DETERMINATION OF EGO-MOTION FROM MATCHED POINTS

C G Harris
Plessey Research Roke Manor

ABSTRACT

We propose an algorithm for the estimation of the motion of a camera moving through a static environment (ie. the ego-motion) from matched points on two images. The algorithm correctly weights the observations by minimising point mis-match distances on the image-plane. Prior knowledge concerning the camera motion may also be included. The algorithm is iterative, but generally converges quickly. Results are both more accurate and more robust than closed-form solutions based on the 8-Point Algorithm [Longuet-Higgins 1981, Faugeras 1987].

INTRODUCTION

This paper concerns the motion part of the 'structure and motion' problem. By this we mean the determination of the camera motion parameters from examination of an image sequence of an otherwise static world. The camera motion parameters may be determined from calculations based on the flow of image detail across the image plane as the camera moves. Our approach is to extract a number of point-like features (such as corners or vertices) from the image that are considered to be consistent projections of the scene - that is, they are assumed to originate from existing 3D points. The correspondence of these points between sucessive images in the sequence gives the magnitude and direction of the flow of image detail at isolated points on the image plane. From this informtion, given a sufficient number of such correspondences, it is possible to deduce the camera motion parameters that gave rise to the flow, regardless of the particularities of the scene structure and illumination.

We concentrate on the recovery of the camera motion from an 'image sequence' of just two images and assume that the feature point extraction and correspondence processes have been succesfully executed. The problem is formulated as that of a camera moving through an otherwise static world. The more general motion determination problem concerning independently moving objects involves data partitioning, and, though closely related, is not here considered.

If the positions of feature-points are located without error, then each matched feature-point pair provides a single constraint on the camera motion parameters. If only image data is used, then only 5 of the camera motion parameters are determinable, due to the speed-scale ambiguity (but see below). Thus, using 5 feature-points, the motion parameters may in general be solved for exactly (though the solution may be non-unique). In real imagery, the feature-points will not be able to be located precisely, and this may produce large errors in the estimated motion parameters if the arrangement of feature-points in 3D leads to the problem being ill-conditioned. The use of more than 5 feature-points will generally make the problem better conditioned, but now we have an over-constrained set of equations, and this requires some form of weighting the relative importance of the satisfaction of the constraint equations.

In this paper we are particularly concerned with the correct treatment of the errors in the positioning of feature-points that are found in real data. The solution we seek is that which minimises the image-plane noise that must be invoked to explain the observed feature-point positions. This formulation of the problem has not been addressed by other researchers [Fang and Huang 1984, Faugeras 1987], whose methods have the attractions either of (relative) simplicity or of being closed-form.

THEORY

Let there be N pairs of points that have been matched between the two images. Consider the i'th such pair, being observed at image locations $\mathbf{r}_1$ and $\mathbf{r}_2$ respectively in the first and second images. Let $\mathbf{r}_2$ be the exact image projection of the causitive 3D feature - if you will, it is the point nominated to represent the particular grey-level patch that is the detected feature-point. To the observed point location on the first image, $\mathbf{r}_1$, is ascribed all the measurement error. For mathematical tractability, this error is assumed to be normally distributed with known covariance C (a 2x2 matrix). This covariance matrix provides the metric for comparing image-plane distances, and may be estimated from, say, the second-order terms in the expansion of the local auto-correlation function situated at the position of the feature-point. More cavalierly, it may just be set to the unit matrix, so giving a Euclidian metric. It is possible to show that an alternative formulation of the problem, that of assigning each of the observations a covariance of C/2, does not, in the limit of small camera motion, change the resulting ego-motion estimate.

Posit that the motion of the camera between capturing the two images was a translation $\mathbf{t}$, followed by a rotation θ about the camera pin-hole (ie. at the new camera position). Rotations are described by the vector θ, whose direction is the axis of rotation, and whose magnitude is the angle of rotation. We wish to obtain a quantitative value, E, of how well this posited camera motion can account for all N of the observerations, and use it to drive an iterative scheme to find optimum values of the camera motion variables.

AVC 1987 doi:10.5244/C.1.26

Since $\mathbf{r}_2$ is deemed to be exact, the causitive 3D point must lie precisely on the ray passing through $\mathbf{r}_2$ on the image-plane of the second camera position, through the location of the pin-hole of the second camera position, and thence out to infinity. The image of this ray in the first camera position will be a straight line starting at **s**, the projection of the second camera positions' pin-hole, and ending at **e**, the projection of the infinitly distant end of the ray (see Figure 1). Note that **s** depends only on the direction of translation of the camera motion, and not on the rotations, while **e** depends only on the rotations and is independant of the translations. Also, **s** will be common for each of the matched points, while **e** depends on $\mathbf{r}_2$, and hence will be different for each point (see Figure 2).

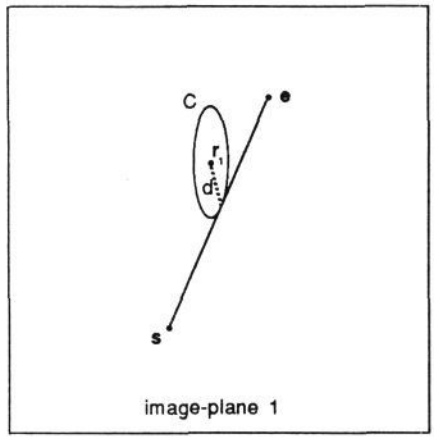

Figure 1 : The image on the first image-plane of the ray defined by $\mathbf{r}_2$. The covariance, C, of the observed point $\mathbf{r}_1$ is illustrated by the contour of constant metric distance, d.

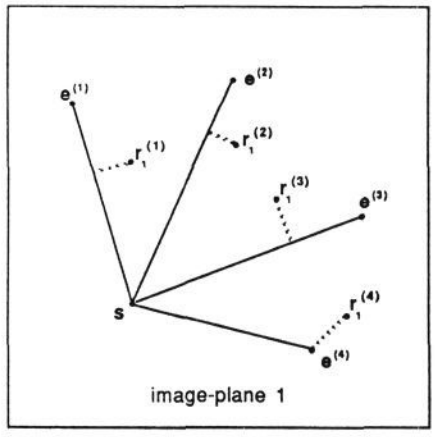

Figure 2 : The images of the set of rays on the first image-plane defined by the points $\mathbf{r}_2$, which are labelled by superscripts. For an Euclidian metric, the minimised distances are indicate the dashed lines.

The causitive 3D point must project in the first camera position to some point lying on the line (**s**,**e**). Since we have no prejudice as to the range of the point, we shall choose the position along the line (**s**,**e**) which has minimum distance to $\mathbf{r}_1$ in metric C. This corresponds to choosing the maximum likelihood range of the 3D point. If C represents a Euclidian metric, then in general the minimum distance, d, is found by dropping the perpendicular from $\mathbf{r}_1$ to the line (**s**,**e**). However, if $\mathbf{r}_1$ lies beyond either **s** or **e**, then d is the distance from $\mathbf{r}_1$ to the nearer terminator of the line (see, for example, point 4 in Figure 2). If C does not represent a Euclidian metric, then it must first be transformed into Euclidian form by a change of basis, so that the distance can be calculated as above.

Summing the squared distances for all N of the matched points gives the desired 'matching energy', $E(\mathbf{t},\theta)$, which is proportional to the negative logarithm of the joint probability of obtaining the observed point positions. To maximise the joint probability, the matching energy should be minimised with respect to the camera motion variables **t** and θ. This six (or five, but see below) dimensional minimisation is carried out by use of an iterative Newton scheme, which requires that the first and second differentials of E be calculated. These differentials are evaluated analytically by performing partial differentials with respect to the coordinates **s** and **e**, and leads to a relatively simple geometric interpretation.

Details of the evaluation of the differentials depend markedly on the variables used to describe the camera motion. Rotations we describe by the three cartesian coordinates of the rotation vector θ. Translations can be more problematical, as the speed-scale ambiguity dictates that only two independant variables can be solved for: the magnitude of the translation can always be traded-off against the range of the points. We need to describe the direction of translation using only two variables, but no coordinate system exists to do this which is free from singularities. This is overcome by changing variables as appropriate during the course of the minimising iteration. As an example, if the translational motion is primarily along the optical axis (the Z axis), then the two variables used to describe the translation are t_X/t_Z and t_Y/t_Z. Near the singularity at $t_Z=0$, the iterative minimisation fails.

If additional sources of knowledge about the camera motion exist, these can be included simply by adding extra terms to the energy, E. Examples of these knowledge sources are, for an arm-mounted camera, the joint angle sensors, and for a mobile robot, steering and odometry measurements, and spot distances to navigational beacons. Each of these measurements will be accompanied by an appropriate uncertainty. If these uncertainties are (or can be approximated by) a distribution function that is normal in the motion variables, then the additional terms to the energy will be quadratic, and hence result in easily calculated differentials. Most of the additional knowledge sources provide information to break to speed-scale ambiguity. If any of these are employed, then the translational components of the camera motion take on their full three degress of freedom, and these we assign to the three cartesian components of **t**.

In most practical applications, estimates of the camera motion are obtainable. These estimates generally act as strong regularisers in the determination of the ego-motion, especially when the camera field of view is narrow, or when the range of depths in the viewed scene is small. In addition, estimates of the camera motion are often essential in achieving the token matches in the first place.

Our minimisation uses an L2 error norm (ie. mean squares), and this is notoriously vulnerable to outliers, which can arise from mis-matched points, from obscuration of edges of different depths, and from non-stationary objects. This problem is overcome by use of a robust weighting scheme. The contribution of each matched pair of points to the energy (and to its differentials), is mediated by a weight whose value is a function of the points' matching distance in the previous cycle of the iteration, normalised by the mean matching distance of all the points. Thus, as convergence proceeds, outliers are discarded with increasing severity.

RESULTS

The ego-motion algorithm generally performs well on both real and computer generated data. The results depend on both the initial guess for the iterative minimisation, and the quality of the data. Often, a good initial guess will be available from the additional knowledge sources. If such a good initial guess is not available, then the iteration can wander off aimlessly, as is common with Newton minimisation. With a reasonably good initial guess, convergence to machine accuracy is generally achieved in 5 to 10 cycles.

With noiseless computer generated data, perfect results are, as expected, achieved. If the quality of the data is good (that is, if the noise is low, the translational motion sufficiently high, and the points adequately span the 3D space), then a unique minimum is found to exist, and is located close to the correct solution. With poor quality data, the energy minimum can split into multiple local minima, and extra local minima can appear elsewhere. Even so, the (lowest) minimum is generally closest to the correct solution.

To illustrate the results, test data was created for a simple model house (see Figure 3) containing 16 matched points. The house was situated some 15 units away from the camera, and filled the 53 degree field of view of the camera. Noise applied to the feature-points corresponded to either 1 or 2 pixels in a 512 pixel square image.

In Table 1 is shown the ego-motion estimates that result from analysing this data by a closed-form method [Faugeras, 1987] based on the 8-point algorithm [Longuit-Higgins 1981], and by our iterative method. The iterative technique is seen to give better results, but one particular trial has little statistical significance. A more revealing test is to examine the ego-motion results from an ensemble of trials generated with differing positional noise on the feature-point positions. This is shown in Figure 4, where 15 noise instantiations are solved for by the closed-form method (o's), and the iterative method (*'s). The abscissa depicts the difference in magnitude of the resulting rotation vectors from the exact value, and the ordinate similarly shows the translational errors. Thus correct ego-motion estimates would occur at the origin of the graph. The iterative method is seen to produce results typically a factor of 2 or 3 more accurate than the closed-form method. We have consistantly found this behaviour on a range of test data, and also on data from real images.

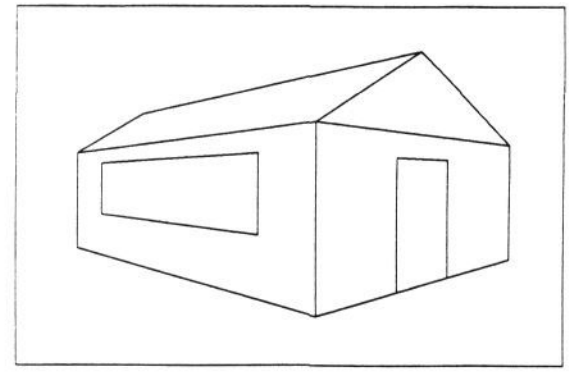

Figure 3 : Feature-points used to test the algorithm are the 16 edge vertices and corners on this model house.

Method	Noise (pixels)	Rotation (degrees) θ_x	θ_y	θ_z	Translation t_x	t_y	t_z
Exact	0	5	10	15	0.6	0	0.8
Closed-form	1	5.52	10.78	14.87	0.41	0.02	0.91
Iterative	1	4.99	10.17	14.97	0.50	-0.01	0.83
Closed-form	2	5.63	12.01	15.29	0.10	0.02	0.99
Iterative	2	5.14	10.18	14.88	0.53	0.02	0.85

TABLE 1 - Comparison of solutions

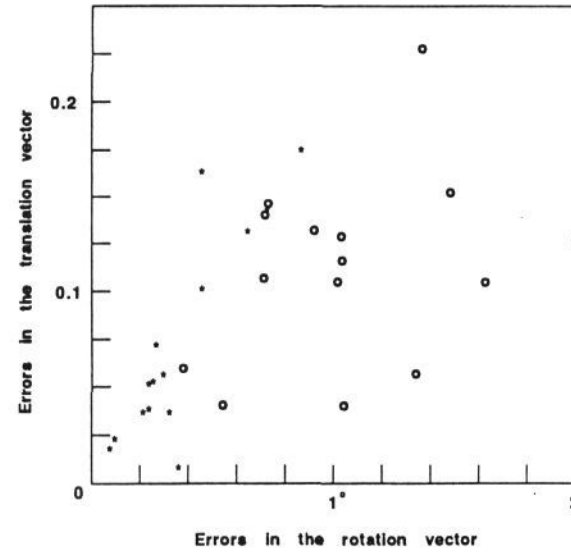

Figure 4 : The performance of the iterative algorithm (shown by *'s) and the closed-form algorithm (shown by o's) are compared for 15 cases of noise.

REFERENCES

Fang,J.Q. and T.S.Huang, *"Solving three-dimensional small rotation equations: Uniqueness, algorithms and numerical results,"* Computer Graphics, Vision and Image Processing, vol **26**, pp. 183-206, 1984.

Faugeras,O., F.Lustman and G.Toscani, *"Motion and Structure from Motion from Point and Line Matches,"* Proceedings IEEE International Conference on Computer Vision, pp. 25-34, 1987.

Longuet-Higgins,H.C *"A Computer Algorithm for Reconstructing a Scene from Two Projections,"* Nature **293**, pp. 133-135, 1981.