International Research Journal of Computer Science (IRJCS)
Issue 05, Volume 3 (May 2016)
ISSN: 2393-9842
www.irjcs.com

Block Based Algorithms for Estimating Motion

Tejas B.N.
Dept. of ECE,
Reva ITM,

Bharathi S H
School of ECE,
Reva University,

Vidya Sagar K.N.
School of ECE
Reva University

Abstract- In this paper we are discussing about ME in video compression that to specifically about the algorithms used for it. Here we can see the different types of BMA which are implemented and compared, which includes from some basic methods to present adaptive methods. The methods which are discussed are used in implementing video coding methods.

Keywords: Motion Vector, Block Matching Algorithm, Motion Estimation (ME), Motion Compensation (MC), Macro-block.

I. INTRODUCTION

For the purpose of motion estimation there is a technique to locate and match macro blocks in a sequence of digital video frames is known as Block matching Algorithm. The main step involved in BMA is to divide and compare the present macro block with that in video. Motion estimation is the process of finding motion vectors. MV explains the conversion of 2D images to another domain. In order to estimate the modification of images with the help of a camera or the object present in the image the motion vectors are applied and it is called as motion compensation. The key part of video compression is the fusion of ME and MC which are used in many video coding methods.

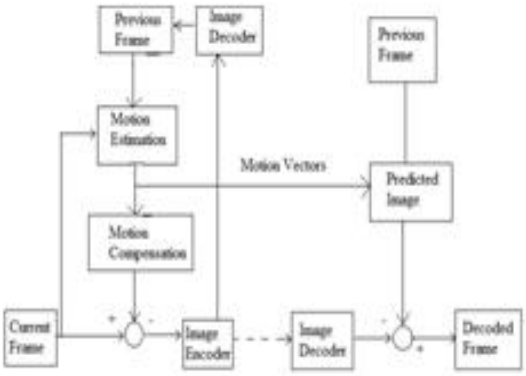

Fig 1: MPEG/H.26x video process flow

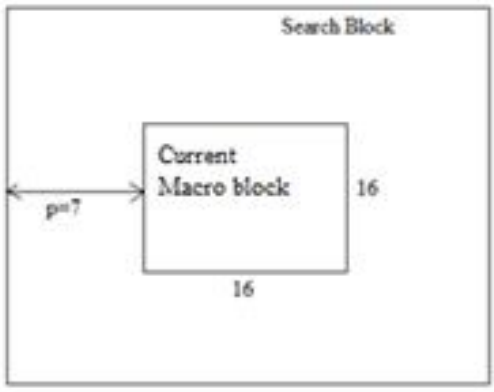

Fig 2: Search Window

II. INTERPRETATION PATTERN:

A pattern is mainly dependent on cost function for matching a macro-block with another.
The most important and popular pattern in terms of computational expenses are as follows

- Mean difference or Mean Absolute Difference

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

- Mean Squared Error

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

Where N is size of the macro-block and C_{ij} and R_{ij} are the pixels

Using reference frame the motion estimation image which is created using the motion vector and macroblocks is characterized by peak signal to noise ratio (PSNR)

$$\text{PSNR} = 10\ log_{10}[\frac{(peak\ to\ peak\ value\ of\ data)^2}{MSE}]$$

III. ALGORITHMS:

A. EXHAUSTIVE SEARCH (ES):

At every position in the search window the error is calculated. Due to this feature we can see the supreme replica of the macro-block in the reference frame with a block in another frame. Here we can see either the image as in the motion compensated form or the highest PSNR, when compared with other methods Yet among all the block matching algorithms exhaustive search takes more time for computation. There is a need for greater number of computations for a large search window.

B. THREE STEP SEARCH (TSS):

One of the earliest fast block matching algorithm is three step search, where location will be at the center in the beginning with search parameter p=7 and step size S=4. Later around the location (0, 0) search 8 locations +/- S pixels. Pick one location with minimum cost function among the searched 9 locations. After that for the picked location set the new search origin with $\frac{S}{2}$ as a new step size. This process is done until we reach S=1. Here the final position for S=1 is the best match for the macro block as well as with minimum cost function.

By a factor of 9 we can see in this algorithm there is a reduction in computation. Three step search evaluates cost for 25 macro blocks whereas Exhaustive search evaluates 225 macro blocks, for p=7.

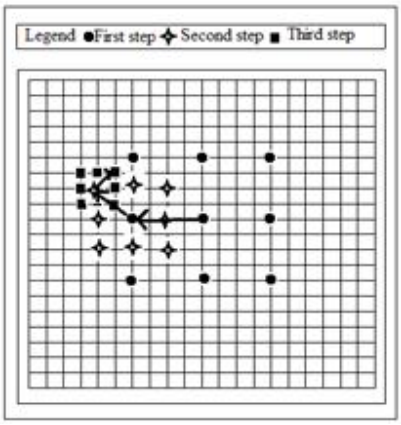

Fig 3: TSS Pattern

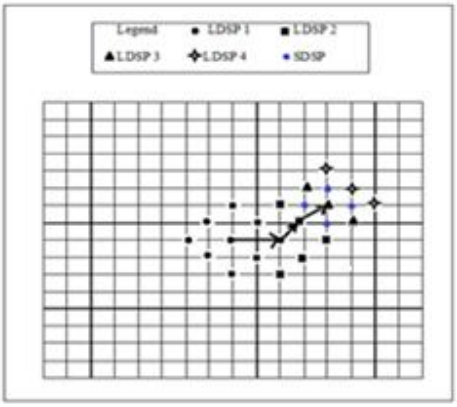

Fig 4: Diamond search

C. DIAMOND SEARCH (DS):

In this method diamond search pattern where there is no restriction for the steps to proceed the method.
We can use any of the two types of fixed pattern for search.

1) Large Diamond Search Pattern (LDSP)
2) Small Diamond Search Pattern (SDSP)

1) LARGE DIAMOND SEARCH PATTERN:

This pattern starts its search by the location at center with step size S=2. Then search for 8 locations all over the point (0,0). Now pick the one with minimum cost function among the 9 locations searched. Observe the location of the minimum weight.

Case 1: *If the minimum cost function is found at center of search window then follow the Small diamond search pattern.*
Case 2: *If cost function is not at the center then fix the new search origin and repeat the above step.*

2) SMALL DIAMOND SEARCH PATTERN:

Begin the procedure by setting the new origin, with step size as $\frac{S}{2} = 1$. Repeat it until we get position with the least error. And then set that location with least weight as motion vector.

The major advantages of Diamond search are as follows:

- *Since the search pattern not even big or small this method helps to find the minimum very accurately.*
- *The computational time is less and gives nearly equal PSNR as exhaustive search.*

D. ADAPTIVE ROOD PATTERN SEARCH (ARPS)

The general motion in a frame is usually coherent is the major fact which is used by ARPS algorithm. This means we can see a possibility with the present macro block which have same motion vector when the macro blocks around the current macro block moved in a particular direction. In order to predict its own MV of the macro block to its neighbors.

This algorithm starts at the center by taking it as a search location. Then for that block we need to find the predicted motion vector. Considering (X, Y) as the co-ordinate of predicted motion vector set step size S = max ($|X|+|Y|$). Then we need to search the distributed points around the origin with step size S. Now set the origin with the point having least weight. Continue the steps of SDSP until least weight is achieved.

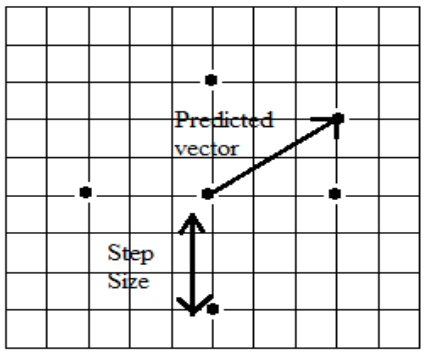

Fig 5: Adaptive Rood Pattern

Some of the main advantages of Adaptive rood pattern search are as follows:

- *Here calculation time is not wasted; instead it directly starts using the SDSP.*
- *This algorithm saves on computations which small diamond search pattern. But Diamond search takes time in doing large diamond search pattern.*

IV. RESULTS:

The above algorithm have been implemented in MATLAB and even applied to different image sequences. The plot and table represents the comparison results.

TABLE I: COMPARISON RESULTS

METHOD	PSNR
THREE STEP SEARCH	30.2692
EXHAUSTIVE SEARCH	30.3572
DIAMOND SEARCH	30.3430
ADAPTIVE ROOT PATTERN SEARCH	30.3065

V. SUMMARY

In this paper we have discussed seven different block matching algorithm techniques where we can get the best PSNR along with the minimum error from full search. But the major disadvantage of this algorithm is computationally expensive. But we can see PSNR closer to full search with reduced computational complexity in adaptive rood pattern search. Hence among the seven algorithms we can say that adaptive rood pattern search is the best algorithm to adopt which gives better PSNR as well as save computation time.

ACKNOWLEDGEMENT

This work is supported in part Department of ECE, REVA INSTITUTE OF TECHNOLOGY and MANAGEMENT.

REFERENCES

[1]. Borko Furht, Joshua Greenberg, Raymond Westwater, *Motion Estimation Algorithms For Video Compression*. Massachusetts: Kluwer Academic Publishers, 1997. Ch. 2 & 3.
[2]. M. Ghanbari, *Video Coding, An Introduction to Standard Codecs*, London: The Institute of Electrical Engineers, 1999. Ch.2, 5, 6, 7 & 8
[3]. Iain E. G. Richardson, Video Codec Design, West Sussex: John Wiley & Sons Ltd., 2002, Ch. 4, 5, & 6.
[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]. 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, April 1997
[6]. 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.

