International Journal of Applied Research 2016; 2(9): 607-613

ISSN Print: 2394-7500
ISSN Online: 2394-5869
Impact Factor: 5.2
IJAR 2016; 2(9): 607-613
www.allresearchjournal.com
Received: 22-07-2016
Accepted: 23-08-2016

Shaifali M Arora
Asstt. Prof., ECE Dept.
Maharaja Surajmal Institute of Technology, Delhi, India

Correspondence
Shaifali M Arora
Asstt. Prof., ECE Dept.
Maharaja Surajmal Institute of Technology, Delhi, India

Trends and issues in developing fast block motion estimation algorithms

Shaifali M Arora

Abstract
With the tremendous use of videos in all spheres of like Video conferencing, video Surveillance, High definition TVs etc., the demand to enhance the compression and quality of videos w.r.t the existing standards is the need of the day. As it is known a sequence of image frames are presented at such a high speed per sec that to the human eyes it seems like a video. Basically video data is composed of sequence of image frames and the change between the consecutive frames is very small. So lots of existing temporal and spatial redundancy is removed to achieve high video compressions. While encoding, instead of coding the whole frames independently, the difference between the consecutive frames is encoded. For this, Frame is divided into Blocks and the displacement of the current block in the current frame with respect to the best matching block in the reference frame is found which is known as motion vector. The most time consuming job of the Video encoder, is to find the motion vectors. To fasten the process of motion estimation a lot of research had been done by many researchers and came up with a no. of standards like MPEG1 to H.264 time to time to meet the needs of upcoming technologies. This review paper gives an over view of the need of video compression, basics of video encoding, the matching criterion for finding Motion Vectors, Fast Block Motion finding Algorithms and factors that may help in improving the quality and reducing the computations in a Video Encoder.

Keywords: Trends, issues, developing fast, block motion, estimation, algorithms

1. Introduction

The need enhancements in processing video data is increasing day by day, with the increasing usage of Videos in applications like television, video over internet, gaming and video surveillance, videoconferencing, video email, and mobile video. Video is nothing but sequence of image frames transmitted one after the other that to human eye it seems a continuous scene. Very Large computations are required in processing such large amount of video data. Also mass storage space is required to store such data. So more and more compression of Video data and fast quality decoding, is required to meet the fast changing technological needs. During the last thirty years, a tremendous explosion in research and applications in the field of video compression has been witnessed.

The advancements in the image capturing technologies and the Video Display equipment's like HDTV, there is a strong need to develop new standards to support these upcoming technologies. In the last three decades the two groups are working on the standardization of Video Compressed data – ITU (ITU-T Rec. H.261/262/263/264, for application domains of telecommunications), and the Moving Pictures Experts Group (MPEG of ISO standards MPEG-1/-2/-4, for applications in computers and consumer electronics)

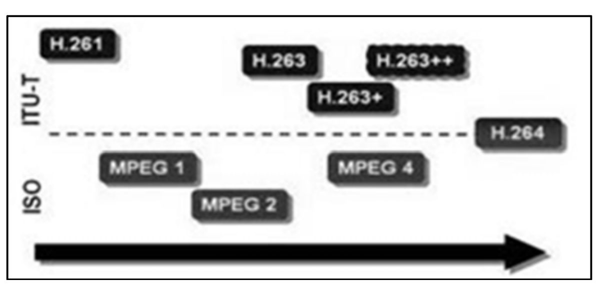

Fig 1: Evolution in video Compression Standards

The ITU-T VCEG and ISO/IEC MPEG standardization bodies have started a new video coding standardization project called High Efficiency Video Coding (HEVC) targeting the reduction of the coding rates up to 50% for the same quality. As H.264/AVC standard does not seem to provide the required compression ratios, needed for the transmission and storage in the currently available facilities. [2]

2. Requirement of Video Compression

Raw Real time Video signal captured is an analog signal. To transmit the analog video signal it is required to first convert it to the digital domain. For example NTSC has 30 frames per second, 858 x 525 luminance samples, 429 x 525 x 2 chrominance samples 8 bits per sample. Therefore, the Bit rate = 30 x 8 x ((858 x 525) + (429 x 525 x 2)) = 216.216 Mbps is required to transmit the video.

A single digital television signal in CCIR 601 format requires a transmission rate of 216 Mbps. For most of transmission communication network this bit rate is very high. So the digital video data need to be compressed (encoded) before transmission so as to efficiently use the networking and storage resources. Table enlightens various video data applications and bit rates required for raw and processed data.

Table 1: Comparison of bit rates before and after compression

Application	Data Rate	
	Uncompressed	Compressed
Video Conference 352x240@ 15fps	30.4 Mbps	64-768 kbps
CD-ROM Digital Video 352x240@30fps	60.8 Mbps	1.5-4Mbps
Broadcast Video 720x480@ 30fps	248.8 Mbps	3-8 Mbps
HDTV 1280x720@60fps	1.33 Gbps	20 Mbps

3. Different Types of Frames in Video Data

A complete video may consist of various scene changes in its entire span and there will not be correct prediction using previous frame as reference frame when there is such scene change. Therefore, entire video is divided into group of pictures (GOPs) with one group continues until there is scene change which is followed by next GOP in that order. These GOPs consists of I (used for intra frame Coding to remove spatial redundancy), P and B pictures (Inter Frame Coding to remove temporal redundancy).

MPEG GOPExample-------- I0 B1 B2 P3B4 B5 P6 B7 B8 I9….. I frames are the intra coded frames, same as normal image coding as in JPEG image compression. But if all the frames are coded as I frames very little compression will be achieved. Since there is very slight difference in the consecutive frames so instead of coding all the frames as I frames, the difference between the previous frame and the current frame is coded and are known as P frames. If along with previous frame forward frame is also used for coding the frame then these are known as B frames.

4. Overview of Video Encoder

In most real video sequences, consecutive frames are very similar to each other (until there is scene change) except movement of some objects within these frames. Since a frame may have multiple objects moving randomly in different directions, motion of these objects cannot be accurately predicted if entire frame is processed as a single unit. Therefore, each frame of input video is divided into fixed size blocks, called macroblocks (MB) and motion of objects is predicted at macroblock level. A current frame block (or macroblock) is searched for its best match in the reference frame within a search window. Location of best match block within the search window in the reference frame with respect to the location of candidate block in the current frame is called Motion Vector. In the current frame Motion Vectors of all blocks are calculated and with the help of these motion vectors a predicted frame is formulated from the reference frame.

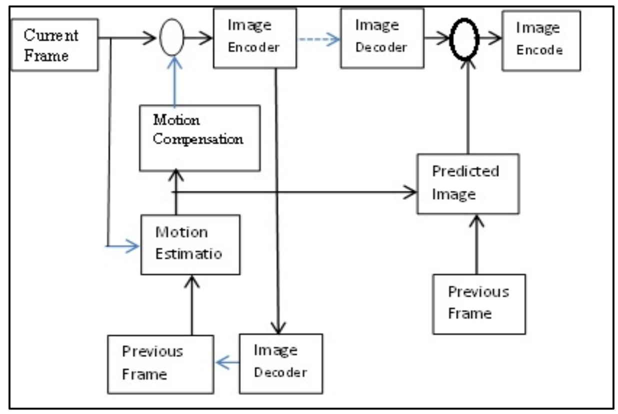

Fig 2: Video Encoder

Finally, the residue error which is the difference between the predicted frame and the actual frame is encoded with an efficient encoding scheme for the purpose of transmission or storage. Finding the Motion Vector is the most time consuming part of the video encoder, In this paper following important factors that have their impact on accuracy and efficiency of motion estimation are discussed

1. Block Matching Criterion,
2. Motion Estimation Algorithm
(a) Fast BMA based on a Fixed Set of Search Patterns
(b) Fast BMA Based on Inter-Block Correlation
(c) Fast BMA Using Subsampled Pixels on Matching
-Error Computations

3. Search Window Size
a) Fixed Search Window Size Selection for Motion Estimation
b) Adaptive Search Window Size Selection for Motion Estimation

4. Block Size Selection
c) Fixed Block Size or Motion estimation
d) Variable Block Size for Motion Estimation

5. Motion Estimation Based on Detecting the Edges of objects in Video Data
6. Accelerating Motion Estimation by Zero Motion Prejudgment

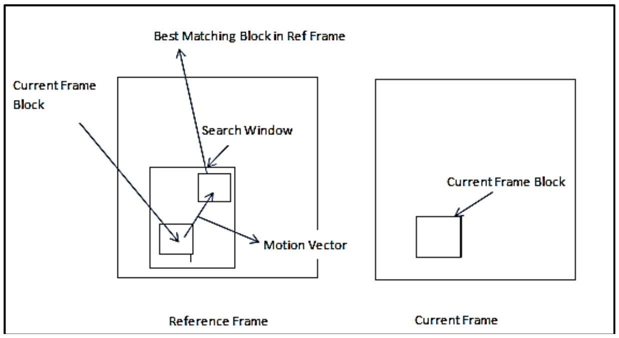

Fig 3: Motion Estimation Based on Block Matching

4.1 Block Matching Criterion

Block Matching criterion gives a means to find the best matching of the current block within the search window in the given reference frame. The criterion should be such that it should give good results even for contrast variations in the scene. Some of the Block matching criterion and their comparison is discussed in this section.

The first proposed criterion [3] was, comparing each pixel of the target block is with its corresponding pixel in candidate block and each pixel pair is classified as matching or not matching depending on the absolute difference value less than some threshold or not. But chosen threshold value has major impact on the performance of this distortion function. Since threshold selection depends on the scene itself, So this criterion was not preferred to be used in existing coding techniques although this criterion requires less hardware.

MAD (Mean Absolute Difference) is the preferred method over other methods because of it is simple and easy in implementation. In this the difference of corresponding pixels from each block are compared and their absolute differences are summed up. [4] The candidate block with the minimum MAD in a given search window is chosen as best match.

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

Where Cij and Rij are the blocks from the current block and the reference block.

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

Mean square difference (MSD) is another block matching criterion. Here, the difference between the corresponding pixels in current and reference frame block is squared and summation of these squares over entire block is taken as similarity measure. The block for which this summation is minimal is the best match block [5]. This although gives better result but at the cost of increased no. of computations

Where Cij and Rij are the blocks from the current block and the reference block.

Average variation problem occurs in the conventional BMAs, Statistically Adaptive Block-matching criterion (SABMC) [6] overcomes this problem by considering time- varying average and shape that represents the fluctuation of pixel values separately, because they are of different importance. Advantage of this technique is that values calculated for a particular candidate block can be reused in calculating the integral projections for overlapping candidate blocks, thus reducing computational cost. This matching criterion to be less susceptible to noise than others but may not be useful for suboptimal searches.

A variant of MAD, called Vector Matching Criterion (VMC) [7] where individual error term has been focused rather than average value over entire block as in MAD. In this technique, a macro block is subdivided into smaller size blocks (authors have taken 2x2 size) and matching is performed at these sub block levels using conventional MAD criterion with a predetermined threshold value. Each macro block is treated as a vector and all its sub blocks as its components. A best match block represents vector which has maximum number of components with their matching error less than a threshold value. To further improve the quality of MAD block variances are included as another feature.

Smooth constrained mean absolute difference (SC-MAD) [8] reduces the number of bits required at encoder. A residue block, obtained by taking absolute difference between the current block and the reference block is divided in to four sub blocks. Difference between the maximum and minimum residue error in each sub block is added which is used as another factor along with conventional MAD as distortion measure.

MAD based matching criterions does not give accurate results for contrast variations. For dealing with this problem, contrast between sub blocks of a given block is added as and interpreted their approach as sub block based sum of absolute hadamard transformed differences. Sub block sum (ignoring sub block variance to reduce computation) feature as intra block feature is used as it is computed within each block individually and contrast among them as inter block feature.

Another approach robust to Contrast variations in the i/p sequence is based on mapping pixel intensities in reference and current frames and matching is performed over these matched pixels. A comparison of Various Existing Block Matching Criterion is shown in Table 3.There is a tradeoff between accuracy and no. of calculations required to find the best match as can be seen from Table 3.

4.2 Motion Estimation Algorithms

There are three kinds of motion estimation algorithms, pixel based, Block based and region based. In pixel based methods, search for the best match is done on a pixel by pixel basis, in block based motion estimation, search is done on a block of nXn pixels, while in region based motion estimation search is done on a region of arbitrary size.

In most of the Video encoding standards like H.261, H.263, MPEG-1, MPEG-2, MPEG-4, and H.264, BLOCK-MATCHING based algorithms are preferred because of its effectiveness and simplicity in hardware implementation as compared to other techniques of motion estimation. Full search (FS) algorithm is the straight forward method to find motion vectors, this exhaustively searches for the best matching block within the search window. FS algorithm, gives best results as it searches all (2p + 1)2 points exhaustively for a search window of size ±p. However the computation cost is very high as it requires large computations and makes ME the main bottleneck in real-ime video coding applications. So fast BMAs are looked out for reducing the computations and fastening the motion estimation. In this section some fast BMA algorithms are discussed.

Table 3: Comparison of Various Motion Estimation Criterion

Author	Publication	Criteria for Block Matching	Advantage	Drawback
Chan *et al*	1994	Pixel of the target block is compared with its Corresponding pixel in candidate block. If the Difference is Greater than threshold, Candidate Block is considered for block matching.	Simple Technique and requires less hardware	Threshold selection Depends on the scene itself.
Kim and Park	1992	Integral Projection based Criteria, Row and column pixels are separately added up	Computational Cost is less, as row and column sum once calculated is reused	Works better for exhaustive search, but not for suboptimal search
Wang & Chen	1999	Marco block is subdivided in smaller size sub blocks. Matching is performed at these sub blocks level.	Better Matching Than MAD	Increased calculations as compared to MAD
Viola & Jones	2001	Block is divided into sub blocks. Integral sum of these Sub blocks is calculated. A candidate sub block is searched for in the reference frame within the search window.	Speed gain of 21-36%over conventional MAD w.r.t. full search, three step search	Not good for Suboptimal search algos.
Jing *et al*.	2003	A block is divided into sub blocks & the difference Between Max. & Min. residue error in each sub block is added along with conventional MAD.	Significant Improvement in required no. of bits at encoder end.	Max and min calculations Requires additional calculations.
Seung Hwan Kim	2004	Motion is estimated by considering time varying average And shape that represents the fluctuation of pixel values separately.	Overcomes the problem of contrast variations	Calculations are Increased.
Nguyen and Tan	2006	Along with Mean absolute differences, Block variances are also added up	Significant improvement in quality	Increase in Calculation because of calculation of variance.
Xing & Zhu	2009	In this criteria four features are used for block matching- Block sum, Horizontal Contrast, Vertical contrast, Diagonal Contrast to find the best Matching Block	Multilevel Structure, Trade off between computation cost and coding efficiency	More calculations
Ravinder *et al*	2009	Similarity measure for block matching has been Proposed. Algo Maps Pixel intensities in reference and current frames and matching is performed over these matched pixels	Robust to Contrast Variations in the i/p sequence.	

4.2.1 Fast BMA Based on Fixed Search Pattern

For finding BMAs using fixed Search Pattern approach, it is assumed that ME matching error decreases monotonically as the search moves toward the position of the global minimum error and the error surface is uni-modal as shown in fig 4. A fixed set of search pattern is used for finding the motion vectors (MV) of each block. A no. of fast motion estimation Algorithms had been designed by many researchers. Here a brief review of these algos are presented.

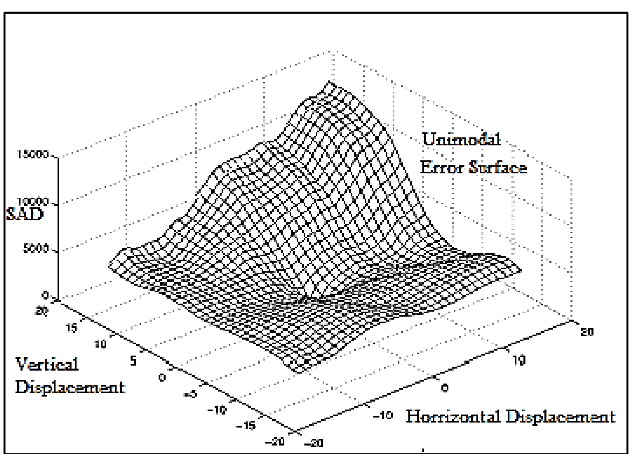

Fig 4: Uni-modal Error Surface

J.R. Jain and A.K. Jain [10] introduced a 2-D logarithmic search for ME. In this technique, search is accomplished by successively reducing half search area in each iteration. Each step consists of searching of five locations which contain center of the SW and the mid points between center and the four boundaries of the area along the axes passing through the center. In each successive iteration, minimum point is used as center point for next step. This process is continued until the plane of search reduces to a 3x3 size where all 9 points are searched for direction of minimum distortion. A new Three Step Search algo suggested by T. Koga [11] for ME. Search is accomplished in three steps only. A step size of 2Log (p+1)-1 is adopted for a SW of size ±p. At each iteration 9 points are checked and the position of the minimal point is the center point of next search. Step size reduces to half in next iteration. TSS always explores 25 search points.

A new refinement of the TSS was next introduced by Li [12] which exploits the center biased characteristics and half way stop to quickly identify the stationary and quasi stationary blocks. In this along with eight points as in TSS eight more points next to the immediate neighbors of the center of SW are checked. If the minimal point is amongst these eight near neighbors search in the next iteration is processed by taking minimal point as center otherwise the algorithm works as normal TSS. This NTSS searches 17 & 33 locations in its best and worst cases respectively. Next 4SS was introduced by Po [13] which possess the center biased and half way stop features of the NTSS. But Authors claimed better performance for complex motions like camera zooming, fast motion etc. In this step size of d/4 is taken for first three steps and further reduces to half in last step. Total no. of SPs vary from min. 17 to max. 27 for SW of size ±7. One at a time Search algorithm [14] uses two adjacent SPs in the horizontal direction at the center of the search window. This process is repeated until the minimum BDM is at the center and then the same process is started in the vertical direction till the minimum BDM is the center point of the search. Orthogonal Search Algo [15] proposed some modification in the OTS. In this firstly two horizontal SPs are checked at d/2 distance and next two vertical SPs are checked around the position of minimum BDM. The step size is halved at each iteration and process is repeated till the SS reduces to one. Gradient Descent Search Algorithm (GDSA) [16] for ME uses 3x3 size checking blocks and all points of this block are evaluated for minimum point. If BDM is found at center, search is stopped otherwise minimum point is treated as center for the next 3x3 checking block. Number of new search points to be evaluated in successive steps may be 3 or 5 depending on whether minimum point is the edge point or corner point of the checking block. Algorithm always moves the search in the direction of optimal gradient descent search where one expects the distortion function to approach its minimum. Authors have claimed competitive performance with reduced computational complexity.

Zhu [17] analyzed the motion vector distribution of various video sequences using full search and found that up to 98% motion vectors are enclosed within a radius of 2 from the center of search window. Using these observations a new algorithm called diamond search (DS) was proposed. This algorithm employs two search patterns– large diamond search pattern (LDSP), comprising nine checking points in total and small diamond search pattern (SDSP) with five checking points. For searching, LDSP is repeatedly used until the center point becomes the minimum distortion point. Several modifications to DS have been suggested by many authors for small motion based sequences like videoconferencing. Hexagon based search algorithm (HEXBS) [18] is based on the same search pattern as DS but in this large Hexagon is used instead of LDPS. Further a New Adaptive Fast BMA [19] was suggested which combines TSS, DS and BBGDS algos to track slow, medium and fast motion MBs. At ZMP BDM values are calculated and decision about the motion is made on this basis. All MBs are sorted according to the initial BDM values

The most important advantages of these fixed search based methods are their Simplicity and regularity. These algorithms reduce the computational loads. But the main disadvantage of these methods is that these are less adaptive and less efficient for tracking large motions and provide suboptimal solutions only. These algorithms tend to be trapped in local minima when motion does not match the predefined pattern.

In paper [20] FS, TSS, 4SS, NTSS, BBGDS and DS are by compared using two 100 frames QCIF (174 X144) ("Foreman" and "Mother and Daughter") and four 300-frame SIF (352X240) ("Football, "Flower Garden, "Table Tannis",and "Mobile Calendar") video sequences. These sequences contain different type of motion activity from low through medium to high motion. It is clear from the table 4 that there is no single algorithm which is performing equally for all type of motions However BBGDS and DS are performing better in terms of no. of computations as compared to other algorithms.

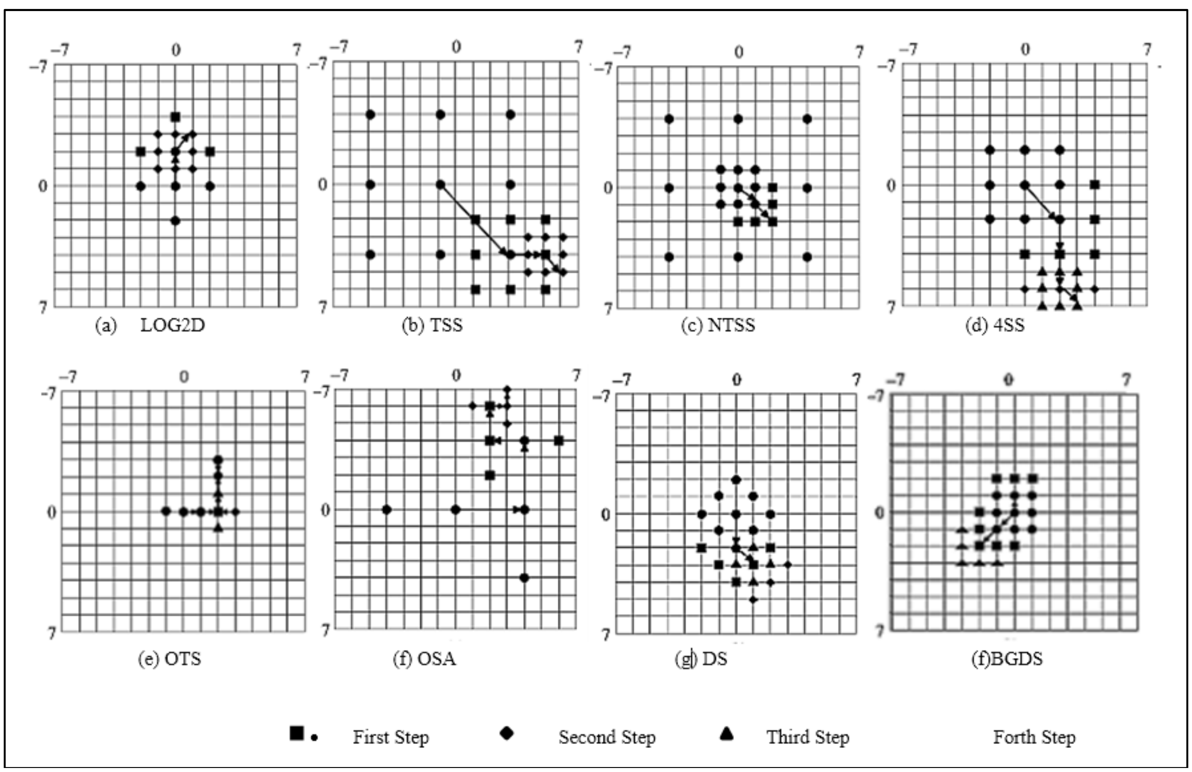

Fig 5: Examples of Search Path

Table 4: Complexity and Performance Comparison of FS, TSS, 4SS, NTSS, BBGDS and DS [20]

Sequence	Video Type And SR	Motion Type	Comparison Strategy	Algorithms					
				FS	TSS	4SS	NTSS	BBGDS	DS
Flower Garden	CIF /±15	Medium	SPs	859.45	30.52	19.03	20.54	14.03	16.89
			PSNR[dB]	29.92	21.18	23.39	21.93	23.52	23.61
Football	CIF /±15	High	SPs	859.45	30.52	19.19	21.66	15.15	17.11
			PSNR[dB]	23.08	21.82	22.27	22.02	22.08	22.24
Foreman	QCIF/±7	Medium	SPs	184.56	21.66	16.70	18.37	12.47	14.50
			PSNR[dB]	30.65	30.16	29.83	30.23	29.73	29.83
Mobile Calendar	CIF /±15	Medium	SPs	859.45	30.52	15.88	18.04	10.73	12.60
			PSNR[dB]	22.61	22.31	22.51	22.58	22.58	22.54
Mother & Daughter	QCIF/±7	Low	SPs	184.56	21.66	14.94	15.42	8.43	11.85
			PSNR[dB]	39.71	39.59	39.63	39.65	39.62	39.63
Table Tennis	CIF /±15	High	SPs	849.45	30.52	18.31	20.17	12.71	15.54
			PSNR[dB]	29.49	25.82	26.42	26.26	26.25	26.61

4.2.2 BMA Algorithms based on Inter Block Correlation
These methods are based on the correlation between the current block and its neighboring blocks. Statistical Averages of the neighboring MVs like (Mean, Median, Weighted Mean etc.) are used for predicting the MV [21]. Based on the prediction, the size of search window and the search center are re-defined, and FS is then performed within this new search window. Fast BMAs instead of FS are used in some methods, around the predicted MV without re-defining the search window. The methods have quite appreciable performance only at the expense of computing the prediction. In these methods additional memory for storing the neighboring MVs is required

4.2.3 Fast BMA Based on Sub sampled Pixels on Matching-Error Computations
One method uses only a fraction of pixels within a block by Performing 2: 1 pixel sub sampling in both horizontal and vertical directions. So the total computations can be reduced effectively by a factor of 4. Such computation reduction methodology can be incorporated into other BMAs and gives higher computational gain.

4.3 Adaptive Selection of size of Search Window for Fastening Motion Estimation
In an image an object occupies more than one block, motions of these neighboring blocks are highly correlated. Also in the blocks of consecutive frames due to the inertia of the moving objects, there is correlation among motion vectors. If the MVs of the previously computed MBs are all large then it is very much possible that the current MB may have large MV. However, the fact is, high probability of large MV does not mean a large search window is required. The size of the search areaeffects the calculation for finding motion vector. The correlation between neighboring blocks can be used to predict the shape and the size of the search window. Predictive search area [25] (A smaller or bigger search range in doing the ME) can, not only reduce the execution time required but also improves he accuracy of motion estimation.

4.4 Varying Block Size for improving the quality of Video
The performance of the motion estimation is affected by the Size of the blocks. Small block sizes have good approximation to the natural object boundaries and also provide good approximations to real motion. But a large amount of raw motion information is produced with small block sizes. This increases the number of transmission bits. From performance point of view, small blocks also suffer from object ambiguity problems, which means similar objects may appear at multiple locations inside a picture and may lead to incorrect displacement vectors and random noise problem.

Large block size may produce less accurate motion vectors since a large block may likely contain pel moving at different speeds and directions. The employment of the Variable Block size motion estimation techniques shows a significant change in quality of the video and therefore implemented advance video coding standards like H.264 [26-29].Available algorithms take lots of time for processing variable block sizes. A no. of new algorithms have been proposed for reducing the computations and adaptively selecting the block sizes.

4.5 Improvement in Quality of BMA by considering the Edges of Objects in Video Data
For exploiting temporal redundancies Intensity based block motion estimation algorithms are widely used [30] in video coding. But blocks located on boundaries of moving objects are not estimated accurately, which are very sensitive to the human eyes. So Edge detecting techniques when included with motion estimation improves the quality of the video.

4.6 Accelerating Motion Estimation by Zero Motion Prejudgement
Zero Motion Prejudgment [31] (ZMP) is the way to find the static macroblocks which contains zero or no motion. In most of the real world video sequences more than 70% of the MBs are static which do not need the remaining search. If we predict the static macroblocks by ZMP procedure before starting motion estimation procedure, significant reduction of computation is possible and the remaining search will be faster and saves memory.

5. Conclusion
Motion Estimation is the key component in most of the Video Data Processing based Applications. There are a number of factors that help in increasing the accuracy and efficiency of Motion Estimation in Video data. Motion Estimation Computational complexity is a major issue in developing the algorithm. Complexity depends upon the number of search operations performed to find the approximate motion vector for each block in a defined search window.

6. References

1. Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-TRec.H.264/ISO/IEC14496- 10AVC in Joint Video

Team (JVT) of ISO/IECMPE Gand ITU-TVCEG, JVT G050, 2003.
2. Francesca De Simone, Lutz Goldmann, Jong-Seok Lee, Touradj Ebrahimi. Towards high efficiency video coding: Subjective evaluation of potential coding technologies. Original Research Article Journal of Visual Communication and Image Representation. 2011; 22(8):734-748
3. Chan E, Arturo A, Ghandi R, Panchanathan S. Experiments on block matching techniques for video coding. Journal of Multimedia System. 1994; 2(5):228-241.
4. Koga T, Iinuma K, Hirano A, Iijima Y, Ishiguru T. Motion compensated interframe coding for video conferencing. In Proceedings of NTC81. New Orleans, LA, 1981
5. Fadil M, Dennis T. A hierarchical motion estimator for Inter frame coding. In IEEElectronics Division Colloquium onApplications of Motion Compensation, 1990.
6. Jing X, Zhu C, Chau L. Smooth constrained motion e stimation for video coding. Elsevier Journal of Signal Processing. 2003; 3:677-680,
7. Wang S, Chen H, An improved algorithm of motion compensa-tion mpeg video compression. In International Vehicle Electronics Conference, IEEE, 1999, 1.
8. Jing X, Zhu C, Chau L. Smooth constrained motion estimation for video coding Elsevier Journal of Signal Processing. 2003; 83:677-680.
9. Kim J, Park R. A fast feature-based block matching algorithm using integral projections. IEEE Journal on Selected Areas in Communications. 1992; (5):968-971.
10. Ravinder Kr Purwar, Nupur Prakash, Navin Rajpal. A Matching Criterion for Motion Compensation in the Temporal Coding of Video Signal International Journal of Signal Image and Video Processing (SIViP), Springer Verlag, 2009, DOI: 10.1007/s11760-009-0149-9.
11. Jain JR, Jain AK. Displacement measurement and its application in interframe image coding IEEE Transactions on Communications. 1981; 29(12):1799-1808.
12. Koga T, Iinuma K, Hirano A, Iijima Y, Ishiguru T. Motion compensated interframe coding for video conferencing. In Proceedings of NTC81. New Orleans, LA, 1981.
13. Li R, Zeng B, Liou M. A new three step search algorithm for Block motion estimation. IEEE Transactions on Circuits and Systems for Video Technology. 1994; (4):438-442.
14. Po L, Ma W. A novel four step search algorithm for fast block Motion estimation. IEEE Transactions on Circuits and Systems for Video Technology. 1996; 6(3):313-317.
15. Mei-Juan Chen. One-dimensional full search motion estimation algorithm for video coding", IEEE Transactions on Circuits and Systems for Video Technology. 1994, 1051-8215
16. Shilpa P Metkar. Fast motion estimation using modified orthogonal search algorithm for video compression", Signal, Image and Video Processing. 2010; 4(1):123-128.
17. Liu L, Feig E. A block based gradient descent search algorithm for block motion estimation in video coding IEEE Transactions on Circuits and Systems for Video Technology. 1996; 6(4):419-422.
18. Zhu S, Ma K. A new diamond search algorithm for fast block Matching motion estimation IEEE Transactions on Image Processing. 2000; 9(2):287-317.
19. Zhu C, Lin X, Chau L. Hexagon based search pattern for fast block motion estimation IEEE Transactions on Circuits and Systems for Video Technology. 2002; 12(5):349-355.
20. Huang SY, Cho CY, Wang JS. Adaptive fast lock matching algorithm by switching search patterns for sequences with wide range motion content, IEEE T Circ Syst Vid. 2005; 15:1373-1384
21. Hsieh CH, Lu PC, Shyn JS, Lu EH. Motion estimation algorithm using interblock correlation, Electron. Lett. 1990; 26(5):76-277
22. KL Chung. A New Predictive Search Area Approach for fast block estimation IEEE Transactions on image processing. 2003; 12(6):648-652.
23. Shih-Chang Hsia, Wei-Chih Hsu, Shih-Ren Wu. A fast rate distortion Optimization algorithm for H.264/AVC codec, Springer, published Online Dec 2011
24. Multiple Block Size Search Algorithm for Fast Block Motion estimation, ICICS, 2009.
25. Ben ayed MA, Samet a, Marzouk s, Souissi n, Ben m, Masmoudi n. New variable block-size motion estimation algorithm for H.264/AVP, ISCCSP, 2006.
26. Yamada T. Fast and accurate motion estimation algorithm by adaptive search range and shape selection, Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05), IEEE Press. 2005; 2:897-900, doi:10.1109/ICASSP.2005.1415550.
27. Yang L, Yu K, Li J, Li S. An effective variable block size early termination algorithm for H.264 video coding, IEEE Trans. Circuits Syst. Video Tech. 2005; 15(6):784-788, doi: 10.1109/TCSVT.2005.848306.
28. An-Chao Tsai. Effective Search Point Reduction Algorithm and Its VLSI Design for HDTV H. 264/AVC Variable Block Size Motion Estimation, IEEE Transactions on Circuits and Systems for Video Technology, 2012; 22(7).
29. YUI-Lam Chan. Block Motion Vector Estimation using Edge Matching : An Approach with Better Frame Quality as Compared to Full Search Algorithm, IEEE International Symposium on Circuits and Systems, Hong Kong, 1997, 9-12.
30. Phat Nguyen. Efficient Motion Estimation Algorithm Using Edge Feature and Arithmetic Coding for Video Compression on WVSNs, International Conference on Software, Telecommunications and Computer Networks, 2011.
31. Amrita Ganguly, Anil Mahanta. Improved mode decision algorithm based on residues And early zero block detection in H.264/AVC, SIViP (Springer-Verlag, London), Online Publication, 2012.
32. Ravindra Kr Purwar, Navin Rajpal. A fast block motion estimation algorithm using Dynamic pattern search" Springer published Online, Dec 2011.