Optical Navigation by the Method of Differences

Bruce D. Lucas and Takeo Kanadc

Computer Science Department
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract. The *method of differences* refers to a technique for image matching that uses the intensity gradient of the image to iteratively improve the match between the two images. Used in an iterative scheme combined with image smoothing, the method exhibits good accuracy and a wide convergence range. In this paper we show how the technique ran be used to directly solve for the parameters relating two cameras viewing the same scene. The resulting algorithm can be used for optical navigation, which has applications in robot arm guidance and autonomous roving vehicle navigation. Because of the regular structure of the algorithm, the prospects of carrying it out with special-purpose hardware for real-time control of a robot seem good. We present experimental results demonstrating the accuracy and range of convergence that can be expected from the algorithm.

1. Introduction

Optical navigation refers to the determination of the position and orientation of a camera analysis of the picture taken by the camera. The objective of such analysis is to determine some or all of the six parameters (three of position and three of orientation) that determine the position of that camera relative to some fixed frame of reference. In our method and in many others the fixed frame of reference is that of a second camera, so that the problem is that of image comparison.

Optical navigation has a number of applications in robotic tasks that require a knowledge of the position and orientation of the robot. This is because mechanical imperfections and environmental uncertainty make it impossible to know exactly how a robot will move in response to the commands sent to it and exactly what it will encounter in its surroundings. Such applications include navigation of autonomous roving vehicles and navigation of a robot arm relative to the object on which it is performing its task.

The approaches to matching for optical navigation may be divided into three categories; sparse two-dimensional matching, continuous two-dimensional matching, and three-dimensional matching. The sparse two-dimensional approach starts with a discrete set of matching points in the two images, and from them deduces the camera motion. The question of how many points are necessary to uniquely solve for the camera parameters has been addressed by Tsai & Huang (1981). With more points, the problem is overspecihed and a least squares approach is required (Gennery, 1980). The continuous two-dimensional matching approach starts with a whole image field of matches (the "optical flow field"); Brass &' Horn (1983) have shown how how to determine the camera motion from the optical flow field, again using a least-squares formulation. Obtaining the optical flow field has been investigated by, for example, Horn & Schunck (1981) and Cornelius *k* Kanade (1983), among others. In the three-dimensional matching approach, corresponding points in three dimensions (obtained e.g. by stereo) are used to determine the camera motion; this technique was used by Moraver (1980) to navigate a rover.

These approaches all split the process into two steps: finding the matches and using those matches to solve for the camera parameters. In this paper we show how to combine the two steps into one, by applying a generalized image matching technique that we term the *method of differences.* The method of differences directly computes the six camera parameters, or any desired subset of them, much as standard matching techniques compute two parameters (the :r and i/ displacements). That is, the camera parameters are explicitly included in the matching process. The method takes advantage of the fact that, in many applications the approximate position and orientation of the camera are known. Starting from that estimate we compute a better estimate by using the image intensity gradient as a guide. By using an iterative scheme our estimates converge to the correct value. The result is a technique that is fast and free of search.

In the remainder of the paper, we first describe the method of differences in a one-dimensional case, which serves to illustrate many of the issues. Then we show how the same technique can be used for multi-parameter estimation. Finally, we present some experimental results and draw some conclusions.

Tin,- research was sponsoied by the Defense Advanced Research IV.nrrls A•••!»< v (|)OI>). AKI'A Order No ;:.V.)7. monitored by the An tore,. Aviom^ Laboratory Lndei Cont ra< t f .L'O I f>-8 I-K-1 VJO. The views .iiid eoiielusioiis contained m this doriiment are those of the ,-nit|i<ii> and should not be interpreted as representing the olli-(la! policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US (Government.

2. The technique

Parameter estimation by the method of differences. The one-dimensional case illustrates the nature of the technique. Given two one-dimensional images $I_1(x)$ and $I_2(x)$ related by a translation, so that $I_1(x) = I_2(x + h)$, we wish to estimate the translation h. We do this by finding that $\hat{h}$ that minimizes the total squared error,

$$E = \sum_x \left(I_2(x + \hat{h}) - I_1(x)\right)^2. \tag{1}$$

Since we want a local, non-searching algorithm, we approximate $I_2(x + \hat{h})$ using $I_2(x)$ on the basis of local information, namely the derivative; this yields the approximation

$$E \approx \sum_x \left(I_2(x) + \hat{h} D_x I_2(x) - I_1(x)\right)^2, \tag{2}$$

where D_x denotes partial differentiation with respect to x. This equation is quadratic in $\hat{h}$, so we can differentiate with respect to $\hat{h}$, set equal to zero, and solve the resulting linear equation for $\hat{h}$, obtaining

$$\hat{h} = \frac{\sum_x \left(I_1(x) - I_2(x)\right) D_x I_2(x)}{\sum_x D_x I_2(x)^2}. \tag{3}$$

We call this the method of differences because it is based on comparing the difference between the images, $I_1(x) - I_2(x)$, with the derivative $D_x I_2(x)$ (which will in fact be implemented as a difference), to obtain an estimate for the parameter h.

We have shown elsewhere (Lucas, 1985) how this method is easily extended to multi-parameter estimation, as required for navigation. Briefly, the scalar disparity *h* is replaced by a vector of camera parameters; the derivatives become gradients, and the division becomes a matrix inversion. The stability of the matrix inversion is investigated in the work cited above, with the conclusion that the matching points should be well-distributed in three-space to guarantee good numerical accuracy.

Iteration and smoothing. Two modifications are required to make the method work. First, because the method yields only an approximation *h* to the disparity h, we must use an iterative scheme to obtain an accurate result. The idea is to calculate an estimated disparity, move I_2 by that amount, and calculate again.

Second, to improve the accuracy and range of validity of the linear estimate used in (2), we must smooth the image. This can be thought of as smoothing out purely local bumps and wrinkles in the image intensity profile that would make a linear estimate accurate only over a small range. This can be made more precise by a Fourier analysis of (3); this shows that removing the high frequency components of the image by smoothing does indeed extend the range of convergence, in rough proportion to the size of the smoothing window (Lucas, 1985). This is because convergence to the correct value with an image consisting of a pure sine wave is possible only for disparities up to one-half the wavelength of the sine wave; for larger disparities, the algorithm will converge to the wrong value.

Since smoothing the image also reduces the accuracy of the method, it is necessary to use an iterative approach in which each successive step uses a less smoothed image, in a sort of coarse fine approach. This allows the algorithm to tolerate a large disparity yet yield an accurate answer.

3. Experimental results

Our experimental data consisted of three views of the same scene taken by a camera mounted on the Stanford cart (Moravec, 1980); they are shown in Figure 1. The camera was mounted on a slider, so we had accurate knowledge of the relative positions of the cameras. The three views were pictures taken by the camera at the left, middle, and right slider positions, with 26 cm separating each position. The left picture was used as the reference image, and a number of points p were selected from this image as reference points. These points correspond to the points *x* that the sum in (I) runs over. 'Then the right picture was used as the second image of a stereo pair to obtain (essentially by hand) the distances -(p) of the reference points p. The method of differences was then used to determine position of the middle camera. Since the exact position of the middle camera was known, we could assess the accuracy of the method. Moreover, we could determine the range of convergence by varying the initial estimate of the middle camera's position around the correct value.

Convergence range. The convergence range for both the one-dimensional case and the multi-dimensional case was investigated using these pictures. As predicted, the convergence ranged was found to increase in rough proportion to the size of the smoothing window. The range for *x* and *y* motions was roughly ±1 meter, and somewhat more in the *z* direction. The range for pan and tilt was approximately i 10 degrees, and about .1.30 degrees for roll. Except for roll, these parameters are limited more by the angle of view of the camera than by the technique. For example, no matching technique could work if there angle of view is so small and the motion between the cameras so large that there is no overlap between the pictures. When this point is reached, the smoothing window required would be so large that each picture would be smoothed to a uniform gray. Nevertheless, these results are useful in that they verify that a useful range of convergence is obtainable using the method.

What is the relationship between these convergence ranges and the convergence ranges in the multi-parameter case? This is shown in Figure 2. We see that if we solve for two parameters (pan and tilt, top graph), the range is smaller than the range that would he expected on the basis of the one-parameter results for pan and tilt alone; and if we solve for all six parameters (bottom graph), it is smaller still. Nevertheless, the range is still quite adequate for the continuous feedback mode. Whether it is adequate for (he stopand-go mode, which involves a larger motion at each step, depends on the accuracy of the aim and on the accuracy of other navigational aids that can provide the initial estimates.

Accuracy. To assess the accuracy under a variety of conditions, we select reference points using a variety of methods, including by hand and by computer, resulting in several sets of data points of various sizes. Then we doubled the number of sets of reference points by either applying or not applying a pruning process to the sets we had. This pruning process, which is described elsewhere (Lucas, 1985), was based on the method of differences and served to improve the accuracy of the stereo matches. It also eliminated some points as being unlit for use by the method, for example because they were in a region of small gradient. The results are shown in Figure 3. Several general trends are observable. First, using more points produces more accurate results. Second, the pruning process can to improve the results, as evidenced by the left endpoints of the lines in the figure being lower than the right endpoints. These two factors are of course in conflict, and the improvement due to the pruning process is apparent only provided the number of points is not reduced too much, finally, the accuracy does not seem to he affected much by the number of parameters solved for.

Implementation. The implementation may be divided into two parts: smoothing and camera parameter estimation. The smoothing must be done over a relatively large window, up to G5 x 65 in our experiments. It is the most time-consuming, part even though we implemented it as uniform smoothing over a rectangular region, which by a well known algorithm takes a constant number of operations (two additions and two subtractions) per pixel, regardless of the size of the smoothing window. However, it is fairly well understood how to build special-purpose hardware for doing smoothing quickly, essentially in real time.

The parameter estimation step is more interesting. Our implementation, in which no attention was paid to efficiency, requires approximately 3 to 4 ms per reference point per iteration on a VAX 11/780. In the continuous feedback mode, only one iteration per time step would be used since only an approximate answer is needed. Thus 50 reference points (the largest number used in the experiments reported above) would require less than 200 ms per time step. This figure could probably be improved severalfold by more careful coding and taking account of the fact that some of the entries in the matrices to be inverted are known *a priori* to be zero. This information, together with the fact that the algorithm has a regular structure free of decision points that could easily be implemented in special-purpose hardware, suggests that it is feasible for real-time control of a robot.

4. Conclusions

We have demonstrated that the method of differences provides a useful technique for optical navigation. We have shown that the algorithm can successfully determine all six camera parameters. It converges to the correct position given an estimate within something on the order of a meter (less if more parameters are solved for), and converges to a result accurate to a centimeter or so (regardless of the number of parameters solved for). Moreover, it can do so using 50 or less reference points. Because of the regular structure of the algorithm, the prospects of carrying out the calculations in real time with special-purpose hardware seem good.

5. References

A. R. Bruss and B. K. P. Horn, 1983. Passive navigation. *Computer Vision, Graphics, and Image Processing,* 21, 3-20.

C. Cafforio and F. Rocca, 1979. Tracking moving objects in television images. *Signal Processing,* 1, 133-140.

N. H. Cornelius and T. Kanade, 1983. Adapting optical-flow to measure object motion in reflectance and x-ray image sequences. Proc. ACM SICCRAPH/SIGART Workshop on Motion: Representation and Perception, Toronto, 50-58.

D. B. Gennery, 1980. Modeling the environment of an exploring vehicle by means of stereo vision. Ph1) Thesis, Department of Computer Science, Stanford University.

B. K. P. Horn and B. G. Schunck, 1981. Determining optical flow. *Artificial Intelligence,* 17, 185-202.

J. (.). Limb and .1. A. Murphy, 1975. Estimating the velocity of moving images in television signals. *Computer Graphics and Image Processing,* 4, 311-327.

B. D. Lucas, 1985. Generalized image matching by the method of differences: algorithms and applications. PhD Thesis (in preparation), Computer Science Department, Carnegie-Mellon University.

13. D. Lucas and T. Kanade, 1981. An iterative image registration technique with an application to stereo vision. Proc. Seventh International Joint Conference on Artificial Intelligence, Vancouver.

11. P. Moravcc, 1980. Obstacle avoidance and navigation in the real world by a seeing robot rover. Tech. Rept. CMU-RI-TR-3, Robotics Institute, Carnegie-Mellon University.

R. Y. Tsai and T. S. Huang, 1981. Uniqueness and estimation of 3-D motion parameters of rigid objects with curved surfaces. Proc. IEEE Conference on Pattern Recognition and Image Processing.

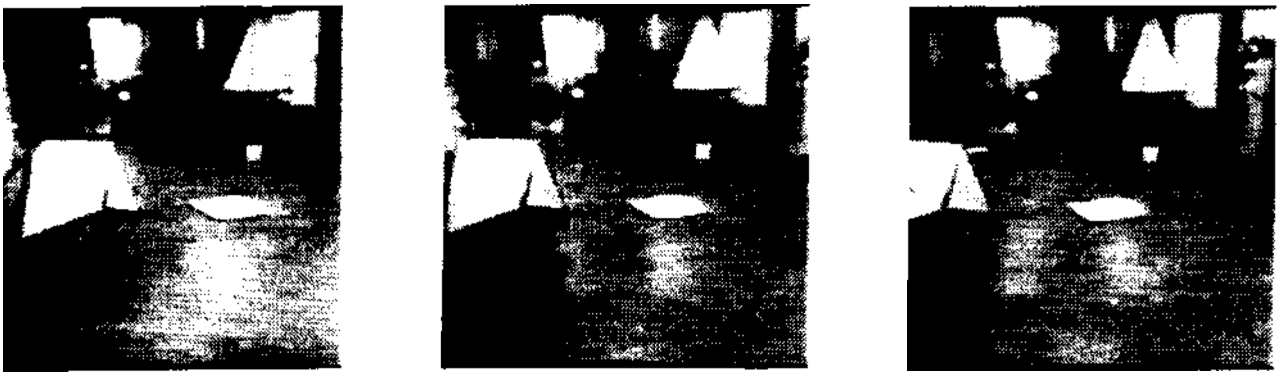

Figure 1. Experimental data. Left, middle, and right view: of the same scene. Reference points are shown on left, (reference) image.

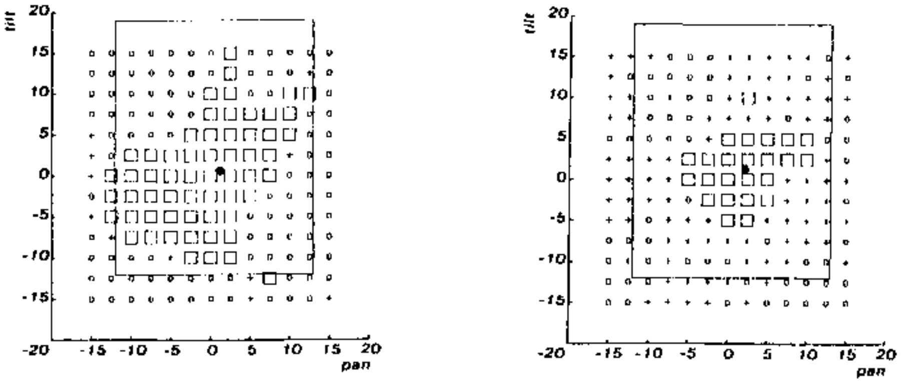

Figure 2. Left graph shows, for each initial value of pan and tilt, whether the algorithm converged to the correct value (large boxes), converged to the wrong value (small circles), or failed to converge (pluses). Solid dot is correct value, big rectangle indicates range predicted by single-parameter results. Right graph is a two dimensional slice of a similar six-dimensional solid, in which all six parameters were solved for.

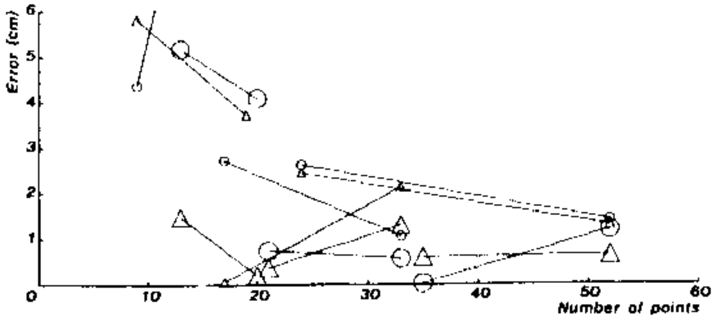

Figure 3. Graph shows the absolute error in x position on images smoothed with 9 x 0 window. Each point represents the result with a different set of reference points, distinguished by resulting error (in cm) on the vertical axis, and by number of points in the reference set on the horizontal axis. Triangles indicate the case where three parameters were solved for, circles six. The point at the left end of each line represents a reference set in which a pruning process was carried out on the points represented by the right end of the line. Large points represent image pair discussed in text, small points represent a different image pair.