A Fast Algorithm for Motion Estimation under the Varying Inter-Frame Brightness Characteristics*

Farzin Ahmadianpour and M. Omair Ahmad, Fellow, IEEE

Department of Electrical and Computer Engineering
Concordia University
1455 De Maisonneuve Blvd. W.,
Montreal, Quebec, Canada H3G 1M8

Abstract

A fast two-stage scheme for the search of the motion vectors under varying inter-frame brightness characteristics is devised. In the first stage of the scheme, a given block and the corresponding blocks in the search window are mapped into the sum-of-pixel value domain, where two subsets of candidate blocks, one consisting the blocks having the DC values closest to the DC value of the block of interest and the other consisting of those having the farthest DC values are selected. In the second stage, the motion vector is determined employing these subsets and using the mean-square error as the matching criterion. Experimental results show that the proposed technique provides a high prediction accuracy with a low computational load.

Index Terms: *Fast motion estimation, enhanced motion vectors, brightness variations, two-stage search for motion vectors.*

I. Introduction

The conventional full search block-based algorithm, even though computationally intensive, is still considered to be the best in terms of the prediction performance [2]. In recent years, many fast algorithms have been developed [2-6] to reduce the amount of computation involved with the process of motion estimation. These algorithms sacrifice the prediction accuracy in order to reduce the computational complexity.

Some of the previously proposed fast block-based methods use the notion that “very good” matches are likely to be found in the vicinity of reasonably good matches. There are a large number of algorithms that make this assumption and they may be classified as algorithms based on the principal of locality [3, 4]. One of the problems with these algorithms is that they can converge to a local rather than a global minimum [3].

There is another class of the algorithms that seek to exploit the natural spatial dependencies that exist in most images. In these algorithms, the motion vector for a block can be predicted based on the motion vectors of the blocks surrounding it [2].

There are algorithms that exploit the limitations of the human observations. These algorithms reduce the computational complexity by eliminating a number of block candidates based on the knowledge that the human eyes cannot perceive fast motion with full resolution. For instance, all the blocks around the center of the search area are considered as potential matches, while far from the center only a subset of the blocks is considered for the evaluation of the motion vectors [7].

---

[*] This work was supported by the Natural Sciences and Engineering Research Council of Canada

Some algorithms reduce the computational complexity by using several stages. These algorithms use a different comparison criterion in each stage. In the first stage, all the blocks are evaluated using a computationally simple criterion. Based on the result of this stage, a subset of the candidates is picked for the next stages where a more complex criterion is used [8, 9].

In the multi-resolution motion estimation, motion vectors are first estimated at the coarsest resolution, that is, in the sub-band(s) at the top of the pyramid for multi-resolution. The motion information from the sub-band(s) of the pyramid is then manipulated for the prediction at finer resolutions [10].

In [1], a new technique for the motion estimation has been proposed, that determines the motion vectors more accurately in situations where there are variations in the brightness values of the corresponding blocks from frame to frame by employing an exhaustive search technique. The motion vector thus obtained has been referred to as enhanced motion vector in contrast to the usual motion vector obtained by using the conventional techniques. Because of exhaustive search technique involved in determining the enhanced motion vectors, the underlying motion estimation technique is computationally expensive. This paper presents a new algorithm for a fast estimation of the enhanced motion vectors. This algorithm uses a DC matching technique and replaces the expensive 2-D block-matching operation by a simpler 1-D matching through the choice of two subsets of blocks as eligible block candidates for the search. An overview of the enhanced motion estimation technique is given in Section II. Section III gives the description of the proposed fast motion estimation algorithm. In Section IV, a study of the computational complexity of the new algorithm is carried out. The test results of applying the proposed method to a number of video test sequences are included in Section V.

II. Overview of the Enhanced Motion Estimation Scheme

In block-based motion estimation techniques, some metrics such as MAD or MSE is used as a measure of similarity between a current block and the reference block. However, in the case when there are changes in the pixel values of the pair of blocks (e.g. due to the brightness variations between the consecutive frames), the similarity alone cannot provide accurate motion estimation. Thus, in order to estimate the motion vectors more accurately, the changes in the characteristics of the corresponding blocks in the current and reference frames should be taken into consideration, in addition to the motion activities of the blocks. In [1], a scheme for the motion estimation, which takes into account the inter-frame brightness variations, has been proposed. In this technique, the variations in the brightness of two consecutive frames are considered on a block-by-block basis using the assumption that the transformation in the intensity value of each pixel within a block from frame $(k-1)$ to frame k takes place through a constant additive parameter. The proposed method uses the difference between the DC values of the corresponding blocks in frames $(k-1)$ and k as the additive parameter of the block. The proposed scheme, in effect, neutralizes the variations in the brightness characteristics of the corresponding blocks in the current and reference frames, and thus results in more accurate motion vectors. The scheme of finding an enhanced motion vector comprises the following steps:

i For each block in the current frame, referred to as the current block, its DC value DC_c is determined.

ii Within the search window located in the reference frame, the DC value of each displaced block is derived and denoted as DC_{ij}, where, $i, j = -D, \ldots, D$, D being the maximum displacement within the search window.

iii The difference between DC_c and DC_{ij} is calculated and denoted as DBV_{ij}, where, $i, j = -D, \ldots, D$.

iv The value of DBV_{ij} is added to all the pixel values of the corresponding displaced block. This procedure modifies the pixel intensities of the displaced blocks.

v A search is conducted to find the best-matched block between the current block and the modified displaced reference blocks.

vi The displacement between this best-matched block and the current block is called the enhanced motion vector and the value of DBV_{ij} corresponding to the best-matched block is called the differential brightness value.

If we denote by f_{pq} the intensity of a pixel at location (p, q) in a block of size $m \times n$ in the current frame, and by f'^{ij}_{pq} the intensity of a pixel at location (p, q) in a displaced reference block at location (i, j) within the search window, then the prediction error PEC_{ij} using the conventional motion vector can be calculated as $PEC_{ij} = \sum_{p=0}^{m-1} \sum_{q=0}^{n-1} (f_{pq} - f'^{ij}_{pq})^2$. The corresponding prediction error PEE_{ij} using the enhanced motion vector is given by $PEE_{ij} = \sum_{p=0}^{m-1}\sum_{q=0}^{n-1}(f_{pq} - (f'^{ij}_{pq} + DBV_{ij}))^2$, where DBV_{ij} is given by $DBV_{ij} = \sum_{p=0}^{m-1}\sum_{q=0}^{n-1}(f_{pq} - f'^{ij}_{pq}) / nm$. The difference between the prediction errors resulting from the two methods can be calculated as

$$PEC_{ij} - PEE_{ij} = mn(DBV_{ij})^2 \qquad (1)$$

Thus, the prediction error calculated by using the enhanced motion vectors is always smaller than or equal to the conventional motion vectors. We use the expression “modification value” to represent the difference between PEC_{ij} and PEE_{ij}. By calculating the prediction error using the conventional motion vectors and the modification value, the prediction error using the enhanced motion vectors can be obtained.

III. Proposed DC-Based Technique for Fast Motion Estimation

The schemes of matching DC components for the conventional motion estimation have been introduced in a number of papers [8, 11]. The basic idea is simple: The blocks in the current and reference frames cannot match well if their corresponding DC components do not match well. By taking advantage of this observation, a majority of the block candidates are quickly eliminated, and the expensive 2-D block-matching operation is transformed into a simpler 1-D block-matching operation. The DC matching [8] is based on the idea that two matched macroblocks must have a very similar sum-of-pixel values (SPV). This method sorts the blocks by matching the SPV of the block in the current frame with the SPVs of the displaced blocks within the search window in the reference frame. Then, a certain percent of all the blocks within the search window are considered for a more precise block matching operation in the next stage.

The scheme presented in this section introduces a new idea of DC matching. In This scheme, two subsets of block candidates are considered. The first subset consists of a fraction of α blocks within the search window that have the DC values closest to the DC value of the block under consideration. The second subset consists of a fraction of β blocks within the search window that have the DC values farthest to the DC value of the block of interest. According to (1), in the case that the differential DC value between the two blocks of two consecutive frames is small, the difference between the prediction errors resulting from using the enhanced and conventional motion vectors is also small. As a matter of fact, an enhanced motion vector produces a far superior prediction performance than the conventional motion vector when the differential DC value of the two blocks is large. This provides the motivation for forming the second subset of block candidates for determining the enhanced motion vector. If indeed one of the blocks from this subset qualifies to be chosen to determine the enhanced motion vector for the block under consideration, then this motion vector results in a smaller prediction error as compared to the one obtained by using the conventional motion vectors.

The partitioning scheme employed in the proposed motion estimation method divides a frame of $H \times W$ pixels into blocks of $N \times N$ pixels. Each of these blocks is further divided into non-overlapping sub-blocks of $n \times n$ pixels, which can be represented by $m \times m$ SPVs, where $m = N/n$.

Let us refer to each block by the coordinates of its upper left corner. We denote by $F_t(i, j)$ the intensity of a pixel with coordinates (i, j) in the current frame t. The SPV of a sub-block located at coordinates $(k\,n, l\,n)$ within the block (i, j) in the current frame is denoted by $M_t(i+kn,\ j+l\,n)$ and it is given by

$$\begin{aligned} M_t(i+kn,\ j+l\,n) &= \sum_{p=0}^{n-1}\sum_{q=0}^{n-1} F_t(i+k\,n+p,\ j+l\,n+q) \\ i &= 0,\ N,\ 2N,\ ...,\ H-N \\ j &= 0,\ N,\ 2N,\ ...,\ W-N \\ k,l &= 0,\ 1,\ 2,\ ...,\ m-1 \end{aligned} \qquad (2)$$

Similarly, the SPV of a sub-block located at the coordinates (kn, ln) within the block (i, j) in the reference frame $(t-1)$ can be calculated as

$$M_{t-1}(i+kn,\ j+ln) = \sum_{p=0}^{n-1} \sum_{q=0}^{n-1} F_{t-1}(i+kn+p,\ j+ln+q) \tag{3}$$

$$\begin{aligned} i &= 0,\ N,\ 2N,\ ...,\ H-N \\ j &= 0,\ N,\ 2N,\ ...,\ W-N \\ k,l &= 0,\ 1,\ 2,\ ...,\ m-1 \end{aligned}$$

To determine the eligible candidates among all the possible blocks within the search window, a full search in the SPV domain using the following sum of square error (SSE) criterion is carried out:

$$SSE_{i,j}(x,y) = \sum_{k=0}^{m-1} \sum_{l=0}^{m-1} \left[M_t(i+kn,\ j+ln) - M_{t-1}(i+kn+x,\ j+ln+y) \right]^2 \tag{4}$$

where $x, y = -D, -D+1, ..., D-1, D$. Next, depending on the SSE values, the proposed method selects two subsets of block candidates for further consideration in the next stage. The first subset consists of $\alpha(2D+1)^2$ block candidates having the lowest SSE values, where $0 < \alpha \le 1$. The second subset of block candidates is chosen as follows.

Inside the search window that contains $(2D+1)^2$ possible locations, we define a smaller window that contains $(2D'+1)^2$ locations, where $D' < D$. If we define a parameter ρ as

$$\rho = \frac{(2D'+1)^2}{(2D+1)^2} \tag{5}$$

where $0 < \rho \le 1$, then, the smaller window contains $\rho(2D+1)^2$ possible locations. Since the search window resides in the previous frame, the motion vectors for all its blocks are already known. We move the position of the smaller window to a location such that the central block of the displaced smaller window is pointed by the motion vector of the corresponding central block of the undisplaced smaller window. If through this process, the smaller window moves out of the larger search window partially, then it should be shifted horizontally or vertically or in both directions in order to be bring it barely inside the larger search window. The second subset consists of $\gamma(2D'+1)^2$ candidates having the largest SSE values, where $0 < \gamma \le 1$. Therefore, the total number of the candidates in the second subset is $\gamma\rho(2D+1)^2$. We define a parameter β as given by

$$\beta = \gamma\rho \tag{6}$$

Thus, $0 < \beta \le 1$. In the next stage, for all the eligible candidate blocks, the enhanced motion vectors are determined. In this stage, the mean-squared error is used as the similarity measure in determining the block matching.

As mentioned earlier, the $\beta(2D+1)^2$ candidates are picked from the area of the smaller search window such that the DC difference between corresponding blocks is large. This subset of candidates is chosen, since the large DC difference may be result of the inter-frame brightness variations. In the case that an enhanced motion vector is selected from this subset of candidates, the prediction error is significantly reduced as compared to the prediction error resulting from the use of the corresponding conventional motion vector.

VI. Computational Complexity

For a given value of the displacement D, the full search method implies that $(2D+1)^2$ locations within the search window must be searched. Each of these locations gives a prediction error for the block. The motion vector of a block is determined by the location that gives the minimal prediction error. The computational complexity of the full search method is the direct consequence of the expensive 2-D block matching process.

The conventional full search algorithm requires $(2N^2-1)(2D+1)^2$ additions to calculate the motion vector for a block of size $N \times N$ based on the minimum prediction error determined by the mean-square error. This calculation does not require any multiplications, if the squared values are read from a lookup table.

To calculate the motion vector for a block of size $N \times N$ using the proposed method, the following operations are required.

- Calculation of the SPVs in the reference frame requires almost $2nN^2$ additions per motion vector. Calculation of the sum of the intensities of the reference frame requires (n^2-1) additions for the first block. For the successive blocks, we just need to shift one column of n pixels to the left or shift one row of n pixels downwards. Therefore, on the average, only $2n$ additions are required for each of the successive blocks.
- Calculation of SPVs in the current frame requires $m^2(n^2-1) = N^2 - m^2$ additions. There are m^2 sub-blocks of size $n \times n$. To calculate the SPV for each of these sub-blocks, (n^2-1) additions are required.
- To calculate the values of SSEs, $(2m^2-1)(2D+1)^2$ additions are required. This calculation does not require any multiplications if the squared values are read from a lookup table.
- Sorting the SSEs for $\alpha(2D+1)^2$ candidates with the lowest SEE values requires almost $\alpha(2D+1)^2 - \alpha(\alpha+1)/2$ comparisons (using the buble sort method).
- Sorting the SSEs for $\beta(2D+1)^2$ candidates with the highest SEE values requires almost $\beta(2D+1)^2 - \beta(\beta+1)/2$ comparisons (using the buble sort method).
- Calculation of the differential SPV values between the block in the current frame and all the $(\alpha+\beta)$ eligible candidates in the reference frame requires $(\alpha+\beta)(2m^2-1)$ additions. The SPV value of a given block is determined by the summation of all the m^2 SPV values of the block. This requires m^2-1 additions for each block. The differential SPV value between a pair of blocks requires one subtraction. Thus a total of, $2 \times (m^2-1)+1 = 2m^2-1$ additions are required for each eligible candidate.
- To calculate the prediction errors using the conventional motion vectors for $(\alpha+\beta)(2D+1)^2$ candidates, $(2N^2-1)(\alpha+\beta)(2D+1)^2$ additions are required. Moreover, one extra addition is required to calculate the prediction error of each candidate according to (1). Thus, the total number of additions required is given as $(2N^2-1+1)(\alpha+\beta)(2D+1)^2 = (2N^2)(\alpha+\beta)(2D+1)^2$. This part of the calculation does not require any multiplication, if the modification values are read from a lookup table using the differential SPV values as indices.

Thus, the total number of additions required for each block using the proposed method is approximately given by

$$\begin{aligned} &2nN^2 + N^2 - m^2 + (\alpha+\beta)(2m^2-1) + \\ &\qquad\qquad \left[\ (2m^2-1) + (\alpha+\beta)(2N^2)\ \right](2D+1)^2 \end{aligned} \tag{7}$$

The enhanced full search algorithm requires the following operations.

- The calculation of the SPV values for all displaced blocks in the reference frame requires almost $2nN^2$ additions per motion vector.
- The calculation of the SPV value of a block in the current frame requires (N^2-1) additions.

- To calculate the differential SPV values between the block in the current frame and all the displaced blocks within the search window in the reference frame, $(2D+1)^2$ additions are required.

- To calculate the prediction errors using the conventional motion vector for $(2D+1)^2$ candidates, $(2N^2-1)(2D+1)^2$ additions are required. Moreover, one extra addition is required to calculate the prediction errors of each candidate according to (1). Thus, the total number of additions required is $(2N^2-1+1)(2D+1)^2=(2N^2)(2D+1)^2$ to calculate the prediction errors using the enhanced motion vectors. This part of calculation does not require any multiplications, if the modification values are read from a lookup table using the differential SPV values as indices.

Thus, the total number of additions required in the calculation of the enhanced full search method for each block is given by

$$2nN^2+N^2-1+(2N^2+1)\,(2D+1)^2 \tag{8}$$

V. Simulation Results

The proposed technique has been tested by using four video test sequences: The Football and Flower sequences with 352×240 pixels/frame format, and the Table Tennis and Claire sequences with the 352×288 pixels/frame format. For the purpose of comparison, the conventional full search method and the full search method using enhanced motion vectors have also been implemented. The results are obtained in terms of AMSPE as given by

$$AMSPE=\frac{1}{K}\sum_{k=1}^{K}MSPE(k) \tag{9}$$

where K is the total number of the frames tested and MSPE (mean-square prediction error) is the average energy per pixel in the residual image as given by

$$MSPE(k)=\frac{1}{HW}\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}(f_k(i,j)-\bar{f}_k((i,j))^2 \tag{10}$$

$f_k(i,j)$ and $\bar{f}_k(i,j)$ being, respectively, the original and predicted pixel intensities at the $(i,j)th$ position of the kth frame.

For a given algorithm, the speed ratio ν is defined as the ratio of the execution time of the conventional full search method to that required by the algorithm. Table I presents the AMSPE and ν for the four video test sequences. The parameter of the proposed method in this experiment is set to $m=4, n=2, \alpha=0.2, \gamma=0.46$ and $\beta=0.1$. This table shows that on the average, the proposed method can reduce the prediction error by 15% as compared to the conventional full search method. It should be noted that the full search method employing the enhanced motion vectors reduces the prediction error by 6% as compared to the fast search method using the enhanced motion vectors.

Fig. 1 shows the plots for the number of addition operations normalized to that of the conventional full search method as a function of α for the fast search method and the full search method using the enhanced motion vectors. The block parameters used in this experiment are set to $D=7$, $N=8$, $m=4$ and $n=2$. Obviously, increasing the value of α or β increases the computational complexity.

Figs. 2(a)-(d) present the MSPEs obtained from using 50 frames of the Football, Flower, Table Tennis and Claire sequences, respectively. The parameters of the proposed method are set as $D=7$, $m=4$, $n=2$, $\alpha=0.2$, $\gamma=0.46$ and $\beta=0.1$. With these parameter values, the proposed method is approximately 2.1 times faster than conventional full search method. From these figures, it is seen that the prediction error yielded by the proposed method is close to the one obtained from the full search technique using the enhanced motion vectors and is superior to that of the conventional full search technique in terms of the prediction error.

Let us consider a modified version of Miss America sequence, referred to as "Modified Miss America" which is obtained from the original sequence by replacing its 2nd and 20th frames by their negatives through a quick negative transformation. To show the efficiency of the method, the MSPE results obtained form the four consecutive frames of the Modified Miss America sequence are represented in Table II. These results demonstrate the efficiency of using the second subset of block candidates in the proposed scheme. As seen from this table, changing α from 0.1 to 0.3 ($\beta = 0$), is not as efficient as choosing α=0.1 and $\beta = 0.16$.

VI. Conclusion

In this paper, a novel fast algorithm for the estimation of enhanced motion vectors has been proposed. The algorithm reduces the computational complexity by developing a two-stage scheme for the estimation of the enhanced motion vectors. In the first stage, the proposed method matches the sum of pixel values of the sub-blocks of a given block in the current frame with those of a block in the reference frame to find the eligible candidates of blocks to be used in the next stage. The result of the first stage is the generation of two subsets of block candidates, each containing only a small percentage of the total number of blocks in the search window. One subset consists of the best-matched blocks and the other one, the worst-matched blocks. In the second stage of scheme, the enhanced motion vector for all the eligible candidates are determined. The proposed method offers a trade-off between the speed and the performance through the use of two controlling parameters. Experiments conducted over several sequences demonstrate that the proposed motion estimation technique is significantly faster than the full search techniques using the conventional or enhanced motion vectors. The prediction error yielded by the proposed method is close to the one obtained from the full search technique using the enhanced motion vectors and is superior to that by using the conventional full search technique.

REFERENCES

[1] F. Ahmadianpour and M.O. Ahmad, **"**Region-wise motion compensation technique using enhanced motion vectors", In *Proc. of SPIE- Image and video communication and processing*, January 2003, Vol. 5022, pp. 102-110.

[2] M. Alkanhal, D. Turaga, and T. Chen, "Correlation based search algorithms for motion estimation", *Picture Coding Symposium*, Portland, Oregon, April 1999

[3] L.M. Po and W.C. Ma. "A novel four step search algorithm for fast block motion estimation". *IEEE Trans. Circuits and Syst. for Video Technol.,* Vol. 6, No. 3, pp. 313-7, 1996.

[4] H.M. Jong, L.G. Chen and T.D. Chiuech, "Accuracy improvement and cost reduction of the 3 step search block matching algorithm for video coding". *IEEE trans. Circuits and Syst. for Video Techno.,* Vol. 4, No.1, pp.88-90, 1994.

[5] C.H. Lin and J.L. Wu, "Fast motion estimation algorithm with adjustable search area.", In *Proc. of SPIE- Image and video communication and processing*, 1995, Vol. 2501, pp. 1328-1336.

[6] L.K. Liu and E. Feig, "A block-based gradient descent search algorithm for block motion estimation in video coding", *IEEE Trans. Circuits and Syst. for Video Techno.*, Vol. 6, No. 4, pp. 419-422, August 1996.

[7] A.M. Tourapis, O.C. Au, and M.L. Liou. Fast motion estimation using circular zonal search., *Proc. SPIE Visual Comm. & Image Processing*, (VCIP'), Vol. 2, pp. 1496-1504, January 1999.

[8] J. Zhang, M.O. Ahamd and M.N.S. Swamy, "DC matching for motion estimation", *Electronic letters*, PP. 33:1447-1448, 1997

[9] F. Dufaux and J. Konrad, "Efficient, robust and fast global motion estimation for video coding", *IEEE Trans. On Image processing,* Vol. 9, No. 3, March 2000.

[10] S. Zafar, Y. Q. Zhang and B. J. Jabbari, "Multiscale video representation using multiresolution motion compensation and wavelet decomposition", *IEEE Journal select Areas Commun.*, Vol. 1, pp. 24-35, January 1993

[11] C. Tu, T.D. Tran, J. L. Prince and P. N. Topiwala, "Fast block-based motion estimation via DC matching", In *Proc. 34th IEEE Conference on Signals, Systems, and Computers,* 2000, Vol. 2, pp. 1379-1383.

Table I
AMSPE AND ν FOR VARIOUS VIDEO TEST SEQUENCES

Sequence	Conventional Full search method	Full search method using enhanced motion vectors		Fast search method using enhanced motion vectors	
	AMSPE	AMSPE	ν	AMSPE	ν
Football	228.11	171.4	0.91	182.82	2.1
Flower	170.87	138.40	0.93	146.91	2.3
Table Tennis	86.11	72.62	0.92	78.03	2.2
Claire	3.74	2.92	0.93	3.13	2.3

Table II
The MSPE results obtained form four consecutive frames of the Modified Miss America sequence.

Frame #	Conventional Full search	Fast Enhanced $\alpha = 0.1, \beta = 0$	Fast Enhanced $\alpha = 0.3, \beta = 0$	Fast Enhanced $\alpha = 0.1, \beta = 0.16$
19	11.23	10.43	10.01	10.01
20	23016	378	297	179
21	22986	377	300	181
22	11.55	10.52	10.26	10.28

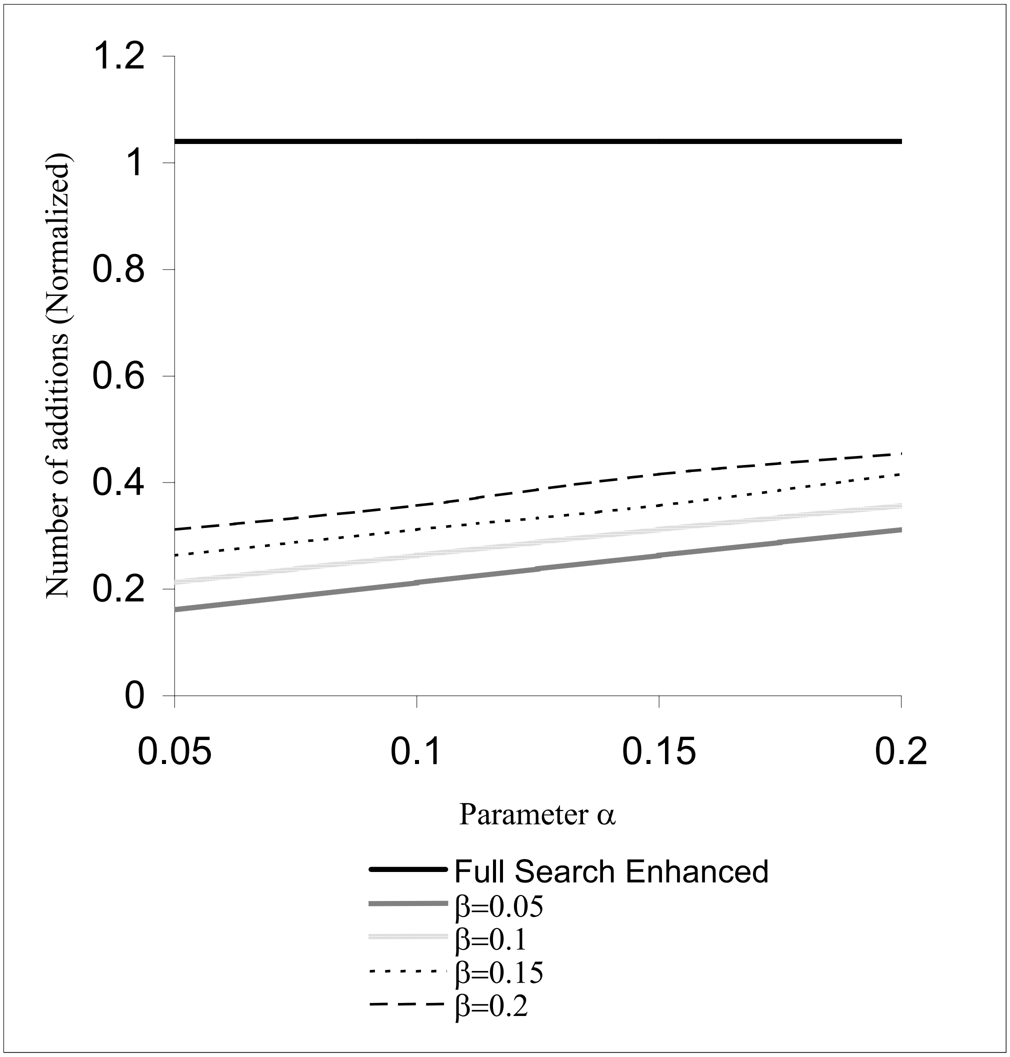

Fig. 1: Number of addition operation per block normalized with respect to that of the conventional full search method.

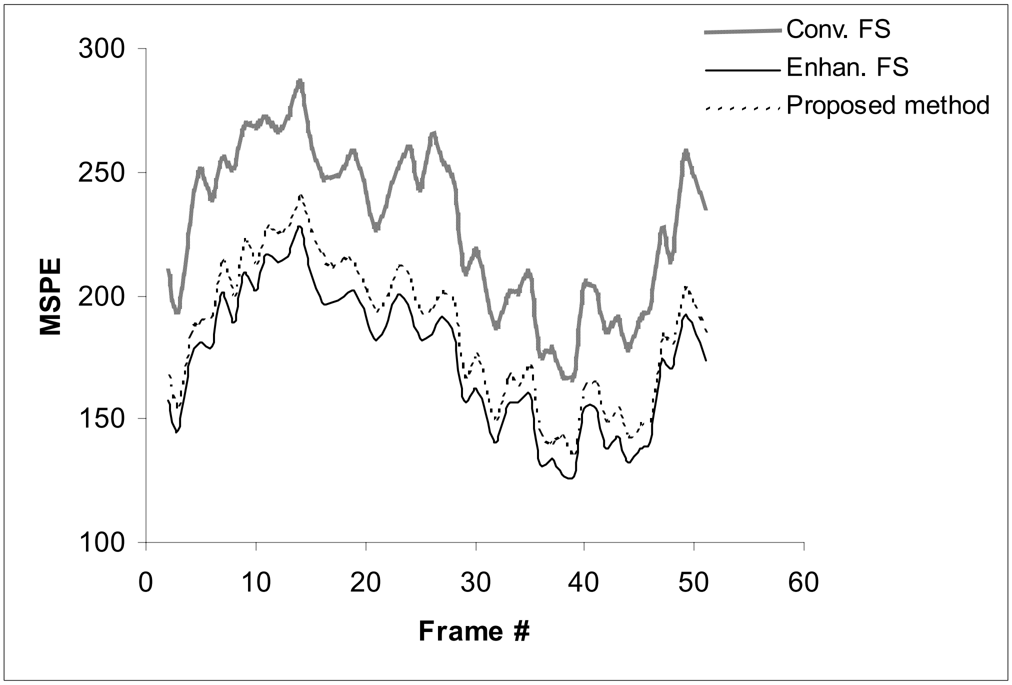

Fig. 2(a): The MSPE obtained from 50 frames of the Football sequence.

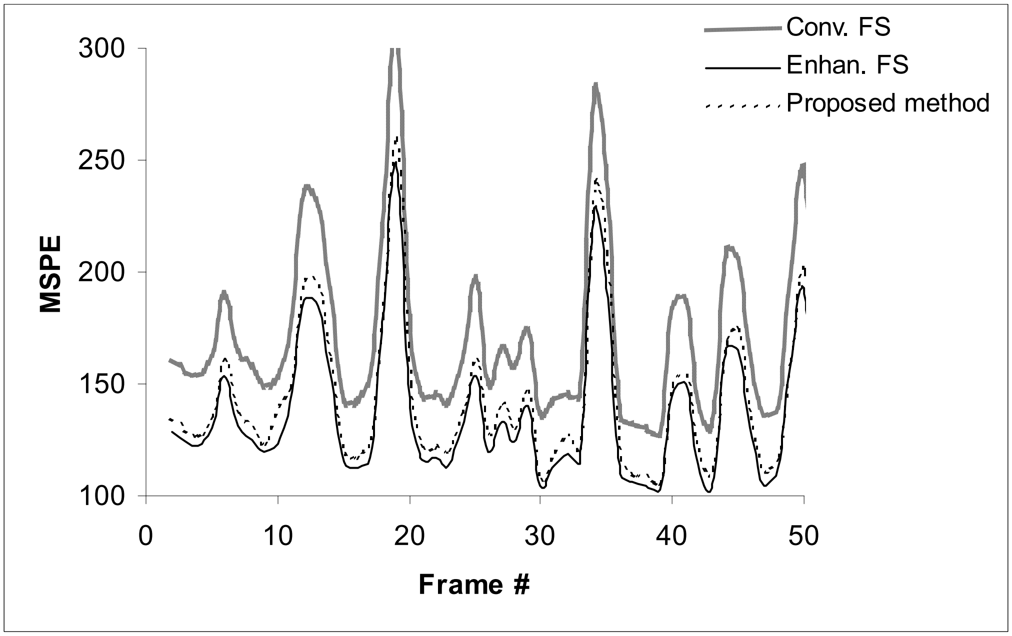

Fig. 2(b): The MSPE obtained from 50 frames of the Flower sequence.

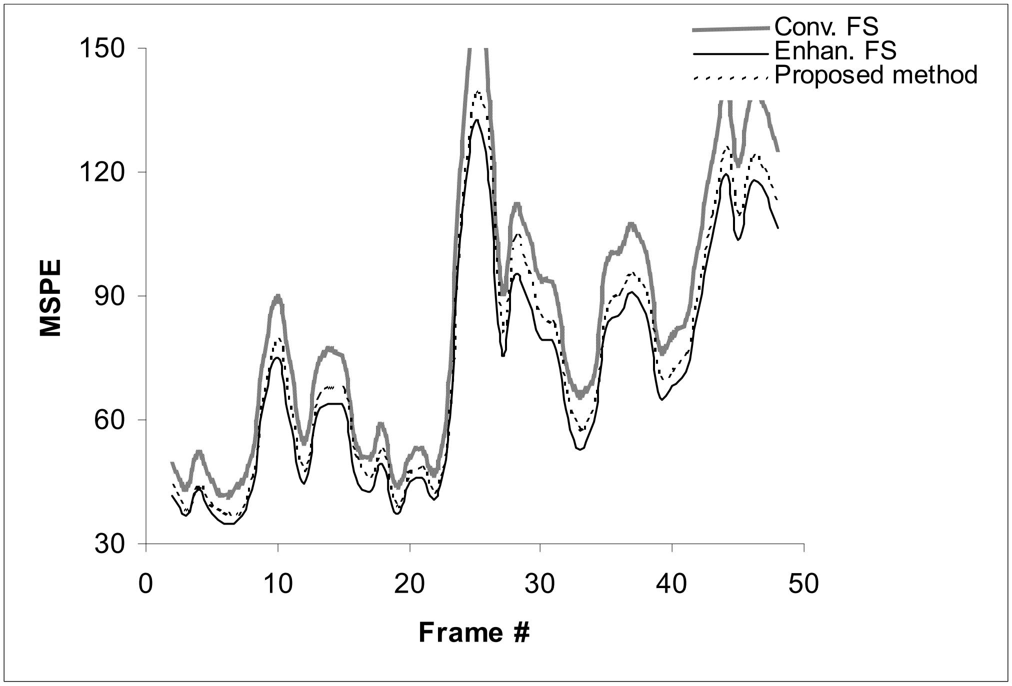

Fig. 2(c): The MSPE obtained from 50 frames of the Table Tennis sequence.

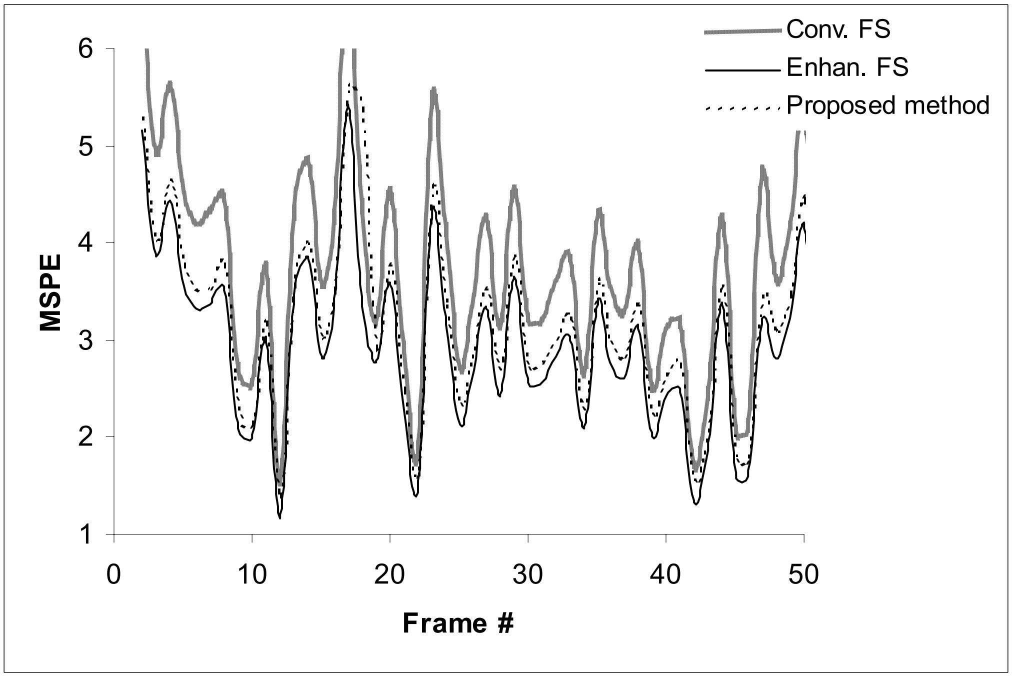

Fig. 2(d): The MSPE obtained from 50 frames of the Claire sequence.