2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Adaptive Vision and Force Tracking Control of Constrained Robots with Structural Uncertainties

Y. Zhao C. C. Cheah J.J.E. Slotine

School of Electrical and Electronic Engineering
Nanyang Technological University
Block S1, Nanyang Avenue, S(639798)
Republic of Singapore

Nonlinear Systems Laboratory
Massachusetts Institute of Technology
77 Massachusetts Ave
Cambridge, MA 02139 USA

***Abstract*— In many applications of robot manipulators, the end-effector is required to make contact with environment. In these applications, it is necessary to control not only the position but also the interaction force between the robot end-effector and environment. Most research so far on motion and force tracking control has assumed that the kinematics and constraint surface are exactly known. In this paper, we propose a visually-servoed adaptive Jacobian controller for motion and force tracking control with structural uncertainties in kinematics, dynamics and constraint surface. It is shown that uniform ultimate boundedness of the tracking errors can be guaranteed. Simulation results are presented to illustrate the performance of the proposed control law.**

I. Introduction

Most of industrial robots are used only for positioning tasks such as spray-painting and spot welding. To expand the applications of robots, such as polishing and deburring, it is important to control not only position but also force of interaction between the robot and environment. Many control methods have been developed for force control of robot manipulators. Despite the diversity of approaches, most of them can be classified into two categories: impedance control [1] and hybrid position/force control [2]. A review for the research in force control can be found in [3]. Several model based approaches have been proposed for motion and force tracking control [4], [5] of robots and these controllers can achieve very good performance when the system is well calibrated. However, exact kinematic and dynamic models of the robot system are required in these approaches, which means that the robot can not adapt to changes and uncertainties in the models. For example, when the robot picks up different tools of unknown lengths, the kinematics and dynamics of the robot changes and are difficult to derive exactly.

To alleviate this problem, much effort has been devoted to understand how the robot cope with dynamic uncertainties. Several adaptive motion and force control laws have been proposed to deal with dynamics uncertainties [6]–[9] but these controllers have assumed that the kinematics of the robot is exactly known.

Recently, several approximate Jacobian controllers [10]–[12] have been proposed to overcome the uncertainties in both kinematics and dynamics. The proposed controllers do not require the exact knowledge of kinematics and Jacobian matrix. However, the results in [10]–[12] are focusing on free motion control of robot where the robot end-effector is not in contact with the environment. In order to expand the feasible applications of robots, several position and force controllers [13]–[15] using approximate Jacobian have been proposed to overcome the uncertainties in both kinematics and dynamics. These controllers do not need the exact knowledge of kinematics and dynamics but the results are limited to setpoint control or point-to-point control of robot manipulators. In some applications, it is necessary to specify the motion in much more details than simply stating the desired final position. Thus, a desired trajectory should be specified. Recently, an adaptive Jacobian controller [16] is proposed for motion and force tracking control with uncertain kinematics and dynamics. In this result, the task space is defined as Cartesian space and hence the constraint surface is assumed to be known exactly. This is due to the fact that the desired trajectory on the constraint surface cannot be obtained if the constraint surface is not known exactly. To overcome this problem, a vision and force tracking controller [17] with uncertain dynamics, kinematics and constraint surface is proposed. However, in this result, the structure of the constraint surface is assumed to be known and the uncertain parameters of the constraint Jacobian are assumed to be linearly parameterizable. In addition, the normal direction of the constraint surface is assumed to be known. In most applications of constrained robots, it is hard to determine the structure of the constraint function. In the presence of uncertainty, it is also difficult to obtain the normal direction of the contact force exactly.

In this paper, we extend the result in [17] to a neural-network vision and force tracking controller. This vision-force controller does not need exact knowledge of kinematics, dynamics, camera model and constraint surface. In addition, the structure of the constraint surface and exact normal direction of the contact force are not required. The use of vision sensor introduces additional uncertainty and transformation from Cartesian space to image space and the motion and force errors are defined in two different coordinate frames. A Lyapunov function is presented to prove the stability of the proposed vision-force controller. It is shown that uniform ultimate boundedness can be guaranteed with uncertainties in kinematics, dynamics, camera model and constraint surface. Simulation results are presented to show the effectiveness of the proposed controller.

II. Robot Dynamics and Kinematics

We consider a vision-force control system consisting of a robot manipulator and camera(s) fixed in the work space. In this system, the end effector is in contact with a constraint surface. First, let $r \in \Re^m$ denote a position of the end-effector in Cartesian space as [11] [18]–[20],

$$r = h(q) \tag{1}$$

where $h(\cdot) \in \Re^n \rightarrow \Re^m$ is generally a non-linear transformation describing the relation between joint space and task space, $q = [q_1, \cdots, q_n]^T \in \Re^n$ is a vector of joint angles of the manipulator.

The velocity of the end-effector $\dot{r}$ is related to joint-space velocity $\dot{q}$ as:

$$\dot{r} = J_m(q)\dot{q} \quad (2)$$

where $J_m(q) \in \Re^{m\times n}$ is the Jacobian matrix from joint space to task space.

For a visually-servoed system, cameras are used to observe the position of the end-effector in image space. The mapping from Cartesian space to image space requires a camera-lens model in order to represent the projection of task objects onto the CCD image plane. We use the standard pinhole camera model, which has been proven adequate for most visual servoing tasks [21]. Let $x \in \Re^m$ denote a vector of image feature parameters and $\dot{x}$ the corresponding vector of image feature parameter rates of change. The relationship between Cartesian space and image space is represented by [21],

$$\dot{x} = J_I(r)\dot{r}, \quad (3)$$

where $J_I(r) \in \Re^{m\times m}$ is the image Jacobian matrix. The image Jacobian was first introduced by Weiss et al. [22], who referred to it as the feature sensitivity matrix. It is also referred to as the interaction matrix [23] and the B matrix [24], [25].

From equations (2) and (3), we have,

$$\dot{x} = J_I(r)J_m(q)\dot{q} = J(q)\dot{q}, \quad (4)$$

where $J(q) \in \Re^{m\times n}$ is the Jacobian matrix mapping from joint space to image space.

The equations of motion of constrained robot with n degree of degrees of freedom can be expressed in joint coordinates as [20] [26]:

$$M(q)\ddot{q} + (\frac{1}{2}\dot{M}(q) + S(q,\dot{q}))\dot{q} + g(q) = \tau + J_m^T(q)f \quad (5)$$

where $M(q) \in \Re^{n\times n}$ is the inertia matrix, $S(q,\dot{q})$ is a symmetric matrix, $\tau \in \Re^n$ is the applied joint torque to the robot, $f \in \Re^n$ is a contact force vector and $g(q) \in \Re^n$ is the gravitational force.

We consider a constraint surface that can be defined in an algebraic term as :

$$\Psi(r) = 0, \quad (6)$$

where $\Psi(r) : \Re^m \to \Re^1$ is a given scalar function. Differentiating equation (6) with respect to time yields the following velocity constraint:

$$\frac{\partial\Psi(r)}{\partial r}\dot{r} = 0. \quad (7)$$

The contact force on constraint surface is then given by

$$f = d(r)\lambda, \quad (8)$$

where $d(r) = \frac{(\frac{\partial\Psi(r)}{\partial r})^T}{\|\frac{\partial\Psi(r)}{\partial r}\|} \in \Re^m$ is a unit vector denotes a normal direction to the constraint surface, $\lambda \in \Re$ is defined as a magnitude of the contact force. Hence, equation (5) can be represented as

$$M(q)\ddot{q} + (\frac{1}{2}\dot{M}(q) + S(q,\dot{q}))\dot{q} + g(q) = \tau + D^T(q)\lambda, \quad (9)$$

where $D(q) = \frac{(\partial\Psi(r)/\partial r)^T}{\|\partial\Psi(r)/\partial r\|}J_m(q)$ is a Jacobian of the constraint function such that

$$D(q)\dot{q} = 0. \quad (10)$$

III. Adaptive Neural-network Vision and Force Tracking Control

In some force control applications, the uncertain parameters of the constraint surface can not be linearly separated or the structure of the constraint surface is unknown. In this section, we consider a robot with unknown dynamics and kinematics, The parameters and structure of the constraint surface are also unknown. An adaptive neural-network vision and force tracking controller is developed to deal with the above mentioned uncertainties.

Neural network has many important properties. For control purposes, the ability to approximate an arbitrary nonlinear function $f(x)$ up to a small error is the most important property. In order to approximate a function, an approximating function is chosen first, then the weights W are updated according to an algorithm based on the output errors [27]. For this purpose, different types of neural networks architecture can be used, such as multi-layer networks and Radial basis function (RBF) networks. In this paper, the neural network is designed so that it can be linearly parameterized and update law can be used to update the weights of the neural network online. The RBF network is suitable for this case and is used in this paper. The function approximation using a RBF network is [28]–[30]

$$f(x) = W\theta(x) + E, \quad (11)$$

where W is the matrix of neural network weights, E is called neural network functional approximation error, it generally decreases when the number of neurons increases. $\theta(x)$ is the activation function. There are many kinds of activation functions that can be chosen for RBF networks. It has shown that a linear superposition of Gaussian radial basis function results in an optimal mean square approximation to an unknown function which is infinitely differentiable and whose values are specified by a finite set of points in $\Re$ [29]. Therefore, Gaussian RBF networks are used in this paper. The Gaussian function is given as [27]

$$\theta(x) = exp[\frac{-(x-\mu)^2}{\sigma^2}], \quad (12)$$

where μ is called center and σ is distance. In this paper, the weight matrix is updated online and the update law will be derived from Lyapunov method.

The manipulator Jacobian $J_m(q)$ and the Jacobian matrix $J(q)$ in equation (4) can be approximated by neural networks as

$$J_m^T(q) = (W_{m1}\theta_m(q), ..., W_{mm}\theta_m(q)) + E_m, \quad (13)$$
$$J(q) = (W_{x1}\theta_x(q), ..., W_{xn}\theta_x(q)) + E_x, \quad (14)$$

where $W_{mi}(i = 1....m)$ and $W_{xi}(i = 1....n)$ are matrices of neural network weights, $\theta_m(q)$ and $\theta_x(q)$ are vectors of activation functions, E_m and E_x are bounded and small approximation errors.

When $J_m(q)$ and $J(q)$ are uncertain, they are estimated as

$$\hat{J}_m^T(q,\hat{W}_m) = (\hat{W}_{m1}\theta_m(q), ..., \hat{W}_{mm}\theta_m(q)),$$
$$\hat{J}(q,\hat{W}_x) = (\hat{W}_{x1}\theta_x(q), ..., \hat{W}_{xn}\theta_x(q)), \quad (15)$$

where $\hat{J}_m(q,\hat{W}_m)$ and $\hat{J}(q,\hat{W}_x)$ are estimations of $J_m(q)$ and $J(q)$ respectively and the estimated neural network weights $\hat{W}_m$ and $\hat{W}_x$ will be updated by update laws to be defined later.

Next, a vector $\dot{x}_r \in \Re^m$ is defined as,

$$\dot{x}_r = (\dot{x}_d - \alpha\Delta x) + \beta(\hat{J}_m(q,\hat{W}_m)\hat{J}^+(q,\hat{W}_x))^{-1}R\hat{d}(r)\Delta F, \quad (16)$$

where α and β are positive constants, $x_d(t) \in \Re^m$ is the desired image motion trajectory and $\dot{x}_d(t) \in \Re^m$ is the desired speed trajectory, $\Delta x = x - x_d$ is image motion tracking error, $\hat{J}(q,\hat{W}_x)$ is an estimation of $J(q)$, $\hat{J}^+(q,\hat{W}_x)$ is the pseudo inverse of $\hat{J}(q,\hat{W}_x)$ and $\Delta F = \int_0^t(\lambda(\sigma) - \lambda_d(\sigma))d\sigma$, $\lambda_d(t)$ is the desired force trajectory, R is a rotation matrix which will be defined later, $\hat{d}(r)$ is a fixed estimation of $d(r)$.

The estimation error $J_m(q)(d(r)-\hat{d}(r))$ is approximated by neural networks as

$$J_m(q)(d(r)-\hat{d}(r)) = W_f\theta_f(q) + E_f \tag{17}$$

where where W_f is a matrix of neural network weights, $\theta_f(q)$ is a vector of activation functions and E_f is a vector of approximation errors that is bounded and small.

In order to prove the stability of the vision-force tracking system, an adaptive sliding vector is defined using equation (16) as,

$$\hat{s}_x = \dot{\hat{x}} - \dot{x}_r = \hat{J}(q,\hat{W}_x)\dot{q} - \dot{x}_r, \tag{18}$$

Differentiating the above equation with respect to time, one has,

$$\dot{\hat{s}}_x = \ddot{\hat{x}} - \ddot{x}_r = \hat{J}(q,\hat{W}_x)\ddot{q} + \dot{\hat{J}}(q,\hat{W}_x)\dot{q} - \ddot{x}_r, \tag{19}$$

Next, let

$$\dot{q}_r = \hat{J}^+(q,\hat{W}_x)\dot{x}_r + (I_n - \hat{J}^+(q,\hat{W}_x)\hat{J}(q,\hat{W}_x))\psi, \tag{20}$$

where $\psi \in \Re^n$ is minus the gradient of the convex function to be optimized [31].

An adaptive sliding vector is defined in joint space as,

$$s = \dot{q} - \dot{q}_r, \tag{21}$$

and

$$\dot{s} = \ddot{q} - \ddot{q}_r. \tag{22}$$

Multiplying both side of equation (21) by $\hat{J}(q,\hat{W}_x)$ and using equation (18), one has

$$\hat{J}(q,\hat{W}_x)s = \hat{J}(q,\hat{W}_x)\dot{q} - \dot{x}_r = \hat{s}_x, \tag{23}$$

Substitute equations (21) and (22) into equation (9) to get,

$$\begin{aligned} &M(q)\dot{s} + (\tfrac{1}{2}\dot{M}(q) + S(q,\dot{q}))s + M(q)\ddot{q}_r \\ &+(\tfrac{1}{2}\dot{M}(q) + S(q,\dot{q}))\dot{q}_r + g(q) \\ &= \tau + D^T(q)\lambda, \end{aligned} \tag{24}$$

The last three terms on the left hand side of equation (24) can be expressed as

$$\begin{aligned} &M(q)\ddot{q}_r + (\tfrac{1}{2}\dot{M}(q) + S(q,\dot{q}))\dot{q}_r + g(q) \\ &= W_d\ \theta_d(q,\dot{q},\dot{q}_r,\ddot{q}_r) + E_d. \end{aligned} \tag{25}$$

where W_d is a matrix of neural network weights, $\theta_d(q,\dot{q},\dot{q}_r,\ddot{q}_r)$ is a vector of activation functions and E_d is a vector of approximation errors that is bounded and small. Then the dynamics equation (24) can be expressed as:

$$\begin{aligned} &M(q)\dot{s} + (\tfrac{1}{2}\dot{M}(q) + S(q,\dot{q}))s \\ &+W_d\ \theta_d(q,\dot{q},\dot{q}_r,\ddot{q}_r) + E_d \\ &= \tau + D^T(q)\lambda, \end{aligned} \tag{26}$$

The vision and force tracking controller is proposed as:

$$\begin{aligned} \tau &= -\hat{J}^T(q,\hat{W}_x)K_p\Delta x - K_v s \\ &+\hat{W}_d\ \theta_d(q,\dot{q},\dot{q}_r,\ddot{q}_r) - \hat{J}_m^T(q,\hat{W}_m)\hat{d}(r)\lambda - \hat{W}_f\theta_f(q)\lambda \\ &+\hat{J}_m^T(q,\hat{W}_m)R\hat{d}(r)(\kappa\Delta\lambda + \gamma\Delta F) - K_m\hat{d}(r)\lambda - K_f\lambda, \end{aligned} \tag{27}$$

where $\Delta\hat{x} = \hat{x} - x_d$, κ and α are positive constants, K_p and K_v are positive diagonal gain matrices, K_m is a matrix designed to compensate the estimation error of $\hat{J}_m^T(q,\hat{W}_m)$,

$$k_{mij} = -\bar{k}_{mij}sgn(s_i\hat{d}_j(r)) \tag{28}$$

and k_{mij} is the element in the i^{th} row and j^{th} column of K_m, K_f is a vector to compensate the estimation error of $J_m(q)(d(r)-\hat{d}(r))$ and $K_{fi} = k_{fi}sgn(s_i)$. The estimated parameters $\hat{W}_d$, $\hat{W}_m$ and $\hat{W}_x$ are updated by,

$$\dot{\hat{W}}_{di}^T = proj(\Omega_{di}), \tag{29}$$

$$\dot{\hat{W}}_{mij}^T = proj(\Omega_{mij}), \tag{30}$$

$$\dot{\hat{W}}_{xij}^T = proj(\Omega_{xij}), \tag{31}$$

$$\dot{\hat{W}}_{fi}^T = proj(\Omega_{fi}), \tag{32}$$

where

$$\begin{aligned} \Omega_{di} &= k_1\hat{W}_{di}^T - L_{di}s_i\theta_d(q,\dot{q},\dot{q}_r,\ddot{q}_r), \\ \Omega_{mij} &= k_2\hat{W}_{mij}^T + L_{mij}s_i\hat{d}_j(r)\lambda\theta_m(q), \\ \Omega_{xij} &= k_3\hat{W}_{xij}^T + L_{xij}\Delta x_i k_{pi}\dot{q}_j\theta_x(q) \\ \Omega_{fi} &= k_4\hat{W}_{fi}^T + L_{fi}s_i\theta_f(q), \end{aligned} \tag{33}$$

where $\hat{W}_{di}$, $\hat{W}_{mij}$, $\hat{W}_{xij}$ and $\hat{W}_{fi}$ are the i^{th} row vectors of $\hat{W}_d$, $\hat{W}_{mj}$, $\hat{W}_{xi}$ and $\hat{W}_f$, s_i, $\hat{d}_i(r)$, $\dot{q}_i$ are the i^{th} elements of s, $\hat{d}(r)$ and $\dot{q}$, Δx_j is the j^{th} element of Δx_j and k_{pj} is the j^{th} element of the diagonal matrix K_p, $L_{di} = l_{di}I, L_{mij} = l_{mij}I, L_{xij} = l_{xij}I, L_{fi} = l_{fi}I$ are positive gain matrices, k_1, k_2, k_3 and k_4 are positive constants and the function $proj(\Omega_d)$ is a projection algorithm defined as [32]

$$proj(\Omega_{di}) = \begin{cases} \Omega_{di} & if \quad \hat{W}_{di} > \underline{W}_{di} \\ \Omega_{di} & if \quad \hat{W}_{di} = \underline{W}_{di} \text{ and } \Omega_{di} \geq 0 \\ 0 & if \quad \hat{W}_{di} = \underline{W}_{di} \text{ and } \Omega_{di} < 0 \\ 0 & if \quad \hat{W}_{di} = \bar{W}_{di} \text{ and } \Omega_{di} > 0 \\ \Omega_{di} & if \quad \hat{W}_{di} = \bar{W}_{di} \text{ and } \Omega_{di} \leq 0 \\ \Omega_{di} & if \quad \hat{W}_{di} < \bar{W}_{di} \end{cases} \tag{34}$$

where $\underline{W}_{di}$ and $\bar{W}_{di}$ are the lower and upper bounds of W_{di}. The projection algorithms $proj(\Omega_m)$ and $proj(\Omega_d)$ can be similarly defined as above. The functions $proj(\cdot)$ are defined to ensure that $\hat{J}^T(q,\hat{W}_x)$ and $\hat{J}_m(q,\hat{W}_m)$ are bounded during adaption.

In the above controller, R is a rotation matrix designed [13] so that

$$s_{xN}^T R\hat{d}(r) = 0, \tag{35}$$

where

$$\begin{aligned} s_{xN} &= \{\beta\Delta x^T K_p(\hat{J}_m(q,\hat{W}_m)\hat{J}^+(q,\hat{W}_x))^{-1}\Delta F \\ &\quad + s_m^T(\kappa\Delta\lambda + \gamma\Delta F)\}^T \\ s_m &= \hat{J}_m(q,\hat{W}_m)\{\dot{q} - \hat{J}^+(q,\hat{W}_x)(\dot{x}_d - \alpha\Delta x) \\ &\quad -(I_n - \hat{J}^+(q,\hat{W}_x)\hat{J}(q,\hat{W}_x))\psi\}, \end{aligned} \tag{36}$$

Substituting equation (27) into equation (26), the closed-loop equation is obtained as

$$\begin{aligned} &M(q)\dot{s} + (\tfrac{1}{2}\dot{M}(q) + S(q,\dot{q}))s + \hat{J}^T(q,\hat{W}_x)K_p\Delta x + K_v s \\ &+\Delta W_d\ \theta_d(q,\dot{q},\dot{q}_r,\ddot{q}_r) + E_d \\ &= \Delta W_m\theta_m(q)\hat{d}(r)\lambda + E_m\hat{d}(r)\lambda + \Delta W_f\theta_f(q)\lambda + E_f\lambda \\ &+\hat{J}_m^T(q,\hat{W}_m)R\hat{d}(r)(\kappa\Delta\lambda + \gamma\Delta F) - K_m\hat{d}(r)\lambda - K_f\lambda, \end{aligned} \tag{37}$$

where $\Delta W_d = W_d - \hat{W}_d$, $\Delta W_m = W_m - \hat{W}_m$ and $\Delta W_f = W_f - \hat{W}_f$.

To carry out the stability analysis, the Lyapunov-like function candidate V is defined as:

$$\begin{aligned} V &= \tfrac{1}{2}s^T M(q)s + \tfrac{1}{2}\Delta x^T K_p\Delta x + \tfrac{1}{2}\Sigma_{i=1}^n\Delta W_{di}L_{di}^{-1}\Delta W_{di}^T \\ &+\tfrac{1}{2}\Sigma_{i=1}^n\Sigma_{j=1}^m\Delta W_{mij}L_{mij}^{-1}\Delta W_{mij}^T + \tfrac{1}{2}\Sigma_{i=1}^n\Delta W_{fi}L_{fi}^{-1}\Delta W_{fi}^T \\ &+\tfrac{1}{2}\Sigma_{i=1}^m\Sigma_{j=1}^n\Delta W_{xij}L_{xij}^{-1}\Delta W_{xij}^T + \tfrac{1}{2}\beta\kappa\Delta F^2. \end{aligned} \tag{38}$$

where $\Delta W_{xij} = W_{xij} - \hat{W}_{xij}$. Differentiating V with respect of time yields,

$$\begin{aligned} \dot{V} &= s^T M(q)\dot{s} + \tfrac{1}{2}s^T \dot{M}(q)s \\ &+\Delta x^T K_p \Delta\dot{x} - \Sigma_{i=1}^n \Delta W_{di}^T L_{di}^{-1}\dot{\hat{W}}_{di} \\ &-\Sigma_{i=1}^n \Delta W_{fi}^T L_{fi}^{-1}\dot{\hat{W}}_{fi} - \Sigma_{i=1}^n\Sigma_{j=1}^m \Delta W_{mij} L_{mij}^{-1}\dot{\hat{W}}_{mij}^T \\ &-\Sigma_{i=1}^m\Sigma_{j=1}^n \Delta W_{xij} L_{xij}^{-1}\dot{\hat{W}}_{xij}^T + \beta\kappa\Delta F\Delta\lambda. \end{aligned} \quad (39)$$

From equations (21) and (16), note that

$$\begin{aligned} \hat{J}_m(q,\hat{W}_m)s &= \hat{J}_m(q,\hat{W}_m)\dot{q} \\ &-\hat{J}_m(q,\hat{W}_m)\hat{J}^+(q,\hat{W}_x)(\dot{x}_d - \alpha\Delta x) \\ &-\hat{J}_m(q,\hat{W}_m)(I_n - \hat{J}^+(q,\hat{W}_x)\hat{J}(q,\hat{W}_x))\psi - \beta R\hat{d}(r)\Delta F \\ &= s_m - \beta R\hat{d}(r)\Delta F. \end{aligned} \quad (40)$$

Substitute equations(37), (18), (16) and (40) into equation (39) and using equations (35) and (18), one has

$$\begin{aligned} \dot{V} &= -\alpha\Delta x^T K_p\Delta x - s^T K_v s - \Delta x^T K_p \Delta\dot{\hat{x}} \\ &+\Delta x^T K_p\Delta\dot{x} - \beta\gamma\Delta F^2 - s^T E_d - s^T(K_m - E_m)\hat{d}(r)\lambda \\ &-s^T(K_f - E_f)\lambda - s^T\Delta W_d\ \theta_d(q,\dot{q},\dot{q}_r,\ddot{q}_r) \\ &+s^T\Delta W_m\theta_m(q)\hat{d}(r)\lambda + s^T\Delta W_f\theta_f(q)\lambda \\ &-\Sigma_{i=1}^m\Sigma_{j=1}^n\Delta W_{xij}L_{xij}^{-1}\dot{\hat{W}}_{xij}^T - \Sigma_{i=1}^n\Delta W_{di}L_{di}^{-1}\dot{\hat{W}}_{di}^T \\ &-\Sigma_{i=1}^n\Sigma_{j=1}^m\Delta W_{mij}L_{mij}^{-1}\dot{\hat{W}}_{mij}^T - \Sigma_{i=1}^n\Delta W_{fi}L_{fi}^{-1}\dot{\hat{W}}_{fi}^T, \end{aligned} \quad (41)$$

where $R^T R = I$ and $\hat{d}^T(r)\hat{d}(r) = 1$. From equations (14) and (15), since $\dot{x} = \dot{\hat{x}} + \Delta W_x\theta_x(q)\dot{q} + E_x\dot{q}$, one has

$$\Delta\dot{x} = \Delta\dot{\hat{x}} + \Delta W_x\theta_x(q)\dot{q} + E_x\dot{q}. \quad (42)$$

Substituting equations (31) and (42) into equation (41), using (29)-(34), gives

$$\begin{aligned} \dot{V} &\le -\alpha\Delta x^T K_p\Delta x - s^T K_v s + \Delta x^T K_p E_x\dot{q} - \beta\gamma\Delta F^2 \\ &-s^T E_d - s^T(K_m - E_m)\hat{d}(r)\lambda \\ &-k_3\Sigma_{i=1}^m\Sigma_{j=1}^n\Delta W_{xij}L_{xij}^{-1}\hat{W}_{xij}^T - k_1\Sigma_{i=1}^n\Delta W_{di}L_{di}^{-1}\hat{W}_{di}^T \\ &-k_2\Sigma_{i=1}^n\Sigma_{j=1}^m\Delta W_{mij}L_{mij}^{-1}\hat{W}_{mij}^T - k_4\Sigma_{i=1}^n\Delta W_{fi}L_{fi}^{-1}\hat{W}_{fi}^T. \end{aligned} \quad (43)$$

Let $\bar{\gamma} = min\{\bar{\gamma}_1, \bar{\gamma}_2, \bar{\gamma}_3, k_1, k_2, k_3, k_4\}$, it can be shown that

$$\dot{V} \le -\bar{\gamma}V + \mu, \quad (44)$$

details of the proof from inequalities (43) to (44) can be found in appendix. The above inequality implies

$$V \le \frac{\mu}{\bar{\gamma}} + \{V(0) - \frac{\mu}{\bar{\gamma}}\}e^{-ct}. \quad (45)$$

Then it can be concluded that the system is uniformly ultimate bounded.

Theorem *The adaptive Jacobian control law (27) and the update laws (29), (31), (30) and (32) for the robot system (9) result in the uniformly ultimate boundedness of vision and force tracking errors when K_p, K_v, α and γ are chosen to satisfy condition (60). Moreover, the errors can be made arbitrarily small by adjusting the control gains.*

Proof: From inequality (45), it can be concluded that s, Δx, ΔW_d, ΔW_m, ΔW_x, ΔW_f and ΔF are uniformly ultimate bounded. This implies that $\hat{W}_d$, $\hat{W}_m$, $\hat{W}_x$, $\hat{W}_f$ and x are bounded, and $\hat{s}_x = \hat{J}(q,\hat{W}_x)s$ is also bounded. Next $\dot{x}_r$, $\dot{\hat{x}}$ are bounded as seen from equations (16) and (18). From equation (20) one can conclude that $\dot{q}_r$ is bounded when $\hat{J}(q,\hat{W}_x)$ is nonsingular. Therefore $\dot{q}$ is bounded since s is bounded. The boundedness of $\dot{q}$ means that $\dot{x}$, $\dot{r}$ are bounded. Hence $\Delta\dot{x}$ is bounded and $\dot{\hat{d}}(r,\dot{r})$ is also bounded because $r, \dot{r}$ are bounded.

Then from equations (37) and (25), one has

$$\begin{aligned} &M(q)\ddot{q} + (\tfrac{1}{2}\dot{M}(q) + S(q,\dot{q}))\dot{q}_r + g(q) \\ &+\hat{J}^T(q,\hat{W}_x)K_p\Delta x + K_v s \\ &= \Delta W_m\theta_m(q)\hat{d}(r)\lambda + E_m\hat{d}(r)\lambda + \Delta W_f\theta_f(q)\lambda + E_f\lambda \\ &+\hat{J}_m^T(q,\hat{W}_m)R\hat{d}(r)(\kappa\Delta\lambda + \gamma\Delta F) - K_m\hat{d}(r)\lambda - K_f\lambda. \end{aligned} \quad (46)$$

Since $D(q)\ddot{q} = -\dot{D}(q)\dot{q}$, one has

$$\begin{aligned} &-\dot{D}(q)\dot{q} + D(q)M^{-1}(q)r_1(t) \\ &= D(q)M^{-1}(q)\{(\Delta W_m\theta_m(q) + E_m - K_m)\hat{d}(r)\Delta\lambda \\ &+(\Delta W_f\theta_f(q) + E_f - K_f)\Delta\lambda + \kappa\hat{J}_m^T(q,\hat{W}_m)R\hat{d}(r)\Delta\lambda\} \end{aligned} \quad (47)$$

where $r_1(t) = (\frac{1}{2}\dot{M}(q) + S(q,\dot{q}))\dot{q}_r + g(q) + \hat{J}^T(q,\hat{W}_x)K_p\Delta x + K_v s - (\Delta W_m\theta_m(q) + E_m - K_m)\hat{d}(r)\lambda_d - (\Delta W_f\theta_f(q) + E_f - K_f)\lambda_d - \gamma\hat{J}_m^T(q,\hat{W}_m)R\hat{d}(r)\Delta F$.

The above equation can be written as:

$$\bar{r}_1(t) = k(t)\Delta\lambda \quad (48)$$

where

$$\bar{r}(t) = -\dot{D}(q)\dot{q} + D(q)M^{-1}(q)r_1(t) \quad (49)$$

and

$$\begin{aligned} k(t) &= D(q)M^{-1}(q)\{\Delta W_m\theta_m(q)\hat{d}(r) + E_m\hat{d}(r) \\ &+\Delta W_f\theta_f(q) + E_f - K_m\hat{d}(r) - K_f \\ &+\kappa\hat{J}_m^T(q,\hat{W}_m)R\hat{d}(r)\} \end{aligned} \quad (50)$$

are bounded scalars. Hence the force tracking error $\Delta\lambda$ is also bounded. △△△

Remark. Overfitting is a common problem in neural network design. However, in these techniques (as in [28]) overfitting is not a problem since the algorithms are designed to achieve task convergence rather than parameter convergence

IV. Simulation Results

In this section, simulation results are presented to illustrate the performance of the proposed controller. Consider a two-link manipulator whose end-effector is required to move on a constraint surface as shown in figure 1. A fixed camera is placed distance away from the robot.

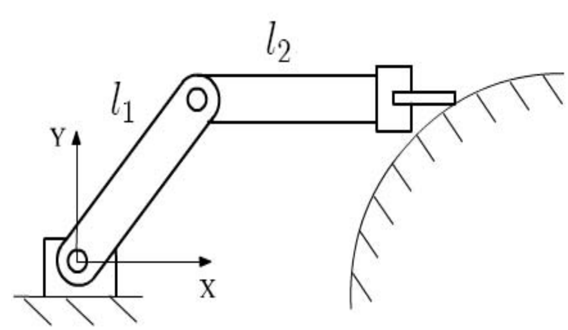

Fig. 1. A two-link robot in contact with a constraint surface

In this simulation, uncertain constraint surface and Jacobian matrix are considered. The constraint surface in Cartesian space is described by

$$\Psi(x) = sin(ax_1 + b) - x_2 = 0, \quad (51)$$

Note that in this constraint function, the parameters in $\Psi(x)$ can not be linearly separated.

An image path in image space is obtained from the camera, The initial position of the end effector on the path is set at $(520, 199)$. $x_d(t)$ is set as $x_d(t) = 520 + 5t$ pixels and $y_d(t)$ is obtained from the image path. The desired contact force is set as 20+5sin(2t) Newton. In this simulation, the control gains are set as $\alpha = 0.6, \beta = 0.01, \gamma = 15, \kappa = 0.5, K = 3.8 \times 10^{-3} I, K_v = 200 I,$.

In this simulation, Gaussian RBF neural networks with input q were used. The centers were chosen so that they were evenly distributed to span the input space of the network. The distance of neural networks was fixed at 1.1 and the number of neurons was set as 40. The gains for the networks were chosen as $k_1 = k_2 = k_3 = 0.001, L_d = L_m = L_x = 0.01, \bar{k}_m = 0.001$.

The simulation results are shown in figures 2, 3 and 4. The results show the effectiveness of the proposed controller in dealing with uncertain structure of constraint surface and Jacobian matrices.

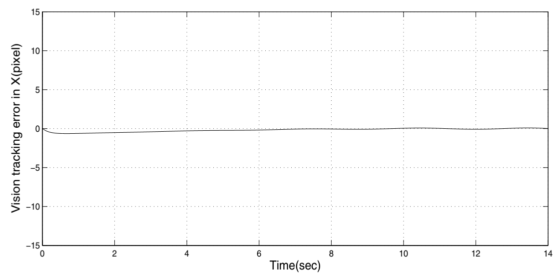

Fig. 2. Image tracking error in X

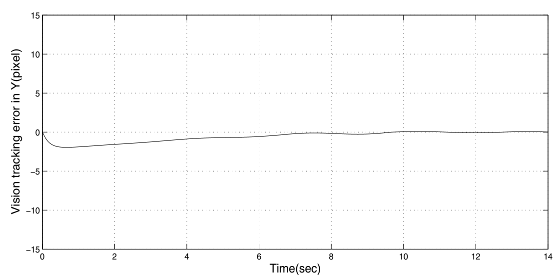

Fig. 3. Image tracking error in Y

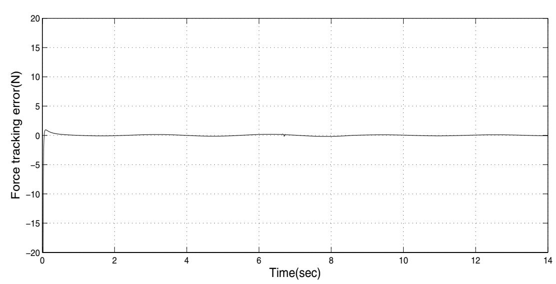

Fig. 4. Force tracking error

V. Conclusion

In this paper, the stability problem of visually-servoed motion and force tracking control system with uncertain kinematics, dynamics and constraint surface has been studied. A neural network Jacobian controller has also been proposed to deal with uncertain structure of the Jacobian matrices and constraint surface. A new Lyapunov-like function has also been presented for the stability analysis of the control systems. It has been shown that that uniformly ultimate boundedness can be guaranteed in the presence of the above-mentioned uncertainties. Simulation results have been presented to illustrate the performance of the proposed control law.

Appendix

The appendix presents the detailed derivation from inequality (43) to inequality (44):

From equations (20) and (21), one has

$$\begin{aligned} \dot{q} &= s + \dot{q}_r \\ &= s + \hat{J}^+(q, \hat{W}_x)\dot{x}_r + (I_n - \hat{J}^+(q, \hat{W}_x)\hat{J}(q, \hat{W}_x))\psi \\ &= s + \hat{J}^+(q, \hat{W}_x)(\dot{x}_d - \alpha\Delta x) \\ &+\beta\hat{J}^+(q, \hat{W}_x)(\hat{J}_m(q, \hat{W}_m)\hat{J}^+(q, \hat{W}_x))^{-1}R\hat{d}(r)\Delta F \\ &+(I_n - \hat{J}^+(q, \hat{W}_x)\hat{J}(q, \hat{W}_x))\psi. \end{aligned} \tag{52}$$

Substituting equation (52) into equation (43), one has

$$\begin{aligned} \dot{V} \leq & -\alpha\Delta x^T K_p \Delta x - s^T K_v s - \beta\gamma\Delta F^2 - s^T E_d \\ & -s^T(K_m - E_m)\hat{d}(r)\lambda - k_3\Sigma_{i=1}^m\Sigma_{j=1}^n \Delta W_{xij} L_{xij}^{-1}\hat{W}_{xij}^T \\ & -k_1\Sigma_{i=1}^n \Delta W_{di} L_{di}^{-1}\hat{W}_{di}^T - k_2\Sigma_{i=1}^n\Sigma_{j=1}^m \Delta W_{mij} L_{mij}^{-1}\hat{W}_{mij}^T \\ & +\Delta x^T K_p E_x s + \Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)\dot{x}_d \\ & -\alpha\Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)\Delta x \\ & +\beta\Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)(\hat{J}_m(q, \hat{W}_m)\hat{J}^+(q, \hat{W}_x))^{-1}R\hat{d}(r)\Delta F \\ & +\Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)(I_n - \hat{J}^+(q, \hat{W}_x)\hat{J}(q, \hat{W}_x))\psi. \end{aligned} \tag{53}$$

Next, note that

$$\begin{aligned} & s^T E_d \leq \tfrac{1}{2}(\|s\|^2 + \|E_d\|^2) \\ & \Delta x^T K_p E_x s \leq \tfrac{b_{ex}b_p}{2}(\|\Delta x\|^2 + \|s\|^2) \\ & \Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)\dot{x}_d \leq \tfrac{b_p b_{ex} b_1}{2}(\|\Delta x\|^2 + \|\dot{x}_d\|^2) \\ & \alpha\Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)\Delta x \leq \alpha b_p b_{ex} b_1\|\Delta x\|^2 \\ & \beta\Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)(\hat{J}_m(q, \hat{W}_m)\hat{J}^+(q, \hat{W}_x))^{-1}R\hat{d}(r)\Delta F \\ & \leq \tfrac{\beta b_p b_{ex} b_1 b_2}{2}(\|\Delta x\|^2 + \Delta F^2) \\ & \Delta x^T K_p E_x \hat{J}^+(q, \hat{W}_x)(I_n - \hat{J}^+(q, \hat{W}_x)\hat{J}(q, \hat{W}_x))\psi \\ & \leq \tfrac{b_p b_{ex} b_1 b_3}{2}(\|\Delta x\|^2 + \|\psi\|^2). \end{aligned} \tag{54}$$

where $b_{ex}, b_p, b_1, b_2, b_3$ are upper bounds of E_x, K_p, $\hat{J}^+(q, \hat{W}_x)$, $(\hat{J}_m(q, \hat{W}_m)\hat{J}^+(q, \hat{W}_x))^{-1}$ and $I_n - \hat{J}^+(q, \hat{W}_x)\hat{J}(q, \hat{W}_x)$. In addition,

$$\begin{aligned} k_1\Delta W_{di}L_{di}^{-1}\hat{W}_{di}^T &\geq \tfrac{k_1}{2l_{di}}(\|\Delta W_{di}\|^2 - \|W_{di}\|^2) \\ k_2\Delta W_{mij}L_{mij}^{-1}\hat{W}_{fij}^T &\geq \tfrac{k_2}{2l_{mij}}(\|\Delta W_{fij}\|^2 - \|W_{fij}\|^2), \\ k_3\Delta W_{xij}L_{xij}^{-1}\hat{W}_{xij}^T &\geq \tfrac{k_3}{2l_{xij}}(\|\Delta W_{xij}\|^2 - \|W_{xij}\|^2). \end{aligned} \tag{55}$$

Using inequalities (54) and (55), then $\dot{V}$ becomes

$$\begin{aligned} \dot{V} \leq & -(\alpha k_{pmin} - \bar{b})\|\Delta x\|^2 - (k_{vmin} - \tfrac{b_{ex}b_p + 1}{2})\|s\|^2 \\ & -s^T(K_m - E_m)\hat{d}(r)\lambda - \beta(\gamma - \tfrac{b_p b_{ex} b_1 b_2}{2})\Delta F^2 + \mu \\ & -\tfrac{k_1}{2l_{di}}\Sigma_{i=1}^n\|\Delta W_{di}\|^2 - \tfrac{k_3}{2l_{xij}}\Sigma_{i=1}^m\Sigma_{j=1}^n\|\Delta W_{xij}\|^2 \\ & -\tfrac{k_2}{2l_{mij}}\Sigma_{i=1}^n\Sigma_{j=1}^m\|\Delta W_{mij}\|^2, \end{aligned} \tag{56}$$

where

$$\begin{aligned} \bar{b} = & \tfrac{b_{ex}b_p}{2} + \tfrac{b_p b_{ex} b_1}{2} + \alpha b_p b_{ex} b_1 + \tfrac{\beta b_p b_{ex} b_1 b_2}{2} + \tfrac{b_p b_{ex} b_1 b_3}{2} \\ \mu = & \tfrac{1}{2}\|E_d\|^2 + \tfrac{b_p b_{ex} b_2}{2}b_d^2 + \tfrac{b_p b_{ex} b_1 b_3}{2}b_\psi^2 + \tfrac{k_1}{2l_{di}}\Sigma_{i=1}^n\|W_{di}\|^2 + \\ & \tfrac{k_3}{2l_{xij}}\Sigma_{i=1}^m\Sigma_{j=1}^n\|W_{xij}\|^2 + \tfrac{k_2}{2l_{mij}}\Sigma_{i=1}^n\Sigma_{j=1}^m\|W_{mij}\|^2, \end{aligned} \tag{57}$$

where b_d and b_ψ are upper bounds of $\dot{x}_d$ and ψ. When $|\bar{k}_{mij}|$ is set sufficiently large so that $|\bar{k}_{mij}| \geq |E_{mij}|$, one has

$$\begin{aligned} \dot{V} \leq & -(\alpha k_{pmin} - \bar{b})\|\Delta x\|^2 - (k_{vmin} - \tfrac{b_{ex}b_p + 1}{2})\|s\|^2 \\ & -\beta(\gamma - \tfrac{b_p b_{ex} b_1 b_2}{2})\Delta F^2 + \mu - \tfrac{k_1}{2l_{di}}\Sigma_{i=1}^n\|\Delta W_{di}\|^2 \\ & -\tfrac{k_3}{2l_{xij}}\Sigma_{i=1}^m\Sigma_{j=1}^n\|\Delta W_{xij}\|^2 - \tfrac{k_2}{2l_{mij}}\Sigma_{i=1}^n\Sigma_{j=1}^m\|\Delta W_{mij}\|^2, \end{aligned} \tag{58}$$

Let $k_{pmin}, k_{vmin}, \alpha, \beta, \gamma$ be chosen sufficiently large so that

$$\begin{aligned} k_{vmin} - \tfrac{b_{ex}b_p+1}{2} &> 0 \\ \alpha k_{pmin} - \bar{b} &> 0 \\ \gamma - \tfrac{b_p b_{ex} b_1 b_2}{2} &> 0. \end{aligned} \quad (59)$$

There exist positive constants $\bar{\gamma}_1,\ \bar{\gamma}_2,\ \bar{\gamma}_3$ such that

$$\begin{aligned} (k_{vmin} - \tfrac{b_{ex}b_p+1}{2})\|s\|^2 &\geq \tfrac{\bar{\gamma}_1}{2} s^T M(q) s \\ (\alpha k_{pmin} - \bar{b})\|\Delta x\|^2 &\geq \tfrac{\bar{\gamma}_2}{2} \Delta x^T K_p \Delta x \\ \beta(\gamma - \tfrac{b_p b_{ex} b_1 b_2}{2})\Delta F^2 &\geq \tfrac{\bar{\gamma}_3}{2} \beta\kappa\Delta F^2, \end{aligned} \quad (60)$$

Let $\bar{\gamma} = min\{\bar{\gamma}_1, \bar{\gamma}_2, \bar{\gamma}_3, k_1, k_2, k_3, k_4\}$, one has

$$\dot{V} \leq -\bar{\gamma} V + \mu. \quad (61)$$

REFERENCES

[1] N. Hogan, "Impedance control: An approach to manipulation: Part I-theory; part II-implementation; part III-applications," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 1–24, 1985.

[2] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 102, pp. 126–133, 1981.

[3] T. Yoshikawa, "Force control of robot manipulators," in *Proc. IEEE Conference on Robotics and Automation*, pp. 220–226, 2000.

[4] T. Yoshikawa, T. Sugie, and M. Tanaka, "Dynamic hybrid position/force control of robot manipulators: Controller design and experiment," *IEEE Journal on Robotics and Automation*, vol. 4, no. 6, pp. 699–705, 1988.

[5] N. H. McClamroch and D. Wang, "Feedback stabilization and tracking of constrained robots," *IEEE Transactions on Automatic Control*, vol. 33, no. 5, pp. 419–426, 1988.

[6] S. Arimoto, T. Naniwa, and Y. H. Liu, "Model-based adaptive hybrid control for manipulators with geometric endpoint constraint," *Advanced Robotics*, vol. 9, no. 1, pp. 67–80, 1995.

[7] R. Carelli and R. Kelly, "An adaptive impedance/force controller for robot manipulators," *IEEE Transactions on Automatic Control*, vol. 36, no. 8, pp. 967–971, 1991.

[8] V. Parra-Vega and S. Arimoto, "A passivity-based adaptive sliding mode position-force control for robot manipulators," *International Journal of Adaptive Control and Signal Processing*, vol. 10, no. 4-5, pp. 365–377, 1996.

[9] L. L. Whitcomb, S. Arimoto, T. Naniwa, and F. Ozaki, "Adaptive model-based hybrid control of geometrically constrained robot arms," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 1, pp. 105–116, 1997.

[10] C. C. Cheah, C. Liu, and J. J. E. Slotine, "Adaptive tracking control for robots with unknown kinematic and dynamic properties," *International Journal of Robotic Research*, vol. 25, no. 3, pp. 283–296, 2006.

[11] C. C. Cheah, S. Kawamura, and S. Arimoto, "Feedback control for robotic manipulator with an uncertain jacobian matrix," *Journal of Robotic Systems*, vol. 16, no. 2, pp. 119–134, 1999.

[12] C. C. Cheah, M. Hirano, S. Kawamura, and S. Arimoto, "Approximate jacobian control for robots with uncertain kinematics and dynamics," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, pp. 692–702, 2003.

[13] Y. Zhao and C. C. Cheah, "Hybrid vision-force control for robot with uncertainties," in *Proc. of IEEE International Conference on Robotics and Automation*, pp. 261–266, 2004.

[14] Z. Doulgeri and S. Arimoto, "A position/force control for a robot finger with soft tip and uncertain kinematics," *Journal of Robotics Systems*, vol. 19, no. 3, pp. 115–131, 2002.

[15] C. C. Cheah, S. Kawamura, and S. Arimoto, "Stability of hybrid position and force control for robotic manipulator with kinematics and dynamics uncertainties," *Automatica*, vol. 39, pp. 847–855, 2003.

[16] C. C. Cheah, Y. Zhao, and J. J. E. Slotine, "Adaptive jacobian motion and force tracking control for constrained robots with uncertainties," in *Proc. of IEEE Conference on Robotics and Automation*, pp. 2226–2231, 2006.

[17] Y. Zhao, C. C. Cheah, and J. J. E. Slotine, "Adaptive vision and force tracking control for constrained robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1484–1489, 2006.

[18] T. Yoshikawa and A. Sudou, "Dynamic hybrid position/force control of robot manipulators: online estimation of unknown constraint," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 2, pp. 220–226, 1993.

[19] Y. Nakamura, *Advanced Robotics-Redundancy and Optimization*. Addison-Wesley Publishing Company, 1991.

[20] S. Arimoto, *Control Theory of Nonlinear Mechanical Systems- A Passivity-Based and Circuit-Theoretic Approach*. Clarendon Press, 1996.

[21] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, 1996.

[22] L. E. Weiss, A. C. Sanderson, and C. P. Neuman, "Dynamic sensor-based control of robots with visual feedback," *IEEE Journal on Robotics and Automation*, vol. 3, no. 5, pp. 404–417, 1987.

[23] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, 1992.

[24] N. P. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Visual tracking of a moving target by a camera mounted on a robot: A combination of vision and control," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 1, pp. 14–35, 1993.

[25] N. P. Papanikolopoulos and P. K. Khosla, "Adaptive robot visual tracking: Theory and experiments," *IEEE Transactions on Automatic Control*, vol. 38, no. 3, pp. 429–445, 1993.

[26] B. Siciliano and L. Villani, *Robot Force Control*. Kluwer Academic Publishers, 1999.

[27] S. Haykin, *Neural Networks- A Comprehensive Foundation*. Prentice-Hall International, 1999.

[28] R. M. Sanner and J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 837–863, 1992.

[29] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*. World Scientific Publishing, 1998.

[30] F. L. Lewis, "Neural network control of robot manipulators," *Intelligent Systems and Their Applications*, vol. 11, no. 3, pp. 64–75, 1996.

[31] G. Niemeyer and J. J. E. Slotine, "Performance in adaptive manipulator control," *International Journal Robotics Research*, vol. 10, no. 2, pp. 149–161, 1991.

[32] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*. Prentice-Hall International, 1989.