Performance Characterization of Visual Behaviors in an Active Vision System

João P. Barreto‡, Paulo Peixoto†, Jorge Batista†, Helder Araujo†⋆

‡Institute of Systems and Robotics
Escola Superior de Tecnologia e Gestão
Inst. Politecnico de Leiria
2400 Leiria
Portugal

†Institute of Systems and Robotics
Dept. of Electrical Engineering - Polo 2
University of Coimbra
3030 Coimbra
Portugal

Abstract. The performance of a binocular active vision system is mainly function of two aspects: vision/image processing and control. In this paper we characterize the monocular performance of smooth pursuit as well as binocular vergence control. 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 image inputs were used: step, ramp, parabola and sinusoid. The responses can be used to identify which subsystems can be optimized.

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, 5]. In [5] 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 [6, 7]. In [6] 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 [7] Hong models the dynamics of a two-axis camera gimbal and also proves that a model reference adaptive controller is Lyapunov stable. In [8] 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 [9, 10] 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 [11].

2 The MDOF binocular system– Block Diagram and Control Structure

The binocular MDOF robot head is a high-performance active vision system with a high number of degrees of freedom [12]. Real-time complex visual behaviors were implemented after careful kinematics modeling and adequate selection of basic visual routines[13, 14].

In most cases visual servoing systems are analyzed as servo systems that use vision as a sensor [15, 16]. Therefore the binocular tracking system should be considered as a servomechanism whose reference inputs are the target coordinates in space and whose outputs are the motor velocities and/or positions. 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 as a regulator whose goal is to keep the target in a certain position in

⋆ Email:{jpbar,peixoto,batista,helder}@isr.uc.pt

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 and/or velocity in the image it has to be compensated for. Our goal is to characterize the control performance of the several visual behaviors of the system namely smooth pursuit and vergence control.

2.1 Smooth Pursuit Block Diagram

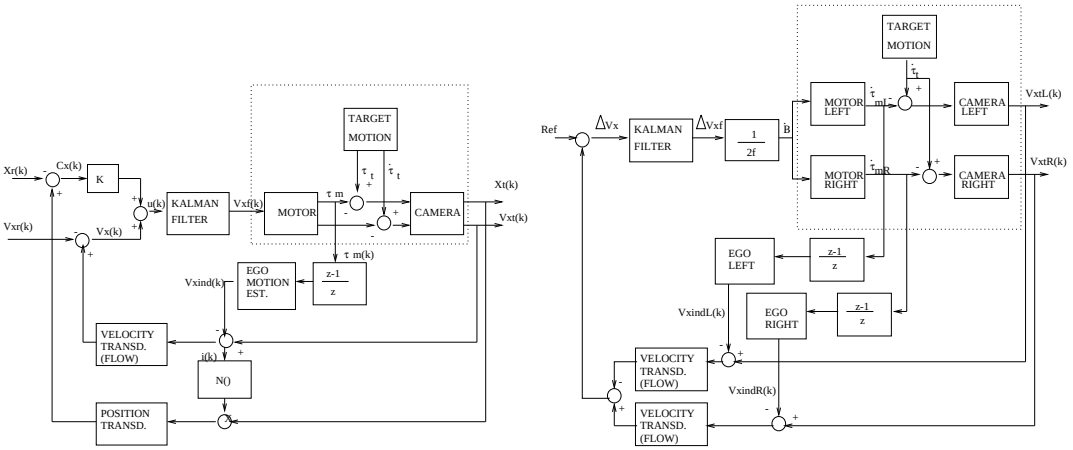

Fig. 1.: Left:Smooth pursuit block diagram. Right:Vergence block diagram

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 1(left) (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. Even though the two quantities are closely related, this formal distinction allows for 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. Block $N(i(k))$ represents a non-linear function described in equation 1.

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

$V_{xf}(k)$ 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. Considering that the motion computed in 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 velocity ($V_{xind}(k)$) is compensated for, target velocity in 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 $N(i(k))$.

2.2 Vergence Block Diagram

In this binocular system, pan and tilt control align the cyclopean Z (forward-looking) axis with the target. Vergence control adjusts both camera positions so that both target images are projected in the corresponding image centers. Retinal flow disparity is used to achieve vergence control. Vergence angles for both cameras

are equal and angular vergence velocity is computed in equation 2.Δv_{xf} is the horizontal retinal motion disparity and f the focal length.

$$\frac{\partial \beta}{\partial t} = \frac{\Delta v_{xf}}{2f} \tag{2}$$

A schematic for vergence control is depicted in Fig.1(right). Horizontal target motion disparity is controlled or regulated. As explained in the smooth pursuit analysis, egomotion is estimated for each camera. After that target velocities in both left and right images are computed using differential flow. Estimated horizontal disparity (Δv_{xf}) is obtained by filtering the difference of measured velocities in both images.

Both in smooth pursuit and vergence control, target motion acts as a perturbation that has to be compensated for. To study and characterize system regulation/control performance usual control test signals must be applied. Two problems have to be considered:

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

3 Reference Trajectories Generation Using Synthetic Images

To characterize the system ability to compensate for the perturbations due to target motion, specific signals have to be generated. 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. Given a predefined motion, captured frames will depend, not only on the target position, but also on the camera orientation. Due to the change on the system geometry as a result of its operation, images have to be generated on line to take into account the specific geometry at each time instant. Therefore at each time instant both target position and camera orientation have to be known in the same inertial coordinate system. The former is calculated using a specific motion model 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 optical center (monocular case) or at the origin of the cyclopean referential (binocular case).

To accurately describe the desired target motion in space the corresponding equations are used. Motion coordinates are converted into inertial cartesian coordinates by applying the suitable transformation equations[17]. Target coordinates in the inertial system are converted in camera coordinates. This transformation depends on motor positions that are known by reading the encoders. Perspective projection is assumed for image formation. These computations are performed at each frame time instant.

4 The reference trajectories equations. Monocular Situation.

To characterize control performance, target motion correspondent to a step, a ramp, a parabola and a sinusoid should be used to perturb the system.

4.1 Reference trajectories defined for the actuators

Consider the perturbation at actuator/motor output. The reference trajectories are studied for both a rotary and a linear actuator. In the former the actuator is a rotary motor and the camera moves along circular trajectories. In the latter the actuator induces motion along a linear path.

Rotary motor/actuator

$$\theta(t) = Const \tag{3}$$

$$\theta(t) = \omega.t \tag{4}$$

$$\theta(t) = \frac{\gamma}{2}.t^2 \tag{5}$$

$$\theta(t) = A\sin(\omega.t) \tag{6}$$

The pan and tilt degrees of freedom of the monocular system cause camera rotation around the Y (pan) and X (tilt) axis. Consider target motion equations defined in spherical coordinates (ρ, ϕ, θ), where ρ is the radius or depth, ϕ the elevation angle and θ the horizontal angular displacement. Equations 3, 4, 5 and 6 describe a step, a ramp, a parabola and a sinusoid for the pan motor. For instance, if the target moves according to equation 4, the motor has to rotate with constant angular velocity ω to track the target. These conclusions can be extended to the tilt motor by making $\theta = 0$ and varying ϕ according to equations 3 to 6.

Linear motor/actuator

$$x_i(t) = Const \tag{7}$$

$$x_i(t) = v.t \tag{8}$$

$$x_i(t) = \frac{a}{2}.t^2 \tag{9}$$

$$x_i(t) = A\sin(v.t) \tag{10}$$

Consider target motion directly described in cartesian coordinates. Cartesian equations 7 to 10 are the equivalent to spherical equations 3 to 6. Notice that depth z_i is made constant. If the actuator causes camera motion along the X axis, equations 7 to 10 define the standard control test signals.

$$\frac{d\alpha_p}{dt} = \frac{v.z_i}{v^2.t^2 + z_i^2} \tag{11}$$

Returning to rotary actuation notice that only equation 7 represents a step. None of the others are standard control test signals.

4.2 Reference test signals in the case of a static camera

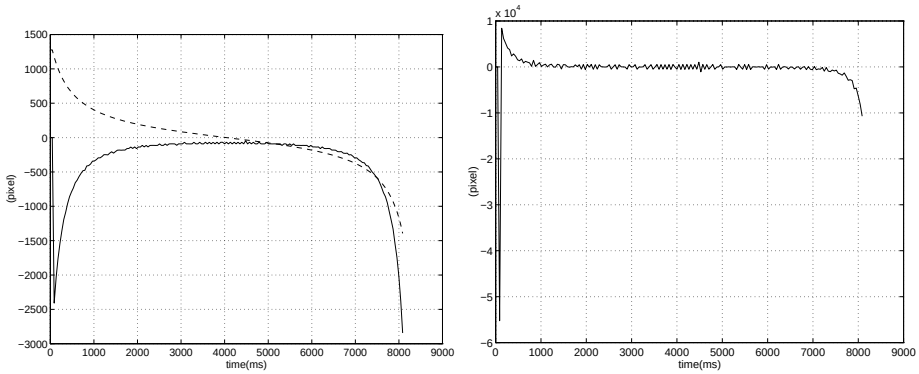

Fig. 2.: Target motion in the image. Left: Position(- -) and velocity(-). Right: Acceleration(-)

$$x_{img} = f.\tan(\theta) \tag{12}$$

$$\frac{dx_{img}}{dt} = f.\frac{d\theta}{dt}.\frac{1}{\cos^2(\theta)} \tag{13}$$

$$\frac{d^2x_{img}}{dt^2} = f.\frac{d^2\theta}{dt^2}.\frac{1}{\cos^2(\theta)} + 2.f.\frac{d\theta}{dt}.\frac{\tan(\theta)}{\cos^2(\theta)} \tag{14}$$

Consider that the camera is not moving. If the target moves in space according to equation 4 with velocity $\omega = 5(degrees)$ the signal generated in the image is depicted in Fig.2(L). This result is in agreement with equations derived in 12, 13, 14 which relate angular position (θ) in space with target image coordinates $(x_{img}, y_{img}, z_{img})$ (f is the focal length and perspective projection is assumed). Notice in Fig.2(R) that, despite the inexistence of an angular acceleration, a residual acceleration can be observed in target image

motion due to the second term of equation 14. Target motion described by spherical equations 4 to 6, does not generate the desired perturbations in the image plane when the camera is static.

$$x_{img} = f.\frac{x_i}{z_i} \tag{15}$$

$$\frac{dx_{img}}{dt} = \frac{f}{z_i}.\frac{dx_i}{dt} - \frac{f.x_i}{z_i^2}.\frac{dz_i}{dt} \tag{16}$$

$$\frac{d^2x_{img}}{dt^2} = \frac{f}{z_i}.\frac{d^2x_i}{dt^2} - \frac{f.x_i}{z_i^2}.\frac{d^2z_i}{dt^2} - \frac{2.f}{z_i^2}.\frac{dx_i}{dt}.\frac{dz_i}{dt} - \frac{2.f.x_i}{z_i^3}.(\frac{dz_i}{dt})^2 \tag{17}$$

However, in the case of a target moving in a rectilinear trajectory parallel to the image plane (constant depth), the standard perturbations are obtained. Whenever images are obtained with a static camera, linear motion described by equations 7 to 10 is adequate to generate the standard control test signals. This conclusion is confirmed by equations 15, 16 and 17 (z_i remains constant) that relate image coordinates with cartesian motion coordinates.

For spherical motion, Fig.2 shows that image motion distortion is only significant when θ is above 50 degrees.

4.3 Reference trajectories for the tracking system

In the case of static cameras the characterization of the system should be performed by using the perturbations generated by equations 7 to 10.

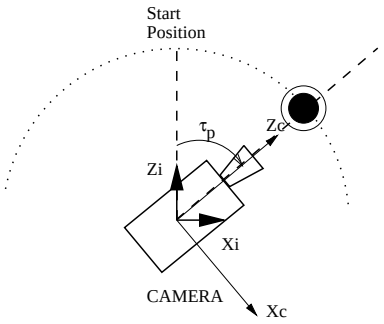

Fig. 3.: Motor angular position for perfect tracking considering target spherical/circular motion

Camera motion along a circular trajectory The MDOF binocular system uses rotary eye joints. Thus, considering the monocular situation, the camera moves along a circular trajectory. We assume camera rotation around X_c axis (pan), the target moving along a circular/spherical path (see Fig.3) and perspective projection modeling image formation.

$$x_{img} = f.\tan(\theta - \alpha_p) \tag{18}$$

$$\frac{dx_{img}}{dt} = f.\frac{d\theta}{dt}.\frac{1}{\cos^2(\theta - \alpha_p)} - f.\frac{d\alpha_p}{dt}.\frac{1}{\cos^2(\theta - \alpha_p)} \tag{19}$$

$$\frac{d^2x_{img}}{dt^2} = f.(\frac{d^2\theta}{dt^2} - \frac{d^2\alpha_p}{dt^2}).\frac{1}{\cos^2(\theta - \alpha_p)} + 2f.(\frac{d\theta}{dt} - \frac{d\alpha_p}{dt}).\frac{\tan(\theta - \alpha_p)}{\cos^2(\theta - \alpha_p)} \tag{20}$$

Target position ($x_{img}(t)$) in the image is dependent both on the camera angular position ($\alpha_p(t)$) and target angular position ($\theta(t)$) (equation 18). To compute the target velocity in the image, equation 19 is derived by differentiating equation 18. Notice that the target and camera angular positions are time dependent. By differentiating equation 19 the expression for target acceleration in image is obtained (equation 20).

As can be noticed in these equations, motion in the image is caused both by target motion and camera motion. For a perfect tracking situation the former is compensated by the latter and no motion is detected

in the image. Whenever perfect tracking does not happen there will be image motion as a result of tracking error. Therefore, the objective of tracking is to move the camera in such a way that egomotion compensates for the motion induced in the image by the target. From this point of view the system perturbation will be the motion induced by the target.

$$\omega_i = f.\frac{d\theta}{dt}.\frac{1}{\cos^2(\theta - \alpha_p)} \tag{21}$$

$$\gamma_i.t = f.\frac{d\theta}{dt}.\frac{1}{\cos^2(\theta - \alpha_p)} \tag{22}$$

$$A\omega_i \cos(\omega_i.t) = f.\frac{d\theta}{dt}.\frac{1}{\cos^2(\theta - \alpha_p)} \tag{23}$$

The reference trajectories that generate a perturbation in ramp, parabola and sinusoid are derived by solving the differential equations 21, 22 and 23 in order to $\theta(t)$ (ω_i, γ_i and A are the desired induced velocity, acceleration and amplitude). The difficulty is that the reference trajectories ($\theta(t)$) will depend on the system reaction to the perturbation ($\alpha_p(t)$). That is due to the fact that the image is not only function of target position in space, but also of camera orientation. Thus to induce a constant velocity in image during operation, target angular velocity must be computed at each frame time instant in function of the the tracking error.

Consider that perfect tracking is going to occur. The tracking error will be null and $\alpha_p(t) = \theta(t)$. With this assumption the solutions of differential equations 21 to 23 are given by equations 4 to 6 (making $\omega = \frac{\omega_i}{f}$ and $\gamma = \frac{\gamma_i}{f}$). These are the reference trajectories that we are going to use to characterize the system. It is true that for instance, trajectory of eq.4 (the ramp) only induces a constant velocity in image if tracking error is null (small velocity variation will occur otherwise). However it is independent of the system reaction and the generated perturbation allows the evaluation of system ability to recover from tracking errors.

5 The reference trajectories equations: vergence control case

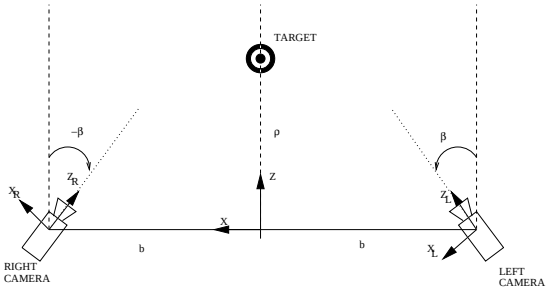

Fig. 4.: Top view of binocular system

Taking into account the considerations of last section, the reference trajectories for vergence control characterization of the binocular system depicted in Fig. 4 are now discussed. The distance between the cameras is $2b$ and symmetric vergence is assumed. The Z coordinate of the target position (in the cyclopean coordinate frame) is ρ.

$$\Delta x_{img} = 2f.\frac{-\rho.\sin(\beta) + b.\cos(\beta)}{\rho.\cos(\beta) + b.\sin(\beta)} \tag{24}$$

$$\beta = \arctan(\frac{b}{\rho}) \tag{25}$$

Vergence control is achieved using retinal disparity. The differences of target position and velocity in the images of both cameras are the system stimuli. The position retinal disparity is calculated in equation 24. Perfect tracking is achieved when β is computed by equation 25. In this case $\Delta x_{img} = 0$.

$$\Delta V x_{img} = -\frac{2fb}{\sqrt{\rho^2 + b^2}}.\frac{d\rho}{dt} \tag{26}$$

Deriving equation 24 the expression for velocity retinal disparity is obtained. Suppressing the egomotion effect (considering $\frac{d\beta}{dt} = 0$), the stimulus generated by target motion is computed in equation 26 assuming a perfect tracking situation.

$$2fb.\frac{d\rho}{dt} + v.\rho^2 = -v.b^2 \tag{27}$$

$$a = -\frac{2fb}{\rho^2 + b^2}.\frac{d^2\rho}{dt^2} + \rho.\frac{4fb}{(\rho^2 + b^2)^2}.(\frac{d\rho}{dt})^2 \tag{28}$$

$$2fb.\frac{d\rho}{dt} + Aw\cos(wt).\rho^2 = -Aw\cos(wt).b^2 \tag{29}$$

The target motion equation $\rho(t)$ that generates a motion corresponding to a ramp in image target position (constant velocity disparity v) is determined solving equation 27 derived from 26. For a parabola (constant acceleration disparity a) equation 28 must be solved. In the case of a sinusoidal stimulus, the relevant target motion equation $\rho(t)$ can be computed by solving equation 29.

6 System response to motion

In this section we analyze the system ability to compensate for perturbations due to target motion. As demonstrated spherical/circular target motion must be used to generate the standard control test signals. Pan and tilt control algorithms are identical except for some of the parameters. Therefore we will be presenting only the results for the pan control.

6.1 Step Response

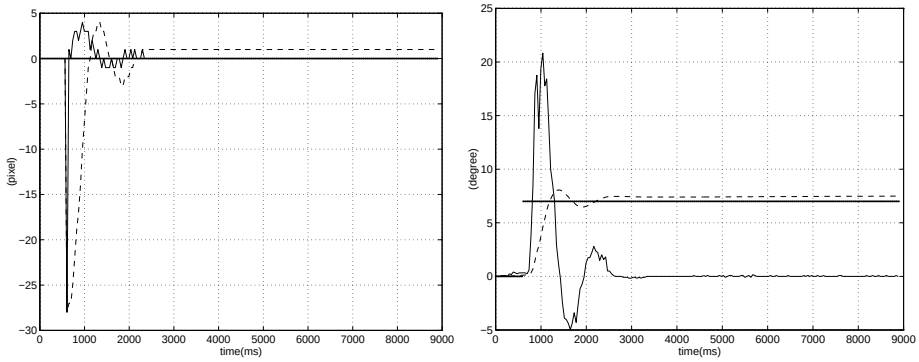

Fig. 5.: Left: Regulation performance-target position (- -) and velocity (-) in the image. Right: Servo-mechanical performance-target angular position (.), motor position (- -) and velocity (-)

A step in position (of amplitude 7 degrees) is applied to the system. Fig. 5(L) shows the evolution of the target position (X_t) in the image. An overshoot of about 10% occurs. The regulation is done with a steady state error of about 1.5 pixels (see Fig. 5(R). These observations are confirmed by Fig. 5(R) that displays the positional servo-mechanical performance. This is a typical second order step response of a type 0 system. The servo steady state error is in agreement with the regulation error. 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 performed. 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 the 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 ($V_{xind}(k)$)

$$i(k) = V_{xt}[k] - V_{xind}[k] = 0 \tag{30}$$

and the position estimate C_x will be 0 (see Fig. 1). Therefore target position would only be estimated at the step transition time instant. Only with egomotion as a pure rotation would this occur. In practice sampling and misalignment errors between the rotation axis and the center of projection introduces small errors. A step in position corresponds to an impulse perturbation in velocity. Fig 5(L) shows the ability of the system to cancel the perturbation. Note that only the first peak velocity is due to real target motion. Fig 5(R) shows the corresponding servo performance. It is a second order response.

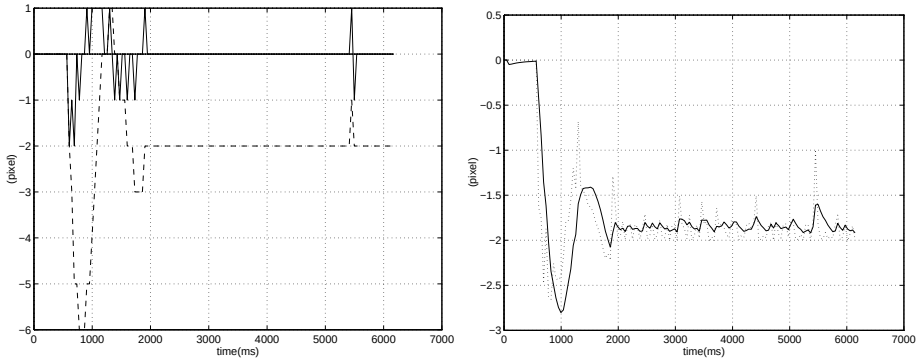

Fig. 6.: Left:Regulation performance-Target position (- -) and velocity (-) in the image. Right: Kalman filtering. Kalman input u(k) (.) and output $V_{xf}(k)$(-)

6.2 Ramp Response

Now consider the target moving around the camera with a constant rotational velocity of $10\,\mathrm{deg}/s$. As can be seen in Fig. 6(L) the target moves about 6 pixels off the center of image 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. Fig. 6(R) displays the input and output of the filter where the inertial effect can be observed. 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

$$e(k) = X_t(k) - C_x(k) \tag{31}$$

will be directly proportional to the speed of motion. The algorithm for velocity estimation using optical flow only performs well for small velocities (up to 2 pixels/frame). For higher speeds of motion 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. Experiments were performed that enabled us to conclude that the system only follows motions with constant velocities of up to $20\,\mathrm{deg}/s$.

6.3 Parabola Response

This 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. 7(L) shows that the system is unable to follow the object and compensate for its velocity. As a consequence the object image is increasingly off center of the image and the error in position increases.

6.4 Sinusoidal response

System reaction to a sinusoidal perturbation of angular velocity $2rad/s$ is studied. Fig. 8(L) shows target position X_t and velocity V_x in the image. Non-linear distortions, mainly caused by velocity underestimation, can be observed. Notice the phase lag and the gain in position motor response in Fig. 8(R).

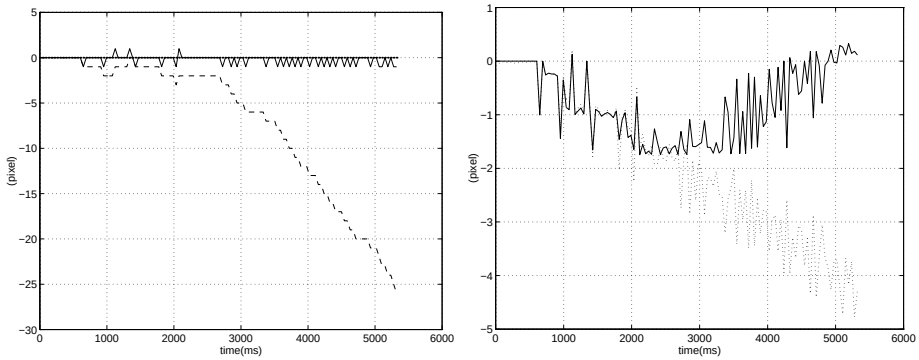

Fig. 7.: Left:regulation performance (target position (- -) and velocity (-) on image). Right:velocity estimation (target velocity (.) and flow (-)

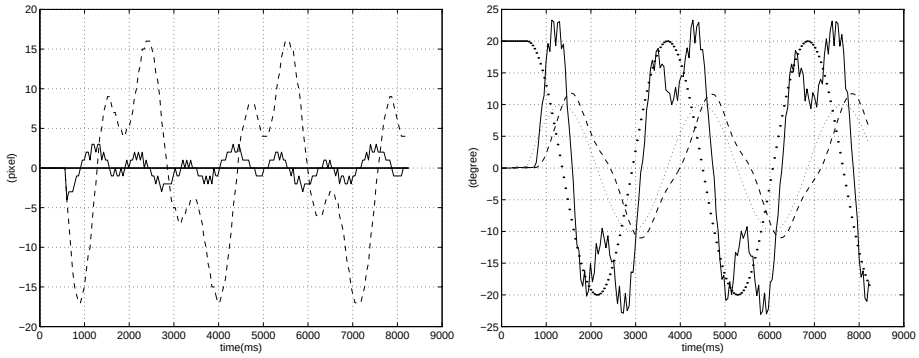

Fig. 8.: Left: Regulation Performance–Target position(- -) and velocity (-) in the image. Right: Servo-mechanical performance in position. Motor position (- -) and velocity (-). Target position (:) and velocity (.)

7 Smooth pursuit identification

During system response analysis non-linear behaviors were observed. Most of the sub-systems depicted in Fig. 1 are highly non-linear. Despite that, linear approximations can be considered for certain ranges of operation and system identification can be done.

$$M(z) = 0.09z^{-3}.\frac{1 + 0.38z^{-1}}{(1 - z^{-1})(1 - 0.61z^{-1} + 0.11z^{-2})} \tag{32}$$

Transfer function $M(z)$ (equation 32) relates the computed velocity command ($V_{xf}(k)$) in $pixel/sec$, with motor angular position in degrees. It represents, not only the motor, but also the low level PID controllers, the encoders, etc. A pure delay of 3 frames was estimated. The pole in $z = 1$ is due to the integration needed for velocity-position conversion.

$$CAM(z) = -7.15\frac{1 - 0.84z^{-1}}{(1 - 0.36z^{-1})} \tag{33}$$

$$CAM(z) = -4\frac{1 - 0.81z^{-1}}{(1 - 0.72z^{-1})(1 - 0.06z^{-1})} \tag{34}$$

The non-linear equation 12 describes the camera behavior assuming perspective projection. The approximation by a single linear model is not possible. Some restrictions in the model estimation must be considered. The transfer function described by equation 33 is adequate when the target is projected near the center of image (at a maximum distance of 10 pixels). Equation 34 displays a more general model. $EGO(z)$ models egomotion estimation. After egomotion compensation, target velocity is estimated using optical flow. Non-linear distortions were previously detected due to velocity underestimation. In fact the algorithm does not

detect velocities larger than 2 pixels/frame. $VEL(z)$ models velocity estimation for speeds lower than 2 pixels/frame. The transfer function is nearly 1 (perfect velocity estimation).

$$p(k) = 0.5 * (x(k) + x(k-1) + Vxind(k)) \tag{35}$$

As was mentioned previously, in the current implementation the center of motion is computed instead of the target position. Thus the input of the position estimation algorithm is $p(k)$ (equation 35). The non-linearity described by block $N()$ (Fig. 1) was omitted. Once again the transfer function $POS(z)$ is nearly 1 (perfect position estimation).

8 Summary and Conclusions

In this paper we address the problem of performance characterization of visual behaviors in a binocular active vision system. In order to enable the evaluation of the robustness of both vision and control algorithms in a common framework, we decided to use a typical control methodology. The different subsystems were characterized by their responses to test inputs. Due to the specific features of an active vision system several questions related to the definition of system reference inputs had to be addressed. As a result we propose and justify a methodology for the definition and generation of such reference inputs. Finally we performed the system identification of all modules of the smooth pursuit behavior including the visual processing routines (which required their linearization). As a result we think that the robustness questions have to be addressed globally, i.e. by taking into account vision and control modules simultaneously.

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 robot manipulators– a review. In K. Hashimoto, editor, *Visual Servoing*, pages 1–31. World Scientific, New York, 1993.
5. P. I. Corke. *Visual Control of Robots: High-Peformance Visual Servoing*. Mechatronics. John Wiley, 1996.
6. R. Kelly. Robust asymptotically stable visual servoing of planar robots. *IEEE Trans. on Robot. and Automat*, 12(5):697–713, October 1996.
7. W. Hong. Robotic catching and manipulation using active vision. Master's thesis, MIT, September 1995.
8. 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.
9. 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.
10. P. Sharkey and D. Murray. Delays versus performance of visually guided systems. *IEE Proc.–Control Theory Appl.*, 143(5):436–447, September 1996.
11. C. Brown. Gaze controls with interactions and delays. *IEEE Trans. on Systems, Man and Cybern.*, 20(2):518–527, 1990.
12. H. Araujo J. Batista, J. Dias and A. Almeida. The isr multi-degrees-of-freedom active vision robot head: design and calibration. In *M2VIP'95–Second International Conference on Mechatronics and Machine Vision in Practice*, Hong–Kong, September 1995.
13. J. Batista, P. Peixoto, and H. Araújo. Real-time visual behaviors with a binocular active vision system. In *ICRA97–IEEE Int. Conf. on Robotics and Automation*, New Mexico, USA, April 1997.
14. J. Batista, P. Peixoto, and H. Araújo. Real-time vergence and binocular gaze control. In *IROS97–IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Grenoble, France, September 1997.
15. 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.
16. 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.
17. J. Barreto, P. Peixoto, J. Batista, and H. Araújo. Evaluation of the robustness of visual behaviors through performance characterization. In *Workshop WS2- Robust Vision for Vision-Based Control of Motion,ICRA98–IEEE Int. Conf. on Robotics and Automation*, Louven, Belgium, May 1998.

This article was processed using the TEX macro package with SIRS98 style