3.1 is proven by taking into account Assumption 3.1. The constraint imposed on the time derivative of the premise membership functions and hence on the derivative of the premise variables, *i. e.* the speed of the state variables for the case of $z(t) = x(t)$, is transformed into LMIs of Theorem 3.2 that are solved simultaneously with those of Theorem 3.1 to stabilize the T-S fuzzy systems. The new LMIs that support Assumption 3.1 allow to increase the performance by limiting the displacement rate in the polytope, implying a facility to find the Lyapunov functions and thus a faster stabilization.

Theorem 3.2. Assumption 3.1 holds if there exist positive definite matrices P_1, P_2, \dots, P_r and matrices F_1, F_2, \dots, F_r satisfying

$$\begin{bmatrix} 1 & x^T(0) \\ x(0) & P_i^{-1} \end{bmatrix}, \begin{bmatrix} \varphi_\rho P_i & W_{ij\rho l}^T \\ W_{ij\rho l} & \varphi_\rho I \end{bmatrix} \geq O, \quad (24)$$

$\forall i, j, \rho \in 1, 2, \dots, r$ and $\forall l$, where $W_{ij\rho l} = \xi_{\rho l}(A_i - B_i F_j)$.

Proof:

For $|\dot{h}_\rho(z)| \leq \phi_\rho$ and $(z = x)$ one has

$$\left| \dot{h}_\rho(z(t)) \right| = \left| \frac{\partial h_\rho(z(t))}{\partial x(t)} \dot{x}(t) \right| \leq \varphi_\rho, \quad (25)$$

where

$$\frac{\partial h_\rho(z(t))}{\partial x(t)} = \sum_{l=1}^s \nu_{\rho l}(z(t)) \xi_{\rho l}, \quad (26)$$

with $\nu_{\rho l} \geq 0$ and $\sum_{l=1}^s \nu_{\rho l}(z(t)) = 1$. In (26) $\nu_{\rho l}(z(t))$ depends on the number of membership functions. This term represents the maximum value that each input can have when evaluated in the different functions. $\xi_{\rho l}$ depends on the maximum and minimum values that can take a membership function evaluated at the extreme values of the universe of discourse. This term represents the interval that covers each function. Using (26) we obtain LMIs that satisfy (25), from which we can derive

$$\left(\frac{\partial h_\rho(z(t))}{\partial x(t)} \dot{x}(t) \right)^T \left(\frac{\partial h_\rho(z(t))}{\partial x(t)} \dot{x}(t) \right) \leq \varphi_\rho^2. \quad (27)$$

By replacing (16) in (27) it is

$$\frac{1}{\varphi_\rho^2} x^T \left[\begin{pmatrix} \sum_{l=1}^s \nu_{\rho l} \xi_{\rho l} \left(\sum_{i=1}^r \sum_{j=1}^r h_i h_j [A_i - B_i F_j] \right) \\ \sum_{l=1}^s \nu_{\rho l} \xi_{\rho l} \left(\sum_{i=1}^r \sum_{j=1}^r h_i h_j [A_i - B_i F_j] \right) \end{pmatrix} \right] x \leq 1. \quad (28)$$

If the fuzzy Lyapunov function (18) satisfies

$$V(x(t)) \leq V(x(0)) \leq 1, t \geq 0, \quad (29)$$

then

$$\begin{aligned} \sum_{i=1}^r h_i(z(t)) x^T(t) P_i x(t) &\leq \\ \sum_{i=1}^r h_i(z(0)) x^T(0) P_i x(0) &\leq 1, \end{aligned} \quad (30)$$

so that

$$1 - \sum_{i=1}^r h_i(z(0)) x^T(0) P_i x(0) \geq 0 \quad (31)$$

and

$$1 - x^T(0) \left(\sum_{i=1}^r h_i(z(0)) P_i \right) x(0) \geq 0. \quad (32)$$

This last equation can be expressed via LMIs by using the Schur complement as follows:

$$\begin{bmatrix} 1 & x^T(0) \\ x(0) & \left(\sum_{i=1}^r h_i(z(0)) P_i \right)^{-1} \end{bmatrix} \geq O, \quad (33)$$

because

$$\begin{bmatrix} 1 & x^T(0) \\ x(0) & P_i^{-1} \end{bmatrix} \geq O, \text{ for } i = 1, \dots, r. \quad (34)$$

This leads to the LMI condition (24). On the other hand, by considering inequalities (28) and (30), inequality (25) holds if

$$\begin{aligned} \frac{1}{\varphi_\rho^2} &\left[\left(\sum_{l=1}^s \nu_{\rho l}(z) \xi_{\rho l} \left(\sum_{i=1}^r \sum_{j=1}^r h_i h_j A B F_i^T \right) \right) \right. \\ &\left. \left(\sum_{l=1}^s \nu_{\rho l}(z) \xi_{\rho l} \left(\sum_{i=1}^r \sum_{j=1}^r h_i h_j A B F_i \right) \right) \right] \\ &- \sum_{i=1}^r h_i P_i \leq O, \end{aligned}$$

where $A B F_i^T = A_i - F_i F_j$. Equivalently one has

$$\begin{bmatrix} \varphi_\rho \sum_{i=1}^r h_i P_i & \sum_{l=1}^s \nu_{\rho l} \xi_{\rho l} Q^T \\ \sum_{l=1}^s \nu_{\rho l} \xi_{\rho l} Q & \varphi_\rho I \end{bmatrix} \geq O, \quad (35)$$

where $Q = \sum_{i=1}^r \sum_{j=1}^r h_i h_j [A_i - B_i F_j]$. This leads to the LMI condition (24).

Remark 3.1. $\xi_{\rho l}$ can be obtained from $h_i(z(t))$ by using the procedure given in Tanaka and Wang (2001). However, note that the conditions of Theorem 3.2 depend on the initial system state, so that they need to be known.

3.3 Observer Design

In practice, all states may not be fully measurable and it may be necessary to design an observer to implement (13). If the pairs (A_i, C_i) are observable, the fuzzy system (3) is called locally observable. In this work we will assume that this is the case. The design is based on fuzzy implications, with fuzzy sets in antecedents, and a Luenberger observer in the consequents. Each rule is responsible for estimating the states of a locally linear subsystem. The final outputs are obtained by fuzzy blending the local observers outputs. Based on the PDC structure, the local observers are designed as

$$\begin{aligned} \text{Rule } i: & \text{ if } z_1(t) \text{ is } M_{i1} \text{ and } \dots \text{ and } z_g(t) \text{ is } M_{ig} \\ & \text{Then } \dot{\hat{x}} = A_i \hat{x} + B_i u + L_i (y - \hat{y}) \\ & \hat{y}_i = C_i \hat{x}, \quad i = 1, 2, \dots, r \end{aligned} \quad (36)$$

where L_i with $i = 1, 2, \dots, r$ is the observer gain for the i -th rule. The corresponding PDC fuzzy controller takes form

$$u = - \sum_{i=1}^r h_i F_i \hat{x}, \quad (37)$$

instead of (13). Note that the same weight $w_i(z)$ is used. The dependence of the premise variables on the state makes necessary to consider two cases for fuzzy observer design:

1. z_1, \dots, z_g do not depend on the estimated state
2. z_1, \dots, z_g depend on the estimated state

In the second case in general $h_i(z) \neq h_i(\hat{z})$ because $z \neq \hat{z}$ *a priori*. Then the premise variables can be the outputs or a combination of the measured states. In this work, we make the following assumption.

Assumption 3.3. $z(t)$ is independent of x .

After Assumption 3.3, it is proposed

$$\dot{\hat{x}} = \sum_{i=1}^r h_i [A_i \hat{x} + B_i u + L_i (y - \hat{y})], \quad (38)$$

with final output

$$\hat{y}(t) = \sum_{i=1}^r h_i(z(t)) C_i \hat{x}(t). \quad (39)$$

The design consists in finding the gains L_i in the consequent part, for which it is necessary to compute the observer error dynamics. Substitute (5), (37) and (39) in (38) to obtain

$$\dot{\hat{x}} = \sum_{i=1}^r h_i A B F_i \hat{x} + \sum_{i=1}^r \sum_{j=1}^r h_i h_j L_i C_j (x - \hat{x}). \quad (40)$$

On the other hand, by substituting (37) in system (12) one has

$$\dot{x}(t) = \sum_{i=1}^r \sum_{j=1}^r h_i h_j (A_i x - B_i F_j \hat{x}). \quad (41)$$

Define the observer error as

$$\tilde{x} = x - \hat{x}. \quad (42)$$

Then, by subtracting (40) from (41) one gets the observer error dynamics as

$$\dot{\tilde{x}} = \sum_{i=1}^r \sum_{j=1}^r h_i h_j (A_i - L_i C_j) \tilde{x}. \quad (43)$$

Now consider the following theorem.

Theorem 3.3. The observer error dynamics (43) is asymptotically stable if there exist positive definite matrices $P_{01}, P_{02}, \dots, P_{0r}$ and matrices L_1, L_2, \dots, L_r such that the following is satisfied:

$$P_{0i} > 0 \quad (44)$$

$$\sum_{\rho=1}^r \varphi_{\rho} P_{0\rho} + (G_{jj}^T P_{0i} + P_{0i} G_{jj}) < 0 \quad (45)$$

$$\left(\frac{G_{jk} + G_{kj}}{2} \right)^T P_{0i} + P_{0i} \left(\frac{G_{jk} + G_{kj}}{2} \right) < 0 \text{ for } j < k$$

$$\begin{bmatrix} 1 & x^T(0) \\ x(0) & P_{0i}^{-1} \end{bmatrix} \begin{bmatrix} \varphi_{\rho} P_{0i} & W_{ij\rho}^T \\ W_{ij\rho} & \varphi_{\rho} I \end{bmatrix} \geq 0 \quad (46)$$

$$\forall i, j, k, p \in \{1, \dots, r\} \text{ and } \forall l,$$

where

$$G_{jk} = A_j - L_j C_k, G_{jj} = A_j - L_j C_j \text{ and } W_{ijpl} = \xi_{pl} (A_i - L_i C_j)$$

Remark 3.2. By augmenting the states of the system with the state estimation error, we obtain the following $2n$ dimensional state equations for the observer/controller closed-loop system

$$\begin{bmatrix} \dot{\hat{x}} \\ \dot{\tilde{x}} \end{bmatrix} = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ 0 & \bar{A}_{22} \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} \quad (47)$$

$$y = \begin{bmatrix} \sum_{i=1}^r h_i C_i & 0 \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix}, \quad (48)$$

With

$$A_{11} = \sum_{i=1}^r \sum_{j=1}^r h_i h_j (ABF_j), \quad A_{12} = \sum_{i=1}^r \sum_{j=1}^r h_i h_j (A_i - L_i C_j).$$

Finally, as shown in Abdelmalek (2009), the separation principle holds for system (4)–(5), so that the error dynamics (47) is asymptotically stable.

4. Visual Servoing Fuzzy Controller Design

Consider a two-link robot arm (see Figure 2). In this section we are interested in a T-S fuzzy model by using PDC laws. The fuzzy controller design is to determine the local feedback gains F_i for the closed loop T-S fuzzy system (16).

We define $X_i = P_i^{-1}$, $F_i = M_i X_i^{-1}$, $X_i = \alpha_{ij} X_j$ for $i, j = 1, \dots, r$, where $\alpha_{ij} \neq 1$ and $\alpha_{ij} > 0$ for $i \neq j$, and $\alpha_{ij} = 1$ for $i = j$. By giving $\varphi_\rho > 0$ and α_{ij} for $i, j, \rho = 1, \dots, r$, we obtain the following LMIs conditions that constitute a stable fuzzy controller design problem:

$$X_i > 0 \quad (49)$$

$$\sum_{\rho=1}^r \varphi_\rho X_\rho + X_i A_j^T - \alpha_{ij} M_j^T B_j^T + A_j X_i - \alpha_{ij} B_j M_j < 0 \quad (50)$$

$$X_i A_j^T - \alpha_{ik} M_k^T B_j^T + X_i A_k^T - \alpha_{ij} M_j^T B_k^T + A_j X_i - \alpha_{ij} B_j M_k + A_j X_i - \alpha_{ij} B_j M_j < 0, \quad (51)$$

for each set of $i, j, k \in \{1, \dots, r\}$ such as $j < k$

$$\begin{bmatrix} 1 & x^T(0) \\ x(0) & X_i \end{bmatrix} \geq 0 \quad \text{for } i = 1, \dots, r, \quad (52)$$

$$\begin{bmatrix} \varphi_\rho X_i & W_{ij\rho}^T \\ W_{ij\rho} & \varphi_\rho I \end{bmatrix} \geq 0, \quad (53)$$

where $W_{ij\rho} = \xi_{\rho l} (A_i X_i - \alpha_{ij} B_i M_j)$. Note that from $X_i = \alpha_{ij} X_j$ we have $X_j = (1/\alpha_{ij}) X_i = \alpha_{ji} X_i$, so that $\alpha_{ij} = 1/\alpha_{ji} \forall i, j \in \{1, \dots, r\}$, and hence, for given i and j , the relation $\alpha_{ij} \alpha_{ji} = 1$ is used. Note that X_i and F_i are computed by solving (13), while M_j can be obtained from these two matrices.

The coefficients α_{ij} and φ_ρ for $i, j, \rho = 1, \dots, r$ and $i \neq j$, can be chosen heuristically according to the application. In particular, the 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 and Wang (2001). α_{ij} and β_{ij} must be different from 1 (for $i = j, \alpha_{ij} = \beta_{ij} = 1$) and selection of $\xi_{\rho l}$ is obtained from $h_i z(t)$. Subsequently, we will consider that the premise

variables do not depend on the estimated states $\hat{x}(t)$. 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. The fuzzy observer-based control is summarized as follows

1. Select the fuzzy rules and membership functions for nonlinear system (3).
2. Calculate the matrices A_i , B_i and C_i by using model (3).
3. Establish the reference model or path to follow.
4. Compute L_i .
5. Solve the LMI to obtain F_i .
6. Solve the LMI to obtain P_i .
7. Solve the LMI to obtain P_{0i} .
8. Construct the fuzzy observer (36).
9. Construct the fuzzy controller (37).

The system state is given by

$$x(t) = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T = \begin{bmatrix} y_1 & y_2 & y_3 & y_4 \end{bmatrix}^T, \quad (54)$$

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

$$\begin{aligned} x_1(t) &\in [x_{1_{\min}}, x_{1_{\max}}] = [0, 1000] [\text{pixels}] \\ x_2(t) &\in [x_{2_{\min}}, x_{2_{\max}}] = [-50, 50] [\text{pixels/s}] \\ x_3(t) &\in [x_{3_{\min}}, x_{3_{\max}}] = [0, 1000] [\text{pixels}] \\ x_4(t) &\in [x_{4_{\min}}, x_{4_{\max}}] = [-50, 50] [\text{pixels/s}] \end{aligned}$$

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 images acquired. x_1 and x_3 are measurable through the camera. Based on the dynamic model equation which is a function of image coordinates are calculated matrices A_i , B_i and C_i . The T-S fuzzy model is given by the following rules whose membership functions are of triangular form. The premise variables are the tracking error in image coordinates, which is obtained by

$$\begin{bmatrix} \Delta y_1 \\ \Delta y_2 \end{bmatrix} = \begin{bmatrix} y_1 - y_{d1} \\ y_2 - y_{d2} \end{bmatrix}, \quad (55)$$

where (y_1, y_2) are obtained through the camera image on the computer screen, and (y_{d1}, y_{d2}) represent the desired trajectories.

Rule1: If Δy_1 is about -10 and Δy_2 is about 10
Then $\dot{x}(t) = A_1x(t) + B_1u(t), y = C_1x(t),$

Rule2: If Δy_1 is about -10 and Δy_2 is about 0
Then $\dot{x}(t) = A_2x(t) + B_2u(t), y = C_2x(t),$

Rule3: If Δy_1 is about -10 and Δy_2 is about -10
Then $\dot{x}(t) = A_3x(t) + B_3u(t), y = C_3x(t),$

Rule4: If Δy_1 is about 0 and Δy_2 is about -10
Then $\dot{x}(t) = A_4x(t) + B_4u(t), y = C_4x(t),$

Rule5: If Δy_1 is about 0 and Δy_2 is about 0
Then $\dot{x}(t) = A_5x(t) + B_5u(t), y = C_5x(t),$

Rule6: If Δy_1 is about 0 and Δy_2 is about 10
Then $\dot{x}(t) = A_6x(t) + B_6u(t), y = C_6x(t),$

Rule7: If Δy_1 is about 10 and Δy_2 is about -10
Then $\dot{x}(t) = A_7x(t) + B_7u(t), y = C_7x(t),$

Rule8: If Δy_1 is about 10 and Δy_2 is about 0
Then $\dot{x}(t) = A_8x(t) + B_8u(t), y = C_8x(t),$

Rule9: If Δy_1 is about 10 and Δy_2 is about 10
Then $\dot{x}(t) = A_9x(t) + B_9u(t), y = C_9x(t).$



Figure 2. Experimental test bed.

In order to get the matrices (A_i, B_i, C_i) , a model of the form (57) is necessary. In Appendix A a sketch of the robot model in image coordinates, as well as an example of a set (A_i, B_i, C_i) , with $i = 1$, are given. The amount of 9 sets was gotten by trial and error. The image coordinates and their corresponding velocities were considered as the system states, which will be stabilized to zero by the T-S fuzzy controller and observer. The experimental results show that tracking errors tend to zero. The tracking errors are considered in the premise of fuzzy rules; the time

derivatives are not considered because the trajectories are small and this does not affect the design of the controller. It is noted that the T-S fuzzy model for the controller and observer design is only an approximation of the robot dynamics given in (57). Therefore, asymptotical stabilization of the fuzzy model does not mean that the original robot dynamics can be asymptotically stabilized, unless the T-S fuzzy model is an exact representation of the original robot dynamics. The number of fuzzy rules allow having a robust system that operates correctly in the operation zone defined. Finally, by employing the MATLAB LMI Toolbox, the corresponding gains F_i and L_i are computed. The results are also shown in the appendix. Note that the MATLAB LMI Toolbox easily allows to satisfy the conditions given in Theorems 3.1, 3.2 and 3.3 to guarantee asymptotical stability of tracking and observation errors.

5. Experimental Results

In order to test the theory given in Section 4, the robot A465 of CRS Robotics shown in Figure 2 has been used. It has six degrees of freedom, but only joints 2 and 3 (renumbered 1 and 2, respectively) are employed. The visual system consists of a CCD monochrome camera (Pike F-505B). A circular object is attached to the robot end-effector to recognize it. In order to have a point of comparison, two algorithms have also been programmed. Although they are not explicitly designed to cope with two angles of rotation, we consider that the comparison is fair because both of them are robust schemes.

The *Comparative Algorithm 1* is an adaptive second order sliding mode global tracking visual feedback controller developed in Parra-Vega et al. (2003). It has been chosen because it is also meant for planar manipulators in a modified image-based approach. Furthermore, the basic assumption is that the robot model parameters are unknown. The *Comparative Algorithm 2* is the one given in Zergeroglu et al. (2001). It is an adaptive calibration controller that compensates for uncertain camera parameters and ensures global asymptotic position tracking. The interested reader can see the reference to check out these schemes, omitted here for lack of room.

5.1 Experimental Outcomes

One experiment has been carried out for all control schemes which is considered representative enough to check out tracking performance. It consists in following a circle in the $y_1 - y_2$ plane described by

$$y_d(t) = \begin{bmatrix} 60 \times \sin(0.2t) + 650 \\ -60 \times \cos(0.2t) + 450 \end{bmatrix} \text{ [pixel]}. \quad (56)$$

With raw eye it was set $\phi \approx 45^\circ$ and $\psi \approx 25^\circ$. Note that the comparative algorithms are only designed for the

nominal case $\psi \approx 0^\circ$, thus they rely on their robustness properties to get good results.

In Figure 3, the desired and actual trajectories in the image plane are shown for all approaches. The initial position has been chosen the same in each case and inside the circle. As can be appreciated, the performance appears to be similar in all cases. Note that since the origin (0,0) is in the upper left corner, a minus sign has been added to have the y_2 in the right direction even though it is actually a positive value always. To have a better insight, we have computed the Root Mean Square Error (RMSE) as $RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n e_i^2}$ where i is the current sample number, e_i is the error associated to i and n is the total number of samples. As can be appreciated in Table 1, the performance of the proposed approach is better. Another advantage of fuzzy logic is that, on the contrary to the adaptive approaches, it does not need any regressor matrix and thus its implementation may be more direct.

Error	Proposed Algorithm	C.Algorithm 1	C.Algorithm 2
Δy_1	4.3123	5.7234	5.9960
Δy_2	10.7642	15.1442	14.1335

Table 1. Root mean square errors [PIXELS].

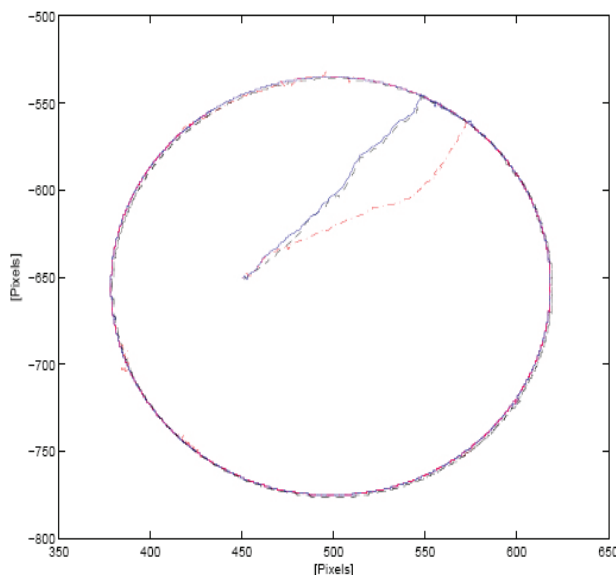


Figure 3. Trajectory in the image plane (y_1, y_2). Desired trajectory (---). Proposed Algorithm (—). Comp. Algorithm 1 (- · -). Comp. Algorithm 2 (- - -).

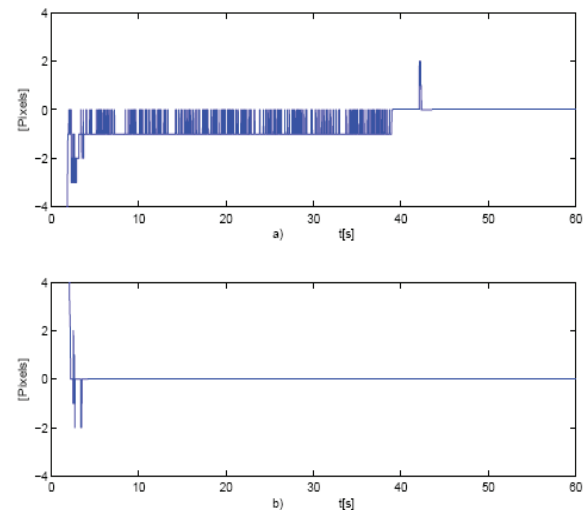


Figure 4. Observer errors for the proposed algorithm. a) \tilde{y}_1 . b) \tilde{y}_2 .

Finally, the observation errors for the proposed algorithm are shown in Figure 4. While the error \tilde{y}_2 becomes zero, \tilde{y}_1 has a value of 1[pixel] during more than half the experiment to become zero in the end.

6. Conclusions

This paper presents a new fuzzy tracking control scheme for visual servoing applied in a robot manipulator. Based on the T-S model, a fuzzy observer and a fuzzy controller are developed to reduce the tracking errors arbitrarily. Furthermore, the stability of the closed-loop nonlinear system is discussed. The advantage of the proposed tracking control design is that only a simple fuzzy controller is used without feedback linearization or an adaptive scheme. A state space model for a robot manipulator with two degrees of freedom in image coordinates has been suggested which can be employed with other controllers. Experimental results are shown by comparing the proposed scheme with well-known adaptive control laws. The outcomes clearly show the good performance of the algorithm introduced in this work.

Acknowledgment

This work was supported by the CONACYT under grant 58112 and by the DGAPA-UNAM under grant IN109611.

Appendix A

In order to implement the schemes for the experiments shown in Section 5, the dynamic model of robot A465 is necessary for implementation. By considering (1) as two degrees of freedom manipulator and computing the derivative of (2) one gets the differential perceptual kinematic model given by $\dot{y} = \alpha_\lambda R_\phi \dot{x}_R$, with $\dot{x}_R = J(q)\dot{q}$. $J(q) = \partial f_k(q) / \partial q$ is the so-called geometrical Jacobian

matrix of the robot Sciavicco and Siciliano (2000). As long as the robot is not in a singularity, one gets $\dot{q} = \frac{1}{\alpha} J^{-1}(q) R_{\phi} \dot{y}$. This relationship defines the visual mapping that relates the velocity in Cartesian space with the velocity in image space. To use a visual control algorithm is necessary to rewrite the model of the robot manipulator in terms of image coordinates y . This is a rather standard procedure leading to

$$\bar{H}(q)\ddot{y} + \bar{C}(q, \dot{q})\dot{y} + \bar{g}(q) = \bar{\tau} - \bar{\tau}_p, \quad (57)$$

where

$$\begin{aligned} \bar{H}(q) &= \frac{1}{\alpha} R_{\phi} J^{-T}(q) H(q) J^{-1}(q) R_{\phi} \\ \bar{C}(q, \dot{q}) &= \frac{1}{\alpha} R_{\phi} J^{-T}(q) \left[H(q) J^{-1}(q) \left[R_{\phi} + C(q, \dot{q}) J^{-1}(q) R_{\phi} + D(q) J^{-1}(q) R_{\phi} \right] \right] \\ \bar{g}(q) &= R_{\phi} J^{-T}(q) g(q) \\ \bar{\tau}(q) &= R_{\phi} J^{-T}(q) \end{aligned}$$

The parameters of model (57) can be found in Pérez et al. (2009) and the references cited there. Note that it is exclusively used to generate the state space representation (A_1, B_1, C_1) for the Takagi–Sugeno model. Because of lack of room, we write down only the matrices for the operation point p_1 . In this case one has

$$\begin{aligned} A_1 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 3.9 & -0.001 & -0.32 & -8.4 \cdot 10^{-6} \\ 0 & 0 & 0 & 1 \\ -6.9 & 0.002 & 3.2 & 6.2 \cdot 10^{-6} \end{bmatrix} \\ B_1 &= \begin{bmatrix} 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & 2 \end{bmatrix}^T \\ C_1 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{aligned}$$

Achieving the conditions given in Theorem 3.1 can be readily done by using the MATLAB LMI Toolbox, so that the control–observer gains are found to be

$$\begin{aligned} L_1 &= \begin{bmatrix} 3.2458 \times 10^2 & -5.5324 \times 10^2 \\ 1.2084 \times 10^2 & -2.7651 \times 10^2 \\ -5.7012 \times 10^2 & 1.0698 \times 10^2 \\ -7.1989 \times 10^2 & 7.1763 \times 10^2 \end{bmatrix} \\ F_1 &= \begin{bmatrix} -263.49 & -100.42 & -143.45 & -54.11 \\ -164.11 & -63.37 & -138.30 & -53.05 \end{bmatrix} \\ P_1 &= \begin{bmatrix} 0.0034 & 0.0005 & -0.0004 & 0.000 \\ 0.0005 & 0.0005 & 0.0000 & 0.000 \\ -0.0004 & 0.0000 & 0.0190 & 0.0011 \\ 0.0000 & 0.0000 & 0.0011 & 0.0004 \end{bmatrix} \\ P_{01} &= \begin{bmatrix} 0.6571 & 0.0010 & 0.0013 & 0.0161 \\ 0.0010 & 0.6615 & 0.0044 & 0.0764 \\ 0.0013 & 0.0044 & 0.0054 & 0.0565 \\ 0.0161 & 0.0764 & 0.0565 & 0.8634 \end{bmatrix} \end{aligned}$$

7. References

- [1] Abdelmalek, I. (2009). *Non linear systems control: LMI fuzzy approach*, Master's thesis, Université Hadj Lakhdar de Batna.
- [2] Abdelmalek, I., Golea, N. and Hadjili, M. L. (2007). A new fuzzy Lyapunov approach to non quadratic stabilization of Takagi–Sugeno fuzzy models, *International Journal of Applied Mathematics and Computer Science* 17(1): 39–51.
- [3] Fares, E. A., Elbardiny, M. and M.Sharaf, M. (2007). Adaptive fuzzy logic controller of visual servoing robot system by membership optimization using genetic algorithms, *Computer Engineering and Systems*, 2007. ICCES.
- [4] Goncalves, P., Mendonca, L., Sousa, J. and Pinto, J. (2008). Uncalibrated eye-to-hand visual servoing using inverse fuzzy models, *IEEE Transactions On Fuzzy Systems* 16(2): 341–353.
- [5] Kelly, R. and Reyes, F. (2000). On vision systems identification with application to fixed camera robotic systems, *International Journal of Imaging Systems and Technology* 11(3): 170–180.
- [6] Kim, C., Seol, W. H., Han, S. H. and Khatib, O. (2001). Fuzzy logic control of a robot manipulator based on visual servoing, *ISIE 2001. IEEE International Symposium on Industrial Electronics*. Pusan, Korea.
- [7] Liu, Y. and Lam, K. (2006). Uncalibrated visual servoing of robots using a depth-independent interaction matrix, *IEEE Transactions On Robotics Systems* 22(4): 804–817.
- [8] Meza, A. G. (2003). *Observadores difusos y control adaptable difuso basado en observadores*, Master's thesis, Cinvestav.
- [9] Parra-Vega, V., Fierro-Rojas, J. and Espinosa-Romero, A. (2003). Adaptive sliding mode uncalibrated visual servoing for finite-time tracking of 2D robot, *International Conference on Robotics and Automation* 3(3): 3048–3054.
- [10] Pérez, R., Arteaga-Pérez, M., Kelly, R. and Espinosa, A. (2009). On output regulation of direct visual servoing via velocity fields, *International Journal of Control* 82(4).
- [11] Sciavicco, L. and Siciliano, B. (2000). *Modeling and Control of Robot Manipulators*, London, Great Britain, Springer-Verlag.
- [12] Takagi, T. and 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.
- [13] Tanaka, K., Hori, T. and Wang, H. O. (2003). A multiple Lyapunov function approach to stabilization of fuzzy control systems, *IEEE Transactions On Fuzzy Systems* 11(4): 582–589.
- [14] Tanaka, K., Ikeda, T. and Wang, H. O. (1998). Fuzzy regulators and fuzzy observers: Relaxed stability

- conditions and LMI-based designs, *IEEE Transactions on Fuzzy Systems* 6(2): 250–265.
- [15] Tanaka, K. and Sugeno, M. (1992). Stability analysis and design of fuzzy control systems, *Fuzzy sets and systems* 45(2): 135–156.
- [16] Tanaka, K. and Wang, H. (2001). *Fuzzy Control Systems Design and Analysis. A Linear Matrix Inequality Approach*, John Wiley and Sons.
- [17] Wai, R. J. and Yang, Z.W. (2008). Adaptive fuzzy neural network control design via a T_S fuzzy model for a robot manipulator including actuator dynamics, *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*. 38(5): 1326–1346.
- [18] Zergeroglu, E., Dawson, D., de Queiroz, S. and Behal, A. (2001). Vision based nonlinear tracking controllers with uncertain robot camera parameters, *IEEE/ASME Transactions on Mechatronics* 6(3): 332–337.
- [19] Zhang, J. and Fei, M. (2006). Analysis and design of robust fuzzy controllers and robust fuzzy observers of nonlinear systems, *6th World Congress on Intelligent Control and Automation. Dalian, China*, Vol. 1, pp. 3767–3771.