A Potential Heuristic-based Block Matching Algorithms for Motion Estimation in Video Compression

Siti Eshah Che Osman[1], Hamidah Jantan[2], and Mohamad Taib Miskon[3]
[1]*Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA (UiTM), 40450 Shah Alam, Selangor, Malaysia.*
[2]*Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA (UiTM), 23000 Dungun, Terengganu, Malaysia.*
[3]*Faculty of Electrical Engineering, Universiti Teknologi MARA (UiTM), 23000 Dungun, Terengganu, Malaysia.*
cteshahco@gmail.com

***Abstract*— Motion estimation (ME) is one of the element keys in video compression that takes up to 60% in processing time. Block matching algorithm (BMA) is a technique that is used to reduce the computational complexity of ME algorithm due to its efficiency and good performance. Strategy of searching is one of the factors in developing motion estimation algorithm that has the potential to provide good performance. This study aims to implement several selected BMAs for achieving the least number of computations and to give better Peak Signal to Noise Ratio (PSNR) values using different video sequences. The proposed algorithms are modified based on the search strategy adapted from the standard algorithms approach. The results have proved that both modification algorithms (MDS and MARPS) have the potential in reducing the number of computations and achieved good PSNR values in all motion types as compared to DS and ARPS respectively. This work could be improved by using meta-heuristic algorithms approach such as particle swarm optimization (PSO), artificial bee colony (ABC), tabu search (TS) and etc to provide the better result of PSNR values without increasing the number of computations.**

***Index Terms*— Block matching algorithm; Motion estimation; Number of computations; PSNR.**

I. INTRODUCTION

The processing of image sequences have become very challenging and valuable because analyzing the motion of natural moving objects is difficult [1, 2]. In video compression process, motion estimation (ME) reduces the temporal redundancy that commonly exists in a video sequence by identifying the match for a macro block in the current frame with another one in the reference frame(s) [3-5]. In an effort to reduce the computational complexity of ME algorithms in video coding, several methods have been presented by many researchers such as parametric-based models, percussive techniques, optical flow and block matching algorithms [6, 7]. Block matching algorithm is chosen mainly due to its efficiency, simplicity and good performance for both software and hardware implementations [8].

The most basic block matching motion estimation algorithm is Full Search (FS). Then, various fast search algorithms are developed to reduce the search time; hence to increase the processing speed and at the same time maintaining the quality of the reconstructed signals of FS [9]. Three Step Search (TSS) [3], New Three Step Search (NTSS) [10], Simple and Efficient Search (SES) [11], Four Step Search (4SS) [12], Diamond Search (DS) [13] and Adaptive Rood Pattern Search (ARPS) [14] are the well-known fast search algorithms. There are three main factors in developing the motion estimation algorithms which are selection of search pattern, search strategy and initial center and all these factors affecting to the computational performance and PSNR [2, 9].

Recently, optimization based on heuristic and meta-heuristic algorithms are used for solving optimization problems in several fields but there is no guarantee that optimal solutions are reached. While the heuristic algorithms are designed to find the optimal solution with decision maker's expertise, experience and discovery by trial and error, the meta-heuristics tend to be more intelligent and adaptive [15]. In this paper, Exhaustive Search (ES) or Full Search, Diamond Search (DS) and Adaptive Rood Pattern Search (ARPS) also two modification methods of DS and ARPS are discussed. Modification is made based on the search strategy that was adapted from standard algorithms. The algorithms involved were implemented and evaluated by using different video sequences. The effectiveness of the proposed heuristic-based of modified algorithms were evaluated to test the performances from the view point of computational complexity as well as estimation accuracy.

The organization of this paper is as follows: Section II explains the related work on block matching motion estimation and then followed by matching criteria involved as well as clarifying block matching algorithms. Section III explains the proposed modified algorithms; Section IV describes experiment setup; Section V disclosed the result and discussion; finally conclusion and future work in Section VI.

II. RELATED WORK

A. Block Matching Motion Estimation

The block matching motion estimation plays a very important role in all video coding standards such as H261, H.263, H.264 MPEG1, MPEG2 and MPEG4 [9, 16]. Motion estimation is defined as searching the best motion vector, which is the placement of the coordinate of the best similar block in reference frame for the block in current frame [11]. Block matching (BM) is a technique used to compare images taken at two different time frames and to estimate the direction motion of two frames [17]. In a BM approach, a frame is segmented into N×N non overlapping blocks and each block in the current frame is matched with candidate blocks of size N×N within the search area in the reference frame in the same coordinates. The number of total blocks that needs to be processed in each frame will decrease and the computational complexity decreases if the number of N is increased [2]. Generally, the search area is decided by the 'search range parameter', p, for a good macro block match, where p is the number of pixels on all four sides of the corresponding macro block in the previous frame. The larger the value of p, the larger is the potential motion and thus more computational power. The standard inputs are a macro block of size 16 pixels and a search parameter of p = 7 pixels [6].

Block matching motion estimation algorithm can be approached as an optimization problem that will achieve fast transmission and reduce data storage because motion estimation based video compression helps in reducing bits by sending encoded images which have less data rather than sending original frame [18, 19].

B. Matching Criteria

Block matching plays a major role in image matching to improve the efficiency. The goal of image matching is to determine the similarity between the images and portions of images. The similarity measure is a key element in the matching process. In order to find out the best matching block within a search window from the previous frame, some matching criteria are considered. The micro block that results in the minimum cost is the one that matches closely to current block.

There are numerous matching criteria or distortion function that have been proposed such as Mean Absolute Difference (MAD), Mean Squared Error (MSE) and Peak Signals to Noise Ratio (PSNR), which are represented in (1-3) [2, 16, 17]. MSE is evaluated between original frame and reconstructed frame or between current block by the motion vectors. MAD is most commonly used as its computation cost is low and also due to its simplicity. MSE is given by (1) and MAD is given by (2).

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

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

PSNR given in (3) is the most popular and it determines the motion compensated image that is created by using motion vectors and macro blocks from the reference frame. The accuracy of motion estimation can be measured using PSNR.

$$PSNR = 10Log_{10}\left[\frac{255^2}{MSE}\right] \tag{3}$$

All the above matching criteria are shown, where N is the side of the macro block, Cij and Rij are the pixels being compared in the current macro block and the reference macro block, respectively.

C. Block Matching Algorithms

Recently, there have been several block matching algorithms proposed for motion estimation. The details of the block matching algorithms involved are discussed in following sections.

i. Exhaustive Search (ES)

The Exhaustive Search (ES) algorithm, also known as Full Search, is a very simple method for motion estimation [11]. Besides, it is by nature a brute force algorithm and involves a high computational cost [12]. In ES, the correlation window is moved to each candidate position within the search window in searching for the best match. There are a total of (2p+1)×(2p+1) positions that need to be examined, where p is the search range for the block. In this algorithm, the cost function is calculated at each possible location within the search window. The smallest distortion gives the best match. The best match is found and it provides the best quality with the highest PSNR values amongst any block matching algorithms. It is unsuitable for real-time video coding due to the lengthy computation time.

ii. Diamond Search (DS)

Diamond search (DS) is developed to solve computational complexity of ES. It uses the same basic principle as 4SS but the pattern of search point is changed from a square to a diamond, and there is no limit on the number of steps that this algorithm can take [1, 11, 20]. There are two different types of DS algorithm; Large Diamond Search Pattern (LDSP) and Small Diamond Search Pattern (SDSP). Just like in FSS, the first step is to use LDSP. Start the search location at the center with the step size = 2 and it will check eight points around the center in horizontal and vertical direction. If the minimum weight is found at the center, the process stops and go to SDSP step. It will save more computational time. Otherwise, set the new point of origin to this location and repeat LDSP. The last step uses SDSP around the new search origin with the step size=1 then searching of four points around the center. The search procedure is repeated until the minimum distortion point to declare as the best match is found. As the search pattern is neither too small nor too big and the fact that there is no limit to the number of steps, this algorithm can find global minimum very accurately. However, when the movement in the video is very high, it takes some LDSP steps to converge which can cause increasingly in number of total searching points.

iii. Adaptive Rood Pattern Search (ARPS)

The most important feature of ARPS is that most of the time the general motion in a frame is usually coherent [2, 14]. It means that if the macro block around a current micro block moved in a particular direction then it highly probable that neighboring block share similar motion vectors (MVs). This algorithm uses the motion vector of the macro block to its immediate left to predict its own motion vector. The coordinate of the predicted motion vector for the block is checked by setting up the step size=max (|X|,|Y|), where (X,Y) is the coordinate. The main advantage of this algorithm over DS is if the predicted MV is (0, 0), it directly starts using SDSP and does not waste computational time in doing LDSP. Besides, if the predicted motion vector is far away from the center, then again ARPS saves on computations by directly jumping to that vicinity and using SDSP, whereas DS takes its time doing LDSP. The search strategy of APRS can be enhanced to improve the performance of block matching further. Besides, the number of searching points in the initial step in algorithm can also be reduced [21].

III. Proposed Modified Algorithms

Generally, DS and ARPS are selected because both algorithms provide PSNR values which are very close to those of ES and both have significantly less computational cost as compared to other well-known methods. A modification of DS and ARPS are made based on the search strategy, called Modified DS (MDS) and Modified ARPS (MARPS), are proposed. There are many possibilities of searching strategy could be approached with the different initial checking points direction that will affect the performances. These algorithms are implemented which aim to reduce the number of computations and at the same time to maintain estimation accuracy.

A. Modified Diamond Search (MDS)

As compared to the DS, most of the steps are the same except the initial checking point direction. The initial checking points of DS are (i,j-2)(i-1,j-1) (i+1,j-1) (i-2,j) (i,j) (i+2,j) (i-1,j+1) (i+1,j+1) and (i,j+2), where (i,j) is the center,(0,0) [20]. In this proposed MDS algorithms, a new searching strategy is approached with initial checking points based on (i,j) (i,j+2) (i-2,j) (i-1,j+1) (i-1,j-1) (i+1,j-1) (i,j-2) (i+2,j) and (i+1,j+1), where (i,j) is the center,(0,0). It has been experimentally proven that the proposed MDS shows a good performance in terms of number of computation as compared to that of DS in high motion sequence.

B. Modified Adaptive Rood Pattern Search (MARPS)

As compared to the ARPS which uses initial checking point based on (i,j-1)(i-1,j) (i,j) (i+1,j) and (i,j+1), where (i,j) is the center [20]. In proposed MARPS algorithms, a new searching strategy is approached with initial checking points are (i,j) (i-1,j) (i,j+1) (i+1,j) and (i,j-1), where (i,j) is the center (0,0). This proposed algorithm has been experimentally proven that it has the potential to reduce the computational complexity and provides good PSNR values in all motion types as compared to ARPS.

IV. Experiment Setup

In this study, three video sequences were used for performance comparison of different algorithms. Each algorithm has been implemented in MATLAB. Three video sequence standards are formatted in Quarter Common Intermediate Format (QCIF) (176×144). Each of the video sequences represents the various types of motion which include Foreman for the high motion, Hall Monitor for less background motion (medium motion) and Claire for the small motion. The first 100 frames of the video sequences were used for testing. The block size is considered as 16×16 pixels and the search parameter, p=±7.

The measurement of computational complexity is based on the search efficiency by counting the average number of computations. The least average number of computations shows the fastest algorithm. In addition, as an alternative index, the computational complexity degradation ratio (D_{COMP}) is used in comparison as shown in (4). This ratio is expressed in percentage (%) between the number of computations of the proposed modified algorithms (MDS and MARPS) and DS and ARPS as a reference respectively. If the result shows the highest D_{COMP} percentage value, the proposed modified algorithms are considered as the fastest algorithms.

$$D_{comp} = -\left(\frac{comp\_complexity_{DS} - comp\_complexity_{MDS}}{comp\_complexity_{DS}}\right)x100\% \qquad (4)$$

The estimation accuracy is characterized by the PSNR by counting the average of the PSNR values. The higher the average values of PSNR yield the better quality of the compensated image. The PSNR degradation ratio (D_{PSNR}) is also used in the comparison as shown in (5). This ratio is expressed in percentage (%) between the PSNR of the proposed modified algorithms (MDS and MARPS) and DS and ARPS as a reference respectively. The least D_{PSNR} percentage values yields the better quality of the compensated image.

$$D_{PSNR} = -\left(\frac{PSNR_{DS} - PSNR_{MDS}}{PSNR_{DS}}\right)x100\% \qquad (5)$$

This paper presents experimental results to evaluate the performances of the block matching algorithms which are ES, DS, ARPS and also the proposed modified algorithms, MDS and MARPS from the view point of computational complexity as well as prediction accuracy. Table 1, Table 2 and Table 3 show the results of average number of computations and average of PSNR values with the distance, D=2 between the current frame and the reference frame by using Foreman (High motion), Hall_Monitor (Medium motion) and Claire (Low motion) video sequences respectively. ES shows the highest average number of computations. However, ES gives the best PSNR values as compared to other algorithms and DS comes quite close to those of ES. MARPS has the least average number of computations as compared to other algorithms for every motion types. MDS and MARPS give very close values of PSNR to those of DS and ARPS respectively for all motion types. In all motion types, the proposed modified algorithms still cannot gain high average PSNR values over ES but both have provided very close average PSNR values to those of ES.

Table 1
Performance of Block Matching Algorithms in terms of no. of Computations and PSNR of High Motion Type

Algorithms	Avg. no of Computations	Avg. PSNR
ES	191.2	28.27
DS	16.14	28.02
ARPS	9.30	27.82
MDS	14.92	27.68
MARPS	7.63	27.09

Table 2
Performance of Block Matching Algorithms in terms of no. of Computations and PSNR of Medium Motion Type

Algorithms	Avg. no of Computations	Avg. PSNR
ES	191.20	31.40
DS	12.43	31.30
ARPS	5.97	31.24
MDS	12.41	31.24
MARPS	4.85	31.14

Table 3
Performance of Block Matching Algorithms in terms of no. of Computations and PSNR of Low Motion Type

Algorithms	Avg. no of Computations	Avg. PSNR
ES	191.2	36.94
DS	12.36	36.93
ARPS	5.90	36.90
MDS	12.31	36.90
MARPS	4.76	36.10

Table 4 shows the percentage degradation of the number of computations and PSNR values of the proposed MDS and MARPS algorithms as compared to those of DS and ARPS algorithms respectively for all motion type (H, M and L represent for high, medium and low motion type respectively). The sign (-) in D_{COMP} and D_{PSNR} indicate a loss values in their performances. MDS has reduced the number of computations by 7.56% for high motion sequence, 0.16% for medium motion sequence and 0.40% for low motion sequence. MARPS is considered as the fastest algorithm because it has greatly reduced the number of computations by 19.32%, 18.76% and 17.96% for low, medium and high motion sequence respectively. MDS and MARPS also provide good PSNR values with small degradation ratio for all motion types.

Figure 1 and Figure 2 show the graph performance comparisons of the proposed modified algorithms as compared to DS and ARPS in terms of number of computation values based on 100 number of frames for Foreman video sequences (high motion) and Claire video sequence (low motion) respectively. Both sequences are the most affected performances by implementing the proposed modified algorithms since they have greatly reduced the number of computation values as compared to other sequences.

Table 4
D_{COMP} and **D_{PSNR}** Comparison of the MDS and MARPS Algorithms for all Motion Types

Algorithms	D_{COMP} (%)			D_{PSNR} (%)		
	H	M	L	H	M	L
MDS	-7.56	-0.16	-0.40	-1.21	-0.19	-0.08
MARPS	-17.96	-18.76	-19.32	-2.62	-0.32	-2.17

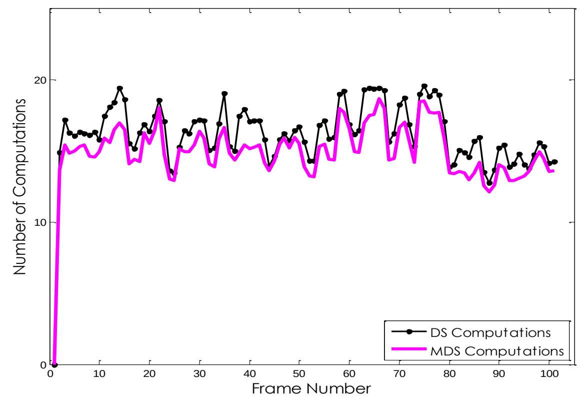

Figure 1: Performance Comparisons of DS and MDS Algorithms in term of Number of Computations of Foreman Sequence

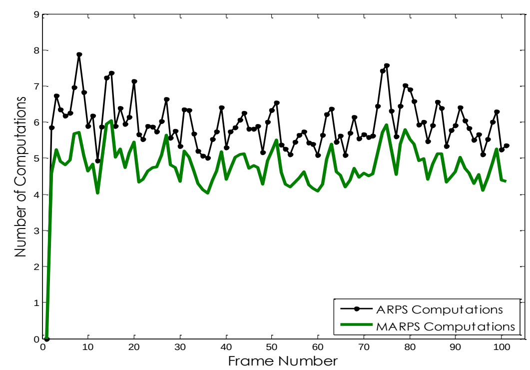

Figure 2: Performance Comparisons of ARPS and MARPS Algorithms in term of Number of Computations of Claire Sequence

V. CONCLUSION & FUTURE WORK

Block matching algorithm (BMA) is the most popular and efficient motion estimations in video compression. The simplest algorithm for BMA is ES or full search algorithms which can give the best matching quality but high in computational complexity. Fast block matching algorithms have been developed to solve ES problem. This paper has conducted an experiment to evaluate the performances in terms of computational complexity and estimation accuracy of ES, DS, ARPS, MDS and MARPS algorithms. These algorithms have implemented using three different sequences which are Foreman, Hall_Motion and Claire video sequences.

Based on the results, MDS and MARPS show better performance due to their potential in reducing the computational complexity and give close PSNR values as compared to others. This work could be improved by using meta-heuristic algorithms approach such as particle swarm optimization (PSO), artificial bee colony (ABC), tabu search (TS) and etc to provide the better result of PSNR values without increasing the number of computations.

ACKNOWLEDGMENT

We wish to acknowledge the support of Ministry of Education (MOE) to Universiti Tekonologi MARA (UiTM) in Research Acculturation Grant Scheme (RAGS) (600-RMI/RAGS 5/3 (205/2013).

REFERENCES

[1] Khammar, M.R., "Evaluation of different block matching algorithms to motion estimation," *International Journal of VLSI and Embedded Systems (IJVES)*, vol. 3, pp. 148 - 153, 2012.

[2] Yaakob, R., Aryanfar, A., Halin, A.A., Sulaiman, N., "A Comparison of Different Block Matching Algorithms for Motion Estimation," *The 4th International Conference on Electrical Engineering and Informatics (ICEEI 2013)* vol. 11, pp. 199 - 205, 2013.

[3] Acharjee, S., Chaudhuri, S.S., "Fuzzy Logic Based Three Step Search Algorithm for Motion Vector Estimation," *International Journal of Image, Graphics and Signal Processing*, vol. 2, pp. 37 - 43, 2012.

[4] Babu, R.V., Tom, M., Wadekar, P., "A survey on compressed domain video analysis techniques," *Multimedia Tools and Applications*, 2014.

[5] Pinninti, K., P.V.Sridevi, "Motion Estimation in MPEG-4 Video Sequence using Block Matching Algorithm," *International Journal of Engineering Science and Technology (IJEST)*, vol. 3, pp. 8466 - 8472, 2011.

[6] Philip, J.T., Samuvel, B., K, P., K, N.N., "A Comparative Study of Block Matching and Optical Flow Motion Estimation Algorithms," *In: Annual International Conference on Emerging Research Areas: Magnetics, Machines and Drives (AICERA/iCMMD)*, pp. 1 - 6, 2014.

[7] Cuevas, E., Zaldívar, D., Pérez-Cisneros, M., Oliva, D. "Block matching algorithm based on Differential Evolution for motion estimation," *Engineering Applications of Artificial Intelligence*, vol. 26, pp. 488 - 498, 2013.

[8] Hadi, I., Sabah, M., "A Novel Block Matching Algorithm Based on Cat Swarm Optimization for Efficient Motion Estimation," *International Journal of Digital Content Technology and its Applications(JDCTA*) vol. 8, pp. 33 - 44, 2014.

[9] Pandian, S.I.A., G.JoseminBala, J.Anitha, "A Pattern based PSO Approach for Block Matching in Motion Estimation," *Engineering ApplicationsofArtificial Intelligence*, vol. 26, pp. 1811–1817, 2013.

[10] Choudhury, H.A., Saikia, M., "Comparative Study of Block Matching Algorithms," *International Journal of Advanced Computational Engineering and Networking* vol. 1, pp. 73 - 78, 2013.

[11] D.V.Manjunatha, Sainarayanan, "Comparison and Implementation of Fast Block Matching Motion Estimation Algorithms for Video Compression," *International Journal of Engineering Science and Technology (IJEST)* vol. 3, pp. 7608 - 7613, 2011.

[12] Santamaria, M., Trujillo, M., "A Comparison of Block-Matching Motion Estimation Algorithms," *In: 7th Colombian Computing Congress (CCC)*, pp. 1 - 6, 2012.

[13] Nayak, A., Biswal, B., "Evaluation and Comparison of Motion Estimation Algorithms for Video Compression," *I.J. Image, Graphics and Signal Processing*, vol. 10, pp. 9 - 18, 2013.

[14] Fazli, S., Tavakkoli, L., "A Fast Adaptive Rood Pattern Algorithm for Video Compression," *International Journal of advanced studies in Computer Science and Engineering IJASCSE* vol. 3, pp. 13 - 15, 2014.

[15] Suh, W.-J., Park, C.-S., Kim, D.-W., "Heuristic vs Meta-Heuristic Optimization for Energy Performance of a Post Office Building," *12th Conference of International Building Performance Simulation Association*, pp. 704 - 711, 2011.

[16] Hadi, I., Sabah, M., "Enhanced Hybrid Cat Swarm Optimization Based on Fitness Approximation Method for Efficient Motion Estimation.," *International Journal of Hybrid Information Technology* vol. 7, pp. 345 - 364, 2014.

[17] Sorkunlu, N., Sahin, U., Sahin, F., "Block Matching with Particle Swarm Optimization for Motion Estimation," *In: IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1306 - 1311, 2013.

[18] George, N.P., J.Anitha, "Motion estimation in video compression based on Artificial Bee Colony," *In: 2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, pp. 730 - 733, 2015.

[19] Pal, M., "An Optimized Block Matching Algorithm for Motion Estimation using Logical Image," *In: International Conference on Computing, Communication and Automation (ICCCA2015)*, pp. 1138 - 1142, 2015.

[20] Barjatya, A., "Block Matching Algorithms For Motion Estimation," *ECE Dept*, pp. 1 - 6, 2004.

[21] Nayak.B, A., E.Nagabhooshnam, "A Study of Efficient Block Matching Algorithms For Real-Time Video Compression Applications," *In: International Conference on Electronics and Communication Systems (ICECS)*, pp. 1 - 5, 2014.