The Facet Approach To Optic Flow

Robert M. Haralick and Jong Soo Lee

 Departments of Computer Science and Electrical Engineering
 Virginia Polytechnic Institute and State University
 Blacksburg, Virginia 24061

Abstract

The facet approach requires low level image processing techniques to be based on fitting each local image neighborhood with a function and interpreting all processing in terms of what the processing does to this locally fit function. Using the facet approach we develop a different meaning of the usual optic flow equation. We show that it represents the intersection line of the isocontour plane with a successive image frame. The intersection line on the successive frame contains the possible match points. A unique match point can be selected by requiring it to have the same brightness as the given pixel. We show that this procedure amounts to assuming that all derivatives of third or higher order are negligible and that gray tone intensity and first partial derivatives in row, column, and time must match.

I Introduction

In this paper we consider the case of a camera in uniform translational motion in a static scene. In section II we derive the optic flow equation for uniform translational motion and show how from the optic flow field it is possible to compute the camera velocity parameters and the depth, both to within an arbitrary scale factor.

Determination of the optic flow field is usually done by matching corresponding points on successive image frames. This kind of technique suffers from a potentially expensive combinatorial complexity problem. In section III we apply a facet model technique to the problem of estimating the optic flow field. We show how the first order derivative optic flow equation represents the intersection line of the isocontour plane with a successive image frame. To select a unique match point on this line we require that the gray tone intensities match. We show that this procedure amounts to requiring that gray tone intensities match, and first order partials in row, column, and time match. The complexity of the technique is linear in the number of pixels on the image. There is no combinatorial matching. In section IV we briefly mention some of the existing approaches for matching. In section V we present results.

II Optic Flow Geometry

Consider an image created by a camera in constant motion, the velocity of the camera being \((a_x, a_y, a_z)\) in the x, y, and z directions respectively. The motion of the camera causes the position of pixels in the image to move. An image in which each pixel contains the velocity vector describing the motion of that pixel is called the optic flow image. We give a brief derivation of the optic flow.

Our perspective geometry model places the lens at the origin looking down the y-axis. The image plane is a distance of \(f\) in front of the lens. Thus a point \((x,y,z)\) in the 3D world will have an x-position \(x_p\) on the image given by

\[
x_p = f \frac{x}{y-a_z t} \quad (1)
\]

At \(t=0\), a point \((x',z')\) on the image corresponds to the ray \(\lambda \begin{pmatrix} x' \\ z' \end{pmatrix}\) where \(\lambda\), the unknown parameter, is most directly related to the depth \(y\) of the 3D point by the relation \(\lambda = y/f\). After substituting \(\lambda x'\) for \(x\) and \(\lambda z'\) for \(y\) in equation (1) there results

\[
x_p = f \frac{\lambda x' - a_x t}{\lambda f - a_z t} \quad (2)
\]

The velocity \(u(x',z')\) of point \((x',z')\) at \(t=0\) can be obtained as

\[
\partial_x \frac{\partial}{\partial t} \left[ (\lambda f-a_z t)(-a_x - (\lambda x'a_x t)(-a_y) \right] \quad (3)
\]

\[
\partial_t \quad = f \frac{\lambda f-a_z t}{y^2}
\]
Figure 5d shows the average difference vectors which resulted from setting the separation threshold to 10 pixels and the length threshold to 3 pixels. A plot of the error function produced using these threshold values is shown in figure 5e. The local search found a minimum at (52, 75). The correct position of the intersection of the translational axis with the image plane for the second image was determined to be at (57.97, 74.58). Since the focal length was rather long, the determined translational axis was well within 5 degrees of the actual one.

Figure 5c

This research was supported by DARPA grant N000014-82-K-04064 to the MOTION group at UMASS.

References


evaluated at $t=0$. Then, we have
\[
u(x', z') = \left. \frac{\partial x}{\partial t} \right|_{t=0} = \frac{-a_x + a_y}{\lambda f} \lambda x'
\]
(4)

The case for the $z$-velocity $v(x', z')$ is similar
\[
v(x', z') = \left. \frac{\partial z}{\partial t} \right|_{t=0} = \frac{-a_z + a_y}{\lambda f} \lambda z'
\]
(5)

For a camera in motion where $a_x \neq 0$, there will be one point $(x_0, z_0)$ on the image whose motion will be zero. To determine this point set
\[
\left. \frac{\partial x}{\partial t} \right|_{t=0} = \left. \frac{\partial z}{\partial t} \right|_{t=0} = 0
\]
(6)

to obtain
\[
\begin{align*}
x_0 &= \frac{a_y}{a_x} \\
z_0 &= \frac{a_y}{a_z}
\end{align*}
\]
(7)

This point is called the focus of expansion or contraction depending on whether the camera is moving toward or away from the scene.

To solve for $a_x, a_y, a_z$, set the parameter $\lambda(x', z')$ to an initial appropriate constant depending on scale and solve in a least square sense the system of equations:
\[
\begin{align*}
\lambda(x', z')u(x', z') &= -a_x + \frac{\lambda}{f} x' \\
\lambda(x', z')v(x', z') &= -a_z + \frac{\lambda}{f} z'
\end{align*}
\]
(8)

This yields
\[
\begin{align*}
\sum \lambda(x', z')[u(x', z')x + v(x', z')z] \\
- \sum u(x', z')\lambda(x', z') \sum x' \\
- \sum v(x', z')\lambda(x', z') \sum z' \\
- \sum u(x', z')\lambda(x', z') \sum \frac{x'^2}{x'} \\
- \sum v(x', z')\lambda(x', z') \sum \frac{z'^2}{z'}
\end{align*}
\]
(9)

\[
a_x = \frac{\sum x' - \sum u(x', z')\lambda(x', z')} {\sum 1}
\]
(10)

where all summations are over all $(x', z')$ in the image domain. If the scale constant for $\lambda(x', z') = k$, an unknown constant, the velocity components $a_x, a_y,$ and $a_z$ will all have the same multiplicative constant $k$. In this case, the velocity magnitude is not determined, but its direction is.

A better solution than the assumed constant $\lambda$ may be obtained by iterating for reduced residual error by redefining $\lambda$ to be a function of the estimated velocities.

\[
\lambda(x', z') = \left[ \frac{\left(-a_x + \frac{a_y}{f} x'\right)^2 + \left(-a_z + \frac{a_y}{f} z'\right)^2} {u(x', z')^2 + v(x', z')^2} \right]^{1/2}
\]
and then solving for $a_x, a_y,$ and $a_z$ in terms of the new $\lambda(x', z')$. This new $\lambda$ can be substituted into equation (9) for a better estimate of the velocities. Smoothness in 3D surface can be insisted upon by taking any of the $\lambda_i$ as an image and performing a slope facet iteration on it (Haralick and Watson, 1981).

III Calculation of Optic Flow From Image Sequence

In this section we discuss the calculation of optic flow in a time sequence of image frames and illustrate the facet approach to the optic flow computation.

Consider the case of a one dimensional sequence of frames as shown in figure 1. These frames are obviously translates of one another with a uniform motion. Instead of considering a correlation search to match each point on one frame with its corresponding place on the next frame, consider the sequence of frames as an image each of whose rows correspond to one frame. Corresponding points on different frames have the same intensity. Thus where the one-dimensional frames are organized as an image, the corresponding points will be on equal intensity contour lines as shown on figure 2. The equal intensity contour line any point is on is easily computed as the line orthogonal to the gradient direction at that point. Thus by fitting a function to the image intensities in a local neighborhood about a point, as the facet model prescribes, and determining the gradient direction from the fit, the equal intensity contour line through the point can be determined. The match point on the next frame can be obtained without any search just as the intersection of the equal intensity contour line passing through the point with the next frame or row.
Figure 1. Illustrates a one-dimensional waveform which is translating in time.

Figure 2. Shows the equal intensity contour lines which match corresponding points.

In a time-varying image sequence, the situation is similar, only the geometry is in a one dimensional higher space, a 4-dimensional space. To understand this geometry, fix attention on one pixel on one frame. Use the 3D neighborhood (by row, by column, by image frame number) around the pixel and fit a function to the gray tone intensities in the 3D neighborhood. From the function, fit determine the gradient vector at the given pixel position. The plane which is normal to the gradient vector is the equal intensity contour plane passing through the given pixel. To determine possible match points on the next frame intersect the equal intensity contour plane with the next image frame. As shown in figure 3, the intersection is a line. The match point can be any place on this line. To determine it uniquely, find that position on the line whose gray tone intensity is equal to the graytone intensity of the given pixel.

III.1 Example

Consider a local 2D neighborhood whose gray tone intensity function appears like a paraboloid of the form \((r+1)^2 + (c+2)^2\). Suppose that image frames are taken each second and that due to the camera motion, the paraboloid translates each successive frame by three rows and one column. Then upon fitting the gray tone intensities in a local 3D neighborhood whose center pixel has coordinates \((0,0,0)\) in a relative coordinate frame, we determine the function

\[ f(r,c,t) = (r-3t+1)^2 + (c+t+2)^2 \]

Thus the paraboloid is translating by \((3\text{ rows},-1\text{ column})\) on successive frames.
The partial derivatives of \( f \) are
\[
\begin{align*}
\frac{\partial f}{\partial r} &= 2(r-3t+1) \\
\frac{\partial f}{\partial c} &= 2(c+t+2) \\
\frac{\partial f}{\partial t} &= 2(r-2t+1)(-3) + 2(c+t+2)
\end{align*}
\]
Evaluating these partials at \((0,0,0)\) yields the gradient vector at the given pixel which is located at the origin
\[
\text{grad } f = \begin{pmatrix} 2 \\ 4 \\ -2 \end{pmatrix}
\]
The plane passing through \((0,0,0)\) and orthogonal to this gradient vector is given by
\[
2r + 4c - 2t = 0
\]
Intersecting this plane with the next frame \((t=1)\) produces the line
\[
2r + 4c - 2 = 0
\]
The gray tone intensity at \((0,0,0)\) is given by
\[
f(0,0,0) = 5.
\]
To find the match point, find that \((r,c)\) simultaneously satisfying the two equations
\[
r + 2c - 1 = 0
\]
\[
f(r,c,1) = (r-2)^2 + (c+3)^2 = f(0,0,0) = 5
\]
Substituting \(r = 1 - 2c\) into \((r-2)^2 + (c+3)^2 = 5\) yields the quadratic equation \((c+1)^2 = 0\) from which \(c = -1\) and \(r = 3\), the correct translation parameters.

III.2 Translational Motion

As the example suggests, the difficulty of the computation might be in determining a real root of a polynomial. It is natural to wonder, therefore, whether it is possible to have polynomials with no real roots. We demonstrate here that for the case of translational motion, there is no possibility of the polynomial roots being only complex. To see this express the local fitted functions \(f(r,c,t)\) as
\[
f(r,c,t) = g(r-at, c-\beta t) \quad (11)
\]
explicitly indicating that the dependence between \(r, c,\) and \(t\) is constrained to translation.

The equation of the isodensity contour plane passing through \((0,0)\) is given by
\[
(r-at) \frac{\partial g}{\partial r} + (c-\beta t) \frac{\partial g}{\partial c} = 0 \quad (12)
\]
where
\[
\begin{align*}
\frac{\partial g}{\partial r} &= -\frac{\partial g}{\partial r} \\
\frac{\partial g}{\partial c} &= -\frac{\partial g}{\partial c}
\end{align*}
\]
At \(t = t_0\), this plane cuts the \(t = t_0\) frame producing the line
\[
(r-at_0) \frac{\partial g}{\partial r} + (c-\beta t_0) \frac{\partial g}{\partial c} = 0
\]
At the desired point \((r,c)\) on this line we must satisfy the match condition
\[
g(r-at_0, c-\beta t_0) = g(0,0)
\]
Assuming \(g\) is a function for which all partial derivatives exist we may represent \(g(r-at, c-\beta t)\) by its Taylor series around \((0,0)\)
\[
g(r-at, c-\beta t) = g(0,0) + (r-at_0) \frac{\partial g}{\partial r} + (c-\beta t_0) \frac{\partial g}{\partial c}
\]
\[
+ \frac{(r-at_0)^2}{2} \frac{\partial^2 g}{\partial r^2} + (r-at_0)(c-\beta t_0) \frac{\partial^2 g}{\partial r \partial c} + \frac{(c-\beta t_0)^2}{2} \frac{\partial^2 g}{\partial c^2} + \ldots
\]
Substituting \(g(r-at, c-\beta t)\) for \(g(0,0)\) and
substituting \(- (c-\beta t_0) \frac{\partial g}{\partial c} / \frac{\partial g}{\partial r}\) for \(r-at_0\) yields
\[
(c-\beta t_0)^2 \left( \frac{\partial^2 g}{\partial c^2} \frac{\partial^2 g}{\partial r^2} + \frac{\partial^2 g}{\partial c \partial r} + \frac{\partial^2 g}{\partial r^2} \right) + (c-\beta t_0)^3 [\ldots]
\]
Factoring out \((c-\beta t_0)^2\) from the left hand side and noting that the right hand side is zero permits us to write \((c-\beta t_0)^3 = 0\) from which we can solve for the double real root \(c = \beta t_0\)

III.3 Comparison

There is a relationship between this procedure and the usual optic flow equation. Letting \(f_r, f_c\), and \(f_t\) designate the partial derivatives of \(f\) with respect to \(r, c,\) and \(t,\) evaluated at the origin, the equation of the isocontour plane is
given by
\[ r f_r + c f_c + tf_t = 0 \] (14)

Intersecting this plane with the next image plane which is taken at \( t_0 \) seconds latter produces the line
\[ \begin{align*}
  r c \\
  -f_t - f_r + f_c \\
  t_0 & t_0 
\end{align*} \] (15)

Equation (15) is the usual optic flow equation (Born and Schunk, 1980). The quantity \( r/r_0 \) represents a movement of \( r \) rows over \( t_0 \) seconds and is therefore the row velocity. Likewise \( c/t_0 \) represents the column velocity.

The difference in what we have done is that we have given equation (15) an enlarged meaning. It is the equation of a line containing the possible match points on the \( t_0 \) image frame. But since the match point must have the same brightness, we use the additional constraint that the match point \( (r,c) \) must satisfy
\[ f(r,c,t_0) = f(0,0,0) \] (16)

the equal brightness constraint. This brightness constraint is used in the usual derivation of the optic flow equation so it would seem to be superfluous to use again. From our perspective we see that the isodensity contour plane is really only isodensity at the origin and as it moves away from the origin, it must be regarded as an approximation. Thus the intersection line of the successive frame is not guaranteed to have all its points be of the same brightness as the given pixel. The match condition just tells us to select that point on the line having the same brightness as the given pixel.

III.4 Why It Works

In this section we give a detailed explanation of why the procedure works. We assume that all derivatives of third or higher order are negligible and that the match conditions consist of matching gray tone intensity and gray tone first partials in rows, columns, and time.

Let \( f \) with a subscript designate the corresponding partial derivative of \( f \) evaluated at \( r = c = t = 0 \). A Taylor series of \( f \) about \( (0,0,0) \) neglecting third or higher order terms is given by
\[ f(r,c,t) = f(0,0,0) + rf_r + cf_c + tf_t \] (17)

\[ + \frac{r^2}{2} f_{rr} + rcf_{rc} + \frac{c^2}{2} f_{cc} + rtf_{rt} + ctf_{ct} + \frac{t^2}{2} f_{tt} \]

A pixel \((r,c)\) having relative neighborhood coordinates on relative time image \( t \) matches pixel \((0,0)\) on time image 0 if

(1) \[ f(r,c,t) = f(0,0,0) \]

(2) \[ \frac{\partial f}{\partial r} \frac{\partial f}{\partial r} \\
-(r,c,t) = -(0,0,0) = f_r \\
\frac{\partial f}{\partial r} \frac{\partial f}{\partial r} \\
-(r,c,t) = -(0,0,0) = f_c \\
\frac{\partial f}{\partial r} \frac{\partial f}{\partial r} \\
-(r,c,t) = -(0,0,0) = f_t \]

Condition (1) states that the gray tone intensities must match. Condition (2) and (3) states that the gray tone spatial pattern around the original and the match pixel must match. Condition (4) states that since the motion is uniform with no acceleration the gray tone time derivatives must match.

Applying these constraints to the Taylor series we have, respectively,

\[ f_r + cf_c + tf_t = 0 \] (18)

\[ f_r + c f_r + r f_t + r c f_{rc} + c f_{cc} + r t f_{rt} + c t f_{ct} + \frac{t^2}{2} f_{tt} = 0 \]

Multiplying equation (19) by \( r \), equation (20) by \( c \), equation (21) by \( t \) and adding yields

\[ r^2 f_{rr} + 2 r c f_{rc} + c^2 f_{cc} + 2 r t f_{rt} + 2 c t f_{ct} + t^2 f_{tt} = 0 \] (22)

Substituting this back into equation (18) yields

\[ f_r + c f_c + tf_t = 0 \] (23)

the usual optic flow equation! Thus, the technique of using equation (23) and the gray tone intensity match condition (18) in essence works because it assumes that all first partials are matching. However, now we see that there need not be any problem of root finding. We just need to
solve the overconstrained system of equations

\[
\begin{bmatrix}
 f_x & f_c & 0 \\
 f_{rr} & f_{rc} & 0 \\
 f_{rc} & f_{cc} & 0 \\
 f_{rt} & f_{ct} & 0 \\
 f_{tt} & f_{tt} & 0
\end{bmatrix}
\begin{bmatrix}
 x_t \\
 x_t \\
 x_t \\
 x_t \\
 x_t
\end{bmatrix} =
\begin{bmatrix}
 f_t \\
 f_t \\
 f_t \\
 f_t \\
 f_t
\end{bmatrix}
\]

(24)

for the row column position \((r,c)\) on the specified image \(t\).

IV Brief Matching Literature Review

Matching frames is an old image processing problem. Classically, it was solved by translating one image against the other until the correlation between the two images was highest. An equivalent calculation can be done through the use of Fourier Transform. Barnea and Silverman (1972) showed how to speed up the search by essentially not doing calculations on positions where errors must exceed the best error so far. These techniques work only for translation of one image relative to the other.

In moving images, the motion is not the same all over the image. Correlation techniques are not appropriate. Martin and Aggarwal (1979) use boundary information as the basis for matching. Bernard and Thompson (1980) used a disparity analysis technique for matching. Ayala, Orton, Larson, and Elliott (1982) use a symbolic technique for matching. Jacobus, Chien, and Selander (1980) use a graph matching technique. Aggarwal, Davis, and Martin (1981) review techniques for establishing corresponding points on images. The problem with most of these techniques is that they must employ some combinatorial computation to establish the match. This kind of computation is very expensive.

Techniques which do not involve combinatorial matching include Limb and Murphy (1975) who relate image intensity changes over time to spatial gradient and Fennema and Thompson (1979) who use a gradient intensity transform method. Both these techniques are similar to the one presented in this paper in that they establish the match using only local neighborhood analysis.

V Results

To confirm that our theory works, we tested our algorithm on 3 kinds of image sequences. The image sequences describe the movement of an ellipsoid in translation, magnification, and rotation. The time interval between two consecutive images in a sequence corresponds to one pixel difference in an image.

To compute an optic flow vector of a pixel on the image at \(t=0\), we determined the underlying function over its 3-D neighborhood using a 3-D cubic discrete orthogonal polynomial basis, and, next, derived the 4 constraining equations (Eq24) on the row and column components of the optic flow vector at the center of the pixel. To solve the over-constrained equations, we, first, obtained two singular values using the Singular Value Decomposition routine of the Linpack, and, next, determined the least square solution from the singular values.

To the time sequence of an ellipsoid moving with the velocity of \(v=1\) and \(c=-0.8\) shown in Fig 4, we applied the above method with the 3-D neighborhood \((5x5x5)\) and obtained the optic flow image shown in Fig 5. At the pixels on or near the boundary of the ellipsoid, the optic flow vector obtained does not show the correct movement of the ellipsoid. The neighborhoods contain a mixture of stationary background ellipsoid, thereby providing inconsistent information for fitting. The reason for the inconsistency is that the center pixel may be in the stationary background but it has neighbors which are not. These neighbors generate an estimated surface which are not. These neighborhoods generate an estimated surface which has some curvature for the center pixel.

To reject an optic flow vector obtained from such a neighborhood, we compute the ratio of two principal curvatures from the underlying gray tone intensity surface determined at \(t=0\). From the histogram of this curvature ratio over all the neighborhoods in the image sequence shown in Fig 6, we can determine a threshold value for the ratio at about 0.05. Fig 7 illustrates the result. The pixels which still have incorrect directions correspond to neighborhoods with large fitting errors over the center pixel. Fig 8 illustrates a histogram of the center pixel fitting error. Thresholding the original optic flow image with the curvature ratio of 0.05 and rejecting the optic flow vectors obtained from the underlying function having fitting error of more than 1, we have the optic flow image shown in Fig 9. Rejecting the vectors having fitting error of more than 0.01, we have the optic flow image shown in Fig 10.

For the time sequence of the ellipsoid moving backwards with the magnification factor 0.95 shown in Fig 11, we obtain the optic flow image shown in Fig 12 where we thresholded the original optic flow image with the ratio 0.05 and rejected the vectors obtained from underlying function having fitting error of more than 1. In the same way, for the time sequence of the ellipsoid rotating clockwise with the angular velocity \(0.1\) radian shown in Fig 13, we obtain the optic flow image shown in Fig 14.

References


Fig 4 Time sequence of an ellipsoid in translation

Fig 5 Original optic flow image

Fig 6 Histogram of curvature ratio

Fig 7 Thresholded optic flow image
Fig 8 Histogram of fitting error

Fig 9 Optic flow image without rejected vectors

Fig 10 Optic flow image

Fig 11 Time sequence of an ellipsoid moving backwards
Fig 12 Optic flow image obtained from Fig 11

Fig 13 Time sequence of an ellipsoid in rotation

Fig 14 Optic flow image obtained from Fig 13