IMAGE BASED CONTROLLER FOR VISUAL SERVOING SYSTEMS

BY

ADRIAN BURLACU and CORNELIU LAZĂR

Abstract. In this paper an image controller using model predictive control (MPC) for visual servoing systems is presented and its performances are analyzed. An eye-in-hand type servoing architecture, composed from a 6 degree of freedom robot and a camera mounted on the gripper is considered. Modeling the visual servo open loop is composed from two stages: first the designing of a model for dynamics of the velocity controlled robot and second a model of the visual sensor. In classical approach, an image interaction matrix maps image space errors into errors in Cartesian space. For the MPC approach, using the open loop visual servo model, an image based predictor is developed. The image based predictor generates the future trajectories of a visual feature ensemble when past and future camera velocities are known. In order to obtain better performances for visual servoing systems an advanced technique is required, for this matter using the developed predictor an image based predictive controller (IbPC) is designed. Implementation, tests and validation of IbPC are conducted and performances in an image based visual servoing scheme are revealed in comparison with a PI image based controller. Experimental results underline the better performances of IbPC as an evaluation regarding the performances of PI classical approach.

Key words: servoing systems, image based predictor, predictive controller.

2000 Mathematics Subject Classification: 53B25, 53C15.

1. Introduction

Visual servoing has been studied in various forms for more than three decades starting from simple pick–and–place tasks to today’s real-time, advanced manipulation of objects. In terms of manipulation, one of the main motivations for incorporating vision in the control loop was the demand for increased flexibility of robotic systems. Image based visual servo control uses image features derived from image plane and servo controls them to a desired
goal configuration [1]. Even if new visual features were proposed [2], [3], point features still remain the most used image measurements in designing visual servo control laws. Usually they are coordinates of the centroids or corners that describe an object. The main aim of the image based visual servoing is to control the end-effector of a robot arm such as a set of image features to reach a desired configuration [4].

Considering the complexity of the visual servoing task, the image based controller is designed using advanced control techniques, which implies having a suitable plant model of the visual servo control system. Thus in [5] is presented an approach using dynamic matrix controller for improvements on visual tracking. A multivariable ARIMAX type which allows a GPC controller implementation for high speed visual servoing of a robot manipulator is described in [6]. Another model of a visual servo system is given in [7] and it employs camera position controller with a robust disturbance observer in the joint space. In this way each joint axis is decoupled and the inner loop can be expressed in the frequency region below the cut-off frequency of the robust disturbance observers as a diagonal transfer matrix. Based on the linear diagonal plant model, a multivariable controller was used to control the visual servo system. For a fast visual servoing of a two links arm that includes flexibilities for compensation of heart movement in surgery, a model which is a linear approximation around the working point was considered [8]. The proposed model takes into account not only the dynamics of the velocity controlled actuators but also the flexibilities in the links, both being experimentally identified. This model was used for the design of two kinds of advanced controllers: one in temporal domain (GPC) and another in the frequency domain ($H_\infty$ controller). An IBVS structure based on nonlinear predictive control is proposed [9] considering the direct dynamic model of the robot, its joints and torques limits, the camera projection model and the visibility constrains. Also, a visual servoing architecture for a six degrees of freedom PUMA robot is presented in [10]. For modeling the plant an ARMAX description is identified and performances evaluations of GPC and MPC, via simulation, are considered. Nonlinear model predictive control (NMPC) plus a linear model, obtained by linearizing and decoupling technique based on the inverse dynamic model, are used for controlling manipulators with catadioptric cameras [11].

In this paper an image based control architecture is developed and a proper plant model for computing future prediction on point feature trajectories is proposed. Using the proposed predictor a multivariable image based predictive controllers was designed for manipulator robots driving. Using cost functions based on errors in image plane, convergence of robot motion has been obtained through nonlinear constraint optimization, which takes into consideration the visibility constrains due to sensor characteristics. Implementation, testing an validation of the predictive control algorithm is realized on a visual servoing simulation architecture. A comparison between a PI image based controller and the predictive image based controller is
conducted and performances are revealed.

2. Image Based Control Architecture

As main objective any image based control architecture Fig.1 generates a video camera trajectory in order to minimize a position error for a visual feature ensemble regarding a desired configuration.

\[
\text{Image based controller} \xrightarrow{e} \text{Power amplif. Robot + Camera}
\]

\[
\text{Feature extraction} \xrightarrow{f} \text{Image}
\]

\[
\text{Object}
\]

\[
\begin{align*}
\text{Feature extraction} & \quad e \\
\text{Image based controller} & \quad v_c^e \\
\text{Power amplif. Robot + Camera} & \quad \text{Object} \\
\end{align*}
\]

Fig. 1—Visual servoing closed loop.

The position error is defined:

\[
e(t) = f(t) - f^*,
\]

where \(f(t)\) represents the visual feature positions at time \(t\) and \(f^*\) is the desired configuration.

Equation (1) is the general representation of the input signal in any image based controller. The outputs of an image based controller are reference velocities of the camera \(v_c^e\). Let \(f(t)\) denote a vector of feature values that can be measured in image, thus being immediately available. Once \(f(t)\) is extracted, the most important stage in a common control scheme is the design of a velocity controller. For this task to be completed it is required a relationship between the time variation of \(f(t)\) and the camera velocity. Let \(v_c = (v_c^e, \omega_c)\) be the camera velocity vector, a \(1 \times 6\) structure composed from the instantaneous linear velocities of the origin of the camera frame \((v_c^e)\) and the instantaneous angular velocities \((\omega_c)\) of the camera frame.

Using

\[
f = f(q, t),
\]
where $q$ is the generalised coordinates vector, results that:

$$f = \frac{\partial f}{\partial q} \dot{q} + \frac{\partial f}{\partial t}.$$

The term $\frac{\partial f}{\partial q}$ can be decomposed like:

$$f = \frac{\partial f}{\partial r} \frac{\partial r}{\partial q} = \frac{\partial f}{\partial q} \frac{\partial r}{\partial q}.$$

where $\frac{\partial r}{\partial q}$ is a way of presentations the robot jacobian.

A relationship between $\dot{f}$ and $v_e$ can be provided via:

$$\dot{f} = L_f v_e,$$

where, by $L_f \in \mathbb{R}^{6 \times 6}$ is denoted the interaction matrix of feature vector $f$. Combining Eq.(1) and Eq.(5) the connection between camera velocity and time variation of error, can be showed:

$$\dot{e} = L_f v_e,$$

If an exponential decoupled decrease of the error is desired it can be obtained:

$$v_e = -\lambda L_f^T e,$$

where $L_f^T \in \mathbb{R}^{6 \times 6}$ is the Moore-Penrose pseudoinverse of $L_f$, that is $L_f^T = (L_f^T L_f)^{-1} L_f^T$. In practice it is impossible to know perfectly $L_f$ or $L_f^T$. So an approximation or an estimation of one of these two matrices must be realised. Considering $L_f^T$ the approximation of the pseudoinverse matrix, the control law (7) is in fact:

$$v_e = -\lambda L_f^T e.$$

The interaction matrix for a visual feature $f = (u,v)$ is computed using:

$$L_f = \begin{bmatrix}
-\frac{1}{Z} & 0 & \frac{uv}{\lambda} & -\frac{\lambda + v^2}{\lambda} & -u \\
-\frac{1}{Z} & \frac{v}{\lambda} & \frac{uv}{\lambda} & -\frac{\lambda + v^2}{\lambda} & -u \\
0 & -\frac{1}{Z} & \frac{v}{\lambda} & -\frac{uv}{\lambda} & -u
\end{bmatrix}.$$
3. Plant Model

The visual servoing system from Fig. 2, composed of a 6 d.o.f robot, camera and object, is considered. The open loop model of the visual servoing system from Fig. 2 consists in two models: Virtual Cartesian Motion Device (VCMD) and visual sensor. The first one has as the input the camera velocity references $v_c^*$ and as output camera velocity screw $v_c$. The second one has the input $v_c$ and the output is the vector $f$ of the coordinates of the features in the image. In the sequel, the VCMD and camera models will be deduced.

![Fig. 2 – Visual servoing control system.](image_url)

3.1. Virtual Cartesian Motion Device

The VCMD model is nonlinear because the robot Jacobian and the model of the robot dynamics are functions of the robot joints angular positions $x_q$. The Jacobian matrix depends on the position of the object with respect to the robot and on the position $x_q$ of the robot. The working configuration of the robot is usually kept far from the singularities of $J_r$. Taking this assumption into account, $J_r$ changes slowly with $x_q$ and thus, it can be considered that during the sampling time, the robot moves only a little. So, $J_r$ is constant between two sampling instants, assumption true in practice when the robot is not close to a kinematic singularity. The model of the robot dynamics is also nonlinear due to the effects such as Coulomb’s friction, Coriolis, centrifugal and gravitational torques. When the robot moves at high speed, the effect of these nonlinearities is significant.

In visual servoing control systems, the joint velocities are controlled by low level servo loops. The controllers of these loops are thus designed in order to eliminate the nonlinear effects considered like load perturbations. It remains only the nonlinear inertia matrix. The joints inertia is varying slowly with the position $x_q$ of the robot and thus, the inertia matrix can be considered constant around a given position of the robot. Consequently, it is possible to linearize the
VCMD model in an area around the position $\mathbf{x}_q$ of the robot where the model of the robot dynamics and $J_r$ can be seen as constant.

If the velocity controllers from the VCMD are designed in order to control the camera velocity $v_c$, the velocity screw can be controlled in the camera coordinates to obtain linear diagonal plant for an 6 DOF system. In this case, it is possible to assume that each joint axis is decoupled under the cut-off frequency of the inner loop.

The typical sampling period of the VCMD system is 0.2-1 [ms] and the typical cut-off frequency is 150-300 [rad/s] [7]. Since the velocity controller is usually a proportional one with an amplification value $k_v$, the inner loop system can be expressed in the frequency region below the cut-off frequency as:

$$v_c(s) = H_0(s)v_c^*(s) = \frac{k_v}{s + k_v}I_0v_c^*(s).$$

Taking into account the model (10), the linearized model of the VCMD from Fig. 3 is resulting. Using the block scheme of the VCMD from Fig. 3, the following linearized discrete model is obtained

$$G(z) = (1 - z^{-1})Z(H_0(s)/s).$$

where $Z$ symbolizes $z$ transform.

The model (11) is an excellent approximation in the design of the IbC controller since the cut-off frequency of the inner loop is much higher than the frequency of the outer loop.

### 3.2. Visual Sensor

Typically, the visual sensor is composed of a camera and an image processing block used to extract the features from the image. Considering the positions and orientations of the camera and object be $\mathbf{x}_c \in \mathbb{R}^6$ and $\mathbf{x}_o \in \mathbb{R}^6$, the camera is modeled by the mapping:

$$i: \mathbb{R}^6 \times \mathbb{R}^6 \rightarrow \mathbb{R}^{2m}.$$

where $m$ feature points are selected to characterize the object.

The camera is modeled using the mapping (12) as a function of $\mathbf{x}_c$ and $\mathbf{x}_o$.
x_0, the positions and the orientations of the camera and the object, respectively. In the sequel, it is considered that m feature points were selected to characterize the object, being defined as:

\[(13) \quad f = [f_1^T, \ldots, f_m^T]^T,\]

where:

\[(14) \quad f_i = [u_i, v_i]^T,\]

are the i-th feature coordinates in the image plane. The image processing block is modeled as:

\[(15) \quad f = g\left(i(x_i, y_i)\right),\]

where g is a function that models the feature extracting algorithms.

In order to get the visual sensor model, there are firstly considered the frames attached to the robot base R_b, to the camera R_c and to the object R_o, as shown in Fig. 4. Let \(T^b_c\) and \(T^b_o\) be the homogeneous transformations between the frames R_c and R_o and, respectively, R_o and R_b. It is assumed that the object feature positions \(x_o^b\) related to the frame R_b is known and that the desired features \(f^*\) were extracted using a suitable operator. The homogeneous transformation \(T^b_c(0)\) for the camera start position is also considered known.

![Fig. 4 - Camera, object and robot base frames.](image)

For getting the homogeneous transformation \(T^b_c(k)\), the Frame motion (FM) block from [7] is used and given in Fig. 5.

![Fig. 5 - Frame Motion block.](image)
The camera velocity screw \( v_c = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T \) is processed by an operator \( O \), which gives the following matrix:

\[
O(v_c) = \begin{bmatrix}
0 & -\omega_z & \omega_y & v_z \\
\omega_z & 0 & -\omega_x & v_x \\
-\omega_y & \omega_x & 0 & v_y \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

After a multiplication and an integral action (block \( I \) in Fig. 5), the result is added with \( T^b_c(0) \) and, thus, \( T^b_c(k) \) is obtained. Computing the inverse of \( T^b_c(k) \), the homogeneous transformation \( T^b_c(k) \) is derived and multiplying it with the object feature position \( x^b_f \) related to the frame \( R_x \), the object feature position \( x^b_c \) related to camera frame \( R_c \) is got. Having the feature position \( x^b_c \), the intrinsic camera parameters \( p_x, p_y \) and the image centre point coordinates \( (u_i, v_i) \), the point features coordinates expressed in pixels are obtained based on the perspective projection:

\[
u_i = \frac{x_i}{z_i} p_i + u_0; \quad v_i = \frac{y_i}{z_i} p_y + v_0.
\]

Thus, the visual sensor model was found which has as input the camera velocity screw \( v_c(k) \) and as output the point feature coordinates \( f(k) \). This model together with VCMD model form the visual servo open-loop model depicted in Fig. 6. The perspective projection is computed using the Image block.

4. Visual Predictive Controller

In order to ensure desired performances of a servoing system it is necessary to use advanced techniques of image based control. The main problem in designing a predictive control strategy is to have a good open loop model attached to the eye-in-hand configuration and also a suitable predictor for point features motion estimation in image plane having reference velocities of the video camera.
The model based predictive control approach for IBVS is developed to visual servoing of a 6 DOF robot manipulator with an eye-in-hand configuration for grasping a fixed object. The object has a known geometry and is described using feature points. The main problem in designing a predictive control strategy is to develop a suitable predictor. In this paper a new image based predictor is created which allows computing the future evolution of the image features over the prediction horizon $H_p$. The plant model for the IbC controller is presented in Fig. 7 and consists of the discrete model of the VCMD and the visual sensor (VS) model obtained from Fig. 6.

![Plant model for prediction computing.](image)

It is assumed that at every sampling period, the visual sensor, composed of a camera and an image feature extraction block, gives the coordinates of the point features $(u_i(k), v_j(k)), i = 1, 4$ and it is possible to compute the depth $z_i(k)$ of the considered points with respect to camera frame. Using the discrete form of (5) and taking into account the discrete model of the VCMD, it results:

$$f(k + 1) = f(k) + TL_G(z^{-1})v_i^*(k),$$

which represents the one-step ahead prediction of the image feature evolution. Shifting the one-step ahead prediction model (18) by recursion, the next predictors are obtained:

$$f(k + 2) = f(k + 1) + TL_G(z^{-1})v_i^*(k + 1)$$

$$f(k + i) = f(k + i - 1) + TL_G(z^{-1})v_i^*(k + i - 1)$$

$$f(k + H_p) = f(k + H_p - 1) + TL_G(z^{-1})v_i^*(k + H_p - 1)$$

The general predictor $f(k + i)$ is found based on the features predicted in the previous step $f(k + i - 1)$, also used to compute the image Jacobian $L_{i+1}$, and on the future command $v_i^*(k + i - 1)$. The prediction is initiated with the features $f(k)$ at time $k$ that are obtained from the visual sensor VS using an
appropriate feature detection operator.

The feature image control errors expressed in Cartesian space are defined by

\[ e_i(k+i) = L_i^* \left( f^* - f(k+i) \right), \]

where \( i = 1, \ldots, H_p \).

The cost function to be minimized is defined as a quadratic form of image control errors expressed in Cartesian space and vector of input feature control over the control horizon \( H_c \):

\[ J = \frac{1}{2} \sum_{i=1}^{H_c} q_{e_i}^2(k+i)Q_{e_i}(k+i) + \frac{1}{2} \sum_{i=0}^{H_c} v_{e_i}^2(k+i)R_{e_i}(k+i), \]

where \( Q \) and \( R \) denote positive definite, symmetric weighting matrices.

The limits of the image introduce visibility constraint which ensures that all the features are always visible. This constrain defined as:

\[ (u_i(k), v_i(k)) \in [u_{\min}, v_{\min}, u_{\max}, v_{\max}], \]

is added to the cost function in order to guarantee for the computing physically valid solution of the image based visual servo predictive control strategy.

5. Experimental Results

Through simulation, two aspects have been evaluated:
1) validation of the proposed predictor;
2) performance comparison between a classical PI image based controller and a predictive image based controller.

Implementation, testing and validation of the proposed predictor were conducted using developed Matlab routines. Each routine has a different objective like computing the interaction matrix, perspective projection, or finding the homogenous matrix transform attached to the camera frame. In order to establish the orientation of the camera frame after a velocity type signal was applied, the roll \( \phi \), pitch \( \theta \), yaw \( \psi \) representation was used. This representation generates a rotation matrix:

\[
R^b = \begin{bmatrix}
    c_\varphi c_\theta & c_\varphi s_\theta s_\psi - s_\varphi c_\psi & c_\varphi s_\theta c_\psi + s_\varphi s_\psi \\
    s_\varphi c_\theta & s_\varphi s_\theta s_\psi + c_\varphi c_\psi & s_\varphi s_\theta c_\psi - c_\varphi s_\psi \\
    -s_\theta & c_\theta c_\psi & c_\theta s_\psi
\end{bmatrix},
\]

where \( c_{\text{angle}}, s_{\text{angle}} \) are denoted the cosine and sinus functions.

Equation (18) was extended and the implementation form was the one depicted in:
Point feature trajectory estimation in image plane, for a servoing system with a VCMD model attached, was analyzed for each type of motion. The predictors efficiency is underlined also by combining basic transformation, and thus deriving a complex camera velocity signal. In Fig. 8 a is presented the structure of a camera velocity signal that generates point features trajectories depicted in Fig. 8 b.

The included robot dynamics for the VCMD model influences the motion trajectories of point features, influence that is showed in the predictor’s response. In conclusion the proposed predictor can be used for motion estimation of point features when images are acquired using a camera mounted on the end-effector of a robot manipulator.

For establishing the performances of the proposed image based architecture that contains a predictive controller a simulation Matlab algorithm was developed. The simulator considers as input data the starting positions of point features and also the desired configuration of point features, all expressed in Cartesian space. The perspective projection was conducted for an image with $512 \times 512$ resolution. Each k-iteration of the algorithm is designed to minimize the cost function from (21) and the optimal velocity is saved. Taking into account the prediction horizon $H_p$ and the command horizon $H_c$, the complexity of the cost function varies.

A proportional integral (PI) IBVS controller and a model based predictive (MBP) IBVS controller have been compared. The PI gains have been tuned to reach the best trade-off between speed and stability. Parameters $k_p$ and
were assigned using the pole placement method based on cancelling the plant constant time, \( k_p = 0.1 \), \( k_i = 0.01 \). For the MBP controller the predictive horizon \( H_p \) was set on 9 and the command horizon \( H_c \) on 1. The lens focal length of the camera is 10 mm and optical center is located at \((u_0, v_0) = (256, 256)\). The PI IBVS controller was included in an existent servoing system simulator developed by Enric Cervera [12].

![Fig. 9](image)

Fig. 9 – PI IBVS controller (a) point feature trajectory (b) image plane error (c) camera velocity (d) camera trajectory.

In this paper only results for rotational transform applied to the camera will be comment, performance for all the other basic transforms (translation on \( x \) and \( y \) axis and scale change) are similar. The experiments were performed under Matlab 7 on a PC Pentium dual core 2.1 Ghz. Four point features were chosen as components of the \( \mathbf{f} \) vector, thus resulting a \( 8 \times 6 \) dimensions matrix interaction. For the rotational transform the \( \mathbf{f}^{'} \) represents positions of chosen point features rotated with \( \pi \) [rad] with respect to \( z \) axis. The results for the PI image based controller are presented in Fig. 9.

In Fig. 10 are presented the results obtained using an MBP IBVS controller with the developed simulator Due to symmetric response obtained for each feature the image plane error is represented just for one feature.
As observed, each controller ensures the desired feature convergence but MBPC has a much faster response time. The desired configuration is reached for the PI controller in 142 sec, while the MBPC enters in steady state after 8 steps, each step is evaluated in 11.2 [sec] thus having a total 89.6 [sec].

6. Conclusions

In this paper a dynamic simulation model for visual servoing of a robot manipulator using image based strategy was presented. The modeling approach proposed in the paper permits to obtain a multivariable model for image based visual servo control open loop systems. It was assumed that the robot joints velocities are controlled by internal feedback loops yielding a linearized and decoupled model of the joints. Using the VCMD and visual sensor models, the visual servo open loop model was derived. Finally, the developed dynamic simulation model has been tested, including open-loop validation, closed-loop validation. Comparisons between simulation results of a predictive controller
and a PI image based controller architecture are presented, the resulted
performances are better for the predictive approach.

Received: December 18, 2007

“Gheorghe Asachi” Technical University of Iaşi,
Department of Automatic Control and
Applied Informatics
e-mails: aburlacu@ac.tuiasi.ro
cizar@ac.tuiasi.ro

REFERENCES

   Control of Motion: Applications in Active Tracking of Moving Targets*. Proc. of
   the 15th Int. Conf. on Pattern Recognition, Barcelona, 2000, Vol. 4, 732-735.
   Using Multivariable Predictive Control*. Advances Robotics, Vol. 21, 10, 2003,
   993-1021.
7. Fujimoto H., *Visual Servoing of 6 Dof Manipulator by Multirate Control with Depth
   Identification*, Proc. of 42nd IEEE Conference on Decision and Control, Maui,
   Hawaii, December 2003, 5408-5413.
   Fast Visual Servoing of a Medical Manipulator Including Flexibilities*. Proc. IEEE
   Int. Conf. on Robotics and Automation, Barcelona, 2005, 4044-4049.
   Through Nonlinear Model Predictive Control*. Proc. of the 4th IEEE
   Conference on Decision and Control, San Diego, December 2006, 1776-1781.
10. Ferreira F., Pinto C., *Visual Based Predictive Control for a Six Degrees of Freedom
    Robot*. IEEE Conf. on Emerging Technologies and Factory Automation, Praga,
11. Allibert C., Courtial E., Toure Y., *Visual Predictive Control for Manipulators with
    Catadioptric Camera*, IEEE Int. Conf. on Robotics and Automation, Pasadena,
    http://sourceforge.net/projects/vstoolbox/.
REGULATOR BAZAT PE IMAGINI PENTRU SISTEME SERVOING VIZUALE

(Rezumat)

Se descrie o metodă de proiectare a unui regulator bazat pe imagini folosind tehnici de control predictiv. Regulatorul bazat pe imagini este evaluat din punct de vedere al performanțelor obținute pentru controlul sistemelor servoing vizuale. Sistemul servoing vizual considerat reprezintă o arhitectură tip „eye-in-hand” compusă dintr-un robot manipulator cu șase grade de libertate și o cameră video montată pe efector. Pentru realizarea modelului în buclă deschisă atașat sunt necesare două etape: prima de modelare dinamică a unui robot controlat prin semnale tip viteză, iar a doua de modelare a senzorului vizual. Abordarea prin prisma tehniciilor de control predictiv presupune dezvoltarea unui predictor bazat pe imagini proiectat folosind proprietățile buclei deschise atașate sistemului servoing vizual. Predictorul generează trajectoare viitoare ale mulțimii trăsăturilor vizuale când sunt cunoscute informații despre viteza camerii. Pentru a obține performanțe îmbunătățite a sistemului servoing vizual sunt necesare tehnici avansate de control, în acest sens predictorul dezvoltat a fost utilizat pentru proiectarea regulatorului predictiv bazat pe imagini. Implementarea, testarea și validarea regulatorului predictiv au fost realizate utilizând un simulator dezvoltat în Matlab iar performanțele sunt evaluate prin comparare cu performanțele unui regulator PI bazat pe imagini. Rezultatele experimentale relevă performanțele mult mai bune ale regulatorului predictiv bazat pe imagini în raport cu regulatorul PI bazat pe imagini.