International Journal of Computer Vision, 14, 67–81 (1995)

Robust Computation of Optical Flow in a Multi-Scale Differential Framework

JOSEPH WEBER AND JITENDRA MALIK
Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, CA 94720
email: jweber@eecs.berkeley.edu, malik@eecs.berkeley.edu

Received April 14, 1993; Revised December 22, 1993

Abstract. We have developed a new algorithm for computing optical flow in a differential framework. The image sequence is first convolved with a set of linear, separable spatiotemporal filter kernels similar to those that have been used in other early vision problems such as texture and stereopsis. The brightness constancy constraint can then be applied to each of the resulting images, giving us, in general, an overdetermined system of equations for the optical flow at each pixel. There are three principal sources of error: (a) stochastic error due to sensor noise (b) systematic errors in the presence of large displacements and (c) errors due to failure of the brightness constancy model. Our analysis of these errors leads us to develop an algorithm based on a robust version of total least squares. Each optical flow vector computed has an associated reliability measure which can be used in subsequent processing. The performance of the algorithm on the data set used by Barron et al. (IJCV 1994) compares favorably with other techniques. In addition to being separable, the filters used are also causal, incorporating only past time frames. The algorithm is fully parallel and has been implemented on a multiple processor machine.

1 Introduction

A number of different approaches to recovering optical flow have been proposed. These can be roughly grouped into correlation, energy and differential approaches. A recent survey is due to Barron et al. (1993 and 1994) where the different approaches were compared on a series of synthetic and real images. They found that a phase-based approach by Fleet and Jepson (1990) performed the best numerically.

We have developed a new algorithm for computing optical flow in the differential framework which performs comparably to the Fleet and Jepson approach but with less computational cost and a higher density of estimates. We start with a multi-channel filtering of the intensity response, thus producing an overconstrained system of equations in the components of the optical flow. The convolution with a series of filters is a common starting point for a number of early vision tasks such as edge detection, stereopsis and texture discrimination (Bergen and Adelson 1988; Canny 1986; Jones and Malik 1992; Jones and Malik 1992; Malik and Perona 1990; Turner 1986 and Heeger 1988). In Section 3 we analyze the sources of error in the differential method as falling into 3 categories: (a) stochastic error due to sensor noise, (b) systematic error due to large displacements and (c) model error where the underlying model is violated. This analysis leads to an algorithm based on a robust version of total least squares. This algorithm is outlined in Section 4.

The implementation is described in Section 5. In Section 6 the algorithm is tested on the series of synthetic and real sequences used by Barron et al. Thus a direct comparison between our work and others can be made. A high density of estimates was found for all sequences, implying that the "aperture problem" occurs rarely in most images. A confidence measure is available as a byproduct of the total least squares formulation. Through a simple experiment we demonstrate that this measure is related to the estimated accuracy of the motion vector. We

also look at a scale pyramid implementation of the filter responses in this section to demonstrate that this more efficient method of computing multiple scale responses does not degrade the performance significantly.

In the Appendix we describe a parallel implementation on a multiple processor machine and examine the speedup of the algorithm. This demonstrates that a real-time parallel version of the algorithm may be possible.

2 The Differential Constraint Equation

The starting point of differential approaches to the estimation of optical flow is the *brightness constancy assumption*. The image brightness of the projection of a single point is assumed to remain constant with time. This is strictly true only in the idealized context of lambertian surfaces being viewed by a moving camera. It is a reasonable approximation for a wide range of practical situations. The brightness constancy assumption implies:

$$I(x, y, t) = I(x + u\delta t, y + v\delta t, t + \delta t) \qquad (1)$$

where $I(x, y, t)$ is the brightness or some function of the brightness at location (x, y) and time t. The vector field $\mathbf{v} = (u, v)$ is the optical flow and is a function of image coordinates (x, y). In the limit as $\delta t \rightarrow 0$ we get the constraint equation

$$I_x u + I_y v + I_t = 0 \qquad (2)$$

where I_x, I_y and I_t are partial derivatives with respect to space and time, evaluated at the point (x, y, t). This equation can also be derived from the differential form of the constancy assumption, $dI/dt = 0$. This constraint was introduced by Fennema and Thompson (1979). The use of the equation in practice will require that we be able to estimate partial derivatives I_x, I_y and I_t. This can be done if there is no temporal or spatial aliasing. Temporal aliasing however is common, leading us to use instead equation (3), the discrete variant of the constancy assumption.

For finite time intervals, we make a Taylor expansion of the right hand side of equation (1) to obtain:

$$I_x u\delta t + I_y v\delta t + I_t\delta t = \mathcal{O}(\mathbf{v}^2\delta t^2) \qquad (3)$$

The right hand side of equation (3) represents the remaining terms of the Taylor expansion. This contains products of higher spatial and temporal derivatives of the brightness function as well as higher powers of the displacements. The partial derivatives are calculated from discrete finite differences. By a suitable choice of units ($\delta t = 1$), this equation can be written in a more compact form:

$$\nabla I \cdot \mathbf{v} + I_t = \mathcal{O}(\mathbf{v}^2) \qquad (4)$$

The right hand side is usually assumed small and set to zero. The validity of ignoring the right hand side of equation (3) is dependent on the spatial frequency content of the intensity pattern and the magnitude of the displacement ($\mathbf{v}\delta t$).

The differential constraint equation (2) has been used in motion detection for some time (Fennema and Thompson 1979). It is a single equation in the two unknowns which forms a single constraint line in velocity space. Any velocity on this line satisfies the constraint. This was called the "aperture problem" since it implies that locally the velocity can not be determined uniquely. Horn and Schunck (1981) introduced a smoothness constraint in order to solve uniquely for displacement. A number of other authors (Tretiak and Pastor 1984; Nagel 1987; Uras et al. 1988; Verri et al. 1990 and Srinivasan 1990) produced two or more linear equations in u and v by assuming constancy of partial derivatives and other functions of the intensity. A third approach (Lucas and Kanade 1981 and Campani and Verri 1990) is to assume the velocity field is locally constant and to combine constraint equations from neighboring pixels. A review of these and other approaches such as correlation and energy models can be found in (Baron et al. 1993).

In our approach, we first convolve the image sequence with a set of linear spatio-temporal filter kernels, $f_i(x, y, t)$. These are Gaussian derivatives of first or second order at a number of orientations and scales (see Figure 5). These are the same filter kernels used in previous work on early vision, such as in stereopsis and texture discrimination (Jones and Malik 1992; Jones and Malik 1992 and Malik and Perona 1990). Each convolved image, $I_i = I * f_i$, has its own

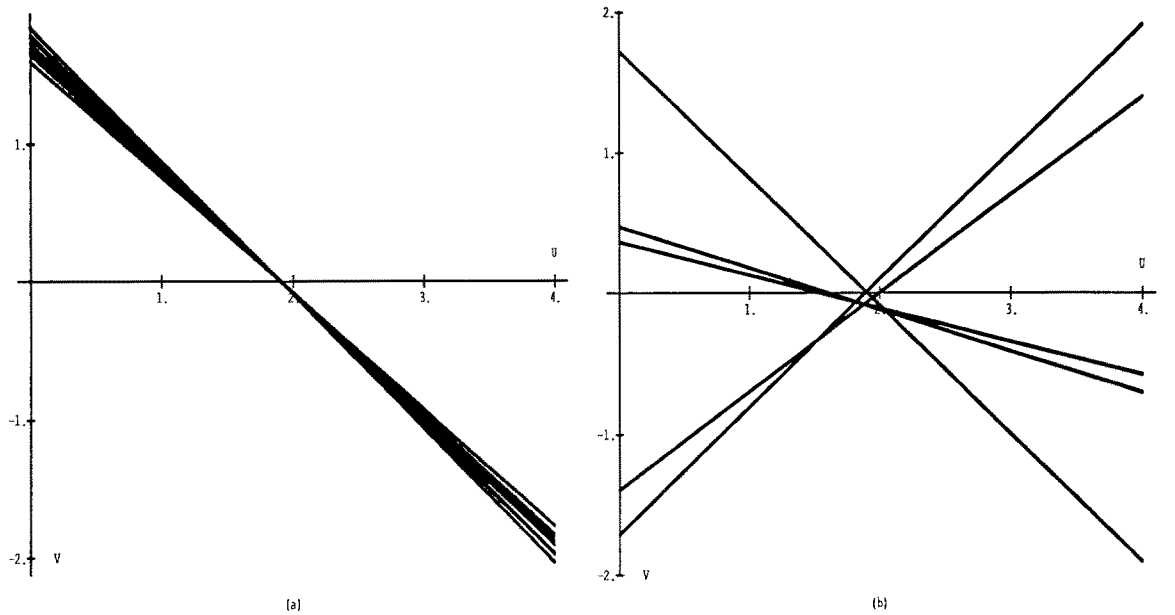

Fig. 1. Constraint equations produced by (a) using the constraint equations from a 3×3 neighborhood about the center pixel and (b) the constraint equations from the 5 filters used in the paper. The signal was white noise filtered with a Gaussian of standard deviation 8 pixels translating 2 pixels to the right per frame.

constraint equation of the form (3). This results in an overconstrained system of equations in the unknowns u and v.

$$\begin{pmatrix} I_{1x} & I_{1y} \\ I_{2x} & I_{2y} \\ \cdot & \cdot \\ I_{nx} & I_{ny} \end{pmatrix} \cdot \mathbf{v} = \begin{pmatrix} -I_{1t} \\ -I_{2t} \\ \cdot \\ -I_{nt} \end{pmatrix} + \vec{\mathcal{O}}(\mathbf{v}^2) \qquad (5)$$

The spatio-temporal partial derivatives, I_{ix}, I_{iy}, I_{it}, can be considered as the result of convolution of the image sequence I with linear, spatio-temporal filter kernels since $I_{ix} = (I * f_i)_x = I * f_{ix}$. Defining the matrix $\mathbf{A}$ and vector $\mathbf{b}_t$ the system of equations can be written as:

$$\mathbf{A} \cdot \mathbf{v} = -\mathbf{b}_t + \mathcal{O}(\mathbf{v}^2) \qquad (6)$$

The spatial extent of the filters brings in information from neighboring pixels so the aperture problem exists only for degenerate cases.

2.1 Comparison to Region Techniques

The multiple filters approach is very similar to region approaches (Lucas and Kanade 1981; Campani and Verri 1990 and Wang et al. 1992) where multiple constraint lines are obtained from the set of constraints at neighboring pixels. However, for band-pass signals the constraint equations in a local region will be very similar since the spatial first derivatives vary slowly. The resulting measurement matrix is close to singular. The orthogonal filter kernels used in the multiple filters approach can produce constraint equations which are more orthogonal (except in degenerate cases where there is a true aperture problem). An extreme example can be seen in Figure 1. The constraint lines produced from a small neighborhood for a band-pass signal are almost parallel, while the constraint lines from the filters are better distributed.

Of course the constraint lines from each point in the support of the filter kernels provides the same information as the filters. Since we are using linear filters, it is just a change of basis. However, the orthogonality of the filters can produce in fewer constraint lines a stable measurement matrix and thus a more stable estimate. Each filtered image samples a different part of the original signal's spatial spectrum and

thus gradient directions can be very different. Only when the spectrum of the original signal is degenerate (lies on a line in frequency space for example) will the gradients be parallel.

3 Noise Considerations

The fundamental problem now is to solve the overconstrained system of equations (6) so as to obtain as accurate an estimate of **v** as possible. We begin by analyzing the sources of error.

1. *Stochastic error.* In the presence of sensor noise, we expect that the measurements of I_{ix}, I_{iy}, I_{it}, the spatiotemporal derivatives of $I * f_i$, would be corrupted with noise. We will make the standard convenient assumption that sensor noise is independent from pixel to pixel and has a Gaussian distribution. This is analyzed further in subsection 3.1.
2. *Systematic error for large displacements.* The system of equations (6) is derived by neglecting second order terms in the displacement, so we expect systematic errors whenever the local velocity is large. The magnitude of the error is dependent on a number of factors including the scale of the filter being used and the local spatial frequencies present in the image neighborhood. This is analyzed further in subsection 3.2.
3. *Errors due to model failure.* In subsection 3.3 we group together the errors that arise due to violation of certain key assumptions of the differential approach: (a) Constancy of image brightness, which is not strictly true whenever there is a significant specular component, and (b) that the optical flow field is locally constant over the support of the filters, which is not true if the filter support straddles a depth discontinuity or when there is a significant rotational or divergence component in the flow field.

3.1 Stochastic Error and Total Least Squares

If we knew that the errors were confined to the measurements of I_{ti}, i.e. the right hand side of the system (6), then the correct approach is well known from estimation theory. We find the classical weighted least squares solution which from the Gauss-Markov theorem is the best one can do[1]. The weight matrix can be determined by examining the covariance matrix of the filters f_{ti}.

However the classical least squares method makes the implicit assumption that the measurements on the left hand side I_{xi}, I_{yi} are error-free and that the errors are confined to the measurements on the right hand side I_{ti}. This assumption is not true, impelling us to use the *total least squares* method. Total least squares is also known as *orthogonal regression* or *errors-in-variables regression* (Van Huffel and Vandewalle 1991).

The essential difference between classical least squares and total least squares can be made clear by a simple example. Suppose we wish to fit a line to a group of points, (x_i, y_i). In classical least squares we wish to find the values of the slope and intercept, (m, b), which minimize the sum squared difference between the y_i and the predicted y.

$$\min_{b,m} \sum_i (y_i - mx_i - b)^2 \tag{7}$$

This minimizes the vertical distances between the line and the measurements y_i. It assumes the variables x_i are error free and all noise is contained in the y_i. Total least squares allows for errors in the x_i variables too. It wishes to minimize the perpendicular distance between the line and the measured points (see Figure 2). This was referred to as *eigenvector fit* in (Duda and Hart 1973). The idea of allowing errors in all variables when fitting data has been around for some time (Pearson 1901 and Madansky 1959). The concept was extended to multivariate problems about 20 years ago (Sprent 1969). The connection to the singular value decomposition of the measurement matrix was pointed out by Golub and Van Loan (1980) and Van Huffel and Vandewalle (1991). Total least squares was used for motion estimation in (Shizawa and Mase 1990 and Wang et al. 1992).

In the total least squares framework, (6) is

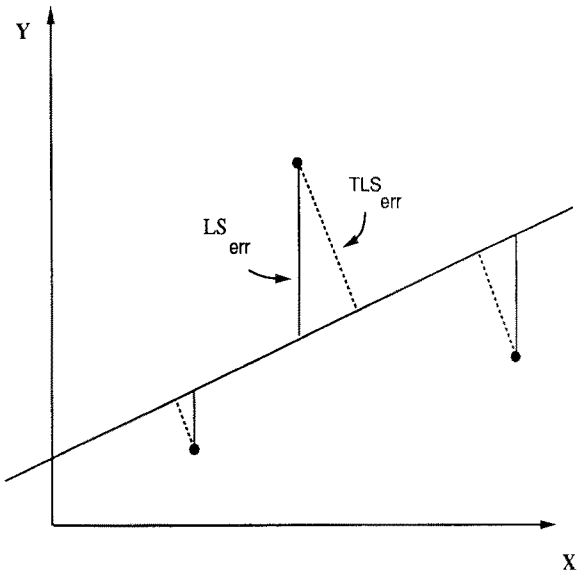

Fig. 2. Difference between least squares and total least squares for fitting a line to a collection of points. Least squares assumes the errors are in the y variables and thus minimizes the vertical distance between the line and the points. Total least squares allows for errors in both x and y and thus minimizes the perpendicular distance between the line and the points.

usually written as

$$[\mathbf{A} \mid \mathbf{b}_t] \begin{pmatrix} \mathbf{v} \\ 1 \end{pmatrix} = \mathbf{0} \tag{8}$$

The combined matrix $[\mathbf{A} \mid \mathbf{b}_t]$ is referred to as the measurement matrix. This form recognizes that each entry in the measurement matrix is subject to noise. In total least squares, an estimate is found by making the smallest, in terms of its Frobenius norm, perturbation to the measurement matrix such that (8) has a solution (Van Huffel and Vandewalle 1991). This is in contrast to least squares where only the measurement vector $\mathbf{b}_t$ is perturbed to find a solution. The estimate using total least squares is

$$\mathbf{v} = -(\mathbf{A}^T\mathbf{A} - \sigma_3^2\mathbf{I})^{-1}\mathbf{A}^T\mathbf{b}_t \tag{9}$$

where σ_3 is the smallest singular value of the measurement matrix. The Frobenius norm of the perturbation needed to make (8) consistent is simply σ_3. Equation (9) is very similar to the standard least squares solution. The latter is obtained by setting σ_3 to zero. The linear least-squares solution of constraint equations is used by a number of other authors (Duda and Hart 1973; Campani and Verri 1990, etc.). A total-least squares approach was used by Wang et al. (1992). Our approach differs from both approaches in that the constraint equations come from different filters at the same location whereas their constraints come from neighboring pixels (see Section 2.1).

Equation (9) is also very similar to the result obtained by Simoncelli et al. (1991) in which a Bayesian prior for small velocity magnitudes was used. Interestingly, the prior introduces a plus sign into the matrix to be inverted in (9) whereas we obtain a minus sign in the total least squares formulation. The prior assures that an inverse exists but will bias the estimate towards smaller velocity magnitudes. In Section 3.3 we will develop a *consistency ratio* which will guarantee that the magnitude of σ_3 is sufficiently less than the magnitude of the remainder of the measurement matrix. Thus the matrix inversion in (9) will not be unstable.

Total least squares assumes the error in each element of the measurement matrix is independent and identically distributed (the error matrix is white). If this is not the case, total least squares can actually perform worse than standard least squares. This is similar to the requirement in standard least squares that the errors in the measurements be normalized. We can use prior estimates of the measurement variances to whiten the measurement matrix.

3.2 Systematic Errors Due to Large Displacements

The finite differences used to approximate derivatives in the constancy equation (3) make a linear approximation to the underlying intensity function and are thus inaccurate for large displacements. The assumption breaks down quadratically in the displacements. For high frequency signals, this term can easily be larger than the stochastic error for relatively small displacements. If we operate in a single spatial dimension we can examine the relative magnitude of this term. If the signal is a simple sinewave, it is obvious that the linear

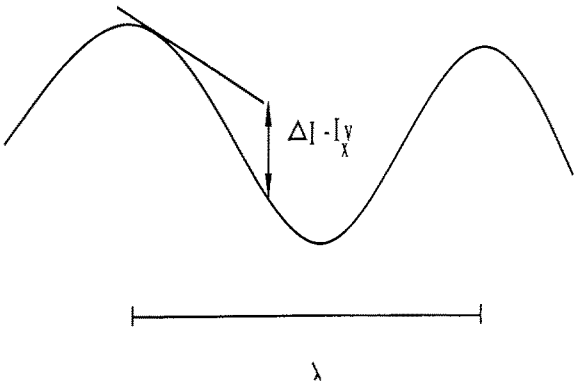

Fig. 3. The linear slope approximation for a sinusoid is only valid for a fraction of a wavelength. The range of velocity estimates a gradient-based approach can reliably detect is limited by the spatial frequencies of the underlying intensity function.

approximation is valid only for a fraction of the wavelength of the sinewave (see Figure 3).

If the wavelength, $\lambda = 2\pi/\omega$, of the signal is known, we can limit the acceptable displacement between time frames to some fraction of the wavelength.

$$|v| < \beta\lambda = 2\pi\beta/\omega \tag{10}$$

An natural upper bound is $\beta = 1/2$ since displacements greater than half a cycle would introduce aliasing. This limit on displacement as a function of signal frequency has a biological basis too (Cleary and Braddick 1990). Battiti et al. (1991) examined the systematic error implicit in gradient techniques which use finite differences to estimate partial derivatives. In the one dimensional case for a translating sinusoid of frequency ω, they find that the estimated velocity, $\hat{v}$, as a function of the true velocity, v, is

$$\hat{v} = \sin(\omega v)/\sin(\omega) \tag{11}$$

The difference between v and $\hat{v}$ comes from the linear slope approximation of finite differences. The relative error for this component is

$$\left|\frac{\hat{v} - v}{v}\right| = 1 - \frac{\sin(\omega v)}{v \sin(\omega)} \tag{12}$$

Writing this as a function of β at the maximum velocity,

$$\left|\frac{\hat{v} - v}{v}\right| = 1 - \frac{\omega \sin(2\pi\beta)}{2\pi\beta \sin(\omega)} \tag{13}$$

We find that for wavelengths greater than 4 pixels, a value of β about $1/2\pi$ results in a fractional error of less than 15%. Thus we set $\beta = 1/2\pi$. Since the filters used are bandpass, we feel this is a reasonable range of wavelengths. The resulting limit on the displacements allowed is $|v| < 1/\omega$.

Unfortunately we do not know the spectrum of the intensity function before filtering. If however we use a low-pass filter with a cutoff frequency ω_c then the maximum velocity estimate which can be considered valid is $|\mathbf{v}| < 1/\omega_c$ from the above analysis. In our implementation, the intensity function is convolved with a series of Gaussian and Gaussian derivative functions. These are either low or band-pass filters. The Gaussian has an associated scale factor, σ which is the standard deviation of the distribution. The n'th derivative of a Gaussian of standard deviation σ has its maximum frequency response at $\omega = \sqrt{n}/\sigma$ (Young 1985). If we use this frequency in limiting the maximum displacement we find that for the response formed by filtering with the n'th derivative of a Gaussian, the maximum displacement we can accept is

$$|\mathbf{v}| < \sigma/\sqrt{n} \tag{14}$$

3.3 Systematic Errors Due to Model Failure

There are situations where the underlying assumptions of the model are violated. Constancy of image brightness (1) is not strictly true whenever there is a significant specular component or when occlusion occurs. The model also assumes that the optical flow field is locally constant over the support of the filters, which is not true if the filter support straddles a depth discontinuity or when there is a significant rotational or divergence component in the flow field. In these situations, regression is not valid and these measurements should be labeled as outliers.

When calculating the total least squares solution, the singular values of the measurement matrix are available. The smallest singular value, σ_3, is equivalent to the Frobenius norm of the perturbation needed to make the equations consistent. We define a *consistency ratio* $\frac{\sigma_3}{\sigma_2}$. This is the ratio of the norm of the perturbation to

the smallest eigenvector of the resulting measurement matrix. If all the constraint lines are consistent (intersect at a common point), the perturbation and σ_3 will be zero. This would be true for a noiseless signal undergoing constant translation. When the assumptions are violated, a large relative perturbation will be needed to make the equations consistent. We discard scale groups which require a perturbation so large that the consistency ratio becomes larger than a given threshold, C_t, and assume that the model fails for this scale group. This discards the outliers before combining scales in a second total least squares. The ratio is scale independent and therefore can be used to compare estimates between scales.

Whereas we reject measurements which create outliers in the regression, other authors have used them to gain information about the underlying flow. Shizawa and Mase (1990) used the magnitudes of the perturbation in order to distinguish multiple flows. In this case the constraint lines would be inconsistent due to the presence of multiple transparent motions. Black and Anandan (1993) use the residuals of a non-linear cost function to identify and remove outliers and thus produce a robust estimator.

4 Model

Based on the above analysis, we propose the following model for multi-scale motion analysis. The image is first convolved with a collection of filters. These filters are separated according to scale. Thus we may have m different filters each of the same scale but differing in orientation and frequency response, and n such groups of these filters. The common scales of these groups form a geometric sequence. If the smallest scale is of size σ_0, then the i'th scale is of size $\sigma_o^{(i-1)}$.

Partial derivatives are computed via finite differences and weighted according to known prior noise variances.

Total least squares is used on the filtered responses in a two step method. First the n scale groups each individually form an estimate for the velocity via the total least squares formula in (8). This velocity estimate is deemed valid if the magnitude is less than the maximum allowed for that scale via equation (14). The estimate is also rejected if the ratio of the two smallest singular values of the measurement matrix is above the consistency threshold, C_t.

The remaining valid estimates are combined into a second total least squares formulation. The weights in this step have been divided by the consistency estimate of that scale's equations. This was the ratio of the two smallest singular values of the measurement matrix. A consistency ratio for the combined scales is computed along with the combined estimate. If this combined consistency ratio is larger than the threshold C_t the combined estimate is rejected.

This two step method contains two elements which make it robust. First, the ratio threshold prevents scale groups with poor estimates from participating in the second stage. Those scales which do participate are weighted by their individual residuals. In many iterative robust techniques, the process of finding an estimate, weighing by updated covariances and repeating is common (Huber 1981). By weighing the second stage by the singular values ratio, we insure that the scale which most accurately estimates the motion has the strongest influence in the multi-scale fusion. This is in contrast to coarse-to-fine methods which assume larger scales have a correct but coarse estimate. Secondly, if different scales see different motion because of either aliasing or transparency, the combined measurements will result in a large consistency ratio since the constraints will not be from the same motion. In this case we currently reject all scales and make no estimate. Our linear estimator assumes the constraints are from the same motion and thus can not be used to resolve the different motions. Individual residuals could be used to separate these different motions (Black and Anandan 1993).

Figure 4 outlines the method.

5 Implementation

The input to the model is simply two response frames separated in time. These two inputs are created from a sequence of images via convolution with separable Gaussian kernels. The

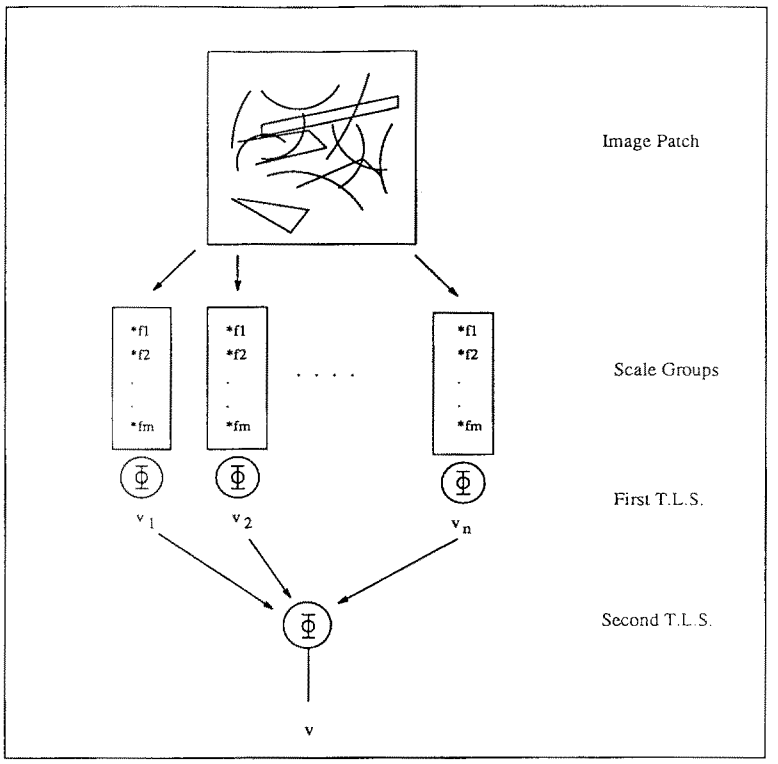

Fig. 4. Multi-scale gradient technique for motion. A patch of the image is convolved with groups of linear spatio-temporal filters. Each group is tuned to the same spatial scale. Each group makes its own estimate for the velocity using total least squares. The estimates are combined in a second re-weighted total least-squares formulation. The magnitude of the velocity estimate a group may present is limited by the expected systematic error.

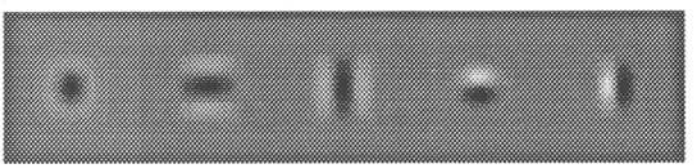

Fig. 5. Spatial impulse response of the filter set used for a single spatial scale. The filters are either the Laplacian of a Gaussian or products of Gaussians and a Gaussian derivative. These filters all have a zero DC response. The same set of filters is used in some approaches to early vision.

spatial kernels used were derivatives of the normalized Gaussian function, $G(x, y; \sigma_x, \sigma_y)$:

$$\partial_x G(x, y; \sigma_x, \sigma_y), \partial_y G(x, y; \sigma_x, \sigma_y), \\ \partial_{xx} G(x, y; \sigma_x, \sigma_y), \partial_{yy} G(x, y; \sigma_x, \sigma_y), \\ \nabla^2 G(x, y; \sigma_x, \sigma_y) \qquad (15)$$

For the directional derivatives, the Gaussian was elongated in the direction perpendicular to the derivative (i.e. $\sigma_y = 1.4\sigma_x$ for $\partial_x\ G(x, y; \sigma_x, \sigma_y)$). The Laplacian was rotationally symmetric. These filters have been used to model receptive fields of neurons in the visual cortex (Young 1985). The scale of each Gaussian function was set so that each filter shared a common peak frequency response. Thus they shared a common scale, σ_0. Figure 5 is the spatial responses for the scale group of $\sigma_0 = 16$ pixels. Note that each filter has a zero DC response and thus is not influenced by global lighting changes.

The sizes of the scale groups followed the progression $\sigma_0 = 1, \sigma, \sigma^2, \ldots, \sigma^{n-1}$ where σ was 1.8 for the experiments. This is a natural scale space representation as used in pyramid

implementations. A pyramid scheme can be used to decrease the computational load for the many convolutions required without a significant loss in performance (see Section 6.4).

Next, the filtered responses are convolved in the time dimension with the causal half of a standard Gaussian. This is non-zero only for past time frames. The standard deviation of this Gaussian was set to 3 video frames in order to emulate human response curves which show temporal recruitment up to about 100 milliseconds. Thus only the past 10 frames contribute significantly to any filter response. These numbers could change depending on the frame rate or known motions.

The partial derivatives of the responses are computed through a forward finite difference cube. This is simply the average of 4 adjacent pixel forward finite differences. Since the signal was already convolved with Gaussian functions, we felt a more sophisticated scheme for obtaining first partials was not needed. The forward differences actually provide an estimate of the partial derivatives on a lattice which is offset one half pixel in each spatial dimension and one half of a frame in the temporal dimension.

If a given scale group contains adequate texture such that the condition number of the measurement matrix was finite, a velocity estimate for that scale group is computed.

It is known that it is important to 'whiten' the measurement matrix such that each element is identically distributed and independent (Van Huffel and Vandewalle 1991). We assume that the output of each filter is independent. The oriented filters within a scale group actually are orthogonal. Even though it is not simply a linear combination of the elongated second derivatives, the symmetric Laplacian is not orthogonal to them. In addition, when the different scale groups are combined, the filters across scales are not orthogonal. We assume in both cases however that the magnitudes of the interaction terms are much smaller than the power of each filter and can be ignored.

The partial derivatives within a single equation do not have the same noise distribution due to the elongated Gaussian filters used (i.e. $\langle I_{ix}^2 \rangle \neq \langle I_{iy}^2 \rangle$). The partial derivative along the orientation of the filter has a higher noise response than the derivative perpendicular to the orientation. Since the oriented filters come in rotated pairs, a simple sum and difference of each pair results in two responses with partials of equivalent noise variance. The total least squares solution (8) is simply

$$\mathbf{v} = \left(\begin{bmatrix} \sum I_{xi}^2 \phi_i & \sum I_{xi} I_{yi} \phi_i \\ \sum I_{xi} I_{yi} \phi_i & \sum I_{yi}^2 \phi_i \end{bmatrix} - \sigma_3^2 \mathbf{I} \right)^{-1} \times \begin{pmatrix} -\sum I_{xi} I_{ti} \phi_i \\ -\sum I_{yi} I_{ti} \phi_i \end{pmatrix} \tag{16}$$

The summations are over the filtered responses within that scale group. The weights ϕ_i are the inverse of the expected variances of each measurement.

Note that we need only invert a 2×2 matrix whose entries are weighted sums of filter outputs. Thus only simple operations are required and can be performed in parallel. The filter responses can be accumulated as they are produced and need not be stored in memory. Since the measurement matrix has rank 3, simple explicit formulas exist for the three singular values, σ_i. They come from solving a cubic equation whose coefficients are combinations of the summations which appear in (16). Since singular values are always real and non-negative, a simpler form of the general solution of the cubic equation can be used. Testing the condition number of the matrix is then just the ratio of the two largest singular values (since these are the two singular values of the measurement matrix after the perturbation has been removed). If the condition number is above 100 the estimate is discarded. This rarely occurred in our simulations. Each scale group now evaluates its velocity estimate. The magnitude of this velocity estimate is compared with the maximum magnitude allowed for this scale group and rejected if larger. The ratio of the smallest singular values is compared with the consistency threshold.

The scale estimates that are not discarded are combined in a final, multi-scale total least squares framework. The weighting terms for this combination are modified by scaling the original weights by the inverse of the relative error term.

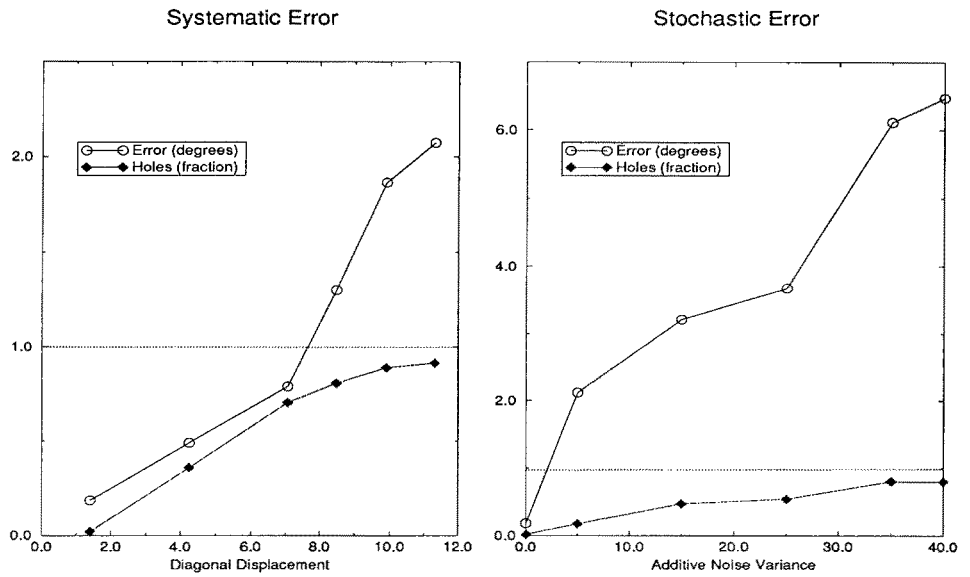

Fig. 6. Systematic error due to increasing displacements with no noise (left), and stochastic error due to additive noise with fixed dispacement (right). In both cases the signal was a Gaussian white noise pattern with variance 1000 units translated diagonally. The consistency threshold was set at 1×10^{-2}. This value keeps the errors to only afew degrees for the full range of allowed displacements.

Thus the weighting factor ϕ_i is replaced by:

$$\phi_i = \begin{cases} \frac{\phi_i}{\sigma_3/\sigma_2+\epsilon} & \sigma_3/\sigma_2 < C_t \text{ and } |\mathbf{v}| < \mathbf{v}_{\max} \\ 0 & \text{else} \end{cases}$$

where ϵ is a small number to prevent division by zero. This weighting gives more credit to scale groups with estimates which best match the constraint equations.

The final estimate computed by combining scales is rejected if the ratio of singular values indicates the equations are deemed inconsistent according to the consistency threshold. One place where this can happen is if different spatial scales overlap regions of different motions due to a motion boundary. Our future work will look at ways of identifying and resolving these situations. For now, they are simply labeled as places without estimates (holes).

6 Experimental Results

We tested the algorithm on a series of synthetic and real image sequences. For the experiments where the true optical flow is known, we use the angular error measure used by Barron et al. (1993) to evaluate the results. They measure the error between the true velocity $\mathbf{v} = (u, v)$ and the estimate $\widehat{\mathbf{v}} = (\widehat{u}, \widehat{v})$ as the angle between the unit vectors in 3 space, $\mathbf{v}_3 = (|\mathbf{v}|^2+1)^{-1/2}(u, v, 1)$.

$$\psi_\epsilon = \arccos(\mathbf{v}_3 \cdot \widehat{\mathbf{v}}_3) \tag{17}$$

This is calculated at every pixel value where an estimate is formulated. Also reported are the percentage of pixels without estimates (holes).

6.1 *Synthetic Data*

Systematic Error. A random dot pattern is translated diagonally. Each pixel was spatially uncorrelated, thus all frequencies were present. Figure 6 shows the angular error and fraction of holes (no estimate) as a function of diagonal displacement. As the displacement increases the average error increases, but remains less than two degrees. The number of estimates decreases up to the maximum displacement allowed, $1.8^4 \simeq 10.5$ pixels. Thus, up to the maximum displacement, we can keep errors to less than a few degrees. A higher density of

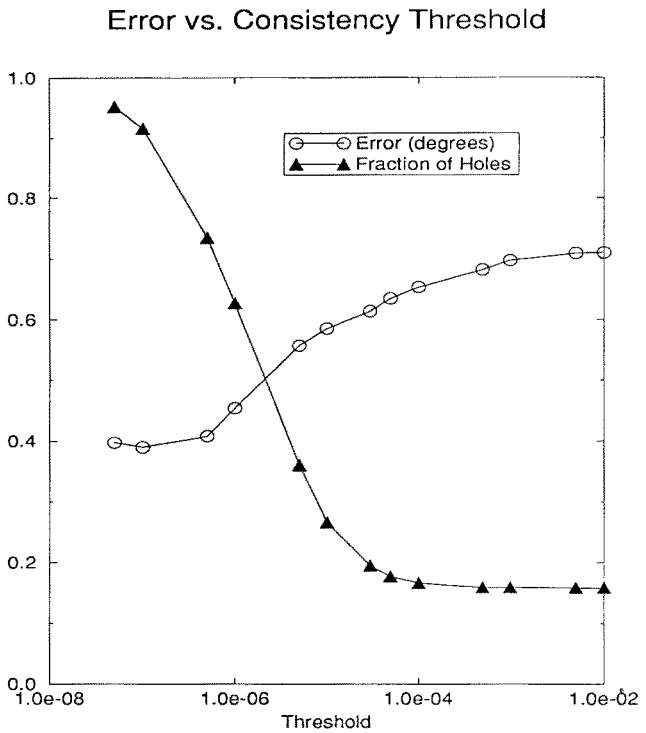

Fig. 7. Average error in degrees and fraction of holes as a function of the consistency threshold, C_t, the maximum of the ratio σ_3/σ_2 allowed. This ratio relates how much the measurements must be changed in order to fit the velocity model.

estimates can be obtained by lowering the consistency threshold at a cost of increased errors.

Stochastic Error. In order to examine the results of stochastic error we translated the random dot pattern diagonally 1 pixel and added uncorrelated noise to each frame. Figure 6 shows the results as the standard deviation of the added Gaussian noise is increased. Again, the errors remain less than a few degrees as the density decreases.

Consistency Threshold. A random dot pattern of unit variance was translated diagonally 1 pixel and white noise of variance 5 units was added to each frame. Figure 7 displays the average error and fraction of holes as a function of C_t, the equation consistency threshold. The number of estimates decreases and the accuracy of the remaining estimates increases until only a few estimates remain. This demonstrates that the singular values ratio is a good measure of estimate reliability. Such a statistic is often useful in algorithms which use optical flow as input, such as for determining ego-motion or shape from motion.

Comparison Sequences. A recent technical report by Barron et al. (1993) and the corresponding paper to appear in IJCV (Barron et al. 1994) examined the performance of a number of different optical flow techniques on a series of synthetic and real images. They found that the phase-based approach of Fleet & Jepson (1990) was the most accurate. We examined the performance of our algorithm on these same images.

Ten frames of the *Yosemite Sequence* were used as input to the algorithm. The true optical flow is known because this is a syntheticaly generated sequence. The sequence is of a platform flying over Yosemite valley. Clouds in the image deform as they move. The optical flow ranged in magnitude from zero at the focus of expansion to over 5 pixels per frame. The *Translating Tree* sequence consists of a tilted plane with a texture mapped onto it. The motion is perpendicular to

Table 1. Comparison of synthetic sequences results with those reported by Barron et al. (1993). The Weber & Malik algorithm uses only 10 'frames and 30 linear filters. The Fleet & Jepson algorithm used 46 3-d convolutions (realized in 74 1-d filters) and 21 frames (15 frames for Yosemite).

Sequence	Algorithm	Avg. Error	Std. Dev.	Density
Translating Tree	Horn & Schunck	38.72	27.67	100
	Heeger	4.53	2.41	57.8
	Anandan	4.54	2.98	100
	Lucas & Kanade ($\lambda_2 > 1.0$)	0.66	0.67	39.8
	Fleet & Jepson ($\tau = 2.5$)	0.32	0.38	74.5
	Weber & Malik	0.49	0.35	96.8
Diverging Tree	Horn & Schunck	12.02	11.72	100
	Heeger	4.49	3.10	74.2
	Anandan	7.64	4.96	100
	Lucas & Kanade ($\lambda_2 > 1.0$)	1.94	2.06	48.2
	Fleet & Jepson ($\tau = 2.5$)	0.99	0.78	61.0
	Weber & Malik	3.18	2.50	88.6
Yosemite	Horn & Schunck	32.43	30.28	100
	Heeger	10.51	12.11	15.2
	Anandan	15.84	13.46	100
	Lucas & Kanade ($\lambda_2 > 1.0$)	4.10	9.58	35.1
	Fleet & Jepson ($\tau = 2.5$)	4.25	11.34	34.1
	Weber & Malik	4.31	8.66	64.2

the optical axis, but since the plane is tilted the flow ranged in magnitude from 1.8 to 2.3 pixels per frame. The *Diverging Tree* consisted of the same tilted plane and texture, but the motion is along the optical axis. Velocities ranged from 1.4 pixels per frame on one side to 2.0 on the other. Table 1 lists the average and standard deviation of the angular error. The data for the other algorithms was copied from the revised technical report by Barron et al. (1993). The performance was comparable, performing better for the Yosemite sequence and worse on the translating planes. However, our algorithm uses only 10 frames and 30 linear filters whereas the Fleet & Jepson algorithm used 46 3-d convolutions (which can be realized by 74 separable and causal 1-d kernels) and 21 frames making it computationally more expensive. In addition, our algorithm consistently produced a higher density of vectors. The threshold experiments show that slight improvements can be made by decreasing the error threshold (fixed at 10^{-2} for synthetic sequences) at a cost of fewer estimates. Ultimately, the performance must be based on how well the flow field can be used for calculating quantities such as motion and shape parameters.

6.2 *Real Sequences*

The algorithm was tested on a group of real video images obtained from J.L. Barron who received them from the database at Sarnoff Research Center. We used 10 frames and 25 filters for each. Selected frames of the three sequences and the flow produced are shown in Figure 8. The first sequence is of the camera translating towards the soda can. In the second, the observer translates perpendicular to the line of sight. The tree in the foreground translates more due to perspective. The third sequence is of three independently moving cars. The car on the lower right is obscured by some trees.

6.3 *Large Displacements*

Differential techniques often have poor results for large displacements. We saw that this is a consequence of temporal aliasing of high frequency spatial components. We applied our algorithm to every second and every third frame of the Yosemite sequence. The results were compared with the true flow scaled by 2 and 3 respectively. The largest displacements in these sequences were 10 and 15 pixels per frame.

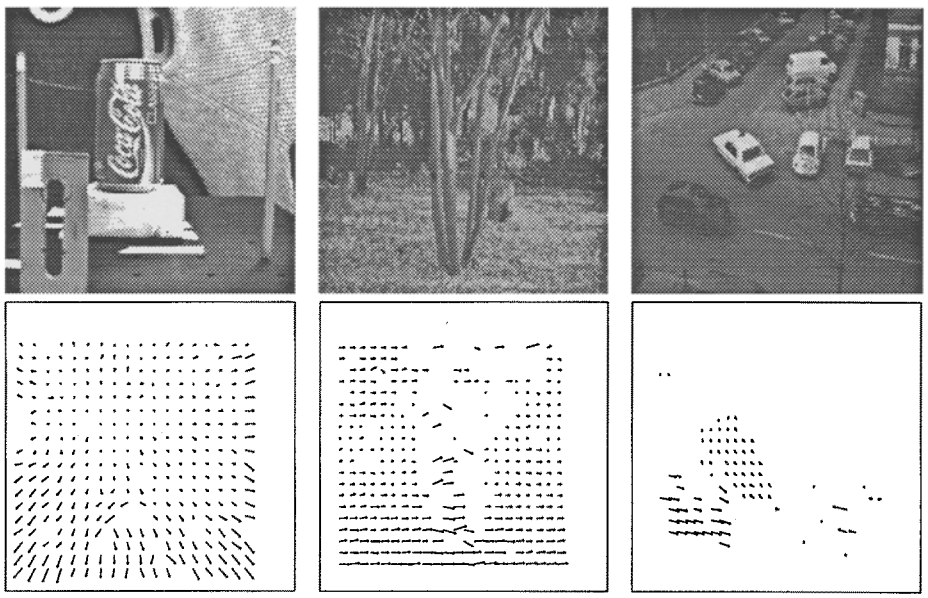

Fig. 8. Three frames from real image sequences and the flow recovered.

Table 2. Estimate accuracy and density as displacements are made larger. The Yosemite sequence was subsampled in time by 2 and 3 frames. Errors remain about the same as density decreases. The magnitudes of the estimates increased proportionally with the subsampling.

Temporal Subsampling:	1	2	3
Avg. Error	4.31	4.18	3.61
Std. Deviation	8.66	8.75	8.22
Density	64.2	39.4	23.4

Table 2 shows the average error, standard deviation and estimate density for the original sequence and the two temporally subsampled sequences. We can see a decrease in the estimate density while the error remains the same. The average magnitude of estimates increased roughly by the same scale factor as the true flow, thus estimates were not just clipped to small values.

6.4 Subsampled Filters

The largest scales consist of filters with Gaussian responses many pixels in width. The response of such filters does not change significantly within a range of a few pixels. The full version of the algorithm computes the response of every filter at each pixel. A more efficient implementation would use a pyramid scheme in which filters of larger scales are calculated at a subset of the full pixel lattice. We created a modified version of the algorithm in which the response of a filter with Gaussian response of size σ is subsampled every n pixels, where $n = \lfloor \sigma \rfloor$. This reduced by about half the number of convolutions for each scale from the previous scale. The results of the subsampled version of the algorithm for the synthetic images are tabulated in Table 3.

A Multiprocessor Implementation

The algorithm described is massively parallel. Each estimate is formed from a small spatio-temporal window of the motion sequence. The previous results were obtained from a SUN Sparc1 workstation. Processing time was dominated by the convolutions since velocity estimates required only a few simple operations on the convolution results per pixel. A series of 36, 3-d separable convolutions on a 128 pixel square image took about 4 minutes per frame. To examine the speedup possible with a par-

Table 3. Comparison of the full implementation where each filter output is computed at each pixel and a subsampling scheme where a filter output is calculated at every 2^n pixels.

	Full Convolutions		Subsampled Filters	
Sequence	Avg. Error	Std. Dev.	Avg. Error	Std. Dev.
Translating Tree	0.49	0.35	0.55	0.36
Diverging Tree	3.18	2.50	3.30	2.72
Yosemite	3.43	5.35	3.77	4.83

allel implementation, a parallelversion of the algorithm was created for the 128 processor CM5 from Thinking Machines Corporation.

Instead of the linear convolutions with filter kernels that was used on the serial machine, we used the interconnectivity of the processors and the Central Limit Theorem to approximate Gaussians by iterating nearest neighbor operations. The total number of steps required by this processes is $4\sigma_0^{2(s-1)}$ where σ_0 is the base scale and s is the number of different scale groups. The amount of data values which must be passed between processors on each iteration is $4n/\sqrt{N}$ where the image is of size n^2 using N processors. Details can be found in (Weber and Malik 1992). Convolution of a 128 pixel squared image took less than 10 seconds, including I/O. We believe that with specialized hardware real-time implementations are possible.

Acknowledgments

This research was partially supported by Texas Instruments, NSF Presidential Young Investigator Grant IRI-8957274 to J.M., Xerox and the PATH project MOU 83. NSF Infrastructure Grant number CDA-8722788 supported the use of the CM-5. We wish to thank D. Fleet and J. Barron for providing the sequences and true flow. We also thank A. Verri, D. Fleet and D. Heeger for helpful discussions.

Notes

1. Of course, this is just a consequence of assuming that the sensor noise has a Gaussian distribution, an assumption rarely verified in practice. One appeals to the Central limit theorem and hopes for the best.

References

Barron, J., Fleet, D., and Beauchemin, S. 1993. "Performance of optical flow techniques," Tech. Rep. RPL-TR-9107, Queen's University, Ontario. Revised version of U. Western Ontario TR 299.

Barron, J., Fleet, D., Beauchemin, S., and Burkitt, T. 1994. "Performance of optical flow techniques," *International Journal of Computer Vision*, vol. 12, pp. 43–77.

Fleet, D. and Jepson, A. 1990. "Computation of component image velocity from local phase information," *International Journal of Computer Vision*, vol. 5, pp. 77–104.

Bergen, J., and Adelson, E. 1988. "Early vision and texture perception," *Nature*, vol. 333, pp. 363–364.

Canny, J. 1986. "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 679–698.

Jones, D. and Malik, J. 1992. "Computational framework for determining stereo correspondence from a set of linear spatial filters," *Image and Vision Computing*, vol. 10, no. 10, pp. 699–708.

Jones, D. and Malik, J. 1992. "Determining three-dimensional shape from orientation and spatial frequency disparities," in *Proceedings of the Second European Conference on Computer Vision*, pp. 661–669.

Malik, J. and Perona, P. 1990. "Preattentive texture discrimination with early vision mechanisms," *Journal of the Optical Society of America A*, vol. 7, no. 5, pp. 923–932.

Turner, M. 1986. "Texture discrimination by Gabor functions," *Biological Cybernetics*, vol. 55, pp. 71–82.

Heeger, D.J. 1988. "Optical flow using spatiotemporal filters," *International Journal of Computer Vision*, vol. 1, pp. 279–302.

Fennema, C. and Thompson, W. 1979. "Velocity determination in scenes containing several moving objects," *Computer Graphics and Image Processing*, vol. 9, pp. 301–315.

Horn, B. and Schunck, B. 1981. "Determining optical flow," *Artificial Intelligence*, no. 17, pp. 185–203.

Tretiak, O. and Pastor, L. 1984. "Velocity estimation from image sequences with second order differential operators," in

Proceedings of the International Conference on Pattern Recognition, (Montreal).

Nagel, H.-H. 1987. "On the estimation of optical flow: relations between different approaches and some new results," *Artificial Intelligence*, vol. 33, pp. 299–324.

Uras, S., Girosi, F., Verri, A. and Torre, V. 1988. "A computational approach to motion perception," *Biological Cybernetics*, vol. 60, pp. 79–87.

Verri, A., Girosi, F. and Torre, V. 1990. "Differential techniques for optical flow," *Journal of the Optical Society of America A*, vol. 5, pp. 912–922.

Srinivasan, M. 1990. "Generalized gradient schemes for the measurement of two-dimensional image motion," *Biological Cybernetics*, vol. 63, pp. 421–431.

Lucas, B. and Kanade, T. 1981. "An iterative image restoration technique with an application to stereo vision," in *Proceedings of the DARPA IU Workshop*, pp. 121–130.

Campani, M. and Verri, A. 1990. "Computing optical flow from an overconstrained system of linear algebraic equations," in *Proceedings of the 3rd International Conference on Computer Vision*, (Osaka), pp. 22–26.

Wang, S., Markandey, V. and Reid, A. 1992. "Total least squares fitting spatiotemporal derivatives to smooth optical flow fields," in *Proceedings of the SPIE: Signal and Data Processing of Small Targets*, vol. 1698, pp. 42–55.

VanHuffel, S. and Vandewalle, J. 1991. *The Total Least Squares Problem: Computational Aspects and Analysis*. Frontiers in Applied Mathematics, Philadelphia: SIAM.

Duda, R. and Hart, P. 1973. *Pattern Classification and Scene Analysis*. New York, Chichester, Brisbane, Toronto, Singapore: John Wiley & Sons.

Pearson, K. 1901. "On lines and planes of closest fit to points in space," *Philos. Mag.*, vol. 2, pp. 559–572.

Madansky, A. 1959. "The fitting of straight lines when both variables are subject to error," *J. Amer. Statist. Assoc.*, vol. 54, pp. 173–205.

Sprent, P. 1969. *Models in Regression and Related Topics*. London: Methuen.

Golub, G. and VanLoan, C. 1980. "An analysis of the total least squares problem," *SIAM Journal Numer. Anal.*, vol. 17, pp. 883–893.

Shizawa, M. and Mase, K. 1990. "Simultaneous multiple optical flow estimation," in *Proceedings of the 10th International Conference on Pattern Recognition* (Atlantic City, New Jersey), pp. 274–278.

Simoncelli, E., Adelson, E. and Heeger, D. 1991. "Probability distributions of optical flow," in *Proceedings ofthe IEEE Computer Vision and Pattern Recognition Conference*, pp. 310–315.

Cleary, R. and Braddick, O. 1990. "Directional discrimination for band-pass filtered random dot kinematograms," *Vision Research*, vol. 30, pp. 303–316.

Battiti, R., Amaldi, E. and Koch, C. 1991. "Computing optical flow across multiple scales: An adaptive coarse-to-fine strategy," *International Journal of Computer Vision*, vol. 6, no. 2, pp. 133–145.

Young, R. 1985. "The gaussian derivative theory of spatial vision: Analysis of cortical cell receptive field line-weighting profiles," Technical Report GMR-4920, General Motors Research.

Black, M.J. and Anandan, P. 1993. "A framework for the robust estimation of optical flow," in *Proceedings of the Fourth ICCV*, (Berlin), pp. 231–236.

Huber, P.J. 1981. *Robust Statistics*. Wiley Series in Probability and Mathematical Statistics, John Wiley & Sons.

Weber, J. and Malik, J. 1992. "Robust computation of optical flow in a multi-scale differential framework," Tech. Rep. UCB/CSD 92/709, Computer Science Division (EECS), University of California, Berkeley.