Control Performance Issues in a Binocular Active Vision System

João P. A. Barreto
Inst. Systems and Robotics
Inst. Politecnico de Leiria
Leiria 2400
PORTUGAL

Paulo Peixoto
Inst. Systems and Robotics
Dep. of Electrical Eng.
Coimbra 3030
PORTUGAL

Jorge Batista
Inst. Systems and Robotics
Dep. of Electrical Eng.
Coimbra 3030
PORTUGAL

Helder Araujo
Inst. Systems and Robotics
Dep. of Electrical Eng.
Coimbra 3030
PORTUGAL

Abstract

The performance of a binocular active vision system depends mainly on two aspects: vision/image processing and control. In this paper we characterize the monocular performance of smooth pursuit. This system is used to track binocularly targets in a surveillance environment. One of the aspects of this characterization was the inclusion of the vision processing. To characterize the performance from the control point of view four standard types of inputs were used: step, ramp, parabola and sinusoid. The responses can be used to identify which subsystems can be optimized. We show that prediction and a velocity estimate are essential for a good tracking performance.

1 Introduction

Visual servoing and active vision are important research topics in robotics and computer vision. Many aspects have been studied and several systems demonstrated [1, 2]. One of these aspects is the issue of system dynamics. System dynamics is essential to enable the performance optimization of the system. Other aspects are related to stability and the system latencies [3, 4]. In [4] Corke shows that dynamic modeling and control design are very important for the improved performance of visual closed-loop systems. One of his main conclusions is that a feedforward type of control strategy is necessary to achieve high-performance visual servoing. Nonlinear aspects of system dynamics have also been addressed [5, 6]. In [5] Kelly discusses the nonlinear aspects of system dynamics and proves that the overall closed loop system composed by the full nonlinear robot dynamics and the controller is Lyapunov stable. In [6] Hong models the dynamics of a two-axis camera gimbal and also proves that a model reference adaptive controller is Lyapunov stable. In [7] Rizzi and Koditschek describe a system that takes into account the dynamical model of the target motion. They propose a novel triangulating state estimator and prove the convergence of the estimator. In [8, 9] the control performance of the Yorick head platform is also presented. In special it is considered the problem of dealing with the inherent delays and in particular with variable delays. Problems associated with overcoming system latencies are also discussed in [10, 11]. A performance characterization of an active vision system was also performed in the GRASP laboratory[**?**].

2 The MDOF System–Block Diagram and Control Structure

The MDOF binocular system is a high-performance active vision system with a high number of degrees of freedom. Real-time complex visual behaviors were implemented after careful kinematics modeling and adequate selection of basic visual routines. This platform is now integrated in a visual surveillance system that enables the selection of the target to be pursued by the MDOF head [12]. One of the features of this system is its ability to track targets that can change their shape (e.g. human intruders). This is possible because

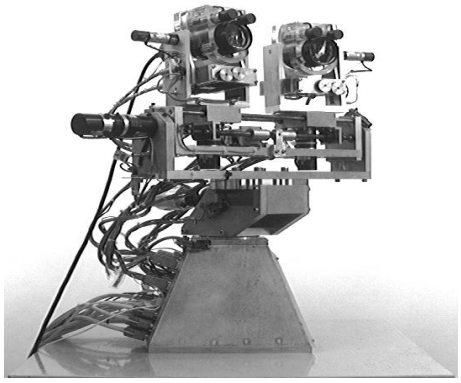

Figure 1: The MDOF binocular system

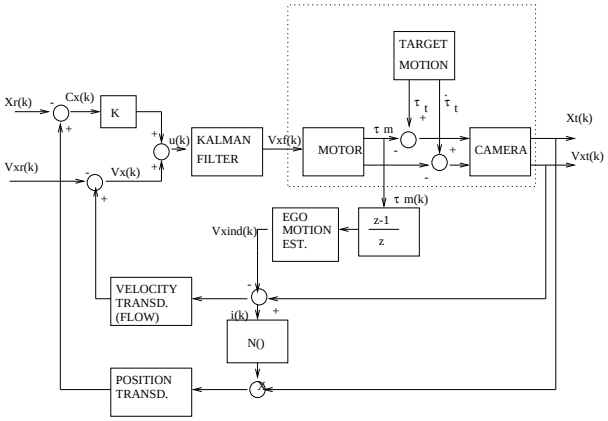

Figure 2: System block diagram

differential flow was chosen for velocity estimation.

The MDOF binocular tracking system can be considered as a servomechanism whose reference inputs are the target coordinates in space and whose outputs are the motor velocities and/or positions [13, 14]. However in the case of this system, and as a result of both its mechanical complexity and its goal (tracking of targets with unknown dynamics), we decided to relate the system outputs with the data measured from the images. Thus this system can be considered a regulator whose goal is to keep the target in a certain position on the image (usually its center). As a result of this framework target motion is dealt with as a perturbation. If the perturbation affects the target position on the image it has to be compensated for. Servoing is therefore a resource used by the system to achieve its objectives.

Our goal is to characterize the control performance of the several visual behaviors of the system namely smooth pursuit, saccadic motion and vergence control. However in this paper we will focus on the analysis of the smooth pursuit performed by one of the cameras. Each camera joint has two independent rotational degrees of freedom: pan and tilt. Even though pure rotation can not be guaranteed we model these degrees of freedom as purely rotational. A schematic for one of the these degrees of freedom is depicted in Fig 2(both degrees of freedom are similar and decoupled). Notice that 2 inputs and 2 outputs are considered. Both position and velocity of the target in the image are to be controlled or regulated . If it is true that the two quantities are closely related, this formal distinction allows a better evaluation of some aspects such as non-linearities and limitations in performance.

The dotted box encloses the analog components of the structure. All the other elements are digital. Digital to analog, as well as analog to digital conversions are omitted. Block $N(i(k))$ represents a non-linear function described in equation 1.

$$\begin{cases} i(k) = V_{xt}(k) - V_{xind}(k) \\ N(i(k)) = 1 \Longleftarrow i(k) > 0 \\ N(i(k)) = 0 \Longleftarrow i(k) = 0 \end{cases} \quad (1)$$

$V_{xf}(k)$ is the command sent to the motor, obtained by filtering $u(k)$, the sum of the estimated velocity with the position error multiplied by a gain K (2).

$$u(k) = V_x(k) + K.C_x(k) \quad (2)$$

Notice that the Kalman filter predicting capability is not being used. Instead the filter is used to achieve smooth motion without oscillations. Considering that the motion computed on the image is caused by target motion and by camera motion, the computation of the target velocity requires that the effects of egomotion are compensated for. The egomotion is estimated based on the encoder readings and on the inverse kinematics. Once egomotion is estimated, target velocity on the image plane is computed based on an affine model of optical flow. Target position is estimated as the average location of the set of points with non-zero optical flow in two consecutive frames (after egomotion having been compensated for). This way what is actually computed is the center of motion instead of target position. The estimated value will be zero whenever the object stops, for it is computed by using function $N(i(k))$.

3 System perturbation

The MDOF tracking system compensates for the perturbations due to target motion. To study and

characterize its regulation/control performance usual control test signals must be applied. Two problems must be considered.

- The accurate generation of perturbation signals;
- The generation of perturbation signals functionally defined, such as steps, ramps, parabolas and sinusoids;

Instead of using real targets, we decided to use synthetic images so that the mathematical functions corresponding to reference trajectories could be accurately generated. These images are then used as inputs in the binocular active vision system.

The captured frame at a given time instant depends, not only on the target position, but also on the camera orientation. Due to the change of the system geometry as a result of its operation, synthetic images have to be generated on line to take into account the specific geometry at each frame time instant. Therefore both target position and camera orientation have to be known in the same inertial coordinate system. The former is calculated using a specific motion equation that enables the computation of any kind of motion in space. Camera orientation is computed by taking into account the motor encoders readings and the inverse kinematics. The inertial coordinate system origin is placed at camera optical center.

$$ {}^{c}\boldsymbol{A_i} = \begin{bmatrix} C(\alpha_p) & S(\alpha_p)S(\alpha_t) & -S(\alpha_p)C(\alpha_t) & 0 \\ 0 & C(\alpha_t) & S(\alpha_t) & 0 \\ S(\alpha_p) & -C(\alpha_p)S(\alpha_t) & C(\alpha_p)C(\alpha_t) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3) $$

Mathematic equations are used to accurately describe the desired target motion in space. Motion coordinates are converted into inertial cartesian coordinates applying suitable transformation equations. Notice that motion coordinates can be of any kind (cartesian, spherical or cylindrical). Inertial coordinates are converted in camera coordinates using matrix 3. This transformation depends on motor positions (α_p, α_t), that are known by reading the encoders. Perspective projection is assumed for image formation. These computations are performed at each frame time instant[**?**].

$$ \theta(t) = Const \quad (4) $$

$$ \theta(t) = \omega.t \quad (5) $$

$$ \theta(t) = \frac{\gamma}{2}.t^2 \quad (6) $$

$$ \theta(t) = A\sin(\omega.t) \quad (7) $$

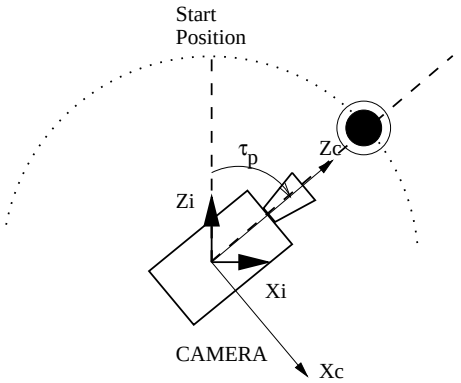

Figure 3: Motor angular position for perfect tracking considering target spherical/circular motion

We decided to relate the system outputs with the data measured from the images. Therefore target motion must generate the standard control test signals on the image plane. For the MDOF system, where cameras move along circular paths, it was demonstrated that target trajectories of equations 4 to 7 (spherical coordinates) generate the desired perturbations if perfect tracking is assumed [**?**]. Thus a step (in position) is an abrupt change of target position in image (equation 4). A ramp/parabola (in position) occurs when the 3D target motion induces a constant velocity/acceleration in the image plane (equations 5 and 6). And a sinusoid is generated whenever the image target position and velocity are described by sinusoidal functions of time (with a phase difference of 90 degrees)(equation 7).

4 System response to motion

In this section we analyze the system ability to compensate for perturbations due to target motion. Pan and tilt control algorithms are equal except for some of the parameters. As a result the behaviors of both (including when they are combined) are identical and due to lack of space we will only present results for the pan control.

4.1 Step response

If the target moves instantaneously to a position 7 degrees ahead on a circumference centred in the projection center of the camera, we will have a step perturbation in position of amplitude 7. Fig. 4 shows the

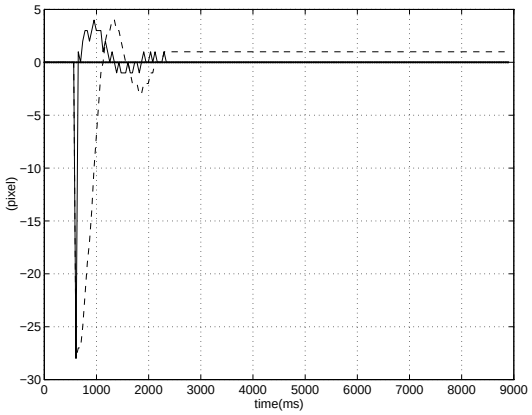

Figure 4: Regulation performance. Target position (- -) and velocity (-) on the image

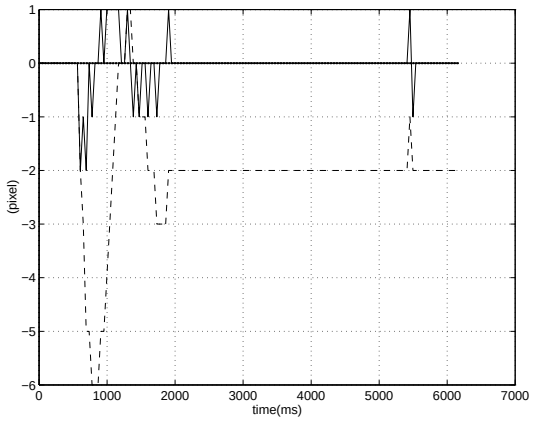

Figure 5: Regulation performance. Target position (- -) and velocity (-) in the image

evolution of target position (X_t) in the image. The system detects the motion and immediately moves the camera on that direction. An overshoot of about 10% occurs and stability is reached. The regulation is done with a steady state error of about 1.5 pixels. In experiments done with smaller amplitude steps the system fully compensates for target motion. In these situations the regulation error is 0 and we have a type 1 system. The type of response depends on the step amplitude. This clearly indicates a non-linear behavior. One of the main reasons for the non-linear behavior is the way position feedback is implemented. After compensating for egomotion, target position is estimated as the average location of the set of points with non-zero optical flow in two consecutive frames. Thus the center of motion is calculated instead of target position. If the target stops, any displacement detected in the image is due camera motion. In that case target velocity ($V_{xt}(k)$) is equal to induced velocity ($V_{xind}(k)$)

$$i(k) = V_{xt}[k] - V_{xind}[k] = 0$$

and the position estimation C_x will be 0 (see Fig. 2). Therefore target position would only be estimated at the step transition time instant. Only if egomotion is a pure rotation would this occur. In practice sampling and misalignment errors between rotation axis and center of projection introduce small errors. These errors result in a $i(k)$ different from 0 and target position is actually estimated. As a result of computing the center of motion instead of the target position, errors become significant when large displacements between consecutive frames occur. This can be considered as a non-linearity in the system.

A step in position corresponds to an impulse perturbation in velocity. Fig 4 shows the ability of the system to cancel the perturbation. Notice that only the first peak velocity is due to real target motion.

4.2 Ramp response

Now consider the target moving around the camera with a constant velocity of $10\,\mathrm{deg}/s$. As can be seen in Fig. 5 the target moves about 6 pixels from the image center before the system starts to compensate for it. It clearly presents an initial inertia where the action of the Kalman filter plays a major role. The Kalman filters the commands ($u(k)$) to the motors, limiting the effect of measurement errors and allowing for smooth motion without oscillations. After the transition period the system stabilizes following the motion of the target with a steady state error in position of 2 pixels. Considering the motor performance we have a type 1 position response to a ramp and a second order type 1 velocity response to a step. The position measurement error will be directly proportional to the speed of motion. A more accurate position estimation can lead to significant performance improvements and can contribute to compensate for the non-linearities mentioned in the step analysis. Another issue is the value of the proportional gain K. Increasing its value will reduce the effects of position error underestimation. However oscillation and loss of stability can occur on low speed target movements. An adaptive proportional control can be the solution. The use of an integration is another option despite the fact that it

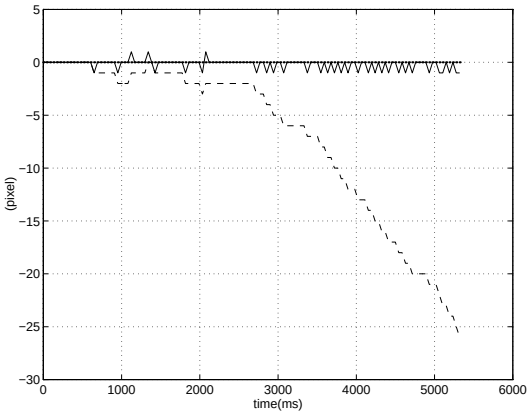

Figure 6: Graph1:regulation performance (target position (- -) and velocity (-) in the image).

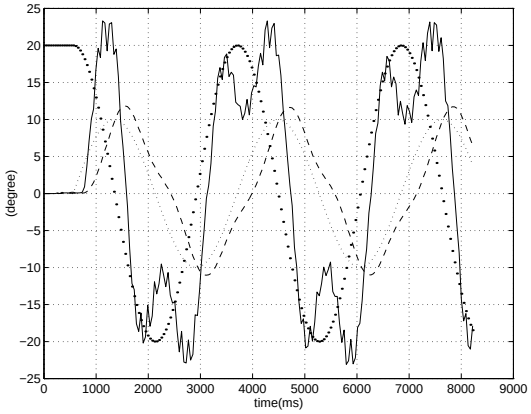

Figure 7: Servo mechanic performance in position. Motor position (- -) and velocity (-).Target position (:) and velocity (.)

might lead to some unstable behavior. The algorithm for velocity estimation using optical flow only performs well for small velocities (up to 2 pixels). For higher speeds the flow is clearly underestimated. This represents a severe limitation that is partially compensated for by the proportional position error component on the motor commands. Some experiments were performed and we concluded that the system only follows motions with constant velocities up to $20\,\mathrm{deg}/s$.

The errors in velocity transduction, in particular in high speed perturbations, generate another nonlinearity. The conclusions drawn on the type and order of the responses are only valid for a strict range of velocities.

4.3 Parabola response

The perturbation is generated by a target moving around the camera with a constant angular acceleration of $5\,\mathrm{deg}/s^2$ and an initial velocity of $1\,\mathrm{deg}/s$. When the velocity increases beyond certain values flow underestimation bounds the global performance of the system. Fig. 6 shows that the system is unable to follow the object and compensate for its velocity. As a consequence the object progressively leaves the image center and the error in position increases. The time until the target disappears from the field of view is a function of the increasing acceleration.

4.4 Sinusoidal response

System reaction to a sinusoidal perturbation of angular velocity $2rad/s$ is studied. Non linear distortions, mainly caused by velocity underestimation, can

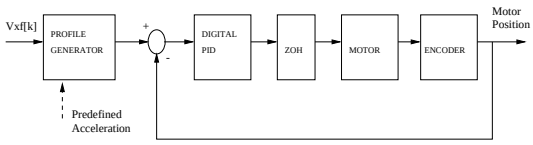

Figure 8: Low level motor control loop

be observed. Notice the phase lag and the gain in position motor response in Fig. 7.

5 Additional control aspects

In the MDOF robot head actuation is done using DC motors with harmonic drive controlled by Precision Microcontrol DCX boards. As shown in Fig. 2 the actuation input is the velocity $V_{xf}[k]$ and the output is the camera position read at the encoders. Fig.8 depicts a detailed schematic of the sub-system. Motor position is controlled using a classic closed-loop configuration with a digital PID controller. The reference inputs (in position) are computed by a profile generator. This device is controlled by the velocity command and a predefined acceleration.

$$M(z) = 0.09z^{-3}.\frac{1+0.38z^{-1}}{(1-z^{-1})(1-0.61z^{-1}+0.11z^{-2})} \tag{8}$$

Using system identification techniques the motor/actuator discrete transfer function was obtained

(equation 8). Notice that $M(z)$ represents not the motor but the low level loop of Fig.8. The pole in $z = 1$ is due to the integration needed for velocity-position conversion. A pure delay of 3 frames was estimated. This latency can be minimized by a new PID tuning.

6 Summary and Conclusions

In this paper we characterize the behavior of a smooth pursuit visual behavior by using typical test signal inputs. Smooth pursuit is based essentially on the computation of the differential flow. By considering the time responses to step, ramp, parabola and sinusoidal inputs specific performance limitations can be identified. We not only characterize the control performance of the servo-mechanical structure but also and also the performance of the visual processing routines. The results enable the identification of specific non-linearities of the system and show that prediction is essential for a tracking behavior.

References

[1] G. Hager and S. Hutchinson. Special section on vision-based control of robot manipulators. *IEEE Trans. on Robot. and Automat.*, 12(5), October 1996.

[2] R. Horaud and F. Chaumette, editors. *Workshop on New Trends in Image-Based Robot Servoing*, September 1997.

[3] P. I. Corke and M. C. Good. Dynamic effects in visual closed-loop systems. *IEEE Trans. on Robotics and Automation*, 12(5):671–683, October 1996.

[4] P. I. Corke. *Visual Control of Robots: High-Peformance Visual Servoing*. Mechatronics. John Wiley, 1996.

[5] R. Kelly. Robust asymptotically stable visual servoing of planar robots. *IEEE Trans. on Robot. and Automat*, 12(5):697–713, October 1996.

[6] W. Hong. Robotic catching and manipulation using active vision. Master's thesis, MIT, September 1995.

[7] A. Rizzi and D. E. Koditschek. An active visual estimator for dexterous manipulation. *IEEE Trans. on Robot. and Automat*, 12(5):697–713, October 1996.

[8] P. Sharkey, D. Murray, S. Vandevelde, I. Reid, and P. Mclauchlan. A modular head/eye platform for real-time reactive vision. *Mechatronics*, 3(4):517–535, 1993.

[9] P. Sharkey and D. Murray. Delays versus performance of visually guided systems. *IEE Proc.–Control Theory Appl.*, 143(5):436–447, September 1996.

[10] C. Brown. Gaze controls with interactions and delays. *IEEE Trans. on Systems, Man and Cybern.*, 20(2):518–527, 1990.

[11] J. Dias, C. Paredes, I. Fonseca, H. Araujo, J. Batista, and A. Almeida. Simulating pursuit with machines: Experiments with robots and artificial vision. *IEEE Trans. on Robot. and Automat.*, 14(1), 1998.

[12] J. Batista, P. Peixoto, and H. Araujo. Real-time active visual surveillance by integrating peripheral motion detection with foveated tracking. In *Proc. of the IEEE Workshop on Visual Surveillance*, pages 18–25, 1998.

[13] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robot. and Automat.*, 8(3):313–326, June 1992.

[14] P. Allen, A. Timcenko, B. Yoshimi, and P. Michelman. Automated tracking and grasping of a moving object with a robotic hand-eye system. *IEEE Trans. on Robot. and Automat.*, 9(2):152–165, 1993.