Adaptive Motion Search Based on Block Difference and Motion Magnitude

Dong-Keun Lim and Yo-Sung Ho

Kwangju Institute of Science and Technology
1 Oryong-dong, Puk-gu, Kwangju, 500-712, Korea
{dklim, hoyo}@kjist.ac.kr

Abstract. In this paper, we derive optimal search patterns for fast block matching motion estimation. Since the motion search pattern is important in terms of search speed and correctness of the motion information, we consider various search patterns and search strategies. By analyzing the block matching algorithm as a function of the block size and distance in the search area, we find analytic search patterns for initial motion estimation. We also propose an adaptive motion search algorithm, where we exploit the correlation between block difference and motion magnitude. The proposed idea can provide an analytical ground for the MPEG-4 algorithms for fast motion search. We can improve the prediction accuracy of motion estimation, while reducing the required computational complexity compared to other fast block matching algorithms.

1 Introduction

Block matching motion estimation algorithms are popularly employed in several video coding standards, such as H.261, H.263, MPEG-1, MPEG-2, and MPEG-4. The main objective of motion estimation is to reduce temporal redundancy between successive picture frames. After partitioning the current image frame into non-overlapping rectangular blocks, the block matching algorithm attempts to find the best-matched block in the search area of the reference frame. Its performance is determined by motion prediction accuracy and computational complexity.

Important parameters in motion estimation are: size of block, search area, search pattern, search strategy, and matching criterion. Various algorithms for fast block search have been developed to reduce the computational burden associated with the full-search block matching algorithm (BMA) [1-12]. Since most fast BMAs generally take heuristic approaches to reduce the computational complexity, they sacrifice reconstructed image quality. They mainly focus on the search strategy using heuristic methods. Recently, it is known that the motion search pattern has an important influence on search speed and correctness of the motion information. Since the search strategy depends on search patterns for efficient motion estimation, we analyze search patterns for fast motion estimation.

F. Boavida et al. (Eds.): IDMS/PROMS 2002, LNCS 2515, pp. 118-129, 2002.

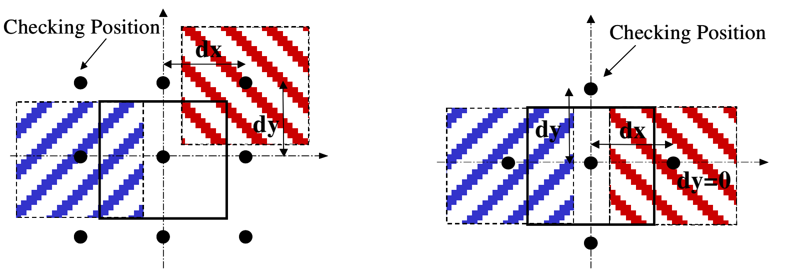
Although mathematical modeling and analysis for the block matching algorithm include ill-posed problems, a theoretical approach is possible by simplifying the block matching problem.

In this paper, we derive an optimal search pattern for fast motion estimation analytically. By examining the relationship between block difference and motion magnitude, we propose an adaptive motion search algorithm. Based on statistical correlation between object displacement and frame difference in each block, we change the search pattern for motion estimation adaptively. We also explain how we can generate initial search patterns. The proposed adaptive motion search algorithm can reduce the computational burden substantially relative to full-search block matching algorithm, while providing a good motion prediction. The analysis presented in this paper can support several ideas proposed for the MPEG activities [8-12].

2 Optimal Search Pattern

2.1 Problem Statement

Since the shape and the size of the search pattern in the fast block matching algorithm jointly determine the convergence speed and motion estimation performance, we consider various search patterns: rectangle or diamond of different sizes. Fig. 1(a) and Fig. 1(b) show search patterns for the three-step search (TSS) and the 2-D logarithmic search (TDL), respectively [1-3].



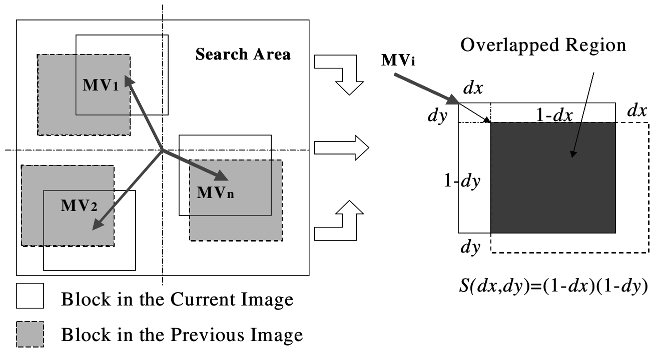
(a) Three-Step Search (b) Two-Dimensional Logarithmic Search

Fig. 1. Search Patterns for Two Fast BMAs

As shown in Fig. 1, the search pattern for the three-step search algorithm is a rectangle, but that of the two-dimensional logarithmic search algorithm has a diamond shape. In Fig. 1, *dx* and *dy* represent distances between adjacent checking positions. In the block matching algorithm, we measure the correlation between the current block and the reference block by shifting the center of the reference block to candidate positions in the search pattern.

In the first step of TSS, block correlations at nine checking positions are examined, as indicated in Fig. 1(a). In the second step, eight new checking positions are spaced less coarsely around the best matched position in the first step. In the third step, the distance between checking positions is further reduced to one pixel, and the minimum distortion position is selected.

TDL tracks the direction of minimum distortion. In each step, five checking positions located in a diamond-shaped search pattern are examined, as shown in Fig. 1(b). If the minimum distortion is observed in the center of the checking positions or at the boundary of the search area, the distance between checking positions is reduced. The final motion vector is determined as the minimum distortion position among all the checking positions of one pixel distance.



(a) Motion Vector Distribution (b) Corelation Value

Fig. 2. Correlation of Image Blocks

In order to make a simple analysis, we consider a rectangular image block whose pixel values are unity and its background has all zero values. As shown in Fig. 2(a), motion vectors may locate at any position in the search area. Although we do not know the direction and the magnitude of the actual object displacement MV_i in Fig. 2, we can derive an optimal search pattern for motion estimation. Since we use checking positions to find the candidates of the motion vector in the block matching algorithm, we investigate equi-correlation contours as a function of the distances, *dx* and *dy*, between checking positions in the search pattern.

2.2 Derivation of Optimal Search Pattern

In our analysis, the correlation *S(dx,dy)* between two rectangular image blocks can be calculated as the normalized area of the overlapped region, as shown in Fig. 2(b).

$$S(dx,dy) = (1-|dx|)(1-|dy|), \quad 0 \le |dx| \le 1, \quad 0 \le |dy| \le 1 \tag{1}$$

We note that *S(dx,dy)* has the following symmetry property:

$$S(dx,dy) = S(-dx,dy) = S(dx,-dy) = S(-dx,-dy) \tag{2}$$

From Eq. (1), we can find locations of distance (*dx*,*dy*) having the same correlation value, and draw equi-correlation contours.

$$|dx \cdot dy| - |dx| - |dy| + 1 - S(dx,dy) = 0 \tag{3}$$

If we represent each point (*dx*,*dy*) in the block by a linear function,

$$dy = c \cdot dx, \quad c \in \Re \tag{4}$$

we obtain dx as a function of c and $S(dx,dy)$

$$dx = \frac{(c+1) \pm \sqrt{(c-1)^2 + 4c \cdot S(dx,dy)}}{2c} \tag{5}$$

In order to obtain a distribution of block correlation values, we apply boundary conditions to Eq. (4) and Eq. (5).

Case 1: $S(dx,dy)=0$, two blocks are not overlapped.

$$dx = \frac{(c+1)+\sqrt{(c-1)^2}}{2c} = \begin{cases} \dfrac{c+1+c-1}{2c} = 1, \ c>1, \ dy = c \cdot dx = c \\ \dfrac{c+1+1-c}{2c} = \dfrac{1}{c}, \ c<1, \ dy = c \cdot dx = 1 \end{cases} \tag{6}$$

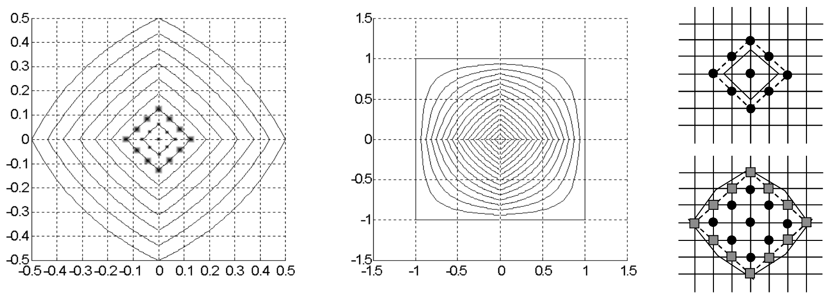
$$dx = \frac{(c+1)-\sqrt{(c-1)^2}}{2c} = \begin{cases} \dfrac{c+1-c+1}{2c} = \dfrac{1}{c}, c>1, dy = c \cdot dx = 1 \\ \dfrac{c+1-1+c}{2c} = 1, c<1, dy = c \cdot dx = c \end{cases} \tag{7}$$

Case 2: $S(dx,dy)=1$, two blocks are completely overlapped.

$$dx = \frac{(c+1)+\sqrt{(c+1)^2}}{2c} = 1 + \frac{1}{c}, \quad dy = c \cdot dx = c+1 \tag{8}$$

$$dx = \frac{(c+1)-\sqrt{(c+1)^2}}{2c} = 0, \quad dy = c \cdot dx = 0 \tag{9}$$

Since only Eq. (6) and Eq. (9) meet the boundary condition, we know that contour lines converge to rectangular shapes. The same correlation $S(dx,dy)$ exists along the boundaries of the block when there is no overlap between two blocks. By changing the value of c from 0 to infinity in Eq. (5), we can plot equi-correlation contours, as shown in Fig. 3(a).



(a) Equi-correlations (7/16≈1/2) (b) Equi-correlations (15/16≈1) (c) CP Sampling

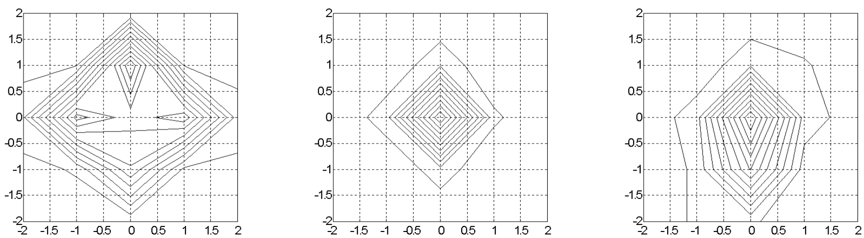
Fig. 3. Generation of Search Patterns

In Fig. 3(a), we show the result for limited search area to [-7, 7], or 7/16 ≈ 0.5. Furthermore, when limiting search area to [-15, 15], or 15/16 ≈ 1, we can obtain the derived search pattern in Fig. 3(b). From this result, conventional search patterns can be regarded as subsets of the proposed one.

The analyticallly derived search pattern can be used to analyze search patterns of conventional block matching algorithms. We note that the diamond search [8-9,11] and the advanced zonal search [10-12] have similar characteristics. By sampling checking positions from continuous analytical equi-correlations in Fig. 3(a) and Fig. 3(b), we choose discrete search patterns, as shown in Fig. 3(c). If the contour line is not aligned with a pixel position, we choose the closest pixel position as a checking point. The diamond shape can have different sizes and sampling positions according to motion characteristics in the search pattern. Therefore, there are different trade-off points between search strategy and search pattern. Accuracy and efficiency of motion estimation depend on the selected points.

2.3 Experimental Results for Optimal Search Patterns

In order to verify the derived search pattern, we perform computer simulations on ITU-T monochrome test sequences: MISS AMERICA, CLAIRE, CALENDAR, and SALESMAN, each of which contains 88 frames of the CIF format; FOOTBALL, having 88 frames of the ITU-T format. For the full-search block matching algorithm, the block size is 16×16 and the search area is [-7,+7]. In other words, $0 < (|dx|, |dy|) < 7/16 \approx 1/2$.



(a) MISS AMERICA (b) SALESMAN (c) FOOTBALL

Fig. 4. Experimental Search Patterns

While Fig. 3(a) indicates analytically derived search patterns for $S(dx,dy) > 1/2$, Fig. 4 shows experimental search patterns obtained from motion vector fields for MISS AMERICA, SALESMAN, and FOOTBALL, respectively. From Fig. 3 and Fig. 4, we observe that the derived search patterns and experimental ones are all diamond-shaped, which implies that our derivation is valid for the optimal search pattern for the block matching algorithm. The diamond search pattern is simple, but efficient for fast BMA. From Fig. 4, we can note that most displacements are included within a two-pixel spaced diamond shape. It corresponds to the inside of a diamond having the normalized size of $2/7 \approx 0.29$ in Fig. 3(a). In order to improve the motion estimation accuracy, we can increase the size of the diamond and the number of checking positions, and modify the diamond shape along its external boundary.

3 Block Matching Algorithm Using Optimal Search Pattern

3.1 Criteria for Block Matching Algorithm

The degree of the matching of image blocks can be measured by various criteria [1], including the mean absolute difference (MAD), the mean square error (MSE), and the matching pel count (MPC).

$$MAD(i,j) = \frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\left|I_t(k,l) - I_{t-1}(k+i,l+j)\right| \tag{10}$$

$$MSE(i,j) = \frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\left[I_t(k,l) - I_{t-1}(k+i,l+j)\right]^2 \tag{11}$$

$$T(k,l,i,j) = \left\{\begin{array}{ll} 1, & if\ \left|I_t(k,l) - I_{t-1}(k+i,l+j)\right| \le THS \\ 0, & otherwise \end{array}\right\} \tag{12}$$

$$MPC\,(i,j) = \sum_{k=1}^{N}\sum_{l=1}^{N} T(k,l,i,j)$$

where $I_t(k,l)$ and $I_{t-1}(k,l)$ be the luminance pixel values in the current frame and in the previous frame, respectively.

In order to reduce the computational complexity of the block matching operation, we define a sum of absolute differences (SAD) by combining MAD and MPC.

$$T(k,l,i,j) = \left\{\begin{array}{l} 1,\ if\left|I_t(k,l) - I_{t-1}(k+i,l+j)\right| \ge THS \\ 0,\ otherwise \end{array}\right\} \tag{13}$$

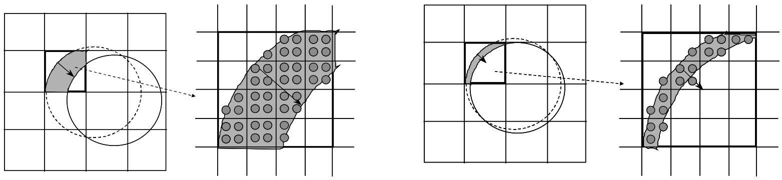
$$SAD(i,j) = \sum_{k=1}^{N}\sum_{l=1}^{N} T(k,l,i,j)\left|I_t(k,l) - I_{t-1}(k+i,l+j)\right|$$

The main idea of the new matching criterion is that we only count pixel differences that have significant changes of luminance values. Since pixels that have small changes of luminance values are not included in the computation for SAD, its complexity is reduced substantially. One remaining issue with the new matching criterion is how to select the threshold value THS properly. Since the human visual system (HVS) is sensitive to large changes of luminance values, we can set a just noticeable difference (JND) as THS. For blocks of small displacement, we set the threshold value to 0; then, Eq. (13) is equivalent to MAD.

3.2 Adaptive Motion Search Algorithm

In a teleconferencing video, most image blocks are regarded as stationary or quasi-stationary. Motion vectors for stationary image blocks are mostly around (0,0). In order to decide if a block is stationary, we consider the following situations.

In general, a large object displacement would produce a large block difference (BD) within the search area, as shown in Fig. 5. In Fig. 5, black dots indicate pixel positions having significant changes of luminance values. To calculate the block difference, we use SAD. However, we may have the following exceptional cases.



(a) Large Motion (b) Small Motion

Fig. 5. Object Movement and Block Difference

Case 1: Even if the actual object displacement is large, the block difference is smaller than the threshold value. This situation can occur when similar blocks in the same image object move to the same direction by the same amount. Since it generates an aperture problem [4], we cannot solve it properly. In this case, we simply assume that the block has a small displacement.

Case 2: Even if the actual object displacement is small, the block difference is larger than the threshold value. This case may happen when the background and the object have large luminance differences. If the number of pixels having significant luminance changes is small, we assume that this case occurrs and the block has a small displacement.

Fig. 6 explains the overall procedure of the proposed adaptive motion search (AMS) algorithm, where *Ns* denotes the number of significant pixels in the block.

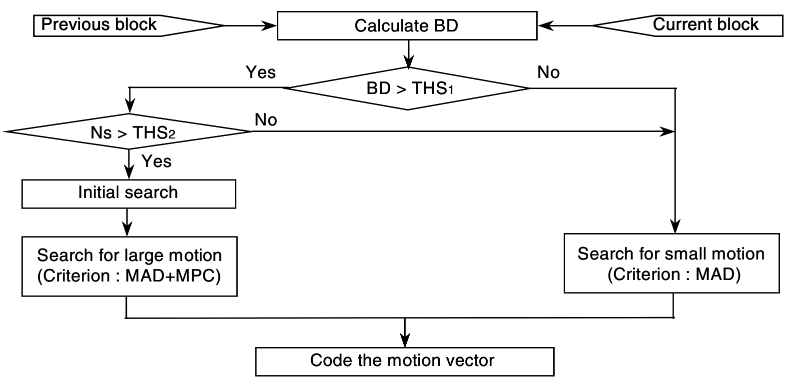
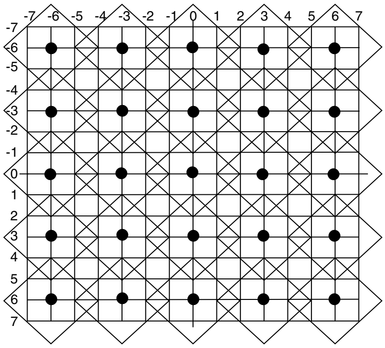
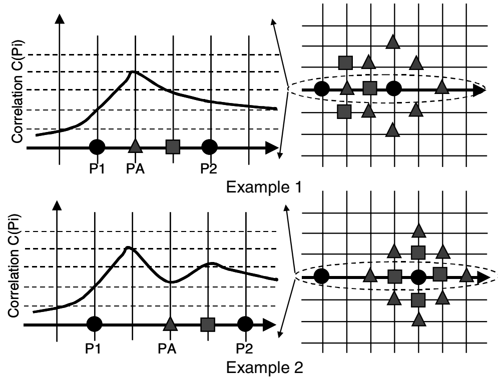


Fig. 6. Adaptive Motion Search Algorithm

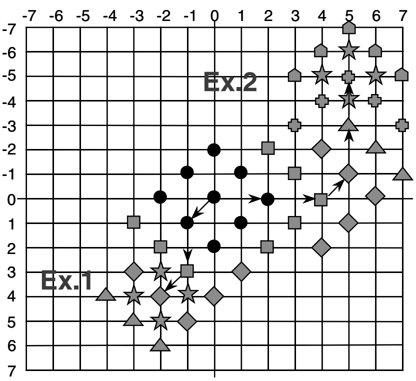
If the block difference is large, we use the initial search pattern, shown in Fig. 7(a). The initial search pattern consists of regularly spaced lattice positions that cover the whole or the part of the search area. Once the position of the minimum distortion is selected during the initial search, other positions near the minimum distortion position are examined in the next stages, as illustrated in Fig. 7(c) to Fig. 7(f).



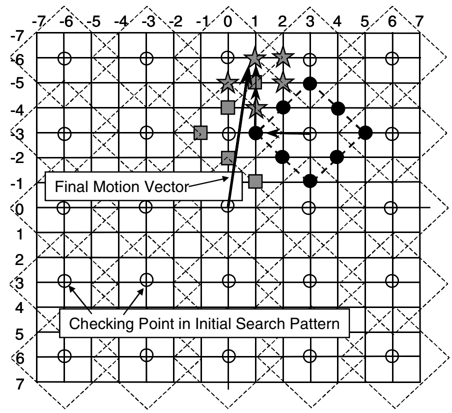
(a) Initial Search Pattern (ISP)



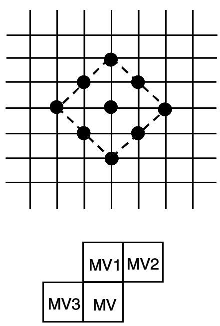
(b) Mathematical Considerations for ISP



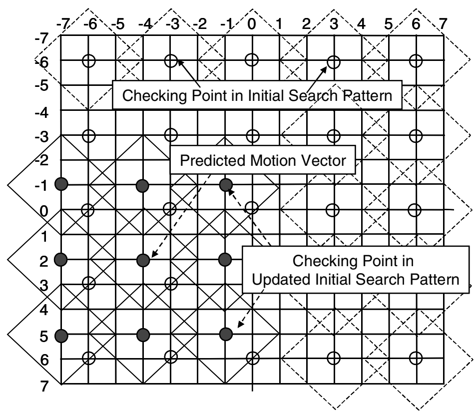
(c) Next Stage Examples



(d) Next Stage Search in a Lattice



(e) Predictive Motion Vector



(f) Updated Initial Search Pattern

Fig. 7. Search Strategy and Search Pattern

Now we explain how we obtain the initial search pattern. Fig. 7(b) shows a problem with two examples when we generate the initial search pattern. The problem arises when we use improper resolutions for the center of distributed search patterns, especially when we treat image blocks having large motions. Rounded dots indicate pixel positions in the lattice in Fig. 7(a). In Fig. 7(b), each 1-D graph in the left side is the cross view of the horizontal axis at the center of each 2-D diagram that is drawn in the right side.

We analyze Example 1 in Fig. 7(b) as follows, where C( ) is a correlation function.

1. Initial Search Pattern: C(P1) < C(P2) → Choose P2
2. Check PA in Diamond Search Pattern:
   C(P2) < C(PA) → Choose PA
3. If C(P1-1) > C(PA), diamond search will be progressed in (P1-1). (Not in this example)
4. One Pixel-resolution Search:
   Final MV is PA, since C(PA) > C((PA+P2)/2).
   (Correct Operation)

From Example 2 in Fig.7(b), we have

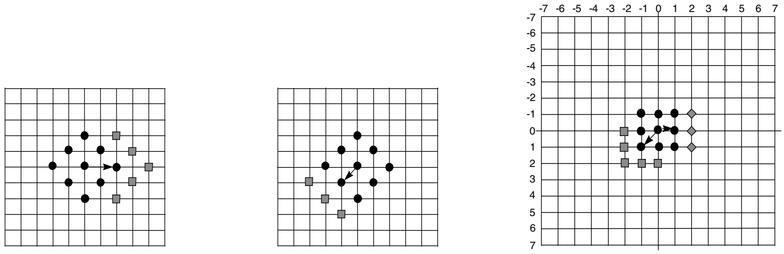
1. Initial Search Pattern: C(P1) < C(P2) → Choose P2
2. Check PA in Diamond Search Pattern:
   C(PA) < C(P2) → Choose P2
3. One Pixel-resolution Search:
   Final MV is (PA+P2)/2, since C(P2) < C((PA+P2)/2).
4. But C((PA+P2)/2) < C((P1+PA)/2)
   → Global maximum exists (Incorrect Operation).

The output of Example 2 is incorrect. Since the distance between diamond search patterns is too long, the border of each diamond search pattern cannot cover candidate positions properly. In order to solve this problem, we can increase the size of diamond search patterns or reduce the distance between the diamond search patterns. The diamond search pattern resembles a cell in a lattice.

Fig. 7(c) shows two examples of motion search. The role of the initial search pattern is the lattice, as shown in Fig. 7(d). With this lattice, we can reduce the number of search steps in the next stage. This initial search pattern can be generated according to the size of the diamond search pattern. We also consider a predictive motion vector to change the center of the lattice, as shown in Fig. 7(f). In order to find a predictive motion vector, we use a median value among motion vectors of the neighboring blocks, MV1, MV2, and MV3 in Fig. 7(e). Around this predicted motion vector, we generate an updated initial search pattern, as shown in Fig. 7(f).

Fig. 8(a) and Fig. 8(b) explain two different search strategies for large motion. Depending on the last position of the minimum distortion, we add three or five new checking positions. This procedure is repeated until we find the minimum distortion in the center of the search pattern or at the boundary of the search area. The final displacement vector is the position of the minimum distortion among all one-pixel spaced positions around the last position of the minimum distortion.

If the block difference is small, the search area is limited to a local region. The procedure for small motion is similar to that for large motion; however, we start with 3×3 neighboring positions, as shown in Fig. 8(c).



(a) Horizontal Direction (b) Diagonal Direction (c) Small Motion

Fig. 8. Search Strategies for Large Motion and Small Motion

4 Experimental Results

Computer simulations have been performed on the monochrome test sequences. In our simulation, the original image of the previous frame was used as a reference frame to generate a motion-compensated prediction image. Quality of the motion-compensated prediction image is measured by the peak signal-to-noise ratio (PSNR), which is defined by

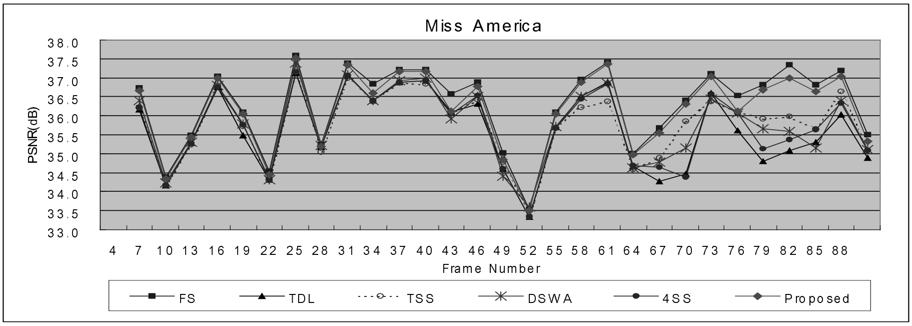
$$PSNR = 10\log_{10}\frac{255^2}{MSE} \quad [dB] \tag{14}$$

$$MSE = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}(I_t(m,n) - \tilde{I}_t(m,n))^2 \tag{15}$$

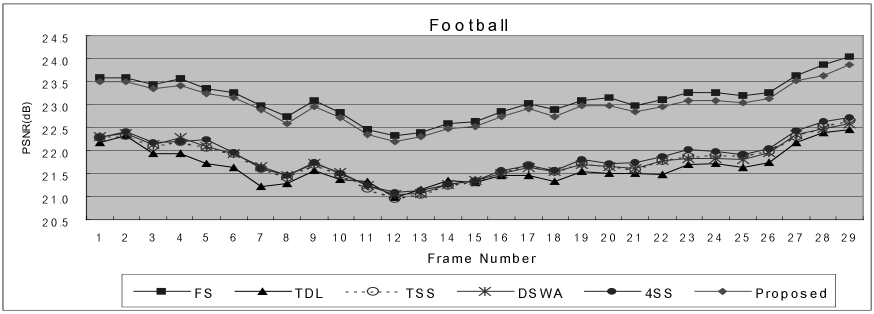
where I_t denotes the original current image frame, and $\tilde{I}_t$ denotes the motion-compensated prediction image frame.

Experimental results with MISS AMERICA and FOOTBALL are depicted in Fig. 9, where the proposed AMS (Adaptive Motion Search) algorithm is compared with FS (full search), TDL (two-dimensional logarithmic search) [1,4], TSS (three-step search)[2,4], 4SS (four-step search) [6], and DSWA (dynamic search window adjust and interlaced search) [7] algorithms. From frame number 60 to 85 in Fig. 9(a), where the image sequence has large motion, the proposed method has good prediction results while the other algorithms fail to estimate large motions. This can be observed in Fig. 9(b), where FOOTBALL sequence has large motions.

Simulation results are summarized in Table 1, where we note that AMS provides better image quality than any other fast BMAs. It also reduces the average number of search points for MISS AMERICA having small motion, while slightly increasing the complexity for FOOTBALL having large motion. The matching criterion defined in Eq. (13) also contributes to reduce the required number of the searching operations.



(a) MISS AMERICA



(b) FOOTBALL

Fig. 9. Performance Comparison

Table 1. Average Performance of Block Matching Algorithms

Algorithm	PSNR(dB)		Average Number of Checking Positions	
	MISSA	FOOTB	MISSA	FOOTB
Full Search	36.29	23.12	225.0	225.0
Three Step Search	35.78	21.77	25.0	25.0
4-Step Search	35.81	21.84	20.9	21.9
2-D Logarithmic Search	35.62	21.63	16.6	19.3
Dynamic Search	35.77	21.78	19.8	17.8
Diamond Search	36.08	21.22	17.0	17.2
Diamond Zonal Search	35.99	22.31	16.1	24.0
Adaptive Motion Search (AMS)	36.17	22.99	14.8	29.7

5 Conclusions

In this paper, we have proposed a new algorithm for fast block matching motion estimation based on an optimal search pattern. By a simple analysis, we have shown that the diamond search pattern is compact and optimal in terms of correlation of rectangular blocks. The proposed motion estimation algorithm employs the derived optimal search pattern and a new matching criterion. We also develop an efficient motion search strategy for the given initial search pattern. Performance improvement compared to other fast BMAs is 0.4 dB for MISS AMERICA, and 1 dB for FOOTBALL. The proposed algorithm reduces computational complexity about 6.5% and 13.2% compared to FS BMA for MISS AMERICA and FOOBALL, respectively. It chooses an adaptive search strategy based on the amount of block difference, and reduces the required computational complexity drastically compared to other fast block matching algorithms, while maintaining good prediction accuracy.

Acknowledgements. This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) through the Ultra-Fast Fiber-Optic Networks (UFON) Research Center at Kwangju Institute of Science and Technology (K-JIST), and in part by the Ministry of Education (MOE) through the Brain Korea 21 (BK21) project.

References

1. Jain, J.R., Jain, A.K.: Displacement Measurement and Its Application in Interframe Image Coding. IEEE Trans. Commun., vol. COM-29 (1981) 1799-1808
2. Srinivasan, R., Rao, K.R.: Predictive Coding Based on Efficient Motion Estimation. IEEE Trans. Commun., vol. COM-33, no. 8 (1985) 888-896
3. Ghanbari, M.: The Cross-search Algorithm for Motion Estimation. IEEE Trans. Commun., vol. COM-38 (1990) 950-953
4. Musmann, H., Pirsh, P., Grallert, H.: Advances in Picture Coding. Proc. IEEE, vol. 73, no. 4 (1985) 523-548
5. Liu, B., Zaccarin, A.: New Fast Algorithm for Estimation of Block Motion Vectors. IEEE Trans. Circuit and Systems for Video Technology, vol.3 (1993) 148-157
6. Po, L.M., Ma, W.C.: A Novel Four-step Search Algorithm for Fast Block Motion Estimation. IEEE Trans. Circuit and Systems for Video Technology, vol. 6, no. 3 (1996) 313-317
7. Lee, L.W., Wang, J.F., Lee, J.Y., Shie, J.D.: Dynamic Search-window Adjustment and Interlaced Search for Block-matching Algorithm. IEEE Trans. Circuit and Systems for Video Technology, vol. 3, no. 1 (1993) 85-87
8. Tham, J.Y., Ranganath, S., Ranganath, M., Kassim, A.A.: A Novel Unrestricted Center-biased Diamond Search Algorithm for Block Motion Estimation. IEEE Trans. Circuit and Systems for Video Technology, vol. 8 (1998) 369-377
9. Zhu, S., Ma, K.: A New Diamond Search Algorithm for Fast Block Matching Motion Estimation. IEEE Trans. Image Processing, vol. 92 (2000) 287-290
10. Tourapis, A., Au, O., Liou, M., Shen, G.: An Advanced Zonal Block Based Algorithm for Motion Estimation. ICIP-99, vol. 2 (1999) 610-614
11. Chiang, T., Sun, H.: Report of Ad hoc Group on Encoder Optimization. ISO/IEC/JTC1/SC29/WG11 MPEG99/ M5528 (1999)
12. Tourapis, A.M., Au, O.C., Liou, M.L.: New Results on Zonal Based Motion Estimation Algorithms-Advanced Predictive Diamond Zonal Search, ISCAS, vol.5 (2001) 183-186