Determining Constant Optical Flow

Berthold K.P. Horn

The original optical flow algorithm [1] dealt with a flow field that could vary from place to place in the image, as would typically occur when a camera is moved through a three-dimensional environment—or if objects moved in front of a fixed camera.

A related, but simpler problem, is that of recovering the motion of an image, all parts of which move with the *same* velocity (section 4.3 in [2]). The solution of this problem enables "optical mice," as well as motion compensation in hand-held video cameras—critical to their operation.

We provide here background on moving images, a method for the estimation of image velocity using least squares applied to brightness derivatives, and an analysis of the sensitivity of the estimated image velocity to noise in brightness measurement.

Constant Brightness Assumption

As in the solution of the general optical flow problem [1], we start with the "constant brightness assumption." If $E(x, y, t)$ is the brightness at image point (x, y) at time t, then we assume that

$$\frac{d}{dt}E(x, y, t) = 0 \tag{1}$$

that is, we assume that as the image of some feature moves, it does not change brightness.

If the image moves δx in the x-direction and δy in the y-direction during the interval δt, then

$$E(x + \delta x, y + \delta y, t + \delta t) = E(x, y, t) \tag{2}$$

Taylor series expansion yields

$$E(x, y, t) + \frac{\partial E}{\partial x}\delta x + \frac{\partial E}{\partial y}\delta y + \frac{\partial E}{\partial t}\delta t + \ldots = E(x, y, t) \tag{3}$$

where the ellipsis (...) indicates terms that are of higher order in the increments. Cancelling $E(x, y, t)$ from both sides, we obtain

$$\frac{\partial E}{\partial x}\delta x + \frac{\partial E}{\partial y}\delta y + \frac{\partial E}{\partial t}\delta t + \ldots = 0 \tag{4}$$

Dividing through by δt and considering the limit as $\delta t \to 0$[1]:

$$\frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial y}\frac{dy}{dt} + \frac{\partial E}{\partial t} = 0 \tag{5}$$

[1] We could have obtained eq. 5 more directly by expanding the total derivative in eq. 1 using the chain rule, see [1].

Brightness Change Constraint Equation

This result, called the “brightness change constraint equation” can be written in the more compact form

$$uE_x + vE_y + E_t = 0 \tag{6}$$

if we let

$$E_x = \frac{\partial E}{\partial x}, E_y = \frac{\partial E}{\partial y}, \quad \text{and} \quad E_t = \frac{\partial E}{\partial t}, \tag{7}$$

be the brightness gradient (E_x, E_y) and the time-rate of change of brightness at a point in the image, and if we let

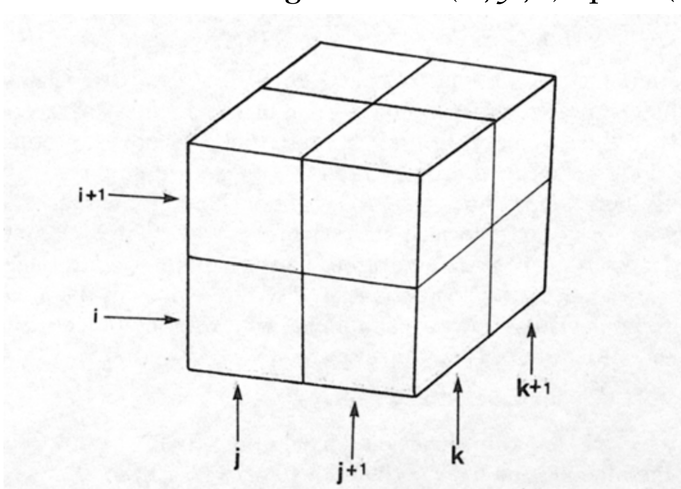
$$u = \frac{dx}{dt} \quad \text{and} \quad v = \frac{dy}{dt} \tag{8}$$

be the image velocity components in the x and y directions.

The brightness change constraint equation is a linear constraint on the image velocity components u and v with the partial derivatives of brightness, E_x, E_y, E_t as coefficients. This constraint equation applies at every picture cell in each image frame.

Numerical Estimation of Derivatives

Before we go on to use the brightness change constraint equation, let us consider just how we can estimate the partial derivatives of brightness. The derivatives can be estimated from images using first differences. It is, however, important to be consistent in how the three derivatives are estimated. The three derivative estimates should be “centered” at the same point in space and time. This can be accomplished by considering a $2 \times 2 \times 2$ cube of values of brightness in (x, y, t) space (see Fig. 1).



Each of the derivatives is based on the difference between the average value over one 2×2 side of the cube and the opposite 2×2 side.

For example,

$$\{E_x\}_{i,j,k} \approx \frac{1}{\epsilon_x}\left(\frac{1}{4}(E_{i,j+1,k} + E_{i+1,j+1,k} + E_{i,j+1,k+1} + E_{i+1,j+1,k+1}) - \frac{1}{4}(E_{i,j,k} + E_{i+1,j,k} + E_{i,j,k+1} + E_{i+1,j,k+1})\right) \tag{9}$$

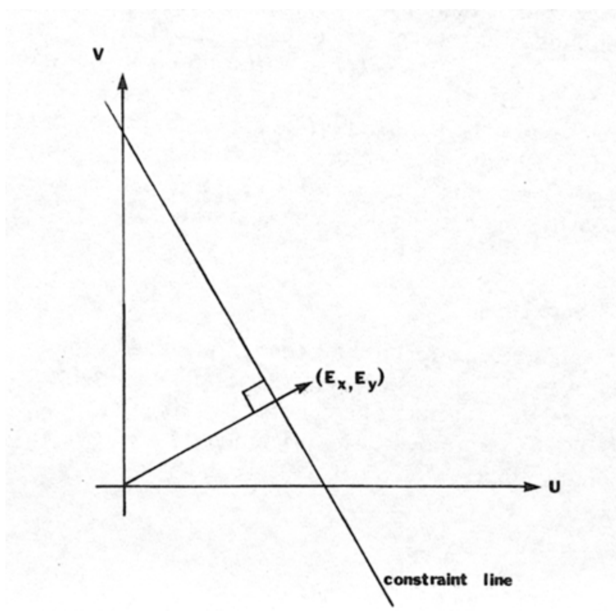
where $E_{i,j,k}$ is the brightness at pixel (i,j) (that is, the pixel in row i and column j) in frame k, and ϵ_x is the pixel spacing in the x direction.

Note that the estimates of the derivatives are centered at points that are in-between pixels and in-between image frames—rather than being aligned with them. In a rectangular array of pictures cells, there will be one fewer row and one fewer columns of derivative estimates than there are picture cell rows and columns. So $\{E_x\}_{i,j,k}$ is the estimate of the x-derivative of $E(x,y,t)$ at pixel $(i+1/2, j+1/2)$ at frame $(k+1/2)$.

Averaging four values[2] has the added benefit of reducing the standard deviation of random noise in the signal by one half. Additional smoothing may be applied to the image before estimating the derivatives in order to further reduce noise, as long as enough bits of precision are retained.

Analysis of Constraint and "Normal Flow Magnitude"

A *single* constraint does not, by itself, provide enough information to pin down *both* components of image velocity (u,v). The constraint can be expressed as a straight line in image velocity space (see Fig. 2).



All points (u,v) on that line are valid solutions. The constraint reduces the possibilities for (u,v) from two degrees of freedom to one.

[2] One may want to just keep the sum of the four values (rather than dividing by four) in order to retain more precision when using integer arithmetic.

We can see more clearly just what constraint a single measurement provides by dividing the brightness change constraint equation by the magnitude of the brightness gradient:

$$\frac{1}{\sqrt{E_x^2 + E_y^2}}(E_x, E_y) \cdot (u, v) = -\frac{E_t}{\sqrt{E_x^2 + E_y^2}} \tag{10}$$

The term to the left of the dot is a unit vector in the direction of the brightness gradient[3]. The dot-product yields the component of (u, v) in the direction of the brightness gradient. So the component of image velocity in the direction of the brightness gradient *can* be determined.

The component orthogonal to this direction (i.e. parallel to $(E_y, -E_x)$), however, can not. That is, the component in the direction along isophotes (contours of constant brightness) cannot be determined. The component in the direction of the brightness gradient (i.e. the right hand side term in eq. 10) is sometimes referred to as the "normal flow"—i.e. the flow "normal" or orthogonal to the isophotes.

Using Brightness Gradients at Two Points

To determine both components of image velocity, we need one more constraint. One way to obtain another constraint is to consider another point in the image—one where the local brightness gradient $(E_{x,2}, E_{y,2})$ is different from that at the first point $(E_{x,1}, E_{y,1})$. The brightness change constraint equation applied at these points—assuming that the image velocity is the same in the two places—give us two linear equations in u and v:

$$\begin{aligned} uE_{x,1} + vE_{y,1} &= -E_{t,1} \\ uE_{x,2} + vE_{y,2} &= -E_{t,2} \end{aligned} \tag{11}$$

or, equivalently two straight lines in velocity space.

We can write the two equations compactly in matrix-vector form as

$$M\mathbf{s} = \mathbf{b} \tag{12}$$

where $\mathbf{s} = (u, v)^T$, M is the 2×2 matrix of coefficients, and $\mathbf{b}$ has as components the right hand side values in eq. 11. The equations can be solved for the intersection of the lines as long as the determinant

$$\Delta = E_{x,1}E_{y,2} - E_{x,2}E_{y,1} \tag{13}$$

of the matrix M is non-zero. Then

$$\mathbf{s} = M^{-1}\mathbf{b} \tag{14}$$

[3] One can think of $E(x, y)$ as the height of a surface as a function of x and y. Then isophotes are contours of constant height and the direction of the brightness gradient is the direction of steepest ascent, while the magnitude of the brightness gradient is the slope in that direction.

or, in component form,

$$\begin{aligned} \Delta u &= E_{y,1}E_{t,2} - E_{y,2}E_{t,1} \\ \Delta v &= E_{x,2}E_{t,1} - E_{x,1}E_{t,2} \end{aligned} \tag{15}$$

The determinant will be zero when the two brightness gradients are parallel, that is, when the ratio $E_{y,2} : E_{x,2}$ equals the ratio $E_{y,1} : E_{x,1}$. In that case the two equations in eq. 11 are not independent.

The determinant is essentially the magnitude of the cross-product of the two brightness gradients $(E_{x,1}, E_{y,1})$ and $(E_{x,2}, E_{y,2})$. Consequently, the determinant will be small when the gradients are near parallel—or when the gradients are weak. In this case noise in measurements will be greatly amplified when computing u and v because of the division by the small determinant. So this method works best when the two brightness gradients point in very different directions—and are strong.

Another way of looking at this is to consider that the solution of the pair of equations in eq. 11 gives the intersection of two constraint lines in velocity space. The position of that intersection will be more sensitive to small errors in estimates of the coefficients of the implicit equations for the lines (eq. 11) when the angle between the two lines is small.

Using Constraint from All Picture Cells

Images are noisy, and derivatives estimated from them noisier still, so a velocity estimate based on measurements at just *two* image points will not be very reliable. Since we have many picture cells in an image we can use a least squares approach that adds up contributions from all places where image derivatives can be estimated. That is, we can consider this a highly overconstrained problem with many more equations than unknowns. We assume here that u and v are the same for all points in the image—unlike the situation in the general optical flow problem [1].

First, consider the continuous case. The integral

$$\iint_I (uE_x + vE_y + E_t)^2 \, dx \, dy \tag{16}$$

should be zero if we plug in the correct values for u and v—and if there are no measurement errors. In the presence of inevitable noise in brightness measurements, there will not be any values of u and v that make the integral exactly zero. However, the values of u and v that minimize the integral provide good estimates of the actual image velocity. Differentiating w.r.t. u and v and setting these derivatives equal to zero, we obtain:

$$\begin{aligned} 2\iint_I (uE_x + vE_y + E_t)E_x \, dx \, dy &= 0 \\ 2\iint_I (uE_x + vE_y + E_t)E_y \, dx \, dy &= 0 \end{aligned} \tag{17}$$

or

$$\begin{aligned} \left(\iint_I E_x^2\right) u + \left(\iint_I E_x E_y\right) v &= -\iint_I E_x E_t \\ \left(\iint_I E_y E_x\right) u + \left(\iint_I E_y^2\right) v &= -\iint_I E_y E_t \end{aligned} \tag{18}$$

These are two linear equations in u and v with integrals of products of brightness gradients as coefficients[4].

We can write the two equations compactly in matrix-vector form as

$$M\mathbf{s} = \mathbf{b} \tag{19}$$

where $\mathbf{s} = (u, v)^T$, M is the 2×2 matrix of coefficients and $\mathbf{b}$ has as components the right hand side values of eq. 18. The pair of linear equations for u and v in eq. 18 can be solved as long as the determinant

$$\Delta = \iint_I E_x^2 \iint_I E_y^2 - \left(\iint_I E_x E_y\right)^2 \tag{20}$$

of the matrix M is non-zero. Then

$$\mathbf{s} = M^{-1}\mathbf{b} \tag{21}$$

or in component form,

$$\begin{aligned} \Delta\, u &= \iint_I E_x E_y \iint_I E_y E_t - \iint_I E_y^2 \iint_I E_x E_t \\ \Delta\, v &= \iint_I E_x E_y \iint_I E_x E_t - \iint_I E_x^2 \iint_I E_y E_t \end{aligned} \tag{22}$$

It is easy to see that the determinant in eq. 20 is zero if the brightness gradient has the same direction everywhere, that is if the ratio $E_y(x, y) : E_x(x, y)$ is constant. It can in fact be shown that this is the *only* situation in which the determinant is zero.

This makes sense, since motion along isophotes does not produce any change in image brightness and if the isophotes are all parallel, then motion of the whole image along that direction produces no change. Hence such motions cannot be detected. The image has to have enough "two-dimensional texture"—or "curvature of isophotes"—to allow us to estimate image velocity (At the same time, motion in the direction of the brightness gradient *can* be estimated).

[4]The extremum found in this way is in fact a minimum, rather than say a maximum, since the integral in eq. 16 can be made as large is we like simply by making u and/or v large—so there can't be a finite maximum.

Discrete Case

In practice, image brightness is known only on a discrete grid of picture cells. In this case the double integrals in the above analysis can be replaced with double sums over all the places where the brightness derivatives are known. So we now wish to minimize

$$\sum_{i=1}^{n-1}\sum_{j=1}^{m-1}(uE_x + vE_y + E_t)^2\,dx\,dy \tag{23}$$

In the sum, E_x, E_y, and E_t are to be read as the estimates of these derivatives at picture cell position $(i + 1/2, j + 1/2)$, as discussed above. Note again that in an image of size $n \times m$ (rows $\times$ columns), derivatives can be estimated at $(n-1) \times (m-1)$ points.

By the way, the expression in eq. 23 is the weighted sum of the squares of the perpendicular distances of the point (u, v) from the constraint lines (defined by eq. 6) in image velocity space. The weights are the squares of the magnitudes of the gradients—so contributions from picture cells where the gradient is stronger are more heavily weighted.

The minimum of eq. 23 occurs at

$$\begin{aligned}\Delta\, u &= \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x E_y \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_y E_t - \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_y^2 \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x E_t \\ \Delta\, v &= \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x E_y \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x E_t - \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x^2 \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_y E_t\end{aligned} \tag{24}$$

where

$$\Delta = \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x^2 \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_y^2 - \left(\sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x E_y\right)^2 \tag{25}$$

To summarize, working with two successive image frames of n rows and m columns, the algorithm is:

(i) Estimate E_x on a grid of $(n-1)$ rows and $(m-1)$ columns using eq. 9. Use corresponding equations to similarly estimate E_y and E_t;

(ii) Accumulate the five sums of products needed in eq. 24 and eq. 25;

(iii) Find u and v from eq. 24, using Δ computed using eq. 25.

The method fails if the determinant is zero, which happens only if the direction of the brightness gradient is the same at all points in the image.

Stability and Accuracy

We have already seen that the equations cannot be solved for the image velocity if the brightness gradient has the same direction everywhere in the image. This occurs when

$$E(x,y) = f(ax+by) \tag{26}$$

for some arbitrary function $f(.)$. In this case, brightness is constant along parallel straight lines defined by $(ax+by) = c$. Then

$$\begin{aligned} E_x(x,y) &= af'(ax+by) \\ E_y(x,y) &= bf'(ax+by) \end{aligned} \tag{27}$$

and so the ratio $E_y : E_x$ is constant.

Clearly there must be borderline cases where the determinant, while not *exactly* zero, is quite small. In this case, while we *can* solve the equations, we cannot rely on the results, since the estimated image velocities will be very sensitive to measurement errors. The reason is that in solving for u and v we divide by the determinant. We conclude that one (crude) measure of the quality of the result is the magnitude of the determinant Δ of the coefficient matrix M. The larger the determinant, the smaller the error in the estimated image velocity.

But this is not the whole story. It is possible, for example, that the velocity component in one direction is quite well constrained while it may not be in another direction. A single measure of quality cannot capture this kind of detail. The question is how much the inverse M^{-1} amplifies small errors in the right-hand side terms of the equations, and how that amplification depends on direction.

This is controlled by the eigenvalues and eigenvectors of the real symmetric matrix M. Let λ_1 and λ_2 be the two eigenvalues, and let $\mathbf{e}_1$ and $\mathbf{e}_2$ be the corresponding (unit) eigenvectors[5]. Then

$$M\mathbf{e}_1 = \lambda_1\mathbf{e}_1 \quad \text{and} \quad M\mathbf{e}_2 = \lambda_2\mathbf{e}_2 \tag{28}$$

Note that the determinant equals the product of the eigenvalues:

$$\Delta = \lambda_1\lambda_2 \tag{29}$$

The inverse matrix M^{-1} has the same eigenvectors—but its eigenvalues are the algebraic inverse of the eigenvalues of M. That is

$$M^{-1}\mathbf{e}_1 = \frac{1}{\lambda_1}\mathbf{e}_1 \quad \text{and} \quad M^{-1}\mathbf{e}_2 = \frac{1}{\lambda_2}\mathbf{e}_2 \tag{30}$$

[5] The two eigenvectors will be orthogonal if the eigenvalues are distinct. If the eigenvalues happen to coincide, then the eigenvectors are not uniquely determined, but one can always pick two that are orthogonal.

Now any vector **b** can be decomposed into a weighted sum of the two eigenvectors (because they form an orthogonal basis):

$$\mathbf{b} = \alpha_1 \mathbf{e}_1 + \alpha_2 \mathbf{e}_2 \tag{31}$$

Then

$$M^{-1}\mathbf{b} = \frac{\alpha_1}{\lambda_1}\mathbf{e}_1 + \frac{\alpha_2}{\lambda_2}\mathbf{e}_2 \tag{32}$$

We conclude that error components in **b** in the direction of the eigenvector with the *smaller* eigenvalue have a *larger* effect than error components in the direction of the other eigenvector. Correspondingly, the image velocity is less well constrained in the direction of the eigenvector corresponding to the smaller eigenvalue.

So another measure of the quality of the estimated image velocity is the magnitude of the smaller eigenvalue. The larger this is, the less the error amplification. Since the determinant is the product of the two eigenvalues, it also will be small when the smaller of the two eigenvalue is small. But the determinant is also effected by the larger of the two eigenvalues, so the magnitude of the smaller eigenvalue by itself—while requiring more computation—is a more useful measure of quality.

Finding the Eigenvalues

The eigenvalues of the real symmetric matrix

$$M = \begin{pmatrix} a & b \\ b & c \end{pmatrix} \tag{33}$$

are the roots of the characteristic polynomial[6]

$$\lambda^2 - (a+c)\lambda + (ac - b^2) = 0 \tag{34}$$

Hence

$$\lambda_{+,-} = \frac{(a+c) \pm d}{2} \tag{35}$$

where

$$d = \sqrt{(a-c)^2 + 4b^2} \tag{36}$$

In our case,

$$a = \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x^2, \quad b = \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_x E_y, \quad c = \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} E_y^2 \tag{37}$$

So a and c are non-negative, and so is d. Further $\lambda_+ \geq 0$, $\lambda_- \leq \lambda_+$ and

$$|\lambda_-| \leq \lambda_+ \tag{38}$$

So we can easily find the smaller eigenvalue, the one that is a measure of quality of image velocity estimation. It is only a bit more work to compute than the determinant, $\Delta = ac - b^2$, and it is more useful.

[6] The characteristic polynomial is the determinant of the matrix $(M - \lambda I)$.

Finding the Eigenvectors

If we wish to find the directions in which the image velocity is most constrained and least constrained, then we also have to determine the eigenvectors of the matrix M. To find the eigenvector corresponding to the eigenvalue λ we solve the homogeneous linear equations

$$\begin{pmatrix} a-\lambda & b \\ b & c-\lambda \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{39}$$

Now

$$\begin{aligned} a - \lambda_{+,-} &= \frac{(a-c) \mp d}{2} \\ c - \lambda_{+,-} &= \frac{(c-a) \mp d}{2} \end{aligned} \tag{40}$$

so solutions of eq. 39 can be written in either of the two forms

$$\begin{pmatrix} 2b \\ (c-a) \pm d \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} (a-c) \pm d \\ 2b \end{pmatrix} \tag{41}$$

The magnitude squared of the first form is $2d(d \pm (c-a))$, while that of the second is $2d(d \pm (a-c))$. Correspondingly, the unit eigenvectors can be written in either of the forms

$$\frac{1}{\sqrt{2d}\sqrt{d \pm (c-a)}} \begin{pmatrix} \pm 2b \\ d \pm (c-a) \end{pmatrix} \tag{42}$$

or

$$\frac{1}{\sqrt{2d}\sqrt{d \pm (a-c)}} \begin{pmatrix} d \pm (a-c) \\ \pm 2b \end{pmatrix} \tag{43}$$

In each occurence of the $\pm$ sign, the plus sign is chosen if the plus sign is chosen in the expression for the eigenvalue (i.e. λ_+). The direction with the weaker constraint is that associated with λ_-, while stronger constraint on the image velocity exists in the direction associated with λ_+.

References

[1] Horn, B.K.P. & B.G. Schunck, "Determining Optical Flow," *Artificial Intelligence*, Vol. 16, No. 1–3, August 1981, pp. 185–203.

[2] Horn, B.K.P. "Parallel Networks For Machine Vision" Artifical Intelligence Memo 1071, Dec 1988. also in *Artificial Intelligence at MIT: Expanding Frontiers*, edited by Patrick H. Winston and Sarah A. Shellard, MIT Press, Vol. 2, pp. 531–573, 1990, and in *Research Directions in Computer Science: An MIT Perspective*, edited by A. Meyer, J.V. Guttag, R.L. Rivest, & P. Szolovits, MIT Press, pp. 277-315, 1991. and in *Vision Chips: Implementing Vision Algorithms with Analog VSLI Circuits*, C. Koch & H. Li (Eds.) IEEE Computer Society Press, 1995.

[3] Horn, B.K.P. and B.G. Schunck "Determining Optical Flow — a Retrospective" *Artificial Intelligence*, Vol. 59, No. 1–2, February 1993, pp. 81–87.