A ROBUST ALGORITHM FOR DETERMINING THE TRANSLATION OF A RIGIDLY MOVING SURFACE WITHOUT CORRESPONDENCE, FOR ROBOTICS APPLICATIONS.

Anup Basu
John (Yiannis) Aloimonos

Computer Vision Laboratory, Center for Automation Research
University of Maryland, College Park, MD 20742

ABSTRACT

A method is presented for the recovery of the three-dimensional translation of a rigidly translating object. The novelty of the method consists of the fact that four cameras are used in order to avoid point correspondences. The method is immune to low levels of noise and has good behavior when the noise increases.

1. Introduction

The potential of motion estimation in such applications as image coding, tracking and robot vision has long been appreciated and demonstrated. Up to now there have been three approaches to the solution of this problem:

(1) The first method assumes the dynamic image to be a three dimensional function of two spatial arguments and a temporal argument. Then, if this function is locally well behaved and its spatiotemporal gradients are computable, the image velocity or optical flow may be computed [I].

(2) The second method considers the cases where the motion is "large" and the previous technique is not applicable. In these instances the measurement technique relies upon isolating and tracking highlights or feature points in the image through time[2],

(3) In the third method, the three-dimensional motion parameters are directly computed from the spatial and temporal derivatives of the image intensity function. In other words, if / is the intensity function and (u,v) the optic flow at a point, then the equation $f_x u + f_y v + f_t = 0$ holds approximately [6,9].

As the problem has been formulated over the years one camera is used, and so the three-dimensional motion parameters that have to be and can be computed are five in number (two for the direction of translation and three for the rotation). In our approach, four cameras are used to recover the three translation parameters, instead of only the direction of translation.

2. Motivation and previous work

The basic motivation for this research is the fact that optical flow (or discrete displacements) fields produced from real images by existing techniques are corrupted by noise and are partially incorrect [5]. Most of the algorithms in the literature that use the retinal motion field to recover three-dimensional motion fail when the input (retinal motion) is noisy.

Some researchers have developed sets of nonlinear equations[7] with the three dimensional motion parameters as unknowns, which are solved by iteration and initial guessing. These methods are very sensitive to noise. Others, developed linear equations[8], but the sensitivity did not improve.

Several other authors use the optic flow field and its first and second spatial derivatives at corresponding points to obtain the motion parameters. But these derivatives seem to be unreliable in the presence of noise, and there is no known algorithm that can determine them reliably in real images.

Even if we had some way however to compute retinal motion in a reasonable fashion, with at most an error of 10% for example, all the algorithms proposed to date that use retinal motion as input (and one camera) would still produce non-robust results.

So, a natural question arises: is it possible to recover three dimensional motion from images without having to go through the very difficult correspondence problem? And if such a thing is possible, how immune to noise will the algorithm be? In this paper, as in [3, 9] we prove that if we combine stereo and motion in a certain way and we avoid any static or dynamic correspondence by using four cameras, then we can compute the three dimensional translation of a moving object.

3. Technical prerequisites

Consider a coordinate system $OXYZ$ fixed with respect to the camera, where O is the nodal point of the eye and the image plane is perpendicular to the Z axis, that is, pointing along the optical axis. Let us represent points on the image plane with small letters (x,y) and points in the world with capital letters (X,Y,Z). Let a point $P = (X_1,Y_1,Z_1)$ in the world have perspective image (x_1,y_1), where $x_1 = \frac{fX_1}{Z_1}$ and $y_1 = \frac{fY_1}{Z_1}$, with f the focal length (see Figure 1).

If the point P moves (translates) to $P' = (X_2,Y_2,Z_2)$ with

$$X_2 = X_1 + \Delta X \quad Y_2 = Y_1 + \Delta Y \quad Z_2 = Z_1 + \Delta Z$$

where $(\Delta X,\Delta Y,\Delta Z)$ is the three dimensional translational translation, and P' has the perspective image (x_2,y_2), then it can be easily shown that

$$x_2 - x_1 = \frac{f\Delta X - x_1\Delta Z}{Z_1 + \Delta Z} \qquad y_2 - y_1 = \frac{f\Delta Y - y_1\Delta Z}{Z_1 + \Delta Z}$$

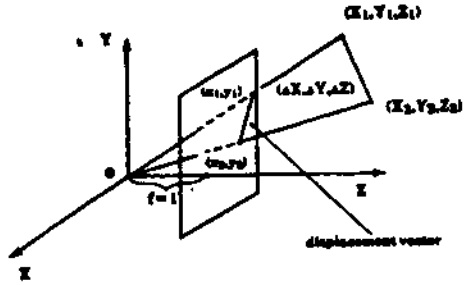

Figure 1

4. The model

Let $OXYZ$ be a Cartesian coordinate system, fixed with the Z-axis pointing along the optical axis, and consider the image plane plane Im_1 perpendicular to the Z-axis at a point $(0,0,f)$ (focal length=f). This is obviously the model of a camera. Furthermore, consider three more cameras with image planes Im_2, Im_3, Im_4 with nodal points $(\delta x, 0, 0)$, $(\delta x, \delta y, 0)$, and $(0, \delta y, 0)$ respectively (see Figure 2).

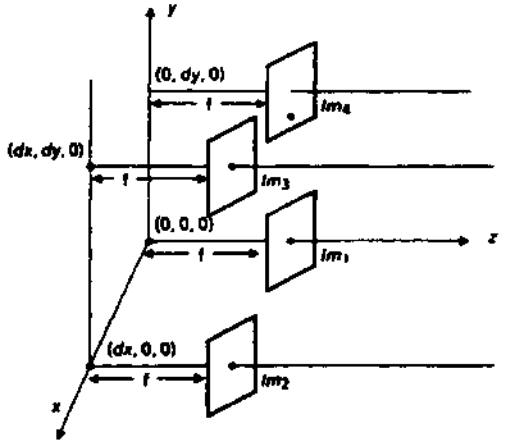

Figure 2 The Imaging (Four-Eye) System

On each of the image planes a coordinate system is defined exactly as it was done for Im_1. From now on, coordinates of three dimensional points will be denoted by (X,Y,Z), while coordinates of points in each of the images will be denoted by $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$, respectively. Coordinates of image points in the second dynamic frame (i.e. projections of three-dimensional points after the motion) will be denoted by the same symbols as before the motion, but primed (i.e. (x'_1, y'_1), etc.). Consider a set $A = \{(X_i, Y_i, Z_i) : i = 1,..,n\}$ of points in the world, which translates rigidly along the vector $(\Delta X, \Delta Y, \Delta Z)$ to form a new set $A' = \{(X'_i, Y'_i, Z'_i) : i = 1,..,n\}$, where $X'_i = X_i + \Delta X$, $Y'_i = Y_i + \Delta Y$ and $Z'_i = Z_i + \Delta Z$, for $i = 1,..,n$.

Let the projections of the set A on the four image planes be $\{(x_{1i}, y_{1i}), i = 1,...,n\}$, $\{(x_{2i}, y_{2i}), i = 1,...,n\}$, $\{(x_{3i}, y_{3i}), i = 1,...,n\}$, $\{(x_{4i}, y_{4i}), i = 1,...,n\}$, respectively, and the projections of the set A' be $\{(x'_{1i}, y'_{1i}), i = 1,...,n\}$, $\{(x'_{2i}, y'_{2i}), i = 1,...,n\}$, $\{(x'_{3i}, y'_{3i}), i = 1,...,n\}$, $\{(x'_{4i}, y'_{4i}), i = 1,...,n\}$ respectively (see Figure 3).

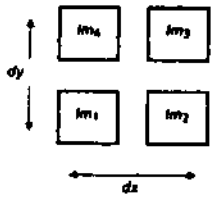

Figure 3 Orthographic Projection of the System on the Plane YZ

We now prove the following propositions.

Proposition 1: Using the aforementioned nomenclature, the quantity

$$\sum_{i=1}^{n} \frac{1}{Z_i}$$

is directly computable from the projections of the points of the set A on Im_1 and Im_2.

Proof: Consider a point $P = (X_i, Y_i, Z_i) \in A$ and its projections $P_1 = (x_{1i}, y_{1i})$ and $P_2 = (x_{2i}, y_{2i})$ on Im_1 and Im_2 respectively. Then

$$\frac{1}{Z_i} = \frac{x_{1i} - x_{2i}}{f\delta x} \tag{4.1.1}$$

Therefore,

$$\sum_{i=1}^{n} \frac{1}{Z_i} = \frac{1}{f\delta x}\left(\sum_{i=1}^{n} x_{1i} - \sum_{i=1}^{n} x_{2i}\right) \quad q.e.d.$$

Corollary 1: The quantity $\sum_{i=1}^{n} \frac{1}{Z'_i}$ is also directly computable from the projections of the set A' on Im_1 and Im_2.

Proposition 2: The quantity $\sum_{i=1}^{n} \frac{y_{1i}}{Z_i}$ is directly computable from the projections of A on Im_1 and Im_2.

Proof: From equation (4.1.1) we have

$$\sum_{i=1}^{n} \frac{y_{1i}}{Z_i} = \frac{1}{f\delta x}\left(\sum_{i=1}^{n} y_{1i}x_{1i} - \sum_{i=1}^{n} y_{1i}x_{2i}\right)$$

But corresponding points in Im_1 and Im_2 have the same y coordinates, and so

$$\sum_{i=1}^{n} \frac{y_{1i}}{Z_i} = \frac{1}{f\delta x}\left(\sum_{i=1}^{n} y_{1i}x_{1i} - \sum_{i=1}^{n} y_{2i}x_{2i}\right) \tag{4.3.1}$$

Equation (4.3.1) proves Proposition 2.

Proposition 3: The quantity $\sum_{i=1}^{n} \frac{x_{1i}}{Z_i}$, is directly computable from the projections of the set A on Im_1 and Im_4.

Proof: Similar to proposition 2.

5. Recovering three dimensional translation without correspondence

Consider the projections of the sets A and A' on Im_1. From Section 3 we have

$$x'_{1i} - x_{1i} = \frac{f\Delta X - x_{1i}\Delta Z}{Z'_i} \tag{5.1}$$

$$y'_{1i} - y_{1i} = \frac{f\Delta Y - y_{1i}\Delta Z}{Z'_i} \qquad (5.2)$$

If we write equation (5.1) for all pairs of corresponding points and we sum up these equations, we get

$$\sum_{i=1}^{n} x'_{1i} - \sum_{i=1}^{n} x_{1i} = f\Delta X \sum_{i=1}^{n} \frac{1}{Z'_i} - \Delta Z \sum_{i=1}^{n} \frac{x_{1i}}{Z'_i} \qquad (5.3)$$

Assuming that the motion in depth is small with respect to the depth, (5.3) can be approximated by

$$\sum_{i=1}^{n} x'_{1i} - \sum_{i=1}^{n} x_{1i} = \Delta X (f \sum_{i=1}^{n} \frac{1}{Z'_i}) - \Delta Z \left(\sum_{i=1}^{n} \frac{x_{1i}}{Z_i} \right) \qquad (5.4)$$

Similarly, with equation (5.2) we obtain

$$\sum_{i=1}^{n} y'_{1i} - \sum_{i=1}^{n} y_{1i} = \Delta Y \left(f \sum_{i=1}^{n} \frac{1}{Z'_i} \right) - \Delta Z \left(\sum_{i=1}^{n} \frac{y_{1i}}{Z_i} \right) \qquad (5.5)$$

If we apply the same procedure for the projections of the sets A and A' on Im_2 we get two more equations. One of them is the same as (5.5) and the other is

$$\sum_{i=1}^{n} x'_{2i} - \sum_{i=1}^{n} x_{2i} = \Delta X \left(f \sum \frac{1}{Z'_i} \right) - \Delta Z \left(\sum \frac{x_{2i}}{Z_i} \right) \qquad (5.6)$$

Equations (5.4) through (5.6) constitute a linear system in the unknowns $\Delta X, \Delta Y, \Delta Z$ (we will call this system Σ from now on) which always has a unique solution, given by

$$\overline{\Delta Z} = \frac{\left(\sum_{i=1}^{n} x'_{2i} - \sum_{i=1}^{n} x'_{1i} \right) - \left(\sum_{i=1}^{n} x_{2i} - \sum_{i=1}^{n} x_{1i} \right)}{\sum_{i=1}^{n} \frac{x_{1i}}{Z_i} - \sum_{i=1}^{n} \frac{x_{2i}}{Z_i}} \qquad (5.7)$$

$$\overline{\Delta X} = \frac{\sum_{i=1}^{n} x'_{1i} - \sum_{i=1}^{n} x_{1i} + \overline{\Delta Z} \left(\sum_{i=1}^{n} \frac{x_{1i}}{Z_i} \right)}{f \sum_{i=1}^{n} \frac{1}{Z'_i}} \qquad (5.8)$$

$$\overline{\Delta Y} = \frac{\sum_{i=1}^{n} y'_{1i} - \sum_{i=1}^{n} y_{1i} + \overline{\Delta Z} \left(\sum_{i=1}^{n} \frac{y_{1i}}{Z_i} \right)}{f \sum_{i=1}^{n} \frac{1}{Z_i}} \qquad (5.9)$$

Note that the denominators in the expressions (5.7) through (5.9) are always different from zero (for $\delta x\, \delta y \neq 0$).

We now proceed with an error analysis and implementation issues.

6. Theoretical error analysis

6.1. Error due to assumptions in the development of the constraints

Equation (5.3) is exact, but equation (5.4), used in the subsequent analysis, is an approximation of equation (5.3). It seems that the approximation used is

$$Z'_i = Z_i$$

But this is not the case, as we show in the sequel. Let the three-dimensional points be

(X_i, Y_i, Z_i), $i = 1,\ldots,n$ *then*

$x_{1i} = \frac{fX_i}{Z_i}$ and $x'_{1i} = \frac{f(X_i + \Delta X)}{Z_i + \Delta Z}$ *gives*

$$\sum_{i=1}^{n} (x'_{1i} - x_{1i}) = f \sum_{i=1}^{n} \left(\frac{X_i}{Z_i + \Delta Z} - \frac{X_i}{Z_i} \right) + f\Delta X \sum \frac{1}{Z'_i}$$

But we can write $\frac{X_i}{Z_i + \Delta Z}$ as $\frac{X_i}{Z_i} \left(1 + \frac{\Delta Z}{Z_i} \right)^{-1}$, or

$$\frac{X_i}{Z_i + \Delta Z} = \frac{X_i}{Z_i} \left(1 - \frac{\Delta Z}{Z_i} + O\left(\left(\frac{\Delta Z}{Z_i} \right)^2 \right) \right) \text{ giving}$$

$$\sum_{i=1}^{n} (x'_{1i} - x_{1i}) = f\Delta X \sum_{i=1}^{n} \frac{1}{Z'_i} - \Delta Z \sum_{i=1}^{n} \frac{x_{1i}}{Z_i} + O(d^{-2})$$

where $d = \min_i \left(\frac{Z_i}{\Delta Z} \right)$.

A similar analysis can be carried out for the computation of

$$\sum_{i=1}^{n} y'_{1i} - \sum_{i=1}^{n} y_{1i}$$

The above equation (which is exact) becomes equation (5.4) (which is used in the algorithm) neglecting $O(d^{-2})$ terms, hence the accuracy of the algorithm depends on the assumptions that $\left(\frac{\Delta Z}{Z_i} \right)^2$ is negligible.

6.2. Error due to small perturbation of the points

It is observed that when random noise is added to the image points, the equations remain consistent [4] (details omitted due to lack of space.)

6.3. Error due to the addition and deletion of random noise points

Unlike random perturbation of points, throwing in points at random makes the estimators based on the image points inconsistent[4]. However, from actual experiments it is observed that the effect of this error is not significant if we consider a window around the object of interest.

6.4. Stability of the solution of the linear system Σ

From our analysis[4], the necessary and sufficient condition for the system not to be critically ill-conditioned (and therefore for its solution to be stable) is

$$\sum_{i=1}^{n} \sum_{j=1}^{n} |b_{ji}| \epsilon_{ij} < 1$$

where $B = (b_{ij}) = \Sigma^{-1}$.

It is worth stating at this point that discretization noise is enough to destroy many algorithms in the literature that do dynamic analysis (measurement of motion) [8]. However, we have the following proposition.

Proposition 4: A sufficient condition for critical ill-conditioning not occuring in our model under discretization error is

$$\delta x > HM(Z)$$

where HM(Z) denotes the harmonic mean of the depths of the world points corresponding to the image points.

Proof: For proof see [4], (deleted here for lack of space).

7. Experiments

Here we only describe one experiment with real images. Note that we have eight frames in all, four before the motion and four after the motion. When we say that we have an error of B% in the translation, we mean

$$\beta = 100\frac{1}{3}\left(\frac{|\Delta X - \Delta \bar{X}|}{|\Delta X|} + \frac{|\Delta Y - \Delta \bar{Y}|}{|\Delta Y|} + \frac{|\Delta Z - \Delta \bar{Z}|}{|\Delta Z|}\right)$$

where $(\Delta X, \Delta Y, \Delta Z)$ is the actual translation (with $(\Delta X \Delta Y \Delta Z \neq 0)$, and $(\Delta \bar{X}, \Delta \bar{Y}, \Delta \bar{Z} \neq 0)$ is the computed one.

The experiments were carried out using images of a circuit board. Image acquisition was accurately controlled using a "American Robot" arm and a VICOM processor. Figure 4 shows the result of the point extraction operator on the images obtained after motion. The number of points extracted in the four frames were respectively 1767, 1643, 1665, 1687 before motion and 1491, 1547, 1578, 1529 after motion. The actual motion was (60.0, -60.0, -30.0) and the estimated translation was (63.5, -63.1, -35.2). The error is due to the factors that were explained in the paper and to the fact that our actual measurements (ground truth) were not perfect (cameras set up, calibration and motion).

Due to lack of space several proofs and experiments were omitted. The interested reader is refered to [4].

References

1. Horn, B.K.P. and Schunck, B.G., "Determining optical flow", *Artificial Intelligence,* 17, 185-204, 1981.

2. Ullman, S., "The interpretation of visual motion", Ph.D. thesis, MIT, 1977.

3. Aloimonos, J., Basu, A., "Shape and motion from contour without correspondence: general principles**, *Proc. IEEE conf. on CVPR,* Miami, Florida, 1986.

4. Basu , A. and Aloimonos, J., "A robust algorithm for determining the translation of a rigidly moving surface without correspondence", CAR-TR-279, Computer Vision Laboratory, University of Maryland, 1987.

5. Ullman, S., "Analysis of visual motion by biological and computer systems", *IEEE Computer,* 14 (8), 57-69, 1981.

6. Aloimonos, J. and Brown, CM., "The relationship between optic flow and surface orientation", *Proc. 7th ICPR,* Montreal, Canada, 1984.

7. Longuet-Higgins, C. and Prazdny, K., "The interpretation of a moving retinal image", *Proc. Royal Society of London,* B208, 385-397, 1980.

8. Tsai, R.Y. and Huang, T.S., "Uniqueness and estimation of three dimensional motion parameters of rigid objects", in S. Ullman and W. Richards (Eds.), *Image Understanding 1984,* New Jersey: Ablex Publishing Co., 1984.

9. Aloimonos, J., "Low level visual computations", Ph.D. thesis, Dept. of Computer Science, University of Rochester, August 1986.

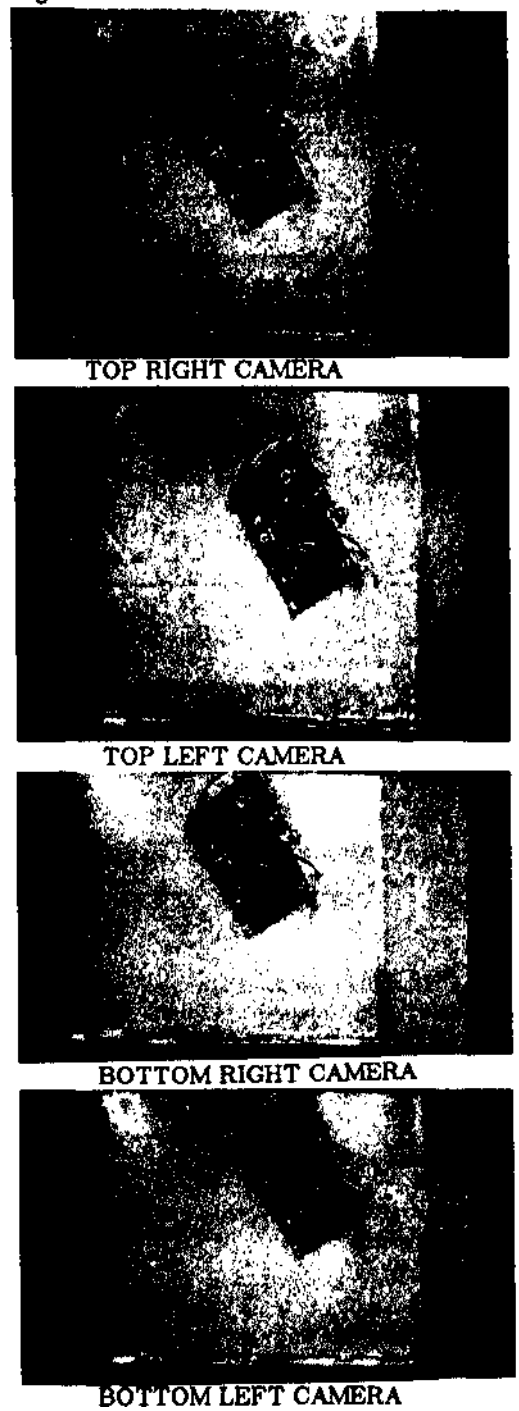

Figure 4,