

## A Three-Point Directional Search Block Matching Algorithm

A. V. Paramkusam, D. Laxma Reddy

Department of Electronics and Communication Engineering, MLR Institute of Technology, India

### Article Info

#### Article history:

Received Oct 7, 2016

Revised Dec 30, 2016

Accepted Jan 14, 2017

#### Keyword:

Block matching algorithm

Fast search algorithm

Motion estimation

### ABSTRACT

This paper proposes compact directional asymmetric search patterns, which we have named as three-point directional search (TDS). In most fast search motion estimation algorithms, a symmetric search pattern is usually set at the minimum block distortion point at each step of the search. The design of the symmetrical pattern in these algorithms relies primarily on the assumption that the direction of convergence is equally alike in each direction with respect to the search center. Therefore, the monotonic property of real-world video sequences is not properly used by these algorithms. The strategy of TDS is to keep searching for the minimum block distortion point in the most probable directions, unlike the previous fast search motion estimation algorithms where all the directions are checked. Therefore, the proposed method significantly reduces the number of search points for locating a motion vector. Compared to conventional fast algorithms, the proposed method has the fastest search speed and most satisfactory PSNR values for all test sequences.

Copyright © 2017 Institute of Advanced Engineering and Science.  
All rights reserved.

### Corresponding Author:

A. V. Paramkusam,  
Department of Electronics and Communication Engineering,  
MLR Institute of Technology, India.  
Email: adapa74@gmail.com

## 1. INTRODUCTION

Motion estimation (ME) in video sequences, has been deeply analyzed by the research community, due its importance in the visual experience for a huge variety of visual tasks. Motion estimation is an essential part in video coding process and greatly achieves significant compression by removing the temporal redundancy existing in a video sequence. However, the intensive computation of ME is the major problem for real-time encoders. In block based motion estimation, the current frame is divided into non-overlapping macro blocks of 16 x 16 pixels. The ME part of video codec calculates the motion vector for each macroblock by searching best matching candidate block from the previously encoded reference frame.

Full Search (FS) is an exhaustive search algorithm and it is the simplest method to find the optimal motion vectors for each macroblock. In it, the Sum of absolute difference (SAD) is calculated at each search point in the search area. The SAD between a macroblock of size  $N \times N$  at position  $(p, q)$  in the current frame  $f_t$  and candidate block at position  $(p + x, q + y)$  in the reference frame  $f_{t-1}$  is defined in the eq (1).

$$SAD(x, y) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |f_t(p+i, q+j) - f_{t-1}(p+x+i, q+y+j)| \quad (1)$$

With search range of  $W$  there will be  $(2W+1)^2$  candidate blocks or search points. So, for each motion vector the SAD function has to be evaluated  $(2W+1)^2$  times. At each search point  $N^2$  computations are required to calculate the SAD and one computation consists of one subtraction, one addition and one absolute value operations. Thus the total computation per one motion vector is  $(2W+1)^2 \times N^2 \times 3$  operations. Let a

frame size is  $K \times L$  and frame rate is  $T$  fps, the amount of computation is  $3TKL(2W+1)^2$  per second. This makes full search block matching algorithm as a very computationally intensive method. To reduce the complexity of full-search, several fast search motion estimation algorithms have been proposed at the price of slightly impaired Peak Signal-to-Noise Ratio (PSNR) performance. A possible classification of these fast search motion estimation algorithms into the following categories: reduction in search positions [1-9], predictive search [10-14], search pattern switching [15], [16], multi-resolution search [17-20] and Fractional-Pixel Interpolation [21], [22]. Existing fast search motion estimation algorithms use one or a combination of these categories.

The main assumption of most fast search motion estimation algorithms is that the block distortion measure for example SAD is monotonic over the search range, implying that the SAD increases monotonically as the search position moves away from the minimum distortion point. So, the best match point may be found by following the distortion trend without checking all search points in the search window. Accordingly, these fast search motion estimation algorithms employ various search patterns to reduce the number of search points, thereby speeding up the search. The main drawback of these algorithms is that the search pattern of these algorithms is usually symmetric, and the magnitude of block matching error is not effectively used. For most fast search motion estimation algorithms, the initial point of search is usually set at the center of the search window and the search occurs according to a symmetric pattern. After comparison, the new center point is set at the point with the least amount of SAD, and then generates a new symmetric search pattern for the next search. This procedure continues until the conditions of convergence are satisfied. In block matching algorithm, the most important assumption is that the error surface is monotonic. But, the structure of the symmetrical pattern assumes that the direction of convergence is equally alike in each direction with respect to the search center. As a result, the monotonic property is not properly used. If the search direction can be correctly determined, the search speed will be further enhanced.

In this paper, a novel three-point directional search (TDS) algorithm is developed. This algorithm consists of 9 possible search patterns of which eight are directional patterns and the remaining one is a compact symmetric search pattern which consists of nine search points. The remainder of this paper is organized as follows. In Section 2, we briefly describe the some conventional fast search motion estimation algorithms. The details of the proposed TDS are given in Section 3. Section 4 comprises a discussion of the experimental results. Finally, conclusions are provided in Section 5.

## 2. FAST SEARCH MOTION ESTIMATION ALGORITHMS

The diamond search algorithm (DS) is proposed by S. Zhu and K. K. Ma [3]. It employs two search patterns as shown in Figure 1(a) and (b). The first one is a coarse search with a large diamond search pattern (LDSP) to find a small area of optimal motion vector, and second one is a fine search using a small diamond search pattern (SDSP) to find the best motion vector in the located small area. The search starts with the LDSP which is centered at the origin of the search window, and the 9 checking points of LDSP are tested. If the minimum sum of absolute difference ( $SAD_{\min}$ ) calculated is not located at the centre position, new LDSP is formed which is centered at the  $SAD_{\min}$  point found in the search. The search procedure is repeated until  $SAD_{\min}$  occurs at the centre point. At any stage if  $SAD_{\min}$  occurs at centre of LDSP, the search pattern is switched to SDSP as reaching to the final search stage. Among the five checking points in SDSP, the position yielding the  $SAD_{\min}$  provides the motion vector of the best matching block.

Generally, a circle-shaped search pattern with a uniform distribution of a minimum number of search points is desirable to achieve the fastest search speed uniformly. The diamond shaped pattern is not close enough to a circle, which is just  $90^\circ$  rotation of a square. Ce Zhu, Xiao Lin, and Lap-Pui Chau [4] propose a hexagonal search pattern (HS) to achieve substantial speed improvement over the DS algorithm with similar distortion performance. The HS algorithm uses two search patterns, a large hexagonal pattern which is close enough to a circle and a small hexagonal pattern as shown in Figure 1(c) and (d). In the first step of search, the large hexagonal pattern with seven checking points is used for coarse search. In the second step if the  $SAD_{\min}$  is found at the centre, switch to use the small hexagonal pattern, including four checking points for the focused inner fine search. Otherwise, the search continues around the point with  $SAD_{\min}$  using the same large hexagonal pattern. The hexagonal search algorithm reduces the search points by improving the coarse search speed against the DS algorithm.

An enhanced hexagonal search (EHS) algorithm [6] is proposed to reduce the search points further. This EHS uses the 6-side-based fast inner search technique to improve the inner search speed against the HS [4]. In HS, the entire inner search points inside the large hexagon have to be evaluated, which is computationally ineffective. The EHS algorithm only checks the most promising inner search points by exploiting the group-sum distortion information of the six checked endpoints of the large hexagon. At first EHS starts coarse search with the large hexagonal pattern to trace a small area of optimal motion vector.

After locating a small area in the coarse search, EHS groups the six checked endpoints of the large hexagon as shown in Figure 2(a). At each group, the group-sum distortion is calculated and then focused inner checking points nearest to the smallest distortion group will be evaluated to obtain the minimum distortion point. Three inner searching points nearby the smallest distortion group will be calculated in the focused inner search if the smallest distortion group is 3 or 6. Otherwise, two inner searching points nearest to the smallest distortion group will be used in the focused inner search.

An enhanced hexagonal-based search using point-oriented inner search (EHS-POIS) is presented in [7] for further reduction of the search points over EHS. The EHS-POIS uses an efficient grouping method for the large hexagon which is based on minimizing mean internal distance (MID) for each inner point, as shown in Figure 2(b). The eight inner points enclosed by large hexagon are divided into two sets by the MID value. First set contains a, c, e, f, g and h points and second set includes b and d points. These points are surrounded by two or three nearest checked large hexagon points. The normalized group distortions (NGDs) of the hexagon are calculated and the minimum NGDs in each set are found. The two inner points related to minimum NGDs in two sets are finally evaluated to find the final motion vector. The fine search of EHS requires 2 or 3 search points depending on the portion of inner points with smallest group distortion whereas the fine search of EHS-POIS requires only 2 search points constantly.

An enhanced hexagonal-based search using direction-oriented inner search (EHS-DOIS) [8] has shown the considerable improvement over EHS-POIS by increasing the inner search speed double. The EHS, EHS-POIS, and EHS-DOIS algorithms use same large hexagonal search pattern for coarse search and find small hexagonal area using same procedural steps. But, these algorithms have different prediction strategies to find the portion of inner checking points for fine search. After finishing coarse search, EHS-DOIS forms pseudo-points prediction pattern  $\{A, B, C, D, E, F, G, H\}$  as shown in Figure 2(c). The group distortions of these eight pseudo-points are calculated through six checked points of large hexagon. Select one point in  $\{a, b, c, d, e, f, g, h\}$  that would be on the arrow from the center of large hexagon to the pseudo-point with minimum distortion among eight pseudo-points. The SAD at this selected point is finally evaluated to find the final motion vector.

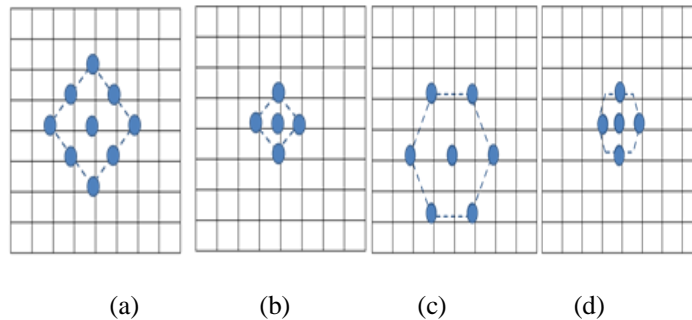


Figure 1. Search patterns of DS and HS algorithms (a) LDSP with 9 search points (b) SDSP with 5 search points (c) Large hexagonal pattern (d) Small hexagonal pattern

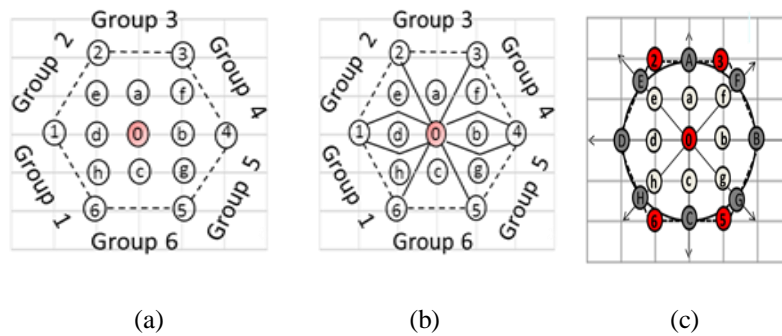


Figure 2. Strategies used to predict portion of inner searches in EHS, EHS-POIS and EHS-DOIS. (a) Group oriented prediction of EHS. (b) Point-oriented prediction of EHS-POIS. (c) Direction-oriented prediction of EHS-DOIS

### 3. THREE-POINT DIRECTIONAL SEARCH (TDS) ALGORITHM

The TDS has been proposed to find out the motion vectors with fewer number of search points. The strategy of TDS is to keep searching for the minimum block distortion (MBD) point in the most probable directions, unlike the previous fast search motion estimation algorithms where all the directions are checked. As shown in Figure 3, the proposed TDS consists of 9 possible search patterns of which eight are directional patterns and the remaining one is initial search pattern. The initial search pattern is a compact directional symmetric search pattern (Figure 4) which consists of nine search points. In the first step of the search, we follow the initial search pattern where we check eight adjacent points of the search centre in eight directions including the search centre in order to find out the most probable search direction in whose vicinity the MBD is present. Thereafter, we continue the search using a particular search pattern from among the eight small three-point directional search patterns at each step of the search. At any given search step, selection of a search pattern depends on the search direction of the previous search step. Search direction is defined as the direction from the search centre to location of the MBD point. The search centre and the MBD point are unique at each step of the search. While the search continues, if the MBD is found in the centre at any search step, the search ceases and that search centre is the required motion vector. There are eight directional search patterns corresponding to the eight possible directions. Once the search direction is obtained, three additional search points according to the search direction (the red diamonds in each directional search pattern as shown in Figure 3 are selected for the next step as follows:

Let the search centre and the MBD point in a given search step be denoted by 'S' and 'M' respectively. The straight line from 'S' toward 'M' indicates the search direction. Extend the line SM in the search direction and select the search point which is closest to the MBD point on the extended line. Let this search point be denoted by 'A' as shown in Figure 3. The remaining two points 'B' and 'C' (Figure 3) are selected in such a way that the lines MB and MC are at angles  $\pi/4$  and  $-\pi/4$  with respect to MA. The search procedure with TDS is summarized as follows.

Step 1) The initial compact directional symmetric search pattern (Figure 4) is centered at the origin of the search window.

Step 2) The block distortions are calculated at nine search points of the compact directional symmetric search pattern. If the location of the MBD occurs at the center, the search comes to an end and the search centre is considered as the motion vector. Otherwise, proceed to step 3.

Step 3) Set the location of the MBD as the new center, find the search direction, and select the proper search pattern for the next step according to the search direction.

Step 4) Three additional search points (the red diamonds in Figure 3) are checked basing on the selected pattern. If the location of the MBD remains unchanged, the search discontinues, and the motion vector is set at the location of the MBD. If not, return to step 3. An example of a search procedure by the TDS to locate the motion vector (3, -2) is shown in Figure 5.

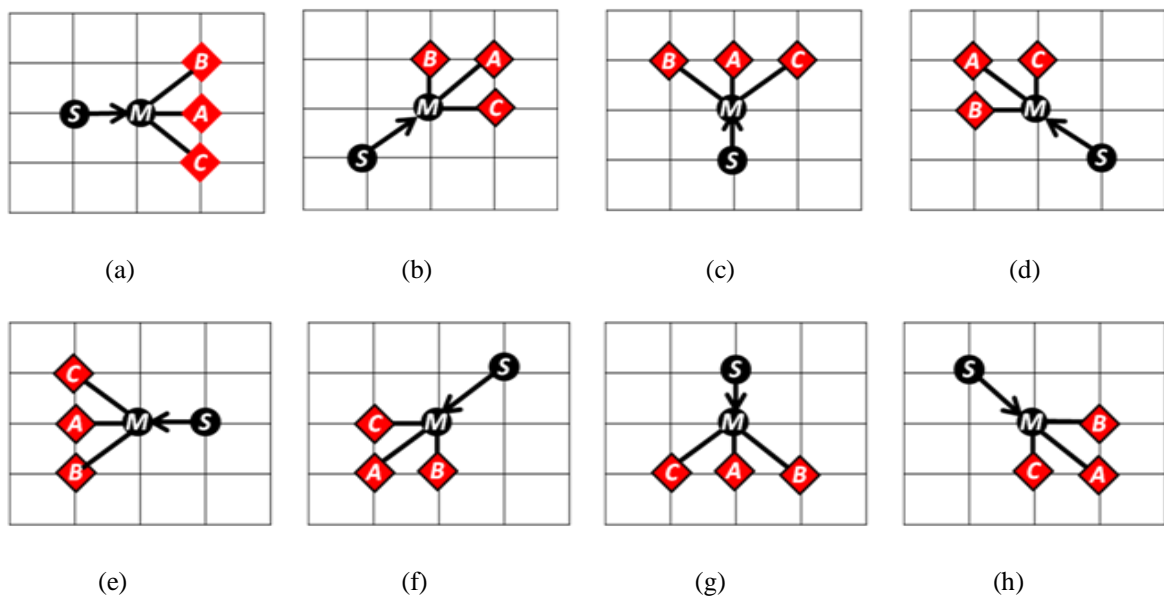


Figure 3. Search patterns of TDS. (a) Right pattern (b) Up-right pattern (c) Up pattern (d) Up-left pattern (e) Left pattern (f) Down-left pattern (g) Down pattern (h) Down - right pattern

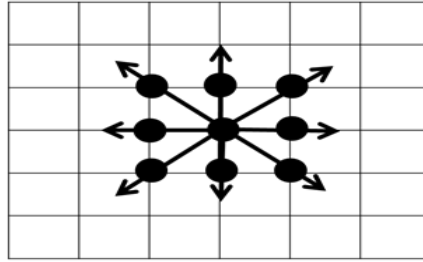


Figure 4. The initial search pattern of the TDS

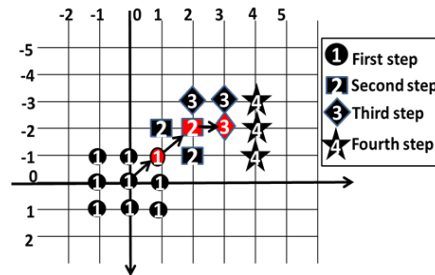


Figure 5. An example of a search procedure by the TDS to locate the motion vector (3, -2). Each candidate is marked with its step number, within which only one is the minimum BDM point (filled by red color).

#### 4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed method, eleven video sequences which consist of different types of motion content and various formats including QCIF, CIF, and HD are used. The luminance components of the first 100 frames of these video sequences are considered in the simulation. The block size is set to  $16 \times 16$  and the search range to  $\pm 63$  for high definition video sequences (Kirsten-Sara, Rocket launch and Football sequences) and  $\pm 15$  for the remaining video sequences. The SAD is used as the error measurement for finding the motion vector.

The Full-Search algorithm achieves the highest PSNR as it searches all possible search points in the search window and thus is guaranteed to find global minimum point. So, the PSNR achieved by Full-Search algorithm is used as reference to compare or evaluate other fast block matching algorithms. In fact, it is not possible to ground the quality of the results by excluding Full-Search algorithm. The simulation results with proposed TDS algorithm are contrasted with those of other algorithms in two aspects, which are: 1) The average search points per block and 2) The average peak signal-to-noise ratio (PSNR) per frame.

The results pertaining to the speed performance (i.e., the average search points per block) of all the fast block matching algorithms including the proposed TDS algorithm are summarized in Table 1. From Table 1, it is clear that the number of search points can be reduced more by applying TDS algorithm than any other state-of-the-art fast block matching algorithms. In other words, the TDS may find any motion vector in motion field with fewer search points than any other state-of-the-art fast block matching algorithms.

The results pertaining to the motion prediction quality (i.e., the average PSNR per frame) of all the fast block matching algorithms including the proposed TDS algorithm are shown in Table 2. It can be observed from Table 2 that the average PSNRs obtained by TDS are better than those of DS, HS, EHS, EHS-POIS, EHS-DOIS and DGDS algorithms in many cases. The total average PSNR of the TDS is better by 0.24dB, 0.52dB, 0.64dB, 0.51dB, 1.29dB and 0.15dB when compared to the total average PSNRs of the algorithms of DS, HS, EHS, EHS-POIS, EHS-DOIS and DGDS respectively.

The proposed TDS is better than DS algorithm with respect to the average PSNR per frame with least computational cost. The PSNR performance of TDS is either approximately equal to that of DGDS or better. In a few cases, the PSNR performance is less which is insignificant. However, the number of average search points per block has greatly reduced in the TDS when compared to this algorithm.

Coming to the HS algorithm, the proposed TDS shows better PSNR performance as well as speed performance. When compared to the enhanced versions of HS i.e., EHS, EHS-POIS and EHS-DOIS algorithms, the TDS shows noticeable speed improvement against these algorithms. However, the average PSNR per frame has greatly reduced in the TDS when compared to these algorithms.

Table 1. Average Search Points per Block

Video	TDS	DS	HS	Fast search motion estimation algorithms				
				EHS	EHS-POIS	EHS-DOIS	DGDS	Full-search
Foreman	10.57	17.71	13.04	10.89	10.92	10.61	18.64	782.21
Mobile	9.75	13.26	10.71	8.71	8.63	7.71	11.60	869.33
Rhinos	13.28	28.74	19.90	17.92	17.85	16.59	34.27	855.50
Robot boat	13.77	27.00	18.68	16.85	16.64	15.87	32.67	855.50
Suzie	9.07	12.96	10.44	8.40	8.31	7.48	11.16	782.21
Akiyo	7.87	11.43	9.64	7.56	7.5	6.68	7.87	782.21
Cricket	10.37	18.13	13.73	11.63	11.65	10.71	16.48	855.50
Flower	9.49	15.11	11.89	9.91	9.81	8.91	13.65	855.50
Kirsten-Sara	8.50	13.50	11.18	9.43	9.16	8.19	9.58	13977.21
Rocket launch	9.04	18.53	13.87	11.97	11.85	10.87	15.50	13977.21
Football	10.50	28.15	19.57	17.72	17.56	16.57	27.62	13977.21
Total Average	10.20	18.59	13.87	11.90	11.80	10.92	18.09	4415.4

Table 2. Average PSNR per Frame

Video	TDS	DS	HS	Fast search motion estimation algorithms				
				EHS	EHS-POIS	EHS-DOIS	DGDS	Full-search
Foreman	28.51	28.16	27.77	27.40	27.60	26.70	28.28	28.89
Mobile	23.88	23.83	23.66	22.96	23.47	22.71	23.87	23.95
Rhinos	27.92	27.06	26.84	26.86	26.89	26.75	27.45	29.16
Robot boat	27.78	27.35	27.20	27.20	27.21	27.05	27.71	28.71
Suzie	35.45	35.21	34.82	34.63	34.92	33.87	35.25	35.30
Akiyo	44.44	44.16	44.13	44.03	44.15	43.25	44.16	44.16
Cricket	33.81	33.80	33.61	33.59	33.60	32.91	33.90	35.57
Flower	32.57	32.32	31.78	31.97	32.03	30.41	32.40	32.62
Kirsten-Sara	45.69	45.74	45.17	45.16	45.25	44.05	45.54	45.97
Rocket launch	39.95	40.15	39.81	39.72	39.75	38.91	40.22	41.31
Football	27.99	27.58	27.49	27.49	27.51	27.21	27.62	33.66
Total Average	33.45	33.21	32.93	32.81	32.94	32.16	33.30	34.48

To figure out the efficiency of proposed TDS algorithm more garishly, the average PSNR per frame and the average search points per block of all the fast block matching algorithms including the proposed TDS algorithm using Foreman and Football video sequences have been plotted in Figure 6 and Figure 7 respectively. Figure 6(a) and Figure 7(a) plot a frame-by-frame comparison of the average search points per block for the proposed TDS algorithm and fast block matching algorithms applied to the Foreman and Football video sequences respectively. Figure 6(b) and Figure 7(b) plot a frame-by-frame comparison of average PSNR for the proposed TDS and fast block matching algorithms applied to the Foreman and Football video sequences respectively. Figure 6(a) and Figure 7(a) clearly show the supremacy of the proposed TDS against DS, HS, EHS, EHS-POIS, EHS-DOIS and DGDS algorithms in terms of the average search points per block, while Figure 6(b) and 7(b) manifest that the PSNR performance of the proposed TDS is closer to that of Full-Search like DS and DGDS algorithms and better than that of the enhanced versions of HS i.e., EHS, EHS-POIS and EHS-DOIS algorithms.

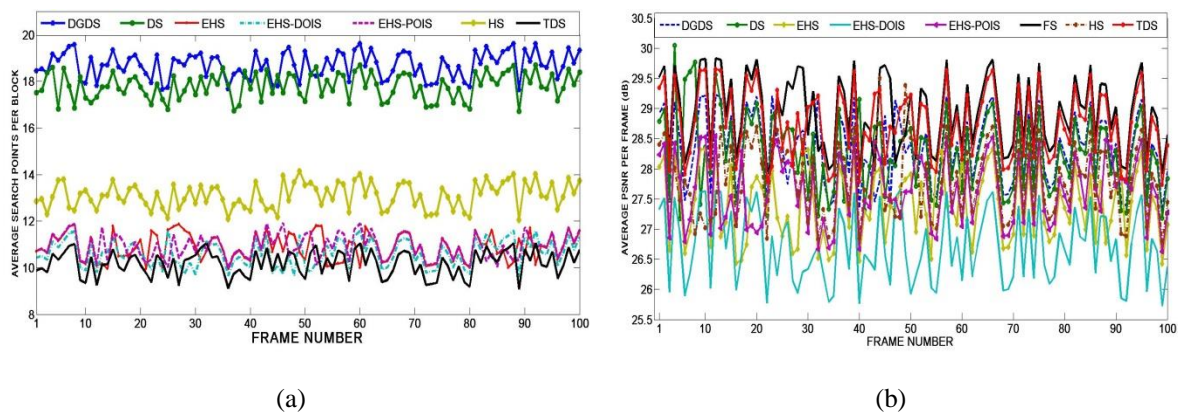


Figure 6. Performance comparison of the fast block matching algorithms for “Foreman” sequence in terms of: (a) the average number of search points per block and (b) average PSNR per frame

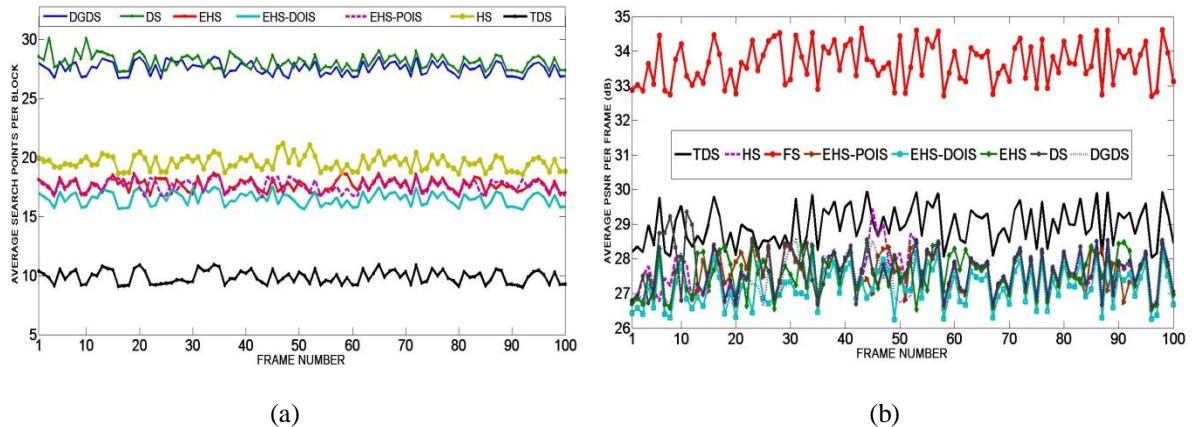


Figure 7. Performance comparison of the fast block matching algorithms for “Football” sequence in terms of: (a) the average number of search points per block and (b) average PSNR per frame

## 5. CONCLUSION

In this paper, we have proposed a novel three-point directional search (TDS) for motion estimation in video coding. We explored the relationship between search direction and search patterns. With a known search direction, asymmetric search patterns are developed, and the search points on the outside of the direction were disregarded. Since the unnecessary search points are eliminated, the search speed is greatly improved. The experimental results have demonstrated the superiority of our proposed method.

## REFERENCES

- [1] T. Koga, *et al.*, “Motion compensated interframe coding for video conferencing,” in *Proc. Nat. Telecommun. Conf.*, pp. C9.6.1–C9.6.5, 1981.
- [2] R. Li, *et al.*, “A New Three-step Search Algorithm for Block Motion Estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol/issue: 4(4), pp. 438–442, 1994.
- [3] S. Zhu and K. K. Ma, “A New Diamond Search Algorithm for Fast Block-matching Motion Estimation,” *IEEE Trans. Image Processing*, vol/issue: 9(2), pp. 287–290, 2000.
- [4] C. Zhu, *et al.*, “Hexagon-based Search Pattern for Fast Block Motion Estimation,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol/issue: 12(5), pp. 349–355, 2002.
- [5] C. H. Cheung and L. M. Po, “Novel Cross-Diamond-Hexagonal Search Algorithms for Fast Block Motion Estimation,” *IEEE Trans. on Multimedia*, vol/issue: 7(1), pp. 16–22, 2005.
- [6] C. Zhu, *et al.*, “Enhanced Hexagonal Search for fast block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol/issue: 14(10), pp. 1210–1214, 2004.
- [7] L. M. Po, *et al.*, “Novel point oriented inner searches for fast block motion estimation,” *IEEE Trans. on Multimedia*, vol/issue: 9(1), pp. 9–15, 2007.
- [8] B. J. Zou, *et al.*, “Enhanced Hexagonal-Based Search Using Direction-Oriented Inner Search for Motion Estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol/issue: 20(1), pp. 156–160, 2010.
- [9] O. Ndili and T. Ogunfunmi, “Algorithm and Architecture Co-Design of Hardware-Oriented, Modified Diamond Search for Fast Motion Estimation in H.264/AVC,” *IEEE Trans. Circuits Syst. Video Technol.*, vol/issue: 21(9), pp. 1214–1227, 2011.
- [10] Z. Shi, *et al.*, “A motion estimation algorithm based on Predictive Intensive Direction Search for H.264/AVC,” *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, pp. 667, 672, 19–23 July 2010.
- [11] Y. H. Ko, *et al.*, “Adaptive search range motion estimation using neighboring motion vector differences,” *Consumer Electronics, IEEE Transactions on*, vol/issue: 57(2), pp. 726, 730, 2011.
- [12] L. Hsieh, *et al.*, “Motion estimation using two-stage predictive search algorithms based on joint spatio-temporal correlation information,” *Expert Systems with Applications*, vol/issue: 38(9), pp. 11608–11623, 2011.
- [13] H. Nisar, *et al.*, “Content adaptive fast motion estimation based on spatio-temporal homogeneity analysis and motion classification,” *Pattern Recognition Letters*, vol/issue: 33(1), pp. 52–61, 2012.
- [14] Y. H. Ko, *et al.*, “Fast motion estimation algorithm combining search point sampling technique with adaptive search range algorithm,” *Circuits and Systems (MWSCAS), 2012 IEEE 55th International Midwest Symposium on*, pp. 988, 991, 5–8 Aug 2012.
- [15] K. H. Ng, *et al.*, “A Search Patterns Switching Algorithm for Block Motion Estimation,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol/issue: 19(5), pp. 753, 759, 2009.
- [16] J. J. Tsai and H. M. Hang, “On the Design of Pattern-Based Block Motion Estimation Algorithms,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol/issue: 20(1), pp. 136, 143, 2010.

- [17] F. Varray and H. Liebgott, "Multi-resolution transverse oscillation in ultrasound imaging for motion estimation," *Ultrasonics, Ferroelectrics, and Frequency Control, IEEE Transactions on*, vol/issue: 60(7), pp. 1333, 1342, 2013.
- [18] J. Stuckler and S. Behnke, "Multi-resolution surfel maps for efficient dense 3D modeling and tracking," *Journal of Visual Communication and Image Representation*, vol/issue: 25(1), pp. 137-147, 2014.
- [19] M. Nieuwenhuisen and S. Behnke, "Hierarchical planning with 3d local multiresolution obstacle avoidance for micro aerial vehicles," *Proceedings of the Joint Int. Symposium on Robotics (ISR) and the German Conference on Robotics (ROBOTIK)*, 2014.
- [20] D. Droschel, *et al.*, "Local multi-resolution representation for 6D motion estimation and mapping with a continuously rotating 3D laser scanner.In: Robotics and Automation (ICRA)," *IEEE International Conference on*, 2014.
- [21] Y. Lin and Y. C. Wang, "Improved parabolic prediction-based fractional search for H.264/AVC video coding," *Image Process. IET*, vol/issue: 3(5), pp. 261–271, 2009.
- [22] S. Dikbas, *et al.*, "Fast motion estimation with interpolation-free sub-sample accuracy," *IEEE Trans. Circuits Syst. Video Technol.*, vol/issue: 20(7), pp. 1047–1051, 2010.

## BIOGRAPHIES OF AUTHORS



**Adapa Venkata Paramkusam** is a Professor of Electronics and communication Engineering at the MLR Institute of Technology, Autonomous, Hyderabad. He is a self-directed, action-oriented professional with over 16 years teaching experience at various reputed Engineering Colleges. He qualified in GATE-2000 and 2001 with percentiles 93.48 and 94 respectively. He is a member of ISTE. He received his B.E and M.E, degrees in Electronics and communication from Andhra University, Visakhapatnam, India, in 1996 and 2004, respectively, he completed Ph.D in the area of video coding at the JNTUH, Hyderabad. He has outstanding contribution with 18 Publications in the National, IEEE International Conferences and International Journals. His research interests include Image processing, video coding and video water marking.



**D. Laxma Reddy** is Assistant Professor of Electronics and communication Engineering at the MLR Institute of Technology, Autonomous, Hyderabad. He has 5 years of teaching experience in various reputed Engineering Colleges. He is a life time member of ISTE. He received B.Tech in Electronics and communication and M.Tech in Computers and Communications from JNTU Hyderabad in 2008 and 2011 respectively. He has 6 publications in National and International journals. His research interests include Video Coding, Embedded Systems, Communication Systems.