Automatic enhancement of noisy image sequences through local spatiotemporal spectrum analysis

Oscar Nestares*
Carlos Miravet
Javier Santamaria, MEMBER SPIE
SENER Ingeniería y Sistemas, S.A.
Aerospace Division
Severo Ochoa 4
28760 Tres Cantos, Madrid
Spain

Rafael Navarro
Instituto de Óptica "Daza de Valdés"
(C.S.I.C.) Serrano 121
28006 Madrid
Spain

Abstract. A fully automatic method is proposed to produce an enhanced image from a very noisy sequence consisting of a translating object over a background with a different translational motion. The method is based on averaging registered versions of the frames in which the object has been motion-compensated. Conventional techniques for displacement estimation are not adequate for these very noisy sequences, and thus a new strategy has been used, taking advantage of a simple model of the sequences. First, the local spatiotemporal spectrum is estimated through a bank of multidirectional, multiscale third-order Gaussian derivative filters, yielding a representation of the sequence that facilitates further processing and analysis tasks. Then, energy-related measurements describing the local texture and motion are easily extracted from this representation. These descriptors are used to segment the sequence according to a local joint measure of motion and texture. Once the object of interest has been segmented, its velocity is estimated applying the gradient constraint to the output of a directional bandpass filter for all pixels belonging to the object. Velocity estimates are then used to compensate the motion prior to the average. The results obtained with real sequences of moving ships taken under very noisy conditions are highly satisfactory, demonstrating the robustness and usefulness of the proposed method. [S0091-3286(00)01906-1]

Subject terms: image enhancement; noisy sequences; joint transforms; image segmentation; motion estimation.

Paper 990099 received Mar. 8, 1999; revised manuscript received Nov. 29, 1999; accepted for publication Nov. 30, 1999.

1 Introduction

Image sequences contain highly redundant temporal information that can be used for restoration, resolution improvement, or enhancement of the scene. One particularly simple case is a sequence of a still scene taken under time-varying degradations, such as random turbulence or noisy conditions. The effects of turbulence[1] or noise[2,3] can be diminished by simply averaging several frames, which tends to cancel the random variations.

Most image sequences, however, contain some motion. This motion causes different frames to contain the same object of interest but shifted and, possibly, rotated and scaled, thus providing slightly different views of the object. This fact has been used to reduce aliasing and then to improve resolution,[4–6] especially in sparse sampling arrays used in infrared imaging systems for low-noise situations. In general, motion prevents the direct temporal integration of the frames, and hence it is difficult to reduce large amounts of noise. In these cases, it is necessary to determine the motion of the object of interest to compensate for it, prior to integration.

Image motion or optical flow estimation is an ill-posed problem,[7] usually giving noisy results even in noise-free sequences.[8] Recent results suggest that an optimal strategy is to simultaneously estimate motion and segment the sequence according to its motional contents.[9,10] This strategy is more robust than a direct estimation of optical flow. Such methods usually segment the sequence according to motion and compute optical flow iteratively (using algorithms like expectation/maximization), each step refining the previous one.[11] They have been successfully tested in complex sequences that include rotations and scaling, but that are free of noise. They are computationally expensive, since the convergence is usually slow. Other methods have been proposed for the simultaneous estimation of multiple optical flow present in the same spatial location,[12] which are optimal when the sequence can be locally expressed as the linear superposition of several motion signals (*additive transparency*). This is not the case of our image sequences (as explained below), because there are not multiple motions at the same spatial location except at the occlusion boundaries, where the signal is not simply described as the linear superposition of several motion signals.

In this paper we consider the case of sequences containing an object of interest undergoing translational motion against a background undergoing a different translational

*Current address: Stanford University, Department of Psychology, Jordan Hall, Bldg. 420, Stanford, California 94305.
E-mail: oscar@white.stanford.edu.

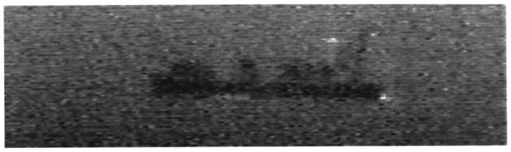


Fig. 1 Sample frame of a typical sequence that has been taken from a static CCD camera in the visible range, and that corresponds to a ship translating to the left.

motion, and with a very low signal-to-noise ratio due to the poor imaging conditions. The absence of relevant rotations or scaling of the target allows applying more robust methods than in the general case. This type of sequences is of great interest in certain applications, like ground-based maritime surveillance systems.

The method applied for image enhancement is based on a visual representation of the image sequence that provides an estimate of the local spatiotemporal spectrum.[13] This generic scheme of representation was previously developed as a first multipurpose stage of sequence processing that was inspired by human vision and that facilitates further analysis and processing. In particular, it helps to segment, robustly, very noisy sequences according to their texture and motion contents, a task where our visual system shows high performance. After segmenting the moving object, we estimate its velocity by applying a modified version of a method for optical flow estimation based on the same representation of the image sequence and therefore involving little additional cost.[13] The estimated velocity is then used to register the object in all frames, which are finally integrated to improve the signal-to-noise ratio.

We have tested the performance of each step of the method with synthetic test sequences contaminated with spatiotemporal noise of the same power as the original sequence (signal-to-noise ratio of 0 dB). The method is able to detect and segment the moving object, and to estimate its velocity reliably. It was also tested with real sequences acquired from maritime surveillance (infrared and visible) imaging systems. The results show a remarkable enhancement in all cases.

2 Method

The method for image-sequence enhancement has been developed for a particular class of sequences that consist of an object of interest moving against a background and contaminated with spatio-temporal noise, so that the visibility of the target is very low. Concretely, these are the main assumptions about the sequences for which this method is applicable:

1. There is one object of interest, or *target*, which is undergoing a smooth translation motion, i.e., the image velocity of all the points belonging to the object is the same at a particular time (i.e., there are no noticeable rotations or scaling), and it varies slowly in time.
2. The object is against a background that is also undergoing a smooth translation motion, different from the motion of the target. We consider a static background as the particular case of zero velocity.
3. The resulting sequence is contaminated with white spatiotemporal noise of high power (in extreme cases, the same power as the signal).

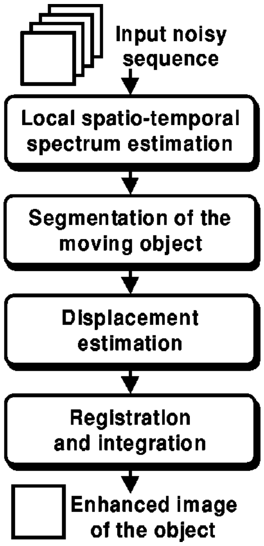

Input noisy sequence
Local spatio-temporal spectrum estimation
Segmentation of the moving object
Displacement estimation
Registration and integration
Enhanced image of the object

Fig. 2 Block diagram of the proposed method for image enhancement from noisy sequences.

Figure 1 shows a frame of a typical sequence that consists of a ship translating from right to the left, and that was taken with a visible-light CCD camera under noisy conditions. The ship is both blurred and highly corrupted by noise, making its recognition difficult.

The proposed method obtains an enhanced version of the object of interest automatically, following these steps: (1) segmenting the object of interest; (2) estimating the velocity of the object and hence the displacements between frames; and (3) averaging motion-compensated versions of the frames together.

The high level of noise present in our sequences makes the two first steps critical and especially difficult, requiring highly robust methods. Therefore, we have avoided using standard techniques relying on the analysis of static frames only (like segmentation by thresholding, or displacement estimation by correlation or block matching). Instead, the proposed method uses local spatiotemporal frequency information at each point in the sequence. The main stages of the method, displayed in the block diagram of Fig. 2, are the following:

1. The local spatiotemporal spectrum of the sequence is estimated through a bank of multidirectional, multiscale spatiotemporal bandpass filters. This produces a visual representation of the image sequence[13] that facilitates further analysis tasks.
2. The target is segmented from the background, using spatiotemporal descriptors derived from the above visual representation of the sequence.[14] These descriptors are defined at each point as local measurements of the energy in each frequency band, and hence of the texture-motion energy in each location.[15] These descriptors are an extension of those used previously in multichannel static texture segmentation.[16–19] In

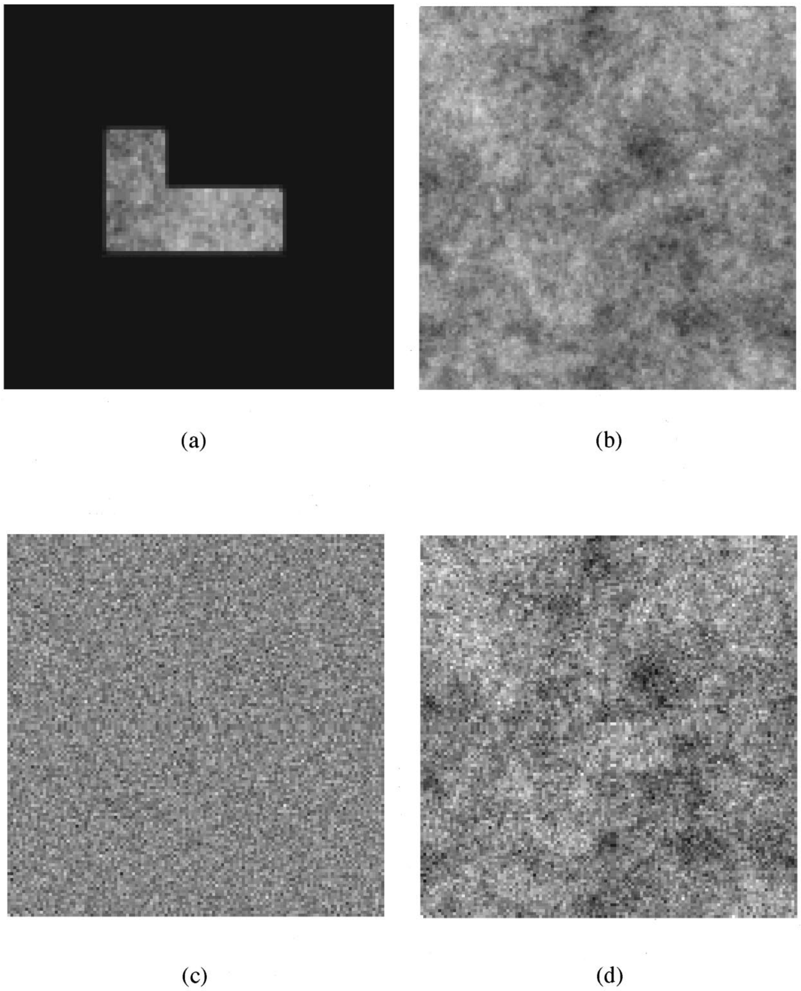


Fig. 3 Generation of synthetic test sequences: (a) moving object filled with a synthetic fractal noise texture; (b) background consisting of a different sample of the same fractal noise; (c) additive spatiotemporal Gaussian white noise; (d) resulting frame.

this dynamic case, differences in velocity, texture, or both are the key features to discriminate the target from the background.

3. The target's velocity is estimated for each frame, to compute its displacements and to register it in the different frames. Velocity is estimated using a modified version of a previously developed method for probabilistic multichannel optical flow estimation.[13] Optical flow is estimated by applying the gradient constraint[20] to the output of a directional bandpass filter, chosen to respond mainly to the moving object, and thus eliminating a great amount of noise. In addition, since the prior segmentation has already detected the points of the target, which share the same velocity, combining gradient constraints in this large number of points strongly improves the robustness and accuracy of the velocity estimate.
4. The motion is compensated for every image so that the object appears registered, making possible the final integration of all frames to give a single enhanced image of the object of interest.

We give a detailed explanation of these stages in the following subsections. To test and optimize these stages, we have generated synthetic test sequences imitating real ones, but that constitute an extreme case where the only cue to segment the object is motion. The patterns used for the object and the background, displayed in Figs. 3(a) and 3(b), respectively, are different samples of the same spatial fractal noise with power spectra proportional to $1/f_s^2$, where f_s is the radial spatial frequency. An ensemble of test sequences has been generated translating the object and the background with different velocities, and adding different amounts of spatiotemporal Gaussian white noise [Fig. 3(c)]. A frame of one of the resulting sequences (with $\mathrm{SNR}=0\,\mathrm{dB}$) is shown in Fig. 3(d), where the object is invisible to the eye. The object becomes clearly perceptible, however, when the sequence is displayed in motion.

2.1 *Local Spectrum Estimation*

The local spectrum has been estimated using a bank of linear, multidirectional, multiscale spatiotemporal bandpass filters. For this purpose we have applied a previously developed scheme for visual representation of image sequences that was designed as a multipurpose preprocessing stage to facilitate many image-sequence processing and analysis applications.[13]

The basis functions of this scheme are spatiotemporal third-order Gaussian derivatives along specified spatiotemporal directions (GD3). Gaussian derivatives have been used by different authors[21,22] to model the early linear stages of the visual system. These filters have their tuning (peak) frequencies distributed over the surface of a sphere (more generally, over an ellipsoid) for a given scale. The basis functions can be expressed as linear combinations of the separable functions obtained by third-order (partial) differentiation of a spatiotemporal Gaussian. The general expression, in the spatiotemporal frequency domain, for the separable basis GD3 is the following:

$$\frac{\partial^3 g(x,y,t)}{\partial x^{3-k-l}\,\partial y^k\,\partial t^l} \overset{F}{\leftrightarrow} (j2\pi f_x)^{3-k-l}(j2\pi f_y)^k(j2\pi f_t)^l \times G(f_x)G(f_y)G(f_t), \qquad (1)$$

where $\overset{F}{\leftrightarrow}$ means Fourier transformation, (f_x, f_y, f_t) are the frequencies along the (two) spatial and the temporal axes, and $g(x)$ [with Fourier transform $G(f_x)$] is the following basic 1-D Gaussian function:

$$g(x)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{x^2}{2\sigma^2}\right)\overset{F}{\leftrightarrow}G(f_x)=\exp\left[-\frac{\sigma^2}{2}(2\pi f_x)^2\right]. \qquad (2)$$

There are 10 independent GD3s at each scale, which correspond to all possible combination of indices $0\leqslant l\leqslant 3$ and $0\leqslant k\leqslant 3-l$. A very efficient implementation using 1-D convolution masks is possible, since the set of partial derivatives is separable. The directional filters needed to obtain good samples of the local spectrum are easily obtained from the set of partial derivatives through linear combination, a general property of directional derivatives closely related with the steerability of derivative filters[23,24]:

$$\frac{\partial^3 g(x,y,t)}{\partial \mathbf{h}_0}=\sum_{l=0}^{3}\sum_{k=0}^{3-l}\binom{3}{l}\binom{3-l}{k}\cos^{N-l-k}\theta_0\sin^k\theta_0 \times\sin^{N-l}\varphi_0\cos^l\varphi_0\frac{\partial^3 g(x,y,t)}{\partial x^{3-k-l}\,\partial y^k\,\partial t^l}, \qquad (3)$$

Table 1 Spherical angular coordinates of the directions where the 10 directional GD3s are placed at each scale.

θ	φ
0	$\frac{1}{2}\pi$
$\frac{1}{4}\pi$	$\frac{1}{2}\pi$
$\frac{1}{2}\pi$	$\frac{1}{2}\pi$
$\frac{3}{2}\pi$	$\frac{1}{2}\pi$
0	arcsin $\frac{5}{8}$
$\frac{2}{5}\pi$	arcsin $\frac{5}{8}$
$\frac{4}{5}\pi$	arcsin $\frac{5}{8}$
$\frac{6}{5}\pi$	arcsin $\frac{5}{8}$
$\frac{8}{5}\pi$	arcsin $\frac{5}{8}$
—	0

where $\mathbf{h}_0$ is the spatiotemporal direction of differentiation, which is determined by its spherical angular coordinates (θ_0, φ_0).

Table 1 contains the spherical angular coordinates of the 10 directions in the frequency domain that we have chosen to place the directional GD3 channels, at each scale. Four channels are placed in the static plane, defined by $f_t=0$, all their centers (tuning frequencies) having $\varphi=\pi/2$ in spherical coordinates, and being equally spaced in azimuth. Five channels are placed in a dynamic plane parallel to the static plane. If we want to maintain the same amount of overlap in azimuth for these five dynamic channels as for the static ones, then it follows that $\varphi=\arcsin\frac{5}{8}$ for this dynamic plane. (Each channel has two lobes, but whereas the four static channels yield a total of eight lobes in one single static plane, dynamic channels produce five lobes in the positive dynamic plane, and five more lobes in another negative plane placed symmetrically with respect to the origin.) The remaining (10th) channel is placed on the temporal-frequency axis.

The highest frequency scale is placed at a radial tuning frequency of half the Nyquist frequency (1/4 cycle/sample). The GD3 filters have been implemented efficiently using the small 1-D (nine-tap) convolution masks shown in Table 2, obtaining high-fidelity approximations of the theoretical

Table 2 One-dimensional nine-tap convolution masks corresponding to the Gaussian prefilter (g_0) and their derivatives of first (g_1), second (g_2), and third order (g_3), and to a five-tap cubic B-spline filter. Only the positive-axis coefficients are shown, since g_0, g_2, and the cubic B-spline are even, and g_1 and g_3 are odd.

g_0	g_1	g_2	g_3	Cubic B-Spline
3.616E−1	0	−3.254E−1	0	0.375
2.400E−1	−2.072E−1	−4.084E−2	4.107E−1	0.25
6.968E−2	−1.203E−1	1.468E−1	−3.534E−2	0.0625
9.066E−3	−2.347E−2	5.036E−2	−8.794E−2	—
4.437E−4	−1.541E−3	6.428E−3	−1.934E−2	—

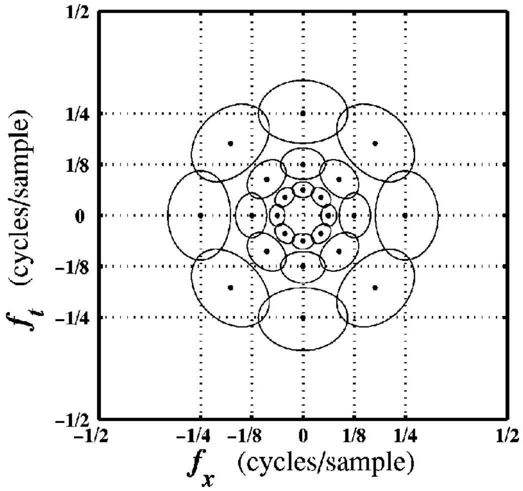


Fig. 4 Schematic view of the directional GD3 channels, as a contour-level plot, in a slice of the 3-D spatiotemporal Fourier space corresponding to (f_x, f_t).

frequency responses (to better than 30 dB). Once the response to the separable (partial) derivative filters is obtained, it is straightforward to obtain the response to the directional ones, applying Eq. (3). Coarser scales have been obtained using an efficient pyramidal implementation, where the same set of filters corresponding to the finest scale is applied to successive subsampled versions (in space and time by a factor of 2) of the original sequence. The filter used to avoid aliasing in the subsampling operations is the five-tap cubic B-spline shown in Table 2. The resulting distribution of channels in the frequency domain is shown in Fig. 4. This schematic view displays one slice including one spatial frequency axis (f_x) and the temporal-frequency axis (f_t). Channels corresponding to three spatiotemporal scales are represented by level curves of uniform magnitude response.

This implementation introduces a delay of 4 frames for the highest resolution level. The delay is multiplied by 2 as we move to coarser resolution levels. If the delay for coarser scales becomes critical for some applications, it is possible to use a version of the filter bank that uses multiresolution only in space,[13] so that the maximum temporal delay is always 4 frames. The main disadvantage of this version is that the samples of the spatiotemporal frequency domain are not evenly distributed.

2.2 *Segmentation*

The target is segmented by unsupervised pixel classification, consisting of an automatic grouping of the pixels corresponding to the moving object. This process requires a local description of specific features that serve to discriminate between object and background. One such feature is precisely motion, since object and background display different motions in the sequences of interest.

Motion can be described taking into account the special form that takes the spatiotemporal spectrum of a given pattern undergoing uniform translation. It is well known that this spectrum is confined to a plane, whose azimuth and elevation depend on the orientation and modulus of the velocity, respectively.[25] More precisely, this plane is given by

$$f_x v_x + f_y v_y + f_t = 0, \tag{4}$$

where (f_x, f_y, f_t) are the components of the spatiotemporal frequency vector, and (v_x, v_y) the components of the velocity vector, respectively. The spatiotemporal spectrum of such a moving pattern is formed by the projection, parallel to the temporal frequency axis, of the 2-D spatial spectrum of the spatial pattern (which is contained in the static plane given by $f_t = 0$) onto the plane given by Eq. (4). Patterns undergoing different motions will have their spectra in separated planes. Thus, each pixel in the sequence can be characterized using measurements of the local energy of the spatiotemporal spectrum (joint texture-motion descriptors). The local spatiotemporal spectrum will be different for pixels belonging to regions undergoing different motions or having different textures. Local spectrum descriptors can be directly obtained from the representation of the sequence described in Sec. 2.1.

2.2.1 *Local spectrum energy descriptors*

The GD3 filter bank samples the local spectrum, but it is not convenient to use the direct output of the filters as descriptors, since these outputs depend on the local phase, and thus they are not shift-invariant. This problem is the same as in static texture discrimination, where it is usual to apply a nonlinearity at the output of the filter bank[16,19,26] to obtain shift-invariant descriptors. Thus, we have extended a previous method for static texture segmentation[19] to the present problem of spatiotemporal segmentation. It has been shown that the complex modulus produces better results than the energy and other tested nonlinearities.[27]

This can be implemented by first obtaining complex samples of the local spectrum, using a pair of filters in phase quadrature, and then computing the moduli of these complex samples. However, to avoid designing and computing the responses to the quadrature pair corresponding to each GD3 filter, we have followed an approximate strategy that uses only the responses of the GD3 filters. The modulus of a complex sample obtained through a pair of quadrature filters is equivalent to that of the response to one of the filters, previously shifted in the frequency domain so that its peak coincides with the zero frequency, and filtered with an appropriate ideal lowpass filter. In our case, since the GD3 filters are not perfectly bandlimited and the ideal lowpass filter has been approximated using a spatiotemporal separable cubic B-spline filter (see Table 2), the result is not exactly the same as the modulus of the quadrature pair. Nevertheless, it is a very good approximation for our purposes.

Figure 5 illustrates how the signal (modulus) is distributed across the 10 channels of the highest frequency level, for the background (a) and the object (b) of the synthetic test sequence of Fig. 3. The channel energy is represented by two opposite spheres centered at the channel's tuning frequency, and whose size is proportional to the average modulus of the channel response inside the corresponding region (background or object). The graph in panel (a) corresponds to the average energy distribution of each channel in the static background. In this case, the static channels are

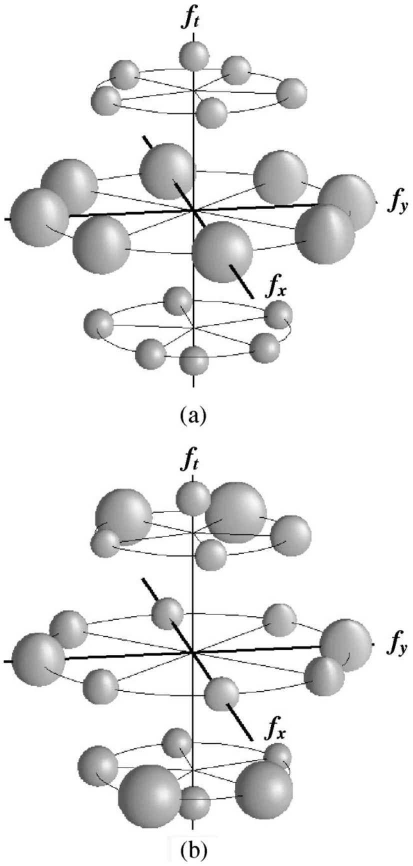


Fig. 5 Schematic view of the average energy of the finest-scale channels for the two regions, (a) background and (b) moving object, of the synthetic test sequence of Fig. 3. Each channel is represented by a pair of spheres centered at its tuning frequency, whose radius is proportional to the average energy inside the corresponding region.

most excited (bigger spheres), but dynamic channels also display a moderate amount of energy, coming from the spatiotemporal additive white noise. The moving object (b) displays a clearly different distribution of energy. Here, most static channels are excited solely by the spatiotemporal white noise. In addition, two dynamic channels, which are close to the plane given by the translation velocity of the object, exhibit higher responses (bigger spheres), while those dynamic channels that are far away from that plane are not excited by the target, but only by the white noise (smaller spheres). The approximately constant size of the smaller spheres in both cases is the result of the white noise.

These spatiotemporal spectral descriptors seem to be able to characterize, and hence to discriminate, regions undergoing different translational motions, even under extremely noisy conditions, such as in the example shown here. Although Fig. 5 shows the average, over all pixels, of the sequence belonging to the background (a) or target (b), it is worth noting that the descriptors are defined at each point, on the basis of a local spatiotemporal neighborhood, whose size is adapted to the frequency band according to the multiresolution scheme.

2.2.2 *Selection of the best feature channels*

As we have just seen, descriptors obtained from the modulus (square root of the energy) of the channel responses can be used to discriminate between regions having different textures and/or motions. Each spatiotemporal location in the sequence is then characterized by a set of $10\,(\text{orientations})\times 3\,(\text{scales})=30\,\text{descriptors}$, which are the responses of the channels that are sampling the local spectrum. The number of components of this feature vector (30) is too large, making the computational cost of segmentation almost unaffordable. Moreover, as shown in Fig. 5, many of the channels will not contribute to discriminate object from background, but can even increase the segmentation errors (in particular for channels dominated by noise). Therefore, we have included an additional step, consisting of selecting a reduced number (four) of channels providing maximum discrimination. This selection improves the segmentation and greatly reduces the computational cost.

The method to select of the most representative channels relies on the characteristic features of our input sequences, which contain a moving object against a larger background either static or undergoing a different motion. As shown in Fig. 6, the statistics of the channel responses can provide useful information for selecting them. This figure compares the moduli of two of the ten channels corresponding to the finest scale, for a frame of the test sequence of Fig. 3. One channel [Fig. 6(a)] is well tuned to the target motion, and therefore displays higher values (brighter gray levels) in the region occupied by it. Conversely, it tends to show lower responses (darker gray levels) in the background region, and therefore, this channel can be useful for discriminating between object and background. Interestingly, this discrimination capability is reflected in its histogram, which displays [Fig. 6(b)] (1) a big lobe near the origin, corresponding to pixels in the background, and (2) a long tail extending towards higher values, which accounts for the strong responses given by the (fewer) pixels belonging to the object. Channels dominated by noise, and hence less appropriate for segmenting the sequence [Fig. 6(c)], display the big lobe near the origin, but not the long tail.

Therefore, a good selection criterion is to pick those channels whose histograms display longer tails (i.e., contain some strong responses). This feature is appropriately characterized by conventional statistical normalized moments, like skewness or kurtosis. As shown in the example of Fig. 6, both skewness and kurtosis are significantly higher (by a factor of nearly 3) for the upper discriminating channel than for the lower one [see values in Figs. 6(b) and 6(d)].

We have checked that both moments produced reasonable results in channel selection for our test sequences, but we finally adopted kurtosis because it produced slightly better results. The number of selected channels is a tradeoff between having as much information as possible and a low computational cost. We found that those four channels (among the total of 30) having the highest kurtosis produced good segmentation results for our sequences at a reasonably low cost.

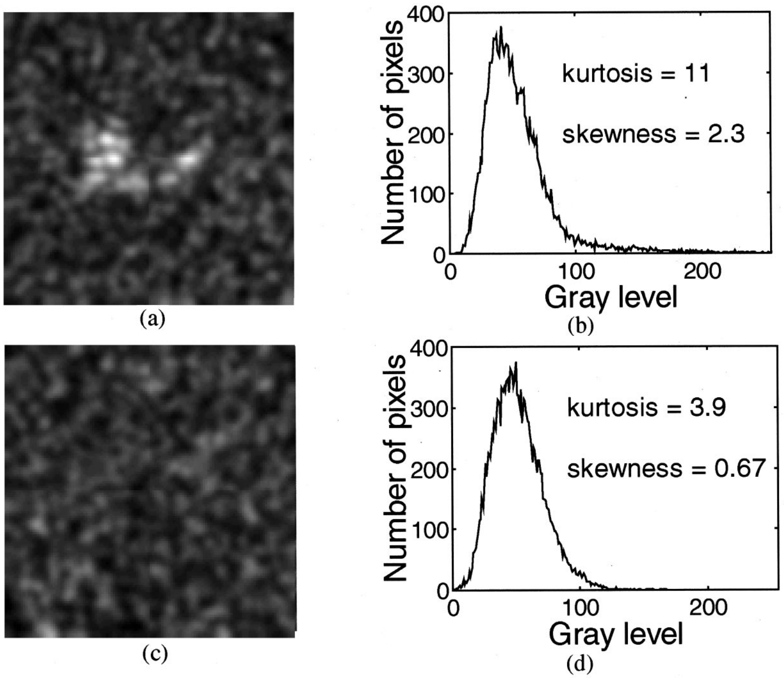


Fig. 6 Moduli of the analytic channel responses and their corresponding histograms (with skewness and kurtosis values) for two typical channels: (a) and (b), a channel well tuned to the moving object; (c) and (d), a channel tuned equally to the object and the background.

2.2.3 *Clustering*

The resulting feature vectors, composed of the moduli of the outputs of the four selected channels, are used to segment the sequence into two classes (object and background), using the standard k-means clustering algorithm.[28] We found that it is convenient to normalize the modulus of every channel independently, between 0 and 1, so that the selected channels enter with equal weight in the clustering process.

The clustering is performed frame by frame, because the velocity of the translating object can change slowly in time. This strategy permits us to initialize the clustering algorithm with the cluster centers found in the previous frame, which are very likely to be close to the present centers, thus reducing the number of iterations significantly for most frames. Finally, the segmentation results are refined by spatial median filtering (window size 11×11) of each segmented frame, which removes possible small isolated regions with a high probability of corresponding to segmentation errors.

Results of segmentation for the synthetic test sequence (Fig. 3) are presented in Fig. 7. Both the original (a) and segmented (b) frames include the true border, overlaid on them to facilitate the visual localization of the target. Gray pixels in the segmented image have been assigned to the background, and the white pixels to the object. The target has been successfully segmented from the background by our fully automatic method. The few segmentation errors are concentrated along the boundaries of the object, because the finite spatial support of the filters is mixing information from object and background. The average percentage of pixels correctly segmented was 98% for an ensemble of synthetic test sequences with different levels of noise and different velocities of the object and the background, which is a highly satisfactory result.

2.3 *Velocity Estimation*

After segmenting the target, the next step consists in estimating its displacement between consecutive frames, which is necessary to register the object in all frames. For this purpose we estimate the velocity of the pixels inside the region occupied by the object, using an optical flow estimation algorithm.[13] Although there are other, simpler techniques (estimating the displacements of the segmented area through correlation, matching, etc.), they are less robust, since segmentation results may be too noisy (particularly in real sequences with very low SNR) to estimate accurately the displacements of the segmented target. In our case, the optical flow estimation is performed by a specially adapted version of a probabilistic multichannel method,[13,29] which is based on the visual representation of the sequence already obtained (see Sec. 2.1), thus requiring little additional cost. This step is performed frame by frame, and it produces a robust and accurate estimation of the translation velocity of the object (and of its associated covariance matrix) for each frame.

The optical flow algorithm is based on the classical gradient constraint,[20] which is obtained under the assumption that intensity levels in the sequence can change their position but remain constant over time, so that the derivative of the image with respect to time is zero. In traditional methods, the gradient constraint is set up at each spatiotemporal location (x,y,t) as follows:

$$s_x v_x + s_y v_y + s_t = 0, \qquad (5)$$

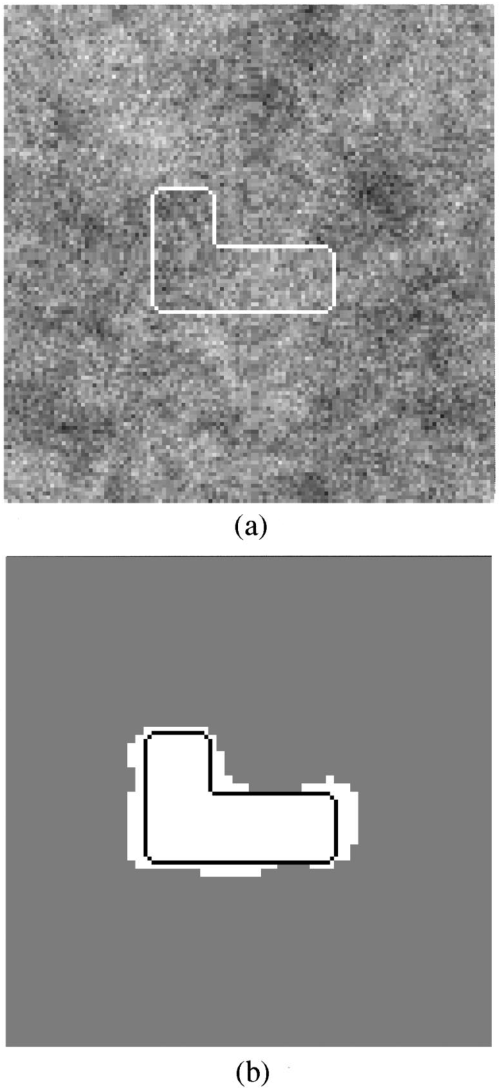


Fig. 7 Segmentation results for the synthetic test sequence: (a) original frame; (b) results of segmentation with two classes. The edge of the true object is overlaid in both images.

where (s_x, s_y, s_t) is the spatiotemporal gradient of the input sequence s, and (v_x, v_y) is the 2-D velocity vector. However, differentiating discrete sequences is numerically unstable and requires some regularization, such as prefiltering with a lowpass filter, to obtain good results. Since both filtering and differentiation are linear operations, it turns out that prefiltering and then differentiating is equivalent to filtering with the derivative of the prefilter. If the regularizing prefilter is chosen to be a Gaussian filter, then spatiotemporal gradients are obtained by filtering the sequence with the appropriate Gaussian derivative filters. Therefore, in our case the subscript on the input sequence s means that the sequence is filtered with the corresponding Gaussian derivative filter.

In the original multichannel method,[13] the gradient constraint is applied to the output of a set of directional second-order Gaussian derivative (GD2) filters. This strategy increases robustness, since the gradient constraint is applied to the meaningful events in the sequence extracted by the GD2 filters, such as bars, edges or texture. Equation (5) takes then the specific form

$$s_{\mathbf{h}_0\mathbf{h}_0x}v_x+s_{\mathbf{h}_0\mathbf{h}_0y}v_y+s_{\mathbf{h}_0\mathbf{h}_0t}=0, \tag{6}$$

where $\mathbf{h}_0$ is the direction vector along which the second derivative is taken. This unique constraint does not permit one to solve for the two components of the image velocity (v_x, v_y) at each point, and thus different methods have been proposed to obtain additional constraints (equations) to solve for the 2-D velocity. In our case, the object has been previously segmented and, according to the assumption of pure translation, all pixels in the object share the same velocity. Thus, it is possible to combine all their corresponding constraints, which gives the following overdetermined linear system:

$$\mathbf{Av}=\mathbf{b}, \qquad \text{with} \quad \mathbf{A}=\begin{pmatrix} s^0_{\mathbf{h}_0\mathbf{h}_0x} & s^0_{\mathbf{h}_0\mathbf{h}_0y} \\ s^1_{\mathbf{h}_0\mathbf{h}_0x} & s^1_{\mathbf{h}_0\mathbf{h}_0y} \\ \vdots & \vdots \\ s^{N-1}_{\mathbf{h}_0\mathbf{h}_0x} & s^{N-1}_{\mathbf{h}_0\mathbf{h}_0y} \end{pmatrix},$$

$$\mathbf{v}=\begin{pmatrix} v_x \\ v_y \end{pmatrix}, \quad \mathbf{b}=-\begin{pmatrix} s^0_{\mathbf{h}_0\mathbf{h}_0t} \\ s^1_{\mathbf{h}_0\mathbf{h}_0t} \\ \vdots \\ s^{N-1}_{\mathbf{h}_0\mathbf{h}_0t} \end{pmatrix}, \tag{7}$$

where the superscript covers all the N points previously assigned to the target by the automatic segmentation. We solve this system using least squares, which gives an estimate of the velocity $(\tilde{\mathbf{v}})$ and of its associated covariance matrix $(\mathbf{C}_\mathbf{v})$:

$$\tilde{\mathbf{v}}=\begin{pmatrix} \tilde{v}_x \\ \tilde{v}_y \end{pmatrix}=(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}, \tag{8}$$

$$\mathbf{C}_\mathbf{v}=(\mathbf{A}^T\mathbf{A})^{-1}\frac{1}{N-1}\sum_{i=0}^{N-1}(s^i_{\mathbf{h}_0\mathbf{h}_0x}\tilde{v}_x+s^i_{\mathbf{h}_0\mathbf{h}_0y}\tilde{v}_y+s^i_{\mathbf{h}_0\mathbf{h}_0t})^2. \tag{9}$$

Covariance matrices are important because they are 2-D confidence measures of the velocity estimates, which are very useful for combining estimates from different sources,[13] (spatiotemporal locations,[30,31] channels, etc.).

The system in Eq. (7) may be set up for different directional GD2 channels, obtaining different estimates that can be combined to obtain a unique less noisy estimate (as is done in the original method[13]). In this particular application, we use only one GD2 channel, which is chosen to be tuned to the target motion (i.e., the GD2 channel with the strongest response to the object). Estimating the velocity at the output of a directional channel tends to increase the aperture problem, because the range of orientations is limited at the output of the filter. However, this effect is not too severe in our case, for two reasons: (1) the GD2 filters are not very selective in orientation, and (2) there is a large number of points from which the gradient constraint is

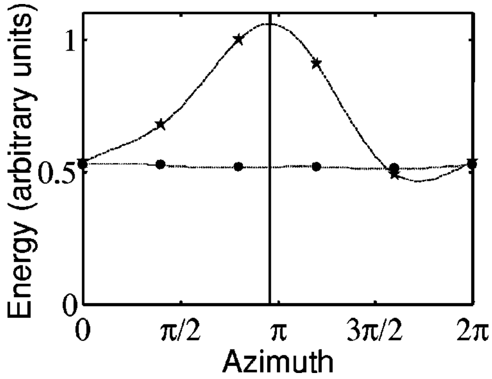


Fig. 8 Interpolation of the five energy measurements for the regions segmented as background (circles, dotted line) and as object (stars, dashed line) as a function of the azimuth, for the test sequence in Fig. 3. The azimuth corresponding to the true maximum for the object velocity of $(4/3,-1/5)$ cycles/pixel is shown as a vertical line.

combined, making highly likely the presence of different orientations, and thus diminishing the aperture problem. On the other hand, the great advantage of using a directional channel tuned to the object is that a large amount of noise is eliminated before estimating the velocity, and it also helps minimize the influence of the background velocity (which can be important for pixels near the border of the target).

The channel selected for the velocity estimation is placed on the dynamic plane, given by $\varphi = \arcsin \frac{5}{8}$ in the frequency domain, so that only the azimuth is tuned to obtain maximum energy response to the object. This is achieved by evaluating the channel energy response as a function of the azimuth of the center frequency, and finding the azimuth of the channel giving maximum response. This function can be represented as a curve that can be easily obtained, since the energy response of derivative channels placed at arbitrary values of azimuth (directions in general) can be interpolated theoretically from the responses of a reduced set of channels because derivative filters are steerable.[24] The responses of five GD3 channels in the dynamic plane are available (Sec. 2.1), from which it is possible to obtain five energy measurements corresponding to the object (averaging the modulus of the response to the analytic channels across all pixels in the region corresponding to the object). In theory, we would need more than nine measurements to obtain an accurate approximation of the interpolation,[24] but we found that using the five available measurements produces good results for all tested sequences. Furthermore, the precision required in azimuth tuning is not critical for this application.

The graph in Fig. 8 shows the results of interpolation for the same test sequence used previously. The five average energy measurements are marked with stars and circles for the object and the background, respectively. The interpolation curves are obtained by applying theoretical interpolation functions for the case of five harmonics (see Appendix F in Ref. 24). The background curve (dashed) is almost constant, and is below the one for the target. Since in this case the background is static, there is no preferred direction, and the motional energy is only due to noise. The curve corresponding to the target (dotted) shows a clear maximum close to the theoretical azimuth (marked by the vertical line) corresponding to the direction of the object velocity, $(4/3,-1/5)$ pixels/frame (known in this synthetic sequence).

It is important to note that there is no need to compute explicitly the response to the directional GD2 channels, since the spatiotemporal gradient $(s_{\mathbf{h}_0\mathbf{h}_0x}, s_{\mathbf{h}_0\mathbf{h}_0y}, s_{\mathbf{h}_0\mathbf{h}_0t})$ of such channels is easily obtained from the representation using the separable basis of GD3 (i.e., from the sequence filtered with all the partial GD3 derivatives) through the following linear combinations[13]:

$$s_{\mathbf{h}_0\mathbf{h}_0x} = h_{0_x}^2 s_{xxx} + h_{0_y}^2 s_{xxy} + h_{0_t}^2 s_{ttx} + 2h_{0_x}h_{0_y}s_{xxy} + 2h_{0_x}h_{0_t}s_{xxt} + 2h_{0_y}h_{0_t}s_{xyt}, \tag{10}$$

$$s_{\mathbf{h}_0\mathbf{h}_0y} = h_{0_x}^2 s_{xxy} + h_{0_y}^2 s_{yyy} + h_{0_t}^2 s_{tty} + 2h_{0_x}h_{0_y}s_{yyx} + 2h_{0_x}h_{0_t}s_{xyt} + 2h_{0_y}h_{0_t}s_{yyt}, \tag{11}$$

$$s_{\mathbf{h}_0\mathbf{h}_0t} = h_{0_x}^2 s_{xxt} + h_{0_y}^2 s_{yyt} + h_{0_t}^2 s_{ttt} + 2h_{0_x}h_{0_y}s_{xyt} + 2h_{0_x}h_{0_t}s_{ttx} + 2h_{0_y}h_{0_t}s_{tty}, \tag{12}$$

where $(h_{0_x}, h_{0_y}, h_{0_t})$ are the components of the directional vector $\mathbf{h}_0$, given that its modulus is one $(|\mathbf{h}_0| = 1)$.

2.4 *Registration and Integration*

The last step is the registration of the moving object and the integration of all registered frames to increase the SNR. Registration can be achieved using the velocities, estimated in the previous stage, frame by frame. However, a further refinement of the velocity estimates of each frame is possible, by taking advantage of the assumption that the velocity of the object does not have large variations along the sequence. One simple strategy would be averaging the velocity estimates within a temporal window, but a more intelligent combination of velocities is possible on taking into account the covariance matrices, which will produce a more robust and accurate estimate.[13,29–31] The combination of a set of estimates of the velocity from N consecutive frames, $\mathbf{v}_i = \{(\tilde{v}_{x_i}, \tilde{v}_{y_i})\}$, taking into account their associated covariance matrices $(\mathbf{C}_{\mathbf{v}_i})$, results in the following new estimate for the velocity (assuming that the different estimates are uncorrelated):

$$\tilde{\mathbf{v}} = \left(\sum_{i=0}^{N-1} \mathbf{C}_{\mathbf{v}_i}^{-1} \right)^{-1} \begin{pmatrix} \displaystyle\sum_{i=0}^{N-1} \frac{\tilde{v}_{x_i}}{\sigma_{u_i}^2} + \frac{\tilde{v}_{y_i}}{\sigma_{uv_i}} \\ \displaystyle\sum_{i=0}^{N-1} \frac{\tilde{v}_{x_i}}{\sigma_{uv_i}} + \frac{\tilde{v}_{y_i}}{\sigma_{v_i}^2} \end{pmatrix}, \tag{13}$$

where $\sigma_{u_i}^2$, σ_{uv_i}, and $\sigma_{v_i}^2$ are the components of the covariance matrix:

$$\mathbf{C}_{\mathbf{v}_i} = \begin{pmatrix} \sigma_{u_i}^2 & \sigma_{uv_i} \\ \sigma_{uv_i} & \sigma_{v_i}^2 \end{pmatrix}. \tag{14}$$

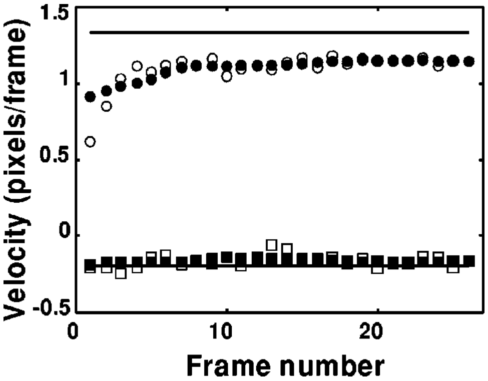


Fig. 9 Estimates of the horizontal (circles) and vertical (squares) velocity for every frame of the test sequence, before (open symbols) and after (solid symbols) the combination in a temporal window of nine frames. Also shown in continuous lines are the real horizontal and vertical velocities of (4/3,−1/5) cycles/pixel.

The graph in Fig. 9 displays the horizontal (circles) and vertical (squares) estimates of the velocity as a function of frame number, before (open symbols) and after (filled symbols) the temporal combination, for a synthetic test sequence with 0-dB SNR. The actual velocity of the object is constant for all frames: (4/3,−1/5) pixels/frame, and these two components are represented as two continuous lines in the graph. The horizontal velocity component has been underestimated. This may have been caused by a bias in the estimate as a result of the simplified noise model that assumes noise-free spatial gradients.[13,29] The velocity estimates are much more stable after the temporal combination, which eliminates much of the noise present in each single-frame estimate.

Finally, the frames are motion-compensated using bilinear backward interpolation, based on the displacements predicted by the velocities estimated in the previous step. Bilinear interpolation introduces a slight lowpass filtering effect, as can be appreciated in Fig. 10. This figure compares the original noise-free object (a), the object in one of the noisy frames (b), and the result of integrating the 26 central frames using the actual velocity (c) and using the velocity estimates provided by our method (d). Even if we use the actual velocity, there is some blurring effect due to the bilinear interpolation [compare images in Figs. 10(a) and 10(c)]. The result obtained by applying all the steps of the automatic method [Fig. 10(d)] is satisfactory: the target is clearly visible, although blurred, mainly in the horizontal direction, as a result of the slight underestimation of that velocity component. Nevertheless, the shape and most of the texture of the original object appear clearly in the final result. It must be also considered that this test case corresponds to a very extreme situation with SNR=0 dB [see Fig. 10(b)].

3 Results

We have applied this method to real sequences of ships recorded with static ground-mounted cameras for maritime surveillance. The sequences were taken using either an infrared camera operating in the 8- to 12-μm range, or a conventional CCD camera operating in the visible range. These sequences were severely affected by noise introduced by the system and (mainly) by bad atmospheric conditions. In these sequences, the observed ship is translating without appreciable rotation or scaling, therefore satisfying the assumptions of the method. We present here results from two visible and one infrared sequence. The enhanced images are compared with one of the original frames.

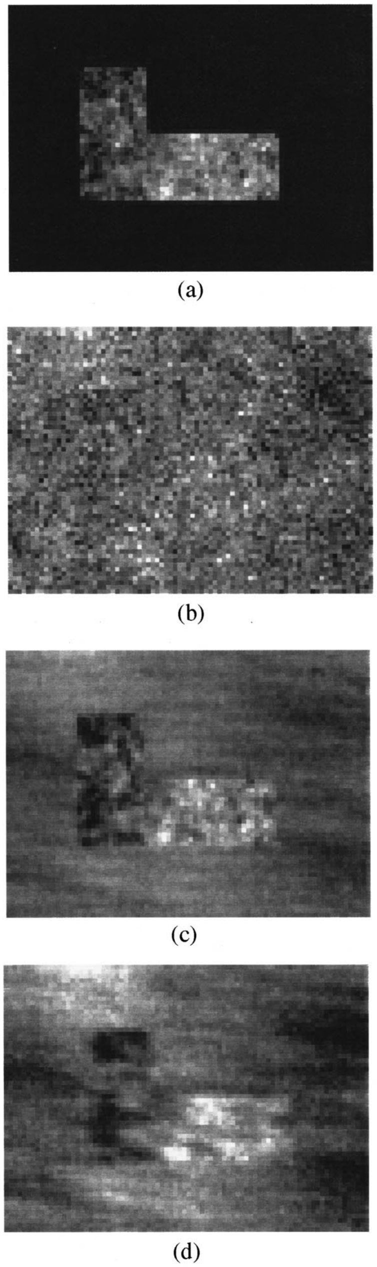
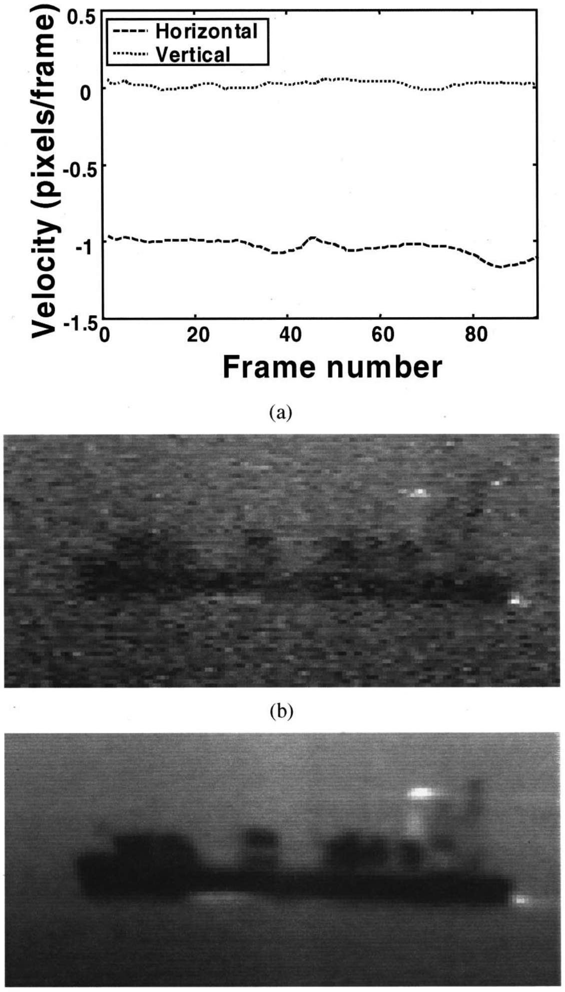


Fig. 10 Results after the integration step for the test sequence: (a) original moving object; (b) noisy frame; (c) integration using the real velocity to register the object; (d) integration using the estimated velocities.

The first sequence is 100 frames long and was taken with a visible camera. The ship is translating from right to the left, with a mean horizontal velocity of about 1 pixel/frame (estimated manually as an average velocity from the global displacement between the first and last frames). Figure 11(a) plots the estimates of the horizontal and vertical velocity components at each frame, being approximately

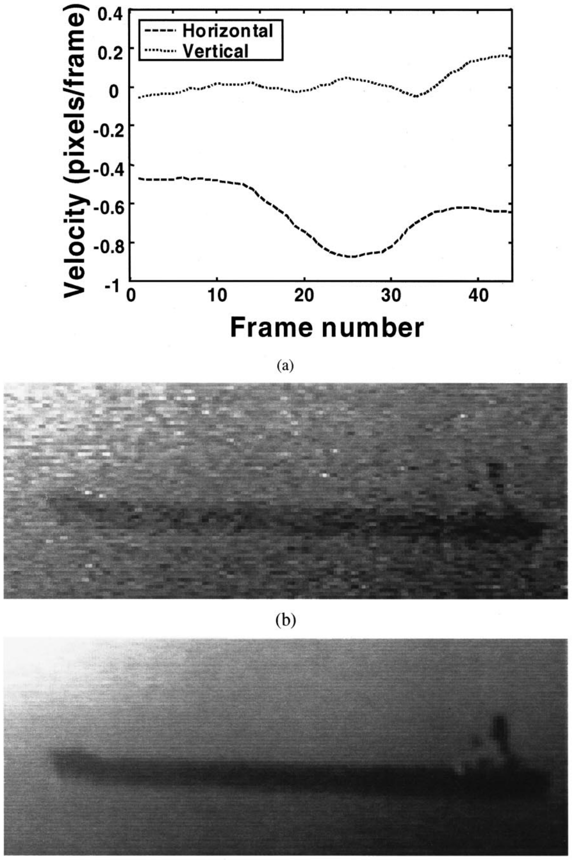


(a)

(b)

(c)

Fig. 11 Results for the first real sequence of a ship taken from a static CCD camera in the visible range: (a) plot of the horizontal (dashed line) and vertical (dotted line) velocities of the object at each frame; (b) subregion of original middle frame; (c) final enhanced image.



(a)

(b)

(c)

Fig. 12 Results for the second real sequence of a ship taken from a static CCD camera in the visible range: (a) plot of the horizontal (dashed line) and vertical (dotted line) velocities of the object at each frame; (b) subregion of original middle frame; (c) final enhanced image.

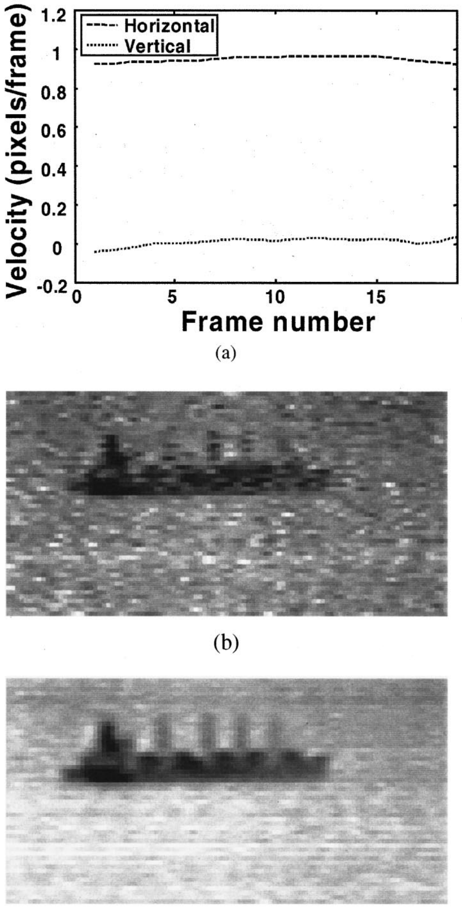
constant and equal to the mean velocity estimated manually. Figure 11(b) displays one of the original frames, where most details of the ship are lost or masked by noise. The result after applying our fully automatic method is shown in Fig. 11(c). The image has been strongly enhanced, the target displays now many details that were not visible in the original frame, and the SNR is much higher. There is again a blurring effect that comes from both the bilinear interpolation and errors in the displacement estimation.

An original frame of the second visible sequence, 50 frames long, is shown in Fig. 12(b). The ship is translating from right to left with an average horizontal velocity of about 0.7 pixels/frame (estimated manually). The horizontal estimated velocities in the graph of Fig. 12(a) oscillate around this average value, between 0.5 and 1 pixel/frame. The vertical velocity component is approximately equal to zero. The noise level is similar here to that in the previous sequence, making the recognition of details difficult, especially on the right side of the ship. These details are clearly visible, however, in the final enhanced image, Fig. 12(c). In addition, the hull of the ship appears more uniform (almost noise-free).

The last example corresponds to a sequence 25 frames long, taken with the infrared camera, where the ship is translating from left to the right at an average horizontal velocity of 0.9 pixels/frame. The estimated velocities are close to this average value, as shown in the graph in Fig. 13(a). An original frame and the resulting enhanced image, both in inverse video (hot areas darker), are also shown in Figs. 13(b) and 13(c), respectively. The original frame is slightly less noisy than in the visible case, since atmospheric conditions affect infrared images less. Nevertheless, there is again a great improvement in the resulting image. Four vertical bars that seem to be funnels, which were hardly visible in the original image [Fig. 13(b)], appear clear in the enhanced image [Fig. 13(c)].

4 Conclusions

We have developed a fully automatic method that produces an enhanced image of an object from a very noisy sequence, where the object is subject to a smooth translational motion. This is achieved by the well-known technique of averaging several frames to reduce the random



(a)

(b)

(c)

Fig. 13 Results for a real sequence of a ship taken from an infrared imaging system in the range of 8 to 12 μm: (a) plot of the horizontal (dashed line) and vertical (dotted line) velocities of the object at each frame; (b) subregion of original middle frame; (c) final enhanced image.

noise, but since the object is moving, it is necessary to estimate and compensate motion before averaging. This task is not trivial, even when the object is simply translating, due to the high level of noise, which makes the application of conventional techniques to estimate the displacements impractical.

An optimal solution is the simultaneous estimation and segmentation of the image velocity, using algorithms like expectation/maximization (E/M).[11] This solution is applicable to general models of image sequences including translation, rotation, and scaling. Indeed, these methods have proven very useful for noise-free images. However, they are usually very costly, they have critical parameters that have to be finely tuned, the convergence is usually slow, and the final result depends critically on the initial condition. Given the large amount of noise present in our sequences, we have taken advantage of the simple translation model of the sequences to apply a more efficient and robust technique. We segment the pixels according to the local spatiotemporal spectrum as a reliable description of the local texture-motion content, which is possible due to the simple translational model of sequences of interest. The clustering is then performed using a simple clustering algorithm (k-means). The results obtained in segmentation of synthetic test sequences, with different amounts of noise and different translation velocities of the object and the background, confirm the robustness of this strategy (mean percentage of correctly segmented pixels greater than 98%).

The velocity of the target can then be estimated robustly, since the segmentation process has grouped together all the pixels sharing the same motion. Other segmentation strategies not based on motion (e.g., based on the intensity level) will produce poor results due to noise, and can group together pixels having different motion, thus spoiling the robust estimation of the velocity. Additional robustness is achieved by estimating the optical flow at the output of a directional bandpass filter, in the spatiotemporal frequency domain, tuned to the moving object, which minimizes the effect of noise as well as the potential contamination by responses due to the background at the occlusion boundaries.

The velocity estimates are then used to register the object along all the frames. Once the target is registered, all the frames are averaged to reduce the noise, producing an enhanced image of the object. The excellent results obtained in real sequences from maritime surveillance systems (both visible and infrared) demonstrate the validity of the approach and the usefulness of the method.

Acknowledgment

This research has been supported by the Comisión Interministerial de Ciencia y Tecnología of Spain, grant TIC98-0925-C02-01. We wish to thank David Fleet (Xerox PARC) for revising the final version of the manuscript.

References

1. B. Cohen, V. Avrin, M. Belitsky, and I. Dinstein, ''Generation of a restored image from a video sequence recorded under turbulence effects,'' *Opt. Eng.* **36**(12), 3312–3317 (1997).
2. W. K. Pratt, *Digital Image Processing*, Wiley-Interscience, New York (1991).
3. J. C. Russ, *The Image Processing Handbook*, CRC Press, Boca Raton, FL (1995).
4. R. C. Hardie, K. J. Barnard, J. G. Bognar, E. E. Armstrong, and E. A. Watson, ''High-resolution image reconstruction from a sequence of rotated and translated frames and its application to an infrared imaging system,'' *Opt. Eng.* **37**(1), 247–260 (1998).
5. M. S. Alam, J. G. Bognar, S. Cain, and B. Yasuda, ''Fast registration and reconstruction of aliased frames by use of a modified maximum-likelihood approach,'' *Appl. Opt.* **37**(8), 1319–1328 (1998).
6. D. Granrath and J. Lersch, ''Fusion of images on affine sampling grids,'' *J. Opt. Soc. Am. A* **15**(4), 791–801 (1998).
7. M. Bertero, T. Poggio, and V. Torre, ''Ill-posed problems in early vision,'' *Proc. IEEE* **76**, 869–889 (1988).
8. J. Barron, D. Fleet, S. Beauchemin, and T. Burkitt, ''Performance of optical flow techniques,'' *Int. J. Comput. Vis.* **12**, 43–77 (1994).
9. Y. Weiss and E. H. Adelson, ''Representing moving images with layers,'' *IEEE Trans. Image Process.* **3**(5), 625–638 (1994).
10. J. Shi and J. Malik, ''Motion segmentation and tracking using normalized cuts,'' *Proc. Int. Conf. on Computer Vision*, pp. 1154–1160 (1998).
11. Y. Weiss and E. H. Adelson, ''A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models,'' in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 321–326 (1996).
12. M. Shizawa and K. Mase, ''Simultaneous multiple optical flow estimation,'' in *Proc. IEEE Int. Conf. on Image Processing*, pp. 274–278 (1990).
13. O. Nestares and R. Navarro, ''Probabilistic multichannel optical flow analysis, based on a multipurpose visual representation of image sequences,'' *Proc. SPIE* **3644**, 429–440 (1999).
14. O. Nestares, C. Miravet, J. Santamaria, and R. Navarro, ''Automatic segmentation of low visibility moving objects through energy analysis of the local 3D spectrum,'' *Proc. SPIE* **3642**, 13–22 (1999).

15. E. H. Adelson and J. R. Bergen, ''Spatiotemporal energy models for the perception of motion,'' *J. Opt. Soc. Am. A* **2**(2), 284–299 (1985).
16. H. Knutsson and G. H. Granlund, ''Texture analysis using two-dimensional quadrature filters,'' presented at IEEE Workshop on Computer Architecture for Pattern Analysis and Image Data Base Management, pp. 206–213 (1983).
17. M. Unser, ''Multichannel static texture segmentation,'' *Signal Process.* **11**, 61–79 (1986).
18. A. K. Jain and F. Farrokhnia, ''Unsupervised texture segmentation using Gabor filters,'' *Pattern Recogn.* **24**(12), 1167–1186 (1991).
19. O. Nestares, R. Navarro, J. Portilla, and A. Tabernero, ''Automatic computation of the area irradiated by ultrashort laser pulses in Sb materials through texture segmentation of TEM images,'' *Ultramicroscopy* **66**(1–2), 101–115 (1996).
20. B. K. Horn and B. G. Schunk, ''Determining optical flow,'' *Artif. Intel.* **17**, 185–203 (1981).
21. R. A. Young and R. M. Lesperance, ''A physiological model of motion analysis for machine vision,'' *Proc. SPIE* **1913**, 48–123 (1993).
22. E. P. Simoncelli and D. J. Heeger, ''A model of neuronal responses in visual area MT,'' *Vision Res.* **38**, 743–761 (1998).
23. C. H. Edwards, *Advanced Calculus of Several Variables*, Dover Publications, New York (1973).
24. W. T. Freeman and E. H. Adelson, ''The design and use of steerable filters,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(9), 891–906 (1991).
25. A. B. Watson and A. J. Ahumada, ''Model of human visual-motion sensing,'' *J. Opt. Soc. Am. A* **2**(2), 322–342 (1985).
26. M. Unser and M. Eden, ''Nonlinear operators for improving texture segmentation based on features extracted by spatial filtering,'' *IEEE Trans. Syst. Man Cybern.* **20**, 804–815 (1990).
27. T. Aach, A. Kaup, and R. Mester, ''On texture analysis: local energy transforms versus quadrature filters,'' *Signal Process.* **45**(2), 173–181 (1995).
28. H. Nyblack, *Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ (1986).
29. E. P. Simoncelli, E. H. Adelson, and D. J. Heeger, ''Probability distributions of optical flow,'' in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 310–315 (1991).
30. A. Singh, ''Incremental estimation of image-flow using a Kalman filter,'' *J. Visual Commun. Image Represent* **3**, 39–57 (1992).
31. M. R. Luettgen, W. C. Karl, and A. S. Willsky, ''Efficient multiscale regularization with applications to the computation of optical flow,'' *IEEE Trans. Image Process.* **3**(1), 41–64 (1994).

Oscar Nestares graduated in electrical engineering in 1994 from Universidad Politécnica de Madrid (Spain). He worked on image and sequence processing in the Image and Vision Group of the Instituto de Optica, Consejo Superior de Investigaciones Científicas from 1992 to 1997, when he received his PhD degree from Universidad Politécnica de Madrid. During 1998 he worked as an R&D engineer in the Electro-optics Section of the Aerospace Division of SENER Ingeniería y Sistemas S.A. He is currently a postdoctoral student (under the Fulbright Program) in the Department of Psychology of Stanford University. He is interested in human and computer vision, computer models of the human visual system, and image-sequence processing.

Carlos Miravet received his MS in physics from Madrid Complutense University in 1986. Since then, he has held posts in national and international industrial and research and development institutions related to the fields of electro-optics and image processing. Currently he is a senior engineer in the Electro-optics Section of the Aerospace Division of SENER Ingeniería y Sistemas S.A.

Javier Santamaría received his MS and PhD degrees in physics from the University of Zaragoza, Spain, in 1969 and 1973, respectively. In 1972 he joined the Instituto de Optica, Madrid, where he worked as a research scientist in the fields of image evaluation, image processing, and vision. He was head of the Imaging and Vision Department until 1988, when he began working at SENER Aerospace Division, where he is responsible for the Electro-optics Group. He has been regularly publishing scientific papers and presenting communications to international meetings. He was president of the Spanish Optical Society (1990–1992) and a member of the Advisory Committee of the European Optical Society (1993–1996). His current research interests include electro-optical system performance, automatic target recognition, and image enhancement and restoration.

Rafael Navarro received the MS and PhD degrees in physics from the University of Zaragoza, Spain, in 1979 and 1984 respectively. From 1985 to 1986 he was an optical and image-processing engineer at the Instituto de Astrofísica de Canarias. Since 1987 he has been a senior scientific researcher at the Instituto de Optica, Consejo Superior de Investigaciones Científicas, where he has headed the Imaging & Vision Group since 1988, and currently is associate director. He has been visiting researcher at the University of Rochester and at U.C. Berkeley. He is member of the OSA, EOS, IEEE Signal Processing, and ARVO, and his research interests are physiological optics, vision, and image processing.