International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 1– No.6, February 2012 – www.ijais.org

Comparative Analysis of Exhaustive Search Algorithm with ARPS Algorithm for Motion Estimation

Kaushik M.
Department of Instrumentation Technology
Proudhadevaraya Institute of Technology
Hospet, Karanataka, India

ABSTRACT
Accurate motion estimation is a key factor for achieving enhanced compression ratio. It is the process of determining an offset to a suitable reference area in previously coded frame and has a significant effect on performance of coders and decoders (CODEC). This paper compares the performance of Adaptive Rood Pattern Search (ARPS) motion estimation algorithm with Exhaustive search (ES) algorithm in terms of computational complexity being applied on MATLAB platform for different dimensional images. Several significant parameters such as number of computations and cost to determine the motion vectors are determined by applying to different dimension images.

General Terms
Compression, Multimedia, Spatial domain, Temporal domain,

Keywords
Advanced Video Coding (AVC), Motion Picture Experts Group (MPEG), High Definition Television (HDTV), Motion Estimation, Super High Definition Television (SHDTV)

1. INTRODUCTION
Recent developments in the multimedia systems brought a new dimension to the potential market and have become an integral part of communication environment.

Multimedia applications require dynamic handling of data since it involves massive amounts of data storage. Thus, in order to manage such large multimedia data objects efficiently data compression is required. Data compression algorithms attempts to eliminate redundancies in the pattern of data. Multimedia data in the form of video is just a sequence of images. Successfully encoding a single image with a particular algorithm, and then applying the same algorithm to each image in succession directs towards video compression but at the cost of quality. Thus necessitating CODECs.

Design of faster and more efficient coders and decoders (CODECs) involves exploiting the property of redundancy. The redundancy that exists in the temporal domain is exploited by a technique called Motion Estimation. MPEG achieves its high compression rate by the use of Motion estimation and compensation. MPEG exploits the mere difference that exists between frame to frame and hence comparing macro block areas between frames and also instead of encoding the whole macro block again, the difference between the two macro block is encoded and transmitted. Motion estimation attempts to obtain the motion information for various regions in a scene by coding an n X n pixel block in the current frame by a motion vector followed by the Discrete Cosine Transform (DCT) coefficients of the estimated errors [1].

2. MOTION ESTIMATION
Accurate motion estimation is a key component in a high quality video compression. Motion estimation block exploits the temporal redundancies through the prediction of current frames using the stored information of the past frames. The video captured is converted into frames and stored in the memory. The frame to macro block unit of motion estimation accesses each of the frame and considers frame at time instant't' as 'current frame' while at time instant 't-1' as 'reference frame'. Both reference and current frames are further divided into macro blocks of sizes 16 X 16, 8 X 8, 4 X 4 based on the motion activity.

The motion estimation block works only if there is a past frame stored. In order to encode the first frame in a video sequence intra frame redundancy can be exploited. The frames that use only intra frame redundancy for coding are referred to as the Intra *Coded Frames*. The first frames of every video sequence are an intra coded frames. Then from second frame on wards, both spatial and temporal can be exploited. The frames that uses inter frame redundancy for data compressions are referred to as Inter *Coded Frames*.

One may consider a pixel belonging to the current frame, in association with its neighborhood as the candidates and then determine its best matching position in the reference frame. The difference in position between the candidates and its match in the reference frame is defined as the *Displacement Vector* or *Motion Vector*. These motion vectors can be used to predict the current frame by applying the displacements to the reference frame. This is achieved in the *Motion Compensation* block.

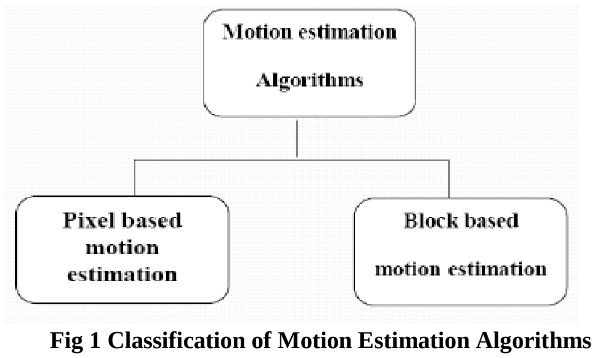

Fig 1 Classification of Motion Estimation Algorithms

Fig 1 represents two basic approaches to motion estimation. Pixel based motion estimation algorithms evaluate the displacement of each pixel individually and do not require the transmission of motion information but recursively use the luminance changes to find motion information. This approach seeks to determine motion vectors for every pixel in the image and referred to as optical flow method. However, the drawback of this algorithm is involvement of overwhelming information

3. BLOCK MATCHING ALGORITHMS

Several block matching algorithms like Exhaustive search (ES), Three step search (TSS), New Three step search (NTSS) [4], Simple and efficient search (SES) [6], Four step search (4SS) [7], Diamond search (DS) [8], Adaptive rood pattern search (ARPS) [1] have been proposed till now. Further part of the paper deals with the computational complexity of ES algorithm is compared with ARPS algorithm.

3.1 Exhaustive Search algorithm

The most straight forward Block Matching Algorithm (BMA) is the Exhaustive search algorithm. In this algorithm in order to find the best matching block, it exhaustively searches all the macro blocks with in the search window to find the best match. The implementation detail in [2] shows that though exhaustive search method is accurate but extremely computationally complex especially when applied to HDTV and SHDTV signals. Thus becomes main bottle neck in real time video applications.

In our previous work [2], Implementation details include conversion of video in to frames and its storage in the memory. Among them first frame is taken as reference frame after which few of the remaining frames are considered as current frame where current frames are compared with the reference frames to exploit the redundancy that exists between the frames. Then a macro block from current frame, considered as candidate block is taken and compared with the macro block of reference frame to determine the motion, in case of backward motion estimation while a macro block from a reference frame is considered as candidate block and compared with the macro block of current frame to determine the motion, in case of forward motion estimation. Let X i,j be the N X N pixel reference block located at coordinates (i, j) , Y i+k,j+1 be the N X N pixel candidate block at coordinates (i+k,j+1) and p be the maximum displacement. The search area is therefore, of the size $(n+2p)^2$. The current frame (t) is divided into overlapping reference blocks. Each reference block in frame (t) is compared with the candidate blocks with in a search area in the previous frame (t-1) in order to obtain the best match. Once best match is found then those motion vectors are stored that helps in constructing compensated image.

3.2 Adaptive Rood Pattern Search algorithm

This algorithm consists of two sequential search stages namely initial search and refined search. For each of the macro block initial search is performed only once at the beginning in order to find a good starting point then followed by refined search. For the initial search stage, an adaptive rood pattern (ARP) is proposed and ARP's size is dynamically varied for each MB, based on the available motion vectors of the neighboring MB's. Further in the refined search, a unit size rood pattern is exploited until the final motion vector is found. [1]

This algorithm tries to exploit the coherency that exists in a frame. If the macro block around the current macro block moved in a particular direction then there is a maximum probability that the current block also have a similar motion vector. It uses the motion vector of the macro block to its immediate left to predict its own motion vector. It checks at a rood pattern distributed points as shown in fig 2. The main structure of the ARP takes the rood shape and its size refers to the distance between centre point and other vertex point. If we observe several video sequences, it can be said that motion vector distribution is almost in horizontal or vertical directions when compared to other directions. Since rood pattern spreads in both vertical & horizontal directions, it can quickly detect the motion vectors and also can jump directly into the local region. This rood pattern search is always the first step; it directly puts the search in an area where there is a high probability of finding a good matching block. The point that has the least weight becomes the origin for subsequent search steps and the search pattern is changed to Small Diamond Search Pattern (SDSP). The procedure keeps on doing SDSP until least weighted point found to be at the centre of the SDSP. The main advantage of this algorithm is that if the predicted motion vector is at (0, 0) then it does not waste comp time in doing Large Diamond Search Pattern (LDSP) it rather directly starts using SDSP. Symmetric shape of ARP is another advantage which is useful in terms of hardware implementation

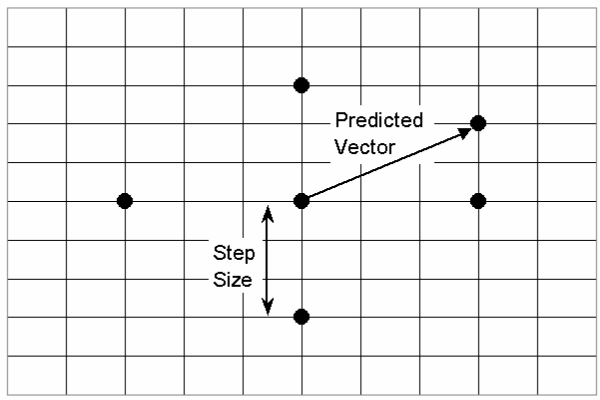

Fig 2 Adaptive Rood Pattern[1]

4. EXPERIMENTAL RESULTS

During this project both ES and ARPS algorithms are applied on various resolution images from 188 X 188 to Super High Definition Television images. The results of both the algorithms being applied on two different resolution images are tabulated respectively in Table 1, 2, 3, 4. Same algorithms are further implemented with different macro block sizes which clearly show the increase in the number of computations thus giving the efficiency of the algorithm. Hardware implementation of ES algorithm [2] shows the several limitations of the algorithm. Also once the ES algorithm is applied to high definition signals, computations and time consumed to compute the required motion vectors are too high.

Table 1 Exhaustive search algorithm results when applied to 188X 188 dimensional images

Mb size	Mb count	Computations	Vectors	ES computations
4	2210	4.6923 e+005	2209	212.42
8	530	1.1223 e+005	552	212.5
16	122	24,964	138	206.31
32	26	4,624	34	184.96

Table 2 ARPS algorithm results when applied to 188X 188 dimensional images

Mb size	Mb count	Computations	Vectors	ARPS computations
4	2210	21,079	2220	9.5423
8	530	4640	4640	8.7713
16	122	967	967	7.9917
32	26	185	185	7.4

Table 3 Exhaustive search algorithm results when applied to 3072X2304 dimensional images

<table>
<tr><th>Mb size</th><th>Mb count</th><th>Computations</th><th>Vectors</th><th>ES computations</th></tr>
<tr><td>4</td><td>4.4237 e^{+005}</td><td colspan="3" rowspan="4">Computations ,Vectors are very high and time duration taken to calculate is also very high (< 24 hours) to just calculate motion vectors that generates compensated image only between frame1 and frame2 .Hence it is not tabulated and not recommended for real time applications</td></tr>
<tr><td>8</td><td>1.1059 e^{+005}</td></tr>
<tr><td>16</td><td>27649</td></tr>
<tr><td>32</td><td>6913</td></tr>
</table>

Table 4 ARPS algorithm results when applied to 3072X2304 dimensional images

Mb size	Mb count	Computations	Vectors	ARPS computations
4	4.4237 e^{+005}	7.0812 e^{+006}	4,42,368	16.008
8	1.1059 e^{+005}	1.8636 e^{+005}	1,10,592	16.851
16	27649	4.4634 e^{+005}	27,648	16.1437
32	6913	1.023 e^{+005}	6912	14.801

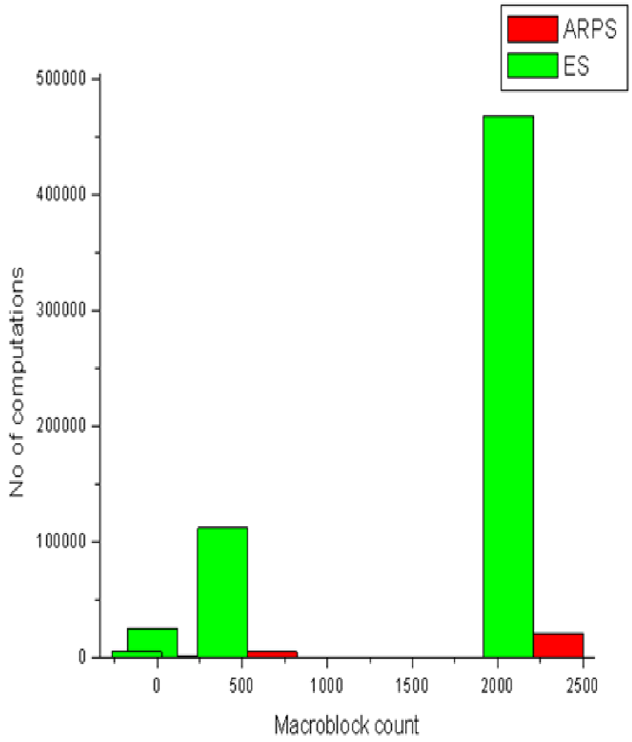

Fig 3. Comparative analysis of ARPS with ES algorithm in terms of computational complexity shown for 188 X 188 dimensional image.

5. CONCLUSION

Though ES algorithm is a bench mark and accurate algorithm, but its application can be restricted to lower dimensional images. However when applied to high definition signals, the processing time increases exponentially and the concept of efficient compression will be ruled out. Thus ARPS algorithm is most suitable for higher dimensional images also the accuracy obtained by the reconstructed image is fairly good to adopt into CODEC design

6. ACKNOWLEDGMENTS

I would like to thank our beloved Principal Dr. P. Khageshan and management of PDIT for providing learn and grow environment.

7. REFERENCES

[1] Yao Nie and Kai-Kuang Ma 2002 Adaptive Rood Pattern Search for Fast Block Matching Estimation. IEEE Transactions on Image Processing, vol 11 No 12, 1442-1447

[2] Mr. Kaushik M, Dr. R. M. Banakar, Performance Analysis of Full Search Grid Based Motion Estimation Algorithm for SHDTV Applications, 2011 IEEE ICCIC conference at Levingipuram

[3] Aroh Barjatya, Block Matching Algorithms For Motion estimation

[4] Renxiang Li, Bing Zeng, and Ming L. Liou, "A New Three-Step Search Algorithm for Block Motion Estimation", IEEE Trans. Circuits And Systems For Video Technology, vol 4., no. 4, pp. 438-442, August 1994.

[5] Hangu Yeo and Yu Hen Hu A Modular Architecture for Real Time HDTV Motion Estimation with Large Search Range

[6] Jianhua Lu, and Ming L. Liou, "A Simple and Efficent Search Algorithm for Block-Matching Motion Estimation", IEEE Trans.Circuits and Systems for Video Technology, vol 7, no. 2, pp. 429-433.

[7] Lai-Man Po, and Wing-Chung Ma, "A Novel Four-Step Search Algorithm for Fast Block Motion Estimation", IEEE Trans. Circuits And Systems For Video Technology, vol 6, no. 3, pp. 313-317, June 1996.

[8] Shan Zhu, and Kai-Kuang Ma, " A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation", IEEE Trans. Image Processing, vol 9, no. 2, pp. 287-290, February 2000.Forman, G. 2003.

[9] Spector, A. Z. 1989. Achieving application requirements. In Distributed Systems, S. Mullender

[10] Sanipini Venkata Kiran, Kantipudi MVV Prasad, N.V.Lalitha, Dr.D.S.Rao, Architecture for the ARPS Algorithm, International Journal of Advances in Computer Networks and its Security pp 443-448

8. AUTHORS PROFILE

Kaushik. M received his Bachelor of Engineering from the Department of Instrumentation Technology, and pursued his Master of Technology in VLSI Design and Embedded Systems. Visveswaraya Technological University, Belgaum, Karnataka, India. His main research areas include Embedded Memories, VLSI System Design, testing, Video Coding techniques to name a few.