A Fast Block Matching Motion Estimation Algorithm in Video Coding Standards

Mrs.K.P. Joshi   Ms.S.D.Oza   Mrs.P.M.Jadhav

***Abstract* − To enable wireless Internet and other data services using mobile appliances, there is a critical need to support content rich cellular data communication, including voice, text, image services and video. Video Compression has played an important role in Multimedia data storage and transmission. Video compression techniques remove spatial as well as temporal redundancy using intra-frame and inter-frame coding respectively. A large level of compression can be achieved through inter-frame coding. This paper gives new method of the Block Matching Algorithm 'Helical Search Block Matching Algorithm' (HSBMA) used for motion estimation in video compression. It uses centre-biased checking point patterns on 15x 15 window which requires relatively small number of search locations to perform fast block matching. Simulation results are presented which show that the HSBMA algorithm requires less search points than the well-known Exhaustive Search (ES) and Three Step Search (TSS) maintaining the PSNR quality.**

***Keywords*—Block Matching Algorithm, Motion estimation, Video image**

I. INTRODUCTION

Due to limited channel bandwidth and stringent requirements of real-time video playback video coding is an indispensable process for many visual communication applications and always requires a very high compression ratio. The high temporal correlation or redundancy from the compression viewpoint, between adjacent frames in an image sequence, requires be properly identifying and eliminating to achieve this objective. An effective and popular technique to reduce the temporal redundancy, called block-matching motion estimation, has been widely adopted in various video coding standards, such as CCITT (now ITU-T) H.261, H.263, MPEG-1 and MPEG-2. In fact, block matching motion estimation is instrumental to any motion-compensated video coding technique. Therefore, fast algorithms for block matching are highly desirable.

II. BLOCK MATCHING ALGORITHMS

Motion estimation based video compression is to save on bits by sending JPEG encoded difference images which inherently have less energy and can be highly compressed as compared to sending a full frame that is JPEG encoded. This paper implements and evaluates the fundamental block matching algorithms Full Search (FS), Three Step Search (TSS) and proposed Algorithm Helical Search Block Matching Algorithm. The idea behind block matching is to divide the current frame into a matrix of 'macro blocks' that are then compared with corresponding block and its adjacent neighbors in the previous frame to create a vector that stipulates the movement of a macro block from one location to another in the previous frame. This movement calculated for all the macro blocks comprising a frame, constitutes the motion estimated in the current frame. The search area for a good macro block match is constrained up to p pixels on all four sides of the corresponding macro block in previous frame. This 'p' is called as the search parameter. Larger motions require a larger p, and the larger the search parameter the more computationally expensive the process of motion estimation becomes. Usually the macro block is taken as a square of side 16 pixels, and the search parameter p is 7 pixels.

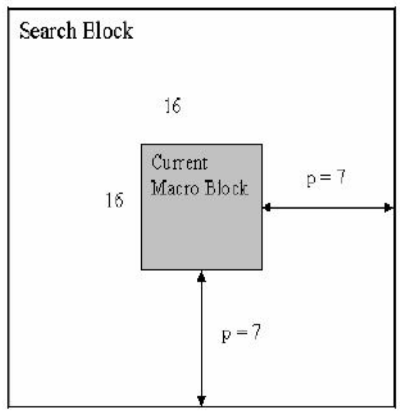

Fig. 1. Macro-block of 16x16 pixels and search range p = 7

The idea is represented in Fig. 1. The matching of one macro block with another is based on the output of a cost function. The macro block that results in the least cost is the one that matches the closest to current block. There are various cost functions, of which the most popular and less computationally expensive is Mean Absolute Difference (MAD) given by Eqn. (1).Another cost function is Mean Squared Error (MSE) given by Eqn. (2).

$$MAD = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|C_{ij} - R_{ij}\right|$$

………..………………………. Eqn. (1)

$$MSE = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left(C_{ij} - R_{ij}\right)^2$$

………………………………. Eqn. (2)

where N is the side of the macro bock, Cij and Rij are the pixels being compared in current macro block and reference macro block, respectively. Peak-Signal-to-Noise-Ratio (PSNR) given by Eqn.(3)characterizes the motion compensated image that is created by using motion vectors and macro clocks from the reference frame.

$$PSNR = 10 \log_{10}\left(\frac{255^2}{MSE}\right)$$

…………………………………Eqn. (3)

A. FULL SEARCH (FS) ALGORITHM

The Full Search algorithm also known as Exhaustive Search (ES) performs an exhaustive search over all possible points of the search area. Thus, for a search range p=7,total 225 checking points must be tested .As a result of which it finds the best possible match and gives the highest PSNR amongst any block matching algorithm. Fast block matching algorithms try to achieve the same PSNR doing as little computation as possible. The obvious disadvantage to ES is that the for larger search window it requires more computations.

B. THREE STEP SEARCH (TSS) ALGORITHM

It contains three steps of search as it is illustrated in Fig.2. In the first step nine points are checked. In the second step eight points are checked, surrounding the minimum found in the first step. In the third step eight points are checked surrounding the minimum of the second step. Thus 25 points are checked in the TSS algorithm.

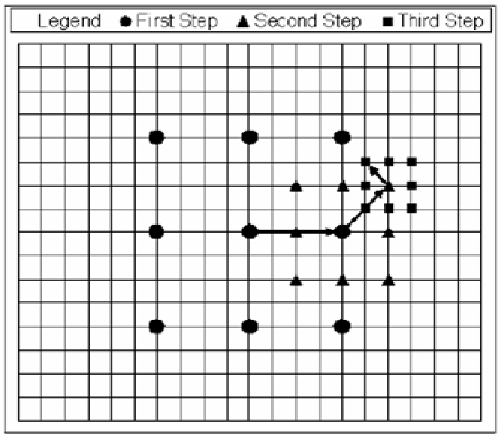

Fig. 2. Path for Convergence of Three Step Search

C. HELICAL SEARCH (HS) ALGORITHM

Proposed Algorithm uses two boundaries parallel to central vertical axis at a distance of four pixels from center at (P/2+ 1) i.e. at search points -4 and +4 as. Initial step comprises 5 checking points from which four points surround the center one to compose a zigzag shape (z). If the minimum Block Distortion Measure (BDM) point is found at the search window center then move the small 3 x 3 search window with 8 points around the center and find the new BDM, is the winning point found. If the Minimum BDM point is located above or below the center point on vertical axis,5 additional checking points are picked. If the Minimum Block BDM point is located diagonally on the helical boundaries then 8 points around the new center at distance=1 of the search are examined. The search window is 3x3 around the minimum BDM point found in step 2 and direction of the overall motion vector is taken as shown in Fig. 3.

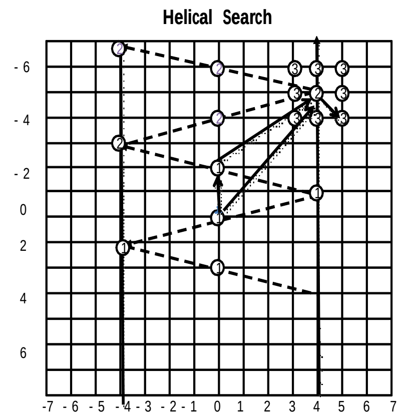

Fig. 3. Path for Convergence of Helical Search

The total number of checking points will vary from a minimum of 13 (5+8) to 18 (5+5+8) points in worst case. Pictorial demonstration of search path is provided by the example shown in Fig. 3.

III. SIMULATION RESULTS AND COMPARISON

In our simulation experiments, the block size is fixed at 16 x 16. Block matching is conducted within a 15 x 15 search window on full-pixel basis, i.e., the maximum block displacement in horizontal or vertical direction is +/-7 pels. Mean absolute distance (MAD) is used as the matching criterion to reduce the block-matching computation in practice for reduced computation. Four test image sequences with large, moderate or small motion are exploited in the experiments: the **Akiyo** sequence(QCIF-176x144,Low motion), the **Foreman** sequence(CIF-352x288, Moderate motion),the **BUS**(QCIF-176x144, High motion) sequence and the **Container** (CIF-352x288, Low motion).The performance of an algorithm is a compromise between the PSNR computed and the Average Search Point needed by each algorithm. The following table summarizes the results of simulation of the four sequences for PSNR expressed in decibel (dB) and the Average Search Point. Figures 4(a), 4(b), 5(a), 5(b), shows PSNR and Average search points for the sequences Akiyo and Bus respectively. The results obtained with the first 100 frames for four video sequences are presented in Table 1.

Table 1 : Comparison of PSNR for four video sequences

Video file	Full Search	Three-Step Search	Prop. Algo.-Helical Search
	PSNR	PSNR	PSNR
Akiyo	42.7441	42.7441	42.7441
Bus	23.3057	21.6035	20.1923
Forman	31.8736	31.2720	30.6841
Container	37.1941	37.1935	37.1928

Results show that the mean PSNR values for first 100 frames of the low-motion videos such as 'Akiyo', Helical Search gives best performance similar to Full Search and Three Step Search, in case of video sequence 'Container' PSNR of HS degrades by 0.0013db with FS and by 0.0007db with TSS. For Moderate and High motion videos PSNR of HS degrades more compare to low motion video sequences. Results show that proposed Algorithm HS is 16 times faster than FS and 1.9 times faster than TSS.

Table 2 : Comparison of Average Search Point for four video sequences

Video file	Full Search	Three-Step Search	Prop. Algo.-Helical Search
	Avg. search Pt.	Avg. search Pt.	Avg. search Pt.
Akiyo	184.5556	21.4848	11.2275
Bus	204.2828	23.5869	12.6097
Forman	204.2828	23.2901	13.0140
Container	204.2828	23.2243	12.1148

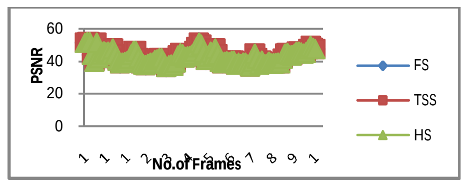

Fig. 4(a). PSNR for Akiyo

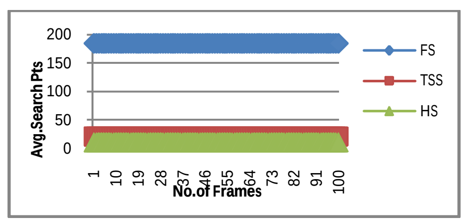

Fig. 4(b) Average search point per frame for Akiyo

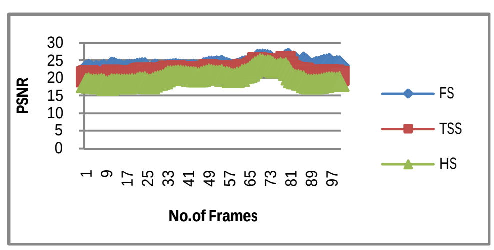

Fig. 5(a) PSNR for Bus

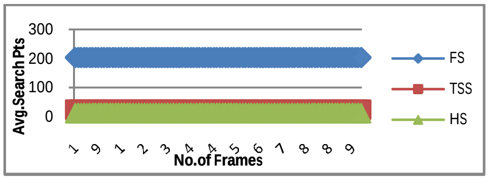

Fig. 5(b). Average search point per frame for Bus

IV. CONCLUSION

In this paper, a new search technique is developed to perform fast Motion Estimation. A high reduction in computational complexity is achieved by using this proposed technique. Performance of the implemented Helical search is compared against Full Search and Three Step Search Block Matching Algorithms. Results for Peak Signal to Noise Ratio (PSNR) and average number of search points per frame are obtained through experimental evaluation. It shows that the proposed Helical Search algorithm reducing its computation by up to 17% approximately compare to Full Search and 47% compare to TSS. Processing speed can be improved significantly by using this proposed approach while maintaining comparable image quality for different video sequences. It is observed that average number of search points required for proposed algorithm is comparatively much less than other two algorithms. Simulation experiments conducted demonstrate that the proposed HS algorithm works better for low and medium motion video sequences than fast motion videos.

V. REFERENCES

[1] Feisul Hadi Jamil, Rosalyn R. Porle, Ali Chekima, Razak Ali Lee, Hayder Ali, Sukhairi , “Preliminary Study of Block Matching Algorithm (BMA) for Video Coding ”4th International Conference on Mechatronics (ICOM), 17-19 May 2011, Kuala Lumpur, Malaysia 978-1-61284-437-4/11/ ©2011 IEEE

[2] Jau-Ling Chen and Pei-Yin Chen, “A new search algorithm for block motion estimation”, IEEE transactions on circuits and systems for video technology, 2000.pp 979-982.

[3] Xuan Jing and Lap-Pui Chau, Senior Member, IEEE, “An Efficient Three-Step Search Algorithm for Block Motion Estimation” IEEE Transactions on Multimedia, Vol. 6, No. 3, June 2004.

[4]Lai-Man Po, and Wing-Chung Ma, “A Novel Four-Step Search Algorithm for Fast Block Motion Estimation”, IEEE Trans. Circuits And Systems For Video Technology, vol 6, no. 3, pp. 313-317, June 1996.

[5] S K Satyanarayana, M Kavya, Dr. V S K Reddy Department of Electronics and Communication Engineering, “Predictive Search Range and Block size in Block Matching Motion Estimation Algorithms in Video Coding,”978-1-4244-4131-0/09/$25.00 ©2009 IEEE

[6] ZHOU Wei1 SHI Hao Shan1 DUAN Zhe Min1 ZHOU Xin, “An Efficient Block-matching Motion Estimation Algorithm for H.264 Video Coding Standard”1-4244-0517-3/06/$20.00 ©2006 IEEE

[7] Iain E. G. Richardson, Video Codec Design, West Sussex: John Wiley & Sons Ltd., 2002

[8] J. R. Jain and A. K. Jain, “Displacement measurement and its application in interframe image coding,” IEEE Trans. Commun., Vol. COM-29, pp. 1799-1808, Dec. 1981.