RESEARCH ARTICLE OPEN ACCESS

An Efficient Block Matching Algorithm Using Logical Image

Manisha Pal, Asst. Prof Vimal Gupta
Student, JSS Academy of Technical Education, India
Faculty Member, Jss Academy of Technical Education

Abstract-
Motion estimation, which has been widely used in various image sequence coding schemes, plays a key role in the transmission and storage of video signals at reduced bit rates. There are two classes of motion estimation methods, Block matching algorithms (BMA) and Pel-recursive algorithms (PRA). Due to its implementation simplicity, block matching algorithms have been widely adopted by various video coding standards such as CCITT H.261, ITU-T H.263, and MPEG. In BMA, the current image frame is partitioned into fixed-size rectangular blocks. The motion vector for each block is estimated by finding the best matching block of pixels within the search window in the previous frame according to matching criteria. The goal of this work is to find a fast method for motion estimation and motion segmentation using proposed model. Recent day Communication between ends is facilitated by the development in the area of wired and wireless networks. And it is a challenge to transmit large data file over limited bandwidth channel. Block matching algorithms are very useful in achieving the efficient and acceptable compression. Block matching algorithm defines the total computation cost and effective bit budget. To efficiently obtain motion estimation different approaches can be followed but above constraints should be kept in mind. This paper presents a novel method using three step and diamond algorithms with modified search pattern based on logical image for the block based motion estimation. It has been found that, the improved PSNR value obtained from proposed algorithm shows a better computation time (faster) as compared to original Three step Search (3SS/TSS) method .The experimental results based on the number of video sequences were presented to demonstrate the advantages of proposed motion estimation technique.
Keywords- Block matching, Video Compression, Motion estimation

I. INTRODUCTION

The need of higher video compression algorithm has escalated due to fastest development in internet and multimedia technology. Serving the requirement in limited bandwidth networks like handheld PCs and mobile phones, transmission bit rate requirements and the variable bit rate from motion compensated video needs to be considered in addition to very low latencies in end to end communication. In order to effectively employ in low bit rate networks, there are many models for next generation communication devices which shows some two fold nature of having some advantages with some disadvantages. They suffer from today's technical challenges like bandwidth limitation and computation time. So requirement of efficient algorithm to take control over the two fold nature like low bit rate network and devices with low power sources, an efficient and effective algorithm is required. So performing motion estimation in real time with acceptable computational time is a major challenge in video compression.

It is worth mentioning that computational time and bit budgeting is principally due to Block matching algorithms used. In block matching algorithm matching different blocks generally takes more time , so in this paper we performed a different approach for matching the blocks to reduce the time. This paper is divided into four parts; Part I deals with the introduction and basic concept of Block Matching: Part II deals with problem definition and proposed solution: Part III deals with simulation result and Part IV concludes the paper.

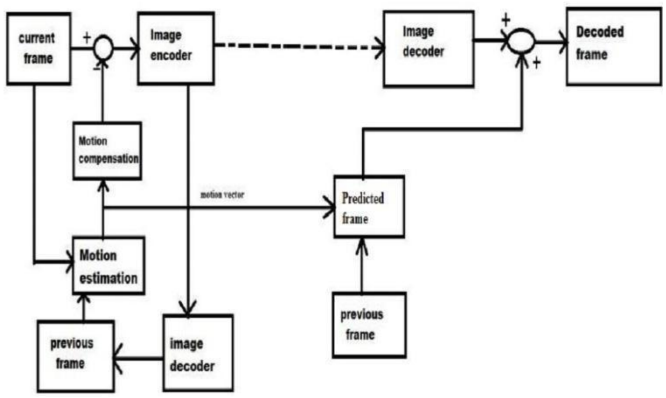

Figure 1: Video Compression Process Flow

A. Block Matching Algorithms

A block matching algorithm removes temporal redundancy i.e. inter frame compression is required to take control over volume of data to be transmitted. Importance of motion estimation lies in motion compression prediction. Frames of moving objects should be tracked first to predict the compression area. The idea behind motion estimation is that the patterns corresponding to objects and background in a frame of video sequence move within the frame to form corresponding objects on the subsequent frame. In block matching, current frame is divided into matrix of macro blocks that are then compared with corresponding blocks and its adjacent neighbors in the previous frame with different algorithms. The movement calculated for all the macro blocks comprising a frame, constitutes the motion estimated in current frame[2]. The matching of one macro block with another is based on the output of a cost function. The macro block that results in the least cost is the one that matches the closest to current block. There are various cost functions, of which the most popular and less computationally expensive is **Mean Absolute Difference (MAD)** given by equation (i). Another cost function is **Mean Squared Error (MSE)** given by equation (ii).

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

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

where N is the side of the macro bock, Cij and Rij are the pixels being compared in current macro block and reference macro block, respectively.

PSNR characterizes the motion compensated image that is created by using motion vectors and macro clocks from the reference frame. It is given by equation(iii) for motion compensated image created by motion vectors.

$$PSNR = 10\log_{10}\left[\frac{(PeakValueOfOriginalData)^2}{MSE}\right] \qquad \text{(iii)}$$

Where
MSE=Mean Squared Error

Temporal redundancy means pixels of two video frames that have the same values in the same location. Primary technique for video compression lies in exploiting temporal redundancy. An assortment of inter-frame compression methods, of various degrees of complexity exist in the literature such as sub-sampling coding, difference coding, and block based difference coding and motion compensation. Motion compensation can be carried out by anticipating the motion of moving objects[6]. Motion compensated prediction assumes that the current picture can be locally modeled as a translation of the pictures of some previous time by comparing the macro blocks.

Video coding standards such as MPEG-1 and MPEG-2 uses block matching methods due to their less computational complexity for motion estimation. Here in block matching method, each frame is divided into sub-blocks of N × N pixels. Each sub block is predicted from the previous or future frame, by estimating the amount of motion of each sub-block called as motion vector during the frame time interval. The video coding syntax specifies how to represent the motion information for each sub-block, but Block Distortion Measure (BDM) function is used for computing such vectors. To locate the best matched sub-block which produces the minimum mismatch error, we need to calculate distortion function at several locations in the search range[2]. One of the first algorithms developed for block based motion estimation was the full search algorithm (FSA) also known as exhaustive search algorithm (ESA), which evaluates the Block Distortion Measure (BDM) function at every possible pixel locations in the search area [4]. Although this algorithm is the best in terms of quality of the predicted frame and simplicity, its computation time is high. In the past two decades, several fast search methods for motion estimation have been introduced to reduce the computational complexity of block matching, some of the algorithms are three-step search (3SS/TSS) [5], four step search (4SS) [5]. As these algorithms utilized uniformly allocated search points in their first step, they all can achieve substantial computational reduction with a drawback of modest estimation accuracy degradation. Search with a large pattern in the first step is inefficient for the estimation of small motion since it will be trapped into a local minimum. In real world video sequences, the distortion of motion vectors is highly center biased which results in a center biased motion vector distribution instead of a uniform distribution [2]. This indicates that the probability increases to get the global minimum at the center region of the search window. To make use of this characteristic, center biased block matching algorithms were then proposed with search points much closer to the center which improves the average prediction accuracy especially for the slow motion sequences[2]. Well known examples of this category are New Three-Step Search (N3SS)[5], Diamond Search (DS) [5] and Cross Diamond Search (CDS) .

II. PROBLEM DEFINITION

Video transmission has gone a long step ahead than audio. In order to be in track with the existing transmission technology considerable research effort has been taken in video compression. The most popular algorithm known in video compression is Three Step Search(3SS/TSS).However 3SS and recently proposed OSA uses uniformly allocated searching points in their first step which becomes insufficient for the estimation of small motions since it gets trapped to local minimum.

Next to 3SS an efficient algorithm termed as DS, very similar to 4SS utilizes a search pattern which gives PSNR close to that of ES with significantly less computational expense.

The proposed algorithm uses a search pattern using 3SS to find local and global minima at the same time at considerable PSNR and reduced computational complexity.

A. Proposed Technique

In this proposed algorithm, we use concept of 3SS and diamond with a different search pattern. Without compromising bit budget and quality, the proposed algorithm tries to find optimally the local as well as global minima. In this method we have used the concept of logical image i.e. image in the form of zeros and ones. We have current frame and previous frame of the image.

1. Read the previous frame of image i.e. imgp.
2. Read the current frame of image i.e. imgc.
3. Find the difference image between current frame and previous
   frame i.e imgd.
4. Find the logical image of the current frame, i.e. imgl.
5. Divide imgc, imgd and imgl into micro blocks.
6. For a given micro block
   Sumd= sum of all bits of micro block in difference image(imgd).
   Suml= sum of all bits of micro block in logical image(imgl).
7. α and β are thresholds set according to application
   here, α and β represents difference between current frame image
   and previous frame image

Algorithm:

{

First Step: If Sumd=0;
Then cost is minimum at the center
X=0,Y=0;
And no. of computations=1

Second Step: Two cases arises
Case-1
If Sumd<=α and Suml=0 or256;
X=x1,Y=y1; Step size=1
Case-2
If sumd<=α and 0<Suml<256;
X=x1,Y=y1; Step size=1

Third Step:Two cases arises
Case-1
If α<Sumd<β and Suml=0 or 256;
X=x1,Y=y1;Step size=2 and then step size=1
Case-2
Else X=x1,Y=y1;
Step size=4 then step size=2 and
Finally step size=1

}

In first step as the difference between two frames is zero so there is no movement in the current frame wrt previous frame. So the cost of searching for motion is zero, as we find it at the center itself.

In second step we again focus on global optimum solution by searching in two different ways (case-1, case-2).In first case sum of difference between frames(Sumd) is less than certain threshold α and sum of logical frame(Suml) is 0 or 256,means it is searching for the block in the frame. So it search for appropriate match by searching with step size=1 and finding X and Y coordinates. In second case sum of difference between frames(Sumd) is less than α but sum of logical frame(Suml) is not equal to 0 or 256. So it will search with step size=1.

In third step we search for local optimum solution in two steps.In first case sum of difference between frames(Sumd) is greater than certain threshold α and less than β and sum of logical frame(Suml) is 0 or 256, then it starts searching with step size=2 and then with step size=1.In second case is sum of difference frames(Sumd) is greater than β or sum of logical frame(Suml) is not equal to 0 or 256 then it will start searching with step size=4, minimum obtained by this, is then searched with step size=2 using diamond approach ,then and to find perfect match it reduces step size=1 .After these three steps finally best matching block is obtained.

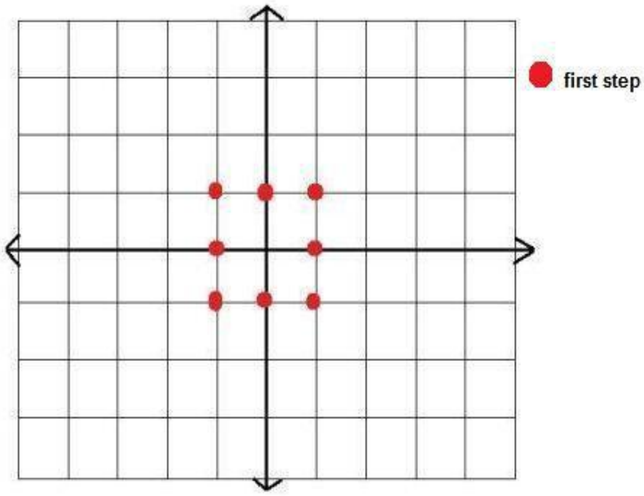

Figure 2:second step(case-1)

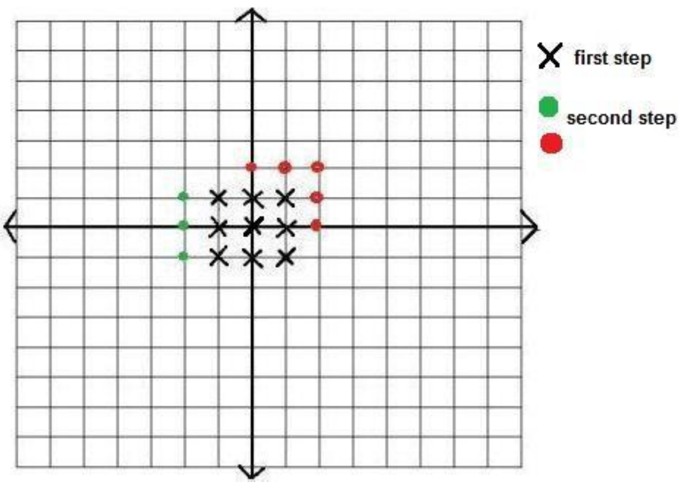

Figure3:second step(case-2)

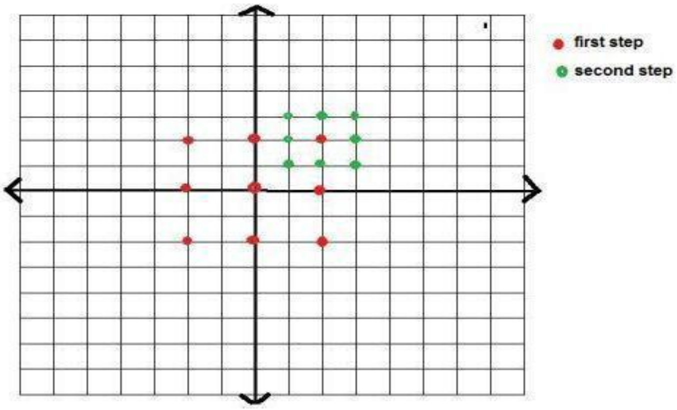

Figure 4:Third step(case-1)

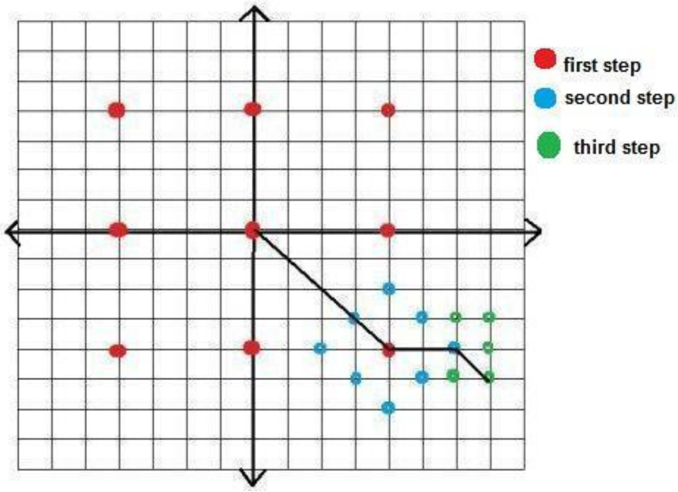

Figure 4:Third step(case-2)

III. SIMULATION RESULTS

Table I: Results for Miss America Frame

S.N	Block Matching Algorithms	PSNR	COMPUTATION(sec)
1	**NTSS**	30.02	22.72
2	**DS**	25.35	29.74
3	**ES**	25.60	204.28
4	**4SS**	25.39	25.09
5	3SS	25.53	23.92
6	**Proposed Algorithm**	**36.45**	**12.52**

Table II: Results for Mother Daughter Frame

S.N	Block Matching Algorithms	PSNR	COMPUTATION(sec)
1	**NTSS**	21.22	21.79
2	**DS**	17.71	29.66
3	**ES**	17.84	204.28
4	**4SS**	17.71	24.931
5	**3SS**	17.77	23.80
6	**Proposed Algorithm**	**38.80**	**11.20**

IV. CONCLUSION

The efficiency of the proposed block matching algorithm has been tested in terms of computation time and PSNR. The result shows a remarkable improvement in terms of PSNR quality and significantly reduced computation time. This algorithm is adaptable as value of thresholds α and β are dependent on the application for which it is used. The comparison table for well known frames clearly reflects improved characteristics of the proposed algorithm compared to other algorithms developed earlier. The efficiency of this algorithm can be best realized when number of blocks will be reduced.

REFERENCES

[1] Santosh Ku Chhotary, & Dheeraj Kannoujia, "*An Efficient Block Matching Algorithm for Fast Motion Estimation using Combined Three Step Search and Diamond Search*", International Journal of Computer and Communication Technology, Vol. 3, Feb, 2012, pp. 113-116

[2] Manojit Panda et al, "*An efficient Three Step Algorithm for Fast Motion Estimation in Mobile Domain*", International Journal of Computer and Communication Technology, Vol. 3, 2014, pp. 109-112

[3] Paramjeet Kaur & Sughandha Sharma, "*Latest Video Compression Standard H.264 within Video Surveillance*", International Journal of Advanced research in Computer science & software Engineering, Vol.2, issue 1,Jan 2012

[4] N.Sobi, " *Survey on Motion Estimation Based on Block Matching Algorithm*" International Journal of Research in Computer Applications and Robotics ,Vol. 1,issue 9, December 2013, pp 131-137

[5] Aroh Barjatya "*Block Matching Algorithms for Estimation*", Member IEEE, 2004

[6] Rajeshwar Dass et al ,"*Video Compression Technique*", International Jouranl of Scientific and Technology Research, vol. 1,issue 10, November 2012, pp 114-119

[7] M.Ezhilarasan and P. Thambidurai, "*Simplified Block Matching Algorithm for Fast Motion Estimation in Video Compression*", Journal of Computer Science, vol. 4, 2008, pp 282-289