Discrete-Time Rigid Motion Constrained Optical Flow Assuming Planar Structure

Jeffrey Mendelsohn
mendels@grip.cis.upenn.edu

Eero Simoncelli
eero@grip.cis.upenn.edu

Ruzena Bajcsy
bajcsy@central.cis.upenn.edu

GRASP Laboratory
Department of Computer and Information Science
University of Pennsylvania
Philadelphia PA 19104-6228

February 5, 1997

Abstract

An algorithm for estimating optical flow based on the discrete-time rigid motion of an uncalibrated camera through a static planar world is presented. The algorithm is based on differential measurements, and estimates are computed within a multi-scale decomposition and propagated from coarse to fine scales. The algorithm is designed to operate properly with large displacements (i.e., large velocities or low frame rates). The quality of the algorithm is demonstrated by comparing with a simple (smoothness-based) multi-scale optical flow estimator and an instantaneous-time direct method.

1. Introduction

Multi-scale algorithms for optical flow estimation allow the estimation of large image displacements and also improve overall accuracy of the flow [1, 5, 10, 14]. However, these techniques make an assumption (often implicitly) of smoothness on the optical flow field. In many situations, smoothness is not a realistic assumption. But there are often more realistic constraints available. The most well known is the assumption that the camera moves through a static scene. If n represents the number of pixels in the image, the basic optical flow problem has $2n$ degrees of freedom while the rigidity-constrained problem has, typically, $n+5+c$, where c represents the number of degrees of freedom from calibration.

Stronger assumptions may reduce the problem complexity even further. For many applications, the structure of the scene is well-approximated by a plane. Even when the scene structure is not planar, it is often useful to estimate a 'relevant' reference plane [9, 12]. The planar structure assumption greatly simplifies the problem, since the number of degrees of freedom no longer depends on the image size.

Given point and/or line correspondences, the discrete time motion problem has been solved by several authors [6, 7, 11, 13, 16]. For instantaneous representations, excellent work has been done in using multi-scale estimation techniques to couple the flow and motion estimation problems to provide a direct method

This document is GRASP Laboratory's Technical Report #410.
Research supported by ARO grant DAAH04-96-1-0007, ARPA grant N00014-92-J-1647, and NSF grants GER93-55018 and SBR89-20230.

[4, 8]. This method uses the multi-scale technique to capture large motions while significantly constraining the flow with a global model. However, this approach contains a hidden contradiction; the algorithm can observe large image motions but can only represent small camera motions due to its instantaneous time assumption.

This paper describes a global optical flow model for discrete camera motion through a static planar world. The algorithm is based on differential measurements, and estimates are computed within a multi-scale decomposition and propagated from coarse to fine scales. Comparisons between simple (smoothness-based) multi-scale optical flow, differential planar model, and two methods for estimating discrete planar models are shown.

2. Discrete-Time Optical Flow for a Planar Surface

The imaging system is assumed to use perspective projection, and is approximated using the following model (similar to the model used in [15]):

$$\mathbf{p}_i = \frac{1}{z_i}\mathbf{C}\mathbf{x}_i, \tag{1}$$

where $\mathbf{x}_i$ is the world coordinate vector and $\mathbf{p}_i$ the normalized image coordinate vector:

$$\mathbf{x}_i \equiv \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \qquad \mathbf{p}_i \equiv \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}.$$

The matrix $\mathbf{C}$ contains the camera calibration parameters:

$$\mathbf{C} \equiv \begin{bmatrix} fs & -f\cos(\theta) & c_u \\ 0 & f & c_v \\ 0 & 0 & 1 \end{bmatrix},$$

where f is the focal length, s a scale factor describing the ratio of vertical to horizontal pixel size, θ is an axis skew parameter (typically close to $\frac{\pi}{2}$), and (c_u, c_v) represents the image center.

For any reasonable system, the matrix $\mathbf{C}$ is invertible. Inverting equation (1) gives an expression for world coordinates in terms of the image coordinates and depth:

$$\mathbf{x}_i = \mathbf{C}^{-1}\mathbf{p}_i z_i. \tag{2}$$

The rigid-body motion constraint on world positions at two discrete times is expressed as:

$$\mathbf{x'}_i = \mathbf{R}\mathbf{x}_i + \mathbf{t},$$

where $\mathbf{R}$ is a three-dimensional (discrete-time) rotation matrix, and $\mathbf{t}$ an arbitrary three-dimensional translation vector.

An expression for the final position $\mathbf{p'}_i$ given calibration, motion, and structure parameters is derived below. Substituting the inverse projection relation of equation (2) into the rigid-body motion constraint:

$$\mathbf{C}^{-1}\mathbf{p'}_i z'_i = \mathbf{R}\mathbf{C}^{-1}\mathbf{p}_i z_i + \mathbf{t}.$$

This can be solved for the image position after the motion:

$$\mathbf{p'}_i = \frac{1}{z'_i}\left(\mathbf{C}\mathbf{R}\mathbf{C}^{-1}\mathbf{p}_i z_i + \mathbf{C}\mathbf{t}\right). \tag{3}$$

The planar assumption provides the following constraint:

$$\mathbf{a}^T \mathbf{x}_i = 1.$$

Combining with equation (2) gives:

$$\frac{1}{z_i} = \mathbf{a}^T \mathbf{C}^{-1} \mathbf{p}_i,$$

Finally, substituting into equation (3) yields an expression that provides the final image coordinates given the initial coordinates, the camera projection matrix, and the rigid-body motion parameters.

$$\mathbf{p}'_i = \frac{z_i}{z'_i} \mathbf{A} \mathbf{p}_i. \tag{4}$$

Where $\mathbf{A}$ is the matrix with arbitrary components defined by:

$$\mathbf{A} \equiv \mathbf{C} \left(\mathbf{R} + \mathbf{t} \mathbf{a}^T \right) \mathbf{C}^{-1}$$

This rigid-world based constraint on the motion of image points must be related to measurements of image displacements. Since differential optical flow techniques have proven to be quite robust [2], the estimator is based upon differential measurements. At each point in the image the differential form of the brightness constancy constraint is:

$$\frac{\partial I}{\partial u} \cdot \frac{\partial u_i}{\partial t} + \frac{\partial I}{\partial v} \cdot \frac{\partial v_i}{\partial t} + \frac{\partial I}{\partial t} = 0.$$

For the differential changes in image positions discrete displacements are substituted. Rewriting in vector notation:

$$\begin{bmatrix} \partial I/\partial u \\ \partial I/\partial v \\ \partial I/\partial t \end{bmatrix}^T \begin{bmatrix} u'_i - u_i \\ v'_i - v_i \\ 1 \end{bmatrix} = 0$$

Displacements can be assumed small since the algorithm is implemented in a multi-scale (coarse-to-fine) framework. In order to combine this equation with the constraint of equation (4), $\mathbf{p}'_i$ is isolated:

$$\begin{bmatrix} \partial I/\partial u \\ \partial I/\partial v \\ \partial I/\partial t \end{bmatrix}^T \begin{bmatrix} 1 & 0 & -u_i \\ 0 & 1 & -v_i \\ 0 & 0 & 1 \end{bmatrix} \mathbf{p}'_i = 0. \tag{5}$$

Assuming the value of $\frac{z'_i}{z_i}$ is well-defined and non-zero, the constraint of equation (5) can be combined with equation (4) to produce:

$$\begin{bmatrix} \partial I/\partial u \\ \partial I/\partial v \\ \partial I/\partial t \end{bmatrix}^T \begin{bmatrix} 1 & 0 & -u_i \\ 0 & 1 & -v_i \\ 0 & 0 & 1 \end{bmatrix} \mathbf{A} \mathbf{p}_i = 0.$$

This constraint is used to produce a local error metric:

$$E_i(\mathbf{A}) = \left(\mathbf{A} \mathbf{p}_i \right)^T \mathbf{D}_i \left(\mathbf{A} \mathbf{p}_i \right),$$

where $\mathbf{D}_i$ is a matrix constructed from the differential image measurements and known position values:

$$\mathbf{D}_i \equiv \begin{bmatrix} 1 & 0 & -u_i \\ 0 & 1 & -v_i \\ 0 & 0 & 1 \end{bmatrix}^T \left(\begin{bmatrix} \partial I/\partial u \\ \partial I/\partial v \\ \partial I/\partial t \end{bmatrix} \begin{bmatrix} \partial I/\partial u \\ \partial I/\partial v \\ \partial I/\partial t \end{bmatrix}^T \right) \begin{bmatrix} 1 & 0 & -u_i \\ 0 & 1 & -v_i \\ 0 & 0 & 1 \end{bmatrix}.$$

This leads to the global error metric:

$$E(\mathbf{A}) = \sum_i \mathbf{p}_i^T \mathbf{A}^T \mathbf{D}_i \mathbf{A} \mathbf{p}_i.$$

The minimization of this metric provides an estimate for $\mathbf{A}$ which can be used to produce the optical flow at each pixel by coupling equation (4) with the constraint that the third component of $\mathbf{p'}_i$ is one.

3. Removing the Implicit Weighting of Observations

The least-squares estimation of $\mathbf{A}$ uses an implicit weighting of the observations, since - by necessity - the factor of $\frac{z_i}{z'_i}$ is ignored. This can be approximately 'corrected' by iteratively re-estimating $\mathbf{A}$ using a weighting factor of magnitude $\frac{z_i}{z'_i}$ for each observation. Given an estimate of $\mathbf{A}$, an expression for the weighting factor comes from considering the third component of equation (4):

$$\frac{z_i}{z'_i} = \frac{1}{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T \mathbf{A}\mathbf{p}_i}.$$

For later reference, the algorithm incorporating these weights is called the 'normalized discrete algorithm'.

4. Multi-Scale Implementation

Since this method is capable of representing large (discrete) camera motions, it should be able to handle large image displacements. This is accomplished by implementing a coarse-to-fine version of the algorithm on a multi-scale decomposition.

First, a Gaussian pyramid on the pair of input frames [3] is constructed. At the coarsest scale of the pyramid, the algorithm is employed as derived. This provides an initial coarse estimate of optical flow. This optical flow is interpolated to give a finer resolution flow field, denoted $(\Delta^c u_i, \Delta^c v_i)$. This motion is removed from the finer scale images using warping; the warped images are denoted I^w.

The optical flow equation (5) is written only in terms of the final positions $\mathbf{p'}_i$. Thus a slightly modified constraint on the warped images may be used:

$$\begin{bmatrix} \partial I^w/\partial u \\ \partial I^w/\partial v \\ \partial I^w/\partial t \end{bmatrix}^T \begin{bmatrix} 1 & 0 & -(u_i + \Delta^c u_i) \\ 0 & 1 & -(v_i + \Delta^c v_i) \\ 0 & 0 & 1 \end{bmatrix} \mathbf{p}'_i = 0.$$

The remainder of the algorithm is as before: new matrices $\mathbf{D}_i$ are computed from this constraint and used to estimate $\mathbf{A}$.

5. Experimental Results

The following are comparative results obtained from the 'Yosemite' sequence[1]. The first method is a simple multi-scale optical flow (msof) algorithm [14]. The second is the well-known differential approximation

[1] This sequence was graphically rendered from an aerial photograph and range map by Lyn Quam at SRI.

(differential) to the rigid camera motion through a static planar world [4, 8]. The third is the normalized discrete (norm. discrete) version presented in this paper and the forth the non-normalized (discrete) version.

The metric presented is the RMSE of the error vector magnitudes. The true optical flow values were computed from the motion, structure, and calibration data provided with the Yosemite sequence. In particular, this means that the textureless top region of the image was assigned the flow appropriate to the rotation of a point at infinity. In order to obtain large motions, the computations on the sequence were subsampled at different temporal rates.

frame interval →	**1**	**2**	**3**	**4**
msof	0.68667	1.80693	3.62789	5.90103
differential	0.37816	0.99433	2.57761	5.33927
norm. discrete	0.44741	0.84904	1.25650	2.12269
discrete	0.45764	0.84916	1.24869	2.11088

Table 1. RMSE of flow error vector magnitudes, for four different algorithms and four different temporal subsampling factors. See text.

At the full temporal resolution, the differential algorithm gives the best performance for this sequence. However, it is clear that the discrete algorithm provides the best optical flow estimate for large motions. The relative performance of the normalized and non-normalized algorithms requires explanation. In the non-normalized algorithm, each observation is weighted by the ratio of its final depth to its initial depth. For the Yosemite sequence, all points are moving closer to the camera implying the ratio of final to initial depth is nearly one at distant points and the ratio decreases as initial depth decreases. By focusing the fit of the plane on the distal data a lower evaluation metric is obtained for the data with the lower temporal samplings. This is, primarily, a statement about the sequence and not the algorithms.

6. Conclusion

The derivation of a multi-scale algorithm for estimating optical flow based on an uncalibrated camera moving rigidly through a static planar world has been presented. Its implementation is only slightly more complicated and time consuming than the instantaneous motion model. In situations where the camera may be undergoing relatively large motions, the superiority of the discrete model has been demonstrated on the Yosemite sequence.

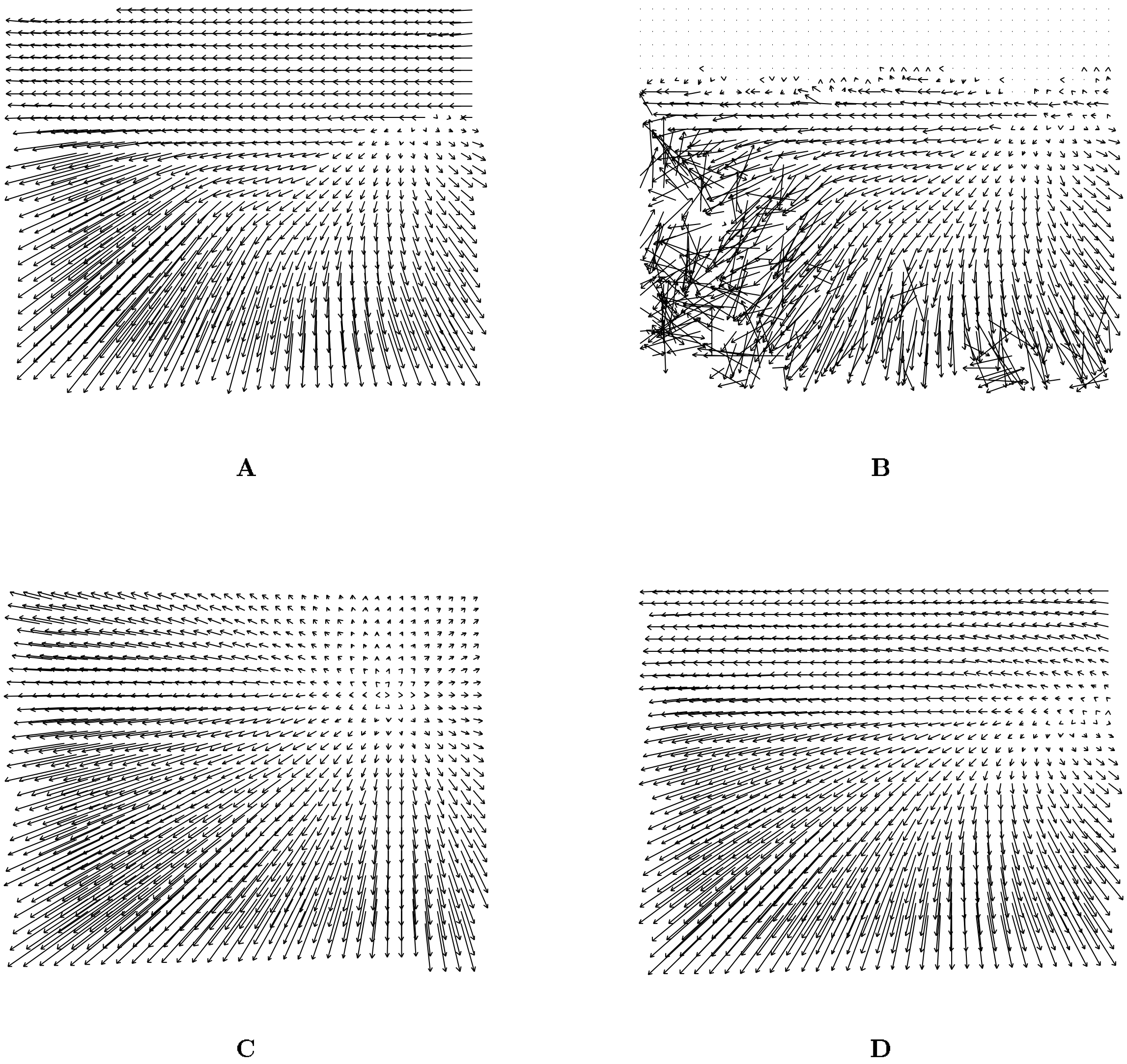

Figure 1. Optical flow fields for the Yosemite sequence. **A** True flow field. **B** Standard (smoothness-based) multi-scale differential flow. **C** Differential flow, using a planar world assumption. **D** Discrete-time flow, using a planar world assumption.

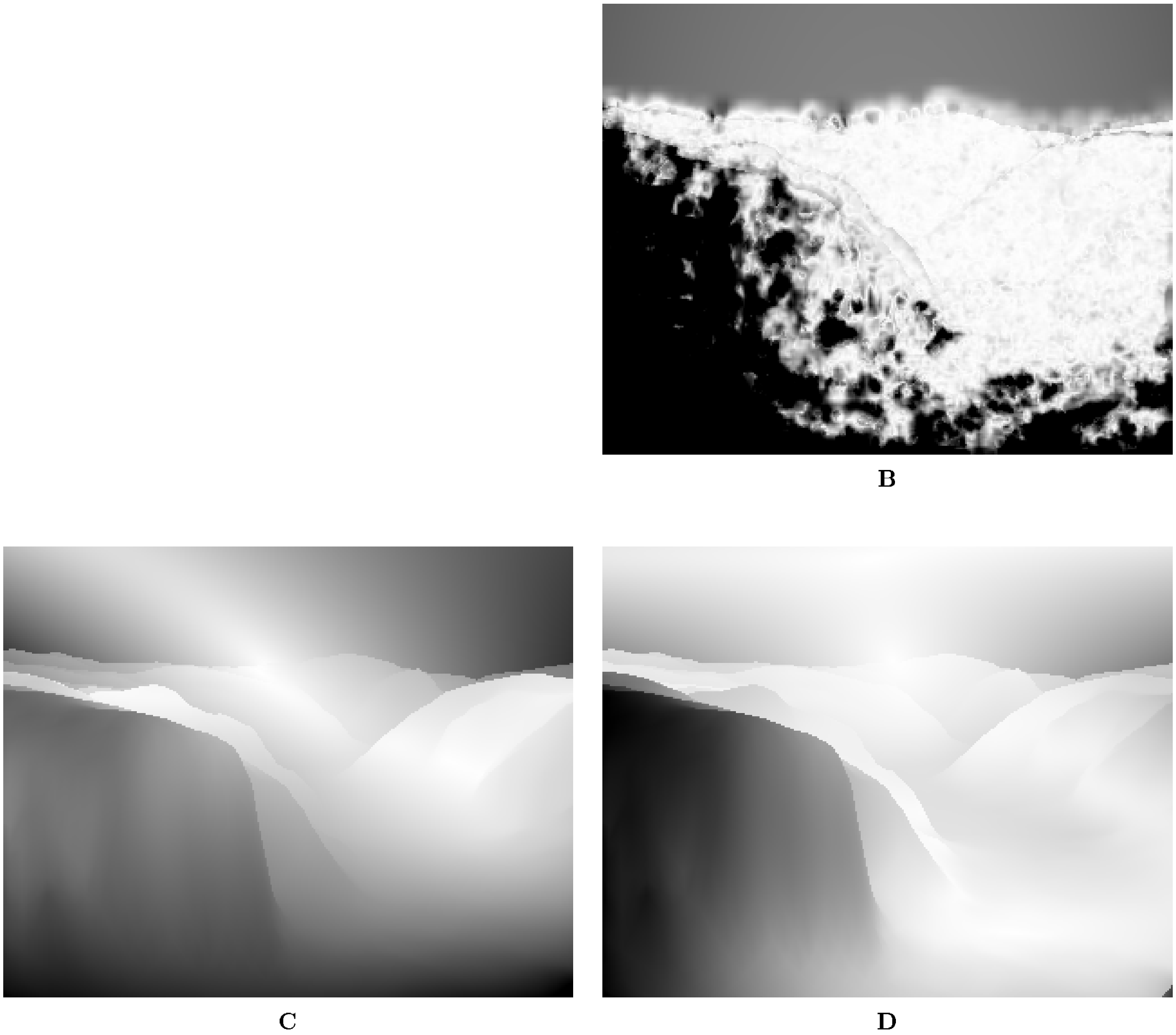

Figure 2. Error magnitudes for optical flow estimates on the 'Yosemite' sequence. Images are scaled so that white indicates no error and black corresponds to an error of magnitude five or more pixels. **B** Standard (smoothness-based) multi-scale differential flow. **C** Differential flow, using a planar world assumption. **B** Discrete-time flow, using a planar world assumption.

References

[1] P. Anandan. A Computational Framework and an Algorithm for the Measurement of Visual Motion. *International Journal of Computer Vision*, 2, 283-310, 1989.

[2] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, 1992.

[3] P. J. Burt. Fast filter transforms for image processing. *Computer Graphics and Image Processing*, 16, 20–51, 1981.

[4] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani. Hierarchical Model-Based Motion Estimation. *Proceedings* 2^{nd} *European Conference on Computer Vision-92*, Springer-Verlag, Santa Margherita Ligure, Italy, 1992.

[5] W. Enkelmann and H. Nagel. Investigation of Multigrid Algorithms for Estimation of Optical Flow Fields in Image Sequences. *Computer Vision Graphics Image Processing*, 43, 150-177, 1988.

[6] R. Hartley. Projective Reconstruction from Line Correspondences. *Proceedings of the CVPR'94 Conference*, 1994.

[7] R. Hartley, R. Gupta, and T. Chang. Stereo from Uncalibrated Cameras. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 1992.

[8] B.K.P. Horn. *Robot Vision.* MIT Press, Cambridge, MA, 1986.

[9] R. Kumar, P. Anandan, K. Hanna. Shape Recovery from Multiple Views: a Parallax Based Approach. *Proceedings of 1994 Image Understanding Workshop*, University of California at Riverside, Monterey, CA, 1994.

[10] M. Leuttgen, W. Karl, and A. Willsky. Efficient Multiscale Regularization with Applications to the Computation of Optical Flow. MIT Laboratory for Information and Decision Systems Technical Report LIDS-P-2115, 1992.

[11] Q.T. Luong and T. Vieville. Canonic Representation for the Geometry of Multiple Projective Views. *Proceedings* 3^{rd} *ECCV*, Stockholm, 1994.

[12] H.S. Sawhney. 3D Geometry from Planar Parallax. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, June 1994.

[13] A. Shashua and N. Navab. Relative Affine Structure: Theory and Application to 3D Reconstruction from Perspective Views. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, June 1994.

[14] E. P. Simoncelli, E. H. Adelson, and D. J. Heeger. Probability Distributions of Optical Flow. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, Mauii, June 1991.

[15] T. Vieville and O. D. Faugeras. The First Order Expansion of Motion Equations in the Uncalibrated Case. *Computer Vision and Image Understanding*, July 1996.

[16] T. Vieville, C. Zeller, and L. Robert. Using Collineations to Compute Motion and Structure in an Uncalibrated Image Sequence. *International Journal of Computer Vision*, 1995.