Motion Estimation Algorithm For Video Compression

A.Anitha
CVR College of Engineering, Department of ECE, Ibrahimpatan, R.R.District, A.P., India
Email:aniravi08@gmail.com

Abstract—**In this paper, two motion estimation algorithms have been implemented and performance features of these two tested. These algorithms are full search and fast search methods. Parameters such as number computation, speed and PSNR have been compared to evaluate performance Implementation has been done using Matlab.**

***Index Terms*—Block matching, motion estimation, motion vector, video compression.**

I. INTRODUCTION

Motion estimation plays an important role in any video compression system, since it can achieve significant compression by exploiting the temporal redundancy existing in a video sequence. In motion estimation the current image is divided into Macro Blocks (MB) [1].

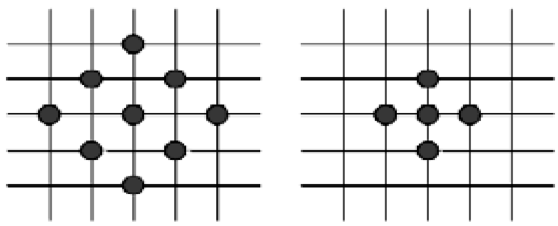
Most of the algorithms have been proposed for motion estimation use from BMA_based (Block Matching Algorithms) bases methods. In these methods, motion estimation is performed for a N×M blocks of current frame, It is done by checking entire N×M blocks from search area situated in the reference frame(s) and calculating the difference between the current block and other reference blocks and finally choosing the block that has the most similarity to the earlier block in the current frame. The difference between the two blocks as residual (motion compensated residual) and the distance between them as motion vector, are coded and transmitted. This is represented as error J_m

II. FULL SEARCH METHOD

In this method, all possible modes are checked. By performing the motion estimation for every block and calculating the R-D criterion for all of them, block sizes that are used are determined. Firstly motion estimation for macroblock (16×16 block) is performed and j_m is calculated. The macroblock is then divided into 16×8 and 8×16 blocks and for each of them, motion estimation and j_m is calculated. The sum of calculated j_m of blocks in each mode, is the j_m of that mode. This splitting process is continued till a 4×4 block is obtained. From this four state, a mode that has minimum value of j_m is choosen. If the selected mode is 8×8, breaking process of each block is continued like the previous. The smallest possible block size is 4×4 and afterward the breaking procedure is terminated [3].

III. METHOD OF FAST SEARCH

Macro Block determination using (fast search) [2] algorithm, fast motion is estimated with modified diamond search for variable block sizes. Motion vector field adaptive search technique (MVFAST) uses a different initial search point and search patterns with selective application of large diamond search (LDS) and small diamond search (SSD) according to the characteristics of motion activity assessed by the similarity of motion vector field among contiguous blocks. LDS pattern is illustrated in Figure 1.



(a) LDSP (b) SDSP

Figure. 1 Diamond search patterns

The search center is modified dynamically by estimating the error and choosing the block error with minimum as the subsequent search center. If the minimum error occurs at the search center of LDSP, then the search pattern is switched from LDSP to SDSP, and the position having the minimum error in SDSP is decided as the motion vector. Otherwise, a new center of LDSP is placed at the point that yields the minimum distortion in the previous step, and all points on the new LDSP are tested again. This process is iteratively repeated until the minimum error falls on the search center [4].

IV. Implementation results:

Figure 2(a)

Figure 2(b)

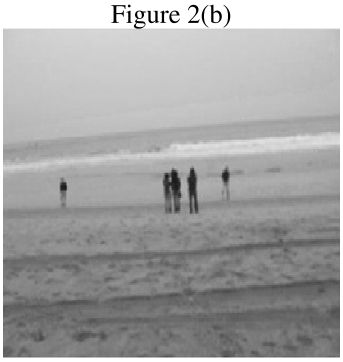


Figure 2(c)

Figure 2(d)

Figures 2(a) to Fig 2(d) are the four successive frames of the input video:

Figure 3(a)

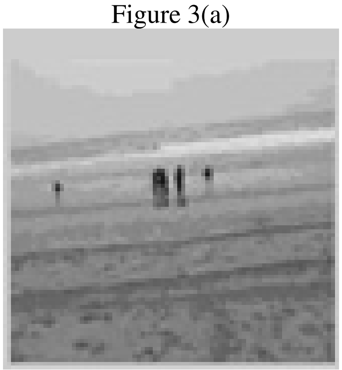


Figure 3(b)

Figure 3(c)

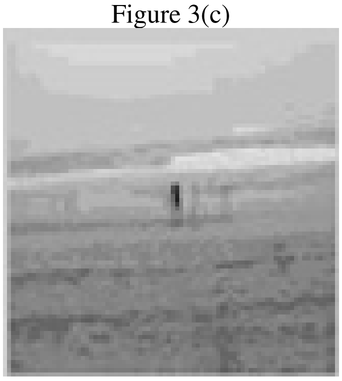


Figure3 (d)

Figures 3(a) to Fig 3(d) are the four successive frames of the recovered output video:

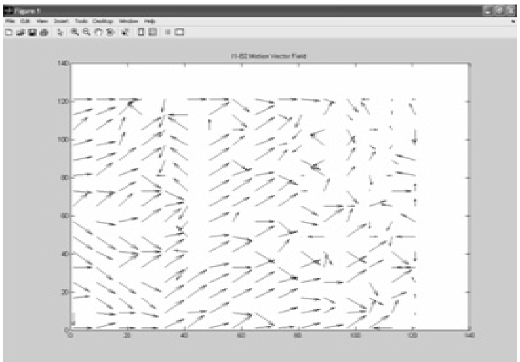


Figure 4. Motion Vector generated by Full Search Algorithm:

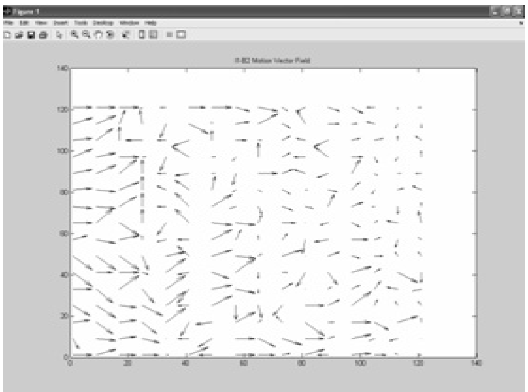


Figure 5. The Motion vector generated by Diamond search algorithm:

Table1:Performance comparison of two algorithms when MB=8:

ME scheme	TCP*	Time sec	PSNR* dB
Full search algorithm	408608	40.922	82.5406
MVFAST	43259	11.359	82.5882

*TCP: total check points, PSNR: peak signal to noise ratio

CONCLUSIONS

MVFAST search algorithm improves time by 29.563 sec and peak signal to noise ratio by 0.0476 dB as compared to with full search algorithm**.** Hence it is concluded that this method is better.

REFERENCES

[1] M. Ghanbari, Standard Codecs : Image Compression to Advanced Video Coding. IEE, 2003.
[2] Paolo De Pascalis, Luca Pezzoni, Gian Antonio Mian, and Daniele Bagni, "fast motion estimation with size-based predictors selection hexagonal search in H.264/AVC encoding", IEEE 2003.
[3] K. Tavassoli and W. Badawy, “A prototype for parallel motion estimation architecture using full-search block matching algorithm,” in International Workshop on Digital and Computational Video (DCV’02), Nov. 2002.
[4] Woong IL Choi; Byeungwoo Jeon; "fast motion estimation with modified diamond search for variable motion block sizes", International Conference on Image Processing,Volume 2, pp 371-4, Sept. 2003.