

Control Aspects of Foveated Vision: The Tradeoff between Precision and Survival

Ehud Rivlin¹

Héctor Rotstein²

Boris Tsypin³

Abstract

Two-mode control provides an adequate solution to the problem of having high-resolution vision together with a large field-of-view, subject to real-time operation constraints. In this paper the fundamentals of smooth-pursuit and saccadic control are reviewed, together with simulation and implementation results.

1 Introduction

When tracking moving objects, the human oculomotor system must deal with different and often conflicting, requirements. Indeed, the visual system must be able to perform precise tasks involving heavy image processing capabilities, but remaining attentive to movements on a relatively large field of view. In order to solve this problem and still operate in real-time, the retina in the human eye has non-uniform resolution: sensor density is very high on the central region called *fovea*, and reduces rapidly outside this region. In this manner, the oculomotor system solves what the authors call the *tradeoff between precision and survival*: precision, since heavy image precision tasks can be performed in a region of interest, survival, since potential “dangers” in the field of view can be detected.

When designing control strategies for active vision systems, researchers have attempted to copy some of the characteristics of the human oculomotor model. Based mainly on heuristic arguments, foveated vision and two-mode tracking, which are arguably the two major characteristics of eye-tracking, have been often implemented in practice, mostly based on *ad-hoc* considerations. On seeking to close the gap between theory and

practical applications, the first two authors have formulated a setup in which the benefits of implementing these two mechanisms in an active vision system can be evaluated systematically. The approach is based on control and image processing considerations, and formulates the tradeoff between survival and precision as a pure control tradeoff.

The starting point in developing the systematic setup was to model the different aspects of the problem, including the different sub-systems involved and proposing a reasonable set of specifications. The resulting model allowed to investigate the advantages of using foveated vision, and to compute the optimal size of the fovea, i.e., the size of the region where heavy image processing is performed. It also lead naturally to consider a two-mode tracking system, which combines two different controllers. The first one is called smooth-pursuit (SC) controller, and should operate during most of the tracking process; the objective of the SC is to keep the target inside the fovea. The second, called “saccadic” controller, is activated when the target gets out of the fovea, since the SC can no longer operate; the objective of the saccadic controller is to move the target back inside the fovea. Our approach, motivated by the behavior of the visual system, is to design a time-optimal control action, i.e. a “saccade”, to drive the control system so that the controlled variable constraint is satisfied after the shortest possible time interval. After that, the SC is switched back into the loop. The way this switching is performed is critical for obtaining “good behavior”; a method is proposed which is based on a careful definition of the target set for the saccade. The interested reader is referred to [1] and [2] for a detailed description and a list of references on related subjects.

The purpose of the present work is to discuss the implementation aspects of the two-mode paradigm, including simulation results and also the application on an active vision system in the Intelligent Systems Laboratory at the Technion.

¹Dept. of Computer Science, The Technion. Haifa 32000 - Israel. E-mail: ehudr@cs.technion.ac.il

²Rafael and Dept. of Electrical Engineering, The Technion. Haifa 32000 - Israel. E-mail: hector@ee.technion.ac.il

³Dept. of Electrical Engineering, The Technion. Haifa 32000 - Israel. E-mail: tsypin@tx.technion.ac.il

The paper is organized as follows. In Section 2 the model of the system is reviewed. The model included here is a refined version of the one in [1], and includes a more detailed description of the visual sensor. In Section 3 the computation of the optimal size of the fovea, of the smooth-pursuit controller and of the saccadic control are briefly reviewed and implementation aspects are discussed. Section 4 describes the tracking system behavior in the event of a controlled variable constraint violation, namely the operation of the saccadic controller. Section 5 is dedicated to present implementation results. More specifically, the section contains details about the smooth-pursuit simulator and reports the performance of the smooth-pursuit controller on the actual active vision system. The paper is concluded in Section 6.

2 Setup and Modeling Considerations

For the purpose of addressing the basic problems of foveal vision and tracking mechanism, consider the configuration illustrated in Fig. 1, consisting of a camera mounted on a motor providing a single degree of rotational freedom. It is assumed that the center of rotation and the focal point coincide.

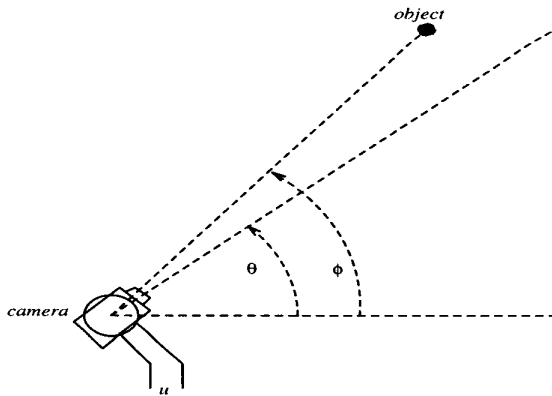


Figure 1: Problem setup.

The image of the object is acquired by a CCD camera connected to a vision card. Image acquisition entails a sampling process, both spatial and in time. The spatial sampling, which is due to the discrete number of pixels on a CCD, can be modeled as measurement noise. Time sampling, which reflects the fact that a discrete number of frames are acquired by the camera, is an essen-

tial characteristic of the system. The resulting sampling interval T is equal to 1/30 sec. for a standard CCD. In addition to sampling, the camera introduces a delay into the system, since the charges formed by the incidence of light need to be transferred to the output and then the frame has to be transferred to the computer. Since this process is synchronized by the 30 Hz frame rate, it is assumed that the picture age is also 1/30 sec.

Once in the computer, each image can be processed in order to extract information about the position of the object, e.g., the angle ϕ that forms the centroid of the object with the horizon, as measured from the axis of rotation. The time consumed by this processing will depend of the amount of data present, i.e., the "size" of the image, and on the sophistication of the image processing algorithm. In the model illustrated in Fig. 2, the image processing stage is lumped together with other effects like control law computation and communication times, in a pure *time delay* τ proportional to the size of the image (plus some overhead). If τ is larger than the sampling period T , the sequence of images has to be down-sampled since not all images can be processed in real-time.

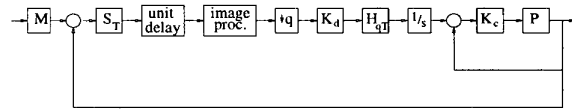


Figure 2: The closed-loop block diagram.

Assuming that the hardware and the image processing algorithms are fixed, the sub-sampling rate q will be a function of the size of the image x only; the notation q^x will be used to stress this fact.

The complete feedback block-diagram is shown in Fig. 2. The camera is modeled by the (ideal) sampler S_T and a unit step delay. Image processing is modeled by a time delay followed by a down-sampler. The block K_d denotes the discrete-time controller, working at the frequency $1/qT$, and is followed by the zero-order hold H_{qT} .

Fig. 2 also shows the plant, consisting of the camera mounted on the motor, around which an *internal* control loop is closed. Notice that the internal controller has very fast sampling rate, and can be safely model as continuous-time. The role of the internal controller is to stabilize the vision system. In the current application, where the motor is assumed to be static, this was done

by using as feedback the angle of the rotor by means of an encoder. In a more general application, where the basis of the visual system can also be moving and subject to various perturbations, stabilization must be achieved with respect to inertial space by using inertial sensors.

Finally, Fig. 2 shows the block M and an integrator $1/s$ placed right after K_d . The filter M is included for taking into account any possible *a priori* knowledge about the movement of the object one wants to track. In the current application, M is a double-integrator and consequently K_d would be designed so as to track the acceleration of the target and not its position or velocity. This particular setup can be motivated by the behavior of the human visual system [1]. By the internal model principle, if $M = 1/s^2$, then the additional integrator must be included so that internal stability can be achieved.

3 Achieving precision through foveated vision

In this section, the advantage of doing heavy processing only over part of the image frame is reviewed. The basic tradeoff can be described as follows. On one hand, using part of the image reduces computational times and leads to faster sampling-rates and smaller time-delays. On the other, keeping the target on a portion of the image presupposes tighter specifications. In order to formulate this tradeoff systematically, assume that the acceleration of the target $w(t)$ belongs to the set $\mathcal{W}(\alpha)$ defined by:

$$\mathcal{W}(\alpha)_\infty \doteq \{w \text{ s.t. } |w(t)| \leq \alpha \forall t \geq 0\}. \quad (3-1)$$

The *half* size of the fovea is denoted by x and measured in the same units as θ and ϕ . If $e(t) = \theta(t) - \phi(t)$ denotes the difference between the position of the camera and the target at time t , the control objective is to design a discrete-time controller C such that 1) The closed-loop system is stable, and 2) $|e(t)| \leq x$ for each $t \geq 0$, whenever $w \in \mathcal{W}(\alpha)$.

The existence of a controller that satisfies the above criterion will in general depend of α , since $e(t)$ cannot be guaranteed to be small for arbitrarily "large" signal. It is then natural to consider the optimization problem [1]:

Problem 1 (Maximum Size of Input)

Given x , find the largest α^x for which there exists

a controller C^x that guarantees $|e(t)| \leq x$ for any $w(t) \in \mathcal{W}(\alpha^x)$.

Let α^* denote the largest α^x ; the corresponding x^* is the size of the fovea which allows tracking the largest possible set of signals. Under the assumptions considered, x^* is then the optimal size for the fovea.

Implementation Notes

As shown in [1], Problem 1 can be solved by computing a solution to a small number of ℓ_1 sampled-data control problems. Unfortunately, the associated optimal controller can be of undesirable large order. Due to implementation constraints, the smooth controller was restricted to be a discrete-time PID which makes the optimization problem very hard to solve. The results shown in this papers are based on an ad-hoc procedure for computing the best parameters by performing a numerical search. The procedure uses two algorithms: one using gradient descent from a large number of starting point, the second using a genetic algorithm.

4 Smooth-Pursuit and Saccade

Suppose now that a smooth controller has been designed as reviewed above, so that the target is tracked inside the fovea as long as $w \in \mathcal{W}(\alpha^*)$. If this condition is violated, then the target may exit the fovea and a new control law, namely a saccade, is required for recentering the target. The saccadic control law presented in [1, 2] is based on the idea that the smooth pursuit controller, if properly initialized, can function appropriately if the state of the plant of the plant could have been achieved by some virtual signal $w^v \in \mathcal{W}(\alpha)$. The steps in a saccade are:

Switch On

Suppose that $|e(t_v) = x/2|$ and $|e(t_v + \epsilon) > x/2$, so that a saccadic action is triggered. Since saccades require relatively lengthy computations, the camera should somehow be operated in a way that will hopefully facilitate the saccade. In the absence of additional information about the variations of the position of the target, the policy adopted in the implementation was to set $e(t) = e(t_v)$, for $t > t_v$.

Modeling

In order to reduce the error signal bellow x at some future instant τ , it is necessary to predict the values of the signal $\phi(t)$ for $t \geq \tau$, based on

measurements which are usually costly to obtain and potentially contaminated by noise.

The computation of models for prediction under different sets of assumptions is considered in detail in [3], which contains an array of different algorithms. The specific algorithm should be selected depending on the standing assumptions for $\phi(t)$ and the noise which is possibly corrupting the measurements. This selection is important since it will determine the time lag required to have a prediction of future position and how accurate that prediction will be. A popular choice in the active vision field is to select an $\alpha - \beta$ or $\alpha - \beta - \gamma$ filters, which have the advantage of their simplicity. Coefficients of this filters are usually selected by using the steady-state solution of a Kalman filtering problem [3]. However, much better predictions can be made if a priori knowledge of the variations of $\phi(t)$ are available and exploited, for instance, if the objective is not to track a moving target but to do a gaze shift.

Saccade

Once the predictions about the position of the target are available at time, say, t_p , it is possible to compute the internal state $\hat{x}_M(t_s)$ of the model for future time instants. With these predictions, it is possible to compute the time-varying target set $\mathcal{O}(\hat{x}_M(k_s))$ for the saccade [2]. The problem is now to generate the control signal $u^{sac}(t)$ that drives the plant from $\hat{x}_P(t_p)$ to $\mathcal{O}(\hat{x}_M(t_s))$.

As shown in [1], the computation of the saccade can be computed by using a fast-sampling approximation, i.e., replacing the continuous-time input $w(t)$ by a piece-wise constant function:

$$w(t) = \hat{w}(k) \quad k\hat{h} \leq t < (k+1)\hat{h}$$

where $\hat{h} \ll T$. This reduces the problem to a discrete-time multi-rate one. The advantage is that in that case linear programming based algorithms exists [4] for solving these problems, and they allow the inclusion of additional constraints, like bounds on the tolerable control actions. Notice that the constraint on the target set is a linear one, and so can be incorporated with minor modification into the formulation.

Switch Off

The special construction for obtaining a saccadic control law, allows to switch the smooth pursuit controller back into the loop without transients. See [2].

Implementation Notes

Following [1, 2], the saccade is computed by solving a linear programming problem with a large number of constraints. Indeed, since the target set \mathcal{O} depends also on how "far back" one looks for choosing w^v , one should use an infinite number of constraints. Even if a finite number is used, the resulting linear programs become quite hard to solve. Instead, one could divide the computations into two subproblems, one used for computing w^v , and the second for finding the saccade. Also, the computation of w^v can be approximated by iterating over quadratic problems which can be solved efficiently. The resulting computations are discussed in detail in the forthcoming paper [5].

5 Simulation and Implementation

We tested our methods on the Technion Robot Head. The head consists of two cameras mounted on a mechanism providing four mechanical degrees of freedom (see Fig 3). For the purposes of the present research, the pan axis of one camera was used for control, while a laser pointer was attached to the other camera. This laser provides a target for the controlled system which can be controlled by the computer.

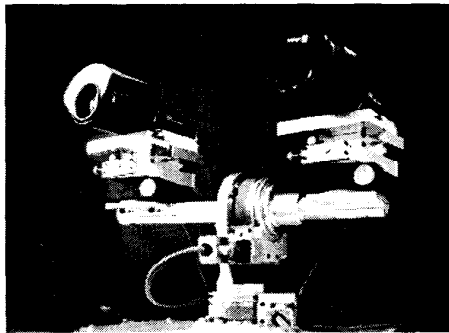


Figure 3: The Technion Robot Head.

Through experiments, the following open-loop transfer function was obtained for the pan axis of interest, relating normalized voltage input to the motor with angular displacement of the axis:

$$p_{in}(s) = \frac{-3.3012s + 3.9197 \cdot 10^4}{s^2 + 64.7983s}$$

The parameters of the controller available with the controller board of the Technion Head was subsequently designed using classical control considerations. It was found that proportional con-

control suffices to provide very good open loop performance, in terms of stability, bandwidth and noise rejection properties. The theory presented above for the design of a smooth pursuit controller was applied. In order to do this, an integrator was added in cascade with the internal closed loop of p_{in} and k_{in} :

$$\frac{1}{s} \frac{p_{in} k_{in}}{s + p_{in} k_{in}} = .5 \frac{-3.3012s + 3.9197 \cdot 10^4}{s^3 + 63.1477s^2 + 1.96 \cdot 10^4s}$$

The optimal performance of the system when considering discrete-time PID controllers with sampling time $h = q(1/30)sec$ is illustrated in Fig. 4. As expected from the interconnection, the performance degrades monotonically with decreasing sampling rate.

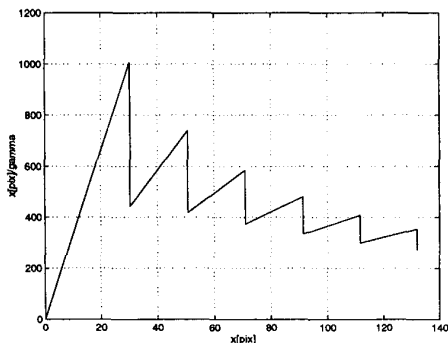


Figure 4: The optimal performance of the system when considering discrete-time PID controllers.

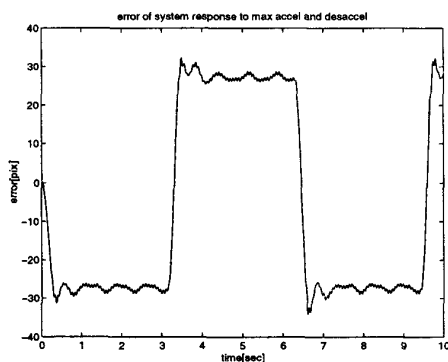


Figure 5: The performance of the smooth pursuit controller.

A model for the image processing block is now required to trade-off against the performance degradation induced by the sampling-rate. Due to the character of the target, a simple correlation algorithm was employed to localize the target from each essentially one-dimensional image. As a consequence, the computational time can be model

through the equation

$$t = a_0 + a_1 x$$

where x is half the size of the fovea, normalized between zero and one. Comparing this cost with the performance in Fig. 4, it was found that the optimal trade-off is achieved at $q = 2$, meaning that the optimal fovea has a size of 30 pixels, and that only one out of two images should be used as input to the control system. The resulting controller is capable of tracking within the fovea, target accelerations of $250^\circ/sec^2$. The performance of the smooth pursuit controller is illustrated in Fig. 5. There, the target is initially at rest at the center of the fovea, then it accelerates with the maximal admissible acceleration in one direction, and finally desaccelerates and accelerates in the opposite direction. The figure shows that at overall the trajectory, the target remains inside the fovea.

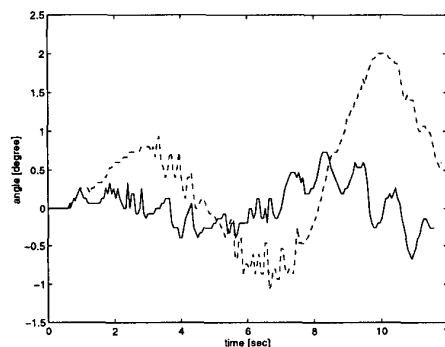


Figure 6: The performance of the smooth pursuit controller (the solid line) vs. a standard PID controller (dashed line).

Figure 6 presents the performance of the smooth pursuit controller (the solid line) vs. a standard PID controller (dashed line) to an $r(t) = t - t * \cos(2\pi/5)$ as performed running on the Technion Robot Head. The size of the fovea is 4 degrees (about 60 pixels). One can see that the standard PID controllers takes the target outside of the fovea at about the tenth second. We also checked the difference in reaction to acceleration of the two controllers. The results are presented in Figure 7. One can see that the standard PID controller converge to a steady-state after about 8 seconds, with a relatively large error (1.5 degree, about 22 pixels).

Figure 8 presents the performance of the system to sinusoidal input when running in two mode control. One can see the system performance. The

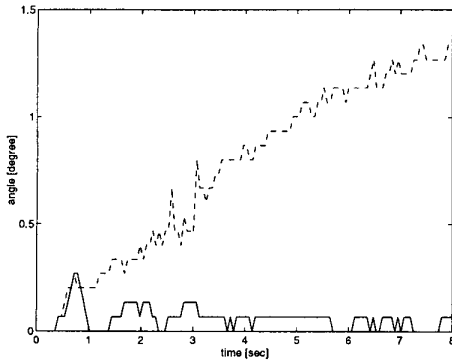


Figure 7: The performance of the smooth pursuit controller (the solid line) vs. a standard PID controller (dashed line).

system needed to execute three saccades and succeeded in going back to the SP controller.

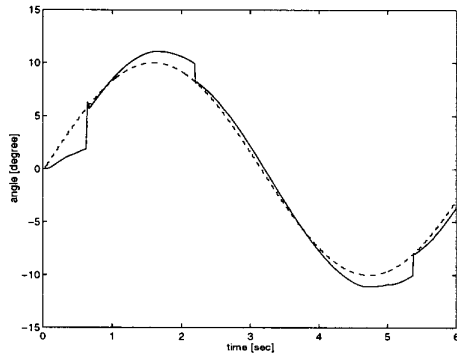


Figure 8: The performance of the system in a two-mode control.

6 Conclusion

In this paper some of the fundamental problems regarding the control of an active vision system have been addressed. It has been shown that the benefit of implementing foveal vision can be formulated as an optimization problem, since a trade-off appears between having a small window which would yield small computational delays but tighter control objectives or relaxing the control objectives but obtaining more challenging dynamics. Following the current approach, the size of the fovea is chosen as the one giving best tracking capabilities, as measured by the size of the signals which the system is guaranteed to track. Foveal vision is tightly related with smooth pursuit, since the solution to the former provides

a controller which makes the latter meaningful.

If the performance provided by the smooth controller does not suffice, which will be the case if the camera is expected to perform in a realistic environment, then one is necessarily led to consider a two-mode controller, in which the smooth controller is replaced whenever the error fails to meet specifications by a saccadic controller. This latter controller is substantially more sophisticated since it has to have capabilities of modeling the evolution of the target, generating a signal which will drive the system momentarily and then another one that should position the camera in such a way that the smooth pursuit controller can be switched-back into the loop and perform according to specifications. It was established that all this requirements can be formulated in a systematic manner.

The applicability of the smooth-pursuit controller was demonstrated through simulations on the Technion Robot Head. In this case, it was found that the processing of one out of two images is optimal for the tracking objective considered. We show also that the proposed controller can maintain a target within the fovea in cases where a standard PID controller fails to do so. We have presented some simulation results on the saccadic controller as well.

References

- [1] Ehud Rivlin and Héctor Rotstein, "Control of a camera for active vision: Foveal vision, smooth tracking and saccade," Submitted, 1995.
- [2] Ehud Rivlin, Héctor Rotstein, and Yehoshua Zeevi, "Two-mode control: An oculomotor-based approach to tracking systems," To appear in the IEEE Transactions on Automatic Control. Preliminary acceptance February 1996, 1997.
- [3] Yaakov Bar-Shalom and Thomas E. Fortmann, *Tracking and Data Association*, Mathematics in Science and Engineering. Academic Press, 1988.
- [4] J. H. de Vlieger, H. B. Verbruggen, and P. M. Bruijn, "A time-optimal control algorithm for digital computer control," *Automatica*, vol. 18, no. 2, pp. 239–244, 1982.
- [5] Ehud Rivlin, Héctor Rotstein, and Boris Tsy-pin, "Control aspects of foveated vision: The tradeoff between precision and survival," To be submitted, 1998.