Multiple Constraints to Compute Optical Flow

Massimo Tistarelli

Abstract—The computation of the optical flow field from an image sequence requires the definition of constraints on the temporal change of image features.

In this paper, we consider the implications of using multiple constraints in the computational schema. In the first step, it is shown that differential constraints correspond to an implicit feature tracking. Therefore, the best results (either in terms of measurement accuracy, and speed in the computation) are obtained by selecting and applying the constraints which are best "tuned" to the particular image feature under consideration.

Considering also multiple image points not only allows us to obtain a (locally) better estimate of the velocity field, but also to detect erroneous measurements due to discontinuities in the velocity field. Moreover, by hypothesizing a constant acceleration motion model, also the derivatives of the optical flow are computed.

Several experiments are presented from real image sequences.

Index Terms—Optical flow, velocity field, differential constraints, dynamic vision, motion analysis, image velocity, dynamic scene analysis, computer vision.

1 Introduction

In many computer vision techniques it is often desirable to over constrain the problem. This practice generally allows us to more precisely determine the solution. Moreover, it is also important to use redundant information to enforce robustness with respect to measurement noise in the input data. This is particularly true dealing with differential methods where high frequency noise is enhanced by derivative operators. For this reason there are many examples in the literature, where optical flow is computed by means of several constraint equations applied to many image points [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12]. Minimization and least squares are the mathematical tools most widely applied to solve this kind of problems.

Not always the constraints are obtained by applying the same equation to multiple points, as in [2], [4], [5], [7], [12], but also by defining multiple constraints for each image point, either based on a set of differential equations [1], [9], [10], [11], or obtained by applying the same set of equations to different functions which are related to the local image brightness [3], [6], [8]. Many researchers have also explicitly addressed the problem of occlusions, in the computation of smooth flow fields [1], [2], [5], [13], by using multiple constraints.

Regardless of the methodology used to over constrain the problem, there is a question to be answered: is always the problem posed in the correct way? What is the best way to over constrain the computational problem and make it *well posed*?

The aim of this paper is to analyze differential flow constraints commonly used to compute optical flow, and understand the relation with the underlying intensity pattern. The problem is faced in terms of the geometrical properties of the constraint equations in relation with the distribution of the image brightness.

2 Motion and Optical Flow

Uras et al. [14], among others [15], claim that the aperture problem is a "false problem." In fact, it can be easily overcome as soon as enough "structure" is present in the image brightness [15]. For example, assuming the flow field to be locally constant:

$$\frac{d}{dt}\nabla E = \vec{0} \tag{1}$$

where $E(x, y, t)$ is the image brightness.

Several researchers [1], [10], [14], [15] exploited the integration of multiple constraint equations, to over constrain the computational problem. By adding the brightness constancy equation to (1), three equations in two unknowns are obtained[1] [9], [10]:

$$\frac{d}{dt}E = 0 \qquad \frac{d}{dt}\nabla E = \vec{0} \tag{2}$$

In this case the optical flow is computed by solving, in closed form, an over-determined system of linear equations in the unknown terms $(u, v) = \vec{V}$.

3 Using Multiple Constraints

In this section, we consider three methods to compute the optical flow using multiple differential constraints:[2]

1) For each point on the image, where the Hessian matrix of the image brightness is not singular, the direct solution of (1) provides the velocity vector $\vec{V}$.
2) Simultaneously solve (2) applied to a single pixel:

$$\mathbf{A}\vec{V} = \vec{b} \quad \Rightarrow \quad \vec{V} = \left(\mathbf{A}^t\mathbf{A}\right)^{-1}\mathbf{A}^t\vec{b} \tag{3}$$

$$\mathbf{A} = \begin{bmatrix} E_x & E_y \\ E_{xx} & E_{xy} \\ E_{yx} & E_{yy} \end{bmatrix} \qquad \vec{b} = -\begin{bmatrix} E_t \\ E_{xt} \\ E_{yt} \end{bmatrix}$$

3) Solve an overconstrained system of linear equations obtained by applying (2) to a set of pixels within a given neighborhood.

These methods deserve advantages as well as disadvantages [16]. Considering Fig. 1, it is interesting to note that in the case of (2) it is impossible to determine, at least from the geometry of the system, which one is the wrong constraint line.

• *X.Y. Zzzzzzz is with xxxxxxx.For additional authors, place cursor just before The author is with the University of Genoa, Department of Communication, Computer and Systems Science, Integrated Laboratory for Advanced Robotics (LIRA - Lab), Via Opera Pia 13 - 16145 Genoa, Italy. E-mail: tista@dist.unige.it.*

Manuscript received Nov. 10, 1994; revised Oct. 3, 1996. Recommended for acceptance by A. Singh.
For information on obtaining reprints of this article, please send e-mail to: transpami@computer.org, and reference IEEECS Log Number P96104.

[1]. It is worth noting that this corresponds to a first order approximation of the optical flow field with a *locally* constant vector field. By taking into account the values of the spatial derivatives of velocity, it is possible to extend the approximation to the second order, but the analysis is here limited to the first order to allow the optical flow computation to be based on single pixel measurements.

[2]. Even though we are considering three basic constraints, the following analysis and results apply for any number of constraints, at least two, derived from the image brightness function [1], [3], [6], [8].

0162-8828/96$05.00 ©1996 IEEE

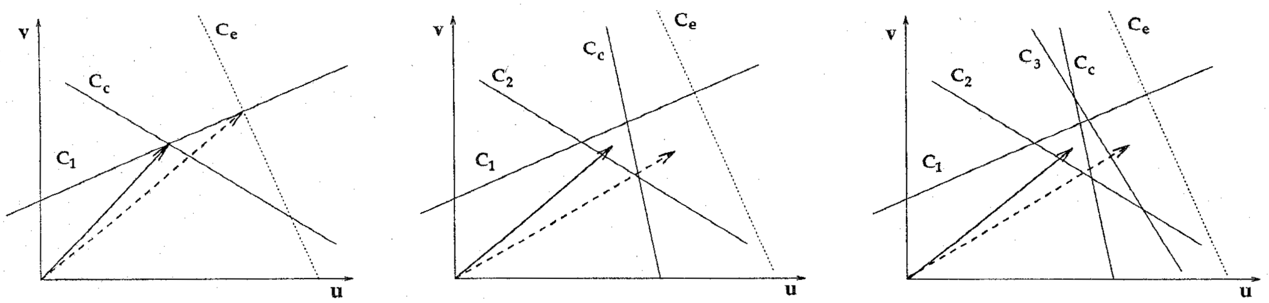


Fig. 1. Effects of the errors in one constraint equation. The wrong constraint is the dotted line C_e, while the correct constraint line is C_c. The wrong velocity vector is dashed. The geometry of the constraint equations is shown in case of: the constraint equation (1) (left); the constraint equation (2) (middle); multiple data points applying the brightness constancy equation (right).

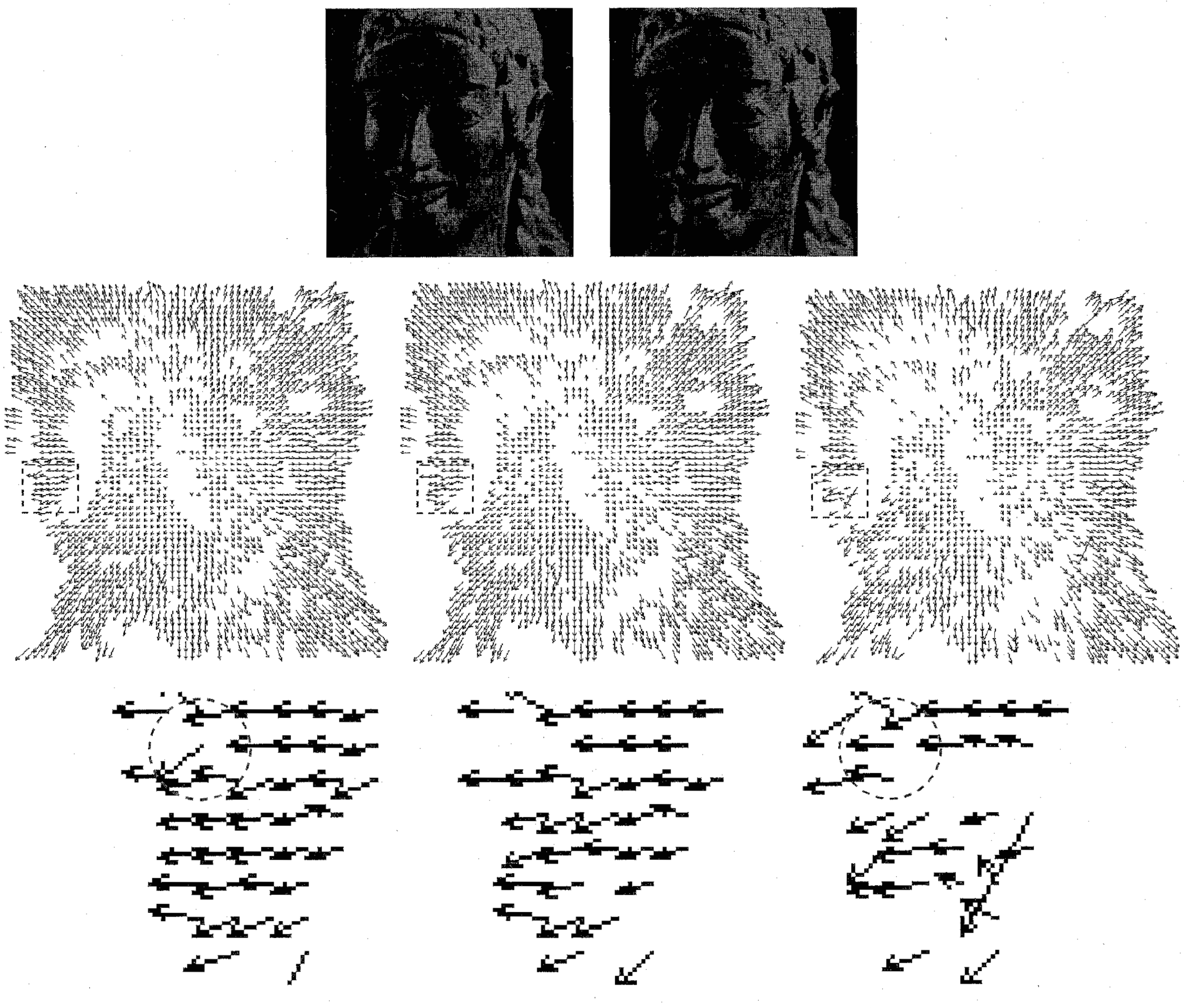


Fig. 2. Two images from a set of 37 of a flat picture moving toward the camera along the optical axis (top). Optical flows computed by using the best intersection (left); the least squares (middle); the mean of the two better intersections (right). On the bottom row, an enlargement of the flow fields relative to the marked window is shown.

3.1 Mapping Flow Constraints to the Image Brightness Structure

Instead of taking all three equations of (2) at the same time, it is possible to consider three equation pairs separately and study their stability. This is equivalent to consider each intersection point in Fig. 1 separately and determine the conditioning of the corresponding system of equations. This is done by analyzing the matrices:[3]

$$\mathbf{M}_1 = \begin{bmatrix} E_x & E_y \\ E_{xx} & E_{xy} \end{bmatrix} \quad \mathbf{M}_2 = \begin{bmatrix} E_{xx} & E_{xy} \\ E_{yx} & E_{yy} \end{bmatrix} \quad \mathbf{M}_3 = \begin{bmatrix} E_x & E_y \\ E_{yx} & E_{yy} \end{bmatrix} \tag{4}$$

A first option is to consider as correct solution the most reliable intersection in velocity space:[4]

$$\vec{V} = \mathbf{M}_{\mathrm{i}}^{-1}\,\vec{b}_i \quad \Leftrightarrow \quad \det\mathbf{M}_i > \left\{\det\mathbf{M}_{\mathrm{j}},\ \det\mathbf{M}_{\mathrm{k}}\right\}, \tag{5}$$

$$i,\ j,\ k \in \{1,\ 2,\ 3\}, \qquad i \neq j \neq k$$

$$\vec{b}_1 = \begin{bmatrix} E_t \\ E_{xt} \end{bmatrix} \quad \vec{b}_2 = \begin{bmatrix} E_{xt} \\ E_{yt} \end{bmatrix} \quad \vec{b}_3 = \begin{bmatrix} E_t \\ E_{yt} \end{bmatrix}$$

In Fig. 2, two images from a sequence of 37 are shown. The images are 256 × 256 pixels with eight bits of resolution in intensity. The sequence is relative to a flat picture moving along the optical axis of the camera. This image sequence has been used to compute the optical flow in the following ways:

- Applying the equation pair with the highest determinant.
- Computing the pseudo-inverse as in (3).
- Taking the average of the flow vectors computed from the equation pairs with higher determinants.

TABLE 1
STATISTICS OF THE THREE ALGORITHMS PRESENTED

	Algorithm (*a*): highest "det ***M***	Algorithm (*b*): least squares	Algorithm (*c*): mean of higher det $\boldsymbol{M}_i$"
N. of vectors	33156	31112	26668
timing	15 sec	17.5 sec	15.4 sec

For 51% of the image points, det $\mathbf{M}_1$ is the highest determinant, as 3% is det $\mathbf{M}_2$, and for 46% is det $\mathbf{M}_2$. (compare with (4)). Optical flows have been computed on a Sun SPARC IPC^{TM} workstation.

The three flow fields are quite similar, but, as shown in Table 1, their density is different. In order to facilitate a comparison of the techniques, a small portion of the optical flows is shown enlarged. As it can be noticed, the marked vector obtained from the mean of the two better intersections in the velocity space has a correct orientation (at least considering the velocity of the neighboring pixels) while the same vector, computed from the best intersection point in the velocity space, has an evidently wrong orientation. Most probably, this is due to the fact that, by taking the best intersection point, some relevant information is still missing. This information can be only included by considering also the remaining constraint line (or lines if more than three constraints are given). On the other hand, there are still errors depending on the local structure of the image brightness [17].

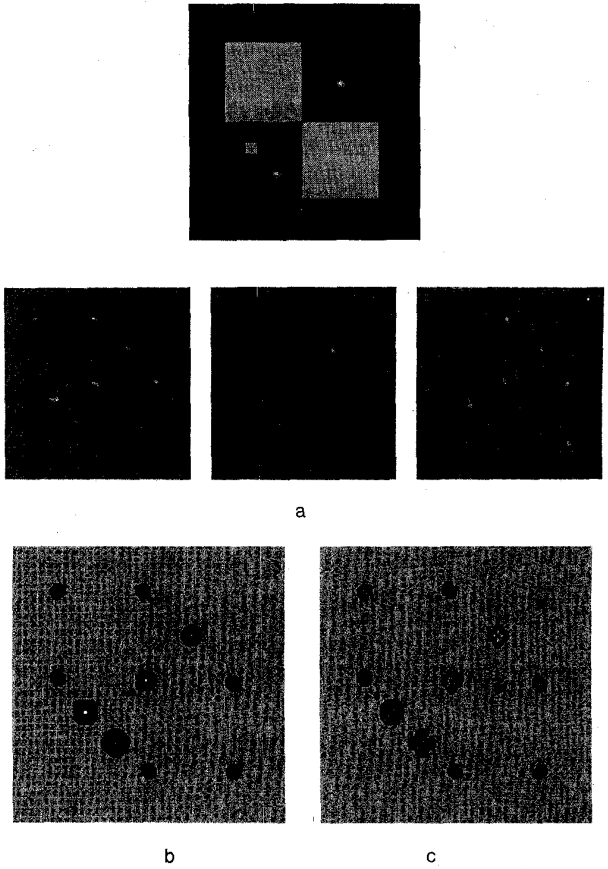


Fig. 3. Synthetic image used to characterize the behaviour of the constraint equations in response to different intensity patterns; a. Values of the determinants for the three equations: $\mathbf{M}_1$ (left), $\mathbf{M}_2$ (middle), $\mathbf{M}_3$ (right). The gray value codes the determinant values, light gray is 0, dark represent a negative value and bright a positive value; b. Dominance of the determinants. White values indicate that det $\mathbf{M}_2$ is the greatest determinant, gray and black are points where det $\mathbf{M}_1$ or det $\mathbf{M}_3$ have the greatest values. The zero value is coded as bright gray; c. The dominance of the determinants is shown as the absolute value of the difference between the two greatest determinant values: $|\det\mathbf{M}_i - \det\mathbf{M}_j|$. The zero value is coded as bright gray.

4 IMPLICIT FEATURE TRACKING

Any algorithm for the computation of the optical flow field, can be regarded as a solution to the tracking problem [16], [18]. In fact, even though differential methods do not establish any explicit correspondence between image features over time, still the differential operators involved (in any framework) track characteristic intensity patterns over time [11]. This is an interesting aspect which dramatically changes the perspective under which differential techniques should be analyzed and applied. Flow and/or motion constraints should not be applied regardless of the image sequence to be processed, but rather, purposively selected as a

[3]. In general one can consider a set of n functions of the image brightness $\vec{F}(E)$ with the constraints:

$$\tfrac{d}{dt}\vec{F}(E) = \vec{0} \qquad \tfrac{d}{dt}\nabla\vec{F}(E) = \vec{0}$$

where $\nabla\vec{F}(E)$ represents the gradient operator applied to each element of the vector of functions $\vec{F}(E)$. These constraints result in a set of $3 \times n$ equation pairs [3], [6], [8].

[4]. The computation of 2 × 2 matrix determinants is used throughout the paper to measure the reliability or confidence of the computed velocity vector, which corresponds to the intersection point of two straight lines in (u, v) space. In the Appendix this aspect is discussed in more detail.

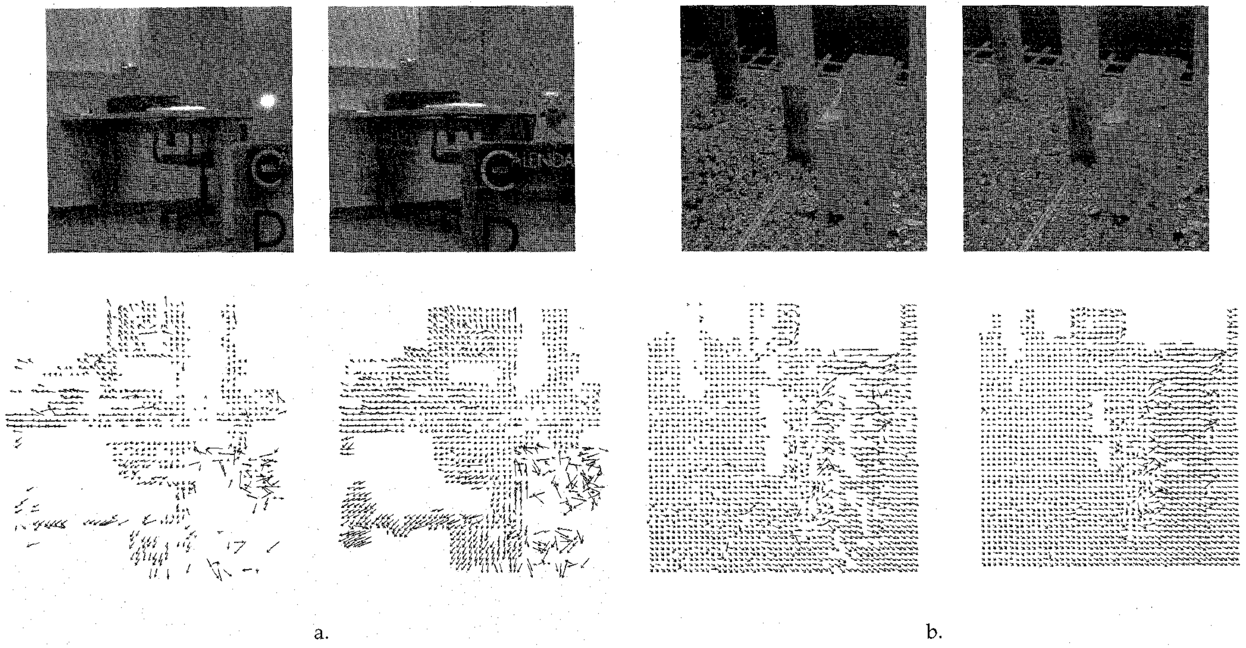


Fig. 4. Optical flows computed by using the weighted sum of the best intersections (on the left is the raw optical flow, and on the right is the smoothed optical flow). The parameters applied for the estimation of the optical flow fields are: a. $\sigma_s = 2$, $\sigma_t = 1$, $\sigma_f = 1.2$, $\Delta = 5\%$, $\tau = 1$; b. $\sigma_s = 2$, $\sigma_t = 1$, $\sigma_f = 1$, $\Delta = 5\%$, $\tau = 1$.

function of the local distribution of the image brightness.[5] In general, this is implicitly performed when tuning thresholds on the parameters of the algorithm or discarding flow vectors with a low confidence (in any way they have been computed in the first place) [11]. This methodology implies the measurement of wrong data from the beginning, while it would be desirable to avoid useless measurements. This is possible only by tuning the constraints to the underlying intensity distribution. As shown in Table 1, a side effect of this procedure is the reduction in the time required to process the image sequence.

4.1 Effects of the Image Intensity Distribution

By analyzing the values of the derivatives of the image brightness, Nagel [15] demonstrated that the brightness constancy equation (BCE) and the stationarity of the intensity gradient (SIG) equations are not defined when characteristic gray patterns occur on the image. In particular, none can be applied to compute the image velocity whenever the gray level variation is linear. In this case, at least one of the derivatives E_x and E_y are null, and $E_{xx} = 0$ and/or $E_{yy} = 0$. On the other hand, the value of det $\mathbf{M}_2$, as from (4), (which corresponds to the SIG equations), is minimum at gray level corners and it is maximum at extrema of the image intensity. This fact explains why the optical flow, computed by means of the constraint equations expressed in (1), is much less dense than applying all three equations of (3) because they also take into account the corners of the image intensity. Therefore, whenever the gray level pattern is not a corner or a maximum, all three equations are "weak," in the sense that the straight lines represented by equation pairs do not intersect at right angles (i.e., the corresponding matrix determinant is low).

The behavior of the considered differential constraint equations is explained in Fig. 3. The value of the dominant determinant for each image point is shown in Fig. 3b. Dark pixels correspond to a maximum for det $\mathbf{M}_1$, the middle gray identifies pixels where det $\mathbf{M}_3$ is maximum, and bright pixels correspond to a maximum for det $\mathbf{M}_2$. The areas where det $\mathbf{M}_2$ is prevalent are very small and limited to the peaks in the image intensity. It is now evident that not all the equations are "well tuned" to "sense" the temporal variation of every intensity pattern. But, rather, each equation is best suited to compute the motion of a particular intensity distribution.

Assuming the determinants of *two equation pairs* to be greater than a given threshold τ (stating the intersections to be admissible), it is possible to consider the probability of the intersections of both line pairs and try to maximize the a posteriori probability. In the (u, v) space this corresponds to move one intersection point toward the other, according to their confidence. If the two intersections have the same probability (or the three intersection points, obtained from the three equation pairs, are the vertexes of an isosceles triangle), then the most probable solution will be located at a point in the middle of the line connecting the two intersections. It is worth noting that this point does not correspond to the least squares solution.

In this way, the correct solution corresponds to the "center of mass" of the two points, where the "mass" of each point is the value of the respective matrix determinant, which is used as confidence measure:

$$\vec{V} = \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} N_h^u & + & N_j^u \\ N_h^v & + & N_j^v \end{bmatrix} \cdot \frac{1}{D_h + D_j} \tag{6}$$

where $N_h^u = b_i^1 M_i^{22} - b_i^2 M_i^{12}$ and $N_i^v = b_i^2 M_i^{11} - b_i^1 M_i^{21}$ are the numerators of the expression for $\vec{V}_i = (u_i, v_i)$, as from (5), and D_i = det $\mathbf{M}_i$. In the general case, selected m constraints from the total n equations, we obtain:

[5]. It is worth noting that in the work by Fleet et al. [19] this is performed by applying multiple filtering stages tuned to characteristic frequency bands in the image sequence.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sum_{i=l}^{m+l} N_i^u \\ \sum_{i=l}^{m+l} N_i^v \end{bmatrix} \cdot \frac{1}{\sum_{i=l}^{m+l} D_i} \qquad (7)$$

where the selected m constraints are the best tuned to the local intensity profile.

4.2 Experimental Results

The results of two experiments, relative to the computation of the optical flow by integrating multiple constraints for each pixel, are reported in Fig. 4.

In order to allow the numerical computation of the spatiotemporal derivatives of the image brightness, a spatio-temporal 3D Gaussian kernel has been applied to the images. The resulting raw optical flow has been further regularized by applying also a small Gaussian filter to both the u and v components of the vector field[6] [14], [20]. The parameters which have been set in the algorithm are the following:

- σ_s and σ_t are the values of the standard deviation of the Gaussian kernel in space and time;
- Δ is the maximum allowed difference in the values of the higher determinants;
- τ is the threshold on the value of the determinant computing the flow vectors;
- σ_f is the value of the standard deviation of the Gaussian kernel used to smooth the u and v component of the computed velocity field;

In order to improve visibility, the flow fields have been subsampled by taking one vector every five, along the x and y image coordinates. The spatial and temporal derivatives of the image brightness have been computed with fixed one-dimensional masks, locally approximating the Taylor expansion of the brightness function [20]. The sequences in Fig. 4 have been acquired with an ordinary CCD camera, under different conditions; all the images in the sequences are 256 × 256 pixels with eight bits of resolution in intensity.

The sequence in Fig. 4a (*moving sequence*) has been acquired inside a room, from a camera mounted on a mobile vehicle moving along a direction slightly inclined with respect to the camera optical axis. The object on the right, in the foreground, is moving along a direction almost orthogonal to the vehicle trajectory. On the bottom, both the raw and the smoothed optical flow are shown. As the object in the foreground is moving very fast, compared to the motion of the camera, the applied differential operators do not allow to compute a correct estimate of the image velocity of the moving object itself.

The second sequence (*blocks sequence*) has been acquired, at the University of Karlsruhe from a camera mounted on a robot arm [21]. Two images (256 × 256 pixels with eight bits per pixel) from the original sequence are shown in Fig. 4b. On the bottom of the same figure, both the raw and the smoothed optical flow are shown. It is worth noting that the moving block in the foreground can not be tracked very well because, during the motion, the surfaces in the background are occluded. This is an intrinsic limitation of the applied constraints and can be overcome by taking into account multiple pixels.

5 Integration of Data from Multiple Points

The application of multiple constraints can be extended to multiple data points. This is possible by assuming the flow field to be constant within a given neighborhood of the current pixel and by applying the constraints relative to each pixel simultaneously. The velocity vector is determined as the center of mass of the cloud of intersections (from the constraint lines), in the velocity space, relative to each pixel within the considered neighborhood.

If the velocity field is locally constant or smooth, a very small cloud is expected, because all the pixels will have the same velocity. On the other hand, if the cloud is large, or the intersection points are spread apart, then the pixels exhibit different velocities (see Fig. 5).

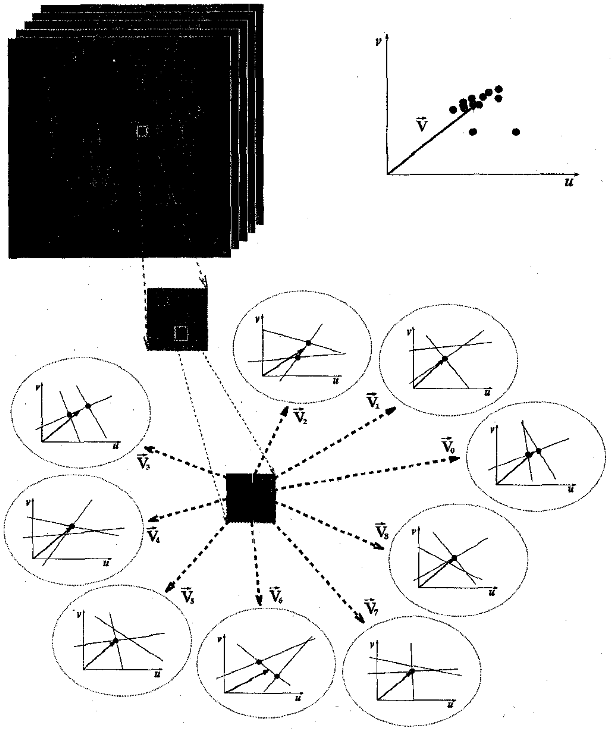


Fig. 5. A small window is shown out of the original image. For each pixel within a 3 × 3 window, three constraint lines are defined in the velocity space. One or two intersection points are selected for each pixel, according to the value of the determinant of the corresponding system of equations. As sketched by the diagram on the upper right corner, all the selected intersection points form a cluster in the velocity space, which determines the velocity vector for the central pixel within the 3 × 3 window.

In order to analyze the distribution of the intersection points, the modulus of the mean $E(\vec{V})$ and of the variance $\sigma(\vec{V})$ of the distribution can be defined:[7]

$$E(\vec{V}) = \left| \frac{1}{N} \sum_{i=0}^{N-1} \vec{d}_i \right| \qquad \vec{d}_i = \begin{bmatrix} |u_i - u| \\ |v_i - v| \end{bmatrix} \qquad (8)$$

[6] It is worth noting that the Gaussian filtering of the optical flow is not mandatory for the proposed algorithms. In the experiments it is mainly applied to fill-in missing flow vectors and to facilitate the understanding of the presented results.

[7] Other measurements could be applied, like the residual of the least square error, or other statistical techniques aimed at the definition of the clustering degree of a given data set [22]. The mean and variance of the point positions have been chosen because they directly reflect the measurements performed by the algorithm. On the other hand, other error measurements, like the residual, do not take into account the explicit non-linearity of the algorithm.

$$\sigma(\vec{V}) = \left| \frac{1}{N} \sum_{i=0}^{N-1} \left[\vec{d}_i - \vec{\delta} \right]^2 \right| \qquad \vec{\delta} = \frac{1}{N} \sum_{i=0}^{N-1} \vec{d}_i \tag{9}$$

where N is the number of selected intersection points, $\vec{V} = (u, v)$ is the velocity vector computed from the set of intersection points, and $\vec{V}_i = (u_i, v_i)$ is the velocity vector defined by each intersection point in the velocity space. The vector $\vec{\delta}$ represents the mean distance of the set of intersection points from the top of the resulting velocity vector $\vec{V}$. By setting a threshold on the value of $E(\vec{V})$ it is possible to determine if the velocity field is locally constant or, conversely, if the current image location corresponds to a flow discontinuity. It is worth noting that this method relies on a *local* assumption of the differential properties of the velocity field, which is used to compute a flow estimate at each pixel position.

5.1 Including Linear Variations of Velocity

By considering multiple pixels it is also possible to include a first order, nonconstant, term in the flow constraints (2). Linear variations of the optical flow [4] can be expressed as:

$$\vec{V}(\vec{x}) = \vec{V}(\vec{x}_0) + \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} \cdot \begin{bmatrix} \Delta_x \\ \Delta_y \end{bmatrix} \tag{10}$$

where u_x, u_y, v_x, v_y represent the spatial derivatives of the image velocity at $\vec{\mathbf{x}}_0 = (x_0, y_0)$, $\Delta_x = x_0 - x$ and $\Delta_y = y_0 - y$.

By substituting this expression into (2), for two points $\vec{\mathbf{x}}_i$ and $\vec{\mathbf{x}}_j$ one obtains:

$$\begin{bmatrix} E^i_x & E^i_x\Delta^i_x & E^i_x\Delta^i_y & E^i_y & E^i_y\Delta^i_x & E^i_y\Delta^i_y \\ E^i_{xx} & E^i_{xx}\Delta^i_x + E^i_x & E^i_{xx}\Delta^i_y & E^i_{yx} & E^i_{yx}\Delta^i_x + E^i_y & E^i_{yx}\Delta^i_y \\ E^i_{xy} & E^i_{xy}\Delta^i_x & E^i_{xy}\Delta^i_y + E^i_x & E^i_{yy} & E^i_{yy}\Delta^i_x & E^i_{yy}\Delta^i_y + E^i_y \\ E^j_x & E^j_x\Delta^j_x & E^j_x\Delta^j_y & E^j_y & E^j_y\Delta^i_x & E^j_y\Delta^j_y \\ E^j_{xx} & E^j_{xx}\Delta^j_x + E^j_x & E^j_{xx}\Delta^j_y & E^j_{yx} & E^j_{yx}\Delta^j_x + E^j_y & E^j_{yx}\Delta^j_y \\ E^j_{xy} & E^j_{xy}\Delta^j_x & E^j_{xy}\Delta^j_y + E^j_x & E^j_{yy} & E^j_{yy}\Delta^j_x & E^j_{yy}\Delta^j_y + E^j_y \end{bmatrix} \cdot \begin{bmatrix} u^0 \\ u_x \\ u_y \\ v^0 \\ v_x \\ v_y \end{bmatrix} = - \begin{bmatrix} E^i_t \\ E^i_{tx} \\ E^i_{ty} \\ E^j_t \\ E^j_{tx} \\ E^j_{ty} \end{bmatrix}$$

where $(u^0, v^0) = \vec{V}(\vec{\mathbf{x}}_0)$, $(\Delta^l_x, \Delta^l_y) = \vec{\mathbf{x}}_0 - \vec{\mathbf{x}}_l$ and the derivatives of the image brightness E^l_{hk} are computed at the point $\vec{\mathbf{x}}_l$. The optical flow and its first order derivatives can be computed if the parameter matrix is not singular. A necessary condition is that the intensity gradients at the image points $\vec{\mathbf{x}}_i$ and $\vec{\mathbf{x}}_j$ are not null and not aligned.

In order to reduce the errors in the velocity estimate, and to improve accuracy, more than just two points are used. In general, matrix inversion is a source of round-off errors. For this reason, singular value decomposition (SVD) has been applied to find a robust numerical solution for each possible intersection. In order to reduce the processing time, not all the available equations for each point[8] are used. For each pixel, det $\mathbf{M}_1$, det $\mathbf{M}_2$, and det $\mathbf{M}_3$ are also computed; only the equation pairs corresponding to the higher determinants are included in the estimation matrix. In this way, not only computations are avoided in areas where there is not enough structure in the gray levels to allow measurements (compare with Fig. 3b), but also sources of possible numerical errors are avoided.

5.2 Experimental Results

In Fig. 6 to Fig. 8, several experiments, performed by applying multiple constraints from multiple pixels, are shown. The parameters which have been set in the algorithm are the same of the previous experiments, while $E(\vec{V})$ is the mean of the distribution of the computed intersection points in the (u, v) space, measured in pixels. In order to improve visibility the flow fields have been subsampled by taking one vector every five, along the x and y image coordinates.

In Fig. 6 an experiment performed on the "moving sequence" is shown. On top one original image (left) and the computed raw optical flow (right) are shown. The parameters used for the velocity computation are: $\sigma_s = 2$, $\sigma_t = 1$, $\Delta = 5\%$, $\tau = 1$, $E(\vec{V}) = 0.2$. In the middle, the optical flow smoothed with a Gaussian kernel with standard deviation σ_f equal to one is shown. On the right, an enlargement of one computed velocity vector in the velocity space, corresponding to the position marked with a cross on the original image, is shown. All the intersection points used to compute the vector are also shown as small dots. On the bottom, are the computed mean (left) and variance (right) for all the image points. The light gray indicates a zero value, all other values range from one (dark) to 255 (white). It is worth noting that the area corresponding to the moving object in the foreground, has very high values both for the mean and for the variance of the distribution of the intersection points in velocity space. High values in the mean are also reported along the depth discontinuities.

In Fig. 7 the same images have been used to compute the optical flow with two different methods:

- by computing the pseudo intersections of all the constraints (least squares solution) and
- by integrating the constraints from multiple image points.

Both optical flows have been smoothed with a Gaussian operator with standard deviation σ_f equal to one.

As can be noticed, the first flow field reports errors in the area corresponding to the moving object. Conversely, the method based on the integration of constraints from multiple points allows to detect and discard the areas corresponding to wrong measurements.

The last experiment has been performed on an image sequence obtained from the original "blocks sequence," by cropping a 256×256 window around the moving marble block. On the top row of Fig. 8 the computed raw optical flow (left) and the optical flow smoothed with a Gaussian kernel with standard deviation σ_f equal to one (right) are shown. The parameters used for the computation are: $\sigma_s = 2$, $\sigma_t = 1$, $\Delta = 5\%$, $\tau = 1$, $E(\vec{V}) = 0.5$. In the middle row, one original image is shown. On the right, an enlargement of the computed velocity vector, in the velocity space, at two different positions (marked with a cross on the original image), are shown. All the intersection points used to compute the vectors are also shown as small dots. On the bottom row, are the computed mean (left) and variance (right) for all the image points. As it can be noticed, most of the higher values of the mean and variance are at the image points corresponding to depth discontinuities or to occluding edges of the two blocks in the image. High values in the mean are also reported, on the lower left corner, within an area with uniform intensity.

[8]. Considering a 3×3 neighborhood around $\vec{x}_0$, a total of $3 \times 8 = 24$ equations are obtained.

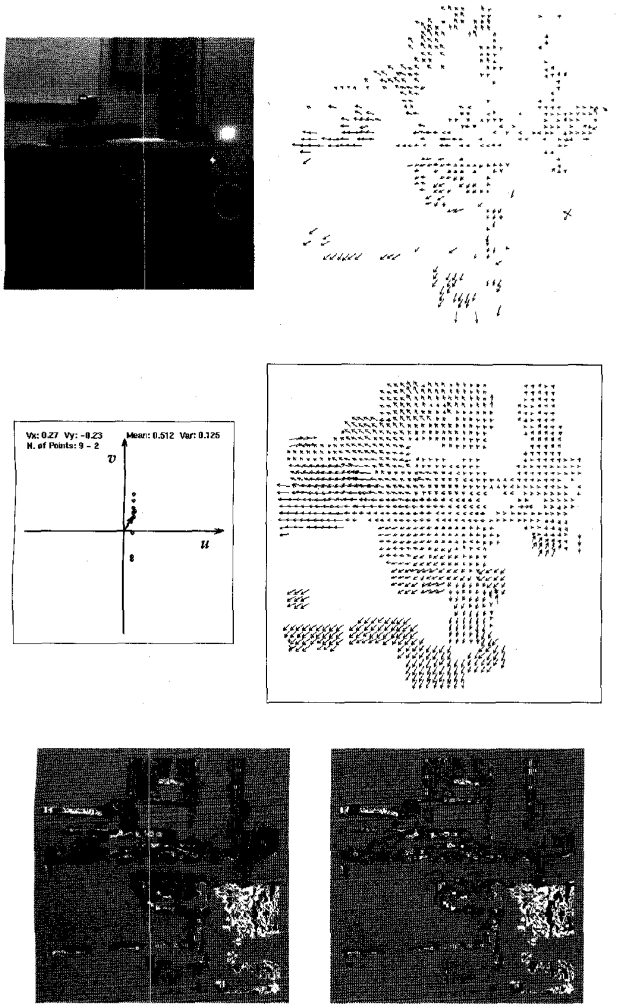
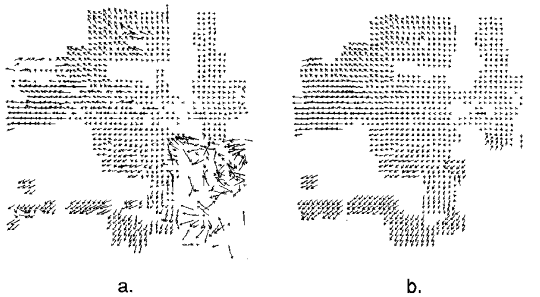


Fig. 6. Computation of the optical flow by applying constraints from multiple points on the "moving" image sequence.



a. b.

Fig. 7. Comparison between the optical flows computed with two different methods from the "moving" image sequence.

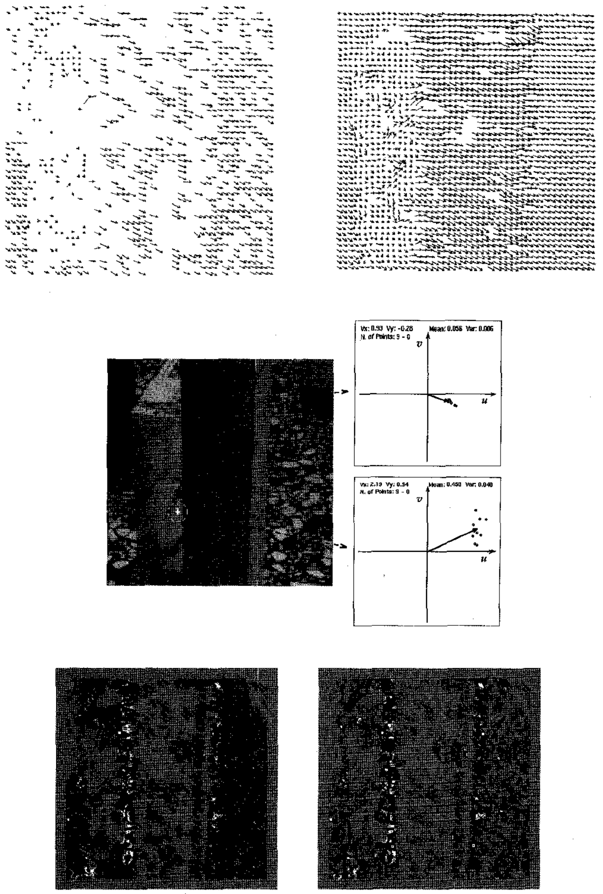


Fig. 8. Optical flow computed by applying constraints from multiple points on the "blocks" image sequence. The sequence has been obtained by extracting a 256 x 256 window from the original 512 x 512 images.

5 Conclusion

In this paper, the problem of combining multiple constraints to compute the optical flow field from an image sequence has been addressed.

One of the main aspects which has been outlined in this paper is that the response of a given constraint strictly depends on the local distribution of the image intensity. Therefore, the choice of the constraints to be applied should depend on the local structure of the image brightness and not only on the confidence associated to the measurement. In fact, there are examples where the local image structure does not allow to apply a given constraint at all, or the information obtained is completely wrong. These observations lead to the conclusion that in order to compute the optical flow field from an image stream, the constraints to be applied to the image sequence should not be chosen only on the basis of the motion to be detected, but also considering the local image structure.

It is demonstrated, both analytically and with experiments, that the same equations applied to different brightness structures can give exact or wrong estimates.

The complex nature of the real world, often makes assumptions for the velocity computation fail. This is due to many phenomena, like occlusions, shadows, depth discontinuities or even an excessive velocity of the objects. Two simple measurements, namely the mean and the standard deviation of the distribution of the intersections among the constraint lines, are devised to detect and possibly threshold the flow vectors corresponding to image points where some of the assumptions may be violated.

By applying different measurements other properties could be detected, and eventually determine which assumptions are violated. This analysis could be usefully applied to better understand the nature of the scene for successive vision tasks.

Appendix

In the literature, different criteria have been proposed to define the confidence of velocity measurements [11]. Considering first order differential techniques it has been pointed out that the magnitude of the spatial gradient is somehow related (or proportional) to the accuracy in velocity estimation. On the other hand, the determinant of the Hessian matrix of the image intensity is the most reliable measure to define the confidence of velocity estimates in second order differential techniques.

In this approach, the accuracy of a velocity estimate is related to the confidence of partial measurements obtained from equation pairs, namely single straight lines intersections in the velocity space. The determinant of each matrix $\mathbf{M}_i$ in (4) is directly proportional to the difference in the direction of the two straight lines:

$$\mathbf{M} = \begin{bmatrix} a_0 & b_0 \\ a_1 & b_1 \end{bmatrix}; \qquad \det \mathbf{M} = b_0 b_1 \left(\frac{\sin(\Delta\theta)}{\cos\theta_0 \, \cos\theta_1} \right) \tag{11}$$

where $\frac{a_0}{b_0} = \tan\theta_0$ and $\frac{a_1}{b_1} = \tan\theta_1$ are the angular coefficients of the two straight lines and $\Delta\theta = \theta_0 - \theta_1$. The value of det **M** is maximum when the straight lines intersect at right angle $\left(\Delta\theta = \pm\frac{\pi}{2}\right)$ and minimum when the straight lines are parallel. The value of the determinant also corresponds to the product of the eigenvalues. Even though each eigenvalue is proportional to the reliability of the corresponding constraint equation, the determinant has the advantage of being easier to compute, and also is directly related to the combination of the two equations. Therefore, the value of det **M** is a good measure for the confidence of the computed velocity vector.

Acknowledgments

This work has been partially funded by the Esprit project VAP, by grants of the Italian National Research Council and by the SMART European network of excellence.

References

[1] M. Otte and H.H. Nagel, "Estimation of Optical Flow Based on Higher Order Spatiotemporal Derivatives in Interlaced and Non-Interlaced Image Sequences," *Artificial Intelligence*, vol. 78, pp. 5–43, 1995.

[2] D. Ben-Tzvi, A. Del Bimbo, and P. Nesi, "Optical Flow from Constraint Lines Parametrization," *Pattern Recognition*, vol. 26, no. 10, pp. 1,549–1,561, 1993.

[3] V. Markandey and B.E. Flinchbaugh, "Multispectral Constraints for Optical Flow Computation," *Proc. Third IEEE Int'l Conf. Computer Vision*, Osaka, Japan, 1990, pp. 38–41.

[4] M. Campani and A. Verri, "Computing Optical Flow from an Overconstrained System of Linear Algebraic Equations," *Proc. Third IEEE Int'l Conf. Computer Vision*, Osaka, Japan, 1990, pp. 22–26.

[5] P. Nesi, "Variational Approach to Optical Flow Estimation Managing Discontinuities," *Image and Vision Computing*, vol. 8, no. 4, pp. 419–439, 1993.

[6] J. Lai, J. Gauch, and J. Crisman, "Using Color to Compute Optical Flow," *Proc. SPIE*, vol. 2,056, pp. 186–194, 1993.

[7] B.G. Schunck, "Image Flow Segmentation and Estimation by Constraint Line Clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 10, pp. 1,010–1,027, Oct. 1989.

[8] R.J. Woodham, "Multiple Light Source Optical Flow," *Proc. Third IEEE Int'l Conf. Computer Vision*, Osaka, Japan, 1990, pp. 42–46.

[9] O. Tretiak and L. Pastor, "Velocity Estimation from Image Sequences with Second Order Differential Operators," *Proc. Seventh IEEE Int'l Conf. Pattern Recognition*, 1984, pp. 16–19.

[10] M. Tistarelli and G. Sandini, "Dynamic Aspects in Active Vision," *CVGIP, Special Issue on Purposive and Qualitative Active Vision*, Y. Aloimonos, ed., vol. 56, no. 1, pp. 108–129, July 1992.

[11] J.L. Barron, D.J. Fleet, and S.S. Beauchemin, "Performance of Optical Flow Techniques," *Int'l J. Computer Vision*, vol. 12, pp. 43–77, 1994.

[12] J. Weber and J. Malik, "Robust Computation of Optical Flow in a Multi-Scale Differential Framework," *Int'l J. Computer Vision*, vol. 14, pp. 67–81, 1995.

[13] C. Schnorr, "Computation of Discontinuous Optical Flow by Domain Decomposition and Shape Optimization," *Int'l J. Computer Vision*, vol. 8, no. 2, pp. 153–165, 1992.

[14] S. Uras, F. Girosi, A. Verri, and V. Torre, "A Computational Approach to Motion Perception," *Biological Cybernetics*, vol. 60, pp. 79–87, 1988.

[15] H.H. Nagel, "On the Estimation of Optical Flow: Relations Between Different Approaches and Some New Results," *Artificial Intelligence*, vol. 33, pp. 299–324, 1987.

[16] M. Tistarelli, "Multiple Constraints for Optical Flow," *Proc. Third European Conf. Computer Vision*, J.O. Eklundh, ed., Stockholm, Sweden, May 2-6, 1994, pp. 61–70.

[17] T.S. Denney and J.L. Prince, "On Optimal Brightness Functions for Optical Flow," *Proc. 1992 IEEE Int'l Conf. Acoustics, Speech and Signal Processing*, Baltimore, Md., 1992, pp. 257–260.

[18] C.Q. Davis, Z.Z. Karu, and D.M. Freeman, "Equivalence of Subpixel Motion Estimators Based on Optical Flow and Block Matching," *Proc. IEEE Int'l Symp. Computer Vision*, Coral Gables, Fla., Nov. 21-23, 1995, pp. 7–12.

[19] D.J. Fleet and A.D. Jepson, "Computation of Component Image Velocity from Local Phase Information," *Int'l J. Computer Vision*, vol. 5, pp. 77–104, 1990.

[20] V. Torre and T. Poggio, "On Edge Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, pp. 147–163, Feb. 1986.

[21] H.H. Nagel, G. Socher, H. Kollnig, and M. Otte, "Motion Boundary Detection in Image Sequences by Local Stochastic Tests," *Proc. Third European Conf. Computer Vision*, J.O. Eklundh, ed., Stockholm, Sweden, May 2-6, 1994, pp. 305–316.

[22] R.C. Tryon and D.E. Bailey, *Cluster Analysis*. New York: McGraw-Hill, 1970.