On the Evaluation of Optical Flow Algorithms

B. McCane B. Galvin K. Novins

Computer Science Department
University of Otago, New Zealand.
mccane@atlas.otago.ac.nz

Abstract

Evaluating the performance of optical flow algorithms has been difficult because of the lack of ground-truth data sets for complex scenes. We describe a simple modification to a ray tracer that allows us to generate ground-truth motion fields for scenes of arbitrary complexity. The resulting flow maps are used to assist in the comparison of eight optical flow algorithms using two complex, synthetic scenes. We introduce a new performance metric based on the correlation of vector gradient magnitude images which is especially pertinent for evaluating optical flow algorithms for boundary detection applications.

1 Introduction

Optical flow continues to receive a significant amount of research attention, since Horn and Schunck published their influential paper [10]. Yet in almost two decades involving the publication of numerous papers and techniques, there has been only one major review and comparison of optical flow techniques [5]. A contributing factor to this lack of comparisons in the research is the difficulty involved in comparing different algorithms. Typically, this takes the form of subjective evaluation of results, or comparison against ground truth data obtained from extremely simplistic computer generated scenes (eg translating 2D shapes).

This paper builds on our previous published work [8] and the work of Barron et al. [5]. In the work by Barron et al, the output of the optical flow algorithms were compared against the ground truth motion field of several simple scenes and one reasonably complex one (the Yosemite sequence), and were also subjectively evaluated on more complex real scenes. We have developed a ray tracer that can generate ground truth motion fields for arbitrarily complex scenes [8]. The source code for the ray tracer is publicly available at http://heffalump.otago.ac.nz/∼masondr/of/, and is briefly described in Section 2. This was an important addition to the field since no other tool of this kind is publicly available (or even exists as far as the authors are aware). The ground truth motion fields for complex scenes were then compared with the optical flow field produced by each of the algorithms using a number of methods described in Section 5.

Although this has proved to be a reasonable way to evaluate optical flow algorithms, and almost certainly is better than subjective evaluation, it's not the end of the story. The calculation of optical flow is often used as an early process in vision systems. One of the uses of the flow-map is to find object boundaries in the scene which higher level processes can then use. Given that discontinuities in the motion field are highly likely to correspond to object boundaries, it would seem useful to attempt to detect object boundaries from optical flow data. In this paper, we make use of a vector gradient magnitude map to compare the output of optical flow algorithms to ground truth motion fields.

The paper is organised as follows. In Section 2 we briefly describe the ray tracer used to generate ground truth motion fields. In Section 4 we describe the method for calculating the vector gradient magnitude fields and how these can be used to evaluate optical flow algorithms. Finally, some results are shown in Section 5 and conclusions and future work are discussed in Section 6.

2 Generating Ground Truth

The ground-truth motion fields were generated by modifying a public license ray tracer, Mirage [6]. Mirage objects can be animated by specifying affine transformations which describe their rigid motion from one frame to the next.

The motion field is generated during the ray tracing of two consecutive frames. The entire process is illustrated in Figure 1. Rendering of the first frame begins by casting rays from the camera into the scene. Whenever a ray strikes an object, the 3D position of the intersection, the object, and the corresponding pixel position (x, y) are recorded. Before the second frame is rendered, Mirage applies a transformation to each object. In our modified version of Mirage, the transformation applied to each object is also applied to each recorded ray-object intersection.

Once this has been completed, rays are cast from each new intersection point to the camera. Simple geometry yields the new pixel position (x', y'). A simple subtraction then yields the ground-truth motion vector.

3 Error Metrics

Trying to summarize the performance of over a million flow vectors with a single number is a difficult task. It is further complicated by attempting to define a general, non-application specific measure. We felt simplicity was impor-

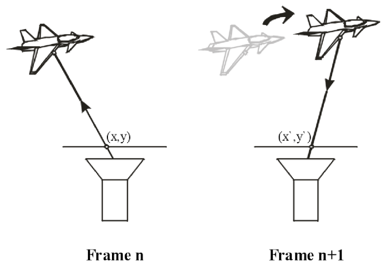


Figure 1: Geometry for calculating the the ground-truth motion field.

tant for any error metric, even if it was at the expense of more subtle characteristics.

Our first error metric is the difference in angle between the correct and estimated flow vectors. We define

$$\text{Angular Error} = \cos^{-1}(\hat{c} \cdot \hat{e}), \tag{1}$$

where c is the correct motion vector, e is the estimate optical flow vector, and ˆ denotes vector normalization.

The second metric is the average magnitude of the vector difference between the correct and estimated flow vectors for each pixel. This can be pictured as the magnitude of the "correction image" that would have to be added to the calculated flow to get the correct flow:

$$\text{Magnitude of Difference} = \|c - e\|. \tag{2}$$

Unfortunately neither of these metrics are ideal. For example, in image areas where the optical flow is small, the angular error of the flow field is less important than in areas where the flow is large. Conversely, a large magnitude error in areas of small flow is more significant than an equivalent magnitude error in areas of large flow. Also, these simple metrics do not relate directly to the way that optical flow is used by higher level processes, e.g. for motion segmentation. For these reasons, we decided to investigate alternative error metrics.

4 Vector Gradient Magnitude

Lee and Cok [12] describe a technique for finding discontinuities in a vector field. Let $f(x, y)$ be an n dimensional vector field over a two dimensional domain (ie an image), and $f'(x, y)$ be the $n \times 2$ matrix of partial derivatives:

$$f'(x,y) = D(x,y) = \begin{bmatrix} \frac{df_1}{dx} & \frac{df_1}{dy} \\ \frac{df_2}{dx} & \frac{df_2}{dy} \\ \vdots & \vdots \\ \frac{df_n}{dx} & \frac{df_n}{dy} \end{bmatrix} \tag{3}$$

where $\frac{df_i}{dx}$ is the partial derivative of the i^{th} component of the vector field with respect to x. If we form the matrix $D^T D$, then the square root of the largest eigenvalue of this matrix is a scalar quantity that can be considered equivalent to the gradient magnitude of a scalar field [12, 4].

In the case of optical flow fields, $n = 2$, so $f'(x, y)$ is a 2×2 matrix as is $D^T D$. We use the maximum eigenvalue field (the vector gradient magnitude) as the base for evaluating optical flow error. For all comparisons here, the flow maps were pre-processed by smoothing with a Gaussian of $\sigma = 1.0$ or $\sigma = 2.0$ pixels to reduce noise and then using a 2 point central difference form for approximating the partial derivatives. Comparisons between the ground truth flow gradient magnitude fields and the output flow gradient magnitude fields were performed using the correlation coefficient for comparing two images [9].

5 Results

We have evaluated 8 optical flow algorithms which we refer to as follows: Anandan [2, 3]; Horn [10]; Lucas-Kanade [13, 1, 11, 18]; Nagel [14, 15, 16]; Singh [19, 20]; Uras [21]; Proesmans [17]; and Camus [7]. For a description of these algorithms, the reader is refered to the original papers or the review paper of Barron et al. [5] (except Proesmans and Camus). We have used the same parameters as Barron et al. unless stated otherwise.

Two test sequences of high complexity were used for evaluation. An office scene is the subject of the first image sequence. The camera moves in the direction of its viewing axis, at a constant velocity. Specular highlights and transparency are deliberately included to simulate a real scene as much as possible. A typical frame is shown in Figure 2(a). The second sequence is set on a simulated busy street corner. The camera follows a car as it turns into a side-street. A typical frame is shown in Figure 2(b). Example flow maps calculated by the various algorithms are displayed in Figure 3. We utilise three error metrics: average angular error difference and average magnitude error difference, and the correlation coefficient of the vector gradient fields. Results were averaged over 20 frames of the office sequence and 15 frames of the street sequence. The results for each of the algorithms tested is shown in Figure 5. Example vector gradient magnitude fields for both the office and street scenes are displayed in Figure 4.

As can be seen from Figure 5 the results are quite differ-

(a) Office Scene (b) Street Scene

Figure 2: Typical scenes from the test sequences. All image sequences were rendered at 200x200 resolution in 256 level gray-scale. The office sequence is 60 frames long. The street sequence is 150 frames long.

ent for the gradient magnitude correlations and the other two metrics. This is because for the gradient magnitude comparisons, we are comparing results at flow discontinuities - the very place that many optical flow algorithms perform poorly. Nevertheless, if we wish to use optical flow to find object boundaries, then this should be exactly the kind of comparison to make, rather than the more usual vector magnitude or angular error where any errors at discontinuities are swamped by the smooth regions of the flow fields. Given these results it is clear that the algorithm of Proesmans et al. is superior to the other algorithms. It is also noteworthy that the algorithm of Horn and Schunck, which performs well for the standard error metrics, performs poorly at flow field discontinuities. This is hardly surprising, however, given the assumptions of smoothness inherent in that algorithm. Interestingly, many algorithms perform well with the street scene, but only Proesmans et al. performs consistently well with the office scene. This is because the office scene has the camera moving into the scene and hence the discontinuities in the flow field are quite subtle. The office scene is in fact a very difficult scene from which to calculate optical flow and it would be fairer to say that the algorithm of Proesmans et al. has done exceptionally well rather than that the other algorithms have performed poorly. The main reason that Proesmans et al algorithm works so well is that it uses anisotropic diffusion to reduce smoothing across flow boundaries.

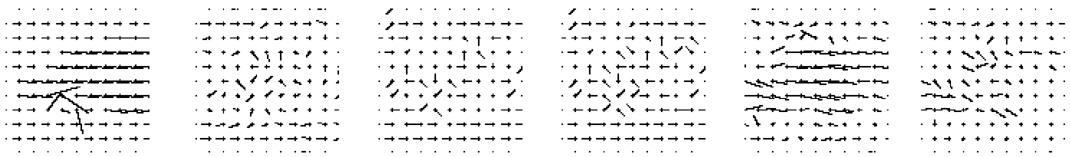
The contrast between the results of Figure 5(c) and 5(d) is also noteworthy. In 5(c), the smoothing parameter is quite small and therefore the localisation of discontinuities is important in the comparison. A small inaccuracy in the localisation of the discontinuity will result in a large error across the image. Hence, even those algorithms that produce results that qualitatively look reasonable, quantitatively, they are still quite inaccurate. As the smoothing parameter is increased, localisation effects are reduced and the gross structure of the flow fields becomes more important, hence the reason that more algorithms are seen to do well.

6 Conclusion

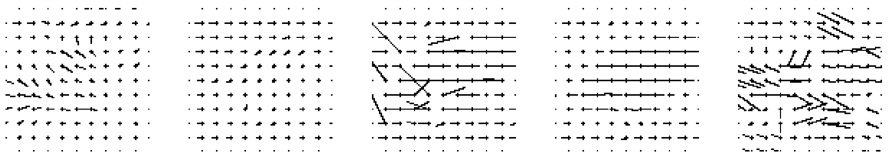
We have developed a method for producing ground-truth optical flow from synthetic scenes of arbitrary complexity. We have used this tool to objectively evaluate the performance of eight optical flow algorithms on complex scenes using standard error metrics and a metric based on discontinuities in the optical flow field. This last metric is more meaningful for evaluating optical flow algorithms if the purpose of utilising such algorithms is to find object boundaries in the image sequences. Proesmans et al.'s algorithm performed the best using this metric and also performed well using the other metrics and as such is the best algorithm to use in this context. Barron et al. [5] found that the algorithm of Lucas and Kanade was superior, but they did not test Proesmans et al.'s or Camus' algorithm, and they allowed for sparse optical flow fields to be used in their comparison. It has also been pointed out that using standard error metrics such as average magnitude error and average angular error does not necessarily give a good measure of algorithm performance, especially in the context of finding flow discontinuities.

References

[1] E.H. Adelson and J.R. Bergen. Spatiotemporal energy models for the perception of motion. In *Proc. IEEE Workshop on Visual Motion*, pages 151–156, 1986.



(a) Ground Truth (b) Anandan (c) Camus (15 images) (d) Camus (3 images) (e) Horn and Schunck (f) Lucas and Kanade



(g) Nagel (h) Proesmans et al. (i) Singh Step 1 (j) Singh Step 2 (k) Uras et al

Figure 3: Some example flow maps. These are taken from a small section of the street scene in which a wheel of a car is rotating.

[2] P. Anandan. *Measuring Visual Motion from Image Sequences*. PhD thesis, Univ. of Massachusetts, 1987.

[3] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *Intern. J. Comput. Vis.*, 2:283–310, 1989.

[4] T.W. Anderson. *An Introduction to Multivariate Statistical Analysis*. Wiley, 2nd edition, 1984.

[5] J. Barron, D. Fleet, S. Beauchemin, and T. Burkitt. Performance of optical flow techniques. *Proc. of the IEEE on Computer Vision and Pattern Recognition*, pages 236–242, 1992.

[6] C. Butcher. Mirage. A ray tracing package, available from http://atlas.otago.ac.nz:800 /students/butcher_cc/pages/mirage.html.

[7] T. Camus. Real-time quantized optical flow. *The Journal of Real-Time Imaging (special issue on Real-Time Motion Analysis)*, 3:71–86, 1997.

[8] B. Galvin, B. McCane, K. Novins, D. Mason, and S. Mills. Recovering motion fields: An evaluation of eight optical flow algorithms. In *British Machine Vision Conference*, 1998.

[9] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley, 1992.

[10] B. K. P. Horn and B. G. Schunck. Determining optical flow. AI Memo 572, Massachusetts Institute of Technology, 1980.

[11] J.K. Kearney, W.B. Thompson, and D.L. Boley. Optical flow estimation: An error analysis of gradient based methods with local optimization. *IEEE Trans. Patt. Anal. Artif. Mach. Intell.*, 9:229–244, 1987.

[12] H.-C. Lee and D.R. Cok. Detecting boundaries in a vector field. *IEEE Transactions on Signal Processing*, 39(5):1181 – 1194, May 1991.

[13] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. DARPA Image Understanding Workshop*, pages 121–130, 1984.

[14] H.H. Nagel. Displacement vectors derived from second-order intensity variations in image sequences. *Comput. Graph. Image Image Process.*, 21:85–117, 1983.

[15] H.H. Nagel. On the estimation of optical flow: Relations between different approaches and some new results. *Artifical Intelligence*, 33:299–324, 1987.

[16] H.H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Trans. Patt. Anal. Mach. Intell.*, 8:565–593, 1986.

[17] M. Proesmans, L. Van Gool, E. Pauwels, and A. Oosterlinck. Determination of optical flow and its discontinuities using non-linear diffusion. In *3rd European Conference on Computer Vision, ECCV'94*, volume 2, pages 295–304, 1994.

[18] E.P. Simoncelli. *Distributed Representation and Analysis of Visual Motion*. PhD thesis, MIT, 1993.

[19] A. Singh. An estimation-theoretic framework for image-flow computation. In *Proc. 3rd Intern. Conf. Comput. Vis.*, pages 168–177, 1990.

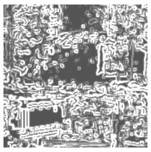
[20] A. Singh. *Optic Flow Computation: A Unified Perspective*. IEEE Computer Society Press, 1992.

[21] S. Uras, F. Girosi, A. Verri, and V. Torre. A computational approach to motion perception. *Biol. Cybern.*, 60:79–97, 1988.

(a) Ground Truth

(b) Anandan

(c) Camus (15 images)



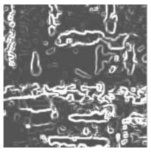
(d) Camus (3 images)

(e) Horn and Schunck

(f) Lucas and Kanade

(g) Nagel

(h) Proesmans et al.

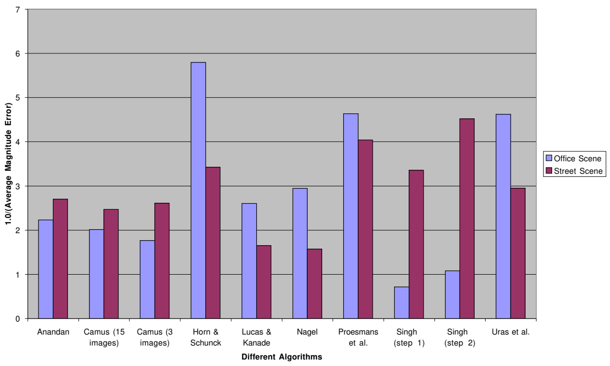


(i) Singh Step 1

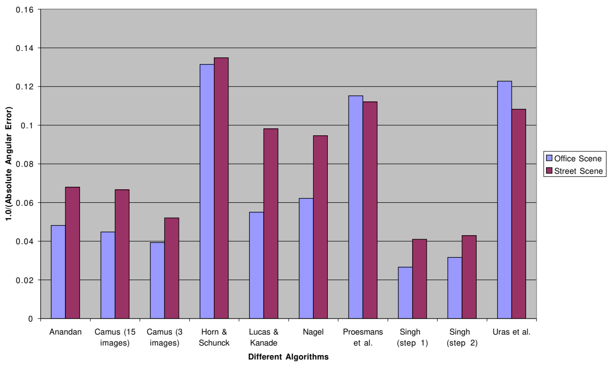
(j) Singh Step 2

(k) Uras et al.

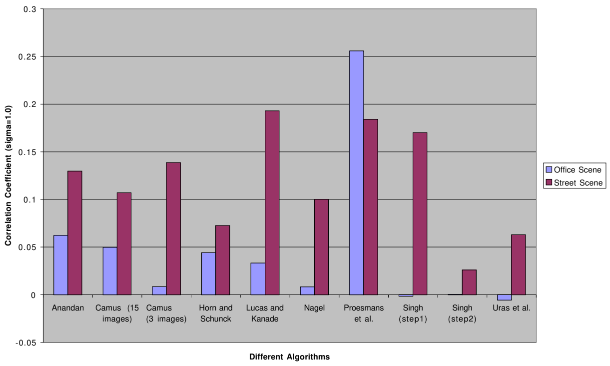
Figure 4: Some example gradient magnitude fields from the office scene.



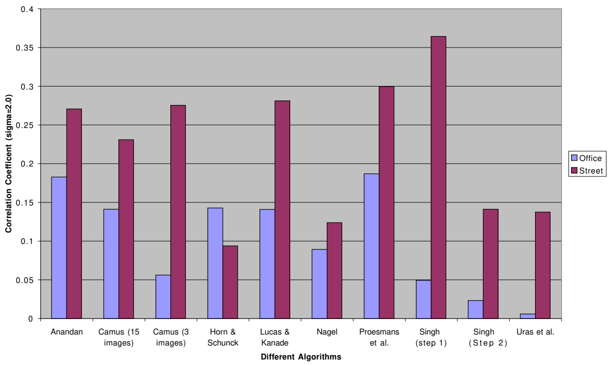
(a) Average Magnitude of Difference



(b) Average Absolute Angular Error



(c) Correlation Coefficient of Vector Gradient Fields with $\sigma = 1.0$



(d) Correlation Coefficient of Vector Gradient Fields with $\sigma = 2.0$

Figure 5: Comparison with ground truth using various error metrics