Multiwindow least-squares approach to the estimation of optical flow with discontinuities

Franco Bartolini
Vito Cappellini
Universitá di Firenze
Dipartimento di Ingegneria Elettronica
via S. Marta 3
I-50139, Firenze, Italy

Carlo Colombo
Scuola Superiore di Studi Universitari e di Perfezionamento S. Anna
Advanced Robotics Technology and Systems Laboratory
via S. Zeno 2
I-56127, Pisa, Italy

Alessandro Mecocci
Universitá di Pavia
Dipartimento di Elettronica
via Abbiategrasso 209
I-27100, Pavia, Italy

Abstract. The use of optical flow fields in image sequence analysis allows us to perform motion-based segmentation as well as 3-D reconstruction. Many techniques for optical flow estimation are based on some *global* or *piecewise global smoothness* assumption. Other techniques compute the flow field based only on *local* information. A local algorithm explicitly addressing the problem of evaluating a reliable optical flow field at motion boundaries is presented. Velocity vectors are computed as solutions of a multiwindow least-squares problem; the field is then regularized by a vector median filter. The algorithm is noniterative and nonparametric. Results on both synthetic and real-world sequences are shown; a performance comparison with two well-known techniques demonstrates the effectiveness of the algorithm in terms of noise rejection, motion boundary preservation, and speed.

Subject terms: image processing; least-squares techniques; motion analysis; motion boundaries; optical flow; regularization; vector median filtering.

Optical Engineering 32(6), 1250-1256 (June 1993).

1 Introduction

Since the end of the 1970s, interest in image sequence analysis—specifically motion analysis—has rapidly increased due to the enhancement of technology, which has greatly broadened the potential fields of application of image processing. Several application areas can take advantage of the extraction of dynamic information embedded in image sequences. These areas include industry (dynamic process monitoring, autonomous mobile robots, automatic vehicles) biomedical engineering (heart motion analysis, blood-flow monitoring), communications (TV bandwidth reduction, teleconferencing), environmental monitoring (accident detection, traffic-flow surveillance), and remote sensing (atmospheric pollution monitoring, cloud motion estimation).

Motion analysis is typically performed in two steps. In the first step, a 2-D velocity field is extracted from the raw image data; then, in the second step, this field is used to infer information about the 3-D imaged scene[1] and/or the motion of the observer.[2]

Two main approaches exist for computation of the 2-D velocity field. A sparse 2-D field is obtained by detecting and tracking salient features of moving objects in the scene.[3] On the other hand, by exploiting the correlation between the spatial and temporal variations of image brightness, a dense motion field is estimated, which is usually referred to as *optical flow* (OF).[4]

The main advantage of a dense field over a sparse field is that the former can be helpful in performing image segmentation based only on motion cues. (There is some experimental evidence that biological vision systems also have this ability.[5]) The spatial discontinuities of OF play a major role in achieving this task. Unfortunately, the evaluation of OF at its points of discontinuity (*motion boundaries*) has proved to be a serious problem, in that the estimated field is usually either noisy or incorrectly smooth in the neighborhood of these points.

In this paper an algorithm that allows the estimation of a dense velocity field with special attention to motion boundary preservation is proposed. The algorithm works by performing the following steps at each image pixel:

1. Partition the pixel's neighborhood into a small number of asymmetric and overlapping subwindows.
2. Compute, at each subwindow, the velocity vector as the solution of a least-squares (LS) problem.
3. Select, as the OF solution, the most reliable among the velocity vectors computed in the previous step.

This paper is organized as follows: First, the OF computation problem is stated and some known methods for its solution are briefly discussed; second, the details of the algorithm are given; then, some results are discussed; and, finally, conclusions are offered.

2 Optical Flow Techniques

A linear equation that constrains the two unknown velocity components u and v at each image pixel (x,y) and any time t can be obtained by assuming the stationarity of image brightness[6]:

Paper SIP-08 received Oct. 16, 1992; revised manuscript received Jan. 5, 1993; accepted for publication Jan. 22, 1993.

$$E_x u + E_y v + E_t = 0 \ , \tag{1}$$

where E_x, E_y, and E_t stand for the spatial and temporal partial derivatives of image brightness $E(x,y,t)$.

Equation (1) alone is not sufficient to determine the OF vector univocally; actually, it constrains the vector (u,v) to belong to a line in the velocity space, usually referred to as a *constraint line.* [However, it is easy to show that Eq. (1) can be used to compute the OF component along the direction of the local brightness gradient.] Computing OF by Eq. (1) is thus an underconstrained problem whose solution needs the introduction of additional constraints. In the following, some approaches to OF computation are summarized.

Horn and Schunck[6] suggest exploiting the intrinsic smoothness of the OF of a single object to achieve coupling of another constraint with Eq. (1). Their algorithm involves the minimization of a functional including two penalty terms: a measure of the distance of the solution from the constraint line and a measure of the departure from smoothness of the local field. The main problem with this algorithm is that the condition of a smooth flow field is not met in particular zones of real images, namely, the boundary points between two occluding moving objects. More generally, the imposition of the smoothness constraint over the whole image space introduces a correlation among the field vectors at virtually all image points, a link that has no physical reason.

The algorithms involving the minimization of similar functionals[7,8] are called *global* and usually produce an incorrect flattening of the computed field.

Other techniques have recently been proposed that are also based on the solution of an optimization problem. Image brightness and OF are modeled as 2-D Markov random fields and the concept of *line processes*[9] is introduced to account for motion discontinuities.[10–12] The OF and motion boundaries are simultaneously and iteratively estimated, penalty terms only being imposed far from discontinuities, thus yielding a *piecewise global* optimization. Despite the interesting result that a simultaneous OF estimation and segmentation can be achieved, such techniques suffer from the disadvantage of being parametric and computationally demanding.

Conversely, *local* techniques are known to be simple and fast. They exploit the information from a small neighborhood around the examined pixel, either by collecting constraint lines of neighboring points and solving the resulting overconstrained set of linear equations[13,14] or by determining more than one constraint at each pixel.[15–17] Local techniques do not impose any *a priori* smoothness over large patches of the image, thus also offering the advantage of eliminating undesirable flattening effects. The main drawback of such techniques is that they are ill-conditioned in certain regions of the image plane—i.e., regions of approximately uniform brightness or where spatial image gradient information is nearly unidirectional—where a quite noisy flow field is evaluated, thus creating the need for a postprocessing regularization step.[16,18]

A straightforward local technique is based on the LS solution of the overconstrained linear system obtained at each image pixel by collecting constraint lines from its neighboring points.[13] Although quite insensitive to image noise, this method is grossly in error when the examined neighborhood is crossed by a motion boundary, i.e., when constraint lines belonging to different moving objects are collected. The algorithm presented in the following section attempts to remove the drawback exhibited by this LS method while preserving its good noise insensitivity.

3 The Multiwindow LS Algorithm

Let us assume we have a small region $w(i,j)$ of the image in which the OF is known to be uniform (for example, a patch completely contained in a translating or very slowly rotating object). Then, in principle, the constraint lines at N points $(p,q) \in w(i,j)$ should intersect at a point of the velocity space representing the speed of the whole patch. Actually, due to image noise and approximation errors in the estimation of derivatives, the constraint line parameters can be grossly incorrect. However, supposing $N>2$ and that these errors are randomly and independently distributed, a LS pseudo-intersection of the constraint lines can give a good estimate of the true solution. Specifically, given the overconstrained linear system

$$E_x(p,q)u + E_y(p,q)v + E_t(p,q) = 0 \qquad (p,q) \in w(i,j) \ , \tag{2}$$

the pseudo-intersection is obtained by minimizing the residual square error:

$$\mathfrak{E}_w(u,v) = \sum_{(p,q)\in w(i,j)} [E_x(p,q)u + E_y(p,q)v + E_t(p,q)]^2 \ , \tag{3}$$

that is, by solving the following 2×2 linear system, obtained by setting to zero the partial derivatives of $\mathfrak{E}_w$ with respect to the velocity components u and v:

$$\begin{bmatrix} \sum_{(p,q)\in w(i,j)} E_x^2(p,q) & \sum_{(p,q)\in w(i,j)} E_x(p,q)E_y(p,q) \\ \sum_{(p,q)\in w(i,j)} E_x(p,q)E_y(p,q) & \sum_{(p,q)\in w(i,j)} E_y^2(p,q) \end{bmatrix} \times \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -\sum_{(p,q)\in w(i,j)} E_x(p,q)E_t(p,q) \\ -\sum_{(p,q)\in w(i,j)} E_y(p,q)E_t(p,q) \end{bmatrix} . \tag{4}$$

Note that Eq. (4) does not have a unique solution if the coefficient matrix has a rank less than two. This can happen in areas where the image gradient is either zero (constraint lines are not defined) or has a uniform orientation (constraint lines are parallel and are thus without any finite point of intersection); in such cases, any local computation of OF becomes ill-conditioned. To avoid this pitfall, the dimensions of the region $w(i,j)$ must be big enough to grant sufficient variability of the gradient.

For OF computation at each image point, the LS technique can be used by collecting constraint lines from a square neighborhood of each pixel. A neighborhood size of 5×5 pixels is often large enough to avoid ill-conditioning and small enough to keep the condition of uniform motion inside it:

$$w(i,j) = [(p,q): \quad -2 \leq p - i \leq 2 \text{ and } -2 \leq q - j \leq 2] \ . \tag{5}$$

As previously mentioned, the LS algorithm is attractive because it is quite noise insensitive (as is every error-minimization method) and is a local technique. Unfortunately, it has the drawback of producing an unsatisfactory flow field at the points of occlusion between objects having different motion, and, as such, the hypothesis of quite uniform motion over the neighborhood is no longer valid.

A similar problem is encountered in image processing when attempting to reduce image noise by low-pass filtering; although a weighted mean operation over a uniform area eliminates random noise due to the acquisition process, it also has the undesirable effect of blurring image contours. It is thus necessary to distinguish between the variations of image brightness due to noise and those due to discontinuities. Nagao and Matsuyama[19] propose a solution to this problem. A square window around the pixel is partitioned into a small number of asymmetrical overlapping subwindows. The variance of the gray value of the pixels in each subwindow is then computed, and the subwindow producing the minimum variance is selected. Subwindows are designed in such a way that, even at a contour pixel, at least one of them will belong entirely to the same uniform region as the center of the window, thus exhibiting a minimum of the variance. The mean value of the gray level of the pixels in this subwindow is then assigned to the center of the window. This algorithm assumes that the SNR of the image is quite high, so that the brightness variations due to noise are smaller than those due to image texture.

In this paper we argue that a similar approach can be used to compute OF and achieve good motion boundary preservation. For example, it is possible to partition a 9×9 neighborhood of the generic pixel where OF has to be computed into a set of nine 5×5 overlapping asymmetrical subwindows, as in Fig. 1. The subwindows are designed so as to enhance the probability that at least one of them will belong entirely to the same moving object as the central pixel. The LS method mentioned earlier is used to compute a velocity solution (u_n,v_n) in $w_n(i,j)$, $n=0, 1, ..., 8$; among the solutions obtained, the one exhibiting the minimum residual square error is chosen and assigned to the center of the neighborhood as its OF vector, that is,

$$(u,v)=(u_k,v_k):\quad \mathfrak{E}_{wk}(u_k,v_k)\leq \mathfrak{E}_{w_n}(u_n,v_n)$$
$$n=0,\ 1,\ ...,\ 8\ . \qquad (6)$$

If the examined pixel is at the border of an object, it is likely that at least one of the considered subwindows is entirely inside the object (as happens for subwindow w_1 in Fig. 2). That subwindow will then almost surely exhibit the minimum residual square error if the errors affecting the constraint lines parameters are assumed to be small enough to produce a residual not bigger than the one obtained in a system with constraint lines related to different speeds. Moreover, it is argued that the algorithm will also improve the performance at pixels far from motion boundaries, in that the multiwindow system gives the possibility of obtaining a LS solution from a selected and reliable subset of neighboring constraint lines.

Since the number K of neighbor pixels actually involved in the LS solution is variable (in each subwindow only the pixels having a gradient magnitude different from zero are considered), to avoid altering the comparison, residuals $\mathfrak{E}_w$ must be normalized by K.

Despite the accurate choice of the subwindow size, some gross errors will result—as mentioned above—in certain image regions, due to ill-conditioning. The computed field will also appear quite irregular due to image noise. Therefore, we must perform an OF regularization step; this can be successfully accomplished by vector median filtering,[20] a technique whose nonlinear nature ensures better motion boundary preservation than the traditional averaging operators.[16]

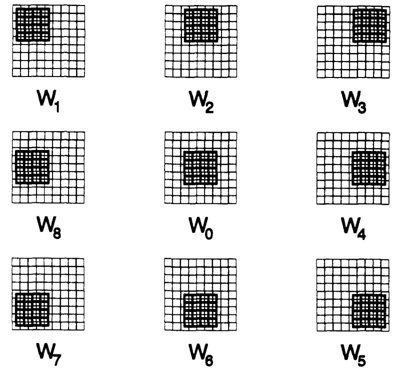


Fig. 1 Structure of the nine subwindows partitioning the processed window.

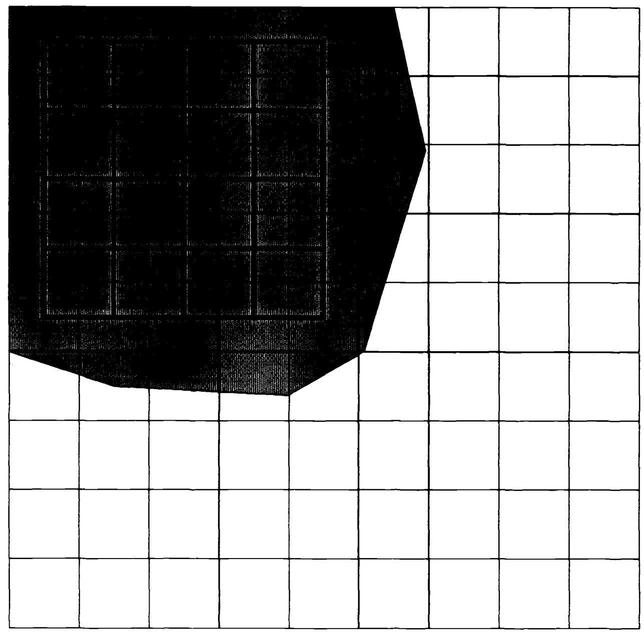


Fig. 2 Case of a window centered at a boundary pixel, and the corresponding subwindow that should give the minimum residual error.

4 Results

The algorithm was tested both on synthetic and real-world image sequences. A comparison of the results obtained running the classical LS algorithm,[13] the multiwindow LS technique proposed in this paper, and the well-known Horn and Schunck iterative method[6] on two 128×128 sequences is presented here. The two LS algorithms use a window size of 9×9; vector median regularization is performed by means of a 5×5 neighborhood and a fast approximation of the L_1 norm.[21] The smoothness parameter for the Horn and Schunck algorithm is set to 1 and the number of iterations to 64; in

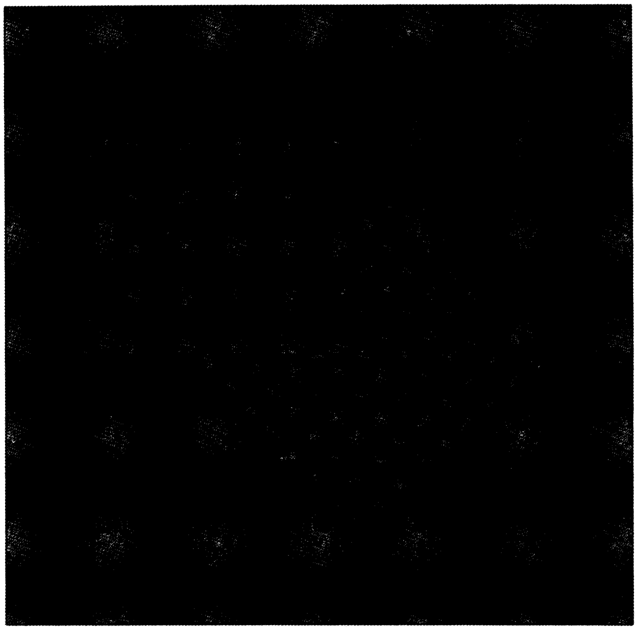


Fig. 3 Second frame of the SYNTH sequence; all textures are sinusoidal gratings.

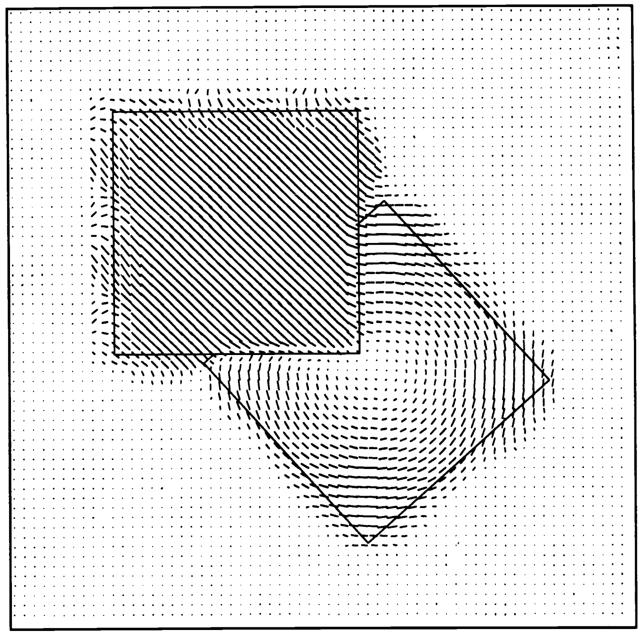


Fig. 5 Field resulting by applying the classical LS algorithm to the SYNTH sequence; with regularization.

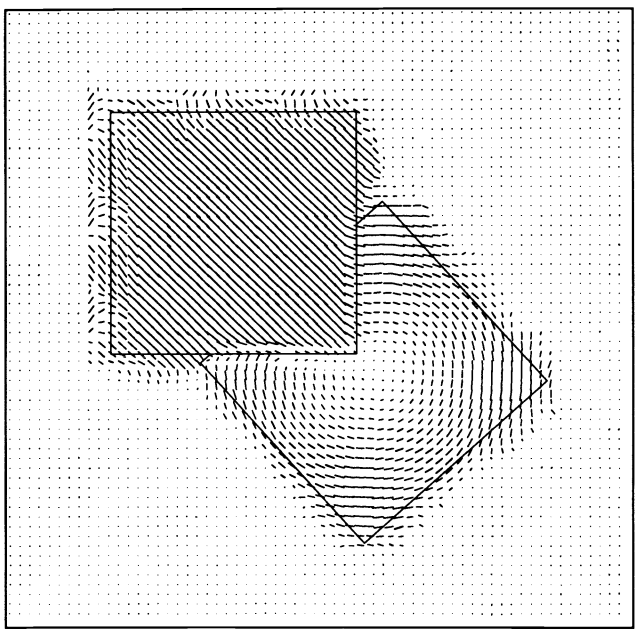


Fig. 4 Field resulting by applying the classical LS algorithm to the SYNTH sequence; without regularization.

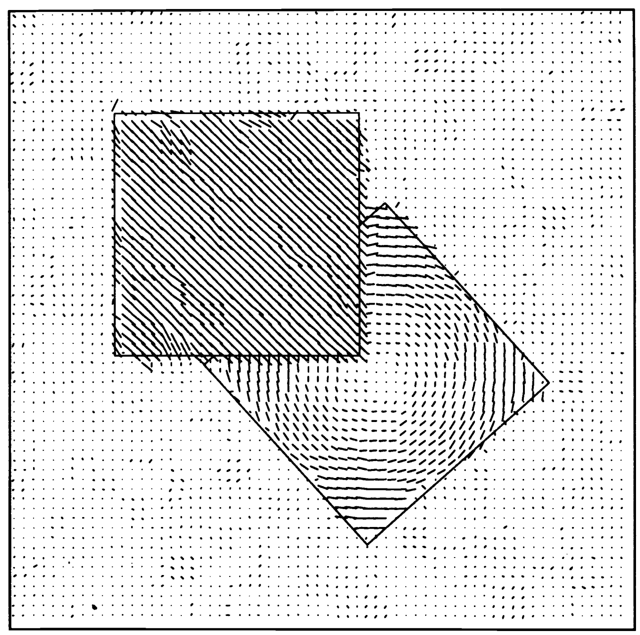


Fig. 6 Field resulting by applying the multiwindow LS algorithm to the SYNTH sequence; without regularization.

this case, no regularization is performed, as the algorithm has an implicit smoothing effect on the produced field. Derivatives E_x, E_y, and E_t are approximated by finite symmetrical differences; temporal derivatives below a threshold of two gray levels are considered to be effects of noise only and are thus set to zero.

Figure 3 shows the second frame of a synthetic sequence—referred to as *SYNTH*—in which two textured squares 50 pixels wide are moving over a still background. The upper left square translates by 1.5 pixels/frame both down and right and partially occludes the other one, which rotates clockwise by 3 deg/frame. A zero-mean white Gaussian noise ($\sigma^2 = 30$) is added to each frame.

The flow fields resulting from processing the SYNTH sequence by the three test algorithms, along with the computed boundaries of the moving objects, are shown in Figs. 4 through 8; it is evident from a qualitative comparison of these results that the multiwindow LS algorithm performs better than the other two methods, especially on motion discontinuities.

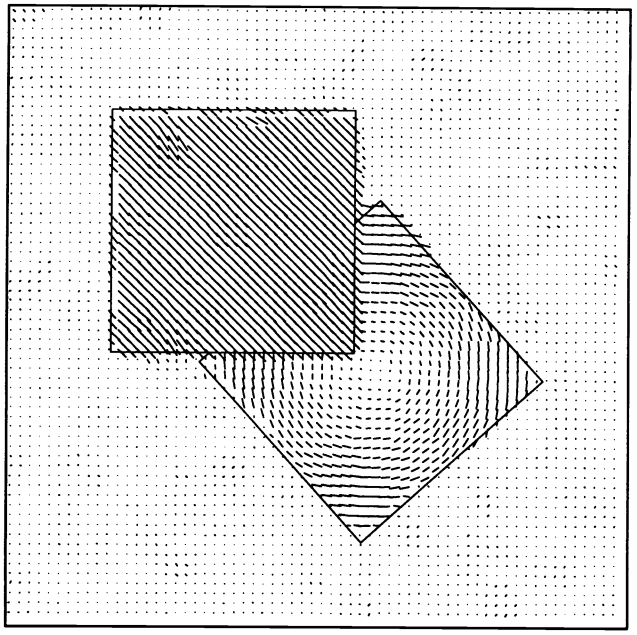


Fig. 7 Field resulting by applying the multiwindow LS algorithm to the SYNTH sequence; with regularization.

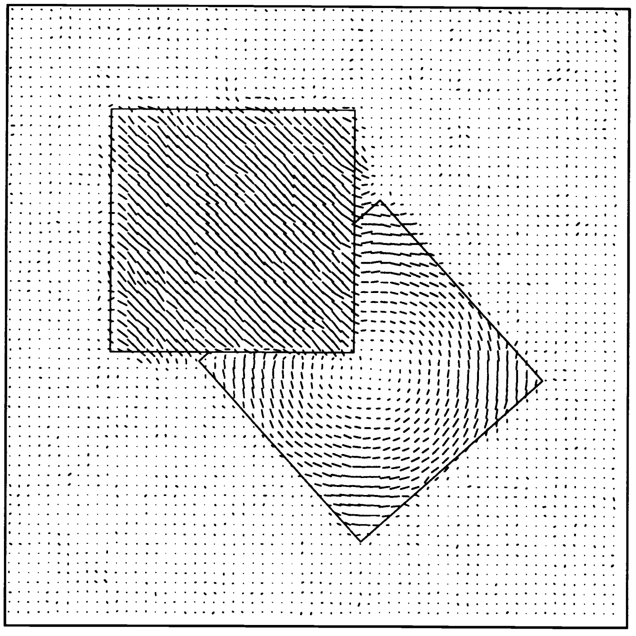


Fig. 8 Field resulting by applying the Horn and Schunck algorithm to the SYNTH sequence.

A quantitative comparison of the performance of the three algorithms can also be made by referring to Table 1, where the rms error (RMSE) together with the processing time required by the various algorithms is shown. It is worth noting that the regularization step has significant effects only on those vectors that are grossly in error (for example, the incorrect vector near the upper corner of the translating square in Fig. 6 has disappeared in Fig. 7), thus having no great influence on RMSE performance. The multiwindow LS approach saves a considerable amount of processing time; in

Table 1 RMSE of the field computed by the various algorithms and relative processing times.

Algorithm	RMSE (edges)	RMSE (overall)	Processing Time (s)
Classical LS without regularization	0.933	0.354	12.0
Classical LS with regularization	0.931	0.346	15.0
Multiwindow LS without regularization	0.789	0.291	3.8
Multiwindow LS with regularization	0.718	0.246	6.8
Horn & Schunck	0.905	0.346	26.6

Fig. 9 Seventh frame of the ROBOX sequence.

fact, because each 5×5 patch is actually a subwindow for nine different 9×9 windows, the computational burden for this algorithm can be made only slightly bigger than that of a classical LS algorithm using 5×5 windows.[22]

Figure 9 shows a frame of the real-world ROBOX sequence. The imaged objects are a cookie box rolling over a table by 3 deg/frame and a small mobile robot approaching the camera plane by 1 cm/frame. The flow fields computed with the classical LS, the multiwindow LS, and the Horn and Schunck algorithms are shown respectively in Figs. 10, 11, and 12. The multiwindow LS approach is again demonstrated to be effective in preserving motion boundaries (compare, for example, Fig. 11 with Fig. 10, where the undesired effect of motion boundary enlargement is particularly evident) and to be more robust than the others with respect to noise.

Figure 13 is a map of the residual error obtained with the multiwindow LS approach; the residual can be taken as a measure of the departure of the OF field from the condition of uniform motion. Note that the highest residual values are reached in the image area where rotation takes place, the nonuniformity of the field being lower where OF is diverging.

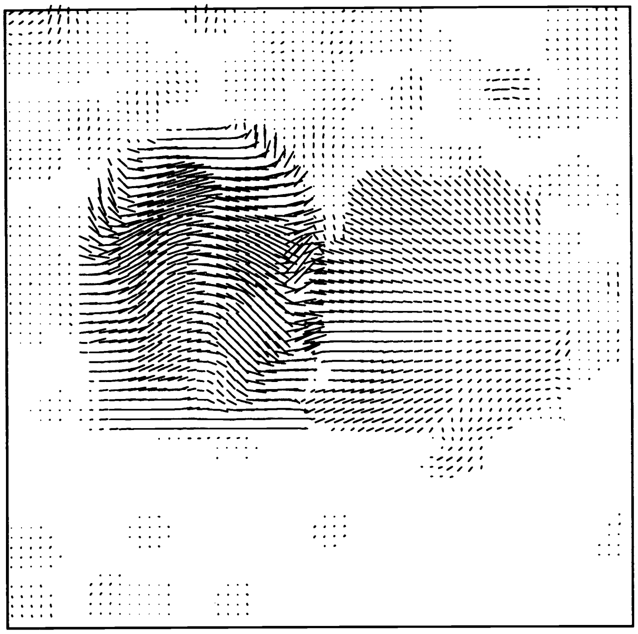


Fig. 10 Field resulting by applying the classical LS algorithm to the ROBOX sequence; with regularization.

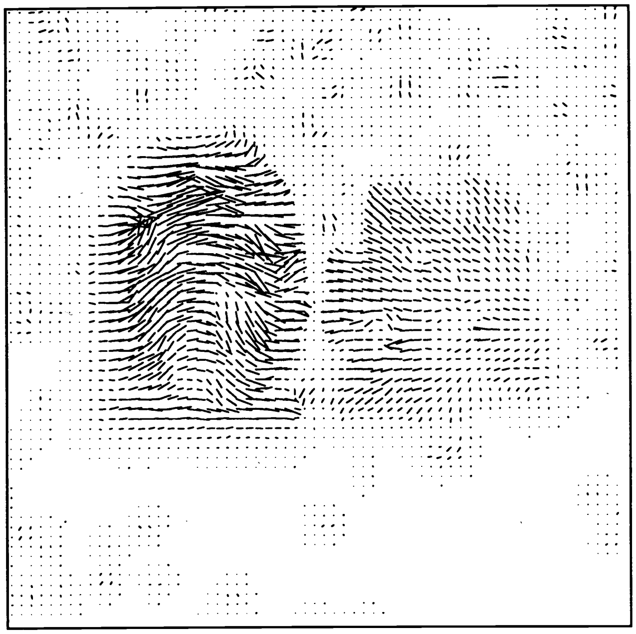


Fig. 12 Field resulting by applying the Horn and Schunck algorithm to the ROBOX sequence.

Fig. 11 Field resulting by applying the multiwindow LS algorithm to the ROBOX sequence; with regularization.

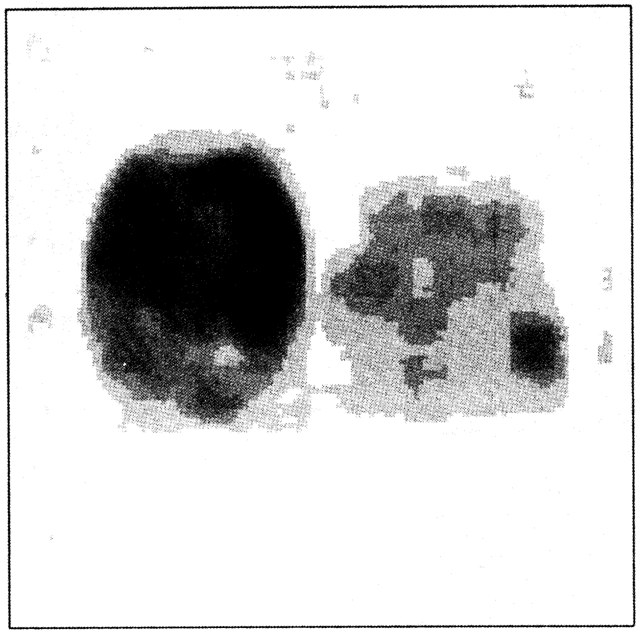


Fig. 13 Map of the normalized residual error resulting from the application of the multiwindow LS algorithm to the ROBOX sequence; darker points denote higher residual values.

5 Conclusions and Future Work

In this paper, a multiwindow LS approach to OF computation has been proposed. While maintaining the advantage of noise insensitivity of a classical LS algorithm, this technique avoids blurring at motion boundaries by the use of a suitable partitioning of the neighborhood of each pixel.

The algorithm is quite simple, nonparametric, and fast, since computations are carried out in a local and noniterative fashion. The ill-conditioned nature exhibited by local techniques in some cases (e.g., low or unidirectional image gradient), the presence of noise, and the absence of any *a priori* smoothing make an *a posteriori* regularization of the resulting flow field necessary. This is accomplished by a nonlinear technique, vector median filtering, which exhibits good noise removal performance and preserves the detected motion discontinuities well.

Further observations can be made, suggesting some directions for future work.

The multiwindow approach proposed here is actually independent of the particular technique used to compute the optical flow in each subwindow. That is, local techniques other than LS, either novel or conventional, could indeed be adopted, given that an *a posteriori* error measure is available. Thus, the multiwindow approach is to be considered a *computational superstructure* applicable to a suitable *substrate* algorithm. A similar concept can be found in the paper by Schnörr,[23] in which the Horn and Schunck algorithm is used inside adequate subsets of the image plane.

The LS technique was chosen in this work for its simplicity and speed; on the other hand, experiments have proved that this technique is quite sensitive to gross (impulsive) errors on the constraint lines, frequently occurring at motion boundaries. In attempting to enhance the robustness of the technique against these kinds of errors, two directions are feasible: (1) considering different choices of the number, dimension, and shape of the subwindows and (2) experimenting with alternative traditional or novel *substrate* algorithms. A particularly promising novel local approach that makes use of vector median techniques is currently being tested.[22]

Acknowledgments

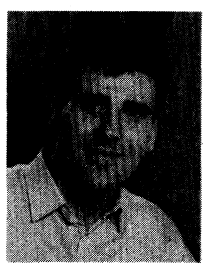
The authors would like to thank the referees for their valuable comments. Robot "Robbie" was kindly provided by Dr. V. Genovese of ARTS Laboratory. This work was done under the support of the "Progetto Finalizzato TRASPORTI 2" of the Italian National Research Council.

References

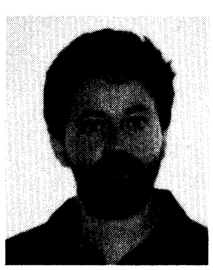
1. G. Adiv, "Determining three-dimensional motion and structure from optical flow generated by several moving objects," *IEEE Trans. Pat. Anal. Mach. Intell.* **7**(4), 384–401 (1985).
2. H. C. Longuet-Higgins and K. Prazdny, "The interpretation of a moving retinal image," *Proc. R. Soc. London* **B208,** 385–397 (1980).
3. S. T. Barnard and W. B. Thompson, "Disparity analysis of images," *IEEE Trans. Pat. Anal. Mach. Intell.* **2**(4), 333–340 (1980).
4. R. M. Haralick and L. G. Shapiro, "Glossary of computer vision terms," *Pat. Recog.* **24**(1), 69–93 (1991).
5. O. Braddick, "A short-range process in apparent motion," *Vision Res.* **14,** 519–527 (1974).
6. B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intell.* **17,** 185–203 (1981).
7. E. C. Hildreth, "Computation underlying the measurement of visual motion," *Artificial Intell.* **23,** 309–354 (1984).
8. H. H. Nagel, "On the estimation of optical flow: relations between different approaches and some new results," *Artificial Intell.* **33,** 299–324 (1987).
9. S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pat. Anal. Mach. Intell.* **6**(6), 721–741 (1984).
10. F. Heitz and P. Bouthemy, "Multimodal motion estimation and segmentation using Markov random fields," in *Proc. IEEE Intl. Conf. Pat. Recog.,* pp. 378–383 (June 1990).
11. M. J. Black, "Combining intensity and motion for incremental segmentation and tracking over long image sequences," in *Proc. 2nd Euro. Conf. Computer Vision,* S. Margherita Ligure (Italy), pp. 485–493 (May 1992).
12. J. Konrad and E. Dubois, "Bayesian estimation of motion vector fields," *IEEE Trans. Patt. Mach. Intell.* **14**(9), 910–927 (1992).
13. K. Wohn, L. S. Davis, and P. Thrift, "Motion estimation based on multiple local constraints and nonlinear smoothing," *Pat. Recog.* **16**(6), 563–570 (1983).
14. B. G. Schunck, "Image flow segmentation and estimation by constraint line clustering," *IEEE Trans. Pat. Anal. Mach. Intell.* **11**(10), 1010–1027 (1989).
15. A. Mitiche, Y. F. Wang, and J. K. Aggarwal, "Experiments in computing optical flow with the gradient-based, multiconstraint method," *Pat. Recog.* **20**(2), 173–179 (1987).
16. S. Uras, F. Girosi, A. Verri, and V. Torre, "A computational approach to motion perception," *Biological Cybernetics* **60,** 79–87 (1988).
17. V. Markandey and B. E. Flinchbaugh, "Multispectral constraints for optical flow computation," *Proc. 3rd Int. Conf. Computer Vision,* Osaka, pp. 38–41 (December 1990).
18. A. Mitiche, R. Grisell, and J. K. Aggarwal, "On smoothness of a vector field—application to optical flow," *IEEE Trans. Pat. Anal. Mach. Intell.* **10**(6), 943–949 (1988).
19. M. Nagao and T. Matsuyama, "Edge preserving smoothing," *Comp. Graphics Image Proc.* **9**(4), 394–407 (1979).
20. J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proc. IEEE* **78**(4), 678–689 (1990).
21. M. Barni, V. Cappellini, and A. Mecocci, "The use of different metrics in vector median filtering: application to fine arts and paintings," *Proc. Euro. Signal Processing Conf. EUSIPCO '92,* Brussels, Belgium, August 24–29, 1992, pp. 1485–1488 (1992).
22. C. Colombo and F. Bartolini, unpublished manuscript (1992).
23. C. Schnörr, "Computation of discontinuous optical flow by domain decomposition and shape optimization," *Int. J. Computer Vision* **8**(2), 153–165 (1992).

Franco Bartolini received a degree in electronic engineering from the University of Florence, Italy, in 1991. Since January 1992 he has worked at the Image Processing and Communications Laboratory of the Department of Electronic Engineering of the University of Florence. His research interests include image sequence processing for machine vision and image compression and applications of neural networks to image and signal processing.

Vito Cappellini obtained a degree in electronic engineering from Politecnico di Torino in 1961. He did research at the Istituto di Ricerca sulle Onde Elettromagnetiche of the National Research Council and at the Dipartimento di Ingegneria Elettronica of Florence University, where he was appointed a full professor of communications in 1975. His main research interests are digital signal/image processing, digital communications, remote sensing, biomedicine, and robotics. He has published over 250 papers, contributed to several books, and is on the editorial board of several international journals. He is president of the Italian Telecommunications Group. In 1984 he received the IEEE Centennial Medal. Cappellini is a senior member of the IEEE, and a member of EURASIP, AEI, IAPR-IC, AIT, and ARI.



Carlo Colombo received a degree in electronic engineering from the University of Florence, Italy, in 1992. In September 1992 he joined the Advanced Robotics Technology and Systems Laboratory of Scuola Superiore di Studi Universitari e di Perfezionamento S. Anna, Pisa, Italy, where he is currently a PhD student. His research interests include image processing, active vision, and vision-based robot control.



Alessandro Mecocci received a degree in electronic engineering from the University of Florence, Italy, in 1983. From 1983 to 1986 he worked as a consultant for several companies, and in 1986 he became responsible for research and development at the Data Processing System Company in Florence, where he developed research on artificial vision systems, robotics, shape analysis, and recognition. In 1990 he joined the Department of Electronic Engineering of the University of Florence as a researcher. In 1991 he joined the Department of Electronics of the University of Pavia, Italy, as an associate professor. His present research interests are image processing and coding, pattern recognition, remote sensing, and artificial intelligence. Dr. Mecocci is a member of IEEE and AEI.