Stereo Fixation using Affine Transfer*

Stuart M. Fairley Ian D. Reid David W. Murray
Robotics Research Group,
Department of Engineering Science,
University of Oxford,
Parks Road, Oxford, OX1 3PJ, U.K.

Abstract

We describe an algorithm which uses affine transfer of the fixation point in a stereo pair to obtain stereo fixation of a moving object. The method runs in real-time using corners tracked temporally and in stereo. The fixation point is transferred to new left and right views using affine structure from four views – old left and right and new left and right. We also present the singular value decomposition as a means by which all possible feature points contribute to the transfer, providing immunity to poor choice of basis features. Early results are presented for the method, showing its speed and reliability.

1 Introduction

The expressions active fixation, smooth pursuit, and gaze holding are synonymous terms for the process by which the attention of an active viewer is maintained on a target of interest. The benefits of active fixation are numerous and have been well documented in the active vision literature. Clearly the most obvious advantage of such tracking is that it extends the period during which a target may be observed. But the benefits run deeper, notably enabling the use of an external or *object centred* reference frame which is better suited to many computational tasks [1, 2] than the viewer-centred coordinate frames employed in much traditional vision research. In this paper we use the theory of *affine transfer* in a binocular fixation algorithm which, among other virtues makes explicit such an object centred frame.

Consider a set of views of an object with a number of feature correspondences between the views. Given a novel view, and the image positions of a small number of the features, various researchers have shown that the image positions of the remaining features may be uniquely computed in the novel view [3, 4]. This novel view construction process is known as transfer. We applied this process to the problem of monocular tracking in [5, 6]. Affine transfer (i.e. transfer under assumptions of affine viewing conditions) was used to determine the image position of the fixation point in a new view of the target, using a set of corner feature correspondences with previous views. This novel tracking method has several advantages over other tracking methods; it is view-point invariant, insensitive to

*Supported by SERC Grant GR/J65372, by an SERC Research Studentship to SMF and by a Glasstone Fellowship to IDR. We are extremely grateful for many useful discussions with Phil McLauchlan, Kevin Bradshaw and Larry Shapiro.

BMVC 1994 doi:10.5244/C.8.74

local occlusions, and invariant to changing camera intrinsics. Importantly the method requires no prior model of the target appearance, but by-products of the fixation point transfer are the computation of an object-centred affine frame, and the target's structure relative to the frame. While monocular cues are sufficient to obtain reliable fixation [5], there are a number of reasons for wanting to extend the work to binocular tracking. In particular using stereo, it is possible to compute relative depth, and (with calibration) absolute depth and 3D trajectories of targets. Further, a binocular system is potentially more robust to occlusions in one or other eye.

Coombs and Brown [7] make use of cooperating vergence and accommodation processes in a stereo vision system for binocular fixation. Vergence is controlled by a cepstral filter process developed by Olson and Coombs [8] and ensures that the target is fixated in both eyes. Pursuit demands are generated by finding the centroid of the zero disparity edges in the stereo disparity map. The method used to generate pursuit demands means that the system in vulnerable if the target is lost in *either* eye.

Pahlavan *et al.* [9, 10] also make use of cooperative processes to achieve smooth pursuit, however they use a pseudo-correlation process (normalized image difference) for both vergence and accommodation control. This has the advantage over the work of Coombs and Brown that the accommodation correlation process can continue to be used as a monocular cue for tracking in the event that the target is invisible to one eye.

An attractive alternative to these methods is to perform symmetric processing on both left and right views. The algorithm we present adopts this approach; corner features are tracked temporally in left and right images and matched spatially between views. Correspondences common to a set of spatial and temporal views are used to construct a common object-centred reference frame. This frame is employed to effect fixation point transfer, ensuring that the trajectory of the fixation point in space (i.e. the intersection of the optic axes of the cameras) is consistent with the target trajectory.

In the following section we review the theory of affine transfer. While most previously published algorithms achieve transfer using a minimal set of feature matches, we extend this to take advantage of all possible feature matches, gaining immunity to poor basis set selections. Section 3 describes the application of transfer to gaze control and extends our previous monocular work to the binocular case. A real-time implementation of the complete tracking algorithm using a network of transputers and Yorick, a high performance head/eye platform, is described in section 4. Results from real-time experiments are given in section 5 and we conclude in section 6 with a number of areas for potential improvement which are the subject of ongoing research.

2 Theory of affine transfer

Affine transfer is based on the assumption that the camera projection can be adequately modelled by the linear equation [4]:

$$\mathbf{x} = M\mathbf{X} + \mathbf{t} \tag{1}$$

where $\mathbf{x}$ is a 2×1 image position vector, M is a 2×3 matrix, $\mathbf{X}$ is a 3×1 world position vector, and $\mathbf{t}$ is a 2×1 translation vector. This is the *affine projection equation*, an approximation to true perspective which is valid when object relief is small in comparison to depth; always the case in our work.

2.1 Four point transfer

Given multiple views of an object, an affine projection $\{M, \mathbf{t}\}$ for each view can be determined from any four point correspondences. Denoting the image position of the i^{th} point in the j^{th} view by $\mathbf{x}_i^{[j]}$, the j^{th} projection $\{M^{[j]}, \mathbf{t}^{[j]}\}$ is given by

$$M^{[j]} = [\mathbf{x}_2^{[j]} - \mathbf{x}_1^{[j]} \;\; \mathbf{x}_3^{[j]} - \mathbf{x}_1^{[j]} \;\; \mathbf{x}_4^{[j]} - \mathbf{x}_1^{[j]}], \quad \mathbf{t} = \mathbf{x}_1^{[j]} \tag{2}$$

Transfer of a point observed in the first two views to a third view may then be accomplished by computing the 3D affine coordinates of the point $\mathbf{X}$ from the first two views using least-squares on the four equations in the three unknown coordinates of $\mathbf{X}$ which result from:

$$\begin{array}{rl} & \mathbf{x}^{[1]} = M^{[1]}\mathbf{X} + \mathbf{t}^{[1]} \\ \text{and} \quad & \mathbf{x}^{[2]} = M^{[2]}\mathbf{X} + \mathbf{t}^{[2]} \end{array} \quad \Rightarrow \mathbf{X} \approx \begin{bmatrix} M^{[1]} \\ M^{[2]} \end{bmatrix}^{+} \begin{bmatrix} \mathbf{x}^{[1]} - \mathbf{t}^{[1]} \\ \mathbf{x}^{[2]} - \mathbf{t}^{[2]} \end{bmatrix} \tag{3}$$

where $^{+}$ denotes the Moore-Penrose pseudo-inverse. The required image position in the third view is therefore given by:

$$\mathbf{x}^{[3]} = M^{[2]}\mathbf{X} + \mathbf{t}^{[3]} = M^{[2]} \begin{bmatrix} M^{[1]} \\ M^{[2]} \end{bmatrix}^{+} \begin{bmatrix} \mathbf{x}^{[1]} - \mathbf{t}^{[1]} \\ \mathbf{x}^{[2]} - \mathbf{t}^{[2]} \end{bmatrix} + \mathbf{t}^{[3]} \tag{4}$$

2.2 Many point transfer

A weakness of the method outlined above is its reliance on the choice of four specific *basis* points, relative to which the affine projections are defined. Any errors in the image positions $\mathbf{x}$ will lead to inaccuracies in the projections. A better method which we have exploited in more recent work [6] makes use of all available points to obtain estimates of the affine projections $M^{[j]}$ and affine structure $\mathbf{X}_i$ which minimize the objective:

$$E = \sum_{j=1}^{m} \sum_{i=1}^{n} \| \mathbf{x}_i^{[j]} - M^{[j]}\mathbf{X}_i - \mathbf{t}^{[j]} \| \tag{5}$$

The factorization method of Tomasi and Kanade [11] is a key element of the improved algorithm, as follows.

The translational component of the projection in each of three views is given by the centroid of the corners, $\mathbf{t}^{[j]}$. Now consider the $2m \times n$ matrix in which each pair of rows consists of the image positions of the points in a particular view expressed relative to the centroid; i.e.

$$\tilde{W} = \begin{bmatrix} \mathbf{x}_1^{[1]} - \mathbf{t}^{[1]} & \ldots & \mathbf{x}_n^{[1]} - \mathbf{t}^{[1]} \\ \vdots & \vdots & \vdots \\ \mathbf{x}_1^{[m]} - \mathbf{t}^{[m]} & \ldots & \mathbf{x}_n^{[m]} - \mathbf{t}^{[m]} \end{bmatrix} \quad \text{where } \mathbf{t}^{[j]} = \sum_{i=1}^{n} \mathbf{x}_i^{[j]} \tag{6}$$

If $\tilde{W} = U\Sigma V^\top$ is the singular value decomposition of $\tilde{W}$ with ordered singular values $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$ such that $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$, and the columns of U are denoted by vectors $\mathbf{u}$ and the columns of V are denoted by vectors $\mathbf{v}$ then it can be shown [6, 12] that the objective E in equation 5 is minimized when:

$$[\mathbf{X}_1 \ldots \mathbf{X}_n] = \begin{bmatrix} \mathbf{v}_1^\top \\ \mathbf{v}_2^\top \\ \mathbf{v}_3^\top \end{bmatrix} \text{ and } \begin{bmatrix} M^{[1]} \\ . \\ M^{[m]} \end{bmatrix} = [\sigma_1\mathbf{u}_1 \;\; \sigma_2\mathbf{u}_2 \;\; \sigma_3\mathbf{u}_3] \tag{7}$$

If $m = 3$ then transfer of a point in the first two views to the third proceeds as in equation 4. Alternatively it is straightforward to extend equation 4 to the case where $m > 3$ by computing the structure $\mathbf{X}$ from more than two views, using least-squares on the $2m$ equations in 3 unknowns:

$$\mathbf{X} \approx \begin{bmatrix} M^{[1]} \\ . \\ M^{[m-1]} \end{bmatrix}^+ \begin{bmatrix} \mathbf{x}^{[1]} - \mathbf{t}^{[1]} \\ . \\ \mathbf{x}^{[m-1]} - \mathbf{t}^{[m-1]} \end{bmatrix} \tag{8}$$

3 Fixation using affine transfer

If we consider a moving cluster of features localized on the target using for example a simple corner detector and tracker, then the fixation or tracking problem is one of reliably describing the collective position of the feature cluster over time. Unfortunately all methods of feature point detection and tracking are prone to feature loss and reappearance through errors, noise and occlusion. The result of this, which we showed in [5], is that simplistic measures of target position such as the position of an individual feature or the centroid of all features are doomed to failure if the temporal stability of individual features cannot be guaranteed, which is the case for any real system. However the theory of affine transfer outlined in section 2 provides the perfect vehicle for obtaining a fixation point whose motion is consistent with the three-dimensional motion of the target features, even if the actual desired fixation point on the target is invisible.

3.1 Monocular fixation

To begin, we review the case of *monocular* fixation using affine transfer, which we addressed in [5, 6]. We then extend this to a binocular tracker which generates demands for the left and right vergence axes which are coupled through the use of a common object centred reference frame, ensuring that the trajectory of the fixation point in space (i.e. the intersection of the optic axes of the cameras) is consistent with the target trajectory.

Suppose that while tracking an object undergoing a linear transformation (more general than a rigid one), the fixation point was $\mathbf{g}^{[1]}$ in frame 1 and $\mathbf{g}^{[2]}$ in frame 2 a short time later. Treating the fixation point as a virtual 3D point on the target, we can compute its affine coordinates $\mathbf{X}_g$. Given temporal correspondences from the first two frames to a third, the image position of the fixation point in the new view can be found using the transfer equation 4 above, yielding a new desired fixation point $\mathbf{g}^{[3]}$ whose position relative to the 3D structure of the target is maintained

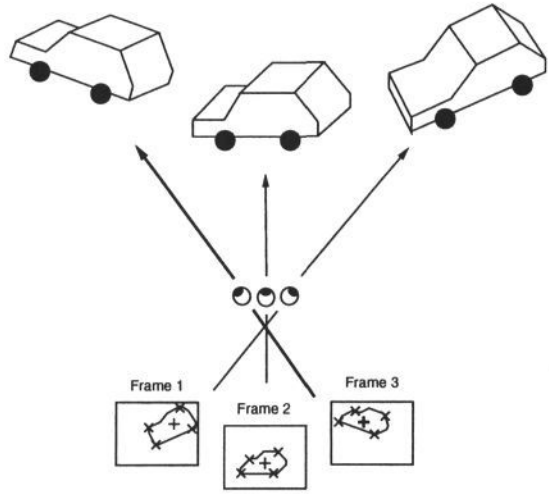

Figure 1: Transfer of the fixation point: point correspondences are marked with a $\times$ while the fixation point is marked by a $+$. The tracking goal requires that a new fixation point be found in the new frame. Affine transfer gives the fixation point in the third frame relative to the 3D positions of the matched features. The new fixation point is marked in bold.

over time. Neither $\mathbf{X}_g$, nor its projections $\mathbf{g}^{[1]}, \mathbf{g}^{[2]}, \mathbf{g}^{[3]}$ need correspond to a physical feature.

Therefore, with *any* four point correspondences (providing they are in general position) in three frames we can reconstruct the position of the desired fixation point given its image coordinates in the first two frames. Figure 1 illustrates the process. The (minimum) four points used need not be the same over time, there must merely be *one* set of point correspondences between each set of three consecutive frames. As the set used varies, we are effectively performing a change of basis relative to which world and image coordinates are computed. Unlike other tracking/fixation methods such as correlation matching, the method is viewpoint invariant, insensitive to local occlusions and/or individual corner dropout or reappearance, will work even if the fixation point in space is occluded, and is invariant to changes in camera parameters such as zooming or focussing since no camera calibration is required.

3.2 Binocular fixation

Whereas monocular cues are sufficient to allow the construction of an object centred reference frame and to obtain reliable fixation, there are the previously mentioned benefits to be gained through the use of stereo. Below we show how to apply the principles of affine transfer to the binocular tracking problem.

Consider a set of point features (minimum size four)[1] which have been matched temporally between frames 1 and 2 in both left and right images, and matched

[1] It is possible to reduce the number of matches required in all views by computing intermediate

spatially between the left and right frames; i.e. we have a set of correspondences $\{\mathbf{x}_i^{[1L]}, \mathbf{x}_i^{[2L]}, \mathbf{x}_i^{[1R]}, \mathbf{x}_i^{[2R]}\}$, $i = 1 \ldots n$.

The centroid of the points in each frame yields the translational component of the affine projection for that frame, and the SVD of the $8 \times n$ registered measurement matrix $\tilde{W}$ constructed from the correspondences then yields a set of affine projections:

$$\{\{M^{[1L]}, \mathbf{t}^{[1L]}\}, \{M^{[2L]}, \mathbf{t}^{[2L]}\}, \{M^{[1R]}, \mathbf{t}^{[1R]}\}, \{M^{[2R]}, \mathbf{t}^{[2R]}\}\}$$

Given the fixation points in frame 1, $\mathbf{g}^{[1L]}$ and $\mathbf{g}^{[1R]}$, we can compute $\mathbf{X}_g$, the coordinates of the virtual 3D fixation point and effect the transfer (see equation 4):

$$\mathbf{g}^{[2L]} = M^{[2L]} \begin{bmatrix} M^{[1L]} \\ M^{[1R]} \end{bmatrix}^+ \begin{bmatrix} \mathbf{g}^{[1L]} - \mathbf{t}^{[1L]} \\ \mathbf{g}^{[1R]} - \mathbf{t}^{[1R]} \end{bmatrix} + \mathbf{t}^{[2L]} \tag{9}$$

$$\mathbf{g}^{[2R]} = M^{[2R]} \begin{bmatrix} M^{[1L]} \\ M^{[1R]} \end{bmatrix}^+ \begin{bmatrix} \mathbf{g}^{[1L]} - \mathbf{t}^{[1L]} \\ \mathbf{g}^{[1R]} - \mathbf{t}^{[1R]} \end{bmatrix} + \mathbf{t}^{[2R]} \tag{10}$$

The matrix inversion involved in the pseudo-inverse calculation is a cheap operation since 3×3 inversion can be performed in approximately 35 flops. The costliest part of the computation is the singular value decomposition, which requires $O(n^2(m+n))$ flops. However there is no need to compute the full structure in order to effect transfer, thus the full SVD is not required. If $A = \tilde{W}\tilde{W}^\top$, then the eigen-decomposition of the 8×8 matrix A supplies the required projections:

$$\tilde{W} = U\Sigma V^\top \Rightarrow A = U\Sigma\Sigma U^\top = U\Lambda U^\top \tag{11}$$

where U is a matrix of eigenvectors $\mathbf{u}$ of A with corresponding eigenvalues $\sigma_1^2, \ldots, \sigma_4^2$. Thus the affine projections, obtained from the three most significant eigenvectors of A may be computed in $O(8^3) =$ constant time.

It is notable that in the stereo case corners need only be tracked temporally between two consecutive frames, not over three frames as was the case for the monocular fixation, meaning that the temporal stability of the corners is less crucial. Secondly, the extra views afforded by the second camera lead to better conditioning on the computations since 3D structure can be computed even for pure translations of the target.

4 Implementation

The algorithm outlined above has been implemented as the final stage of a stereo feature detection and tracking system running at 25Hz on a network of 10 T800 transputers. The network configuration is illustrated in figure 2. Tmax0 and Tmax1 are frame-grabbers whose role is to extract a foveal window from each image and to provide timing and odometry information for subsequent processing and control (details appear in [13]). Corner processors perform corner detection using the Wang/Brady algorithm [14]. The two Track processors do the temporal corner tracking on the left and right images respectively, and also compute monocular

structure, however this has not been attempted.

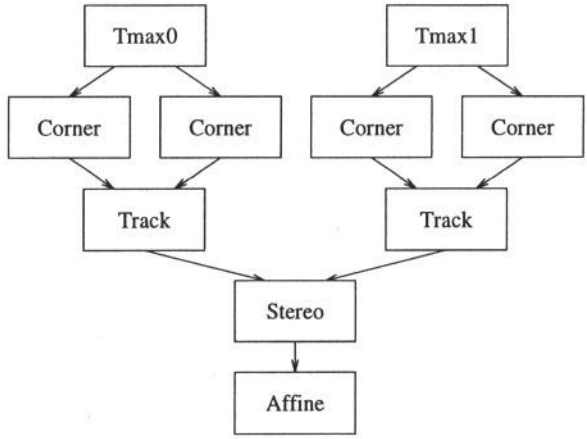

Figure 2: The transputer network used to implement the stereo affine transfer of the fixation point

fixation point transfer which is used as a backup in the event that stereo transfer fails. The Stereo processor does the stereo matching, and the Affine processor transfers the fixation point in left and right cameras using the algorithm of section 2.

4.1 Temporal and stereo tracking

The matching/tracking steps are covered in further detail in Reid and Murray [5, 6]. The strategy is to use an independent constant image velocity Kalman filter for each corner to predict the new position and covariance of previously matched corners. An appropriate search window is derived from the positional covariance and candidate corners which satisfy a cross-correlation threshold are ranked by closeness to the prediction, with the winner being used to update the Kalman filter.

Matches for previously unmatched corners are sought in a larger, fixed size region. If a match is found then a Kalman filter is initialised for that corner using the present and previous positions to provide position and velocity.

The stereo matching assumes that the cameras are already verged on the same area and that the epipolar geometry is approximately known. This enables the use of a simple search strategy again using correlation to test the quality of the matches: for every corner in the left image a search is made in the right image in a rectangular image region centred on the corner's left image position, the search region is defined so that only small vertical offsets (corresponding to deviation from a horizontal epipolar line) are allowed, but much larger horizontal offsets (corresponding to disparities and imprecise initial vergence on the object) can be accepted. Future work will relax the horizontal assumption and make use of the affine epipolar geometry, as outlined in section 6.

The best match over a threshold is accepted, if it conflicts with a previous match made, then the match with the strongest correlation is used.

4.2 Affine stereo transfer

The algorithm as described in section 2 requires that a minimum of four points be matched in all four frames (i.e. $1L$, $1R$, $2L$ and $2R$). If there are fewer than four matches over the four views then tracking reverts to the use of independent demands generated by the monocular trackers. Other strategies for coping with the occasional lack of data are the subject of current investigation.

When tracking is initiated, the choice of fixation point is somewhat arbitrary. Currently we use the centre of mass of the corners which have been matched in all four frames. Assuming the correspondences are indeed valid, this ensures that the 3D position of the fixation point is within the target. Subsequently, new fixation points in the left and right images are obtained via transfer of the old fixation point detailed in sections 2.2 and 3.2.

5 Results

We present samples from one real-time experiment. The sequence shows a remote controlled buggy being tracked using visual feedback from the algorithm to the platform, causing the cameras to follow the target. The sequence covers about a second of data processed in real time.

In each image the corners used in the affine transfer are shown in black while other stereo matches are shown as white circles. Temporal matches are shown as small crosses with their associated displacements represented by a scaled vector. Single small crosses are other corners detected but unmatched. The fixation point calculated by the stereo affine transfer is drawn as a large cross.

The stable demands generated by the algorithm are clearly demonstrated by the plot (figure 4) of the 3D trajectory of the tracked buggy calculated from the demands over several seconds. This is despite a number of incorrect matches.

6 Discussion

We have presented a new method for obtaining stereo fixation, which builds on our previous work using affine transfer as the fundamental construct for a monocular tracker. The virtual 3D coordinates of the fixation point on a target are computed from a stereo pair of images and then transferred to new left and right views using a set of features matched over time and between views. The transfer algorithm is constant time, and so the method is fast: it has been implemented in real-time on a network of transputers. In addition it retains all the benefits of the monocular tracker, namely viewpoint invariance, insensitivity to local occlusions and invariance to camera intrinsics.

The preliminary results presented indicate that the algorithm can provide a stable fixation point which is consistent between left and right views and over

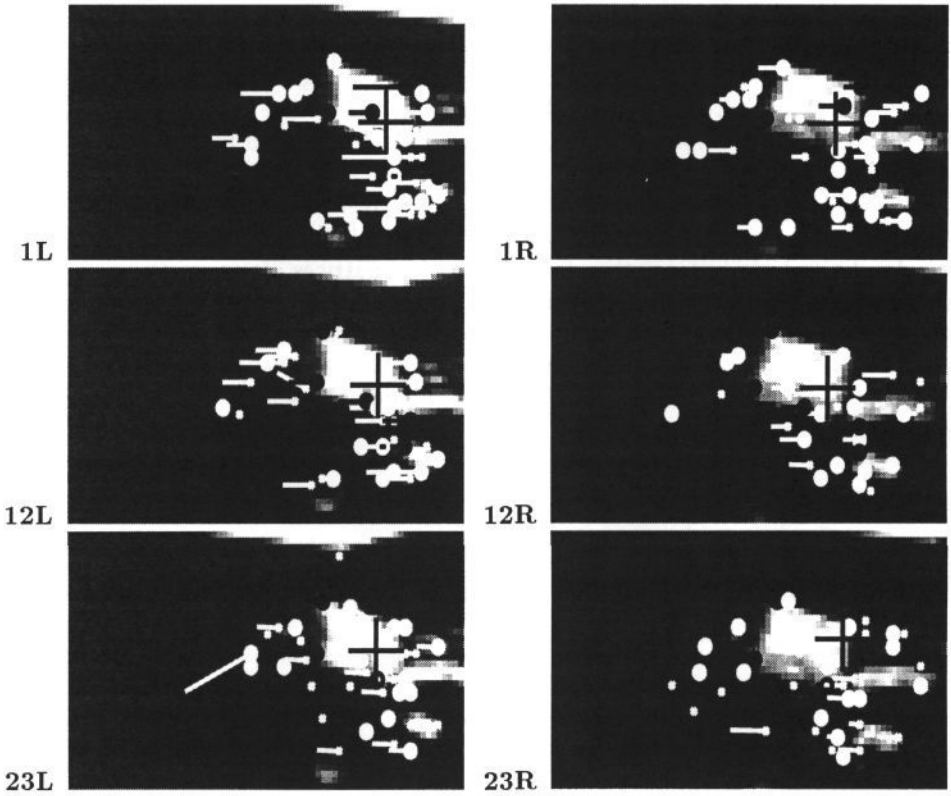

Figure 3: Sample foveal stereo pairs from a sequence during closed-loop tracking of the remote controlled buggy.

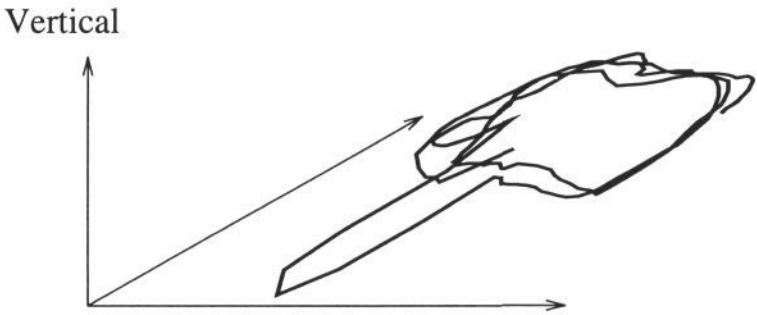

Figure 4: Plot of 3D trajectory of buggy tracked over several seconds, shown from the camera view point

time. However there exist a number of aspects of the implementation which have considerable room for improvement.

The cases of insufficient, erroneous or ill-conditioned data must be handled gracefully by the system if it is to maintain fixation for extended periods. The

monocular tracker described in [6] has a number of "backup" algorithms designed to cope with these eventualities, and similar strategies are being developed for the stereo system. Use of outlier rejection possibly by testing disparity gradient would go some way to addressing the problems of erroneous data.

Finally, further improvements to the temporal and stereo matching processes would help guarantee the volume and integrity of the data used. In particular the assumption of horizontal epipolars is overly restrictive in a verging system, and easily overcome. Shapiro *et al.* have shown [15] that affine epipolar lines can be derived easily from the affine projection matrices. It is worth noting too that the matching of previously unmatched corners in the *temporal* tracking could also be improved using affine epipolar geometry. A future development will be to calculate the epipolar lines between temporally consecutive frames from a set of temporally matched corners to be used to guide the search for candidates for unmatched corners.

References

[1] D. H. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

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

[3] J. J. Koenderink and A. J. van Doorn. Affine structure from motion. *J. Opt. Soc. Am. A*, 8(2):377–385, 1991.

[4] J. L. Mundy and A. P. Zisserman, editors. *Geometric Invariance in Computer Vision*. MIT Press, Cambridge, MA, 1992.

[5] I. D. Reid and D. W. Murray. Tracking foveated corner clusters using affine structure. In *Proc. 4th Int'l Conf. on Computer Vision, Berlin*, pages 76–83, Los Alamitos, CA, 1993. IEEE Computer Society Press.

[6] I. D. Reid and D. W. Murray. Active tracking of foveated feature clusters using affine structure. submitted to Int'l Journal of Computer Vision, March 1994.

[7] D. J. Coombs and C. M. Brown. Real-time binocular smooth pursuit. *International Journal of Computer Vision*, 11(2):147–164, October 1993.

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

[9] K. Pahlavan, J. O. Eklundh, and T. Uhlin. Integrating primary ocular processes. In G. Sandini, editor, *Proc. 2nd European Conf. on Computer Vision, Santa Margharita Ligure, Italy*, pages 526–541. Springer-Verlag, 1992.

[10] 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.

[11] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization approach. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[12] L.S. Shapiro. *Affine Analysis of Image Sequences*. PhD thesis, University of Oxford, 1993.

[13] 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(4):517–535, 1993.

[14] H. Wang and J. M. Brady. Corner detection for 3D vision using array processors. In *Proc. BARNAIMAGE-91, Barcelona*. Springer-Verlag, 1991.

[15] L.S. Shapiro, A. Zisserman, and J.M. Brady. Motion from point matches using affine epipolar geometry. to appear in ECCV '94.