Akhil Mahajan al. International Journal of Recent Research Aspects ISSN: 2349-7688, Vol. 1, Issue 2, Sept. 2014, pp. 78-83

Comparative Analysis of Pixel Based Motion Estimation with Block Based Motion Estimation

Akhil Mahajan[1], Dr. Kamal Sharma[2], Deepak Kumar[3]

[1]Student, M. Tech, ESEAR, Ambala
[2]Professor, Dept. of ECE, E-Max group of Institutions, Ambala
[3]Assistant Professor, Dept. of ECE, E-Max group of Institutions, Ambala

Abstract— **In Digital video communication it is not practical, to store the full digital video without processing, because of the problems encountered in storage and transmission, so the processing technique called video compression is essential. In video compression, one of the computationally expensive and resource hungry key element is the Motion Estimation. The Motion estimation is a process which determines the motion between two or more frames of video. This paper addresses a comparison between block based motion estimation and pixel based motion estimation (ME) algorithms. We present the ME algorithms for both the methods, results of simulations and illustrate the analysis with PSNR values and computation time for different images. In image and video processing, the estimation of motion plays a vital role in video compression as well as multi-frame image enhancement. These applications share one common thread in all such applications, the demand is high for accurate estimates of motion requiring minimal computational cost.In the following paper we propose two methods for evaluation of the motion estimation algorithm: (1) motion estimation by pixel based approach and (2) Block based three step search technique.**

Keywords— **Motion detection, motion estimation, pixel processing, Pel approach.**

I. INTRODUCTION

In multimedia communication, the important requirement is to achieve high processing speed and a low computing time simultaneously without scarifying in image quality . The video compression has become an important part of the way we create, communicate, and consume visual information" thus Video compression is vital for efficient storage and transmission of digital signal in multimedia. The video compression involves Video coding exploits the high correlation between successive frames to improve coding efficiency, which is usually achieved by using motion estimation (ME) and motion compensation techniques. Hence in video compression one of the computationally expensive and resource hungry key element is the Motion Estimation. Motion estimation is defined as searching the best motion vector, which is the displacement of the coordinate of the best similar block in previous frame for the block in current frame. Motion estimation and compensation are used to reduce temporal redundancy between successive frames in the time domain. A typical system needs to send dozens of individual frames per second to create an illusion of a moving picture. For this reason, several standards for compression of the video have been developed. Each individual frame is coded so that redundancy is removed [1]. Motion estimation (ME) techniques have been successfully applied in motion compensatedpredictive coding for reducing temporal redundancies. They belong to the class of nonlinearpredictive coding techniques. An efficient representation of motion is critical in order to reachhigh performance in video coding. ME techniques should on one hand provide good prediction,but on the other hand should have low computational load. The purpose of ME is indeed to globally minimize the sum of these two terms. In block based ME image is partitioned into blocks and the same displacement vector is assigned to all pixels within a block whereas in pixel based motion estimation a pattern search algorithm is used , in which the size is dynamically determined for each macro-block, based on the mean of two motion vectors of the neighbouring macro-blocks . This technique attempts to decrease the computation time between the current macro-block and the macro-blocks in the reference frame, by stopping the calculation of the sum absolute different between the pixels in the current and the reference frame macro-blocks when the current uncompleted sum absolute value is greater than the previous calculated one [2].

The purpose of this work consists in a comparative analysis between block based motion estimation algorithm and pixel based motion estimation algorithm, used in video compression, exploiting the temporal correlation between successive sequence frames we can reduce enormously the memory space needed for video storage and processing.In this context, main objective of motion estimation is no longer to find 'true' motion in the scene, but rather to maximize compression efficiency. In additional words, motion vectors should deliver a precise prediction of signal. Moreover, the motion info should enable a compact representation, as it has to be conveyed as overhead in compressed code stream. The major applications of motion estimation algorithms include traffic movement tracking, studying plant root growth, hand posture analysis, human posture analysis, lip movement for user authentication, cinematography, robotic heart surgery, breathing motion estimation and many more [3].

The paper describes our MATLAB software-based system for comparison of algorithms for video compression.The paper is ordered as follows. In section II, we discuss the block based motion estimation and its block diagram. In Section III,we describe the pixel based motion estimation.In Section IV,we explained the algorithm for both the techniques. After this we simulated the result of both the techniques in Section V.Finally, conclusion is explained in Section VI.

II. BLOCK BASED MOTION ESTIMATION

Block-based motion estimation is extensively used motion estimation method for video coding. Interest in this method was initiated by Jain and Jain and he suggested a block-matching algorithm (BMA) in 1981. The current frame is first divided into blocks of M × N pels. The process then undertakes that all pels within the block undergo same translational movement. Thus, same motion vector, d is allocated to all pels within the block. This motion vector is projected by searching for best match block in a larger search window pels centred at same location in a reference frame.This procedure is based on a translational model of motion of objects between frames. It also assumes that all pels within a block undergo similar translational movement[4].

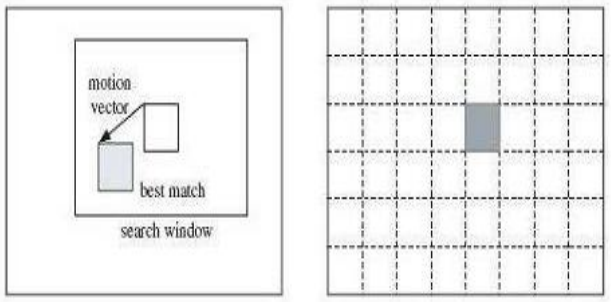


Figure 1: Block-based Motion Estimation [15]

Motion estimation creates a model of the current frame based on one or more previously encoded frames which are known as reference frames. These reference frames may be past frames or future frames. The goal for creating a motion estimation algorithm are to model the current frame as accurately as possible that gives better compression performance with acceptable computational complexity [5].

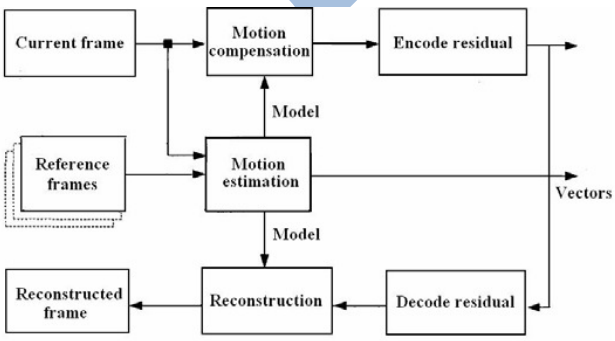


Figure 2:Block diagram of Block-based MotionEstimation

The block diagram of block based motion estimation and compensation is shown in Figure 2. The motion estimation creates a model by modifying reference frames to match the current frame as closely as possible (by using a matching criterion). The current frame is motion compensated by subtracting the model from the frame to produce a motion-compensated residual frame. This is coded and transmitted, along with the information required like a set of motion vector for the decoder to recreate the model. At the same time, the encoded residual is decoded and send to the model to reconstruct a decoded copy of the current frame (which may not be same to the original frame because of some coding losses). This reconstructed frame is stored to be used as a reference frame for further frame predictions.

Some Authors proposed that three-step search (TSS) algorithm has been extensively used as motion estimation technique in low bit-rate video compression applications, owing to its easiness and effectiveness. However, TSS uses a steadily allocated checking point pattern in its first step, which becomes ineffective for estimation of small motions.

Some proposed a new four-step search (4SS) procedure with centre-biased checking point design for fast block motion estimation. Halfway-stop technique is active in new algorithm with searching steps of 2 to 4 and total number of checking points is specked from 17 to 27. Simulation results show that suggested 4SS performs better than well-known three-step search and has similar performance to new three-step search (N3SS) in terms of motion compensation faults. In addition, 4SS also decreasesworst-case computational requirement from 33 to 27 search points and regular computational necessity from 21 to 19 search points as related with N3SS[6].

Authors proposed a block-based grade descent search (BBGDS) algorithm to perform block motion estimation in video coding. The BBGDS assessesvalues of a given objective function starting from a small centralized checking block. The least within checking block is found, and gradient descent direction where minimum is expected to lie is used to determinesearch direction and position of new checking block. The BBGDS is compared with full search (FS), three-step search, one-at-a-time search (OTS), and new three-step search. Experimental results show that proposed technique delivers competitive performance with reduced computational complexity.In this paper we are using three step search method(TSS) of block based motion estimation for comparison with pixel based technique [9].

III. PIXEL BASED MOTION ESTIMATION

Motion estimation (ME) techniques have been successfully useful in motion compensated predictive coding for falling temporal redundancies. They belong to class of nonlinear predictive coding techniques. An effective representation of motion is serious in order to reach high performance in video coding. Estimation techniques should on one hand provide good prediction, but on other hand should have low computational load.Pixel based motion estimation is a new technique of motion estimation.In this technique pattern search algorithm is used, in which the size is dynamically determined for each macro-block, based on the mean of two motion vectors of

the neighbouring macro-blocks instead of one as is the case with adaptive rood patter search. The technique attempts to decrease the computation time between the current macro-block and the macro-blocks in the reference frame, by stopping the calculation of the sum absolute different between the pixels in the current and the reference frame macro-blocks when the current uncompleted sum absolute value is greater than the previous calculated one.

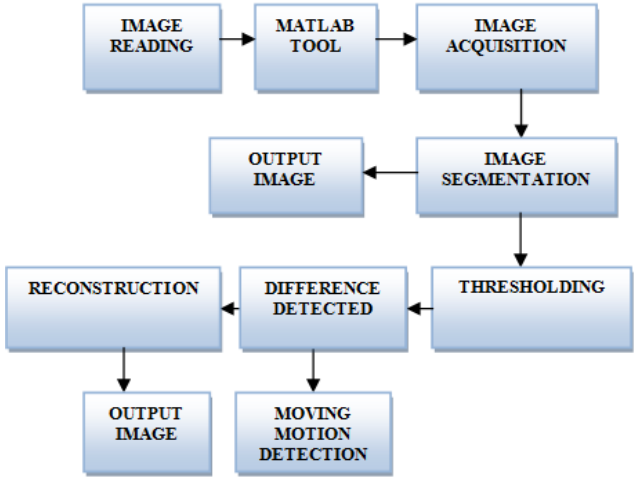


Figure 3:Block diagram of Pixel-based Motion Estimation

The block diagram of Pixel based motion estimation and reconstruction is shown in Figure 3. Firstly the webcam acquire the images from camera depends on lots of software. MATLAB will automatically find the webcam connected to the computer. After getting the image, Segmentation process is started. The objective of segmentation is to classify each RGB pixel in a given image as having a color in the specified range or not. The segmented image is further threshold to get isolated objects the backround. Atlast final motion is being detected and estimated to get the final image and further we get the reconstructed image.

IV. ALGORITHMS OF TSS AND PIXEL BASED MOTION ESTIMATION TECHNIQUE

A. *Three step search Algorithm*

This algorithm was introduced by Koga et al in 1981. It became very popular because of its simplicity and also robust and near optimal performance. It searches for the best motion vectors in a course to fine search pattern. The algorithm may be described as:

Step 1: An initial step size is picked. Eight blocks at a distance of step size from the center (around the center block) are picked for comparison.

Step 2: The step size is halved. The center is moved to the point with the minimum distortion.The point which gives the smallest criterion value among all tested points is selected as the final motion vector m.TSS reduces radically the number of candidate vectors to test, but the amount of computation required for evaluating the matching criterion value for each vector stays the same. TSS may not find the global minimum (or maximum) of the matching criterion; instead it may find only a local minimum and this reduces the quality of the motion compensation system. On the other hand, most criteria can be easily used with TSS.

B. *Pixel based motion estimation Algorithm*

The steps for pixel based motion estimation are:

1. Interface WEBCAM with MATLAB
a) Install image acquisition device.
b) Retrieve hardware information.
c) Create a video input object.
d) Preview video stream (optional)
e) Configure object properties.
f) Acquire image data.
g) Starting the video input object.
h) Triggering the acquisition.
2. Image Reading
3. Image Enhancement
4. Image Conversion
5. Image Segmentation
6. Apply thresholding process
7. Feature Extraction
8. Detect the final motion
9. Reconstruction Original Image

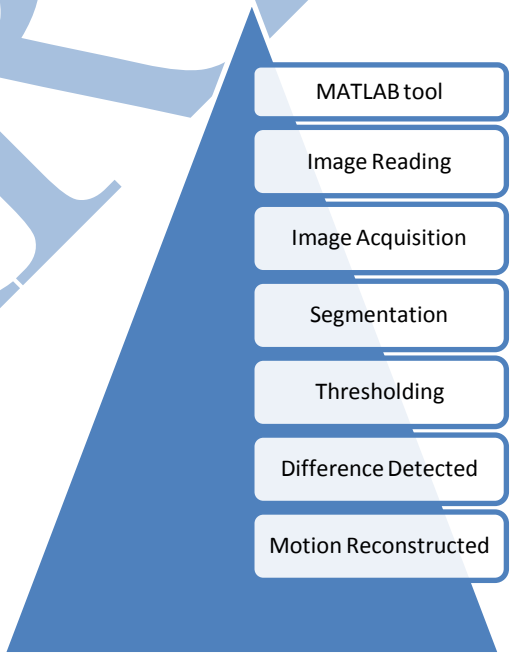


Figure 4: Block Representation of Pixel based motion estimationalgorithm

C. *Description of Pixel based motion estimationalgorithm*

1. *Webcam Interfacing and Video Processing*

It acquire the images from camera depends on lots of software. Creating the interface between computer and the camera can be done using lower level language like C or C++ will give lots of flexibility, but it will also invoke lots of work and background knowledge. MATLAB's image acquisition toolbox has a variety of simple functions.

- Specifications of WEB CAM
  - ImageResolution: 640x480,352x288,320x240,176x144,160x120
  - Frame Rate : Up to 30 fps
  - Image Control : Brightness, contrast ,saturation

- Environment : Indoor, Outdoor
- I/O interface : USB 1.1,2.0
- Image Format : RGB_ 24

2. Image Reading

img= imread(strcat(a,num2str(i),b));

This takes the grey values of all pixels in grey scale image and puts them all into a matrix img. This matrix img is now a MATLAB variable, and so we can perform many matrix operations on it. In general the imread function reads pixel values from an image file, and returns a matrix of all pixel values.

3. Image Segmentation

Segmentation refers to process of partitioning an image into component parts, or into separate objects. Segmentation of an image entails the division or separation of the image into regions of similar attribute. The most basic attribute for segmentation is image luminance amplitude for a monochrome image and color components for a color image. Image edges and texture are also useful attributes for segmentation. There are a collection of ad hoc methods that have received some degree of popularity. Because the methods are ad hoc, it would be useful to have some means of assessing their performance. The objective of segment at ion is to classify each RGB pixel in a given image as having a colorin the specified range or not.

1. Thresholding

A grey scale image is converted into a binary (black and white) image by first choosing a grey level in original image, and then converting every pixel black or white according to whether its grey value is better than or less than T :

A pixel becomes: {white if its grey level is > T

{Black if its grey level is<T

Thresholding is a vital part of image segmentation, where we wish to isolate objects from background. It is also an significant component of robot vision. Thresholding can be done very simply in MATLAB.

2. Displaying the Images

Now to show image, the toolbox includes two image display functions: **imshow** and **imtool**. Imshow is toolbox's fundamental image presentation function. Imtool starts the Image Tool which presents an combined environment for displaying images and performing some common image processing tasks. The Image Tool delivers all image display capabilities of imshow but also offers access to several other tools for navigating and exploring images, such as scroll bars, Pixel Region tool, Image Information tool, and Contrast Adjustment tool. You can use either function to show an image. Here we use **imshow**.

3. Detection of Motion

- Scan the entire area for finding out difference of two images.
- Comparing that point row and column vise.
- If there is difference in pixel value, then it takes the difference of images and hence motion is detected.
- Then image is reconstructed using motion vectors.

4. Motion Compensation

Motion estimation creates a model of the current frame based on available data in one or more previously encoded frames (‘reference frames’). These reference frames may be ‘past’ frames (i.e. earlier than the current frame in temporal order) or ‘future’ frames (i.e. later in temporal order). The design goals for a motion estimation algorithm are to model the current frame as accurately as possible. The encoder forms a model of the current frame based on the samples of a previously transmitted frame. The encoder attempts to ‘compensate’ for motion in a video sequence by translating (moving) or warping the samples of the previously transmitted ‘reference’ frame. The resulting motion-compensated predicted frame (the model of the current frame) is subtracted from the current frame to produce a residual ‘error’ frame.At the same time, the encoded residual is decoded and added to the model to reconstruct a decoded copy of the current Frame (which may not be identical to the original frame because of coding losses). This reconstructed frame is stored to be used as a reference frame for further predictions [13].

V. SIMULATION RESULTS

A.Implementation

MATLAB is one of a number of commercially accessible, sophisticated mathematical computation tools, which also comprise Maple, Mathematica, and MathCAD. Despite what followers may claim, no single one of these tools is “the best.” Each has strengths and weaknesses. Each allows you to perform simple mathematical calculations. They differ in way they handle symbolic calculations and more complex mathematical procedures, such as matrix manipulation. For example, MATLAB (short for **Mat**rix **Lab**oratory) excels at computations relating matrices, whereas Maple excels at symbolic calculations. At a central level, you can think of these programs as sophisticated computer-based calculators. They can perform same functions as your scientific ‘C’ calculator—and **many more**. If you have a computer on your desk, you may find yourself using MATLAB instead of your calculator for even the simplest mathematical applications—for example, balancing your checkbook. In many engineering classes, the use of programs such as MATLAB to perform computations is replacing more traditional computer programming.

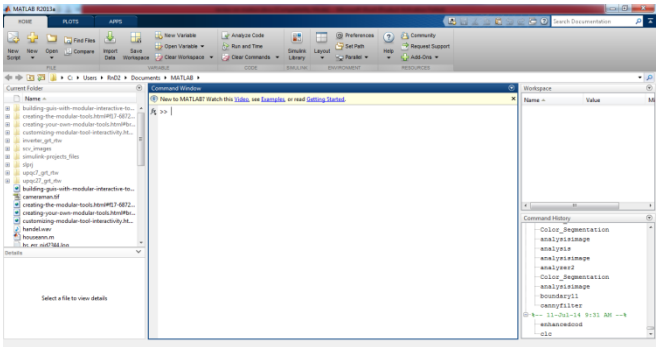
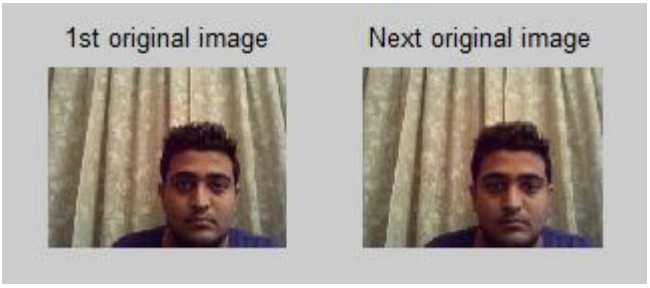


Figure 5: MATLAB Tool

B. *Pixel based motion estimation method Results*

A video image is a projection of a 3-D scene onto a 2-D plane. A still image is a 'snapshot' of the 2D representation at a particular instant in time whereas a video sequence



represents the scene over a period of time. Fig 6 shows the real time current image and its previous image.

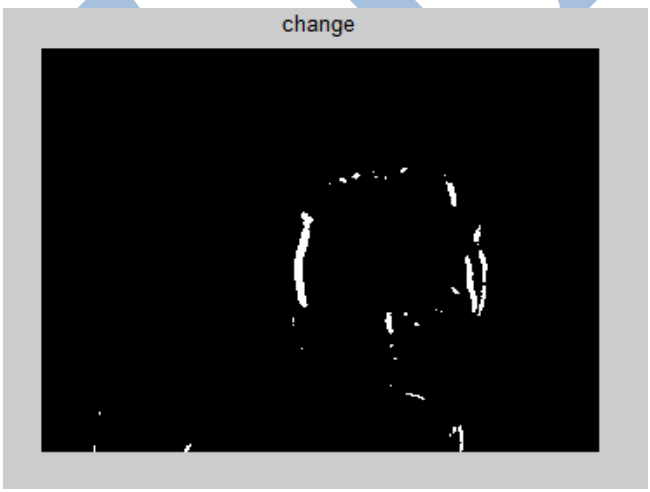
Figure 6: Input Original Images

After reading, it converts the grayscale image to a binary image. The output image substitutes all pixels in the input

image with luminance better than level with value 1 (white) and replaces all other pixels with value 0 (black) as shown in fig 7.

Figure 7: Image Conversion Output

In Figure 8 the motion estimation module creates a model by modifying one or more reference frames to match the



current frame as closely ao possible (according to a matching criterion). And reconstructed image is shown in fig 9.

Figure 8: Motion Detection Output

Figure 9: Reconstruction Image Output

The peak signal to ratio (PSNR) and MSE is an important parameter in image processing. Mean squared error provides a measure of the energy remaining in the difference block. Varying block sizes, or irregular- shaped regions, can be more efficient at matching true motion than fixed 16 x 16 blocks. The PSNR is shown in fig 10.

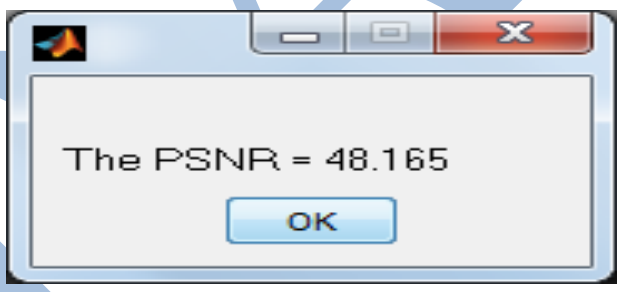


Figure 10: PSNR Output of Pixel Approach

C. Three step search method results

Fig 11 shows the input images taken by the camera at different intervals.

Figure 11: Input Original Images

Both the input images are processed to find the motion between the images. Further we defines a motion vector which is processed to get the final reconstructed image shown in fig 12.

Figure 12: Reconstruction Image Output

Finally the PSNR value is calculated which is shown in fig 13.

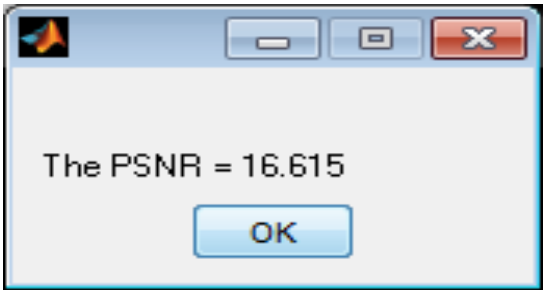


Figure 13: PSNR Output of Three step search method

Comparison of pixel based motion estimation and Three step search method for different values of delay is shown in table 1.

S.No	Name of the Algorithm	Delay	Computation time	Maximum PSNR
1	Pixel based motion estimation	D=0	.981s	48.125
		D=2s	3.006s	34.185
		D=4s	5.116s	25.242
		D=6s	7.028s	24.367
2	Three step search technique	D=0	81.231s	16.538
		D=2s	99.672s	15.567
		D=4s	102.695s	16.175
		D=6s	104.739s	17.035

Table 1: Calculation of computation time and the PSNR value for different values of delay

VI. CONCLUSION

This paper presents the review of pixel based motion estimation algorithm and block based three step search method in order to compare them together. Pixel based motion estimation algorithm gives the best PSNR value. The pixel based motion estimation algorithm can reduce the computational time as compared to block based technique.Motion Estimation (ME) and compensation techniques, which can exclude temporal redundancy between adjacent frames effectively, have been widely useful to popular video compression coding standards such as MPEG-4. The displacement of each picture element in each frame forms the displacement vector field (DVF) and its estimation can be done using at least two successive frames. The pixel based approaches depend upon intensity of image and its performance is affected by presence of noise. While block based techniques depends upon motion vectors and it has high computation time as compared to pixel approaches. The experiment results have shown that the pixel based algorithm can reduce the computational time required to determine matching macro block from the reference frame to the current macro-block as compared to three step search approach. Overall the pixel based motion estimation algorithm is giving the highest PSNR with reduced computation time for both images and video sequences.

REFFERENCES

[1] R. Gonzalez, R. Woods and S. Eddins"Digital Image Processing Using Matlab", 2004, Prentice Hall.

[2] Gonzalez, R., Woods, R., "Digital Image Processing", 2nd Edition, Prentice-Hall, (2002).

[3] Acharya, T. and Ray, A.K, Image processing: Principles and Application, Wiley, and Hoboken, NJ, 2005.

[4] I. E. G. Richardson. H.264 and MPEG-4 video compression. Wiley , Chichester, England, 2003.

[5] Lai-Man Poet. , "Displacement measurement and its application in interframe image coding," IEEE Trans. Communications, vol.COM-29,PP. 1799–1807, Dec.1981.

[6] S.kappagantula and K.R.Rao, "Motion compensated interframe image Prediction", IEEE Trans. Commun 33,PP, 1011–l015,Sept, 1985.

[7] B.Liu and A.Zaccarin, New fast algorithms for the estimation of block motion vectors, IEEE Trans. Circuits Syst. Video technology, Vol.3, pp.440–445,Dec 1995.

[8] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 4, pp. 438442, Aug. 1994.

[9] Ram Srinivasan And K. R. Rao. Predictive Coding Based On Efficient Motion Estimation. Ieee Transactions On Communications, Vol. Com-33, No. 8, August 1985

[10] Renxiang Li, Bing Zeng, And Ming L. Liou. A New Three-Step Search Algorithm For Block Motion Estimation. Ieee Transactions On Circuits And Systems For Video Technology, Vol. 4, No. 4, August 1994.

[11] Lai-Man Po Wing-Chung Ma. A Novel Four-Step Search Algorithm For Fast Block Motion Estimation. Ieee Trans. Circuits Syst. Video Technol., Vol. 6, No.2, Pp. 313-317,Jun. 1996.

[12] Lurng-Kuo Li And Ephraim Feig. A Block-Based Gradient Descent Search Algorithm For Block Motion Estimation In Video Coding. Ieee Transactions On Circuits And Systems For Video Technology, Vol. 6, No. 3, August 1996.

[13] Jo Yew Tham, SurendraRanganath, MaitreyaRanganath, And Ashraf Ali Kassim. A Novel Unrestricted Center-Biased Diamond Search Algorithm For Block Motion Estimation. Ieee Transactions On Circuits And Systems For Video Technology, Vol. 8, No. 3, August 1998

[14] Ce Zhu, Xiao Lin, And Lap-Pui Chau. Hexagon-Based Search Pattern For Fast Block Motion Estimation. Ieee Transactions On Circuits And Systems For Video Technology, Vol. 11, No. 5, May 2002

[15] Chun-Ho Cheung, Member, Ieee, And Lai-Man Po, Member, Ieee. Novel Cross-Diamond-Hexagonal Search Algorithms For Fast Block Motion Estimation. Ieee Transactions On Multimedia, Vol. 6, No. 1, February 2005.

