International Journal of Research in Advanced Engineering and Technology
ISSN: 2455-0876; Impact Factor: RJIF 5.44
www.engineeringresearchjournal.com
Volume 2; Issue 3; May 2016; Page No. 165-167

Variable search windows for slow, Medium and fast motion sequences

Shaifali M Arora

Asst. Prof, Maharaja Surajmal Institute of Technology, New Delhi, India

Abstract

Search Window plays a very important role in the determination of Motion Vectors in motion estimation. Correct choice of the window size may lead to reduction in the number of computations required to find motion vectors without compromising with the quality. In case of Fast motion large search window may be required to find accurate motion vectors and for slow or medium motion smaller search window may give good results. The current work examines the various search window sizes for slow, medium and fast motion sequences. Fixed Search pattern based algorithms that are used for motion estimation are Full Search (FS), Three Step Search (TSS), New Three Step Search (NTSS), Diamond Search(DS) and Adaptive Rood Pattern Search(ARPS) with different search window sizes of ±2, ± 3, ± 4 and ±7 for the analysis. The conclusions are drawn on the basis of PSNR and the number of computations. The simulation results show that the computations are reduced drastically with an appropriate search window size without compromising at the quality of video.

Keywords: Full Search (FS), Three Step Search (TSS), Diamond Search (DS), Adaptive Rood Pattern Search (ARPS)

1. Introduction

The tremendous use of videos in every sphere of life has led to a growing need of fast processing video data which is expensive in terms of computations and storage requirements. This is the reason for an extensive and ongoing research in video compression leading to existing standards such as MPEG-1, 2, 3 and 4 and algorithms in this field. Motion estimation is a key component in the entire process of video compression. It is the process of predicting the motion between two successive frames. It is very effective in reducing the temporary redundancy between two consecutive frames.

From the past three decades many different systems and algorithms have been proposed to fasten the process of Motion estimation. These algorithms can be categorized as pixel matching algorithms, region based algorithms and block motion estimation based algorithms. The pixel matching algorithms are thresholding based and also they do not give an adaptive solution for different scenes thus cannot be used in practice. Region based algorithms are very complex and involve large amount of computations. The most versatile and robust algorithms are block based motion estimation algorithms. 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 can be divided into fixed size blocks say Bc with Size b X b, called macro blocks (MB) and motion of objects is predicted at macro block level. A current frame block (or macro block) is searched for its best match in the reference frame within a search window of size (2W+1) X (2W+1). The Matching criterion or cost function used is MAD.

$$MAD\ (V_x,V_y)=\sum_{x=1}^{b}\sum_{y=1}^{b} Bc(x,y) - Br(x+Vx, y+Vy)$$

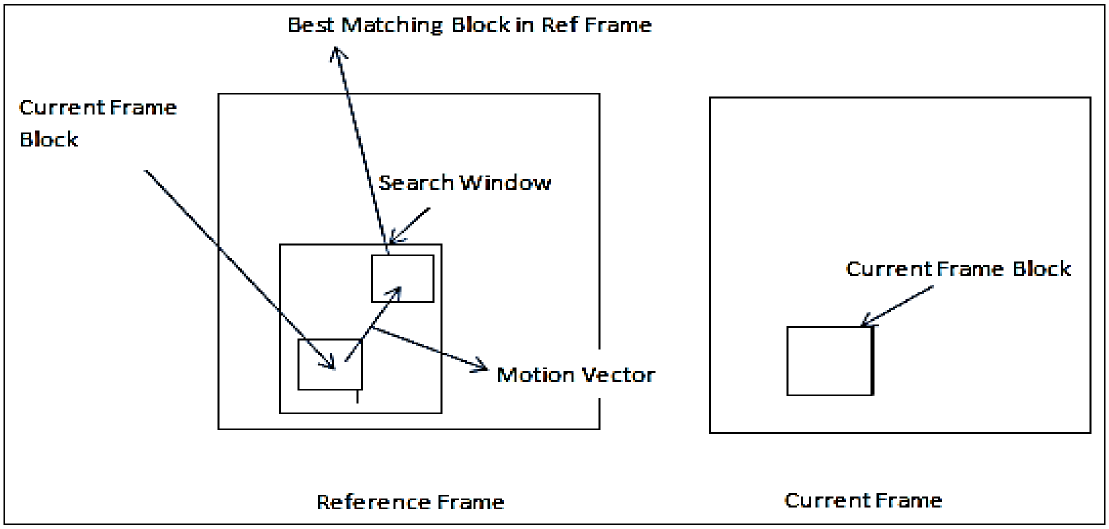

Fig 1: Motion Estimation Based on Block Matching

Where $-W \le Vx, Vy \le W$ where Br denotes the reference block and Bc denotes the current frame block. For finding best matching block current block is matched with all the reference blocks in a given search window. This is a computationally costly technique known as Full Search Algorithm. To maintain the same quality as that of full search researchers tried to reduce the computations and lots of fixed search pattern based algorithms were developed. Some of these algorithms like - Full Search (FS), Three Step Search (TSS), New Three Step Search (NTSS), Diamond Search and Adaptive rood pattern search, are used in the current work. All these algorithms use a fixed search window for slow, medium or fast motion activity based video sequences. But appropriately choosing the size of window may reduce the number of computations.

Based on the motion activity in the consecutive frames, video sequences could be broadly divided in to three categories- slow, medium and fast video sequences. The standard available test sequences of each of these types is taken for estimating the quality and the computations by varying the search window sizes in this paper. The search window sizes that are taken for the analysis are ± 2, ± 3, ± 4 and ±7. The performance of each of the sequences using the mentioned algorithms is compared for each of the previously defined sizes of windows. Performance measures on which analysis is based are PSNR values and average computations.

The next section, section 2 gives a brief review of motion estimation algorithms. Section 3 gives the simulation results and section 4 gives the conclusion on the basis of simulation results.

2. Fixed Search Pattern Based Motion Estimation algorithms

To achieve high compression maintaining the quality and reducing the computations has always been an active area of research. The motion estimation is the most expensive operation in the process of compression and thus has been of great research interest. Various fixed pattern based algorithms have been developed in this field. These methods are based on the assumption 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

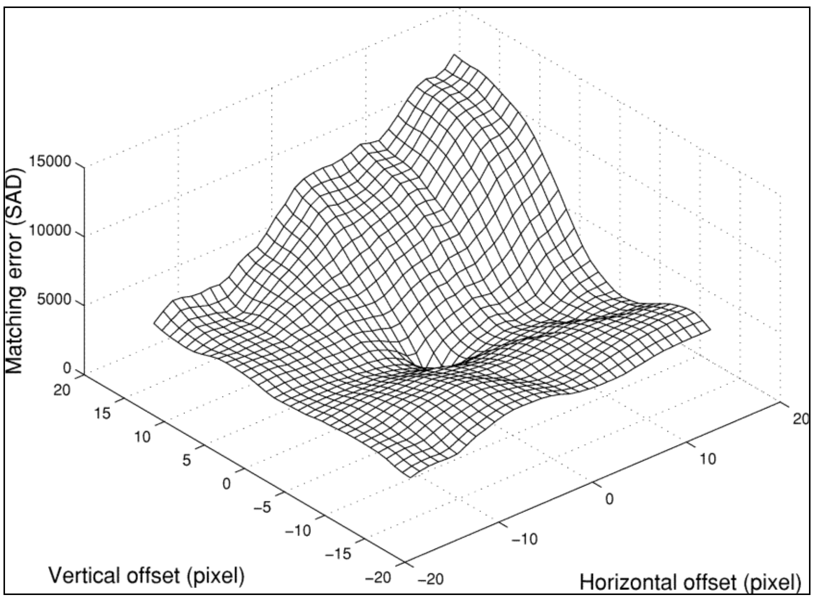

Fig 2: Uni-modal error Surface

Full Search algorithm [1] is the simplest of all algorithms. It evaluates the cost function at every location in the search area and thus gives highest PSNR and so best picture quality but this is achieved at the cost of number of computations.

Various fast algorithms have been proposed which reduce the computational complexity with the help of search point sampling accomplished with their inherent search pattern. Three step search (3SS) [2] given by Koga is based on a coarse to fine approach giving optimal performance with less complexity. But it is not able to detect small motion and also is based on the assumption of uni-modal error surface. New three step search (N3SS) [3] came as a modification to 3SS in which eight additional points are searched in the first step making it more efficient for small motion in terms of complexity but at the same time it tends to be more complex for large motion vectors. Four step search (4SS) [4] uses a smaller initial step size in comparison to 3SS and thus is more efficient for small motion vectors but is more complex than 3SS. Diamond search [5] uses the concepts of 4SS but the search point pattern is diamond rather than square and there is no limit on the number of steps. The result is high PSNR value and fewer computations.

The main problem with the above mentioned techniques is the degradation of quality due to local minima. This problem is avoided using adaptive rood pattern search [6] which uses different search patterns according to the estimated motion behavior for the current block.

The current work uses all the six above mentioned algorithms for analyzing the effect of varying search windows in slow, medium and fast motion sequences

3. Simulation Results

Standard sequences for slow, medium and fast motion namely Container sequence, Car Phone sequence and Flower sequence are taken for simulation. These sequences are tested using the fixed search based motion estimation algorithms namely Full

Search (FS), Three Step Search (TSS), New Three Step Search (NTSS), Diamond Search and Adaptive rood pattern search. PSNR and average number of computations per frame are calculated for 30 frames. Search window sizes that are used in the analysis are ± 2, ± 3, ± 4 and ±7.

Table1: Comparison of PSNR and No. of Computations for Different Motion Activity Sequences using Variable Search Window

Motion Estimation Techniques	Class A Container		Class B Car Phone		Class C Flower	
	Computations Required per Frame	Avg PSNR per Frame	Computations Required per Frame	Avg PSNR per Frame	Computations Required per Frame	Avg PSNR per Frame
Full Search p=7	177.7202	30.0907	177.7202	22.6994	200.5836	13.2073
p=2	20.3389	30.06979	20.3389	20.9981	21.9613	12.5419
p=3	39.3647	30.0784	14.4137	21.5296	42.7555	12.8431
p=4	60.2234	30.08331	60.2234	21.9199	64.3423	13.1252
TSS p=7	20.8556	30.065	21.2798	22.5229	23.0702	13.4342
p=2	7.5383	30.0184	7.5383	20.3258	8.0305	12.1818
p=3	14.2106	30.0711	18.7298	21.4765	15.3361	12.7805
p=4	16.3423	30.0711	19.2248	21.4765	17.2134	12.7805
NTSS p=7	16.3427	30.0834	21.4276	22.5198	27.5954	13.4082
p=2	16.3427	25.4441	19.2313	19.2313	11.6854	11.6854
p=3	16.1597	30.0774	18.7298	21.5046	21.578	12.7762
p=4	16.9987	30.0774	18.9298	21.5046	21.688	12.7762
FSS p=7	15.3423	30.0726	18.2174	22.4221	27.5954	13.2644
p=2	15.3423	30.07256	18.2173	22.4221	21.4713	13.3644
p=3	15.3423	30.07256	18.2173	22.4221	21.4713	13.3644
p=4	15.3423	30.07256	18.2173	22.4221	21.4713	13.3644
DS p=7	12.9357	30.0856	17.73696	22.4767	22.5836	13.2073
p=2	11.3445	30.0693	11.5712	20.9815	12.3789	12.52
p=3	12.2779	30.07668	13.5914	21.4969	15.3341	12.77
p=4	12.4279	30.0803	13.2114	21.8622	15.5341	12.9595
ARPS p=7	6.9431	30.0816	11.2798	22.4278	13.4917	13.3071
p=2	6.4579	30.067	8.0056	20.9583	9.2755	12.5083
p=3	6.722	30.0735	8.9326	21.4616	10.6213	12.7481
p=4	6.833	30.0759	8.9326	21.8137	10.6213	12.9549

4. Conclusion

Various search window sizes are taken to compare the quality and computational cost of slow, medium and fast motion sequences. The simulation results show that the appropriate choice of search range has great impact on the complexity in terms of the number of computations i.e. if search range is chosen appropriately for motion estimation then the computation time can be reduced significantly with little or no change in performance given by PSNR value.

5. References

1. Ahmadi M, Azadfar M, "Implementation of fast motion estimation algorithms & comparison with full search method in H.264", IJCSNS International Journal of Computer Science & Network Security. 2008; 8(3):139-143.
2. Koga T, Iinuma K, Hirano A, Iijima Y, Ishiguru T. "Motion compensated interframe coding for video conferencing", in Proc of NTC81. New Orleans, LA, 1981, C9.6.1-C9.6.5.
3. Li R, Zeng B. Liou ML. "A new three step search algorithm for block motion estimation", IEEE Transactions on Circuits and Systems for Video Technology. 1994; 4:438-442.
4. 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.
5. 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.
6. Yao Nie, Kai-Kuang Ma. "Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation", in IEEE Trans. on Image Processing. 2002; 11(12):1442-1448.