ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 1, July 2014

New Predictive Block Matching Searching Algorithms and Hybrid Predictive Search System

[1]Zainab J. AHMED, [2]Loay E.GOERGE, [3]Bushra A. SULTAN
Dept. of Computer Science, Baghdad University, College of Science, Baghdad, Iraq

*Abstract— **In this paper, two new simple, fast and efficient block matching algorithms are introduced, both methods begins blocks matching process from the image center block and moves across the blocks toward image boundaries. With each block, its motion vector is initialized using linear prediction that depending on the motion vectors of its neighbor blocks that are already scanned and their motion vectors are assessed. Also, a hybrid mechanism is introduced, it depends on mixing the proposed two predictive mechanisms with Exhaustive Search (ES) mechanism in order to gain matching accuracy near or similar to ES but with Search Time ST less than 80% of the ES. Also, it offers more control capability to reduce the search errors. The experimental tests showed that the two proposed search methods outperform the famous three step search algorithm.***

Index Terms— **Motion estimation, Block Matching Algorithm, Predicted Motion Vector, Exhaustive Search, TSS.**

I. INTRODUCTION

The increasing demand to incorporate video data into telecommunication services, corporate environment entertainment industry, and even at home had made digital video technology a necessity. The problem, however, is that the still image and digital video data rate are very large, typically within range around 150 Mp/sec. Data rate of this magnitude requires lot of band width, storage and computing resource in the typical personal computer, for this reason, many video compression standards have been developed to eliminate picture redundancy [1].

In a video scene, data redundancy arises from spatial, temporal and statistical correlations exist between (beside within) frames. These correlations are processed separately because of differences in their characteristic.

Motion estimation and compensation are used to reduce the existing temporal redundancy between successive frames in the time domain. It examines the movement of objects in an image sequence in order to determine the motion vectors representing the estimated motion [2]. Block based motion estimation methods are the most popular methods for motion estimation because of their simplicity and ease of implementation. The matching is, seemingly, used by video compression standards because it can achieve a good balance between complexity and coding efficiency [3].

During block matching, each target block of the current frame is compared with the blocks belong to previous frame in order to find the best match block. Block Matching Algorithms (**BMAs)** calculate the best match distance (i.e., Sum of Absolute Difference, SAD).

Various types of **BMA** have been introduced in the literature. Some of them take wide place in use and application; especially in video coding standard schemes. Exhaustive search (ES) is one of the well-known **BMA**; because it is the most accurate and straightforward approach of **BMA** methods**.** It exhaustively evaluates all possible blocks over the determined search window to find the best match, but this method has too high computational cost which makes the frame coding process too slow [4].

Three Step Search (TSS) algorithm is widely adopted because it is simple, effective and fast algorithm. However TSS uses uniformly allocated search pattern in the test step, which is not very efficient to catch small motions appearing in stationary or quasi-stationary motions. To overcome this problem several adopted techniques have been suggested to make the search process more acceptable to motion scale and uncertainty [5]. A New Three Step Search (NTSS) algorithm was proposed by Li et al in which the search pattern in each step is fixed and no thresholding operations are involved in this algorithm. The main feature of NTSS is that it employs a center- biased checking point pattern in the first step, which is derived to make the search adaptive to motion vector distribution [6]. Also, the halfway-step technique is utilized to reduce the computation cost.

Four Step Search (FSS) [7], Diamond Search (DS) [8], Hexagone-based Search (HEX) [9], and UMHexagons (unsymmetrical-cross-multi-hexagon grid) search [10] are algorithms that have formally adopted to reduce computational complexity by using relatively few points without degrading image quality.

The first One-Time Search (OTS) Block motion estimation was proposed in 1985 by Srinivasan et al [11]. A modified version of OTS based on wavelet transform was proposed to improve OTS algorithm performance [12]. In [13] a new modified OTS (MOTS) proposed, it is based on motion prediction. Its horizontal, vertical and diagonal search scan strategy starts from the center of the frame instead of starting from one the frame's corners.

A fast motion estimation algorithm is proposed in [14], the main features of the proposed algorithm are the motion vectors are classified as the unchanged, slightly changed and the greatly change motion vector according to corresponding local motion similarity. Different search scheme are adopted for different kinds motion vectors.

In this paper, an efficient predictive search system is proposed, the targets "better prediction, small search scan error and transmission bit rate" have been taken into consideration during the design phase. Two simple, efficient and fast block matching search algorithms are given. The first algorithm is called 8-Neighbor Search and the second called 4-Neighbor Search.

The rest of this paper is organized as follows: Section II explains the behavior of the predictive search strategy. Section III describes the steps of proposed search algorithms, including the hybrid search mechanism that mixing them with the Exhaustive Search. The established proposed system is tested using some commonly used videos. The test results are listed and discussed in Section IV, finally the derived conclusions are given in section V.

II. BLOCK MATCHING TECHNIQUE

All block matching technique are proposed to achieve one or more of the three objectives. They are (i) to reduce the computational complexity, (ii) get true motion (high quality) and (iii) to reduce the bit rate (higher compression ratio).

The search process started from the core block (i.e., the block at the center of frame) to be searched using the previous frame. The main idea of the proposed predictive search system is to use the determined motion vectors of the already searched blocks to predict the best search area of the unsearched neighbor block. The prediction depends on the number and positions of the previously searched blocks taken into consideration to do prediction. To cover whole prediction cases the following three prediction cases were taken into consideration:

Case 1: when there is only one previously searched block (i.e., core block), then the initial search position of any new unsearched adjacent block, in all directions (left, right, top and down), is assigned the determined shifts (Δx, Δy) of the core block;

$$\Delta x(r,c) = \Delta x(r-1,c) \qquad (1)$$
$$\Delta y(r,c) = \Delta y(r-1,c) \qquad (2)$$

These shift values are considered as the predicted block displacement along the horizontal and vertical directions, and the search process is started from this predicted position as clarified in Figure (1).

Case 2: If the number of searched blocks on previous row of the current tested block is greater than four, and when there are many previously searched blocks adjacent to the current tested block from left (or right) and other blocks lay at the upper (or lower) side; then seven of the closest adjacent and pre-searched blocks are used; such that one of the adjacent blocks at the left (or right) side and six of the adjacent blocks upper (or lower) side, see figure (2). The shift values (Δx, Δy) of the initial motion vector are determined as the median values of the sorted Δx & Δy of these (7) searched blocks.

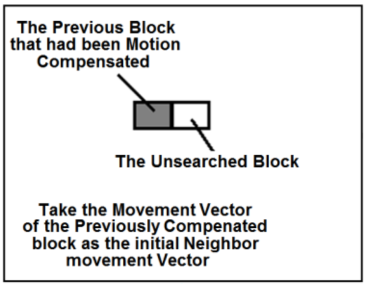

Fig. 1: An Illustration of Case 1

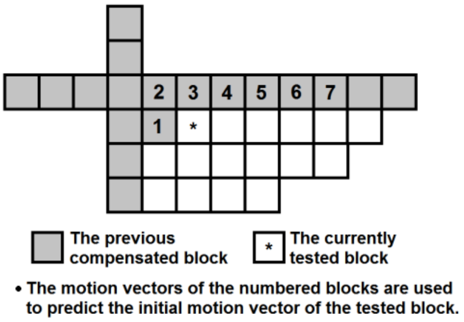
2
3
4
5
6
7
1
*
The previous
compensated block
*
The currently
tested block
• The motion vectors of the numbered blocks are used
to predict the initial motion vector of the tested block.

Fig. 2: An Illustration of Case-2

Case 3: If the number of searched blocks on previous row of tested block is less than four, and when there are many previously searched blocks adjacent to the unsearched current block from left (or right) and many other blocks at the upper (or lower) side; then seven of the adjacent blocks are selected using three of the adjacent blocks at left (or right) side and four of the adjacent blocks at the upper (or lower) side. Also, the shift values (Δx, Δy) of the initial motion vector are determined as the median value of sorted Δx & Δy of the 7 previously motion compensated blocks; Figure (3) illustrates the 3rd case.

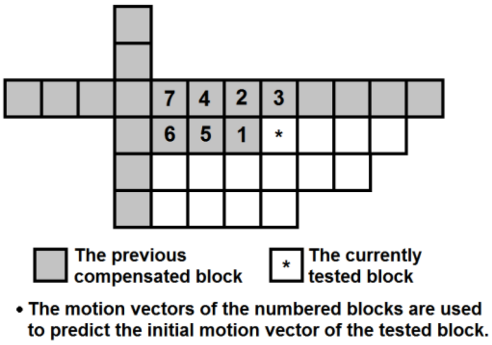
7
4
2
3
6
5
1
*
The previous
compensated block
*
The currently
tested block
• The motion vectors of the numbered blocks are used
to predict the initial motion vector of the tested block.

Fig. 3: An Illustration of Case-3

III. THE THREE PROPOSED SEARCH ALGORITHMS

First Method: 8-Nieghbors Search (8-N)

The following steps explain the flow of the proposed algorithm:

Step1: Compute the sum absolute difference (SAD) at the initial search position; if it is less than the predefined minimum expected error then the search process is stopped and go to Step 6, otherwise continue.

Step 2: Compute the SAD of the 8-neighbors blocks adjacent to the initial block

Step 3: Select the two lowest SAD resulted from steps 1 and 2. If the difference between these two least errors greater than a predefined parameter (α) then move the search to the block position that has least error, see Figure 4. Otherwise, do the next search steps but with the two blocks have least minimum SAD, see figure 5. In this work the value of α is set 12.

Step 4: Calculate the SAD of the remaining 8-neghibors adjacent to the new selected block(s) in step3.

Step 5: Among the remaining new calculated SAD select the minimum one and compare with lowest SAD resulted from step 3 and select the least one.

Step 6: Compute the Δx and Δy of the movement.

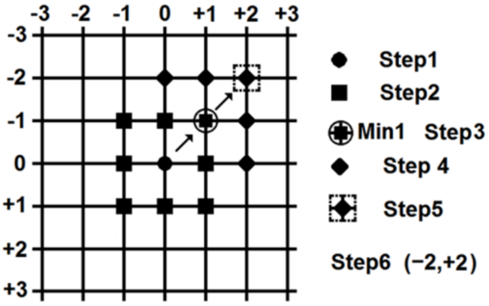

Fig. 4: An illustration of 8-Neighbor Search Algorithm (when the absolute difference between the two least SAD calculated in step 2 is greater than α).

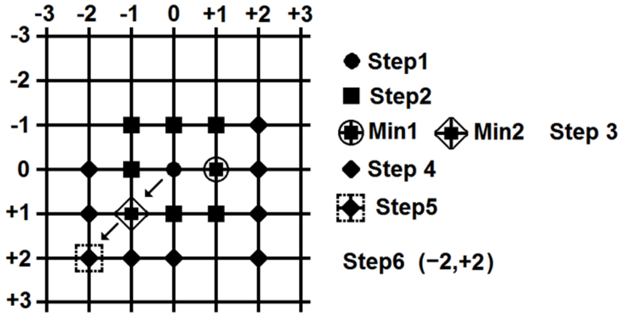

Fig. 5: An illustration of 8-Neighbor Search Algorithm (when the absolute difference between the two least SAD calculated in step 2 less α).

Second Method: 4- Neighbors Search (4-N)

This search method differs from the proposed 8-neighbor search in steps 2 and 3, as shown below:

Step 2: Compute the SAD of the 4-neighbors blocks adjacent to the initial block.

Step 3: In addition to the 4-neighbors blocks searched in step2 one of the 4-diagonal neighbors (i.e., corners) will be included in the tests, see figure (6). The selection of corner block depends on the location of the found minimum error, which is defined according to following criteria:

- **Top-Right**: if (SAD_{top}+SAD_{right}) is the minimum sum of error.
- **Top-Left**: if (SAD_{top}+SAD_{left}) is the minimum.
- **Down-Left**: if (SAD_{down}+SAD_{left}) is the minimum.
- **Down-Right**: if (SAD_{down}+SAD_{right}) is the minimum.

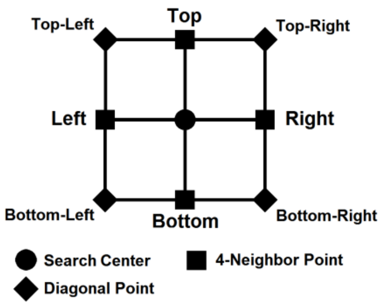

Fig. 6: The search points of the 4-Neighbor method

That is, the minimum sum of error specifies the diagonal neighbor point to which the search window should move, see Figure (7). The reduction of 8-neighbor search mechanism to 4- neighbor search as a second search step, and the addition of the selected diagonal direction (in step3) will make the search window involved with 5-searched blocks (instead of 8 blocks), this will reduce the computational time cost.

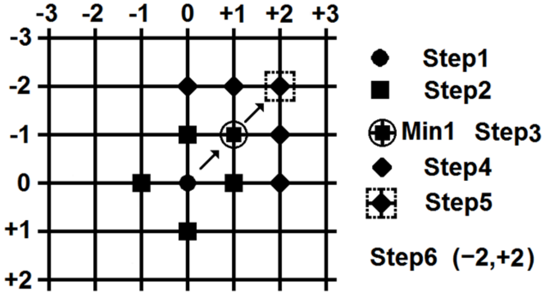

Fig. 7: An illustration of 4-Neighbor Search Algorithm

Third Method: Hybrid Search

For less search error, we proposed an additional step added to steps of the above mentioned two methods. The additional step is using the famous ES (4-N&ES and 8-N&ES) if it is necessary. The mixing depends on the search results of 4-N or 8-N. If one of following cases occurs then the application of ES step is applied:

Case-1: the value of the computed SAD is greater than a predefined threshold value (E_{max}).

Case-2: If the absolute difference between the computed Δx or Δy and the corresponding predicted values is greater than (γ).

The suitable threshold value (E_{max}) is assessed according to the following steps: (i) select randomly four blocks from the frame such that one block from each quarter of the frame; (ii) select the core block (at center of the frame) as the 5th selected block, (iii) apply ES algorithm on these 5 blocks, (iv) determine the mean (m) of determined SAD values for the five blocks, (v) determine the threshold: E_{max}=m*β; where β is multiplication factor.

IV. TESTS RESULTS

Many sets of tests have been conducted to investigate the performance of the proposed blocks searching scheme (4-N, 8-N, 4-N&ES and 8-N&ES). The used video test samples are Family, Foreman and Akiyo (with frame size specifications: size=320x240, size=320x480 and size=320x480 pixels, respectively, and pixel color depth=24 bit). Table (1) lists the values of best attained MAE and the search time (in second) for ES, TSS and the four proposed search methods. The listed results are the averages of 3-runs each applied on one frame (frame1, frame2 and frame3) of the test video sequences. The tests are done with block size 8x8 for all search methods. For 4-N&ES, 8-N&ES the search tests are done with parameters values (β=2 and γ=3). Figure (8) presents a comparison between the test results in terms of MAE; while Figure (9) shows the comparison in terms of search time (ST).

Table 1: Comparison between MAE and ST for testing applied on the test videos using different searches methods (ES, TSS, 4-N, 8-N, 4-N&ES, 8-N&ES)

Method	Video Sequence					
	Family		Foreman		Akiyo	
	ST (sec)	MAE	ST (sec)	MAE	ST (sec)	MAE
ES	0.239	3.152	0.304	2.340	0.306	0.116
TSS	0.021	3.829	0.028	2.844	0.009	0.116
4-N	0.016	4.226	0.021	3.705	0.007	0.117
4-N&ES	0.069	3.207	0.088	2.374	0.039	0.116
8-N	0.022	4.195	0.029	3.916	0.011	0.116
8-N&ES	0.075	3.200	0.093	2.365	0.039	0.116

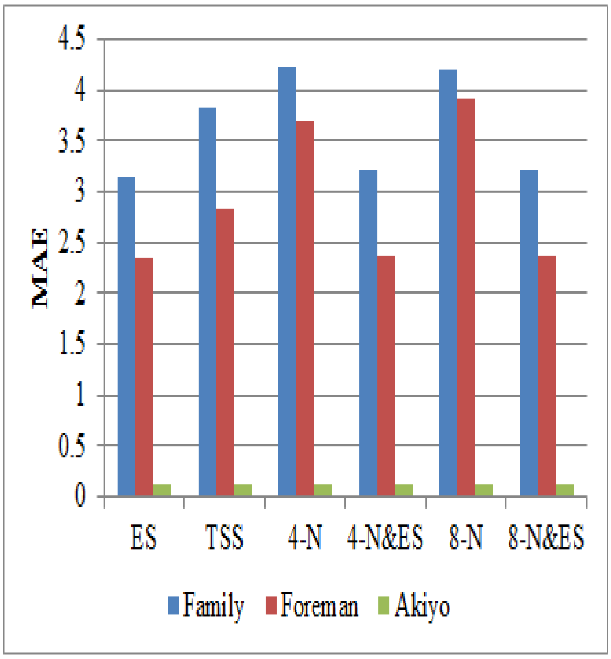

Fig. 8: The test results for comparing the performance of different search methods in term of MAE

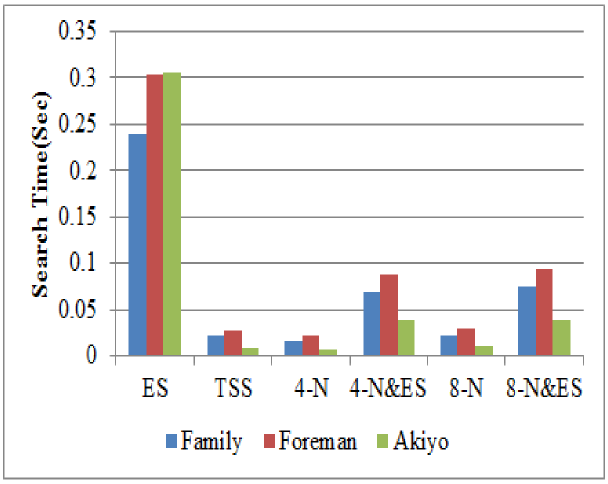

Fig. 9: The test results for comparing the performance of different search methods in terms of ST.

Table (2) shows the motion compensation results for the two proposed methods (8-N&ES, 4-N&ES) using different multiplication factor (β) values. This table shows the effect of (β) on both MAE and the Search Time (in second). The range of β is taken [1.0-3.0]. The results show that the selection of proper β value depends on the two conflicting parameters (i.e., time versus accuracy) requirements. If low MAE is needed then the value β should be low; while if low search time is required then the value of β should be high. We have choose β=2 as best trade-off value, because it leads to acceptable values for both search time and MAE.

Table (3) shows effect of (γ), which is used in hybrid search, on both MAE and search time for the proposed methods (8-N&ES, 4-N&ES). The value of (γ) is taken to be within the range [1,4]. The results in the table indicate that if we take (γ) less than 3, we get low MAE with high search time, but if we take (γ) greater than 3 then the changes in both search time and MAE become not significant. St, the most suitable (γ) value is considered (3) because it leads to acceptable values for both search time and MAE.

Figures (10) and (11), shows the effect of β and γ on search time, while figures (12) and (13) present their effects on MAE. The tests results indicated that the effect of α within the range [10,14] is insignificant, for this reason its value is set fixed at (12).

Table 2: The effects of multiplication factor β between the MAE and ST for testing applied family video in searches methods (8-N&ES, 4-N&ES)

β	8-N&ES		4-N&ES	
	ST (sec)	MAE	ST (sec)	MAE
1	0.121	3.176	0.121	3.174
1.5	0.089	3.182	0.084	3.182
2	0.075	3.200	0.069	3.207
2.5	0.062	3.218	0.057	3.221
3	0.052	3.227	0.048	3.239

Table 3: The effect of (γ) on the MAE and ST for testing applied family video in searches methods (8-N&ES, 4-N&ES)

γ	8-N&ES		4-N&ES	
	ST (sec)	MAE	ST (sec)	MAE
1	0.184	3.156	0.177	3.156
2	0.096	3.161	0.089	3.173
3	0.075	3.200	0.069	3.207
4	0.071	3.200	0.066	3.207
5	0.071	3.200	0.066	3.207

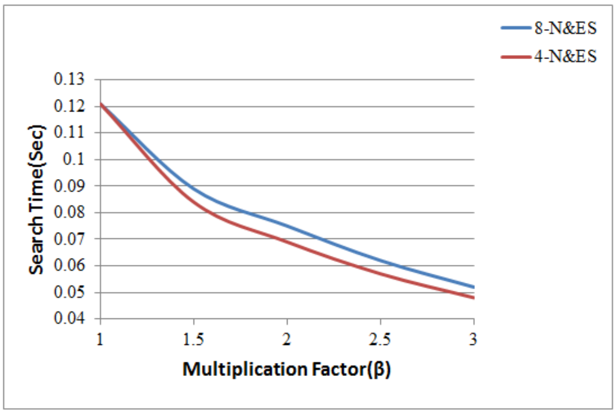

Fig. 10: The effect of multiplication factor β on search time (ST)

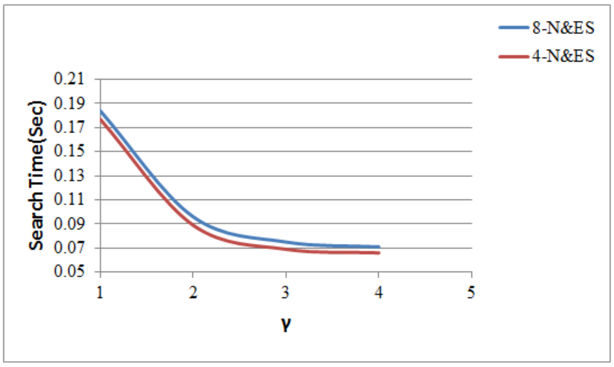

Fig. 11: The effect of (γ) on search time (ST)

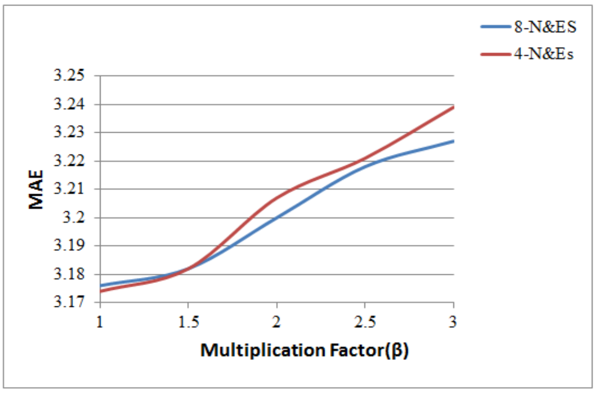

Fig. 12: The effect of multiplication factor β on MAE

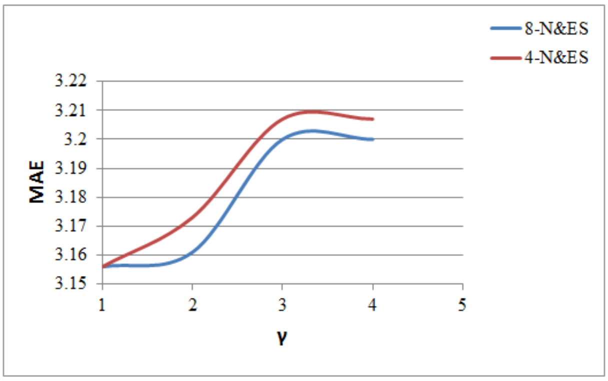

Fig. 13: The effect of (γ) on MAE

V. CONCLUSIONS

From the test results of the proposed systems, the following remarks are stimulated:

- The predictive system enhance the searching system, it enforces 8-N and 4-N searching algorithms such that they produce results similar, or close, to the standard TSS but with less number of blocks matching trials.
- The engagement of proposed searched system algorithms with ES algorithm produces an excellent searching system in term of distortion and searching time.
- For future work, the searching based on making matching between features extracted from the searched blocks can be developed; this will significantly reduce the computational complexity.

REFERENCES

[1] Array Microsystem Inc., "An Introduction to Video Compression", Suit 6, 897 University Ave., Los Gatos, California 95030, United States, White Paper, 1997.

[2] S. I. A. Pandian, D. J. Bala, B. A. George , "A Study on Block Matching Algorithm for Motion Estimation", Inernational Journal on Compuer Science and Engineering (IJCSE), Vol. 3, No. 1, Jan 2001, Pp. 34-44.

[3] S. Sonongsathitanon, S. S. Dlay, "A New Orthogonal Logarithmic Search Algorithm for Fixed Block Based Motion Estimation for Video Coding", University of Newcalstil, a report in 2001.

[4] Deepak J. Jayaswal, Mukesh A. Zaveri, "Probability Based Search Motion Estimation Algorithm", First International Conference on Computational Intelligience, Communication Systems and Networks, Pp. 357-362, 2009

[5] T. Koga, K. Linuma, A. Hirano, Y. Iijima, nd T Ishiguro, "Motion Compensation for Video Conference", in Proc. Nat. Telecommun. Conf., New Orleans LA, Nov. 29_Dec. 1, 1981, Pp. G5.3.1-G5. 3.5.

[6] R. Li, B. Zeng and M. L. Liou, "A New Three-Step Search Algorithm for Block Motion Estimation", IEEE Transactions on Circuits and System for Video Technology. Vol.4, No. 4 , Pp. 438-442, Augest 1994.

[7] L. M. Po and W. C. Ma, "A Novel Four- Step Search Algorithm for Fast Block Motion Estimation", IEEE Transactions on

Circuits and System for Video Technology. Vol. 6 , Pp. 438-442, June,1996.

[8] S Zhu and K. K. Ma, "A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation", IEEE Transactions on Circuits and System for Video Technology, Vol. 9, Pp. 287-290, Feb, 2000.

[9] C. Zhu, X. Lin and L. Chau, "Hexagon-Based Search Pattern for Fast Block-Matching Motion Estimation", IEEE Transactions on Circuits and System for Video Technology, Pp. 349-355, May, 2002.

[10] Z. Chen, Y. He and J. Xu, "Hybrid Unsymmetrical Cross Multihexagon-grid Search Strategy for Integer Pel Motion Estimation in H.264", in Proc. PCS, Pp. 17-22, April, 2003.

[11] R. Srinivasan and K. R. Roa, "Predective Coding Based on Efficient Motion Estimation", IEEE Transactions Communication, Vol. COM-33, Pp. 888-896, 1985.

[12] E. Loay and A. Aree, "Fast Predective Coding Based on an Enhanced Block's Motion Estimation", World Computer International Conference on IPCV, USA, 2008.

[13] A. Aree, "OTS and Cenrtal Expannding Block Scan for Interframe Compression", International Conference on Educational and Information Technology (ICEIT 2010), Pp. v3-71-v3-75, 2010.

[14] L. Liu and Q. Song, "A Novel Fast Motion Estimation for Video Coding", 3rd International Congress on Image and Signal Processing (CISP2010), Pp. 33-37.

AUTHOR BIOGRAPHY

Zainab J. AHMED had graduated for Al- Mustansiriya University at 2004. Now she is a postgraduate student preparing her thesis project at Baghdad University.

Loay E. GEORGE had graduated from Baghdad University at 1979. He got M.Sc at 1982 and PhD degree at 1997 in digital image processing. Now, he is working as teaching staff member in Computer Science Department/ College of Science/ University of Baghdad.

Bushra A. SULATAN had graduated from university of Technology at 1995. She got MSc. at 2001. Now, she is working Computer Science Department/ College of Science/University of Baghdad.