

VISUAL SERVOING SYSTEM OF A PAN-TILT CAMERA FOR MOVING OBJECTS TRACKING

Davi Yoshinobu Kikuchi, davi.kikuchi@poli.usp.br

Lucas Antonio Moscato, lamoscat@usp.br

University of São Paulo – Av. Prof. Mello Moraes, 2231, CEP: 05508-900, São Paulo, SP, Brazil

Abstract. Video applications whose targets are not static need a constant camera positioning to follow their movements. A pan-tilt camera can move around two rotational axes, and proper angular displacements can make it pointed to any point in space. This paper presents a control system for a pan-tilt camera, which rotates the camera as the target moves to keep the tracked object projection in camera image center. This technique makes use of a visual servoing approach, in which the control system uses visual informations obtained from camera images. These data provide the system sensor feedback and produce a closed-loop and real-time control system. The system presents a dynamic look-and-move approach with image-based control and uses a perspective projection camera model. The target movement model is based on optical flow principle and this work proposes two controllers for system stabilization: a proportional-integral (PI) and a proportional with external disturbances estimation by Kalman filter (LQG). The performance of both algorithms are compared through experimental results.

Keywords: visual servoing, image-based control, linear quadratic control

1. INTRODUCTION

Video applications like videoconferencing and monitoring need constant observation of a proper target by a camera. In the first case, for example, a person should be maintained in camera image center even if he or she moves. Instead of a human camera holder, a pan-tilt camera can be used to follow the target displacements. However, the pan-tilt device demands a proper control system to accomplish the desired camera movements.

This paper presents an algorithm to command a pan-tilt camera, which must keep an any moving target projection (not previously known) in image center. In order to achieve this request, this technique consists of a visual servoing system. In visual servoing (Corke, 1994), the camera is a position sensor that provides system real-time measures, allowing a feedback in a closed-loop control system.

Espiau, Chaumette and Rives (1992) divide the visual servoing approach in two different sub-tasks: visual information extration and control algorithm implementation. In this work (pan-tilt camera control), the first problem consists in determining the target position in each image acquired by the camera. This process is called visual tracking. In the second task, a controller must receive the target position and use it to calculate appropriate pan and tilt displacements. The resultant action must keep the target in image center.

This work utilizes the visual tracking algorithm given by Kikuchi and Moscato (2005), with a few adjustments. The technique uses region template tracking based on sum of squared differences (SSD) criterion. That work presents first a base algorithm and then two improvements over it: incremental estimation and multiresolution estimation.

With the visual tracking procedure already available, this paper presents a visual servoing strategy to fulfill the closed-loop control system. According to Sanderson and Weiss (apud Corke, 1994) classification, this work presents a system with a dynamic image-based look-and-move structure. Thus, a controller receives obtained positions in image coordinates and provides calculated position values to the system (pan and tilt displacements). In many works that use pan-tilt cameras, the control scheme do not receive so much treatment as the tracking algorithm. However, this paper utilizes a visual servoing approach aiming a more solid performance.

Different visual servoing techniques can be found in previous works. There are comparatively simple methods, like a kinematic model for position prediction (Xu and Sugimoto, 1998), and more complex procedures, like the task function approach given by Espiau, Chaumette and Rives (1992). As the pan-tilt camera control problem is not much complex, the algorithm utilized in this work is based on a simple and linear model, given by Papanikolopoulos, Khosla and Kanade (1993). For this model control, two controllers are utilized: proportional-integral and proportional with disturbances estimation. The adopted criterion used to determine the controllers is the linear quadratic minimization. Therefore, as the second controller makes use of a Kalman filter to obtain disturbance estimatives, it can be denominated linear quadratic gaussian (LQG). The two control strategies are implemented in the system and their performances are comparatively analyzed through experimental results.

2. SYSTEM MODEL

This work utilizes the system model given by Papanikolopoulos, Khosla and Kanade (1993). According this approach, the optical flow (Tsakiris, 1998) of a point projection (target centroid) in an image has two sources: target motion and

camera movements. Assume a coordinate system with origin in image center, horizontal x_1 -axis pointing right and vertical x_2 -axis pointing up. The state-space system model is:

$$\mathbf{x}(t_{i+1}) = \mathbf{A}\mathbf{x}(t_i) + \mathbf{B}\mathbf{u}(t_i) + \mathbf{E}\mathbf{d}(t_i) + \mathbf{G}\mathbf{w}(t_i) \quad (1)$$

where:

t_i = generic time instant (s)

$\mathbf{x}(t_i) = \begin{bmatrix} x_1(t_i) \\ x_2(t_i) \end{bmatrix}$ = target position (pixels)

$\mathbf{u}(t_i) = \begin{bmatrix} u_1(t_i) \\ u_2(t_i) \end{bmatrix}$ = optical flow component induced by camera movements (pixels/s)

$\mathbf{d}(t_i) = \begin{bmatrix} d_1(t_i) \\ d_2(t_i) \end{bmatrix}$ = optical flow component induced by target motion (pixels/s)

$\mathbf{w}(t_i) = \begin{bmatrix} w_1(t_i) \\ w_2(t_i) \end{bmatrix}$ = process noise vector (pixels)

$\mathbf{A} = \mathbf{G} = \mathbf{I}_2$, $\mathbf{B} = \mathbf{E} = T\mathbf{I}_2$

T = sampling time (s)

\mathbf{I}_2 = identity matrix of size 2

Moreover, the output vector (\mathbf{y}) is:

$$\mathbf{y}(t_i) = \mathbf{C}\mathbf{x}(t_i) + \mathbf{v}(t_i) \quad (2)$$

where:

$\mathbf{y}(t_i) = \begin{bmatrix} y_1(t_i) \\ y_2(t_i) \end{bmatrix}$, $\mathbf{C} = \mathbf{I}_2$

$\mathbf{v}(t_i) = \begin{bmatrix} v_1(t_i) \\ v_2(t_i) \end{bmatrix}$ = measurement noise vector (pixels)

The output vector provides system measures, obtained from a visual tracking algorithm. This measures are x_1 and x_2 target positions, which form the state vector. As the control system must keep the target in image center, its goal is drop and hold the state vector value close to zero.

The control system calculates and provides the control vector (\mathbf{u}) to the system. However, the pan-tilt camera inputs are rotation values, not linear velocities. Therefore, a conversion is necessary, and an image jacobian (Hutchinson, Hager and Corke, 1996) can provide it. Using perspective projection and optical flow principle (Tsakiris, 1998), the following relation can be determined:

$$\begin{bmatrix} R_X \\ R_Y \end{bmatrix} = \begin{bmatrix} \frac{x_1 x_2}{(x_1^2 + x_2^2 + f^2)f} & -\frac{x_1^2 + f^2}{(x_1^2 + x_2^2 + f^2)f} \\ \frac{x_2^2 + f^2}{(x_1^2 + x_2^2 + f^2)f} & -\frac{x_1 x_2}{(x_1^2 + x_2^2 + f^2)f} \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

where:

R_X, R_Y = tilt and pan rotation speeds, respectively (rad/s)

f = focal length of the camera lens (pixels)

3. CONTROL SYSTEM

3.1 Proportional-Integral Controller

The proportional-integral (PI) controller is the first one developed in this paper. A system equipped with this controller can suppress constant disturbance inputs (with unknown values) and drop the state vector value to zero. In this work model, ignoring the noise inputs, steady-state position values can be null if the disturbance input is constant (target moves with constant speed). Also, the PI controller can stabilize the output values at a non-zero reference value, but this property is not necessary in this application.

To determine the PI controller, this work utilizes the procedure given by Maybeck (1979), which makes use of a linear-quadratic optimal criterion. Moreover, this work uses the infinite-horizon approach, where controller gain matrices are constant.

A integral action is provided through an additional state vector \mathbf{q} :

$$\mathbf{q}(t_{i+1}) = \mathbf{q}(t_i) + \mathbf{y}(t_i) \quad (3)$$

From Eq. (1) and Eq. (3), ignoring the noise-related terms (\mathbf{G} , \mathbf{w} and \mathbf{v}) and considering the disturbance vector ($\mathbf{d}(t_i)$) constant, an augmented model is created:

$$\begin{bmatrix} \mathbf{x}(t_{i+1}) \\ \mathbf{q}(t_{i+1}) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{C} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}(t_i) \\ \mathbf{q}(t_i) \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \mathbf{u}(t_i) + \begin{bmatrix} \mathbf{E}\mathbf{d} \\ \mathbf{0} \end{bmatrix}$$

Using perturbation variables approach (Maybeck, 1979) and substituting in the model, it becomes:

$$\begin{bmatrix} \delta\mathbf{x}(t_{i+1}) \\ \delta\mathbf{q}(t_{i+1}) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{C} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}(t_i) \\ \delta\mathbf{q}(t_i) \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \delta\mathbf{u}(t_i) \Rightarrow \mathbf{x}_a(t_{i+1}) = \mathbf{A}_a\mathbf{x}_a(t_i) + \mathbf{B}_a\delta\mathbf{u}(t_i)$$

where:

$\delta\mathbf{x}$, $\delta\mathbf{q}$, $\delta\mathbf{u}$ = perturbation variables vectors

A solution is obtained through the minimization of a cost function J :

$$J = \sum_{i=0}^{\infty} \frac{1}{2} \left(\begin{bmatrix} \mathbf{x}_a(t_i) \\ \delta\mathbf{u}(t_i) \end{bmatrix}^T \begin{bmatrix} \mathbf{X} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{x}_a(t_i) \\ \delta\mathbf{u}(t_i) \end{bmatrix} \right)$$

where:

\mathbf{X} , \mathbf{S} , \mathbf{U} = cost weighting matrices

The solution is given by the following control law:

$$\mathbf{u}_a(t_i) = -\mathbf{K}_c\mathbf{x}_a(t_i) \Rightarrow \delta\mathbf{u}(t_i) = -[\mathbf{K}_{c1} \mid \mathbf{K}_{c2}] \begin{bmatrix} \delta\mathbf{x}(t_i) \\ \delta\mathbf{q}(t_i) \end{bmatrix}$$

where:

$\mathbf{K}_c = -[\mathbf{K}_{c1} \mid \mathbf{K}_{c2}]$ = controller gain matrix, obtained from the solution of a Riccati equation (Maybeck, 1979)

At last, developing the equation (Maybeck, 1979), the result is:

$$\mathbf{u}(t_i) = -\mathbf{K}_{c1}\mathbf{x}(t_i) - \mathbf{K}_{c2}\mathbf{q}(t_i)$$

Figure 1 presents the control system diagram.

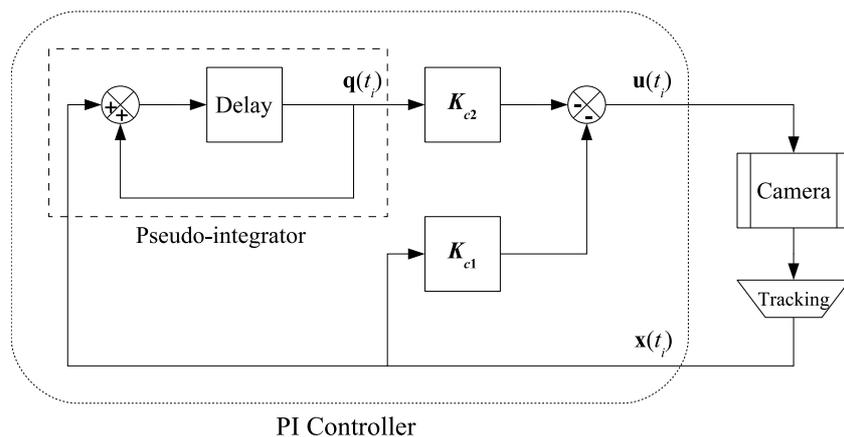


Figure 1. Diagram of the system with a proportional-integral controller.

3.2 Proportional Controller with Disturbances Estimation

This procedure proposes an estimation method of the disturbance input vector, so it can be deduced from the state vector and a simple proportional action can stabilize the system output at zero. The disturbance estimation is provided by a Kalman filter (Fleury, 1981/1982).

Papanikolopoulos, Khosla and Kanade (1993) utilize a random walk as the disturbance model:

$$\mathbf{d}(t_{i+1}) = \mathbf{d}(t_i) + T.\mathbf{w}_d(t_i) \tag{4}$$

where:

\mathbf{w}_d = white noise vector with known statistical parameters

The augmented model using Eq. (1), Eq. (2) and Eq. (4) is:

$$\begin{cases} \begin{bmatrix} \mathbf{x}(t_{i+1}) \\ \mathbf{d}(t_{i+1}) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{E} \\ \mathbf{0} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}(t_i) \\ \mathbf{d}(t_i) \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \mathbf{u}(t_i) + T \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_2 \end{bmatrix} \mathbf{w}_d(t_i) \\ \mathbf{y}(t_i) = [\mathbf{C} \quad \mathbf{0}] \begin{bmatrix} \mathbf{x}(t_i) \\ \mathbf{d}(t_i) \end{bmatrix} + \mathbf{v}(t_i) \end{cases} \Rightarrow \\ \Rightarrow \begin{cases} \mathbf{x}_a(t_{i+1}) = \mathbf{A}_a \mathbf{x}_a(t_i) + \mathbf{B}_a \mathbf{u}(t_i) + \mathbf{G}_a \mathbf{w}_d(t_i) \\ \mathbf{y}(t_i) = \mathbf{C}_a \mathbf{x}_a(t_i) + \mathbf{v}(t_i) \end{cases}$$

The disturbance input vector \mathbf{w} is not shown in the previous equations, because the model assumes that \mathbf{w}_d model include it implicitly.

This work uses a steady-state Kalman filter, so its gain matrix is constant. The state vector estimation ($\hat{\mathbf{x}}_a$) is obtained from equations provided by Fleury (1981/1982):

$$\hat{\mathbf{x}}_a(t_i) = \mathbf{A}_a \hat{\mathbf{x}}_a(t_{i-1}) + \mathbf{B}_a \mathbf{u}(t_{i-1}) + \mathbf{K}_f [\mathbf{y}(t_i) - \mathbf{C}_a \hat{\mathbf{x}}_a(t_i)]$$

where:

$\mathbf{K}_f = \begin{bmatrix} \mathbf{K}_{f1} \\ \mathbf{K}_{f2} \end{bmatrix}$ = Kalman gain matrix, obtained from the steady-state solution of a recursive equation (Fleury, 1981/1982)

Substituting the augmented model components and matrices numerical values:

$$\begin{aligned} \hat{\mathbf{x}}(t_i) &= \hat{\mathbf{x}}(t_{i-1}) + T \cdot \mathbf{u}(t_{i-1}) + T \cdot \hat{\mathbf{d}}(t_{i-1}) + \mathbf{K}_{f1} [\mathbf{y}(t_i) - \hat{\mathbf{x}}(t_i)] \\ \hat{\mathbf{d}}(t_i) &= \hat{\mathbf{d}}(t_{i-1}) + \mathbf{K}_{f2} [\mathbf{y}(t_i) - \hat{\mathbf{x}}(t_i)] \end{aligned}$$

To determine the controller, the augmented model can not be used, because it is uncontrollable. Thus, Papanikolopoulos, Khosla and Kanade (1993) utilize a modified control vector $\mathbf{u}_n(t_i) = \mathbf{u}(t_i) + \mathbf{d}(t_i)$. As the matrices \mathbf{B} and \mathbf{E} are equal, Eq. (1), ignoring the noise input, becomes:

$$\mathbf{x}(t_{i+1}) = \mathbf{A}\mathbf{x}(t_i) + \mathbf{B}\mathbf{u}_n(t_i)$$

Using again a linear-quadratic approach, the cost function to be minimized is:

$$J = \sum_{i=0}^{\infty} \left\{ \frac{1}{2} \begin{bmatrix} \mathbf{x}(t_i) \\ \mathbf{u}_n(t_i) \end{bmatrix}^T \begin{bmatrix} \mathbf{X} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t_i) \\ \mathbf{u}_n(t_i) \end{bmatrix} \right\}$$

The solution is (Maybeck, 1979):

$$\mathbf{u}_n(t_i) = -\mathbf{K}_c \mathbf{x}(t_i) \Rightarrow \mathbf{u}(t_i) = -\mathbf{K}_c \mathbf{x}(t_i) - \mathbf{d}(t_i)$$

Using Kalman filter estimatives:

$$\mathbf{u}(t_i) = -\mathbf{K}_c \hat{\mathbf{x}}(t_i) - \hat{\mathbf{d}}(t_i)$$

The control system diagram is presented in Fig. 2.

4. EQUIPMENT DESCRIPTION AND ALGORITHM SETUP

4.1 Hardware Configuration

A developed software implements the visual tracking and control system techniques. It runs on a computer equipped with a Pentium 3 – 800MHz processor, 304 MB RAM, and Microsoft Windows XP operating system. The software was programmed on Microsoft Visual C++ 6.0 environment.

The computer sends control signals to a pan-tilt device, using serial communication. The pan-tilt device was provided by Escola Politécnica in University of São Paulo. It receives angular pan and tilt positions as input (in degrees). As the values calculated by the controllers are velocities (rad/s), a conversion is necessary:

$$\begin{aligned} pan(t_i) &= pan(t_{i-1}) + R_Y \times T \times \frac{180}{\pi} \\ tilt(t_i) &= tilt(t_{i-1}) + R_X \times T \times \frac{180}{\pi} \end{aligned}$$

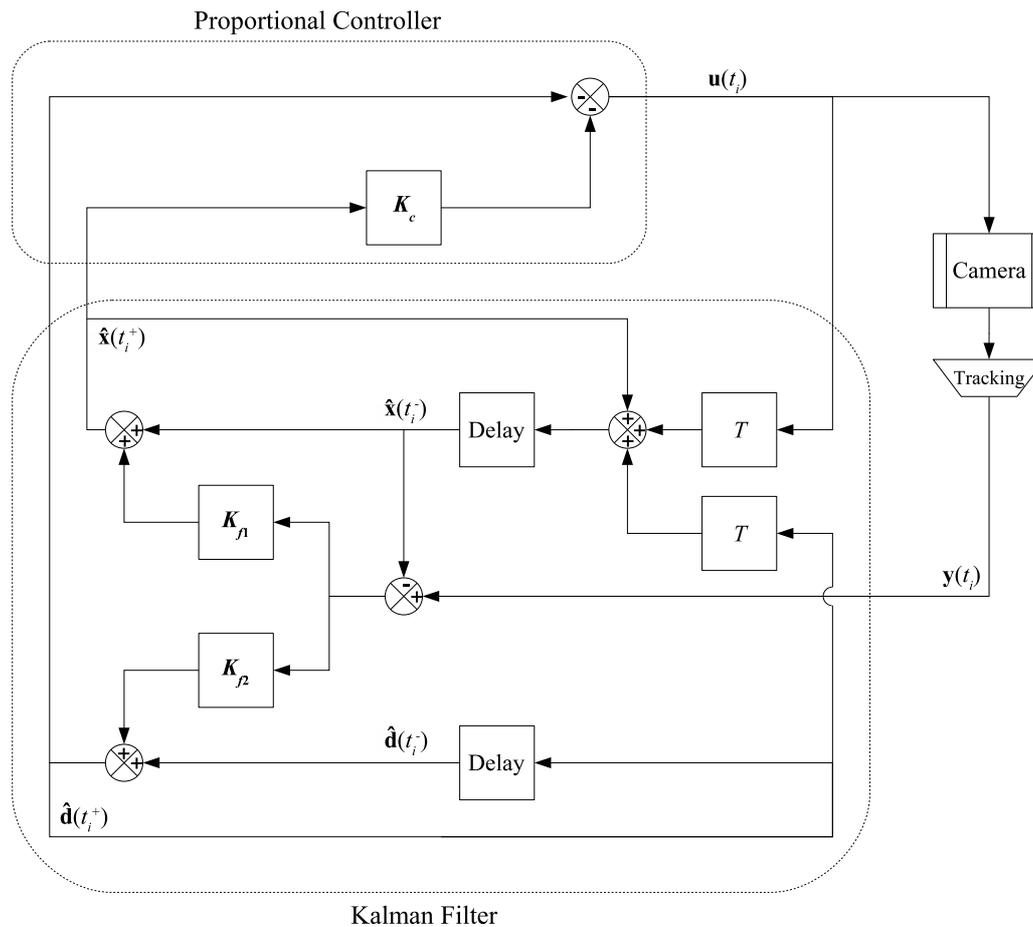


Figure 2. Diagram of the system with a linear quadratic gaussian controller.

where:

$pan(t_i)$ = angular camera pan position (degrees)

$tilt(t_i)$ = angular camera tilt position (degrees)

A camera is connected to the pan-tilt device. It is a regular webcam, fabricated by Shenzhen Akkord Electronics Co., VC2P model. It includes a CMOS 1/4" sensor, communicates with the computer through a USB 1.1 port, and is configured to receive images with 320×240 pixels at 30 fps.

4.2 Visual Tracking Algorithm Considerations

This work utilizes the technique given by Kikuchi and Moscato (2005), with multiresolution estimation extension configuration and some modifications. First, the parametric movement function F has four parameters, including an additional rotation variable. Moreover, all four parameters are iterated, not only the translations. And an additional stop criterion is used: if there is at least one sign change (positive to negative, or negative to positive) in each parameter increment, the algorithm stops the iterations. Yet, the procedure uses only two resolution levels, not three.

The target region utilized in the algorithms is a 50×50 pixels window centralized in image center. The adopted error values for procedure convergence are 0,05 to translations parameters and 0,005 to rotational and scaling variables. The maximum number of iterations is 30.

4.3 Control System Constants

The adopted sampling time of the control algorithms is 500 ms. Moreover, is necessary to define values for the weighting matrices to calculate the controller gains. Additionally, the Kalman filter gains are determined using covariance matrices of the noise vectors (Fleury, 1981/1982). The utilized values are:

- PI Controller

$$\text{Weighting matrix} = \begin{bmatrix} 10 \cdot \mathbf{I}_4 & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_2 \end{bmatrix} \Rightarrow [K_{c1} \mid K_{c2}] = \begin{bmatrix} 2,8040 & 0 & 1,0066 & 0 \\ 0 & 2,8040 & 0 & 1,0066 \end{bmatrix}$$

• LQG Controller

$$\left. \begin{aligned} \text{Covariance matrix of } \mathbf{w}_d &= \begin{bmatrix} 400 & 0 \\ 0 & 400 \end{bmatrix} \\ \text{Covariance matrix of } \mathbf{v} &= \begin{bmatrix} 6,25 & 0 \\ 0 & 6,25 \end{bmatrix} \end{aligned} \right\} \Rightarrow \begin{bmatrix} \mathbf{K}_{f1} \\ \mathbf{K}_{f2} \end{bmatrix} = \begin{bmatrix} 0,8803 & 0 \\ 0 & 0,8803 \\ 1,3841 & 0 \\ 0 & 1,3841 \end{bmatrix}$$

$$\text{Weighting matrix} = \begin{bmatrix} 10 \cdot \mathbf{I}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_2 \end{bmatrix} \Rightarrow \mathbf{K}_c = \begin{bmatrix} 1,5311 & 0 \\ 0 & 1,5311 \end{bmatrix}$$

A last necessary constant is the focal length of the camera lens. This paper utilizes a empirically obtained value, $f = 332,01$ pixels (the nominal value calculated using the camera parameters is 320 pixels).

5. EXPERIMENTAL RESULTS

5.1 System Response to a Single Displacement of the Target

The first test consists of determining the system response to a single horizontal displacement (instantaneous pulse simulation) of the target. The target reference region is given in the first image of Fig. 3 (small centered square). The larger square represents the equivalent reference in minor resolution image. The second image of Fig. 3 represents the target after the displacement (about 14 pixels to left).

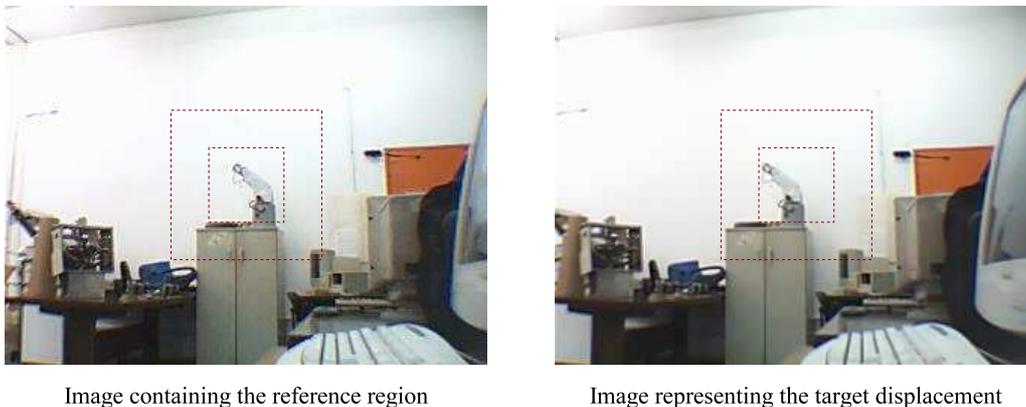


Figure 3. Images representing the first test condition.

Initially, the test procedure consists of positioning the pan-tilt camera so the target reference region is correctly settled. Thus, the target suffers the disturbance and the control system responds. The operation is repeated again and again for both controllers, PI and LQG. Typical system outputs can be found in Fig. 4 (graphs with x_1 and x_2 position values in pixels versus elapsed time).

Both controllers were able to conduct the target to the image center. After the disturbance detection, the LQG response dropped the position value to zero faster than the PI output did. Besides, at the instant right after the displacement occurrence, the PI curve presented a peak with magnitude similar to the disturbance. This occurs because PI controllers stabilize systems with a constant disturbance input. In this case, with a instantaneous pulse, the situation is similar to a constant disturbance occurrence at a certain time, and another constant input appearance at the immediately after sampling instant, but with the opposite sign (orientation). Therefore, the PI controller detects two disturbances, and there are two response peaks.

5.2 System Response to a Compounded Target Movement

The previous experiment is based on a single disturbance artificially imposed to the target during the whole considered period. For a more realistic simulation, this test utilizes a target with more displacements and more natural movements. The target consists of a human face, which is often utilized in this type of application (videoconferencing and distance learning, for example).

A real person is responsible for the face movements, so the displacements are more natural than the ones of the previous test. Initially, the face remains motionless for a few seconds. After that, there is a movement to left, then another pause for a couple seconds. Afterwards, the target moves to a position at right of the initial location, stops again for a few seconds, then returns to the initial position and stays there until the experiment ends. As a person moves the target, there also some vertical, rotational and scaling displacements during the test, including the still periods. The procedure is oft repeated for each controller, and two typical results are given in Figure 6 (once more, position values in pixels).

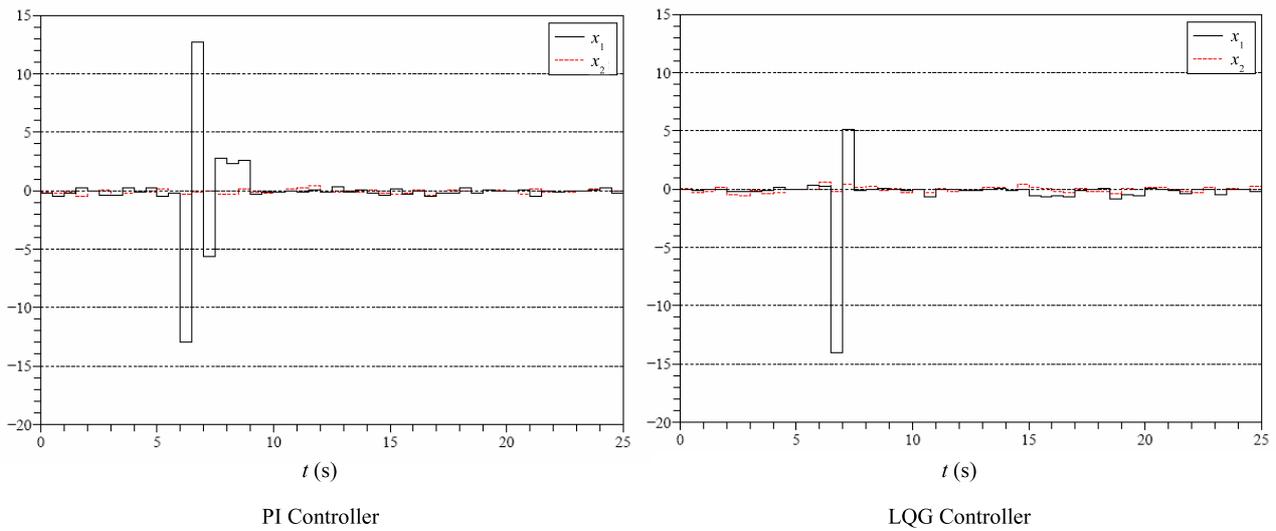


Figure 4. System outputs obtained from a single displacement of the target.



Figure 5. Target image example representing the second test condition.

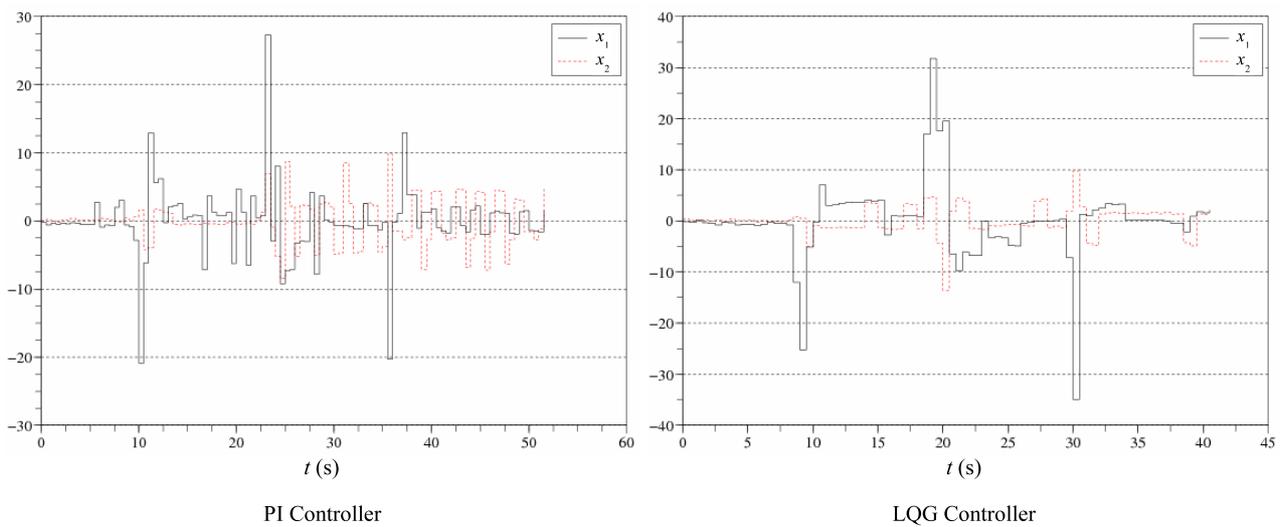


Figure 6. System output obtained from a compounded target movement.

The graph curves showed three moments when there were high magnitude peaks. Obviously, they represent the three target disturbances. Again, the PI response presented peaks with high values and opposite orientations after each single displacement, while the LQG stabilized the output faster. Additionally, the PI curve presented significantly more oscillations than the LQG output.

Another fact provided by the graphs occurred at the instant right after a displacement detection. The position values for PI controller dropped while for LQG leant to rise. The reason for this is again the PI attribute that stabilizes constant external disturbances. Thus, at these instants, the target moves probably with constant approximately velocities. However, this particular situation can not be considered common in this work applications. Therefore, this aspect is not a great advantage of the PI controller.

6. CONCLUSIONS

This paper presented a visual servoing technique applied to a pan-tilt camera. Using the visual tracking algorithm given by Kikuchi and Moscato (2005), this work developed a closed-loop control system to command the camera motion. The utilized system model (Papanikolopoulos, Khosla and Kanade, 1993) is linear and based on optical flow principle.

Two controllers was presented to perform the model control: proportional-integral (PI) and proportional with external disturbances estimation by a Kalman filter (LQG). Both were determined using a linear-quadratic minimization criterion. The LQG controller granted a better performance than the PI in the experiments. It presented outputs with less oscillations and smoother peaks, in particular in situations with more disturbance variety.

For better performances, a hardware improvement would allow shorter sampling times adoption in the control system. Otherwise, as the system model is linear and not complex, other controllers can be designed and easily implemented in the system. Modifications in the visual tracking algorithm (like a illumination model or robust estimators addition) can also be made without severe efforts.

7. REFERENCES

- Corke, P.I., 1994, "Visual Control of Robot Manipulators – A Review", *Visual Servoing: Real-Time Control of Robot Manipulators Based on Visual Sensory Feedback*, (Robotics and Automated Systems, Vol.7), Ed. World Scientific, pp. 1-31.
- Espiau, B., Chaumette, F. and Rives, P., 1992, "A New Approach to Visual Servoing in Robotics", *IEEE Transactions on Robotics and Automation*, Vol.8, No.3, USA, pp. 313-326.
- Fleury, A.T., 1981/1982, "Filtro de Kalman Aplicado à Navegação Aérea", *Lecture Notes on an EMBRAER, INPE Course*.
- Hutchinson, S.A., Hager, G.D. and Corke, P.I., 1996, "A Tutorial on Visual Servo Control", *IEEE Transactions on Robotics and Automation*, Vol.12, No.5, USA, pp. 651-670.
- Kikuchi, D.Y. and Moscato, L.A., 2005, "Visual Tracking System Using Template Matching Algorithm", *COBEM 2005: proceedings*, Rio de Janeiro, Brazil.
- Maybeck, P.S., 1979, "Stochastic Models, Estimation, and Control", *Mathematics in Science and Engineering*, Vol.3, Ed. Academic Press, New York, USA.
- Papanikolopoulos, N., Khosla, P. and Kanade, T., 1991, "Visual Tracking of a Moving Target by a Camera Mounted on a Robot: A Combination of Control and Vision", *IEEE Transactions on Robotics and Automation*, Vol.9, No.1, USA, pp. 14-35.
- Tsakiris, D.P., 1988, "Visual Tracking Strategies", M.S. Thesis, Department of Electrical Engineering, University of Maryland, USA, 94 p.
- Xu, G. and Sugimoto, T., 1998, "Rits Eye: A Software-based System for Realtime Face Detection and Tracking Using Pan-Tilt-Zoom Controllable Camera", *ICPR'98*, Vol.2, Brisbane, Australia, pp. 1194-1197.

8. Responsibility notice

The authors are the only responsible for the printed material included in this paper.