An efficient fast full search block matching algorithm using FFT algorithms

Fedwa Essannouni[1,2]*, Rachid Oulad Haj Thami*[3]*, Ahmed Salam*[1] *and Driss Aboutajdine*[2]

[1]Laboratoire de Mathématiques Pures et Appliquées, Université du Littoral-Côte d'Opale. C.U. de la Mi-Voix, 50 rue F. Buisson, B.P. 699, 62228 Calais, Cedex, France.

[2]GSCM, UFR IT Faculté des Sciences, Université Mohamed V-Agdal, B.P. 1014 Rabat, Maroc

[3] Laboratoire SI2M, Equipe WiM ENSIAS, Université Mohamed V-Souissi, B.P. 713 Rabat, Maroc

Summary

Motion estimation is the most computationally expensive operation in the coding and transmitting of video streams, and the search for efficient motion estimation (in terms of computational complexity and compression efficiency) algorithm has been a challenging problem for years.

The challenge is to decrease the computational complexity of the full search as much as possible without losing too much performance and quality at the output.

In this paper, we propose a fast algorithm which achieves exactly the same optimal result as the direct full search algorithm. The key idea is to express a robust matching criteria sum square difference (SSD) in terms of cross correlation operations. Speed is obtained from computing the cross correlations in the frequency domain via the Fast Fourier Transform (FFT).

Key words:

Block matching algorithms, optimal solution, fast Fourier transform, phase correlation, correlation techniques.

1. Introduction

Local motion estimation is a key issue in the field of moving images analysis. It has many promising applications such as target detection and tracking, video compression, and template matching. Through the years, extensive research has been conducted in developing new methods and designing cost-effective. Block-based methods are the most popular among the several motion estimation techniques used in practice due to their simplicity allowing efficient hardware implementations. The optimal block based motion estimation algorithm is the Full Search Block Matching algorithm (FS). But a major drawback is the high computational cost of the algorithm, which prevents it from being applied in most real-time systems and has led to a great deal of research to develop more efficient specialized processors and many fast block matching algorithms.

All existing block matching algorithms can be roughly grouped into two categories. The first category consists of those algorithms that are not guaranteed to find the best matching block within a given search range, but instead use a heuristic approach to guide the search. These methods examine only a subset of the possible locations within the search range, and hence can be computed very efficiently. Some of the most popular methods are Three Step Search (TSS) [1], New Three Step Search (NTSS) [2], Simple and Efficient TSS (SES) [3], Four Step Search (4SS) [4], Diamond Search (DS) [5], and Adaptive Rood Pattern Search (ARPS) [6]. These methods reduce the computation by testing only the most promising motion vector candidates, and ignoring the rest. This degrades the estimation result, because these algorithms assume that the error criterion is unimodal, which is rarely true. These algorithms are then called sub-optimal because although they are computationally more efficient than the full search, they are prone to getting trapped in local minima.

In the second class, the fast full search algorithms test all candidate vectors without degrading the result. Many algorithms have been developed for this type of search [7-9]. They achieve their speedup through early elimination of candidate search positions. Most of them compute the (SAD) or (SSD) lower bound for the current vector, compare the best SAD or SSD found so far to the bound, and reject the candidate vector, if the lower bound is greater (worse). However, these techniques suffer from the fact that their performance depends largely on the content of the image sequence being encoded. If the video sequence is noisy, or there is a large amount of motion, these algorithms reject only small part of the candidate motion vectors and require more computation.

In this paper however, we present a frequency algorithm to solve the overwhelming complexity of the full search block matching algorithm using the speed and computational efficiency that typifies frequency domain processing, owing to the use of fast algorithms.

The rest of the paper is structured as follows. In Section 2, the proposed algorithm is detailed. The principles of three state-of-the-art correlation motion estimation algorithms are outlined in Section 3. In Section 4 a comparative performance assessment is carried out with respect to three state-of-the-art correlation motion estimation, and six fast

Manuscript received March 25, 2006.
Manuscript revised March 30 , 2006.

block matching algorithms under SSD metric. Conclusions are drawn in Section 5.

2. The proposed method

2.1 Problem formulation

The problem treated in block based methods is to determine the position of a given block *g* from the current frame in a confined area *f* of the reference frame. Let *g(k,l)* denotes the luminance value of the block *g* of size $m \times n$ at the point *(k,l')* and *f(k,l')* the luminance value of the search area *f* of size $w \times h$ at the point *(k,l')*. The most optimal block based motion estimation algorithm is the Full Search Block Matching Algorithm (FS) which compares the current block *g* to all of the candidate blocks in the search area *f* , and selects the motion vector corresponding to the candidate block, which yields the best criterion function value. Typical criteria used is the sum of squared differences (SSD). The latter can be expressed as follows:

$$SSD(x,y)=\sum_{l=0}^{n-1}\sum_{k=0}^{m-1}(f(k+x,l+y)-g(k,l)) \qquad (1)$$

The motion vector (mv_x,mv_y) is measured from the position of the minimum in (1).

2.1 The proposed algorithm

In this paper, to search the position (mv_x,mv_y) of the bloc *g* in the search area *f*, we propose to minimize a new surface based on the frequency domain.
Let:

$$o(x,y)=\begin{cases}1 & for\ (x,y)\in[0,m-1]\times[0,n-1]\\ 0 & for\ (x,y)\in[m,w-1]\times[n,h-1]\end{cases} \qquad (2)$$

The surface *D(x,y)* is defined as follows:

$$D(x,y)=\Re\left\{IFFT\left(\begin{matrix}(H(u,v)O^*(u,v))\\ (-2(F(u,v)G^*(u,v)))\end{matrix}\right)\right\} \qquad (3)$$

where *H(u,v), O(u,v), F(u,v), G(u,v)* denote respectively the FFT of $f^2(x,y)$, *o(x,y)*, *f(x,y)* and *g(x,y)*. On the other hand, the *IFFT* denotes the Inverse Fast Fourier Transform, *R* denotes the real part of a complex number and the asterisk denotes complex conjugation. Note that *f* and *g* are correlated with FFTs by zero padding the size of *g* to the size of *f* prior to taking the forward FFTs. Since the FFT that we have used is cyclic, the last *m-1* rows and *n-1* columns of result (3) will contain wrap around data that should be discarded. Finally, the resulting motion vector (mv_x,mv_y) of the bloc *g* in the search area *f* is measured **from the position of the minimum in (3)**. This vector is exactly the one which minimizes the SSD metric using the Full search block matching algorithm. Indeed using the correlation Fourier theorem and FFT proporeties. (3) can be written as:

$$\begin{aligned}D(x,y)&=\sum_{l=0}^{n-1}\sum_{k=0}^{m-1}\begin{matrix}f?\,k+x,l+y)o(k,l)\\ -2f(k+x,y+l)g(k,l)\end{matrix}\\ &=SSD(x,y)-\sum_{l=0}^{n-1}\sum_{k=0}^{m-1}g?\,k,l).\end{aligned} \qquad (4)$$

Since the sum of *g²(k,l)* is independent of (x,y), then we can conclude, that the motion vector (mv_x,mv_y) which minimizes (3) is exaclty the one which minimizes the SSD metric.

The idea behind our proposed technique is to express the SSD metric in a simple formula using Fast Fourier transforms(FFTs) proprieties.
Remark that, *O(u,v)* can be calculated and stocked once. The proposed method requires only three forward and one backward FFTs of size $w \times h$. So, it has a running time close to $O((wh)\times\log_2(wh))$.

3. Previous frequency algorithms

The most known and popular frequency motion estimation is the phase correlation method. In this section we review the principle of this method and two other correlation techniques that have recently appeared in the literature.
The phase correlation method [10] capitalizes on the well known Fourier shift theorem which states that shifts in the spatial domain correspond to linear phase changes in the Fourier domain. Its formula between two images and is given in [10] by:

$$P(x,y)=\Re\left\{IFFT\left(\frac{F_2(u,v)F_1^*(u,v)}{|F_2(u,v)F_1^*(u,v)|}\right)\right\}, \qquad (5)$$

Where F_1 and F_2 denote the Fourier Transforms of the two images f_1 and f_2.
For block based motion estimation the phase correlation technique works by computing the phase correlation function (5) between two regions of image data, usually in the form of co-sited rectangular blocks. Between such blocks, the phase correlation map results in several peaks instead of one clear peak. The highest peak corresponds to the best match between a large area, but not necessarily the best match for the object block. It follows that many peaks should be examined using image correlation.
This latter is a matching procedure, which is similar to the classical block matching method, except that image correlation is performed after the displacement vectors are already found by the phase correlation peaks .
However, in block based motion estimation, the phase

correlation method fails in several cases. Indeed the mathematical analysis of this technique assumes ideal translation which is not true between two compared blocks from real frames.

Recently, a new type of correlation technique "Fast Robust Correlation" has been proposed [11]. It works by expressing a robust matching surface as a series of correlations using cosine function. Speed is obtained by computing correlations in the frequency domain. An additional version of this cross correlation approach using angle gradient features has been proposed by the same authors in [12]. The "Orientation correlation" method operates by estimating the motion from orientation images. Each pixel in a orientation image is a complex number that represents the orientation of intensity. Authors have reported enhancements to the existing correlation methods. Despite the improvement, we can note through the experiments presented in [11] and [12], that both of these methods remain sub optimal when compared with the direct full search for block matching algorithm. Therefore, even though the existing correlation methods can present an attractive alternative for solving the overwhelming complexity of the full search block matching algorithm, these methods are not able to guarantee the same level of performance as the full search block matching algorithm.

4. Simulations results

The proposed algorithm (FM) is simulated using the luminance of the popular video sequences Carphone, Bus, and Tempete CIF formats 352×288.

In all of the simulations, SSD block distortion measure, block size 16^2, and search window ± 8 are used.

In Figures 1-3, we compare between our proposed frequency method (FM), the phase correlation method (PC) [10], the orientation correlation method (OC) [12], and the fast robust correlation (FRcorr) as implemented in [11] (video coding section).

On the other hand, in Tables 1-3, the proposed algorithm is compared against six traditional BMAs: full search (FS), DS [5], TSS [1], SS4 [4], NTSS [2], and ARPS [6] by two aspects namely MSE and computational ratio to FS. These two aspects reflect the prediction quality and computational reduction.

From these tables, we can see that the proposed frequency method outperforms the other fast spatial block matching algorithms in terms of prediction quality and computational cost.

On the other hand our proposed method greatly outperforms the other frequency methods in terms of PSNR while still keeping the same order of complexity.

Table 1. Performance of the proposed FM against spatial block matching algorithms using Carphone sequence.

Algorithm	*MSE*	*Time (%)*
FM	25.07	2.02
DS	26.67	6.43
TSS	30.65	9.63
SS4	29.44	7.44
NTSS	26.59	8.25
ARPS	27.41	3.50

Table 2. Performance of the proposed FM against spatial block matching algorithms using Bus sequence.

Algorithm	*MSE*	*Time (%)*
FM	220.30	2.01
DS	458.16	7.91
TSS	360.57	9.18
SS4	462.40	8.05
NTSS	349.74	10.50
ARPS	340.33	3.89

Table 3. Performance of the proposed FM against spatial block matching algorithms using Tempete sequence.

Algorithm	*MSE*	*Time (%)*
FM	119.68	2.81
DS	122.75	5.84
TSS	123.37	10.55
SS4	124.37	7.39
NTSS	122.89	7.79
ARPS	123.30	2.97

5. Conclusion

The FFT block matching algorithm discussed in this paper exploits only the FFT algorithms in its computation of the SSD metric. This method is not heuristic-based and thus can consistently identify the best matching blocks, minimizing MSE. The algorithm is well suited for software implementations. It is independent of image content, and can be even faster by using machine specific optimized FFT implementations which are widely available. On the other hand, the proposed frequency method greatly outperforms the existing frequency motion estimation techniques in terms of accuracy while still keeping the same order of complexity of the fast block matching ARPS [6].

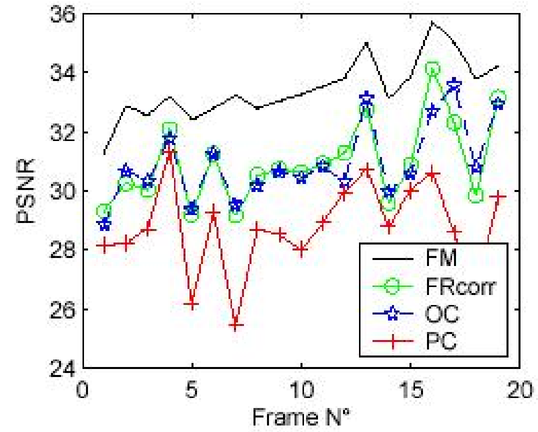

Figure 1. PSNR results for Carphone sequence

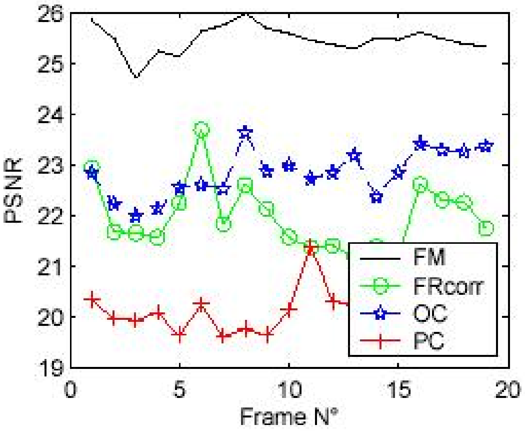

Figure 2. PSNR results for Bus sequence

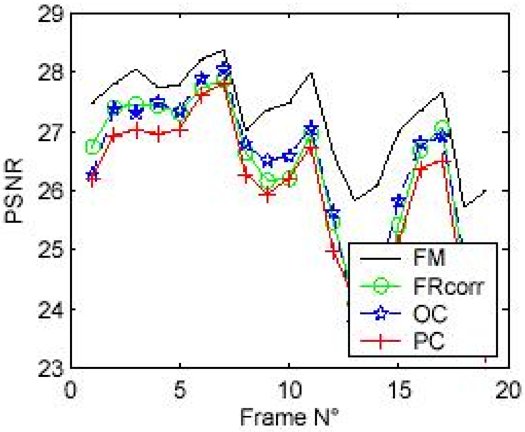

Figure 3. PSNR results for Tempete sequence

References

[1] T. Koga, K. Iinuma, A. Hirano, and Y. Iijima, ``Motion compensated interframe coding for video conferencing," in NTC' 81 Conference Record, 1981, pp. G5.3.1--G5.3.5.

[2] R. Li, B. Zeng, and M. Liou, ``A new three-step search algorithm for block motion estimation," IEEE Trans. On Circuits and Systems for Video Technology, vol. 4, no. 4, pp. 438--442, August 1994.

[3] J. Lu and M. Liou, ``A simple and efficient search algorithm for block-matching motion estimation," IEEE Trans. On Circuits and Systems for Video Technology, vol. 7, no. 2, pp. 429--433, April 1997.

[4] L. Po and W. Ma, ``A novel four-step search algorithm for fast block motion estimation," IEEE Trans. On Circuits and Systems for Video Technology, vol. 6, no. 3, pp. 313--317, June 1996.

[5] S. Zhu and K. Ma, ``A new diamond search algorithm for fast block-matching motion estimation," IEEE Transactions on Image Processing, vol. 9, no. 2, pp. 287--290, February 2000.

[6] Y. Nie and K. Ma, ``Adaptive rood pattern search for fast block-matching motion estimation," IEEE Transactions on Image Processing, vol. 11, no. 12, pp. 1442--1449, December 2002.

[7] W. Li and E. Salari, “Successive elimination algorithm for motion estimation,” *IEEE Transactions on Image Processing*, vol. 4, no. 1, pp. 105–107, January 1995.

[8] Y.C. Lin and S.C. Tai, “Fast full-search block matching algorithm for motion-compensated video compression,” *IEEE Transactions on communications*, vol. 45, no. 5, pp. 527–531, May 1997.

[9] Y.S. Chen, Y.P. Hung, and C.S. Fuh, “Fast block matching algorithm based on the winner-update strategy,” *IEEE Transactions on Image Processing*, vol. 10, no. 8, pp. 1212–1222, August 2001.

[10] C. Kuglin and D. Hines, ``The phase correlation image alignment method," in IEEE 1975 International Conference on Systems, Man and Cybernetics, September 1975, pp. 163--165.

[11] A. Fitch, A. Kadyrov, W. Christmas, and J. Kittler, ``Fast robust correlation," IEEE Transactions on Image Processing, vol. 14, no. 8, pp. 1063--1073, August 2005.

[12] A. Fitch, A. Kadyrov, W. Christmas, and J. Kittler, ``Orientation correlation," in BMVC02, 2002, p. Matching/Recognition.