Determining optical flow

Berthold K. P. Horn, Brian G. Schunck
Artificial Intelligence Laboratory, Massachusetts Institute of Technology
545 Technology Square, NE43-811, Cambridge, Massachusetts 02139

Abstract

Optical flow cannot be computed locally, since only one independent measurement is available from the image sequence at a point, while the flow velocity has two components. A second constraint is needed. A method for finding the optical flow pattern is presented which assumes that the apparent velocity of the brightness pattern varies smoothly almost everywhere in the image. An iterative implementation is shown which successfully computes the optical flow for a number of synthetic image sequences. The algorithm is robust in that it can handle image sequences that are quantized rather coarsely in space and time. It is also insensitive to quantization of brightness levels and additive noise. Examples are included where the assumption of smoothness is violated at singular points or along lines in the image.

1. Introduction

Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image. Optical flow can arise from relative motion of objects and the viewer [8, 9]. Consequently, optical flow can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement [10]. Discontinuities in the optical flow can help in segmenting images into regions that correspond to different objects [29]. Attempts have been made to perform such segmentation using differences between successive image frames [17, 18, 19, 22, 3, 27]. Several papers address the problem of recovering the motions of objects relative to the viewer from the optical flow [12, 20, 21, 23, 31]. Some recent papers provide a clear exposition of this enterprise [32, 33]. The mathematics can be made rather difficult, by the way, by chosing an inconvenient coordinate system. In some cases information about the shape of an object may also be recovered [4, 20, 21].

These papers begin by assuming that the optical flow has already been determined. Although some reference has been made to schemes for computing the flow from successive views of a scene [7, 12], the specifics of a scheme for determining the flow from the image have not been described. Related work has been done in an attempt to formulate a model for the short range motion detection processes in human vision [2, 24]. The pixel recursive equations of Netravali and Robbins [30], designed for coding motion in television signals, bear some similarity to the iterative equations developed in this paper. A recent review [28] of computational techniques for the analysis of image sequences contains over 150 references.

The optical flow cannot be computed at a point in the image independently of neighboring points without introducing additional constraints, because the velocity field at each image point has two components while the change in image brightness at a point in the image plane due to motion yields only one constraint. Consider, for example, a patch of a pattern where brightness[1] varies as a function of one image coordinate but not the other. Movement of the pattern in one direction alters the brightness at a particular point, but motion in the other direction yields no change. Thus components of movement in the latter direction cannot be determined locally.

2. Relationship to Object Motion

The relationship between the optical flow in the image plane and the velocities of objects in the three dimensional world is not necessarily obvious. We perceive motion when a changing picture is projected onto a stationary screen, for example. Conversely, a moving object may give rise to a constant brightness pattern. Consider, for example, a uniform sphere which exhibits shading because its surface elements are oriented in many different directions. Yet, when it is rotated, the optical flow is zero at all points in the image, since the shading does not move with the surface. Also, specular reflections move with a velocity characteristic of the virtual image, not the surface in which light is reflected.

For convenience, we tackle a particularly simple world where the apparent velocity of brightness patterns can be directly identified with the movement of surfaces in the scene.

3. The Restricted Problem Domain

To avoid variations in brightness due to shading effects we initially assume that the surface being imaged is flat. We further assume that the incident illumination is uniform across the surface. The brightness

[1] In this paper, the term brightness means image irradiance. The brightness pattern is the distribution of irradiance in the image.

at a point in the image is then proportional to the reflectance of the surface at the corresponding point on the object. Also, we assume at first that reflectance varies smoothly and has no spatial discontinuities. This latter condition assures us that the image brightness is differentiable. We exclude situations where objects occlude one another, in part, because discontinuities in reflectance are found at object boundaries. In two of the experiments discussed later, some of the problems occasioned by occluding edges are exposed.

In the simple situation described, the motion of the brightness patterns in the image is determined directly by the motions of corresponding points on the surface of the object. Computing the velocities of points on the object is a matter of simple geometry once the optical flow is known.

4. Constraints

We will derive an equation that relates the change in image brightness at a point to the motion of the brightness pattern. Let the image brightness at the point (x, y) in the image plane at time t be denoted by $E(x, y, t)$. Now consider what happens when the pattern moves. The brightness of a particular point in the pattern is constant, so that

$$\frac{dE}{dt} = 0$$

Using the chain rule for differentiation we see that,

$$\frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial y}\frac{dy}{dt} + \frac{\partial E}{\partial t} = 0$$

(See Appendix A for a more detailed derivation.) If we let

$$u = \frac{dx}{dt} \quad \text{and} \quad v = \frac{dy}{dt},$$

then it is easy to see that we have a single linear equation in the two unknowns u and v,

$$E_x u + E_y v + E_t = 0,$$

where we have also introduced the additional abbreviations E_x, E_y, and E_t for the partial derivatives of image brightness with respect to x, y and t, respectively. The constraint on the local flow velocity expressed by this equation is illustrated in Figure 1. Writing the equation in still another way,

$$(E_x, E_y) \cdot (u, v) = -E_t.$$

Thus the component of the movement in the direction of the brightness gradient (E_x, E_y) equals

$$-\frac{E_t}{\sqrt{E_x^2 + E_y^2}}.$$

We cannot, however determine the component of the movement in the direction of the iso-brightness contours, at right angles to the brightness gradient. As a consequence, the flow velocity (u, v) cannot be computed locally without introducing additional constraints.

5. The Smoothness Constraint

If every point of the brightness pattern can move independently, there is little hope of recovering the velocities. More commonly we view opaque objects of finite size undergoing rigid motion or deformation. In this case neighboring points on the objects have similar velocities and the velocity field of the brightness patterns in the image varies smoothly almost everywhere. Discontinuities in flow can be expected where one object occludes another. An algorithm based on a smoothness constraint is likely to have difficulties with occluding edges as a result.

One way to express the additional constraint is to minimize the square of the magnitude of the gradient of the optical flow velocity:

$$\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 \quad \text{and} \quad \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2.$$

Another measure of the smoothness of the optical flow field is the sum of the squares of the Laplacians of the two velocity components. The Laplacians of u and v are defined as

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \quad \text{and} \quad \nabla^2 v = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}.$$

In simple situations, both Laplacians are zero. If the viewer translates parallel to a flat object, rotates about a line perpendicular to the surface or travels orthogonally to the surface, then the second partial derivatives of both u and v vanish (assuming perspective projection in the image formation.)

In this paper, we will use the square of the magnitide of the gradient as our smoothness measure. Note that our approach is in contrast with that of [7], who propose an algorithm that incorporates additional assumptions such as constant flow velocities within discrete regions of the image. Their method, based on cluster analysis, cannot deal with rotating objects, since these give rise to a continuum of flow velocities.

6. Quantization and Noise

Images may be sampled at intervals on a fixed grid of points. While tesselations other than the obvious one have certain advantages [11, 25], for convenience we will assume that the image is sampled on a square grid at regular intervals. Let the measured brightness be $E_{i,j,k}$ at the intersection of the i-th row and j-th column in the k-th image frame. Ideally, each measurement should be an average over the area of a picture cell and over the length of the time interval. In the experiments cited here we have taken samples at discrete points in space and time instead.

In addition to being quantized in space and time, the measurements will in practice be quantized in brightness as well. Further, noise will be apparent in measurements obtained in any real system.

7. Estimating the Partial Derivatives

We must estimate the derivatives of brightness from the discrete set of image brightness measurements available. It is important that the estimates of E_x, E_y, and E_t be consistent. That is, they should all refer to the same point in the image at the same time. While there are many formulas for approximate differentiation [5, 13] we will use a set which gives us an estimate of E_x, E_y, E_t at a point in the center of a cube formed by eight measurements. The relationship in space and time between these measurements is shown in Figure 2. Each of the estimates is the average of four first differences taken over adjacent measurements in the cube.

$$\begin{aligned}
E_x \approx \frac{1}{4}\{&E_{i,j+1,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i+1,j,k} \\
&+ E_{i,j+1,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i+1,j,k+1}\} \\
E_y \approx \frac{1}{4}\{&E_{i+1,j,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i,j+1,k} \\
&+ E_{i+1,j,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i,j+1,k+1}\} \\
E_t \approx \frac{1}{4}\{&E_{i,j,k+1} - E_{i,j,k} + E_{i+1,j,k+1} - E_{i+1,j,k} \\
&+ E_{i,j+1,k+1} - E_{i,j+1,k} + E_{i+1,j+1,k+1} - E_{i+1,j+1,k}\}.
\end{aligned}$$

Here the unit of length is the grid spacing interval in each image frame and the unit of time is the image frame sampling period. We avoid estimation formulae with larger support, since these typically are equivalent to formulae of small support applied to smoothed images [16].

8. Estimating the Laplacian of the Flow Velocities

As will be shown in the next section, we will also need to approximate the Laplacians of u and v. One convenient approximation takes the following form

$$\nabla^2 u \approx \kappa(\bar{u}_{i,j,k} - u_{i,j,k}) \quad \text{and} \quad \nabla^2 v \approx \kappa(\bar{v}_{i,j,k} - v_{i,j,k}),$$

where the local averages $\bar{u}$ and $\bar{v}$ are defined as follows

$$\begin{aligned}
\bar{u}_{i,j,k} &= \frac{1}{6}\{u_{i-1,j,k} + u_{i,j+1,k} + u_{i+1,j,k} + u_{i,j-1,k}\} \\
&+ \frac{1}{12}\{u_{i-1,j-1,k} + u_{i-1,j+1,k} + u_{i+1,j+1,k} + u_{i+1,j-1,k}\} \\
\bar{v}_{i,j,k} &= \frac{1}{6}\{v_{i-1,j,k} + v_{i,j+1,k} + v_{i+1,j,k} + v_{i,j-1,k}\} \\
&+ \frac{1}{12}\{v_{i-1,j-1,k} + v_{i-1,j+1,k} + v_{i+1,j+1,k} + v_{i+1,j-1,k}\}.
\end{aligned}$$

The proportionality factor κ equals 3 if the average is computed as shown and we again assume that the unit of length equals the grid spacing interval. Figure 3 illustrates the assignment of weights to neighboring points. The approximation for the Laplacian using the center cell and all eight neighbors is more stable than the usual one based on the center cell and its four horizontal and vertical neighbors only.

9. Minimization

The problem then is to minimize the sum of the errors in the equation for the rate of change of image brightness,

$$\mathfrak{e}_b = E_x u + E_y v + E_t,$$

and the measure of the departure from smoothness in the velocity flow,

$$\mathfrak{e}_c^2 = \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2$$

What should be the relative weight of these two factors? In practice the image brightness measurements will be corrupted by quantization error and noise so that we cannot expect $\mathfrak{e}_b$ to be identically zero. This quantity will tend to have an error magnitude that is proportional to the noise in the measurement. This fact guides us in choosing a suitable weighting factor, denoted by α^2, as will be seen later.

Let the total error to be minimized be

$$\mathfrak{e}^2 = \int\int \alpha^2 \mathfrak{e}_c^2 + \mathfrak{e}_b^2 \, dx \, dy.$$

The minimization is to be accomplished by finding suitable values for the optical flow velocity (u, v). Using the calculus of variations [6, pp. 191-92], we obtain

$$\begin{aligned}
E_x^2 u + E_x E_y v &= \alpha^2 \nabla^2 u - E_x E_t \\
E_x E_y u + E_y^2 v &= \alpha^2 \nabla^2 v - E_y E_t.
\end{aligned}$$

Using the approximation to the Laplacian introduced in the previous section,

$$\begin{aligned}
(\alpha^2 + E_x^2)u + E_x E_y v &= (\alpha^2 \bar{u} - E_x E_t) \\
E_x E_y u + (\alpha^2 + E_y^2)v &= (\alpha^2 \bar{v} - E_y E_t).
\end{aligned}$$

The determinant of the coefficient matrix equals $\alpha^2(\alpha^2 + E_x^2 + E_y^2)$. Solving for u and v we find that

$$\begin{aligned}
(\alpha^2 + E_x^2 + E_y^2)u &= +(\alpha^2 + E_y^2)\bar{u} - E_x E_y \bar{v} - E_x E_t \\
(\alpha^2 + E_x^2 + E_y^2)v &= -E_x E_y \bar{u} + (\alpha^2 + E_x^2)\bar{v} - E_y E_t.
\end{aligned}$$

10. Difference of Flow at a Point from Local Average

These equations can be written in the alternate form

$$\begin{aligned}
(\alpha^2 + E_x^2 + E_y^2)(u - \bar{u}) &= -E_x[E_x \bar{u} + E_y \bar{v} + E_t] \\
(\alpha^2 + E_x^2 + E_y^2)(v - \bar{v}) &= -E_y[E_x \bar{u} + E_y \bar{v} + E_t].
\end{aligned}$$

This shows that the value of the flow velocity (u, v) which minimizes the error $\mathfrak{e}^2$ lies in the direction towards the constraint line along a line that intersects the constraint line at right angles. This relationship is illustrated geometrically in Figure 4. The distance from the local average is proportional to the error in the basic formula for rate of change of brightness when $\bar{u}$, $\bar{v}$ are substituted for u and v. Finally we can see that α^2 plays a significant role only for areas where the brightness gradient is small, preventing haphazard adjustments to the estimated flow velocity occasioned by noise in the estimated derivatives. This parameter should be roughly equal to the expected noise in the estimate of $E_x^2 + E_y^2$.

11. Constrained Minimization

When we allow α^2 to tend to zero we obtain the solution to a constrained minimization problem. Applying the method of Lagrange multipliers [35, 36] to the problem of minimizing $\mathcal{E}_c^2$ while maintaining $\mathcal{E}_b = 0$ leads to

$$E_y \nabla^2 u = E_x \nabla^2 v, E_x u + E_y v + E_t = 0.$$

Approximating the Laplacian by the difference of the velocity at a point and the average of its neighbors then gives us

$$(E_x^2 + E_y^2)(u - \bar{u}) = -E_x[E_x\bar{u} + E_y\bar{v} + E_t]$$
$$(E_x^2 + E_y^2)(v - \bar{v}) = -E_y[E_x\bar{u} + E_y\bar{v} + E_t].$$

Referring again to Figure 4, we note that the point computed here lies at the intersection of the constraint line and the line at right angles through the point $(\bar{u}, \bar{v})$. We will not use these equations since we do expect errors in the estimation of the partial derivatives.

12. Iterative Solution

We now have a pair of equations for each point in the image. It would be very costly to solve these equations simultaneously by one of the standard methods, such as Gauss-Jordan elimination [13, 14]. The corresponding matrix is sparse and very large since the number of rows and columns equals twice the number of picture cells in the image. Iterative methods, such as the Gauss-Seidel method [13, 15], suggest themselves. We can compute a new set of velocity estimates (u^{n+1}, v^{n+1}) from the estimated derivatives and the average of the previous velocity estimates (u^n, v^n) by

$$u^{n+1} = \bar{u}^n - E_x[E_x\bar{u}^n + E_y\bar{v}^n + E_t]/(\alpha^2 + E_x^2 + E_y^2)$$
$$v^{n+1} = \bar{v}^n - E_y[E_x\bar{u}^n + E_y\bar{v}^n + E_t]/(\alpha^2 + E_x^2 + E_y^2).$$

(It is interesting to note that the new estimates at a particular point do not depend directly on the previous estimates at the same point.)

The natural boundary condition for the variational problem turns out to be a zero normal derivative. At the edge of the image, some of the points needed to compute the local average of velocity lie outside the image. Here we simply copy velocities from adjacent points further in.

13. Filling In Uniform Regions

In parts of the image where the brightness gradient is zero, the velocity estimates will simply be averages of the neighboring velocity estimates. There is no local information to constrain the apparent velocity of motion of the brightness pattern in these areas. Eventually the values around such a region will propagate inwards. If the velocities on the border of the region are all equal to the same value, then points in the region will be assigned that value too, after a sufficient number of iterations. Velocity information is thus filled in from the boundary of a region of constant brightness.

If the values on the border are not all the same, it is a little more difficult to predict what will happen. In all cases, the values filled in will correspond to the solution of the Laplace equation for the given boundary condition [1, 26, 34].

The progress of this filling-in phenomena is similar to the propagation effects in the solution of the heat equation for a uniform flat plate, where the time rate of change of temperature is proportional to the Laplacian. This gives us a means of understanding the iterative method in physical terms and of estimating the number of steps required. The number of iterations should be larger than the number of pictures cells across the largest region that must be filled in. If the size of such regions is not known in advance one may use the cross-section of the whole image as a conservative estimate.

14. Tightness of Constraint

When brightness in a region is a linear function of the image coordinates we can only obtain the component of optical flow in the direction of the gradient. The component at right angles is filled in from the boundary of the region as described before. In general the solution is most accurately determined in regions where the brightness gradient is not too small and varies in direction from point to point. Information which constrains both components of the optical flow velocity is then available in a relatively small neighborhood. Too violent fluctuations in brightness on the other hand are not desirable since the estimates of the derivatives will be corrupted as the result of undersampling and aliasing.

15. Choice of Iterative Scheme

As a practical matter one has a choice of how to interlace the iterations with the time steps. On the one hand, one could iterate until the solution has stabilized before advancing to the next image frame. On the other hand, given a good initial guess one may need only one iteration per time-step. A good initial guess for the optical flow velocities is usually available from the previous time-step.

The advantages of the latter approach include an ability to deal with more images per unit time and better estimates of optical flow velocities in certain regions. Areas in which the brightness gradient is small lead to uncertain, noisy estimates obtained partly by filling in from the surround. These estimates are improved by considering further images. The noise in measurements of the images will be independent and tend to cancel out. Perhaps more importantly, different parts of the pattern will drift by a given point in the image. The direction of the brightness gradient will vary with time, providing information about both components of the optical flow velocity.

A practical implementation would most likely employ one iteration per time step for these reasons. We illustrate both approaches in the experiments.

16. Experiments

The iterative scheme has been implemented and applied to image sequences corresponding to a number of simple flow patterns. The results shown here are for a relatively low resolution image of 32 by 32 picture cells. The brightness measurements were intentionally corrupted by approximately 1% noise and then quantized into 256 levels to simulate a real imaging situation. The underlying surface reflectance pattern was a linear combination of spatially orthogonal sinusoids. Their wavelength was chosen to give reasonably strong brightness gradients without leading to undersampling problems. Discontinuities were avoided to ensure that the required derivatives exist everywhere.

Shown in Figure 5, for example, are four frames of a sequence of images depicting a sphere rotating about an axis inclined towards the viewer. A smoothly varying reflectance pattern is painted on the surface of the sphere. The sphere is illuminated uniformly from all directions so that there is no shading. We chose to work with synthetic image sequences so that we can compare the results of the optical flow computation with the exact values calculated using the transformation equations relating image coordinates to coordinates on the underlying surface reflectance pattern.

17. Results

The first flow to be investigated was a simple linear translation of the entire brightness pattern. The resulting computed flow is shown as a needle diagram in Figure 6 for 1, 4, 16, and 64 iterations. The estimated flow velocities are depicted as short lines, showing the apparent displacement during one time step. In this example a single time step was taken so that the computations are based on just two images. Initially the estimates of flow velocity are zero. Consequently the first iteration shows vectors in the direction of the brightness gradient. Later, the estimates approach the correct values in all parts of the image. Few changes occur after 32 iterations when the velocity vectors have errors of about 10%. The estimates tend to be too small, rather than too large, perhaps because of a tendency to underestimate the derivatives. The worst errors occur, as one might expect, where the brightness gradient is small.

In the second experiment one iteration was used per time step on the same linear translation image sequence. The resulting computed flow is shown in Figure 7 for 1, 4, 16, and 64 time steps. The estimates approach the correct values more rapidly and do not have a tendency to be too small, as in the previous experiment. Few changes occur after 16 iterations when the velocity vectors have errors of about 7%. The worst errors occur, as one might expect, where the noise in recent measurements of brightness was worst. While individual estimates of velocity may not be very accurate, the average over the whole image was within 1% of the correct value.

Next, the method was applied to simple rotation and simple contraction of the brightness pattern. The results after 32 time steps are shown in Figure 8. Note that the magnitude of the velocity is proportional to the distance from the origin of the flow in both of these cases. (By origin we mean the point in the image where the velocity is zero.)

In the examples so far the Laplacian of both flow velocity components is zero everywhere. We also studied more difficult cases where this was not the case. In particular, if we let the magnitude of the velocity vary as the inverse of the distance from the origin we generate flow around a line vortex and two dimensional flow into a sink. The computed flow patterns are shown in Figure 9. In these examples, the computation involved many iterations based on a single time step. The worst errors occur near the singularity at the origin of the flow pattern, where velocities are found which are much larger than one picture cell per time step.

Finally we considered rigid body motions. Shown in Figure 10 are the flows computed for a cylinder rotating about its axis and for a rotating sphere. In both cases the Laplacian of the flow is not zero and in fact the Laplacian of one of the velocity components becomes infinite on the occluding bound. Since the velocities themselves remain finite, reasonable solutions are still obtained. The correct flow patterns are shown in Figure 11. Comparing the computed and exact values shows that the worst errors occur on the occluding boundary. These boundaries constitute a one dimensional subset of the plane and so one can expect that the relative number of points at which the estimated flow is seriously in error will decrease as the resolution of the image is made finer.

In Appendix B it is shown that there is a direct relationship between the Laplacian of the flow velocity components and the Laplacian of the surface height. This can be used to see how our smoothness constraint will fare for different objects. For example, a rotating polyhedron will give rise to a flow which has zero Laplacian except on the image lines which are the projections of the edges of the body.

18. Summary

A method has been developed for computing optical flow from a sequence of images. It is based on the observation that the flow velocity has two components and that the basic equation for the rate of change of image brightness provides only one constraint. Smoothness of the flow was introduced as a second constraint. An iterative method for solving the resulting equation was then developed. A simple implementation provided visual confirmation of convergence of the solution in the form of needle diagrams. Examples of several different types of optical flow patterns were studied. These included cases where the Laplacian of the flow was zero as well as cases where it became infinite at singular points or along bounding curves.

The computed optical flow is somewhat inaccurate since it is based on noisy, quantized measurements. Proposed methods for obtaining information about the shapes of objects using derivatives (divergence and curl) of the optical flow field may turn out to be impractical since the inaccuracies will be amplified.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.