ISSN:0975-9646

Comparative Study of Motion Estimation Techniques in Video

S Sangeeta Mishra[#], Chittaranjan Pradhan[*], Alka Singh[#]

[#]*School of Computer Engineering, KIIT University*
Bhubaneswar, Odisha, India

[*]*School of Computer Engineering, KIIT University*
Bhubaneswar, Odisha, India

***Abstract*— Block matching motion estimation is the essence of video coding systems. Therefore, development of an efficient method for fast motion estimation is the basic requirement of the video encoder design. Block based motion estimation algorithms are used to reduce the memory requirements of any video file and also decrease computational complexity. We have been presented here some block matching algorithms used for motion estimation in video compression. It compares different types of block matching algorithms that range from the very basic Full Search to the recent fast adaptive algorithms.**

***Keywords*— Block Distortion Measure, Block Matching Algorithm, Motion Estimation, Motion Vector, PSNR.**

I. INTRODUCTION

Video is the technology of electronically capturing, recording, processing, storing, transmitting and reconstructing a sequence of still images representing scenes in motion. Video Compression is the process of reducing size in order to save space or transmission time [1]. It refers to reducing the amount of data used to represent digital video images. In a video scene, data redundancy arises from spatial, temporal and statistical correlation between frames. Motion estimation and compensation are used to reduce temporal redundancy between successive frames in the time domain. Motion estimation is the process of searching a fixed region of a previous frame of video to find a matching block of pixels of the same size under consideration in the current frame. Motion compensation is the algorithmic technique employed in the encoding of video data for video compression [1]-[11]. It is the process which measures the motion of objects from one picture to the next, so that it can allow for that motion when coming for redundancy of pictures.

The Moving Picture Experts Group(MPEG) is a working group of experts that was formed by ISO and IEC to set standards for audio and video transmission. MPEG generally produces better quality video than competing formats such as Video for Windows and Quick Time. MPEG algorithms compress data to form small bits that can be easily transmitted and then decompressed. MPEG achieves its high compression rate by storing only the changes from one frame to another, instead of each entire frame.

In Block Matching Algorithm [2], each frame is divided into number of blocks. Each block in the present frame is matched against all candidate blocks within a search area on the reference frame. These candidate blocks are just the displaced versions of original block. The best candidate block is found and its displacement (motion vector) is being recorded.

Block matching uses a value called "Block Distortion Measure" (BDM) to rate similarity between two blocks. The basic idea is to sum of the square differences of the pixel luminance of pixels located at the same position in the two blocks. The "Mean Standard Error" (MSE) is the squared differences between two candidate blocks.

$$\textbf{MSE(i,j)} = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left[F(m,n) - F(m+i,n+j)\right]^2$$

Here, M*N is the size of the sub block. F(m,n) is the intensity of the pixel located at (m,n) of the current frame. F(m+i,n+j) is the intensity of the pixel located at (m+i,n+j) of the reference frame and (i,j) is the motion vector.

This chapter represent a survey of various fast block matching motion estimation algorithm which ranges from Full Search Algorithm to recent fast adaptive algorithm. All of these existing motion estimation algorithms use "Mean Squared Error" (MSE) formula to evaluate the minimum BDM position. Experimental results show the performance of those algorithms mainly used on PSNR (Peak Signal to Noise Ratio).

PSNR [10] value is calculated by the formula:

$$\textbf{PSNR} = \textbf{10*}\log\left(\frac{255*255}{MSE}\right)$$

II. BLOCK MATCHING ALGORITHM

In this section, we have presented some popular block motion estimation algorithms that range from Full Search Algorithm to fast adaptive algorithm.

A. Full Search Algorithm

Full search (FS) [2]-[4] or Exhaustive search algorithm compares the current block with all the candidate blocks of the reference frame within the search area, for finding out the best matched block in the reference frame. The minimum BDM point is calculated by MSE (Mean Squared Error) formula.

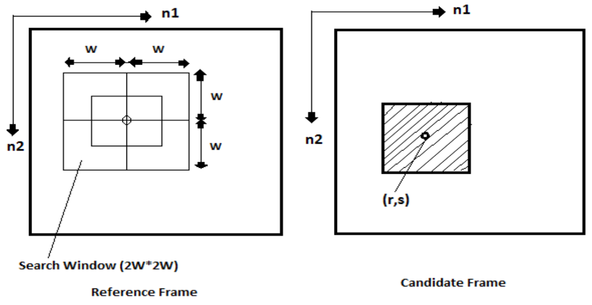

Fig. 1 Full Search Algorithm

Consider a block of N*N pixels from the candidates frame at the coordinate position (r, s) as shown and then consider a search window having a range 'w' in both horizontal and vertical directions of the references frame. For each of the $(2w + 1)^2$ search position (including the current row and the current column), the candidate block is compared, and the best matching block, along with the motion vector is determined. Finally, the global minimum block distortion position which corresponds to the best matching block is found by searching all the candidate blocks. Computation cost is very high because we have to check all the candidate blocks for minimum BDM point.

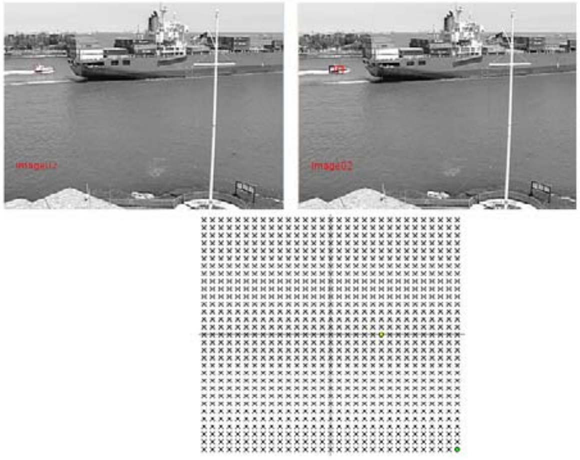

Fig. 1 Screenshot of Full Search Algorithm

In this screenshot "Container" sequence is used and the block size is 8*8. The figure shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position. Here all the points are searched to find out the minimum block distortion (BDM) position. At minimum BDM position, point error value is 70.56, PSNR (Peak Signal to Noise Ratio) is 68.26. This algorithm compares 225 blocks.

B. *Three Step Search Algorithm*

Three-step search (TSS) [3]-[4] is mainly used for low bit-rate video compression applications e.g. video conferencing, because of its simplicity, robustness. The following steps describe the TSS algorithm:

Step-1: Select an initial step size equal or slightly larger than half of the maximum search range . Calculate the error for the block at the centre of search area and 8 square neighbourhood point at the distance of 's' from centre.

Step-2: Halve the search range to 2 pixels. Now the block with minimum error will be considered as centre. Again, find the coordinate with the minimum block distortion (BDM) point. If the step size is greater than one, repeat the step 1, otherwise go to step 3.

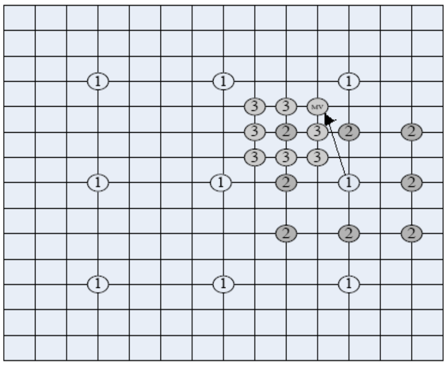

Fig. 3 Three Step Search Algorithm

Step-3: The search range is reduced to 1 pixel. Finally, by creating a search pattern, search for the minimum BDM. The minimum BDM point is evaluated using MSE formula.

TSS is more efficient to find the global minimum particularly for those sequences with large motion. It reduces the quality of the motion compensation system.

Fig. 4 shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position. At point A, MSE=94.64 & PSNR=65.32, At point B, MSE=94.39 & PSNR=65.35, At point C, MSE=97.31 & PSNR=65.04, At point D, MSE=107.57 & PSNR=64.04, At point E, MSE=118.0 & PSNR=63.11, At point F, MSE=73.20 & PSNR=67.89, At point G, MSE=117.5 & PSNR=63.16, At point H, MSE=108.29 & PSNR=63.97 and At point I, MSE=87.48 & PSNR=66.11. So after the 1st iteration, the minimum BDM coordinate is found at (4,0).

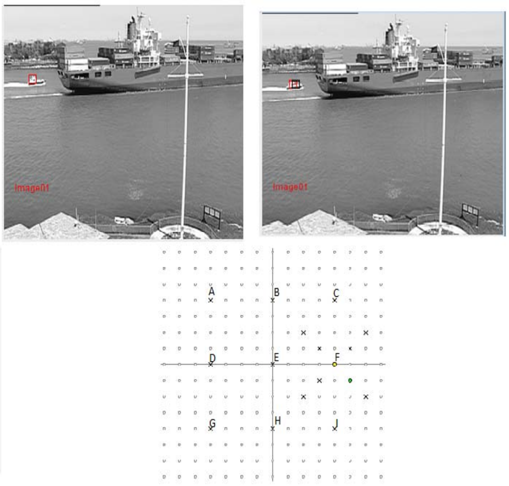

Fig. 4 Screenshot of Three Step Search Algorithm

At (4,0) point error value is 73.20, PSNR (Peak Signal to Noise Ratio) is 67.89. For comparison, each time the pixel will be halved and the process will be repeated three times to determine the minimum BDM point. After all comparisons, the minimum BDM position remains same i.e., at coordinate (4,0). This algorithm compares 17 blocks.

So the algorithm is better than FS algorithm when compared with number of searched blocks.

C. ***New Three Step Search Algorithm***

New Three Step Search Algorithm (NTSS) [4]-[5]-[6] differs from TSS by considering a centre biased searching scheme pattern i.e. by taking additional 8-points in its first step. It has a advantage like half-way stop technique. The steps of the algorithm in detail:

Step-1: A search pattern is being created, consisting of the centre coordinate and eight surrounding coordinates at a distance of one pixel and four pixels as shown in Figure. Search for minimum BDM point of this pattern. If the minimum BDM is located at centre, then the search is finished. Otherwise: Go to step 2.

Step-2: If the minimum BDM is located at one of the eight surrounding coordinates of the centre coordinate, add eight new coordinates and search it again to find out the minimum BDM. After that, the search is done. Otherwise, continue like TSS.

Step-3: Search the four coordinates surrounding the current minimum BDM at a distance of 1 pixel for the final minimum BDM point.

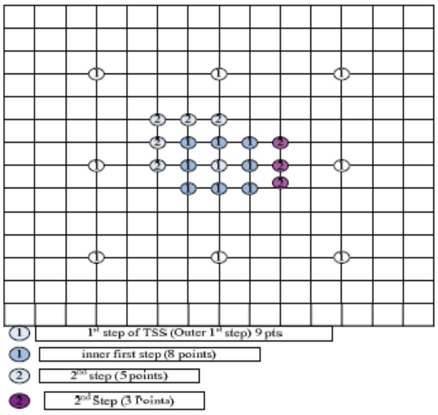
1st step of TSS (Outer 1st step) 9 pts
inner first step (8 points)
2nd step (5 points)
2nd Step (3 Points)

Fig. 5 New Three Step Search Algorithm

The search pattern of this algorithm is fixed and no predefined threshold operations are required. This algorithm does not perform well in worst case condition as it has to compare 33 points in worst case.

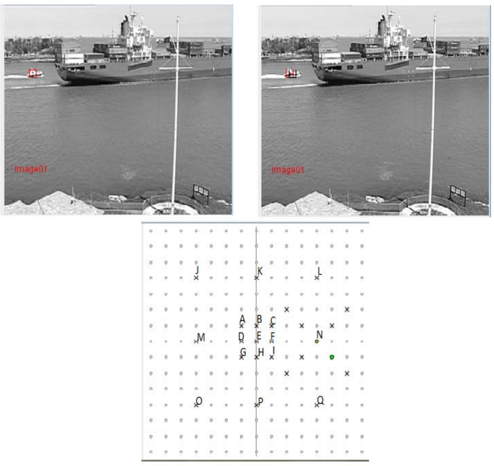

Fig. 6 Screenshot of New Three Step Search Algorithm

Fig. 6 shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position. At point A, MSE=101.14 & PSNR=64.66, At point B, MSE=112.42 & PSNR=63.60, At point C, MSE=100.65 & PSNR=64.70, At point D, MSE=111.57 & PSNR=63.37, At point E, MSE=109.34 & PSNR=63.88, At point F, MSE=92.98 & PSNR=65.50, At point G, MSE=100.26 & PSNR=64.74, At point H, MSE=99.21 & PSNR=64.85, At point I, MSE=97.04 & PSNR=65.07, At point J, MSE=92.79 & PSNR=65.52, At point K, MSE=86.75 & PSNR=66.19, At point L, MSE=85.14 & PSNR=66.38, At point M, MSE=109.73 & PSNR=63.84, At point N, MSE=76.56 & PSNR=67.44, At point O, MSE=116.35 & PSNR=63.25, At point P, MSE=116.67 & PSNR=63.23 and At point Q, MSE=79.79 & PSNR=67.03. So after the 1st iteration, the minimum BDM coordinate is found at (4,0).

Fig. 6 shows the minimum BDM (Block Distortion Measure) position which is (4,0). At (4,0) point error value is 76.56, PSNR (Peak Signal to Noise Ratio) is 67.44. This algorithm compares 25 blocks. For comparison, each time the pixel will be halved and the best candidate block is found. After all comparisons, the minimum BDM position remains same i.e., at coordinate (4,0). So the algorithm is better than FS algorithm but gives poor performance when compared with TSS algorithm when compared with number of blocks.

D. ***Four Step Search Algorithm***

The Four Step Search Algorithm(FSS) [2]-[3] algorithm uses a centre-biased search pattern, that means instead of taking 9 x 9 search window, a 5 x 5 search window is taken. This algorithm reduces the number of searching points as compared to TSS algorithm. These are the steps of FSS algorithm:

Step-1: A search pattern is created with 5 x 5 window. If the minimum block distortion measure (BDM) point is located at the centre of the search window, go to step 4; otherwise go to step 2.

Step-2: The search window size is maintained in 5 x 5. Move the centre to the minimum BDM point. If the minimum BDM is located at any corner of the search window, then five additional checking points are added to that search pattern as shown in Fig. 7. . If the minimum BDM is located at middle of the horizontal or vertical axis of the search window, then three additional checking points are added to that search pattern as shown in Fig. 7. If the minimum BDM point is found at the centre of search window, go to step 4; otherwise go to step 3.

Step-3: The searching pattern strategy is same as step 2, however it will go to step 4.

Step-4: Finally the search window is reduced to 3 x 3 and the minimum BDM point and corresponding motion vector is computed.

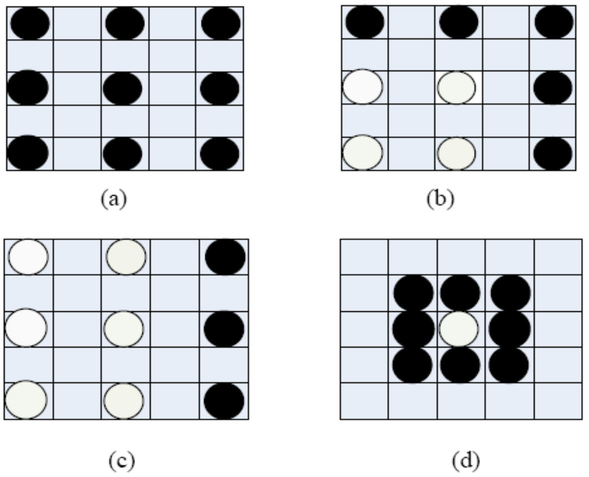

Fig. 7 Four Step Search Algorithm

It takes less number of searching points as compared to TSS. In worst case situation, it compares 25 (9+5+5+8) blocks. Hence, FSS performs better in the worst case when compared with NTSS algorithm. Computational complexity and picture quality remain almost same when compared to the TSS and NTSS algorithm.

Figure shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position. At point A, MSE=98.20 & PSNR=64.95, At point B, MSE=93.78 & PSNR=65.41, At point C, MSE=106.5 & PSNR=64.14, At point D, MSE=97.25 & PSNR=65.05, At point E, MSE=114.95 & PSNR=63.38, At point F, MSE=96.71 & PSNR=65.10, At point G, MSE=111.75 & PSNR=63.66, At point H, MSE=106.85 & PSNR=64.11 and At point I, MSE=108.87 & PSNR=63.92. So after the 1st iteration, the minimum BDM coordinate is found at (0,2).

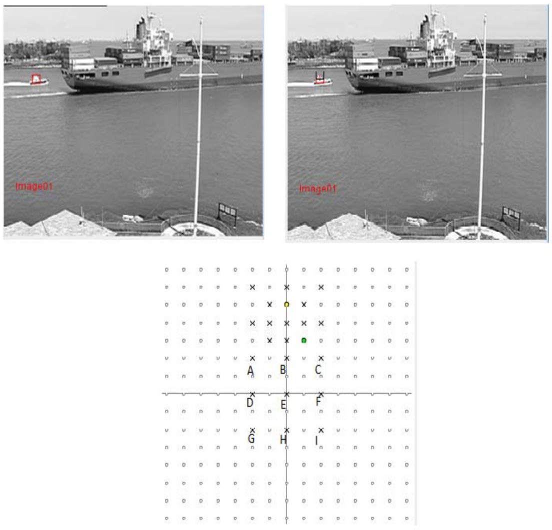

Fig. 8 Screenshot of Four Step Search Algorithm

At (0,2) point error value is 93.78, PSNR (Peak Signal to Noise Ratio) is 65.41. Each time new coordinates will be added to find out the global minimum distortion measure(BDM) and after all comparisons, the minimum BDM point is found at coordinate (0,5) whose point error value is 79.59, PSNR (Peak Signal to Noise Ratio) is 67.05. This algorithm compares 23 blocks. So the algorithm is better than FS algorithm when compared with number of searched blocks.

E. *Diamond Search Algorithm*

Diamond Search(DS) [3]-[4] algorithm is also one of the fast blocks matching motion estimation algorithm. Here the search pattern is created in the shape of diamond. Two search pattern is used in this algorithm. One is Large Diamond Search Pattern (LDSP) with 9 searching points and another is Small Diamond Search Pattern (SDSP) with five searching points. The algorithm is described as follows:

Step-1: Initially for LDSP, a search pattern with 9 coordinates along with centre at (0,0) are tested to find out the minimum BDM. If the minimum BDM is found at centre, then go to step 3; otherwise go to step 2.

Step-2: Move the centre to the minimum BDM position to form an another LDSP. If the new minimum BDM is located at centre of the search pattern, then go to step 3; otherwise repeat this step.

Step-3: Finally the search pattern is switched to a SDSP. The final minimum BDM and its corresponding motion vector is computed in this step.

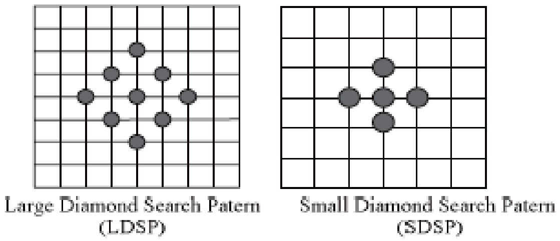
Large Diamond Search Patern (LDSP)
Small Diamond Search Patern (SDSP)

Fig. 9 Diamond Search Algorithm

The DS pattern can find large motion blocks with fewer search points. The compact shape of the DS pattern around the centre reduces the number of search points than other square-shaped algorithm for finding MV. The disdvantage is that the large number of search points per block can hamper on motion estimation time.

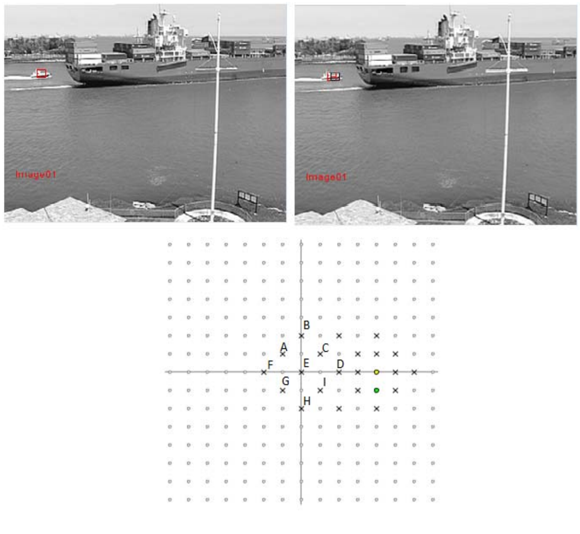

Fig. 10 Screenshot of Diamond Search Algorithm

Figure shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position. At point A, MSE=89.01 & PSNR=65.93, At point B, MSE=86.92 & PSNR=66.17, At point C, MSE=113.62 & PSNR=63.49, At point D, MSE=83.84 & PSNR=66.53, At point E, MSE=99.03 & PSNR=64.87, At point F, MSE=110.90 & PSNR=63.73, At point G, MSE=99.34 & PSNR=64.83, At point H, MSE=99.14 & PSNR=64.85 and At point I, MSE=85.73 & PSNR=66.31. So after the 1st iteration, the minimum BDM coordinate is found at (2,0).

The following screenshot shows the minimum BDM (Block Distortion Measure) position which is (2,0). At (2,0) point error value is 83.84, PSNR (Peak Signal to Noise Ratio) is 66.53. For comparison, each time the pixel will be halved. Each time new coordinates will be added in the shape of a diamond to find out the global minimum distortion measure(BDM) and after all comparisons, the minimum BDM point is found at coordinate (4,0) whose point error value is 60.12, PSNR (Peak Signal to Noise Ratio) is 69.86. This algorithm compares 23 blocks.

F. *Hexagon Based Search Algorithm*

A hexagon based search algorithm [7] consists of seven checking points with the centre surrounded by six endpoints of the hexagon with the two edge points (up and down) being excluded. The HEXBS algorithm is described as follows:

Step-1: A large hexagon with seven checking points along with centre at (0,0) is created. If the minimum BDM point is located at the centre of the hexagon, then go to step 3; otherwise go to step 2.

Step-2: Move the centre to the minimum BDM position to form new large hexagon. Three new coordinates are searched. If the new minimum BDM is located at centre of the hexagon, then go to step 3; otherwise repeat this step.

Step-3: Finally the search pattern is switched from a large hexagon to a small size hexagon. The four points of the small hexagon is searched. The final minimum BDM point and its corresponding motion vector is computed in this step.

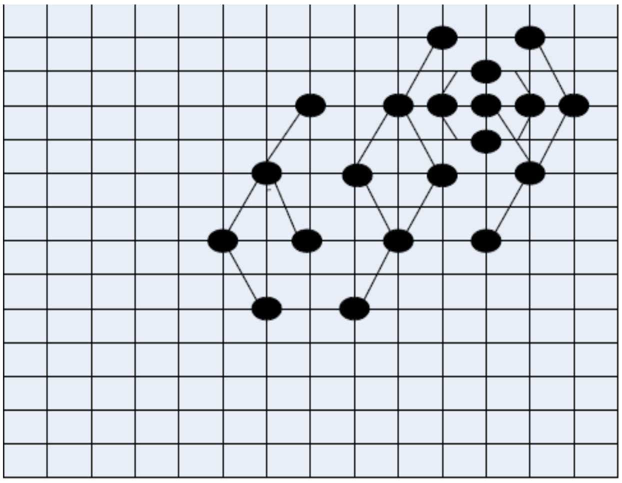

Fig. 11 Hexagon Based Search Algorithm

It takes less number of searching points than DS algorithm. The motion estimation algorithm does not have significant execution speed to reduce encoding time.

Figure shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position. At point A, MSE=103.57 & PSNR=64.42, At point B, MSE=97.82 & PSNR=64.99, At point C, MSE=91.32 & PSNR=65.68, At point D, MSE=110.29 & PSNR=63.79, At point E, MSE=97.96 & PSNR=64.97, At point F, MSE=113.82 & PSNR=63.47 and At point G, MSE=105.5 & PSNR=64.23. So after the 1st iteration, the minimum BDM coordinate is found at (2,0).

The following screenshot shows the minimum BDM (Block Distortion Measure) position which is (2,0). At (2,0) point error value is 91.32, PSNR (Peak Signal to Noise Ratio) is 65.68.

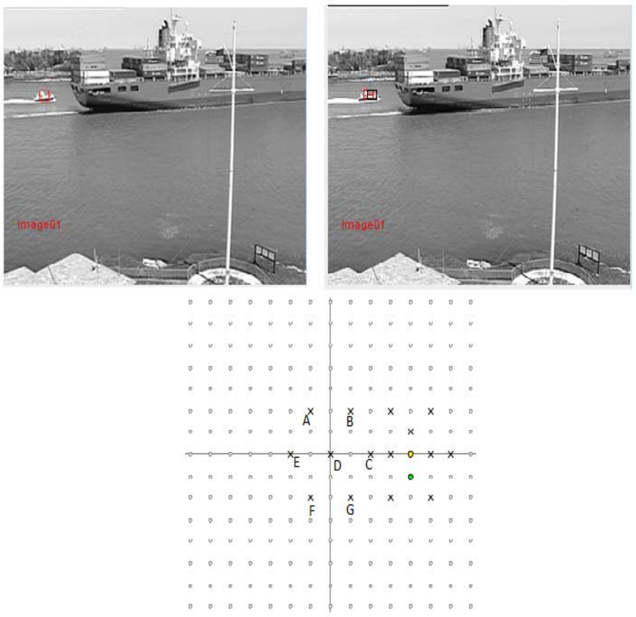

Fig. 12 Screenshot of Hexagon Based Search Algorithm

Each time new coordinates will be added in the shape of a hexagon to find out the global minimum distortion measure(BDM) and after all comparisons, the minimum BDM point is found at coordinate (4,0) whose point error value is 63.12, PSNR (Peak Signal to Noise Ratio) is 69.37. This algorithm compares 17 blocks.

G. *Cross Search Algorithm*

Cross Search Algorithm [8] is one of the motion estimation algorithms quite same as TSS. It uses a threshold based first step stop. This algorithm increases the performance of videos. The algorithm is described as follows:

Step-1: The centre is compared with the current block, if the minimum distortion is less than defined threshold, then the algorithm stops. This is called threshold based first step stop.

Step-2: Move the centre to the minimum BDM position and a search pattern is created in the shape of "X" around the centre. Each time the pixel will be halved.

Step-3: If the pixel value is greater than 1, then halve it and repeat step 2; otherwise go to step 4.

Step-4: Move the centre to the minimum BDM point. If the minimum BDM is located at bottom left or top right, then minimum distortion will be evaluated by checking 4 more points around the new centre in the shape of a "+". If the minimum BDM is located at top left or bottom right point, then minimum distortion will be evaluated by checking 4 more points around the new centre in the shape of a "x".

It performs better than some of the other algorithms with higher computational complexities. Its compensation performance is not satisfactory when compared with other fast block matching algorithms. Computational complexities is very high.

Figure shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position.

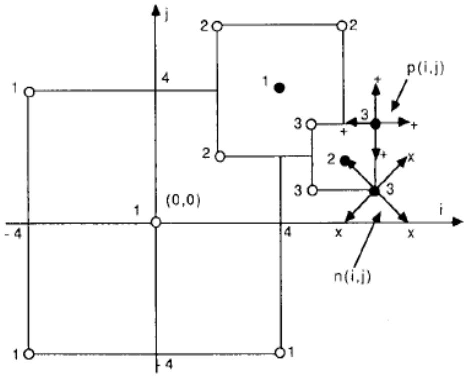

Fig. 13 Cross Search Algorithm

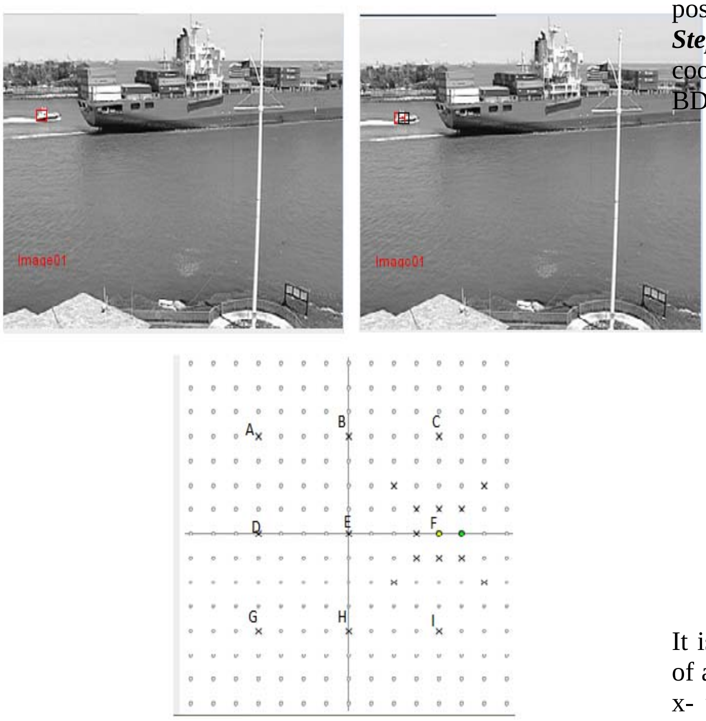

Fig. 14 Screenshot of Cross Search Algorithm

At point A, MSE=110.90 & PSNR=63.73, At point B, MSE=82.51 & PSNR=66.69, At point C, MSE=86.71 & PSNR=66.19, At point D, MSE=86.73 & PSNR=66.19, At point E, MSE=94.01 & PSNR=65.39, At point F, MSE=72.42 & PSNR=68.00 and At point G, MSE=113.73 & PSNR=63.48, At point H, MSE=113.84 & PSNR=63.47 and At point I, MSE=95.29 & PSNR=65.25. So after the 1st iteration, the minimum BDM coordinate is found at (4,0).

For comparison, each time the pixel will be halved and the process will be repeated three times to determine the minimum BDM point. After all comparisons, the minimum BDM position remains same i.e., at coordinate (4,0). The following screenshot shows the minimum BDM (Block Distortion Measure) position which is (4,0). At (4,0) point error value is 72.42, PSNR (Peak Signal to Noise Ratio) is 68.00. This algorithm compares 21 blocks. This algorithm performs better than other algorithms(FS, TSS, NTSS, FSS) but with higher complexities.

*H. **Two Dimensional Logarithmic Search Algorithm***

The two-dimensional logarithmic search (TDL) [9] is the first block-matching algorithm to exploit the quadrant monotonic model to match blocks. The block at the centre of the search area and the four candidate blocks at a distance S from the centre on the x and y axes are compared to the target block to determine the best match. The algorithm is described as follows:

Step-1: A search pattern is created by considering 5 additional points along with the centre at (0,0) at a distance "s" from this on the X and Y axes.

Step-2: If the minimum BDM point is located at the centre, then halve the step size. If minimum BDM point is located at one of the other four points, then move centre to that position and step 1 is repeated.

Step-3: When the pixel value is reduced to 1, then all the 9 coordinates around the centre are checked and the minimum BDM and its corresponding motion vector is evaluated.

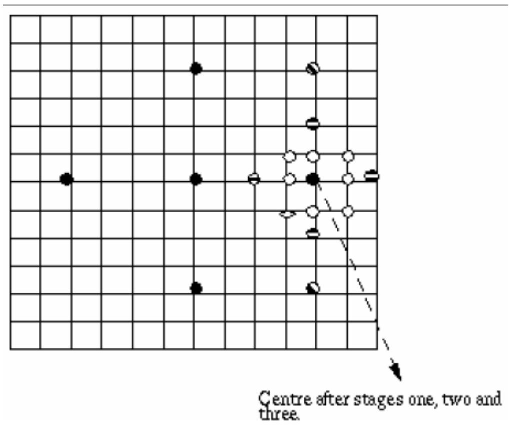

Fig. 15 Two Dimensional Logarithmic Search Algorithm

It is suitable for sequences with fast motion. The accuracy of algorithm is low when the motion vector is at an angle to x- y axis and computational complexities is very high as compared to TSS, CSA.

Figure shows the minimum BDM (Block Distortion Measure) position which is marked as a yellow dot. The green dot is current searched position. At point A, MSE=105.98 & PSNR=64.19, At point B, MSE=98.56 & PSNR=64.91, At point C, MSE=109.04 & PSNR=63.90, At point D, MSE=67.57 & PSNR=68.69 and At point E, MSE=96.25 & PSNR=65.15. So after the 1st iteration, the minimum BDM coordinate is found at (4,0).

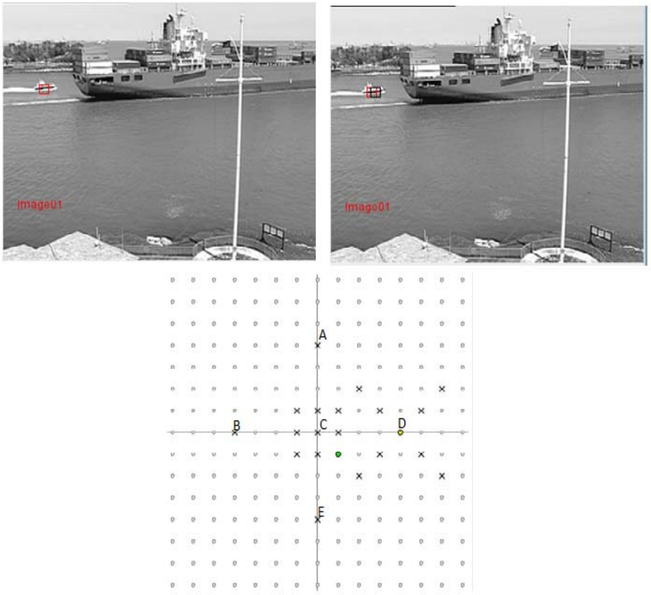

Fig. 16 Screenshot of Two Dimensional Logarithmic Search Algorithm

For comparison, each time the pixel will be halved and the process will be repeated three times to determine the minimum BDM point. After all comparisons, the minimum BDM position remains same i.e., at coordinate (4,0). At (4,0) point error value is 67.57, PSNR (Peak Signal to Noise Ratio) is 68.69. This algorithm compares 21 blocks. Here computational complexities is very high.

COMPARISON STUDY

The performance evaluation is based on the MSE (Mean Standard Error and PSNR (Peak Signal to Noise Ratio) and searching point per macro block. The following table shows the experimental results of the different block matching algorithms by taking the "Container" image sequence as input. Minimum MSE and PSNR values are taken at minimum BDM point. The algorithms have been tested on the video sequences like "Container", "Boat" and "Tennis".

TABLE I
EXPERIMENTAL VALUES FOR "CONTAINER" SEQUENCE

Algorithm	**MSE**	**PSNR**	**No. of Compared Blocks**
FS	70.56	68.26	225
TSS	73.20	67.89	17
NTSS	76.56	67.44	25
FSS	79.53	67.05	23
DS	60.12	69.86	23
HEXBS	63.12	69.37	17
CS	72.42	68.00	21
TDLS	67.57	68.69	21

The Figure 17 shows the graph, based on the PSNR values of these three sequences. The PSNR (Peak Signal to Noise Ratio) value is calculated at minimum block distortion point (BDM) point for each algorithm.

TABLE III
EXPERIMENTAL PSNR VALUES FOR THREE SEQUENCES

Algorithm	Container	Boat	Tennis
FS	68.26	67.43	106.87
TSS	67.89	65.86	105.77
NTSS	67.44	65.90	103.36
FSS	67.05	68.10	104.70
DS	69.86	71.47	97.68
HEXBS	69.37	73.27	105.40
CS	68.00	66.75	97.08
TDLS	68.69	72.09	94.36

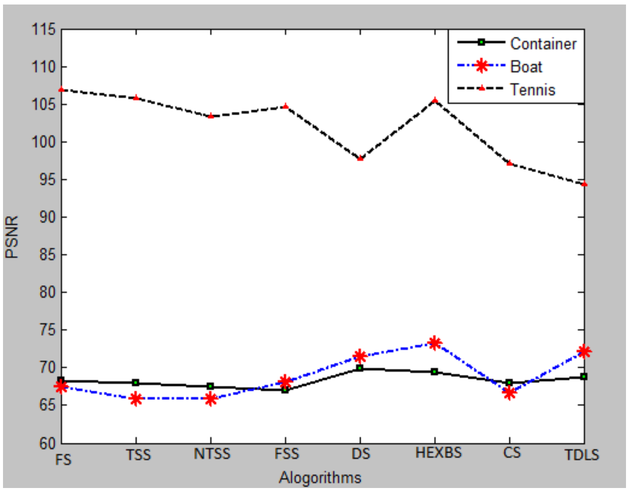

Fig. 17 Graph based on PSNR values of the algorithms

From this analysis, we have found that the full search (FS) technique produces better quality image as it gives better performance in PSNR calculation, but takes larger number of search points; whereas, diamond search (DS) and hexagon-based search (HEXBS) algorithms take a few numbers of search points and also give average performance in PSNR calculation. Other algorithms (i.e. TSS, NTSS, and FSS) take lesser number of search points, but produce distorted image because of poor PSNR performance. As DS and HEXBS algorithms take fewer number of search points, they are faster. The image quality of DS and HEXBS can be improved by increasing the PSNR values. Therefore, getting a better quality image with less number of searching points remains a goal.

I. CONCLUSIONS

Motion estimation is the most vital component of the video compression standard. In this paper, we have presented different motion estmation techniques and its algorithms. Performance of different Block matching motion estimation algorithms are analyzed and compared with respect to their result of PSNR, MSE and computational complexity.

REFERENCES

[1] John Watkinson. *The MPEG Handbook*, 2nd Edition, Chapter 1 and 5, Elsevier Ltd. 2004, Pages:1-34, 230-352.

[2] S.Immanuel Alex Pandian. Dr. G. Josemin Bala, Becky Alma George, *A Study on Block Matching Algorithms for Motion Estimation*, International Journal on Computer Science and Engineering (IJCSE), Vol.3, No.1, pp.34-44, Jan 2011.

[3] Fenta Adnew Mogus, Xinying Liu, Lei Wang. *Evaluation of the Performance of Motion Estimation Algorithms in Video Coding*. IEEE, 2010.

[4] Chittaranjan Pradhan, Dipannita Adak. *Survey on Block Matching Algorithms for Motion Estimation*. International Journal of Computer Applications (0975 8887), Volume 46, No.16, pp.6-10, May 2012.

[5] Renxiang Li, Bing Zeng, and Ming L. Liou. *A New Three-Step Search Algorithm for Block Motion Estimation*. IEEE, Vol.4, No.4, pp.438-442, August 1994.

[6] Shengqi Yang, Wayne Wolf, Fellow, IEEE, and N. Vijaykrishnan, Member, IEEE. *Power and Performance Analysis of Motion Estimation Based on Hardware and Software Realizations*. IEEE, Vol.54, No.6, pp.714-726, June 2005.

[7] Ce Zhu, Xiao Lin, and Lap-Pui Chau. Hexagon-*Based Search Pattern for Fast Block Motion Estimation*. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 12, NO. 5, pp. 349-355, MAY 2002.

[8] M. GHANBARI. *The Cross-Search Algorithm for Motion Estimation.* IEEE TRANSACTIONS ON COMMUNICATIONS, VOL. 38, NO. 7, pp.950-953, JULY 1990.

[9] S. Usama, M. Montaser, O. Ahmed. *A Complexity and Quality Evaluation of Block Based Motion Estimation Algorithms*. Acta Polytechnica Vol. 45, pp. 29-41, No. 1/2005.

[10] PSNR calculating formula. Available at http://www.chasanc.com/index.php/Coding/PSNR-Calculating-Difference.html.

[11] Ranjan Parekh. *Principles of Multimedia*, Chapter 10, TMHPC Ltd. 2006, pages: