Layered Architecture for the Control of Micro Saccadic Tracking of a Stereo Camera Head

J.E.W.Mayhew
Artificial Intelligence Vision Research Unit,
University of Sheffield, Sheffield, S10 2TN,England

Y.Zheng, S.A.Billings
Department of Automatic Control and Systems Engineering,
University of Sheffield, Sheffield, S1 4DU,England

Abstract

The paper describes a 3-layered architecture for the control of the stereoscopic eye-saccade system of a stereo-camera head [1] mounted on an autonomous vehicle.

The 0-level is a proportional feedback controller providing a microsaccadic [2] control for eye movements enabling the head to foveate and track targets but requiring iteration through the vision system with the attendant computational overhead.

The 1-level provides the feedforward inverse kinematics for saccadic eye movements allowing a ballistic movement to replace the 0-level control loop. The training data is provided by the feedback error signal from the 0-level controller.

The 2-level is an adaptive lattice filter which is used to track moving targets. The filter is 'trained' using vision error-feedback from previous saccades. The filter learns to predict the future target position in the next image. This is used by the inverse kinematics module to generate the eye movement commands for the appropriate predictive saccade.

[1] The stereo camera rig used for this work comprises a 3-link kinematic chain, whose degrees of freedom are rotations around the following axes: i) Pan: a vertical axis corresponding to the 'neck'; ii) Tilt: an axis at right angles to the neck; and iii) Verge: each camera ('eye') can rotate independently around an axis at right angles to the tilt axis. The rig has been constructed so that the centres of rotation of the tilt and pan links coincide, and the centres of rotation of left and right verge and the tilt links coincide. The length of the tilt link is approximately 12.5 cm for each eye (i.e. the head is about 25 cm wide); the length of the verge link (i.e. approximately how far the centre of rotation is from the focal centre of the camera) is 5cm so that tilting the eye also produces a small translation. It is also of note that the right camera has been mounted with a 5 degree heterophoria and about 2.5 degrees of cyclotorsion. Stepper motors control the head and give a maximum saccade velocity of 50 degrees/second.

[2] Microsaccades are generally used to refer to the very small saccades which, if they have any function at all, may be used to correct errors arising from drift during fixation of a stationary target (Carpenter, 1988). We use the term microsaccadic tracking to describe a form of tracking which uses small vergence saccades (ranging in size from a few minutes of arc to two degrees), characterised by a fast movement stage, followed by a 80ms image capture stage during which the eyes remain stationary. In humans, this form of tracking may not normally occur in isolation but seems to be an important component of pursuit movements (Carpenter, 1988, page 55).

BMVC 1992 doi:10.5244/C.6.39

1 Introduction

We describe the implementation of a 3-layered architecture for the control of the stereoscopic eye-saccade system of a stereo-camera head mounted on an autonomous vehicle. This system is shown in figure 1 as a functional block diagram and has been implemented on a 4x4 transputer network (see legend for details).

0-level: This layer is a proportional feedback controller providing a microsaccadic control for eye movements enabling the head to foveate and track targets but requiring iteration through the vision system with the attendant heavy image processing overhead. The latter processing in the current implementation is a simple centre-of-gravity blob tracker. This rather crude level of image processing is driven by the real-time demands of the task and current equipment constraints. The image capture has three modes:

1. Tracking: a 3.75 degree square 'foveal' region of interest (64 x 64 pixels) is used when a target has been located and is being tracked. It may be of interest to note that the location of a small target in this fovea takes at least 20 ms (and more depending on the size of the blob).
2. Recovery: a 7.5 degree square region of interest (ROI) is used to recover when the target is temporally lost when tracking.
3. Initialisation: the full 30 degree square image for initialisation of target tracking.

In the tracking mode, image processing is done concurrently and independently in the two images; in the recovery and initialisation modes a subsampling strategy is used to locate a target in one image and then focus the search around the corresponding point in the other image.

The details of the implementation are unimportant but a principle may be worth elaborating. A tracking competence working on primitive, fast and even crude vision processing can provide the 'temporal glue' by which "the thing you saw then is the object you recognise now". Thus during tracking the foveal ROI is distributed as a continuous stream to another image processing system, completely independent of the tracking system, which samples the image stream at a very different and much slower rate. Currently this system is used only to display the images, but the direction of future system evolution is obvious.

1-level: This layer provides the feedforward inverse kinematics for saccadic eye movements allowing a ballistic movement to replace the 0-level control loop. Only a brief description of this level is given because the work has been described elsewhere (Dean et al 1991; Mayhew et al, 1992). They used adaptive PILUTs (Parameterised Interpolating Look-Up Tables) as the architecture to learn the state dependent correction to the 0-level controller. Following Kawato et al (1989) the feedback error signal from the 0-level simple proportional controller was used to provide the training data.

2-level: This is an adaptive lattice filter which is used to track moving targets. The filter is trained using error feedback from previous saccades within the current tracking sequence, so that the filter learns to predict the future target position in the next image. This is used by the inverse kinematics module to generate the eye movement commands for the appropriate predictive saccade.

For the 2-level layer we wished to develop a tracking prediction module with the following properties: i) it should be as general as possible, making minimal assumptions about the complexity and stationarity of the target trajectory; ii) it should adapt in very few time steps or samples, both to the onset of motion and to any discontinuities in the trajectory, yet at the same time it should be robust over sequences of missing data such as frequently result from occlusions and low level image processing infelicities; and iii) the implementation of the predictor should be computationally inexpensive. We describe below the details of the experimental evaluation of this module, using both simulated data, and to control the real stereo while tracking a moving light source.

2 Lattice Predictor

The use of multi-stage lattice filters (Goodwin and Sin, 1984; Alexander, 1986) for prediction is commonplace especially in the speech processing domain (Makoul,1975). The general principle underlying their design is that the successive stages of the filter compute the partial correlations (or regressions) at different delays. We have explored several different adaptive algorithms for doing this. As expected, we found gradient methods of training inefficient compared to recursive data projection algorithms. However, an alternative method has been implemented that calculates the reflection coefficients of the lattice filter directly using a decaying running average of the smoothed partial correlations. By controlling the time constant of the estimator of the partial correlations, both the requirements of fast adaptive response and relative robustness to missing data can be satisfied. Because successive stages of the filter are orthogonal and independent it is easily adapted on-line to the complexity of the signal by the simple expedient of adding or deleting stages of the lattice in response to variations in the partial correlations of the last stage. (See figure 2. For further experimental details see Zheng, et al 1991).

In implementing the adaptive lattice predictor in a simulation environment, we have found that the choice of initial conditions can significantly influence the rate of convergence of the reflection coefficients. If the reflection coefficients are initialised to zero this has the effect of introducing an artificial discontinuity in the input data which would propagate through the stages of the lattice predictor influencing the calculations of all reflection coefficients, resulting in delayed convergence and poor predictive performance.

We noticed that the first stage of a lattice predictor is very similar to a differentiator. Thus an appropriate initial condition for the reflection coefficient of the first stage should be -1. When so initialised a very significant increase in the rate of convergence of all the reflection coefficients is obtained with a much improved predictive performance.

This is of particular importance when the trajectory to be modeled contains discontinuities such as a sudden step change in velocity and/or change in direction. These occurrences can be readily recognised by monitoring the prediction error. Unless dealt with appropriately these discontinuities corrupt the future tracking behaviour. The strategy we have adopted is to maintain a running estimate of the standard deviation of prediction errors assuming they were normally distributed. If the current prediction error exceeds the 95% confidence limit, the memory is immediately flushed and all the stages of the lattice are reinitialised. The strategy is effective only because of the rapid convergence obtainable when correctly initialised.

We have compared the performance of a two-stage lattice filter, a Kalman filter of the same order, and a non-predictive tracker. The performance of the predictor is significantly better than the Kalman filter. This is because the lattice filter is optimal in the least squares sense and, unlike the Kalman filter, incorporates no assumptions about the structure of the trajectory. Another attractive feature of the lattice filter is that, because successive stages are orthogonal, it is very simple to adapt its length on-line as the complexity of the target trajectory increases or decreases. This can be done by monitoring the residuals. The Kalman filter does not have this degree of flexibility.

3 Online Saccadic Tracking

We have evaluated the lattice predictor in several modes:

1. Relative mode: visual target prediction using fixation error feedback. The filter was used to generate predictions of the future retinal coordinates of the target with respect to the fovea. The prediction was then used (via the kinematics) to move the head to a position which nulled off the predicted retinal error. The predictor has no access to the actual target trajectory but must estimate it in the context of its own saccades and the measured retinal errors. Four independent filters are used, one to track each of the retinal coordinates of the target in the left and right images.

2. Absolute mode: motor state prediction using fixation error feedback. The filter was used to generate predictions of the future motor states which would foveate the target. The predictor has access to the absolute motor states at which the image was taken, and the error measured in retinal coordinates is converted via the inverse kinematics to motor commands. Three filters are required: one for each of the verge motors and the other to control the tilt.

3. Image capture modes: serial and pipeline. The above tracking task can be broken into the following four stages: a) image capture; b) image processing; c) prediction and inverse kinematics; and d) head motion.

Pipelining is a form of parallelism which is appropriate when a repetitive activity consists of a sequence of stages. The strategy is to overlap the processing of the stages so that while stage n is being processed, stages n-1 and n-2 etc of successive instances of the action are processed concurrently. Figure 3 shows how it is possible to pipeline the components of the microsaccadic tracking task.

The advantage of pipelining is clear: it increases through-put of a processing stream. Here, the important difference from serial processing is that in pipeline mode the target-locked image sampling frequency is maximised. Furthermore, while maintaining the same sampling frequency or image capture rate, it is possible to treble the amount of time available for the image processing and inverse kinematic stages. Also, because the number of head motion stages has doubled, the maximum target velocity can be increased proportionately. From this it follows that a pipeline tracker is much less vulnerable to temporal noise than a tracker operating in serial mode. There is some potential for oscillation because the sequence involves a two-step lag but this danger is reduced by using the lattice filter to generate 2-step ahead predictions. This stabilises the system and reduces the tracking errors. Figure 4 shows the effect of using the filter to model the trajectory and the advantages over a simple non-predictive pipelined tracker in terms of the off-fovea retinal error.

4 Conclusions

This study has shown several attractive features of the lattice predictor as a component of an architecture for microsaccadic tracking. i) The order of the lattice predictor can be changed by simply adding on or taking off stages, making it easy to adapt to changes in the complexity of the input signal process. ii) The lattice predictor is capable of providing robust several-step-ahead predictions. These may be used to bridge sequences of missing data and the gap produced by sensor action delays. iii) It is robust to discontinuities in the target trajectory. iv) The lattice filter implementation is extremely economical and computationally efficient. v) It plays an important stabilising role in pipelining.

Pipelining and the resulting maintenance of the maximum image sampling frequency is potentially important, perhaps less for its effects on the tracking performance per se, but because a pipelined tracker can support other concurrently operating vision processes which themselves require high sampling frequencies with limited temporal noise (eg modelling object deformations as a time series, or building a model of the target trajectory in world geometry coordinates in order to evaluate the risk of collision). That the tracking competence can be subsumed by other vision processes is an important consideration for the long term development and evolution of the system.

5 Acknowledgements

Dr. Y. Zheng is supported by a grant from RSRE awarded to Prof. Frisby, Prof. Billings and Prof. Mayhew. The authors wish to acknowledge their debt to all the members of AIVRU.

References

[1] Alexander, S. T., (1986) **Adaptive Signal Processing**, Springer-Varlag, New York.

[2] Carpenter, R. H. S., (1988) **Movements of the Eyes**. Pion, London.

[3] Dean, P., Mayhew, J. E. W., Thacker, N., and Langdon, P. M. (1991), Saccade control in a simulated robot camera-head system: neural net architectures for efficient learning of inverse kinematics. Biological Cybernetics, 66, 27-36.

[4] Goodwin, G. C., Sin, K. S., (1984) **Adaptive Filtering, Prediction and Control**, Prentice-Hall, New Jersey.

[5] Kawato, M., (1989) Neural network models for formation and control of multijoint arm trajectory. In: Ito M. (ed) Neural programming. Taniguchi Symposia on Brain Sciences No 12. Japan Scientific Society Press/Karger, Basel, pp 189-201.

[6] Makhoul, J., (April, 1975) Linear Prediction: A Tutorial Review, Proc. IEEE, Vol. 63, No.4.

[7] Mayhew, J. E. W., (1992) ANIT: Architecture for navigation and intelligent tracking (in preparation).

[8] Mayhew, J. E. W., Dean, P., Langdon, P., (1992) Artificial neural networks for the kinematic control of a stereo camera head (in preparation).

[9] Widrow, B. and Stearns, S. D., (1985) **Adaptive Signal Processing**, Prentice-Hall, New Jersey.

[10] Zheng, Y., Mayhew, J. E. W., Billings, S. A., and Frisby, J. P., (1991) Lattice predictor for 3D vision and intelligent tracking. AIVRU Memo No 67.

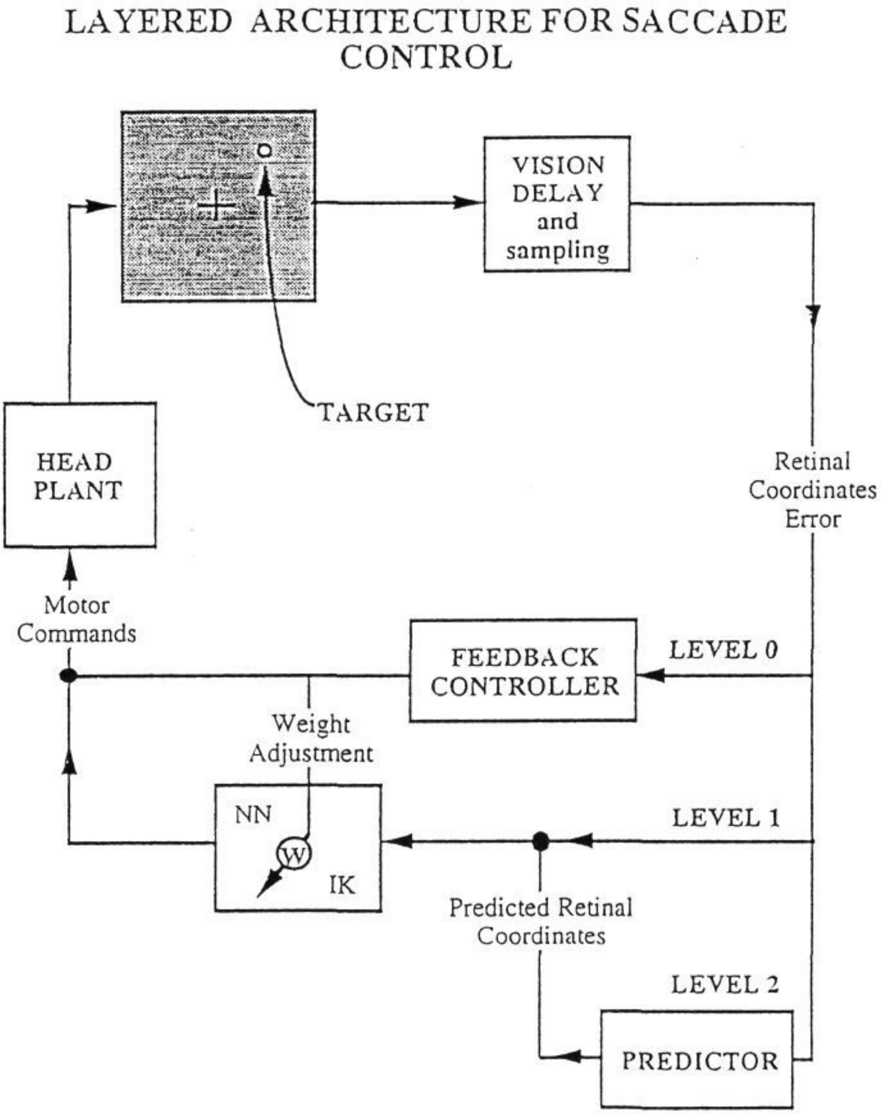
LAYERED ARCHITECTURE FOR SACCADE CONTROL
VISION DELAY and sampling
TARGET
HEAD PLANT
Retinal Coordinates Error
Motor Commands
FEEDBACK CONTROLLER
LEVEL 0
Weight Adjustment
NN
W
IK
LEVEL 1
Predicted Retinal Coordinates
LEVEL 2
PREDICTOR

Figure 1. The philosophy of subsumption applied to a perceptual-motor task: three layers of competences for a microsaccadic tracking system able to maintain zero fixation error. Level-0 is the basic competence, a proportional feedback controller providing a stable starting point that is enhanced by the addition of two further layers of visuo-motor competence. Level-1 subsumes the Level-0 competence, and improves it to provide a single saccade to achieve fixation of a stationary target. Level-2 subsumes both the lower level competences and augments them by providing zero fixation under conditions when the target is moving. See text for details.

LATTICE FILTER

$$y_{t+1} = a_0 y_t + a_1 y_{t-1} + a_2 y_{t-2} \ldots \text{ etc.}$$

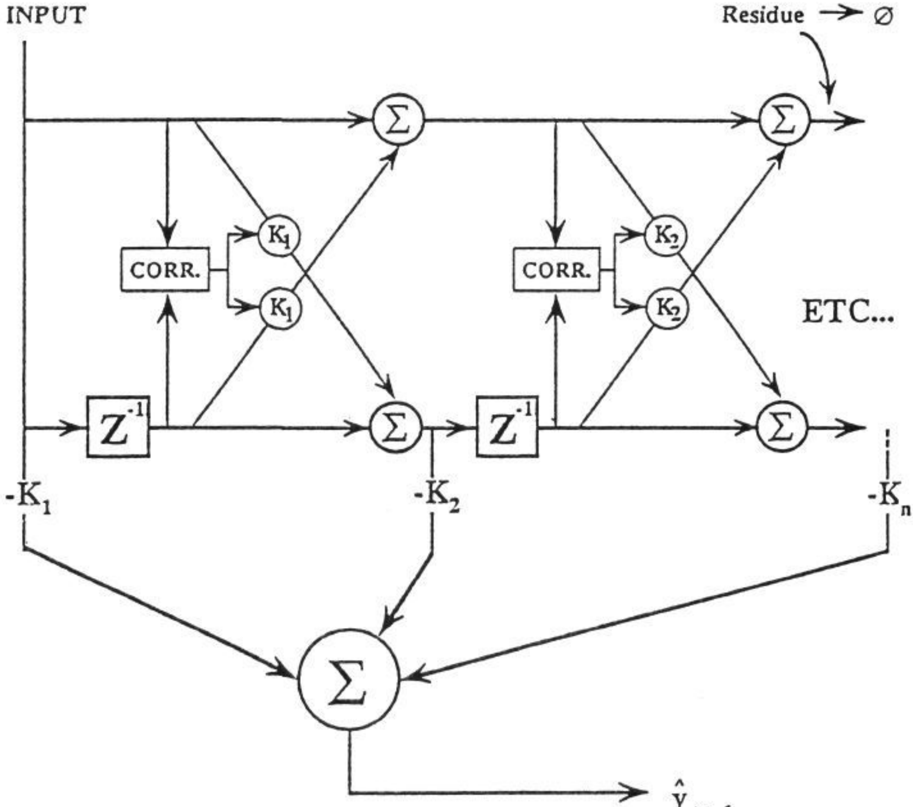

Figure 2. The lattice filter implementation of a transversal filter. The filter uses the current and past observations to form a prediction of the future output. It makes no assumption about the signal being linear and finite dimensional, it simply uses an auto-regressive model for the structure of the filter (which may not be optimal) and chooses the coefficients to minimise the mean square prediction error. K_1, K_2 etc are the reflection or partial correlation coefficients computed between the top and bottom 'delay' lines. (K_1 is generally negative, and initialised to -1 to give rapid convergence). Z^{-1} is a delay operator. Successive stages are delayed by increments of the sampling interval. The one-step-ahead prediction is given by summing the negated reflection coefficients K_1, $K_2 \ldots K_n$.

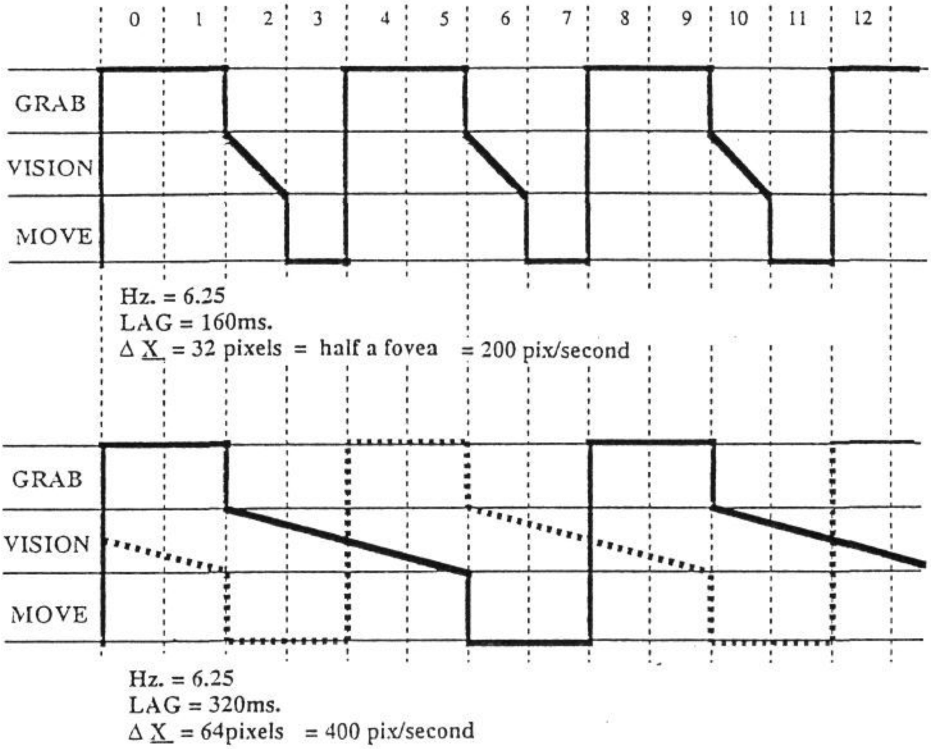

Figure 3. Serial and pipeline modes for micro saccadic tracking. Time is shown left-to-right quantised into 12 chunks, each of 40ms (set by the image frame grabbing rate). The vertical axis depicts the image capture (GRAB), image processing (VISION) and head movement (MOVE) stages. For clarity of exposition, it is assumed that the target velocity is constant and, at the maximum consonant with the time allotted to the head movement stage, is sufficient to maintain tracking without 'slipping' a frame. To minimise blur, the image capture stage is triggered by completion of the head movement stage. In the upper half of the figure (serial mode), the thick line shows which stage is in operation at any given time. In the lower half (pipeline mode), the thick line and the dotted line show the simultaneous operation of different stages.

a) Serial mode: The maximum sampling rate is 6.25 Hz, the samples have a 160 ms lag, and at a maximum head velocity of 50 degrees a second the retinal target velocity is 200 pixels a second. This is equivalent to a displacement across the image of half the 'foveal ROI' per sample. The critical feature is that to maintain this rate the demand on visual processing stage is maintained at 40 ms. This provides serious constraints on both the size of the region that can be processed and the complexity of the algorithms that can be used to support the tracking.

b) Pipeline mode: The sampling rate is maintained at 6.25 Hz. The samples lag by 320 ms, and the maximum allowed average velocity is 400 pixels a second. The feature is that if the lag does not cause instability (a function of the complexity of the target trajectory), the pipelining mode increases the time allowed for image processing by a factor of four. This provides an important buffer for maintaining the temporal stability of the tracking sample frequency.

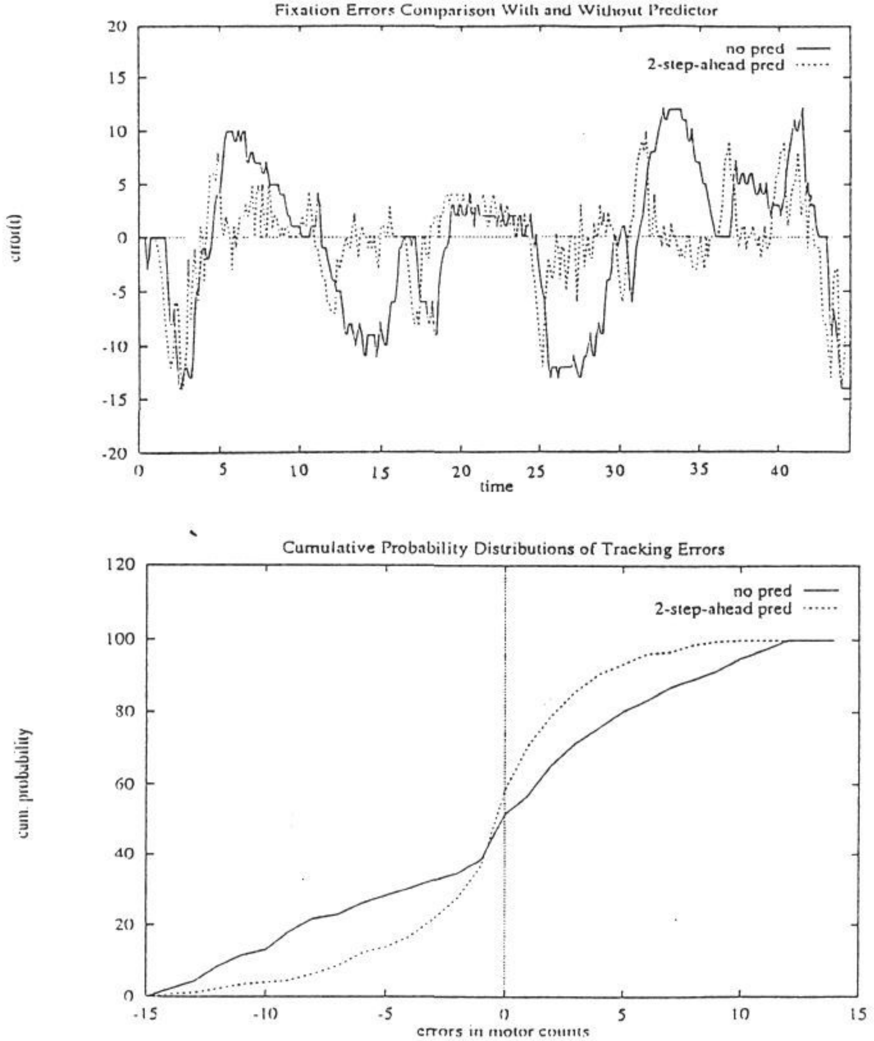

Figure 4. The improvement in microsaccadic tracking performance provided by a 4-stage, 2-step-ahead lattice predictor (absolute mode using vision error feedback) compared with a non-predictive tracker, both running in pipelining mode. The target was a small light source moved by a robot arm over the ground plane in front of the vehicle. (a) Fixation errors (represented as left verge motor counts). (b) Normalised cumulative frequency distribution of the fixation errors. The important point to be noted is the rapidity with which the errors of the predictive tracker quickly return to near-zero after major error excursions caused by a sudden change in the direction of target trajectory. The predictor rapidly learns the trajectory, the non-predictive tracker is one step behind. The difference between the two modes is statistically very marked as can be seen from (b). The rms errors are: predictor 4.16, non-predictive tracker 6.84. Each motor count corresponds to about 7.7 min visual arc so the residual tracking errors of the predictor are very small and roughly of the same order as human vision (Carpenter, 1988, p.125).