A Robust hierarchical motion estimation algorithm in noisy image sequences in the bispectrum domain

El Mehdi Ismaili Alaoui · Elhassane Ibn-Elhaj

Received: 20 June 2007 / Revised: 16 September 2008 / Accepted: 22 September 2008 / Published online: 21 October 2008 © Springer-Verlag London Limited 2008

Abstract The present study describes a robust hierarchical motion estimation algorithm in noisy image sequences using the bispectrum. The motion can be characterized by an affine model and the parameters of an affine motion model are estimated by means third-order auto-bispectrum and cross-bispectrum measures. The basic components of this framework to obtain motion vectors are (i) pyramid construction, (ii) motion estimation and (iii) coarse-to-fine refinement. The entire motion is decomposed as a global and a local motion field, which helps accurately obtain high resolution estimates for the local motion field. Simulation results are presented and compared to those obtained from the phase correlation algorithm. The results demonstrate that the proposed method is more suited than the phase correlation algorithm to analyse complex noisy image sequences. On the other hand, our method produces smoother displacement vectors field with a more accurate measure of object motion in different signal-to-noise ratio scenarios.

Keywords Hierarchical motion estimation · Affine model · HOS · Bispectrum · Phase correlation · Noisy image sequences · Correlated Gaussian noise

1 Introduction

In order to build a video coder that is robust in the presence of noise, the motion estimation process must be able to track objects within a noisy source. In a noisy source, objects appear to change from frame to frame because of the noise, not necessarily as the result of object motion [1]. Noise gets added to video in the process of recording it. This problem is even more acute when converting from video on analog tapes to video in digital format. Noise is undesirable not only because it degrades the visual quality of the video but also because it degrades the performance of subsequent processing such as compression [2].

The sources of noise that can corrupt an image sequence are numerous. Examples of the more prevalent ones include, the noise introduced by the camera, shot noise that originates in the electronic hardware and the thermal or channel noise [3]. Most noise sources are well modeled by additive correlated Gaussian noise (CGN) model.

A number of different motion estimation algorithms have been proposed in the literature. Detailed reviews are given by [4–10]. They can be divided into four main groups: gradient techniques [7, 9, 11, 12], pel-recursive techniques [12, 13], block matching techniques [12, 14], and frequency-domain technique [15–19]. Originally, these algorithms have been developed for very different applications such as image sequence analysis, machine vision, robotics, image sequence restoration or image sequence coding. The vast majority of these algorithms consider noise-free data, although in [20] the displacement vector is estimated from noisy data using the generalized maximum likelihood criterion. If the image frames are severely corrupted by additive CGN of unknown covariance, second-order statistics methods do not work well. In this circumstance, higher order statistics (HOS) may offer more advantages since they are not affected by such noise [21].
HOS-based methods have already been proposed to estimate motion between image frames [21–26]. In [24] the displacement vector is obtained by maximizing a third-order statistics criterion. In [26] the global motion parameters obtained by a new region recursive algorithm. In [22, 23] several algorithms are developed based on a parametric cumulant method, a cumulant-matching method and a mean kurtosis error criterion. The latter is an extension of the quadratic pixel-recursive method by Netravali and Robbins [27].

In this paper, a novel algorithm for the detection of motion vectors in video sequences is proposed. The algorithm uses the bispectrum to obtain a measure of content similarity for temporally adjacent frames and responds very well to scene motion vectors. The algorithm is insensitive to the presence of symmetrically distributed noise. As the proposed scheme is implemented in the frequency domain. Our motivation for using the bispectrum comes from the following facts:

- In any coherent image, a particular pixel is correlated to its neighbouring pixels in both the row and the column directions. Treatment of an image as 2-D data ensures that both the inter-row and the inter-column correlations present in the image are utilized. Slicing of the image into rows leads to loss of the inter-row correlations and an inadequate model is generated based on partial information [28].

- HOS are translation invariant because linear phase terms are canceled in the products of Fourier coefficients that define them. Functions that can serve as features for pattern recognition can be defined from higher order spectra that satisfy other desirable invariance properties such as scaling, amplification, and rotation invariance [29, 30].

- HOS retain both amplitude and phase information from the Fourier transform of a signal, unlike the power spectrum. The phase of the Fourier transform contains important shape information [31].

- The bispectrum of a random zero-mean Gaussian field is identically zero, a fact which may be exploited to reduce additive noise effects [29].

- The bispectrum techniques are also characterized by their insensitivity to correlated and frequency dependent noise which render them robust in the estimation space [31, 32].

The present study proposes a method that estimates the motion by first computing a global motion field and then uses the global motion field as an initialization in estimating the local motion. The global motion field and local motion are estimated using the bispectrum to accurately identify possible vectors.

The organization of the paper is as follows. In the next section, we describe the bispectrum based image motion estimation of the translation model and the affine motion model.

This is followed by the details of the hierarchical motion estimation. The computational cost about our method is detailed in section 4. Experimental results are then given in section 5 for different sequences. Finally, section 6 draws the conclusion.

2 Bispectrum based image motion estimation

2.1 Translation model

The problem of motion estimation can be stated as follows: “Given an image sequence, compute a representation of the motion field that best aligns pixels in one frame of the sequence with those in the next” [21]. This is formulated as:

\[ g_{k-1}(m) = f_{k-1}(m) + n_{k-1}(m) \]  \hspace{1cm} (1)

\[ g_{k}(m) = f_{k}(m) + n_{k}(m) = f_{k-1}(m - d_{k}) + n_{k}(m) \]  \hspace{1cm} (2)

\[ g_{k}(m) = g_{k-1}(m - d_{k}) + w_{k}(m) \]  \hspace{1cm} (3)

where \( w_{k}(m) = n_{k}(m) - n_{k-1}(m - d_{k}) \); \( m = (x, y) \) denotes spatial image position of a point; \( g_{k}(m) \) and \( g_{k-1}(m) \) are observed image intensities at instant \( k \) and \( k - 1 \) respectively; \( f_{k}(m) \) and \( f_{k-1}(m) \) are noise-free frames; \( n_{k}(m) \) and \( n_{k-1}(m) \) are assumed to be spatially and temporally stationary, zero-mean image Gaussian noise sequences with unknown covariance; and \( d_{k} = (d_{kx}, d_{ky}) \) is the displacement vector of the object during the time interval \([k - 1, k]\).

The third-order auto-cumulant, \( C_{3}^{g_{k}g_{k}g_{k}}(r, s) \), and cross-cumulant, \( C_{3}^{g_{k-1}g_{k}g_{k}}(r, s) \), of a zero-mean 2-D random field \( g_{k}(m) \) are defined respectively as follows [22]:

\[ C_{3}^{g_{k}g_{k}g_{k}}(r, s) = E\{g_{k}(m)g_{k}(m + r)g_{k}(m + s)\} \]  \hspace{1cm} (4)

\[ C_{3}^{g_{k-1}g_{k}g_{k}}(r, s) = E\{g_{k}(m)g_{k-1}(m + r)g_{k}(m + s)\} \]  \hspace{1cm} (5)

where \( r = (r_{1}, r_{2}) \) and \( s = (s_{1}, s_{2}) \) are two shifted versions of the \( g_{k}(m) \); \( E\{\cdot\} \) represents the expectation operator.

To understand the theory of triple correlations physically for 2-D data [28], the reader is referred to Fig. 1. The figure shows the spaces occupied by the original data (denoted by continuous box) and two shifted versions of the same data (denoted by dashed boxes). The shifts are made by the amounts \( (r_{1}, r_{2}) \) and \( (s_{1}, s_{2}) \), respectively. It is now obvious that the product of the overlapping data positions (shown by the shaded portion) denotes the triple correlation function as defined by (Eq.4).

The bispectrum, \( B_{3}^{g_{k}g_{k}g_{k}}(u; v) \), is defined as the Fourier transform of the third-order auto-cumulant \( C_{3}^{g_{k}g_{k}g_{k}}(r, s) \) [28]:

\[ B_{3}^{g_{k}g_{k}g_{k}}(u; v) = \mathcal{F}\{C_{3}^{g_{k}g_{k}g_{k}}(r, s)\} \]  \hspace{1cm} (6)

where \( \mathcal{F} \) denotes the Fourier transform operation. Also, the bispectrum is defined as follows [32]:

\[ B_{3}^{g_{k}g_{k}g_{k}}(u; v) = G_{g_{k}}(u)G_{g_{k}}(v)G_{g_{k}}^{*}(u + v) \]  \hspace{1cm} (7)
where $G_3(u)$ correspond to the discrete Fourier transform (DFT) of the frame $g_3(m)$; * indicates the complex conjugate; $u = (u_1, u_2)$ and $v = (v_1, v_2)$ are the frequency coordinates for the 2-D Fourier transform.

Due to the shift-invariance of the bispectrum

$$B_3^{g_3g_3g_3}(u; v) = B_3^{g_3g_3g_3}(u; v) + B_3^{w_kw_kw_k}(u; v)$$

where $B_3^{g_3g_3g_3}(u; v)$ and $B_3^{w_kw_kw_k}(u; v)$ denote bispectrum of the frame $g_3(m)$ and noise $w_k(m)$, respectively.

This assumes $g_3(m)$ is zero-mean; in practice we must subtract the mean. If we further assume that $n_k(m)$ is CGN, then its triple correlation is identically zero, yielding the important result that theoretically the triple correlation of a signal plus an independent gaussian random noise process with zero-mean, is equal to the triple correlation of the signal [24]; [32]. This provides a theoretical basis for using the triple correlation (or the bispectrum) as a method of reducing the effects of additive noise. Note that the additive Gaussian noise need not be white for its triple correlation to be theoretically zero, as it often is not in practice [29]. If the probability density function of the noise is symmetrical, i.e., $p(n) = p(-n)$, or at least not skewed, i.e., $\int n^2 p(n) dn = 0$, then the term $B_3^{w_kw_kw_k}(u; v)$ is negligible which renders the triple-correlation very effective in detecting a signal embedded in noise. Therefore,

$$B_3^{g_3g_3g_3}(u; v) = B_3^{g_3g_3g_3}(u; v)$$

The cross-bispectrum is defined as follows [28, 32]:

$$B_3^{g_3g_3g_3}(u; v) = \mathcal{F} \{ g_3^{*,*}g_3^{*,*} (r; s) \}$$

$$= G_{g_3}(u)G_{g_3-1}(v)G_{g_3}^*(u+v)$$

Then,

$$B_3^{g_3g_3g_3}(u; v) = B_3^{g_3g_3g_3}(u; v)e^{-j2\pi v_dk}$$

where $G_{g_3-1}(v)$ corresponds to the DFT of the frame $g_{3-1}(m)$.

Using the relation (Eq. 8) we can transform (Eq. 10) as:

$$B_3^{g_3g_3g_3}(u; v) = B_3^{g_3g_3g_3}(u; v)e^{-j2\pi v_dk}$$

The third-order hologram, $h(m)$, is then defined by:

$$h(m) = \mathcal{F}^{-1} \{ B_3^{g_3g_3g_3}(u; v) \}$$

where $\mathcal{F}^{-1}$ denotes the inverse Fourier transform.

The application of the inverse Fourier transform in (Eq.13) can be thought of as obtaining the solution to a set of simultaneous equations in order to determine the relative block displacement that would cause such phase shifts. Spectral whitening is essential in providing the desired immunity against global illumination changes. Then,

$$h(m) = \int_{-\infty}^{\infty} du \int_{-\infty}^{\infty} e^{j2\pi vm} e^{-j2\pi v_dk} dv$$

$$h(m) = \int_{-\infty}^{\infty} du \int_{-\infty}^{\infty} e^{j2\pi v(m-d_k)} dv$$

Therefore,

$$h(m) = \delta(m-d_k)$$

(Eq.16) only allows to compute integer motion vectors. In [33], it is shown that half-pixel accuracy motion vectors leads to a very significant improvement when compared to one-pixel accuracy, where as a higher precision results in negligible changes. Therefore, a half-pixel accuracy was chosen in our simulations.

As a result, by finding the location of the pulse in (Eq.16) we are able to tell the displacement, which is the motion vector. Since third-order statistics are used, the method is insensitive (in theory) to both spatially and temporally colored noise which is symmetrically distributed (e.g., Gaussian). In practice, the motion vector is not an impulse; hence, we estimate $d_k$ as the index $m$, which maximizes $|h(m)|$.

2.2 Affine motion model

The other notable advantage of bispectrum is that the affine transformations (i.e. shifting, rotation, scaling and shearing), which are coupled in the spatial domain, are separated from the translation components into the magnitude and the phase spectrum respectively in the Fourier domain. This is evident from the affine Fourier theorem as proposed by [34], which provides the generalization to the Fourier shift theorem under an affine transformation.

The noise-free signals are assumed to be zero-mean non-Gaussian random fields that are statistically independent of the noise. The basic assumption centers around intensity constancy, as follows:

$$f_k(m) = f_{k-1}(m-d_k)$$

Consider two continuous image frames $f(m)$ and $f_{k-1}(m)$ on the 2-D Euclidean space. Let their corresponding Fourier transformations be given by:

$$f_k(m) \mathcal{FT}\ F_k(u)$$

and $f_{k-1}(m) \mathcal{FT}\ F_{k-1}(u)$.

Let the two image frames be related to each other by an affine transformation given by:

$$f_k(m) = f_{k-1}(Mm - d_k)$$
with $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ being the affine matrix (a, b, c, d for the dilation, rotation and shear components of motion).

Computing the Fourier transform of $f_k(m)$ gives the following:

$$F_{f_k}(u) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f_k(m)e^{-j2\pi um^T} dm$$

(19)

Combining (Eq.18) and (Eq.19) we obtain

$$F_{f_k}(u) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f_{k-1}(Mm - d_k)e^{-j2\pi um^T} dm$$

(20)

Let $m^T = Mm^T - d_k^T$ and $|\Delta| = det(M) = ad - bc$. If $\Delta \neq 0$ we can invert the affine matrix to express $m^T$ as:

$$m^T = \frac{1}{|\Delta|} M'(m^T + d_k^T),$$

(21)

where $M' = \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$.

Using the rule for changing variables in integrals [35] we can transform (Eq. 20) into the coordinate system defined by $m'$:

$$F_{f_k}(u) = \frac{1}{|J(m)|} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f_{k-1}(m')e^{-j\frac{2\pi}{|\Delta|} uM'(m^T + d_k^T)} dm'$$

(22)

where $J(m)$ is known as the Jacobian of the transformation [35] defined as:

$$|J(m)| = \frac{dm'}{dm}$$

(23)

Then,

$$J(m) = \frac{\partial(m')}{\partial(m)} = ad - bc = \Delta.$$ 

(24)

Therefore,

$$F_{f_k}(u) = \frac{1}{|\Delta|} e^{-j\frac{2\pi}{|\Delta|} uM'd_k^T} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f_{k-1}(m')e^{-j\frac{2\pi}{|\Delta|} uM'm^T} dm'$$

(25)

where $\times$ denotes the scalar product.

Consider the exponent of the Fourier transform; we have $u' = uM'$. Expressing $u'$ in terms of $u^T$ we get the following $u^T = M'^{-1}u^T$.

Thus, we obtain the following reduction to the equation derived above (Eq.25):

$$F_{f_k}(u) = \frac{1}{|\Delta|} e^{-j\frac{2\pi}{|\Delta|} uM'd_k^T} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f_{k-1}(m')e^{-j\frac{2\pi}{|\Delta|} uM'm^T} dm'$$

(26)

By the definition of the Fourier transformation of a 2-D function we have

$$F(u) = \mathcal{F} \{ f(m) \} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(m)e^{-j2\pi um} dm$$

(27)

Thus we have the following relationship governing the two Fourier transformations:

$$F_{f_k}(u) = \frac{1}{|\Delta|} e^{-j\frac{2\pi}{|\Delta|} uM'd_k^T} F_{f_{k-1}}(u')$$

(28)

Using the relations (Eq. 7 and 27) we can transform (Eq. 12) as:

$$B_{3}^{\text{affine}}(u; v) = \frac{1}{|\Delta|^3} e^{-j2\pi vv_k} B_{3}^{\text{affine}}(u; v')$$

(29)

Therefore,

$$B_{3}^{\text{affine}}(u; v) = \frac{1}{|\Delta|^3} e^{-j2\pi vv_k} B_{3}^{\text{affine}}(uM'; vM')$$

(30)

The magnitude spectra are therefore related by:

$$|B_{3}^{\text{affine}}(u; v)| = \frac{|B_{3}^{\text{affine}}(u; v)|}{{|\Delta|^3}}$$

(31)

The parameters of the motion governed by the affine parameters can be estimated from the magnitude spectrum [36, 37].

Note from (Eq.30) that once the spectra have been aligned by the coordinate transformation defined by $M'$ they are then related by a phase shift proportional to the translation component $d_k$. This the affine transform can be found by correlation using

$$d_k = \max \{ p(m) \}$$

(32)

$$p(m) = \mathcal{F}^{-1} \{ B_{3}^{\text{affine}}(u; v)B_{3}^{\text{affine}}(uM'; vM') \}$$

(33)

with,

$$B_{3}^{\text{affine}}(u; v)B_{3}^{\text{affine}}(uM'; vM') = |B_{3}^{\text{affine}}(u; v)|^2 e^{-j2\pi vv_k} d_k$$

(34)

The affine transform shows separation of the displacement vector and the affine coordinate matrix in the frequency domain. This has an immense potential in motion estimation since this theorem provides a mechanism by which the translation component can be estimated independent of the affine parameters, while regular spatio-temporal methods attempt to estimate the translational parameters under the constrained interference of the affine parameters.

In parametric motion estimation, the motion model for the estimation are obviously closely related. A complex model results in a better description of the motion in a sequence, and thus allows representing efficiently the motion of a larger region of the image. Conversely, a simple model is sufficient to represent the motion of a small region of the image [10, 12].

3 Hierarchical motion estimation

Fig.2 describes the hierarchical motion estimation framework. The basic components of this framework to obtain motion could be similar to the one proposed in [38], that is: (i) pyramid construction, (ii) motion estimation, and (iii) coarse-to-fine refinement.

Arguments for use of hierarchical (i.e. pyramid based) estimation techniques for motion estimation have usually
focused on issues of computational efficiency. A matching process that must accommodate large displacements can be very expensive to compute. Simple intuition suggests that if large displacements can be computed using low resolution image information great savings in computation will be achieved. Higher resolution information can then be used to improve the accuracy of displacement estimation by incrementally estimating small displacements. On the other hand, as it is pointed out in [38] it is not only advantageous to ignore high resolution image information when computing large displacements, in a sense it is necessary to do so. This is because of aliasing of high spatial frequency components undergoing large motion. Aliasing is the source of false matches in correspondence solutions or (equivalently) local minima in the objective function used for minimization. Minimization or matching in a multi-resolution framework helps to eliminate problems of this type. Another way of expressing this is to say that many sources of non-convexity that complicate the matching process are not stable with respect to scale [39].

This section describes a hierarchical estimation framework for the computation of diverse representations of motion information. The key features of the resulting framework are the pyramid construction, motion estimation (global and local), and a coarse-to-fine refinement strategy. An example of a global motion estimation is the rigidity constraint; an example of a local motion estimation is that displacement is constant over a patch. Coarse-to-fine refinement or hierarchical estimation is included in this framework for reasons that go well beyond the conventional ones of computational efficiency.

3.1 Pyramid construction

A multi-resolution pyramid is depicted in Fig.2. We used a 3-level pyramid, the image pyramid is computed for each level of the image using a Gaussian filter followed by a 1:2 signal down-sampling. This helps reduce the aliasing artifacts that otherwise arise due to the down-sampling. Where the full resolution frame \( g_k^0(m) \) is at level 0. Let \( g_k^i(m) \) denote the frame in level \( i \) at time \( k \), and \( N \) be the number of decomposition levels. The coarser resolution frame \( g_k^i(m) \) is obtained by passing one finer resolution frame \( g_k^{i-1}(m) \), as is shown in Fig.3.

3.2 Motion estimation

3.2.1 Global motion estimation

The global motion field is estimated by performing the bispectrum at each level of the image pyramid hierarchy. Starting from the coarsest resolution level \( N \), we divide \( g_k^N(m) \) into non-overlapping blocks. In order to improve both accuracy and efficiency, we employ the bispectrum algorithm scheme to measure the block motion vectors in the coarsest resolution level and hierarchically refine the block motions in the finer resolution level. At the finest resolution of the image hierarchy, the global motion field is obtained this is shown in Fig.3.

3.2.2 Local motion estimation

The global motion field obtained from the previous stage provides the initial estimate for the local motion estimation. The local motion field is obtained in a similar manner as the global motion estimation in that the bispectrum is performed on a tessellation of blocks over the entire image. Thus, for our simulation for the local estimation, the block size was \( 16 \times 16 \) pixels. The block is windowed with a Hamming function prior to taking the fast Fourier transform (FFT) in order to avoid boundary discontinuities due to its periodic nature.
In order to correctly estimate the cross correlation of corresponding blocks in respective frames, we extend the blocks to 32 × 32 in size, centered around the formerly defined 16 × 16 blocks, to calculate the bispectrum. If we do this only for 16 × 16 blocks, their correlation might be very low for particular motion due to the small overlapping area, as shown in Fig. 4(a). Once the block size is extended to 32 × 32, the overlapping area is increased for better correlation estimation, as is shown in Fig. 4(b). Obviously there will be overlap among the extended blocks and a pixel usually exists in multiple object blocks.

The final component, coarse-to-fine refinement, propagates the current motion estimates from one level to the next level where they are then used as initial estimates.

4 Computational cost

One of the major contributions to the field of image processing was the discovery [40, 41] of an efficient computational algorithm for the DFT. The computation of the DFT of a 1-D sequence of \( N \) values requires on the order of \( O(N^2) \) complex multiply and add operations. Implementing the DFT using a FFT algorithm reduces this computational requirement to the order of \( O(N\log_2 N) \) operations. For large images the computational savings are substantial. The original FFT algorithms were limited to images whose dimensions are a power of 2.

The majority of the computational cost of the proposed bispectrum is due to the FFTs. The fundamental computation required for bispectral estimates is given by (Eq. 7), the triple product of the three individual Fourier transformations, while this computation is straightforward, limitations on computer time and statistical variance impose severe limitations on implementation of the definition of the bispectrum [42]. On the other hand, we take advantage of the symmetrical properties of the bispectrum to reduce the computational complexity and memory requirements of calculating third order statistics. It can now be calculated in any one sector and mapped onto the others [43].

In this Section, the bispectrum technique is compared with the phase correlation one in terms of the number of FFTs required. Thus, our method requires six forward FFTs (three computations of \( B_{1}^{16,16,16} \) and three of \( B_{1}^{16,16,16,16} \) as demonstrate in (Eq.11 and 30), related to the translation model and affine motion one respectively. On the other hand, our algorithm requires for translation model one backward FFT (computation of \( b(m) \)) and one backward FFT (computation of \( p(m) \)) for affine motion model. On the whole, for both models the FFT computation of our algorithm requires about \( O(7N\log_2 N) \) operations. On the other hand, the phase correlation algorithm requires two forward and one backward FFTs [37]. Thus, the FFT computation of the phase correlation algorithm requires about \( O(3N\log_2 N) \) operations for both models. Overall, the number of operations required to compute the bispectrum is significantly increased relative to the phase correlation. Note that for a fair comparison we used our technique and phase correlation one with an affine motion model.

5 Experimental results

Simulations were performed on the luminance component of the three sequences Table Tennis, Mobile & Calendar and Vectra. The size of the Table Tennis sequence is 352 × 240 pixels/frame, the size of the Mobile & Calendar and Vectra sequences are 352 × 288 pixels/frame. Both the Table Tennis and Mobile & Calendar sequences were run for 60 frames with a frame rate of 30 frames/sec, but the Vectra sequence run for 35 frames. These sequences were chosen for their difficult motion and their different characteristics. In particular, in the Mobile & Calendar sequence an object moves across a differently moving background, which produces a DVF with several motion boundaries, while Table Tennis sequence contains large displacement and zooming.

To assess the performances of the proposed motion estimation technique we compared it to a phase correlation technique implemented in a similar manner as our approach. The following comparisons were made. First, the subjective quality of the estimated motion field was evaluated, showing the capability of the algorithm to estimate the true motion in the scene. Second, the peak signal-to-noise ratio (PSNR) of the motion compensated displacement field difference (DFD) was measured, giving insight about the quality of the prediction and tells us how much residual energy is in the DFD. The PSNR and DFD are defined as follows:

\[
PSNR = 10 \log_{10} \frac{255^2}{MSE}
\]

\[
DFD = f_k(m) - f_{k-1}(Mm - d_k)
\]

where MSE is mean squared error.

In this section we examine a few examples and compare the performance, efficiency and complexity of the two methods. Fig. 5 shows the values of PSNR for the phase correlation and the bispectrum methods. All image sequences are
degraded with additive zero mean Gaussian noise to a SNR of 10 dB. Here we define

\[ \text{SNR} = 10 \log_{10} \frac{\sigma_f^2}{\sigma_n^2} \]  

(37)

where \( \sigma_f^2 \) is the variance of the frame, \( \sigma_n^2 \) is the variance of the noise. From Fig.5, it is clear that the implemented phase correlation technique is significantly less efficient than the bispectrum technique. It is mainly due to the difficulty of the phase correlation algorithm to cope with large displacement, zooming, rotation and discontinuities in the motion field. But the bispectrum is shift-invariant because linear phase terms are canceled in the products of Fourier coefficients that define them. Functions that can serve as features for pattern recognition can be defined from higher order spectra that satisfy other desirable invariance properties such as scaling, amplification, and rotation invariance.

The ability of the bispectrum algorithm to accurately estimate the DVF from a degraded sequence is demonstrated in Figs.6 and 7. In these figures, the results obtained for the bispectrum in noisy sources can be analyzed with respect to the DVF they produce. In the case of moderate noise with an SNR of 30 dB, the DVF obtained from the bispectrum method is smoother than the DVF for the phase correlation method, as illustrated in Figs.6 and 7. The motion field with an input SNR of 30 dB is shown in Figs.6(a) and 7(a) for the bispectrum and in Figs.6(b) and 7(b) for the phase correlation, when applied to frames 70 and 71 of the Table Tennis sequence, and frames 10 and 11 of the Mobile & Calendar sequence.

In particular it is very interesting to observe that the bispectrum technique outperforms the phase correlation one in the portion of the sequence which contains a zoom. Fig.8 shows motion estimates from two frames of the noisy Table Tennis sequence. This example shows clearly the ability of the bispectrum scheme to provide a concise description of the motion field in terms of relatively few local affine estimates. From Fig.8, it is clear that the motion fields estimated by the bispectrum technique tend to be very smooth due to the smoothness constraint. In contrast, the ones obtained by the phase correlation technique exhibit a few wrong motion vectors in the sense of the motion in the scene. On the whole, the bispectrum technique is able to estimate closely the true zooming motion. In contrast, the phase correlation technique is unable to do so. It can be concluded that the bispectrum motion estimation results globally in motion fields more representative of the true motion in the scene.

Motion estimation is based on the response of the algorithm to content similarities between two frames, as measured by the height of the dominant peaks of the correlation surface. The detection of motion vectors relies on successive phase correlation operations applied to pairs of consecutive block partitioned frames of a video sequence. The heights of the dominant peaks are monitored, and when a sudden magnitude change is detected, then this is interpreted as a displacement vector. Fig.9 shows sample phase correlation surfaces between two blocks \( b_{k-1}(m) \) and \( b_k(m) \), related to frames 17 and 18 of the Mobile & Calendar sequence respectively. The phase correlation surfaces between \( b_{k-1}(m) \) and \( b_k(m) \) with an input SNR of 35 dB is shown in Fig. 9(a) for the bispectrum and in Fig. 9(b) for the phase correlation method. The bispectrum retains both amplitude and phase information from the Fourier transform of a signal, unlike the power spectrum. The phase of the Fourier transform contains important shape information. Therefore, the bispectrum minimizes the influence of the noise and simplifies the identification of the dominant peak on the correlation surface.

Comparisons of the phase correlation and the bispectrum methods indicate the bispectrum is a robust technique for motion estimation. Results of these comparisons are shown...
Fig. 6 Motion estimation for frame 70 and 71 of noisy Table Tennis sequence (SNR=30 dB).

Fig. 7 Motion estimation for frame 10 and 11 of noisy Mobile & Calendar sequence (SNR=30 dB).

Fig. 8 Portion of zooming motion for two frames from the noisy Table Tennis sequence (SNR=30 dB).

Fig. 9 Phase correlation surfaces between two blocks.
for different noise levels and video sequences. CGN was added to image sequences with an input SNR varying from 10 dB to 50 dB.

In Fig.10, the average PSNR (across all input frames) is plotted against input noise level. The average PSNR, $\text{PSNR}_{\text{avg}}$, is given as:

$$\text{PSNR}_{\text{avg}} = \frac{1}{F} \sum_{i=1}^{F} \text{PSNR}_i$$  \hspace{1cm} (38)

where $\text{PSNR}_i$ is the measured PSNR for frame $i$ and $F$ is the total number of frames.

Under normal operating conditions, e.g., input SNR between 30 to 50 dB, the performance of the bispectrum is as much as 1.47 dB better than the performance of the phase correlation algorithm for the Table Tennis sequence, as seen in Figs.10(a) and 11(a).

In Fig. 10(b), the results for the Mobile & Calendar sequence are plotted. In this figure we observe that the bispectrum method produces higher PSNR of more than 2.91 dB if the SNR between (25 and 50 dB), as seen in Figs.10(b) and 11(b).

In Fig.10(c), the results for the Vectra sequence are plotted. In Vectra example, the bispectrum is always better than the phase correlation technique for every noise condition. If SNR between (30 and 50 dB) the improvement in the bispectrum performance has been as much as 1.04 dB compared to the performance of the phase correlation algorithm for this sequence, as seen in Figs.10(c) and 11(c).

At low noise levels (input SNR from 35 to 50 dB) the performance is comparable to its performance in a noise-free environment. For high-noise environments, little can be done to improve motion estimation efficiency. Fig.10 show the phase correlation method fail at high noise levels(10 to 20 dB); however, even at these high levels, the bispectrum yields a performance gain of 0.29 to 0.14 dB over the phase correlation algorithm.

In Fig.11, the performance of the bispectrum is shown at a fixed input noise level of 35 dB across all input frames of the Table Tennis, Mobile & Calendar and Vectra sequences. In these examples, the bispectrum outperforms the phase correlation algorithm at this moderate noise level.

In terms of prediction frame quality, from Vectra and Mobile & Calendar sequences, we observe better prediction by bispectrum technique, we also observe that the bispectrum produces higher PSNR. The bispectrum algorithm is able to measure the motion vector more accurately and is more robust in general. Examples of prediction are shown in Figs.12 and 13. In the case of moderate noise with an SNR of 30 dB, the prediction obtained from the bispectrum method is smoother than the prediction for the phase correlation method, as illustrated in Figs.12 and 13. The prediction with an input SNR of 30 dB is shown in Figs.12(a) and 13(a) for the bispectrum and in Figs.12(b) and 13(b) for the phase correlation. Overall, the bispectrum typically offers better visual quality images than the phase correlation technique.

In terms of complexity, this is measured by the computation time. All the computations are performed on Intel Pentium IV machines with Windows XP. The two algorithms have been implemented using a prototype written in Matlab 6.5 R13. The comparison between the proposed method and the phase correlation confirmed that the two methods have the complexity on the same order. This is shown in Table.1.

6 Conclusion

In case images are corrupted by CGN the estimation of the parameters of an affine model may be difficult to obtain. In this work, we propose the use of a the bispectrum based method to obtain the motion parameters. The bispectrum
provides an advantage over the phase correlation algorithm in the presence of CGN. With our method, the DVF is smoother, providing a more accurate measure of object motion. This characteristic of the bispectrum provides performance gains over the phase correlation algorithm with CGN. At relatively
References


33. G.Madec,:Half pixel accuracy in blockmatching. in Proceedings on the Picture Coding Symposium (PCS 90), Cambridge, Mass, USA, March (1990)


