Real-time Smooth Pursuit Tracking for a Moving Binocular Robot

David Coombs
NIST, Robot Systems Div.
Bldg. 220, Rm. B-124
Gaithersburg, MD 20899
coombs@cme.nist.gov

Christopher Brown
University of Rochester
Dept. of Computer Science
Rochester, NY 14627
brown@cs.rochester.edu

Abstract

This paper examines the problem of a moving robot tracking a moving object with its cameras, without requiring the ability to recognize the target to distinguish it from distracting surroundings. A novel aspect of the approach taken is the use of controlled camera movements to simplify the visual processing necessary to keep the cameras locked on the target. A gaze holding system implemented on a robot's binocular head demonstrates this approach. Even while the robot is moving, the cameras are able to track an object that rotates and moves in three dimensions. The central idea is that localizing attention in 3D space makes simple precategorical visual processing sufficient to hold gaze.

1 Introduction

The goal of *smooth pursuit* contrasts with computer vision's traditional *passive tracking* task. In passive tracking, the cameras move without regard to the goal of tracking the target object. For instance, the cameras on a mobile robot may point straight ahead like automobile headlights. The optical flow observed by the robot will result from the three dimensional structure of the scene and the robot's motion, and the target will move about in the cameras' images. In contrast, during active visual following, the cameras rotate to follow the target. Consequently, the target's retinal slip is minimal, and the target's image is held near the center of the field of view.

There is an increasing amount of work on binocular gaze control, and we only mention a few examples here. Clark and Ferrier [5] built a gaze control system based on the model described in [11]. The system acquires and tracks white and black blobs using the first few moments and intensity value of each object. Vergence has been used cooperatively with focus and stereopsis for surface reconstruction [1] and active exploration of the environment [9]. This work combines stereo, vergence and focus to build precise range maps; vergence enables the systems to "foveate" areas of interest to obtain higher precision and confidence in their range estimates.

The goal of this work has been to build a robot gaze holding system whose only knowledge of the target is essentially that the cameras are initially pointed at it. The gaze holding problem is to maintain fixation on a moving visual target from a moving platform. The errors in camera orientation must be determined, so the location of the target's image on the retina must be found. The approach taken in this work exploits binocular cues and the fact that the cameras are actively following the target. We discuss two gaze-holding controls, binocular vergence and smooth pursuit, and their cooperative use [6].

It is important to note that it is easier to detect the tracking signals for active visual following than for tracking an object in passive stereo-motion image sequences. First, motion blur emphasizes the signal of target over the background. In passive visual following, the target's image slips across the retina and may thus be degraded by motion blur. During active pursuit, however, the eyes move to follow the target and stabilize the retinal image. Thus the image of the surrounding scene rather than the target moves across the retina and suffers from motion blur. The result is that image of the target is emphasized over the image of the background. Second, maintaining vergence isolates the target by disparity filtering. Holding vergence on the target enables the object to be isolated by simple zero-disparity filtering that detects objects at the fixation distance. Thus maintaining vergence on the target makes it possible to locate the target for pursuit control with simple precategorical visual processing. Third, active visual following also enables localized visual processing. The target's retinal location is roughly known because the pursuit system is keeping it near the center of view. This permits spatially localized visual processing, as illustrated in Figure 1.

0-8186-2855-3/92 $3.00 ©1992 IEEE

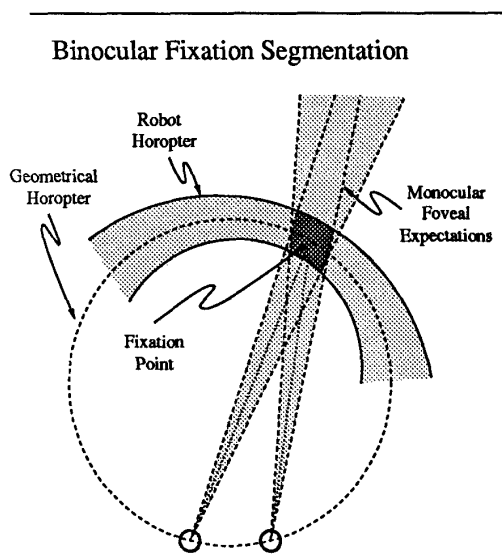


Figure 1: Top view of binocular fixation: the shading denotes the regions of space that are highlighted by foveal and disparity filtering.

2 Vergence

This section describes the vergence system. *Vergence* of eyes or cameras that share a common tilt plane results in the optic axes intersecting at some point in the tilt plane. The vergence angle of a binocular system is the angle between the optic axes of its cameras. By analogy with primate visual systems having a central high-resolution fovea, we say a camera foveates a target if the target is at the center of the visual field.

The vergence system can be thought to control the distance from the cameras to the fixation point along some specified gaze direction. The vergence problem can be defined as that of controlling the vergence angle to keep the fixation distance appropriate for the current gaze target. Since the desired vergence angle is directly related to target distance, any sensory cue to depth or depth changes may be useful to the vergence system. In humans, there is a strong link between the accommodation (focusing) and vergence systems. The most obvious direct cue is binocular disparity, but other depth cues (such as motion, texture, and shading) can also be used, as can information about depth changes (*e.g.* measured or predicted self motions, dilations or contractions of the visual field).

Vergence Error

The most useful visual cue to vergence error is binocular disparity, and the mapping from disparity to vergence error is simple. Disparity measurement has been studied extensively in the context of stereo depth reconstruction. Unfortunately most of the disparity estimators used for stereopsis are too powerful and slow for use in real-time vergence application. We use a simple phase correlation-like measure between subsampled left and right images, or "foveal" windows of such images.

The *power cepstrum* of a signal is the Fourier transform of the log of its power spectrum. It was introduced in [3] as a tool for analyzing signals containing echoes. In this application the right and left images are concatenated into a single rectangular image and compute its power cepstrum using integer FFT on a digital signal processing chip.

The filter yields a disparity histogram, and the vergence error is measured as the (x, y) location of the highest peak in disparity space—it should be brought to $(0, 0)$. The cepstral filter has the advantage of separating the disparity peaks from the central correlation peak and (more deeply) of attenuating low-bandwidth signals, hence acting like a combination of "interest operator" and matched filter correlator [10].

Since this error measure is based on *what* disparities are present but not *where* they are, the target should dominate the scene (be associated with the most common disparity). This in turn can be accomplished by spatial windowing and pursuit in dynamic scenes with many distractors (we describe this approach in Section 4). However, another more general approach is to track the disparity peak of interest regardless of its size, keeping it at zero disparity.

Vergence Control

The host Sparcstation finds the maximum peak in the cepstral filter output, converts the pixel disparity to angular coordinates, and applies the control law to issue identical and opposite (symmetric) vergence velocity commands to the camera motors. Symmetric vergence allows us to decouple the vergence and tracking controls for simplicity, but is not necessary. The Sparc issues the motor commands *after* initiating the next digitization in order to allow digitization to proceed concurrently with motor control. This causes a slight delay in issuing the motor commands, but permits a substantially higher overall sampling rate. The loop consistently takes three frame times to complete. Thus, the system achieves a servo rate of 10 Hz.

We use a proportional-integral-derivative (PID) controller (*e.g.* see [8]) in cascade with the camera motor to generate oculomotor responses to reduce the estimated disparity. The controller gains were chosen empirically to obtain slightly underdamped response, resulting in a small overshoot in the step response. The demonstration system's response to a sinusoidal input is shown in Fig. 5. The system's response to sinusoidal stimuli of frequencies up to 2 Hz suggests that

the system has second order characteristics and that its constant time delay produces the expected linear phase shift.

3 Zero Disparity Filtering

Features that have no stereo disparity can be detected in real time using a disparity filter. The region of space that contains objects that project onto the retinas with no stereo disparity is called the *horopter*. Ideally, disparity filtering can thus be used to isolate a target at a given range from its foreground and background. Our nonlinear zero-disparity filter (ZDF) has vertical edges for features, since they are identifiable features that can give useful information about horizontal disparity (Fig. 2.) The first step is to construct a vertical edge image of each image in the stereo pair. Then these images are compared in corresponding locations. If an edge is present in both images, then a feature appears in the resulting zero-disparity image. The edges must be of like phase (*i.e.* light to dark, or dark to light). The windowed output of the ZDF is input to the smooth pursuit system.

4 Smooth Pursuit Control

The goal of *smooth pursuit* is to keep the target centered in the retinas of actively controlled cameras. We have implemented a smooth pursuit system that uses precategorical visual cues (*i.e.* cues available prior to object recognition). Specifically, the smooth pursuit system simply tracks the centroid of the contents of the windowed ("foveated") ZDF output. The pursuit system rotates the cameras in tandem (the same signal is sent to each pan motor) to keep gaze directed toward the target.

There are two natural measures of target following performance: position error and velocity mismatch. The goals of image-centering and slip-minimizing can conflict (in fact they do in linear control). Smooth camera movements can only improve one of these measures at the expense of the other. A nonlinear scheme could use intermittent saccadic (fast) movements to reduce positional error, while a continuously running smooth pursuit control component tries to minimize target slip. A simpler approach is to give precedence to one of the goals. In this implementation, cameras smoothly pursue the target simply by servoing on its position error (the (x, y) image position of the centroid of the contents of the foveal window.)

Pursuit uses independent PID controls for pan and tilt, and we have experimented with $\alpha - \beta - \gamma$ prediction (a constant acceleration model in laboratory, not image, coordinates) [2]. The $\alpha - \beta - \gamma$ filter is used to apply a simple linear model to the target order to predict its future state. If the error signal can be successfully predicted, the latency of processing can be mitigated by controlling the system with predicted error signals and comparing the observed error with the predicted error once the visual signal is processed. The predictive filter is also useful to cope with signal dropout: the target dynamics can be run forward and the predicted position of the target can be tracked until the system decides to give up.

Using the $\alpha - \beta - \gamma$ predictor in the loop to predict the delayed signal can lead to more accurate tracking. Figure 3(a) shows the camera movement and tracking error as the camera tracks the image of a dark object in approximate harmonic motion with a period of about five seconds. The object is rotating in a plane and thus its distance from the camera varies and its velocity is not purely sinusoidal. The error is measured as the off-axis angle of the centroid of the object's image. There is approximately 80 ms delay in the system. The small phase difference between the sinusoidal waveforms of the target and camera motion induces a surprisingly large error. In Figure 3(a) an $\alpha - \beta - \gamma$ filter is used ($\lambda = 1$) with no predictive advance, so the tracking signal is smoothed somewhat. In Figure 3(b) the filter extrapolates the signal 50 ms into the future. The result is livelier tracking (in fact more advance destabilizes tracking) but error is reduced.

5 Combining Pursuit and Vergence

Fig. 4 shows how vergence, zero disparity filtering, and tracking work together. Vergence and tracking use foveally-processed visual signals.

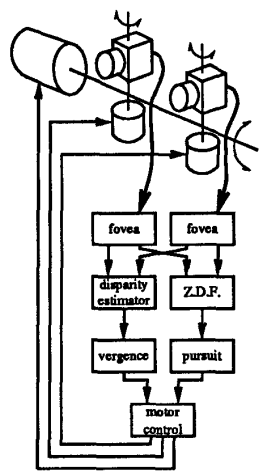
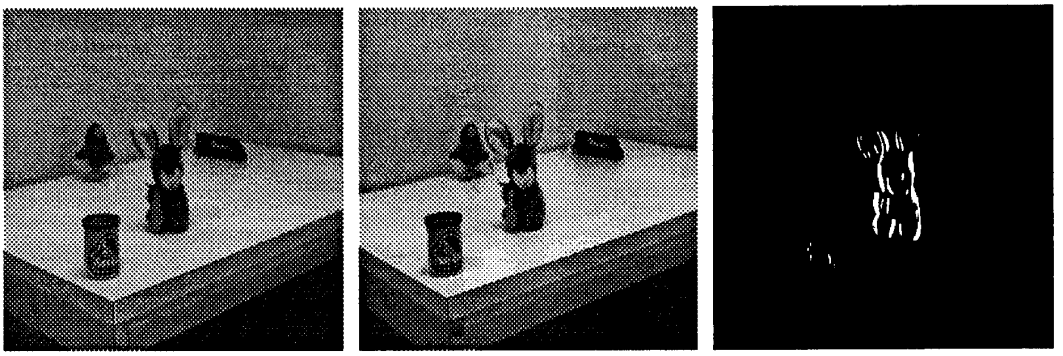
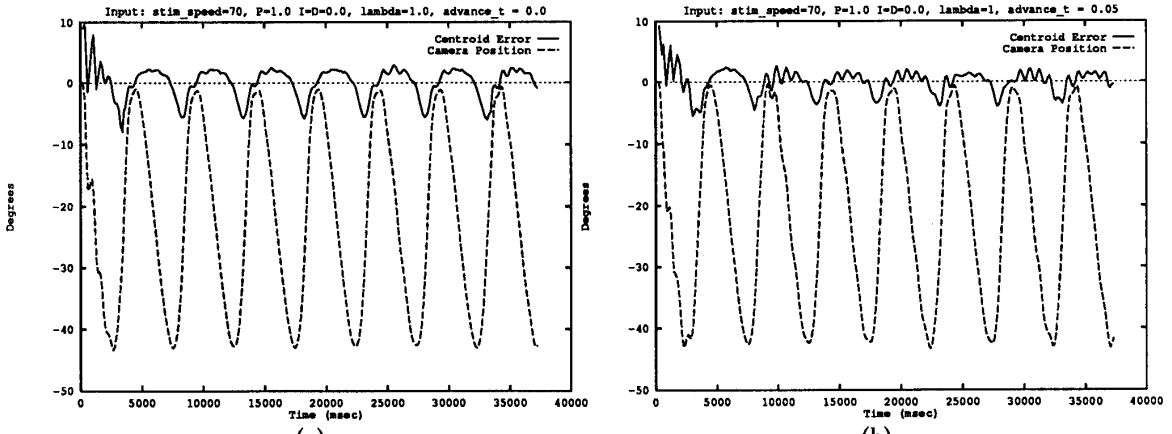


Figure 4: Binocular Vergence and Pursuit system. Nearly all of the visual processing is carried out on a Datacube MaxVideo image processing system.



(a) (b) (c)

Figure 2: Zero disparity filtering of the scene shown in stereo images (a) and (b) yields output shown in (c).



(a) (b)

Figure 3: Camera motion and error when tracking an object in approximate harmonic motion. (a) Delay of approximately 0.08s induces small phase lag but large tracking errors. (b) Camera motion and error using $\alpha - \beta - \gamma$ predictor to advance the signal by 0.050s.

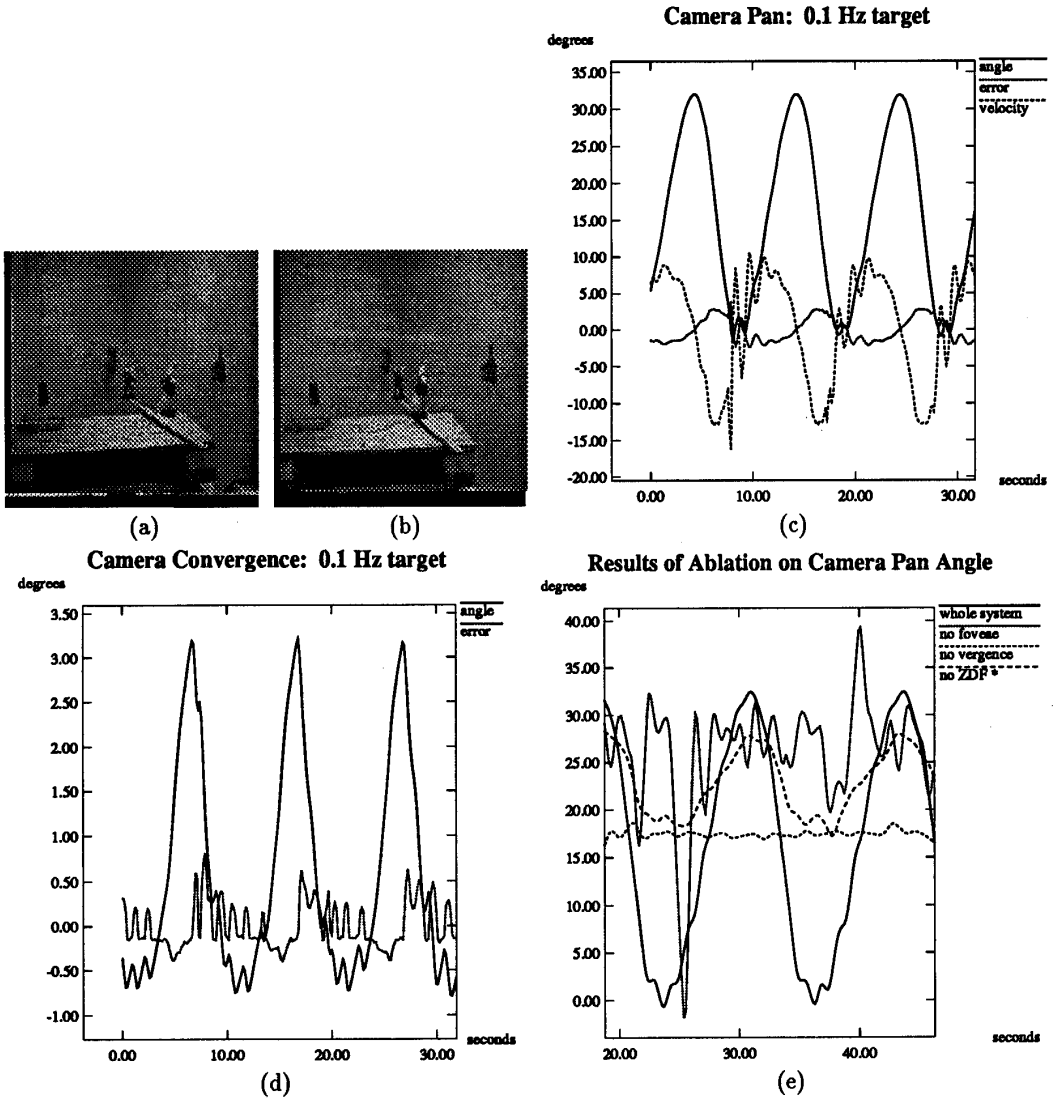


Figure 5: Gaze holding camera traces for PID control with no prediction. (a,b) show the robot's-eye stereo view of a typical setup is shown in (a,b). Measured traces of the pan (c) and vergence changes (d) show the performance of the gaze holding system in following the target. (e) shows the results of "ablation" experiments (see text).

The gaze parameter angles for pan (θ_{pan}), verge (θ_{verge}) and tilt (θ_{tilt}) map fairly directly, though not identically, onto the Rochester head's mechanical degrees of freedom, ($\phi_{left}, \phi_{right}, \phi_{tilt}$). The pan velocity is transmitted to both camera pans, and the vergence is split evenly between them. Each of the three gaze control systems operates independently, with no explicit cooperation. This simplifies the control laws (but see [4, 7]). In this experiment no predictive filtering was used.

Figure 5 shows a stereo robot's-eye view of a typical setup and the measured camera pan, tilt and convergence angles and visual error signals for a target object moving in a horizontal circle through a field of distractors. These measurements were recorded from a run with the target rotating at 0.1 Hz, and the pan angle trace reveals rotational camera velocities as high as $13\,\mathrm{deg}/s$, with the cameras lagging behind the apparent target velocity, as indicated by the non-zero observed retinal error.

In order to illustrate the function of each component of the gaze holding system, selected control components were removed from the system, and the resulting behaviors are compared with the behavior of the complete system. (Fig. 5(e)). The first trace in (e) shows the behavior of the unimpaired system for comparison. The second trace illustrates the loss of track on the target object that results from the removal of foveal processing (or peripheral suppression). The third trace demonstrates the system's inability to track the target if the vergence angle is held constant. The final trace shows how the system is distracted by objects at all distances when zero-disparity filtering is eliminated. For stability reasons, foveal reduction was also eliminated for this experiment. Including extrafoveal edges in the "target" centroid calculation dilutes the effect of features entering and leaving the "foveal" area that caused the instability. These ablation experiments show that each piece of the system contributes to the performance, and it is the combination of the simple components that allows each part to be simple.

In other experiments the Puma arm moved the robot head during the pursuit task. Since gaze-holding is is purely driven by visual feedback we would expect the system to be robust against head motion, and indeed it proved to be.

6 Conclusion

Gaze holding is one functional half of the general gaze control problem—the other half is gaze shifting. Gaze holding with moving cameras stabilizes images on the retina, thus minimizing motion blur and in turn making it easier to hold gaze on a target. During active following, motion blur also de-emphasizes the background, which makes low-level vision even easier. In a binocular system the vergence that is part of gaze holding has other beneficial effects for low level vision, such as limiting disparities for stereo matching, or the zero disparity filter we use in this work.

We have shown a visual-feedback solution to the gaze-holding problem while observer and target are moving. Robust gaze holding can be implemented with cooperating simple mechanisms. In the cooperative verging and tracking reported here, the pursuit system is driven by the centroid of intensity in the windowed output of a filter that passes features with zero stereo disparity. The vergence system uses a global disparity measure applied to the same window to keep the foveated object at zero disparity. This symbiotic cooperation of the pursuit and vergence system enables precategorical visual processing to suffice to support gaze holding.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grants numbered IRI-8903582, CDA-8822724, and IRI-89220771, and by ONR/DARPA research contract number N000114-82-K-0193.

References

[1] A. Lynn Abbott and Narendra Ahuja. Surface reconstruction by dynamic integration of focus, camera vergence, and stereo. In *Proc. of ICCV'88, the Second International Conference on Computer Vision, (Tampa, FL, December 5–8, 1988)*, 1988.

[2] Yaakov Bar-Shalom and Thomas E. Fortmann. *Tracking and data association.* Academic Press, 1988.

[3] B. P. Bogert, M. J. R. Healy, and J. W. Tukey. The quefrency alanysis of time series for echoes: Cepstrum, pseudo-autocovariance, cross-cepstrum, and saphe cracking. In M. Rosenblatt, editor, *Proc. Symp. Time Series Analysis*, pages 209–243, New York, 1963. John Wiley and Sons.

[4] C. M. Brown. Prediction and cooperation in gaze control. *Biological Cybernetics*, May 1990.

[5] James Clark and Nicola Ferrier. Modal control of an attentive vision system. In *Proc. of ICCV'88, the Second International Conference on Computer Vision, (Tampa, FL, December 5–8, 1988)*, 1988.

[6] David J. Coombs. *Real-time Gaze Holding in Binocular Robot Vision.* PhD thesis, University of Rochester, Department of Computer Science, Rochester, New York 14627 USA, January 1992.

[7] David J. Coombs and Christopher M. Brown. Cooperative gaze holding in binocular vision. *IEEE Control Systems*, June 1991.

[8] Richard C. Dorf. *Modern control systems.* Addison-Wesley, 3rd edition, 1980.

[9] Eric Paul Krotkov. *Active computer vision by cooperative focus and stereo.* Springer-Verlag, 1989.

[10] Thomas J. Olson and David J. Coombs. Real-time vergence control for binocular robots. *International Journal of Computer Vision*, 7(1):67–89, November 1991.

[11] David Robinson. Why visuomotor systems don't like negative feedback and how they avoid it. In Michael Arbib and Allen Hanson, editors, *Vision, Brain and Cooperative Computation.* MIT Press, 1987.