A Review on Design and Implementation of Compensated Frame Prediction & Reconstruction

Monika, Er. Suman Rani

***Abstract*— this paper presents a review of compensated motion frames prediction and detection techniques. Motion estimation has classification as- 1) Block based motion estimation 2) Pixel based motion estimation. Block based ME is a search pattern in which different shape or size has a very significant influence on exploration speed and alteration performance. Pixel based ME approach seeks to determine motion vectors for every pixel in the image. This paper uses a technique of pattern similarity based on block based and pixel based approach. In this, various image processing techniques will be used for frames prediction and its reconstruction. This will use MATLAB tool for simulation and will evaluate of MSE and SNR.**

***Index Terms*— Motion estimation techniques, frames prediction, block based techniques, reconstruction of images.**

I. INTRODUCTION

Digital Image Processing is a promising area in the field of electronics and communication engineering. Image can be defined as a two dimensional intensity function f (x,y), where x and y denotes spatial co-ordinates and the value of 'f' at any point is directly proportional to brightness(grey level) of image at that point. Image digitization is a process that converts a pictorial form to numeric data. Image processing is used for improving the visual appearance of image toa human viewer and preparing images for measurement of the features and structures present [1].

Image processing methods originated in the late 1960s and early 1970s to be used in medicinal imaging, remote earth possessions observations, and astrophysics. Digital video coding has regularly increased in importance since 90s when MPEG-1 first emerged. It has had large influence on video delivery, storage and presentation. Compared to analog video, video coding achieves higher data compression rates without significant loss of subjective picture quality. This eliminates the need of high bandwidth as required in analog video delivery. Hybrid video coding designs have been engaged since the first generation of video coding principles, i.e. MPEG. MPEG-1 is a video compression standard developed in joint operation by International Standard Organisation (ISO) and International Electro-Technical Commission (IEC). MPEG can be used at higher bit rates than H.261, at about 1.5 megabits per second, which is suitable for storing compressed video stream on compact disks or for using with interactive multimedia systems [3].

Manuscript received February 12, 2015.

Monika, M.Tech Student, ECE Department, PPIMT Hisar, GJU University, Haryana, India.

Er Suman Rani, Assistant Professor, ECE Department, PPIMT Hisar, GJU University, Haryana, India.

Motion estimation and compensation are used to reduce temporal redundancy between successive frames in the time domain. Transform coding, also commonly used in image compression, is working to reduce spatial dependency within a frame in the spatial domain. Entropy coding is used to reduce statistical redundancy over the residue and compressed data. This is a lossless compression technique commonly used in file compression [2].

Movement of objects causes changes between frames of images or videos. Using an approach of the motion of things between frames, the encoder evaluates the motion that happened between the reference frame and current frame. This process is called motion estimate (ME). The encoder then uses this motion approach and information to move the contents of the reference frame to provide a better estimate of the current frame. This process is known as motion compensation (MC), and the prediction so produced is called motion compensated prediction [4].

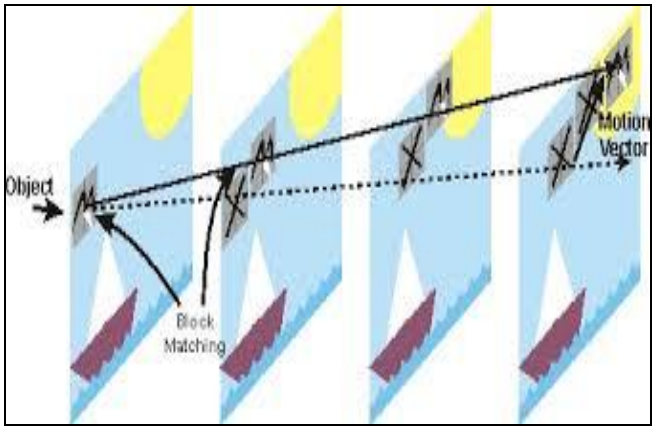

Figure 1: Block Matching [1]

Block-matching motion estimation (BMME) is the most widely used estimation method for video coding. Some of the more popular methods including Block Matching Algorithms (BMA), parametric/motion models, optical flow, and pixel-recursive techniques. Among these methods, BMA seems to be very popular method due to its effectiveness and simplicity for software and hardware. BMA is also used in all international video coding standards like MPEG-1, MPEG-2, H.261 and H.263 [5].

Three step search (TSS) is the simplest algorithm among all and it has faster estimation. A New three step search (NTSS) algorithm is advanced version of TSS which has high computational requirements. Four step search (FSS) has further been introduced to reduce search points of TSS and NTSS, but its performance is similar to NTSS in terms of motion compensation error [6].

If larger central search pattern is used then FSS and Diamond Search (DS) can provide faster search speed than NTSS. Moreover, they give better motion estimation and direction

due to larger size. Cross diamond search (CDS) has performed faster than DS [8].
The paper is organized as follows. In section II, we discuss related work with the motion estimation and compensation. In Section III, It describes various algorithms of motion estimation. In Section IV, it describes the proposed technique related to system. Finally, conclusion is given in Section V.

II. RELATED WORK

In literature, it proposed a search method based on conjugate instructions, and another simpler technique called the one-at-a-time exploration. It is based on the comparison of the two methods; the later technique is accepted as the basis for further investigation. The adopted technique is linked with brute force search, existing 2-D logarithmic search, and a altered version of it, for motion compensated prediction. Extensive computations are required to estimate the motion on a block-by-block basis. These motion estimation techniques can be applied to video sequences. Their superior performance compared to the existing techniques and it is based on quantitative measures of the prediction error [6].

The three step search (TSS) procedure has been mainly used as for estimating motion between frames in some low bit rate video compression application and it is a simple and effective procedure. However, TSS uses a uniformly assigned checking point pattern in its first step. But first step is incompetent for the estimation of small motions. A new three step search (NTSS) algorithm employs a centre-biased testing point pattern in the first step, which is derived by making the search adaptive to the motion vector distribution. It also employs a halfway-stop technique to reduce the computation cost. NTSS is much more robust than TSS, produces smaller motion recompense errors, and has a very compatible computational complication [8].
A new four step search (FSS) procedure with centre biased testing point pattern for fast block motion approximation. In this paper, halfway stop technique is employed in the new algorithm with searching step of 2 to 4 and the total number of checking point is varied from 17 to 27. FSS performs better than the three step search technique. It has comparable performance to the three step search (NTSS) in terms of motion compensation errors. It also reduces the computational complexity from 33 to 27 search point and the average computational requirement from 21 to 19 search points as compared with NTSS [9].
A gradient descent search process was used to perform block motion approximation in video coding. The process evaluated the values of a given objective function started from a small checking block. The minimum value within the checking block was found. Then block based gradient descent direction where the minimum was expected to lie was used to determine the search direction and the position of new checking block. The BBGDS was compared with full search (FS), three step search (TSS), and new three step search (NTSS). Experiment results showed that the proposed technique provided better performance and reduce computational complexity [10].

The widespread use of block-based inter-frame motion estimation for video sequence compression in both MPEG and H.263 standards. It has properties of effectiveness and simplicity of implementation. The high computational complexity of three full search algorithms has motivated a host of suboptimal but faster search strategies and its example is the three step search (TSS) algorithm. Then, the four step search (FSS) algorithm was introduced to reduce the average case from 21 to 19 search points. This paper proposes a novel unrestricted centre biased diamond search (UCBDS) algorithm which is more efficient [11].

III. VARIOUS MOTION ESTIMATION ALGORITHMS

There are two basic approaches to motion estimation.

- Pixel based motion estimation
- Block-based motion estimation

The pixel based motion estimation approach seeks to determine motion vectors for every pixel in the image. This is also referred to as the optical flow method, which works on the fundamental assumption of brightness constancy that is the intensity of a pixel remains constant, when it is displaced. In block based motion estimation, the candidate frame is divided in to non-overlapping blocks and for each such candidate block, the best motion vector is determined in the reference frame. Here a single motion vector is computed for the entire block, whereby we make an inherent assumption that the entire block undergoes translational motion. From the last 2 decades various fast algorithms are proposed to pursue low computational complexity, some of them are TSS, NTSS, FSS, DS, Hexagonal Search algorithm etc.

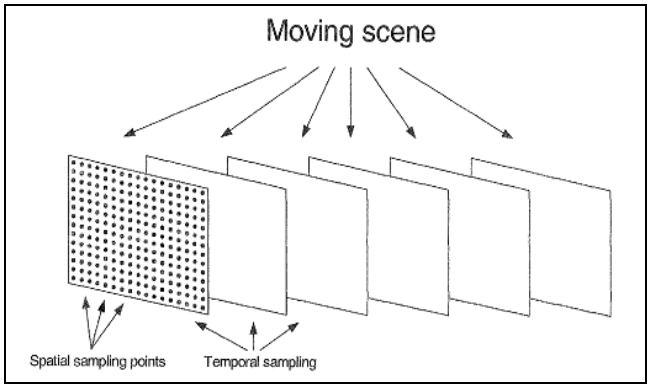

Figure 2: Spatial and Temporal Sampling [1]

A. *Three Step Search Algorithm [TSS]*

The TSS algorithm has been widely used as the motion estimation technique in some low bit rate video compression application, owing to its simplicity and effectiveness. However, TSS uses a uniformly allocated checking point pattern in its first step, which becomes inefficient for the estimation of small motions. To remedy this problem, several adaptive techniques have been suggested to make the search more adaptable to motion scale and uncertainty. Three step searches employs rectangular search pattern with different sizes [12].

B. *New Three Step Search Algorithm [NTSS]*

New Three Step Search is used for fast motion estimation. In this algorithm, the search pattern in each step is fixed and no threshold operations are involved in this algorithm. Nevertheless, it is made to better utilize the motion

distribution of real world image sequence in low bit-rate video applications and is adaptive in the ruled sense that the algorithm may stop at the second or third step. NTSS retains the simplicity and regularity of the original TSS method [13].

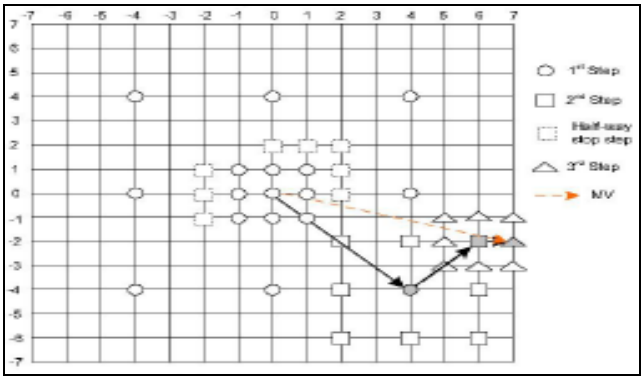

Figure 3: New Three Step Search [2]

It works better than TSS in terms of motion compensation error and robustness, and is quite compatible to TSS in terms of computational complexity [2]. NTSS solves the problem of TSS that it uses a centre biased checking pattern in the first step by adding a smaller 8 point pattern combined with the original TSS search pattern. It also includes a half way stop system for stationary and quasi-stationary chunks. It holds the simplicity and regularity of TSS with speed and accuracy. The drawback of NTSS is that, in the worst case, it requires more search points. For a ±7 search windows NTSS requires 33 search points as compares to 25in TSS. For large motion contents videos, NTSS may need more computations [14].

A. Four Step Search Algorithm [FSS]

FSS produces better performance than the TSS and has similar performance as compared with the NTSS. For the maximum motion displacements of ±7, the proposed FSS algorithm utilizes a centre-biased search pattern with 9 checking points on a 5×5 window in the first step instead of a 9×9 window in the TSS.FSS gives a speed up of six block matches for the worst case, and an average of two block matches less than the NTSS. More importantly, FSS still manages to maintain motion estimation performance comparable to the NTSS, which in turn is better than the TSS [5].

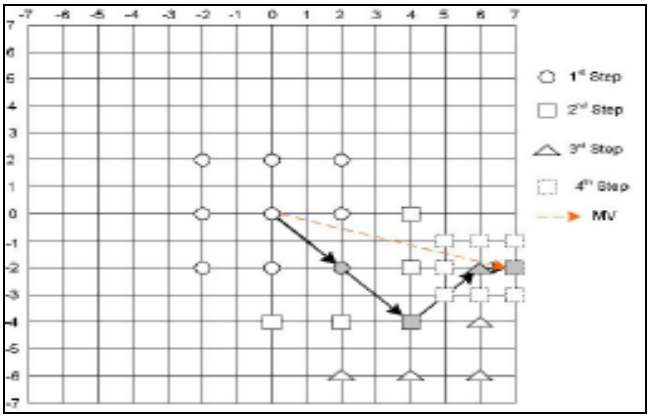

Figure 4: Four Step Search [2]

B. Diamond Search Algorithm [DS]

Diamond search algorithm provides faster searching speed than NTSS. It gives consistently better motion estimation and directions due to larger size. DS uses the same concept as the FSS but its patterns are skewed by 45° (fig. (c)). It maximizes overlap in consecutive iterations to reduce search points but also tries to reach out as far as possible. This makes it faster and more robust than its predecessors. In the final step, a 4 point pattern is used. It is highly centre-biased and has a compact diamond search pattern [15].

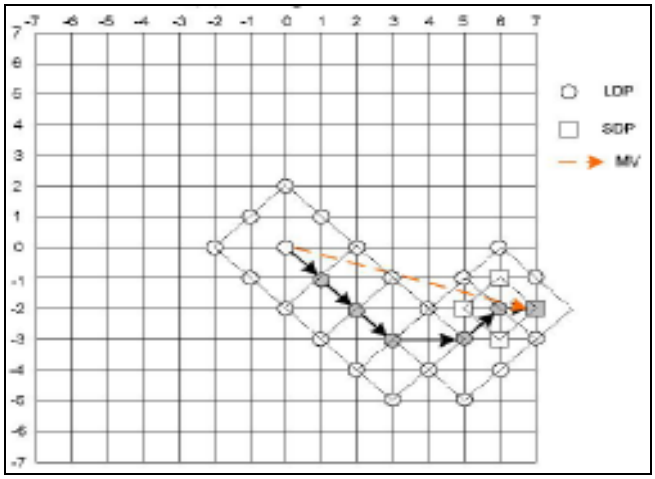

Figure 5: Diamond Search [2]

C. Cross-Diamond Hexagonal Search Algorithm [CDHS]

Another motion estimation algorithm is cross-diamond hexagonal search which is differing from DS, CDS (Cross Diamond Search) by performing a high cross-centre biased search within the first step. In addition, the search may involve up two different patterns: diamond shaped LDSP and hexagonal shaped LDSP. The common strategy amongst them is employing a halfway-stop technique [5].

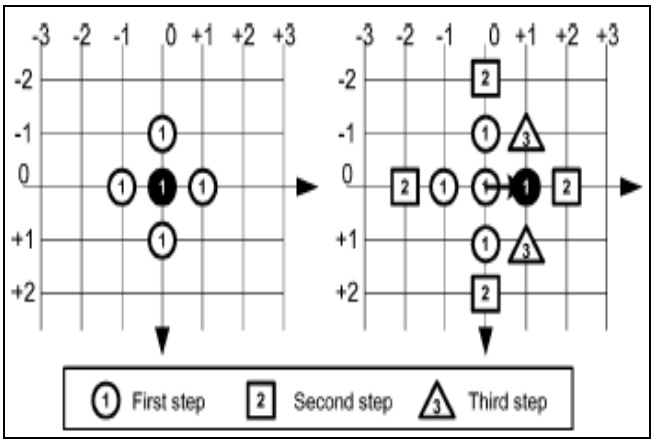

Figure 6: Cross Diamond Hexagonal Search [2]

IV. PROPOSED COMPENSATED FRAME PREDICTION TECHNIQUE

A video sequence can be considered to be a discretized three-dimensional prediction of real four-dimensional continuous space-time. The objects in real world may move, rotate, or deform. The movements cannot be detected directly, but instead the light reflected from the object surfaces and estimated onto an image. The light source can be moving, and the reflected light varies depending on the angle between a surface and a light source. A perfect motion model would take all the factors into account and find motion that has the maximum likelihood from observed video sequence.

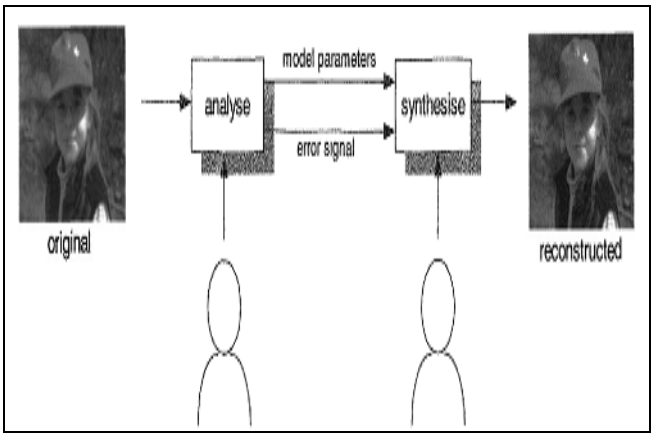

Figure 7: Model Based Coding

The motion estimation module creates a model by modifying one or more reference frames to match the current frame as closely as possible (according to 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 for the decoder to recreate the model (typically a set of motion vectors). 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

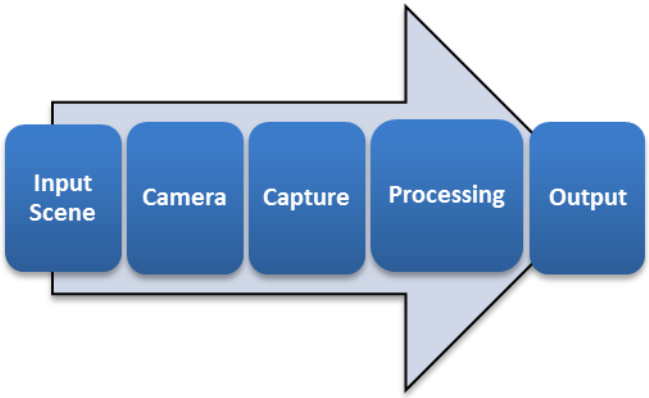

Figure 8: Proposed Steps in Coding

Motion estimation (ME) methods have been successfully useful in motion compensated analytical coding for reducing temporal redundancies. They belong to the class of nonlinear predictive coding systems. In this, it requires the interfacing of webcam with computer. Making the interface between computer and the camera can be done using lower level language like C or C++ will give lots of elasticity, but it will also appeal lots of work and background knowledge. We use MATLAB for this purpose. In this, it depends upon forward prediction coding. In this, we use the frame rate of 30 fps with RGB image. Initially set the large buffer size for frames storage and also set the triggering manually.

Forward prediction involves using an 'older' encoded frame (i.e. a preceding frame in temporal order) as prediction reference for the current frame. Forward prediction performs poorly in certain cases, for example: when there is a significant time difference between the reference frame and the current frame (which may mean that the image has changed significantly); or when a scene change or 'cut' occurs.

V. CONCLUSION

Motion Estimation forms a major computation bottleneck in video processing applications such as the detection of noise in image sequences, interpolation/ prediction of missing data in image sequences and de-interlacing of image sequences. In this, it proposes a compensated motion frame prediction and detection using block based and pixel based technique. The main objective is to reduce the computation time and improves the PSNR value of system. Sometimes, the pixel value is very low, so to detect these pixels, it may use the enhancement concept to improve the pixel value so that it may detect and reconstruct that frame easily.

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] C. Zhu, X. Lin, L. Chau, "Hexagon-Based Search Pattern For Fast Block Motion Estimation", IEEE Transactions On Circuits And Systems For Video Technology, Vol. 12, No. 5, May 2002

[6] C.H. Cheung, L.M. Po, "Novel Cross-Diamond-Hexagonal Search Algorithms for Fast Block Motion Estimation", IEEE Transactions On Multimedia, Vol. 7, No. 1, February 2005.

[7] M. Gallant, F. Kossentini, "An Efficient Computation-Constrained Block-Based Motion Estimation Algorithm for Low Bit Rate Video Coding".

[8] I. Ahmad, W. Zheng, J. Luo, M. Liou, "Fast Adaptive Motion Estimation Algorithm", IEEE Transactions On Circuits And Systems For Video Technology, Vol. 16, No. 3, March 2006.

[9] K. Ng, L. Po, K.M. Wong, "Search Patterns Switching For Motion Estimation Using Rate of Error Descent", ICME 2007.

[10] S. Goel, M. A. Bayoumi, "Multi-Path Search Algorithm for Block-Based Motion Estimation", ICIP 2006.

[11] A. M. Tourapis, "Enhanced Predictive Zonal Search For Single And Multiple Frame Motion Estimation".

[12] D. Kesrarat, V. Patanavijit, "Empirical Study on Performance Comparisons ofBlock-Based Motion Estimation on Multi Sub-PixelDisplacement with Multiples Block Size", 26th International Conference on Advanced Information Networking and Applications Workshops, 2012.

[13] S. Klomp, M. Munderloh, Y. Vatis, J. Ostermann, "Decoder-Side Block Motion Estimation For H.264 / Mpeg-4 Avc Based Video Coding" IEEE Institute For Informations, 30167 Hannover, Germany2009.

[14] L.M. Po, C. W. Ting, K.M. Wong, K.H. Ng, "Novel Point-Oriented Inner Searches For Fast Block Motion Estimation", IEEE Transactions On Multimedia, Vol. 9, No. 1, January 2007.

L.M. Po, K.H. Ng, K.M. Wong, K.W. Cheung, "Multi-Direction Search Algorithm for Block-Based Motion Estimation" Department Of Electronic Engineering, City University Of Hong Kong,2008.