Saccade and Pursuit on an Active Head/Eye Platform [1]

K.J.Bradshaw, P.F.McLauchlan, I.D.Reid and D.W.Murray

Department of Engineering Science,
University of Oxford,
Parks Road, Oxford, OX1 3PJ, U.K.

Abstract

We describe the implementation of, and results from, a real-time active surveillance vision system which detects moving objects in an everyday environment, directs the gaze of a head/eye platform towards the objects and subsequently pursues them smoothly. Target detection and pursuit are performed purely on the basis of image motion, and can continue over extended periods. Two independent parallel processes derive (i) coarse resolution motion over the entire image to direct saccadic shifts in attention over a wide field of view, and (ii) fine resolution motion in a small central region of the image used to perform smooth-pursuit. A gaze controller which selects results from the two visual processes and controls the movement of the head platform is implemented as a finite state machine.

1 Introduction

The implementation of a system capable of performing active surveillance in everyday environments requires careful consideration of the mechanical, control and vision issues involved in a closed-loop sensing system, and of how they relate to the visual task. The primary elements of surveillance are the detection of objects of interest moving in the scene and their subsequent more detailed analysis during tracking over extended periods.

Mechanically, this requires a steerable head/eye platform capable of the large joint accelerations needed to acquire a target before it escapes the range of the system, as well as smooth well-controlled performance at the lower velocities during tracking. The vision system needs a large field of view to perform wide-area surveillance, whilst also requiring accurate position and velocity information if smooth tracking of a target is to be achieved. It must also cope with the comparatively large velocities (relative to the head platform) typical of an untracked object, as well as the small relative velocities which arise when tracking. One way of satisfying these differing requirements is to have two separate visual processes, one of which delivers approximate estimates of position and velocity at a coarse scale on a wide field of view image, and another which provides more accurate information at a fine scale on a region with smaller angular size, the fovea. For real-time response, these processes need to run in parallel, at frame-rate, and exhibit minimal delay between image capture and delivery of processed results.

The use of multiple concurrent visual processes highlights the need for the third component of an active vision system, the gaze controller. This has two parts; the high-level controller which selects which visual feedback to utilize, and the servo-controller which uses that feedback to control the mechanical plant.

In this paper we explore the implementation of the visual processing and gaze control strategies required for real-time surveillance on Yorick, our reactive

[1] Supported by SERC Grant GR/G30003, Esprit Project EP5390 and by an SERC Research Studentship to KJB.

BMVC 1993 doi:10.5244/C.7.4

head/eye platform [1, 2, 3]. We demonstrate that straightforward visual processes and a straightforward gaze control strategy, when run on integrated active vision mechatronics, can create a system capable of functioning over extended periods. We show too that the interaction of control and vision is crucial; it is the gaze controller, through its knowledge of the state of the head at time of image capture, which provides base level robustness, allowing visual tasks to be initiated, interrupted and resumed.

In the next section we describe the vision algorithms running in the periphery and fovea of the system. Section 3 outlines the design of a high-level gaze controller which selects results from visual processing as appropriate and controls the direction of gaze accordingly. Section 4 describes the implementation of the system with a network of MIMD processors which perform both vision and gaze control. In Section 5 we describe some experimental results obtained with the system which illustrate its performance. Conclusions and directions for future work are given in Section 6.

The most relevant prior work is that of Clark and Ferrier who built the Harvard head and of Brown and coworkers on the Rochester Robot head. The Harvard head (see [4, 5, 6]) has a gaze control system with an inner-loop based on a primate oculomotor system and an outer loop embodying what they call a modal method of attention control which determined the most interesting thing to look at. They demonstrated saccades (or gaze transfer) and pursuit (or gaze holding) on scenes of moving blobs. The Rochester Robot [7, 8, 9] has recently been used to explored cooperative effects in simultaneously pursuing and maintaining vergence on a target [10], illustrating well the intricacies involved with coping with visual delay [11, 12]. Recent work on the KTH head [13] has also concentrated on the cooperation of two processes to provide dynamic fixation.

The work presented here differs in some broad ways (in addition to details of the control methodology). First we are concerned with saccade and pursuit in more taxing dynamic and visual environments. Secondly, we make use of image motion and position rather than position alone. Running at frame rate this distinction might seem unimportant. However, in our work image motion drives a precategorical segmentation, and so tells us what to look at and hence *where* to look. Thus motion begets position, not vice versa. Thirdly, in this paper we consider interruption between visual processes rather than process cooperation.

2 Visual Processing

The control of saccades to a moving target and of subsequent smooth pursuit of that target using an steerable head/eye platform requires the vision system to deliver the target's angular position and velocity relative to a set of axes on the head/eye platform. Visual information is, of course, recovered relative to the *camera axes* (which are likely to be moving), but it is a straightforward task to relate these to a set of axes fixed in the world, using the kinematics of the head and the known joint velocities of the platform's pan, elevation and vergence[2] axes.

Angular position and velocity relative to the camera axes are easily recovered from image position and visual motion $(\mathbf{r}, \dot{\mathbf{r}})$ using the known focal lengths of the camera (obtained by a calibration process [14]). However, typical image positions and velocities for driving saccades and pursuit are substantially different. The former are most likely to be large, as the motion is unexpected and hence untracked and most likely to occur at the edge of the image, whereas the latter, measured during successful pursuit, are likely to be near zero. Thus we divide motion processing into two: motion for saccades is derived over the whole but coarsely sampled image (the periphery), and motion for pursuit is found at fine scale in a central (foveal) part of the image. The techniques in both are similar: we

[2] Although this work uses only one camera, we will continue to use vergence to denote left-right movements of a camera.

use gradient-based methods to derive components of the motion field, a grouping process to segment foreground regions, followed by a fitting to a constant motion field to finesse the aperture problem [15].

The tight coupling between vision and control is illustrated by the need for accurate head odometry at all stages of visual processing. Compensating for a moving camera by removing the apparent motion of a stationary background requires an accurate measure of the instantaneous camera angular velocity at the time of image capture. The angular position and velocity data derived at coarse scale in the periphery are sent on to the gaze controller (which deals with the kinematics) to initiate saccadic redirection of gaze, where the aim is of course to centre the target in the foveal region of the camera and to match the target velocity, prior to smooth pursuit.

Motion detection in the fovea is essentially the same as that in the periphery, but is carried out on the fully sampled central region of the image. Like the peripheral process, the background motion arising from motion of the head platform is subtracted and a segmentation performed. However, unlike the periphery, multiple moving regions found are assumed to arise from the same moving object, because of the small size of the fovea. Hence, all moving regions are used in the computation of a single position and velocity estimate. Figure 1(b) shows output from the foveal motion detector, with a hand moving across the fovea. In this case the motion segmentation produces two distinct regions, each of which has an associated velocity and position; the overall position and velocity estimate is obtained by a least-squares fit to the flow vectors in both of the regions. The position estimate is close to the actual centre of the object, and the velocity estimate is close to that determined for each of the individual regions.

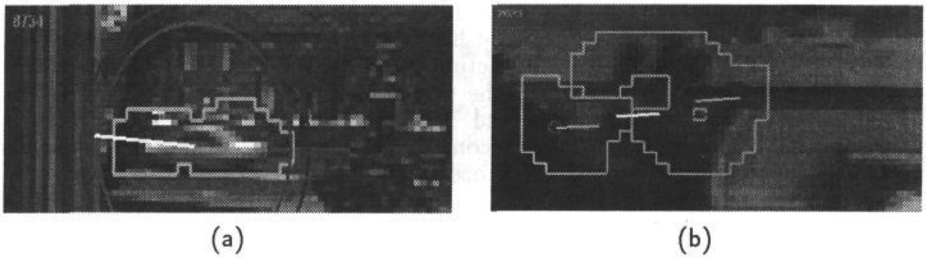

Figure 1: Motion detection and segmentation in (a) periphery (b) fovea.

3 Gaze Control

The surveillance strategy is implemented in the high-level gaze controller as a finite state machine (FSM). The FSM has four active states, *inactive*, *saccade-wait*, *saccade-active* and *pursuit-active* which are entered and left depending on visual observations, the current state and the current "gaze-mode". The demands sent from the high-level gaze controller to the servo-controller depend upon the current state and the visual observations.

The gaze-mode of principal interest here is that of saccade to pursuit, but for experiment the FSM can be set in four further modes: saccade-only, pursuit-only, test and off. The test gaze-mode is a self-test mechanism used to evaluate the performance of saccade to pursuit control (described in Section 5). In the off mode the FSM remains in the inactive state, disregarding the vision results recevied at the gaze controller.

The FSM for saccade to pursuit is given in Figure 2. The process begins in the *saccade-wait* state. When the peripheral motion sensor detects a region moving independently of the background and successfully locates the region in three consecutive frames (integrating the results using a simple Kalman filter) the high-

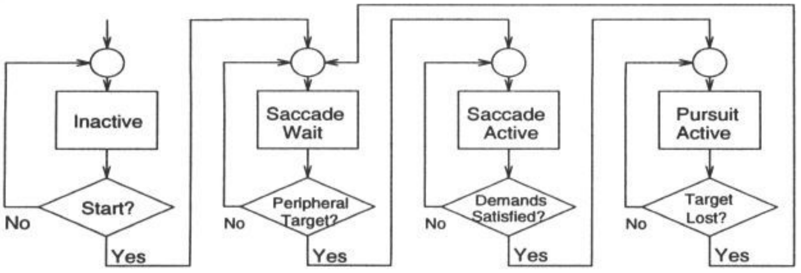

Figure 2: The finite state machine as configured for saccade to pursuit.

level gaze controller receives a result indicating the position and velocity $(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ of the target. Suppose this result is received at time $t = t_r$. The result is actually based on imagery captured at time $t = t_r - \Delta t_{process}$ where the processing time is $\Delta t_{process} \sim 110$ ms. Using a constant velocity assumption, the gaze controller predicts the likely current position $\boldsymbol{\theta}^*(t_r)$ and velocity $\dot{\boldsymbol{\theta}}^*(t_r)$. These are sent as a demand to the servo-controller, and the system enters the *saccade-active* mode. Although visual processing continues throughout the saccade, during periods of high velocity the results are useless and are ignored. Hence no new visual demand is created, the gaze controller predicts a demand $\boldsymbol{\theta}^*(t_r + 0.002n)$ every 2 ms by extrapolation of the last demand, to satisfy the 500 Hz synchronous servo-controller.

The end of the saccade is determined by examining the feedback odometry from the head/eye platform. When the head status indicates that a saccade is active and the difference between the actual gaze direction and that required by the demand is below a set threshold (the choice of which is described in Section 5) the saccade is taken to have completed. It is sufficient for the gaze controller to monitor the delayed odometry which accompanies vision results (which is delayed by $\Delta t_{process}$). This is because visual processing continues to run throughout the saccade, and the decision required of the gaze controller is simply whether or not to utilize the results, not whether to restart processing. The ability to select promptly results from differing visual inputs increases the likelihood of a smooth transition from saccade to pursuit.

When the saccade ends, the FSM enters the *pursuit-active* state. Results from the foveal motion detector are then used to perform smooth-pursuit tracking. The computed object velocity and position are used to predict the current target position and velocity, this demand is sent to the servo-controller which attempts to match the target velocity and position. The FSM will remain in this state until no motion is found in the foveal region, whereupon it returns to the *saccade-wait* state, awaiting subsequent redetection of the target in the periphery before initiating another saccade and restarting pursuit.

4 Implementation

The overall architecture of Yorick is shown in Figure 3(a), where the principal elements of head platform, vision system and gaze controller are evident.

In the present work, the head/eye platform is driven from monocular input from the left camera. Images are captured and smoothed using Datacube Digimax and VFIR-II boards and are transferred on MAXBUS to dual ported VRAM also readable by the transputers on two GEC TMAX interface boards. The TMAX transputers act as image servers to two independent "pipe groups", one performing peripheral motion processing for saccades, and the other foveal motion computation for smooth-pursuit tracking, as shown in Figure 3(b). Note that to achieve

25 Hz operation it is necessary to have two pipelines within each group, each processing half of the image.

Image information and vision results are tagged by a packet of odometry obtained from a buffer in the servo-controller indicating the head state at the time of image capture. The packet includes the current gaze angles and the instantaneous angular velocity and acceleration about each rotation axis. The odometry also contains a control word indicating the status of the head and the errors between the current position and that required by the most recent demand from the high-level controller; these are used in the state-switching operation of the FSM.

Notice that the gaze controller is split into two parts. Of most relevance to us here is the high-level gaze controller which takes possibly asynchronous inputs from the vision algorithms at rates $\leq$ 25 Hz, selects amongst those inputs and creates demands in terms of world angular positions and velocities. The gaze controller obtains prompt odometry information which it matches to the delayed odometry received via the vision processes to determine the exact delay in the pipeline. Typically, processing delays of about $\Delta t_{process} \sim 110$ ms are present between image capture and the receipt of the corresponding vision output at the controller, but the actual figure depends to some extent on the amount of motion in the image. The high-level controller takes account of these time delays (which are known exactly) and interpolates the demand to create synchronous 500 Hz output.

The servo-controller is of relatively little concern to us here. It takes the 500 Hz world coordinate demands from the high-level gaze controller and via the forward kinematics turns these into joint angles and velocities. The servo-controller receives feedback from the encoders on the joints and drives the head platform as a pointing device. The head platform has four degrees of freedom driven by geared DC motors giving axis accelerations of up to 8000 $^\circ s^{-2}$ and velocities of 500 $^\circ s^{-1}$. The mechatronics of the head and servo are fully described in [3].

The host Sparcstation is used only as an X-windows interface to send configuration parameters to the pipe groups and the gaze controller, and to handle requests for results by the user. These requests are relayed via two multiplexers (two are necessary because each transputer has only four communication links). An asynchronous request scheme is used for the display of results on the host, so that delays in the host do not affect the performance of the transputer vision system. Technical details are given in [16, 17].

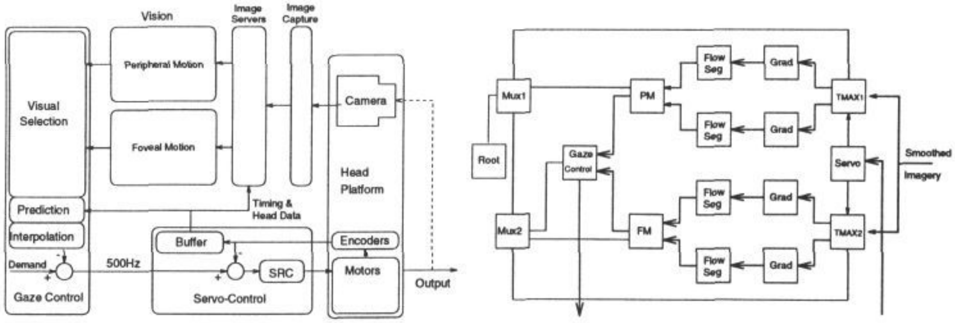

Figure 3: (a) The overall system architecture, and (b) details of the processor configuration for vision and control used in this work.

5 Experimental Results

5.1 Performance during saccades

We first examine the performance of the saccade process. Using the saccade-only mode of the FSM, after executing a saccade the system remains in the *saccade-active* state until the head joint angles reach a software endstop. At this point, the head is reset to gaze in a forward direction and the FSM placed in saccade-wait state. Thus, after the first demand from peripheral vision, no new visual demand is created, and the constant velocity extrapolation by the gaze controller causes the head to continue with constant angular velocity after the saccade.

A typical successful saccade of the head is illustrated in Figure 4. The head is initially stationary, waiting for a result from the peripheral motion sensor to indicate that a target has appeared in the periphery. Once a target is found, the corresponding demand is sent to the controller and a rapid saccade is initiated. In this example, two frames after a saccade has been initiated (i.e. after 80 ms) the target enters the foveal region. The target remains in the fovea over 15 frames (of which 6 are shown), although it is not being tracked. This shows first that for a constant velocity target, the filtered estimates from the peripheral motion processes are indeed of sufficient accuracy to allow target capture in the small fovea, which in this example subtends a solid angle of only 1.5% of that of the whole image. Secondly, it shows that the extrapolation of the demand at the end of the saccade makes timing of the start of the pursuit process noncritical.

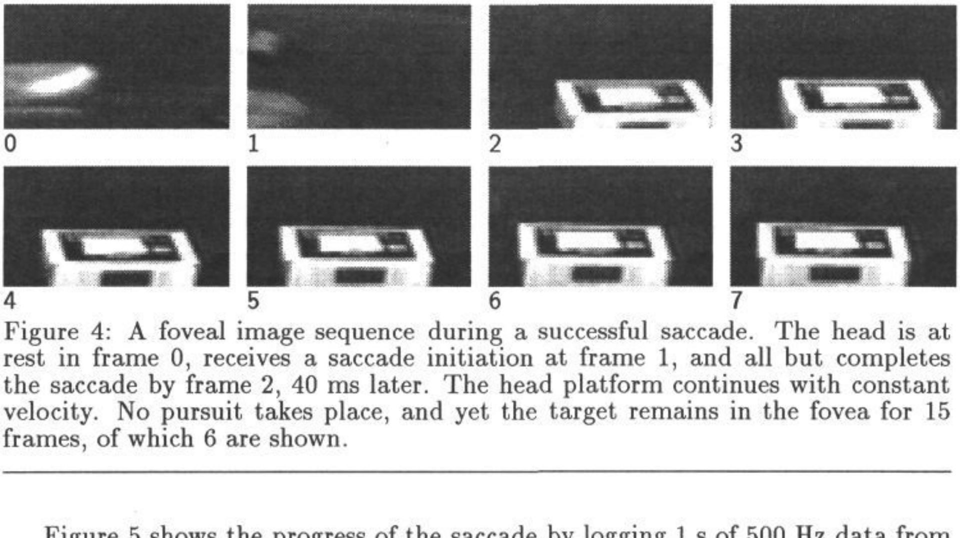

Figure 4: A foveal image sequence during a successful saccade. The head is at rest in frame 0, receives a saccade initiation at frame 1, and all but completes the saccade by frame 2, 40 ms later. The head platform continues with constant velocity. No pursuit takes place, and yet the target remains in the fovea for 15 frames, of which 6 are shown.

Figure 5 shows the progress of the saccade by logging 1 s of 500 Hz data from the servo-controller. Graph (a) shows the angular positions of the left vergence and elevation axes. From these two graphs it can be seen that the head is stationary for the first 0.24 s of the sequence. At this point the saccade demand is received from the controller and a saccade is initiated, which is complete around 50 ms later. Graphs (b) and (c) show the axes' velocities and accelerations, where we see maximum speeds of around $240°\mathrm{s}^{-1}$, and accelerations as large as $5400°\mathrm{s}^{-2}$. Graph (d) plots the instantaneous difference between the current head position and that required by the saccade demand. Obviously this is zero up to the point of receipt of a demand, at which point the error jumps to its maximum value of around 16° in vergence and 8.6° in elevation. The negative values indicate that the target is below and to the right of the current gaze position. Note that 16° corresponds to a target near the edge of the periphery: the field of view is ±20°.

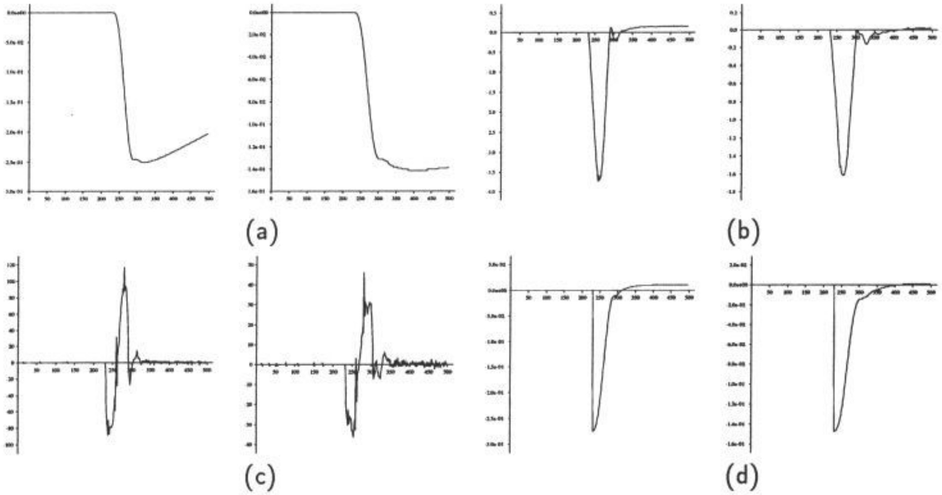

Figure 5: Servo-controller data for the vergence and elevation axes logged at 500 Hz for the saccade shown in Figure 4. (a) angular position; (b) angular velocity; (c) angular accelerations; and (d) errors between demand and current position.

5.2 Pursuit

The foveal image sequence shown in Figure 6 was obtained during an extended period of pursuit. Because segmentation is carried out in the fovea, the process is robust to substantial occlusion of the target by stationary objects.

Figure 7 illustrates the gaze angles of the system when watching the target follow a regular path over a period of 25 minutes. Using a projective transform these angles are used to intersect the gaze direction of the head with a calibrated ground plane, producing the trajectory overlay. A second pursuit sequence is illustrated in Figure 8, where a person walking past the lab is tracked.

5.3 Building in robustness

We have both the ability to make good saccades to constant motion targets and the ability to perform smooth pursuit for extended periods once the target is in the fovea. If pursuit fails, the FSM provides robustness by returning to the saccade-wait state until the motion is picked up again in the periphery.

What is most critical for a successful saccade to pursuit transition is the period for which the target is in the fovea just after the saccade, as we require at least 2 frames or 80 ms worth of image data for the foveal motion process to detect the object. The period for which the target remains in the fovea is affected by (i) the accuracy of the initial velocity estimate; (ii) the acceleration of the target; (iii) the size of the fovea; and (iv) the demand-achieved threshold used to indicate a completed saccade.

Points (ii) and (iii) are easily evaluated. If the field of view of the fovea is α and the time between the original velocity estimation and the capture of the second pursuit image in the fovea is δt, then the maximum permissible angular acceleration is $\ddot{\boldsymbol{\theta}} \sim \alpha/\delta t^2$. With a fovea size of 70x38 pixels and no subsampling in rows or columns this is approximately 2 rad s^{-1}.

The performance of the head is such that saccades are complete within 2-3 frames of the corresponding demand being generated. The gaze controller switches into *pursuit-active* mode when the head odometry, which accompanies vision results from the fovea, indicates that the gaze position is within set thresholds of the

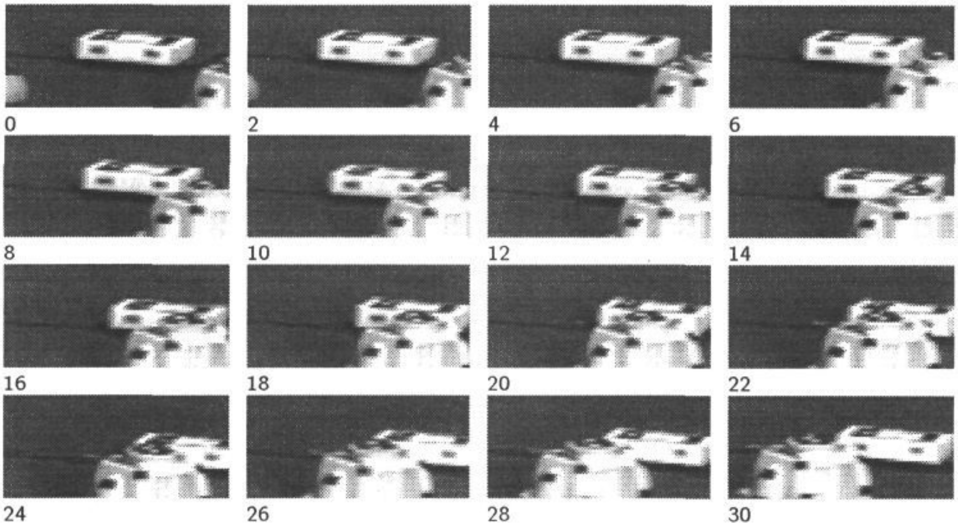

Figure 6: Fovea images for a typical pursuit sequence. The target remains in the fovea at the centre of the image throughout the sequence despite the presence of stationary objects of similar constrast in the scene.

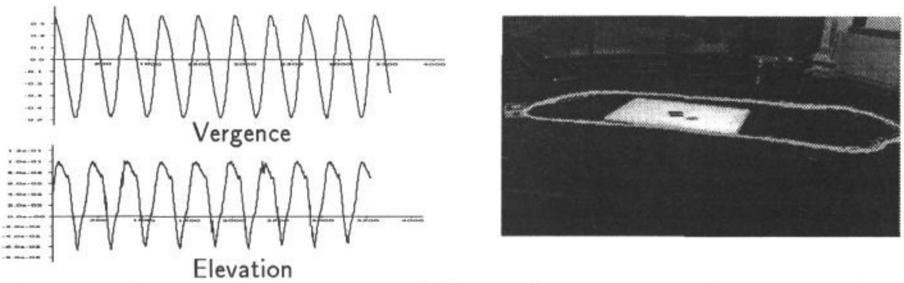

Figure 7: Gaze angles for an extended pursuit sequence, and a trace of the gaze direction of the moving head backprojected onto an image taken in a chosen resting frame.

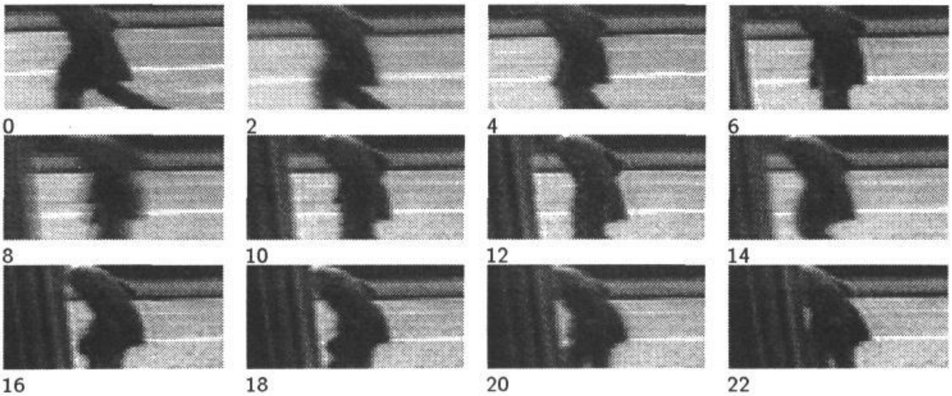

Figure 8: Foveal images for a second pursuit sequence, every second frame is illustrated. A person outside the lab is tracked as they walk past the window.

Threshold (°)	0.5	1.0	2.0	3.0	10.0
Duration (min)	5	5	5	15	50
Attempted	70	101	117	535	1083
Successful	33	68	93	416	849
% Successful	47	67	79	78	78

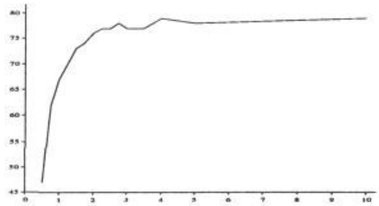

Figure 9: Saccade to pursuit test details. For various error thresholds the test duration, number of attempted and successful saccades and the success ratio are given. Graph of success percentage against error threshold.

expected target position. We use a threshold of about half the width of the fovea (i.e., 3°). If the threshold is much below this, the saccade can take considerably longer to complete, i.e. δt increases and tolerance to acceleration falls.

The "test" mode was built into the FSM to evaluate the percentage of saccades which produce a successful saccade-pursuit transition. The high-level controller counts each saccade attempted and monitors the subsequent behaviour. If pursuit is achieved for a minimum of 10 frames then the saccade is counted as a success. The user is informed of the success/failure of every attempt and the current success ratio. The system is able to monitor its own performance for extended periods; details of typical test runs are given in Figure 9.

The target is a model train running around an oval of track with two straight sections and two sharp bends. The overall success rate of the system appears to lie between 40% and 80%. The hit rate rises to 95% for saccades performed when the target is moving along the straight and falls to around 40% for saccades performed when the target is accelerating around a corner with accelerations close to the predicted limit. However, the system is able to attempt saccades at a frequency of around 4 every second, so despite the fact that any one saccade may have only a 40% chance of success, within 1 s the chance of capture is approximately 90%.

6 Conclusions

We have described an active vision system capable of performing a real-time surveillance task in a changing and unstructured environment. The issues crucial to the implementation of the system in real-time on available hardware were highlighted, in particular the tight coupling between control and vision. The system computes and segments visual motion using optical flow at a coarse scale across an entire image and at fine scale in a central foveal region as required by the different saccade and pursuit actions. Switching between actions is performed by a finite state machine, using coarse scale peripheral motion to fire motion saccades, switching to fine scale foveal motion to drive subsequent smooth pursuit of a moving target over extended periods.

The factors which are significant in achieving a smooth transition between the two behaviours have been illustrated. We have seen that a simple prediction of target motion produces successful and extremely rapid shifts in attention which can be followed by extended periods of accurate target tracking. The pursuit process has some intrinsic robustness to part occlusion, but the ability to recover after failure by attempting to recapture the target using further saccades adds an extra safety net. In conclusion we demonstrate that simple visual processes and straightforward gaze control strategies, when run within a well-designed active vision architecture, can create a system which is able to perform significant tasks over long periods.

We thus have a system which is capable of tracking an unknown target with no prior knowledge of its motion. The future direction of this particular work lies in the classification of targets based on their motion characteristics.

References

[1] D. W. Murray, F. Du, P. F. McLauchlan, I. D. Reid, P. M. Sharkey, and J. M. Brady. Design of stereo heads. In A. Blake and A. Yuille, editors, *Active Vision*, chapter 10. MIT Press, Cambridge, MA, 1992.

[2] D. W. Murray, P. F. McLauchlan, I. D. Reid, and P. M. Sharkey. Reactions to peripheral image motion using a head/eye platform. In *Proceedings of the 4th International Conference on Computer Vision, Berlin*. IEEE Computer Society Press, 1993.

[3] P.M. Sharkey, D.W. Murray, S. Vandevelde, I.D. Reid, and P.F. McLauchlan. A modular head/eye platform for real-time reactive vision. *Mechatronics*, 3, 1993.

[4] J.J. Clark and N.J. Ferrier. Modal Control of an Attentive Vision System. In *Proceedings of the 2nd International Conference on Computer Vision, Tampa FL*, pages 514–523. IEEE Computer Society Press, 1988.

[5] N. J. Ferrier. *Trajectory control of active vision systems*. PhD thesis, Division of Applied Sciences, Harvard University, 1992.

[6] J. J. Clark and N. J. Ferrier. Attentive visual servoing. In A. Blake and A. Yuille, editors, *Active Vision*, chapter 9. MIT Press, Cambridge, MA, 1992.

[7] C. M. Brown, D. H. Ballard, T. G. Becker, R. F. Gans, N. G. Martin, T. J. Ohlson, R. D. Potter, R. D. Rimey, D. G. Tilley, and S. D. Whitehead. The rochester robot. Technical Report TR 257, Computer Science Department, University of Rochester, Rochester, NY, 1988.

[8] D. H. Ballard and C. M. Brown. Principles of animate vision. *CVGIP: Image Understanding*, 56(1):3–21, 1992.

[9] R. C. Nelson and contributors. Special issue on vision as intelligent behavior. International Journal of Computer Vision, vol. 7, No. 1, 1991.

[10] C. M. Brown, D Coombs, and J Soong. Real-time smooth pursuit tracking. In A. Blake and A. Yuille, editors, *Active Vision*, chapter 8. MIT Press, Cambridge, MA, 1992.

[11] C. M. Brown. Prediction and cooperation in gaze control. *Biological Cybernetics*, 63:61–70, 1990.

[12] C. M. Brown. Gaze control with interactions and delays. *IEEE Trans. Sys. Man and Cybernet.*, TSMC-20(2):518–527, 1990.

[13] K. Pahlavan, T. Uhlin, and J.-O. Eklundh. Dynamic fixation. In *Proc. 4th Int'l Conf. on Computer Vision, Berlin*, pages 412–419, Los Alamitos, CA, 1993. IEEE Computer Society Press.

[14] P.F. McLauchlan and D.W. Murray. Active camera calibration for a head-eye platform using a variable state dimension filter. Technical Report OUEL 1975/73, Department of Engineering Science, University of Oxford, 1993.

[15] P.F. McLauchlan, I.D. Reid, and D.W. Murray. Coarse motion for saccade control. In D. Hogg and R. Boyle, editors, *Proceedings of the 3rd British Machine Vision Conference, Leeds UK, September 1992*, pages 357–366. Springer-Verlag, 1992.

[16] P. F. McLauchlan. HORATIO: Libraries for vision applications. Technical Report 1967/92, Department of Engineering Science, University of Oxford, 1993.

[17] P.F. McLauchlan and I.D. Reid. A 2-D Vision System for Real Time Gaze Control. Technical report, Department of Engineering Science, University of Oxford, 1993.