Study of the Contemporary Motion Estimation Techniques for Video Coding

S. Ashwin, S. Jayanthi Sree, S. Aravind Kumar

Abstract*— *Video compression is vital for efficient archival of entertainment based video (CD/DVD) as well as real-time reconnaissance / video conferencing applications. While ISO MPEG sets the standard for the former types of application, ITU sets the standards for latter low bit rate applications. In the entire motion based video compression process motion estimation is the most computationally expensive and time-consuming process. Motion estimation involves interframe predictive coding, one of the most powerful image coding techniques which calculates motion vectors and can eliminate redundancy in natural scenes. The research in the past few decades has focused on reducing both of these side effects of motion estimation. This paper reviews the literature pertaining to the different types of contemporary block matching algorithms used for motion estimation in video compression. The algorithms that are illustrated in this paper can be adopted by the video compressing community to be implemented in various standards. In addition, the advantages, limitations and applications of these techniques are revealed and guidelines for future research are discussed.

Index Terms*— *Video Compression, Motion Estimation, Motion Vectors, predictive coding.

I. INTRODUCTION

The present era is an age of multimedia and multimedia has bolstered the spread of Internet, video storage on CD/DVD and streaming video. High-definition archiving is a better known application that has been gaining a lot of popularity. The ISO Moving Picture Experts Group (MPEG) video coding standards pertain towards compressed video storage on physical media like CD/DVD, where as the International Telecommunications Union (ITU) addresses real-time point-to-point or multi-point communications over a network. The former has the advantage of having higher bandwidth for data transmission. As a result, compression and the coding of video sequences in turn has been the focus of a great deal of research in recent years.

Successive video frames may contain the same objects (still or moving). Motion estimation examines the movement of objects in an image sequence to try to obtain vectors representing the estimated motion. Motion compensation uses the knowledge of object motion so obtained to achieve data compression. In Interframe coding, motion estimation and compensation have become powerful techniques to eliminate the temporal redundancy due to high correlation between consecutive frames.

Motion estimation (ME) is one of the most computational intensive operations in video compression. It can easily account for over 80% of the computation in an MPEG-2 video encoder. Efficient motion estimation reduces the energy in the motion-compensated residual frame and can dramatically improve compression performance. The motion estimation creates a model by modifying one or more reference frames to match the current frame as closely as possible. 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 by determining a set of motion vectors (MV). The computational complexity of a motion estimation technique can then be determined by three factors:

1. Search algorithm.
2. Cost function/evaluate function.
3. Search range parameter

Block matching techniques are widely used motion estimation method to obtain the motion compensated prediction. By splitting each frame into macroblocks, motion vector of each macroblock is obtained by using block matching algorithm (or motion estimation algorithm). In order to get motion vector of each macroblock, the most obvious and simplistic method is full search algorithm. All possible displacements in the search window are evaluated using block-matching criteria (cost function). The advantage of full search is that we can find the absolute optimal solution. However, its high computational complexity makes it impossible for real-time implementation. Because the computational complexity of video compression, the compression efficiency and the compression quality is determined by the motion estimation algorithm, development of Fast Motion Estimation Algorithm for real-time application becomes compelling.

A lot of motion estimation algorithms have been invented to reduce the complexity of motion estimation associated with full search. Some reduce the calculation of cost function. Others reduce the search points. Different block motion estimation algorithms are available, which include full search or exhaustive search algorithm, the fast search algorithm, such as three step search, two dimensional logarithmic search, binary search, etc.

Revised Manuscript Received on 30 March 2013.
* Correspondence Author
S. Ashwin, Post Graduate Student, PSG College of Technology, Coimbatore, India.
Jayanthi Sree, Assistant Professor, Dr. NGP Institute of Technology, Coimbatore, India.
S. Aravind Kumar, Graphics Hardware Engineer, Intel Technologies India Pvt Ltd, Bangalore, India.

There are many computational effective block motion estimation algorithms but with trade-off between the algorithm accuracy and algorithm speed. Also motion estimation in DCT domain & motion estimation in wavelet domain give better performance. DCT based motion estimation has low computational complexity than full search block matching approach. Wavelet based coding has better performance than traditional DCT based coding, since it reduces artifacts in the DCT coding.

Given the proliferation of novel and efficient motion estimation techniques in the past few decades this paper will focus on few of the modern methods of motion estimation and highlights the advantages and future scope of each of these. The paper is organized as follows. Section 2 describes the various types of the most recent motion estimation techniques followed by conclusion in Section 3.

II. MOTION ESTIMATION TECHNIQUES

A. 3D Motion Estimation

Kun Li et al [1] proposed a new three dimensional two stage motion estimation technique based on matrix completion from multiview video sequences. The two stages involved are separating and merging. This involves estimating the initial motions for each view with a neighboring view and later merging each view. The main advantage of the former stage is accuracy of 3-D motion and later stage assures completeness (denseness) of the 3-D motion field.

The recovery of low-rank data from a highly incomplete set of possibly corrupted entries is referred to as matrix completion. This method is the latest development of sparse representation. It is much preferred because it can be used to optimize and solve the three characteristics of the 3-D motion estimation namely high dimensionality, low rank and noise.

In the proposed method, a 3-D motion is estimated via matrix completion and then improves the accuracy with spatiotemporal selection. Particularly, initial motion is estimated for each camera and treated as a column vector to form an incomplete and noisy matrix. In view of the fact that all the estimated results correspond to the same 3-D motion field, the underlying clean version of these vectors lies in a low-dimensional subspace, i.e., the matrix has a low rank which is theoretically one. Hence, the problem of 3-D motion estimation is converted to a problem of recovering a low-rank matrix from noisy and incomplete observations. With this we obtain a complete matrix X with each row corresponding to a vertex and each column corresponding to a view. Finally, accurate estimates are obtained by spatiotemporal selection as a 2-D constraint. Three different spatiotemporal selection criteria are based on spatial consistency, temporal consistency and smoothness.

<u>Advantages and Future Scope:</u>

This method is totally automatic, fast and can estimate both rigid and non-rigid motions within small errors. This work can be extended in future to achieve high- speed motion capture by designing a flexible capturing system without using high-speed cameras.

B. Zoom Motion Estimation Method

When Hyo-Sung Kim, et al [2] developed a novel Zoom Motion estimation (ZME) by employing block-based fast local area scaling. The limitation of traditional block-based ME technique of being used only in parallel translation between frames has been rectified in this method.

The ZME algorithm searches the best zoom vector (ZV) and the motion vector (MV) *mv(x, y)*. Fig. 1 shows the working of ZME algorithm.

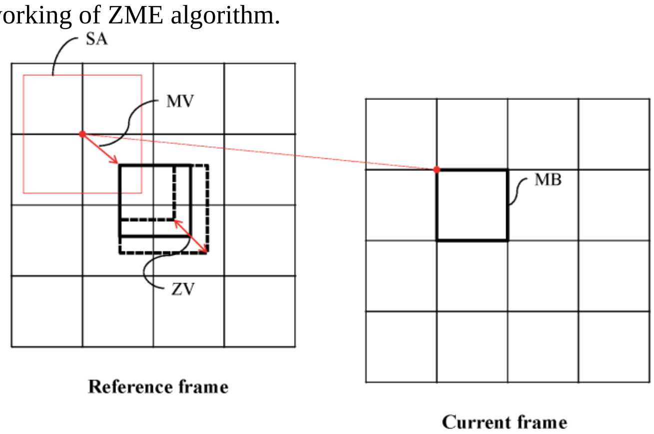

Fig 1. Working of ZME algorithm.

Zoom motion occurs when objects move toward or away from a camera. Therefore, an object near the camera is seen as large, and an object far from the camera is seen as small. Hence, the ZV increases or decreases the candidate block size to achieve the zoom image. The zoomed candidate block is then scaled to the same size as the current block by interpolation for block matching with the current block. Three additional parameters are used namely ZV, zoom range and zoom scale for ZME given by the below relationship

$$\mathbf{zbs = bs + (bs \times zv/zs)} \qquad (1)$$

Where bs is block size, zbs the zoom block size and zs the zoom scale which is defined as a scale ratio for the zoom in each step and the ZV represents the amount of the step.

The candidate block size is increased or decreased to produce a zoomed image and then resized the image to match the size of the current block using bilinear interpolation in order to perform block matching. Finally, a 3-D fast pattern search algorithm is applied to more efficiently find the zoom MV.

<u>Advantages:</u>

Significant advantages are

- 20.73% savings in the bit-rate with a quality enhancement of 1.09 dB in PSNR gain.
- Reduced complexity
- no requirement for memory buffer
- Compatible with existing standard methods like MPEG- 2, H.264/AVC) and the next-generation codec (HEVC)
- Independent of image sequence resolution, the proposed algorithm yielded good results

C. Fast Camera Motion Estimation

Discrete Wavelet Transform is used by Manish Okade and P.K Biswas on the block motion vector to estimate the camera motion parameters in the compressed domain [3].

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

The camera motion parameters are estimated between every successive frame for the entire video sequence. The 8-parameter perspective projection model which is employed in this work for modeling the camera motion is given by equations 3 and 4. The perspective model parameterized by an 8-dimensional vector *m = [m0, . . . , m7].*

$$x' = \frac{m_0 x + m_1 y + m_2}{m_6 x + m_7 y + 1} \qquad (2)$$

$$y' = \frac{m_3 x + m_4 y + m_5}{m_6 x + m_7 y + 1} \qquad (3)$$

where (x,y) is the point in reference frame and (x',y') is the point after the frame has undergone perspective motion represented by the transformation matrix. The wavelet transform of the block motion vector field is performed to decompose the motion vector field into sub-bands. This is followed by gradient descent regression on the LL sub-band wavelet coefficients to estimate the camera motion parameters.

Advantage and application:

The pro of this approach is significant gain in processing time but the con is marginal drop of estimation accuracy for the camera motion parameters. It has been proved through results that 12% savings in the processing time is achieved. Few applications where this technique can be applied are video indexing and video shot segmentation where fast global motion estimation is the requirement and marginal drop of estimation accuracy can be tolerated.

D. Adaptive Block Size for Motion Compensated Frame Interpolation

Hyungjun Lim et al [4] proposed an innovative frame rate up-conversion (FRUC) method. Frame rate up-conversion (FRUC) is a procedure that increases the frame rate of video sequences by generating intermediate frames between successive frames. The application of FRUC is for reducing motion blurriness of displays such as liquid crystal and plasma displays and for reconstructing the frames skipped for data compression. But it produces jerkiness and blurriness around moving objects. As a result a motion-compensated frame interpolation (MCFI) algorithm is used in this method to overcome this limitation of FRUC algorithms. Here L represents the increment level and SABD is the sum of absolute difference between bidirectional predictions.

Fig 2 shows the flowchart of the proposed MCFI algorithm. For each block of $N_0 \times N_0$ the proposed method performs bidirectional motion estimation and obtains the motion vector of the minimum SABD, v, and the refined PMV, v_{rPMV}. Then, if $R(v, v_{rPMV}, N_0)$ is larger than a threshold, we decide that v is reliable. Otherwise, we repeat the bidirectional motion estimation with a increased block size and new v and v_{rPMV} are obtained. R_{th} represents the threshold for reliable SABD comparison. This value is determined as $R_{th} = 0.5$.

Advantages:

- It interpolates successive frames more accurately than the any other traditional MCFI.
- Higher PSNR. (5.0 dB improvement)
- Generates accurate MVF
- Computationally faster

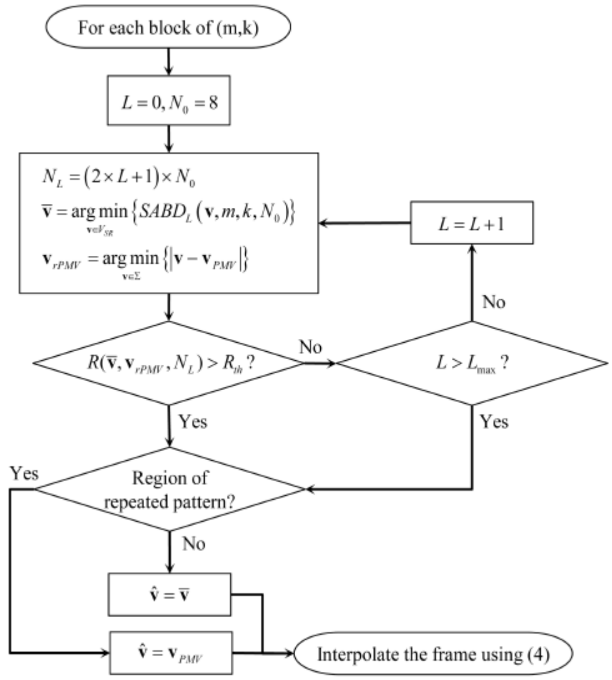

Fig 2. Flowchart of the MCFI algorithm.

E. Motion Region Identification

A speed-up motion estimation algorithm is proposed by Dongming Zhang et al in 2012 which relies on the fact motion region needs more searching points [5]. Efficient motion region detection algorithm based on Visual Rhythm Analysis (VRA) [6] is proposed, which can identify the motion region with a little additional computation.

Motion regions and still regions in picture are first marked using improved VRA at the beginning of one group of picture (GOP) encoding, then one improved motion estimation proceeds. The new motion estimation eliminates unnecessary search stages according to macroblock (MB) motion. It carries out different search scheme in motion MBs and still one. In particular, for motion MBs, it makes full use of the neighboring reference block motion information to choose motion search center (the search center has three types, i.e. integer, half or quarter pixel accuracy search center). Then an adaptive search range is defined according to the type of search center. The proposed search technique is called Search Center Adaptive Motion Estimation (SCAME). The SCAME selects adaptive search center and applies different DS pattern to reduce search points according to MB's mark.

Advantages and Future Scope:

It can eliminate integer pixel search process of 25% blocks with a little performance loss even for video sequence with moderate or high motion. However, the camera pan motion possibly leads to more encoding performance loss. Hence, as an enhancement in future it is possible to track the motion and update motion region after motion region identification in one GOP encoding.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

F. Error Surface-Aware Modeling Algorithm

There are two quarter pixel based motion estimation algorithms previously developed by Junsang Cho et al namely model-based quarter-pixel motion estimation (MBQME) algorithm [7] and hierarchical modelbased quarter-pixel motion estimation (HMBQME) algorithm [8]. Nevertheless, MBQME is an interpolation- free algorithm that has a minimum motion estimation time, while HMBQME has selective interpolation according to the decision process. Consequently, the peak signal-to- noise ratio (PSNR) for HMBQME is better than that of MBQME, but the motion estimation time is also increased.

As an alternative method, we propose an error surface-considered modeling algorithm [9]. In this scheme, the tendency of the error surface is first assessed. Using the strength of the edge at the error surface, we can classify the error surface region as plain or textured. For plain regions, interpolation-free and simple structured modeling is appropriate for the quarter-pixel motion estimation method. In this case, we modified conventional mathematical modeling algorithm suitable for plain region. For textured regions, additional interpolation is needed for more accurate modeling. We calculate the half-pixel SAD values and perform more accurate modeling so as to find the best motion vector (MV).

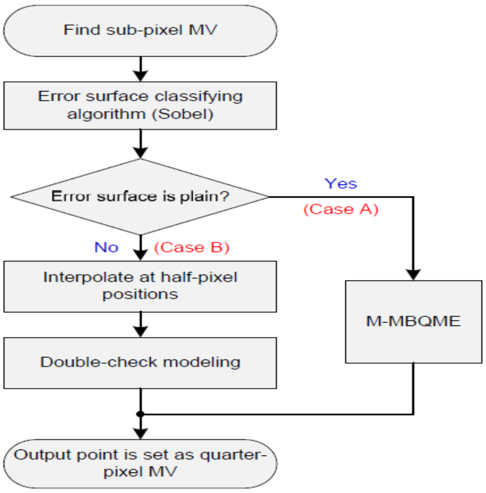

Fig 3. Flowchart of the Error Surface-aware modeling algorithm

Fig 3 illustrates the flowchart of this algorithm. The experimental results show that the proposed scheme has better PSNR performance than any previous algorithms with minimum motion estimation time.

<u>Advantages:</u>

- PSNR better than MBQME and nearly same as HMBQME.
- It is 35% faster than the original MPEG-4 algorithm.
- Utilizes minimum bits per frame
- Better estimation model than competitive algorithms.

G. Winner-Update Strategy

Discrete The compression ratio is in general lower due to the higher matching error. This is the shortcoming not guaranteeing the global minimum for the application in video compression.

The winner-update strategy, proposed by Yong-Sheng Chen, Yi-Ping Hung, and Chiou-Shann Fuh [10], utilizes an ascending lower bound list of the matching error to determine the temporary winner. The basic idea is to avoid, at each search position, the computation of the more costly matching error when there exists a lower bound larger than the global minimum matching error. This strategy is a simple and efficient block matching algorithm. It uses a comparison strategy to switch the computation among the search positions depending on the comparison of the updated partial matching error of the temporary winner with the *so-far minimum partial matching error*. This combination speeds up the three step search without decreasing its PSNR value. The proposed algorithm is evaluated in comparison with five other algorithms based on three metrics like PSNR, number of operations to calculate the matching error and the execution time. These algorithms were implemented in C language on a Sun Ultra-1 workstation.

<u>Advantages:</u>

- The computational efficiency is irrelevant to the visiting order of the search positions.
- The total computational cost is significantly reduced.
- It can save 91.6% to 98.0% of the absolute operations needed by the FS algorithm, depending on which test image sequence is used.
- 86.6% to 96.0% of the execution time can be saved, compared to that of using the FS algorithm.

<u>Future scope:</u>

Authors suggest the below two research directions

- A fast algorithm for k-nearest neighbor search by using the winner-update strategy.
- To adopt Haar wavelet transform, instead of block sum pyramid method, to construct the multilevel images

<u>Applications:</u>

- The winner-update algorithm is suitable for serial computation.
- In stereo vision and visual tracking applications
- In the applications of on-line video compression such as video conference and video phone

H. Hybrid Predictor and Search Pattern Method(HPS)

The HPS method for fast motion estimation was developed by Hui-Yu Huang and Shih-Hsu Chang by combining predictor and search pattern for estimating motion vectors (MV's) with fast speed [11]. This technique not only speeds up the search time for finding the MVs but has a high accuracy in estimating the MVs. It takes advantage of the correlation between MVs in both spatial and temporal domains, predictor, search pattern, and early termination strategy to achieve the best MV estimation. Using the context relationships and the variance of the neighborhood blocks, MVs of the current frame can easily predict.

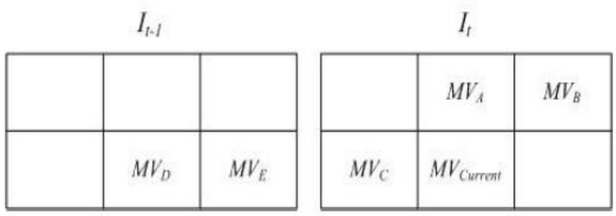

Fig 4. Illustration of predictive vectors.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

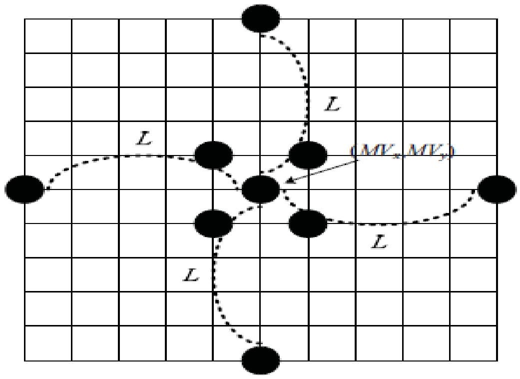

Fig 5. Predictive pattern region search (PPRS).

I_{t-1} and I_t are the previous frames and current frames respectively and the MV's of the region of support (ROS) are shown in Fig 4. The best predictor is the $MV_{current}$. If the best predictor yields the minimum sum of absolute difference (SAD) value less than the threshold then it is the optimum solution. L is maximum of MV_A, MV_B, MV_C and MV_D. .Fig 5 illustrates a basic predictive search-point configuration used in PPRS. It consists of nine candidate search points. A minimum SAD is found from the nine candidate search points of the predictive pattern located at the center (x, y) of search window. If the minimum SAD point is found to be at the center (x, y) of the PPRS, go to ending stage; otherwise go to searching stage.

III. Conclusion

The paper describes a short study on various types of novel motion estimation techniques using predictive coding for efficient video compression. Although only some of the most recent methods were discussed in this paper, one can see that there exists a large range of choice that exist for Interframe predictive coding where one system lacks in bitrates, the other lacks in complexity or cost function. Thus researchers can make a decision on which technique to choose depending on the performance of the algorithm and the nature of application developed and can carry on to improve some of these systems based on the guidelines provided for each.

REFERENCES

1. Kun Li et al, "Three-Dimensional Motion Estimation via Matrix Completion", IEEE Transactions On Systems, Man, And Cybernetics—Part B: Cybernetics, Vol. 42, No. 2, pp 539- 551
2. Hyo-Sung Kim, et al, "Zoom Motion Estimation Using Block-Based Fast Local Area Scaling", IEEE Transactions On Circuits And Systems For Video Technology, Vol. 22, No. 9, pp 1280- 1291
3. Manish Okade and P.K Biswas, "Fast Camera Motion Estimation using Discrete Wavelet Transform on block motion vectors", 2012 Picture Coding Symposium pp 333- 336.
4. Hyungjun Lim et al, "Motion Estimation with Adaptive Block Size
5. for Motion-Compensated Frame Interpolation" 2012 Picture Coding Symposium pp 325-328.
6. Dongming Zhang et al, "Improved Motion Estimation Based on Motion Region Identification", International Conference on Systems and Informatics (ICSAI 2012), pp 2034-2037.
7. [C. Ming-Chieh, et al., "Robust Region-of-Interest Determination Based on User Attention Model Through Visual Rhythm Analysis,", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 19, pp. 1025-1038, 2009.
8. J. W. Suh, J. Cho and J. Jeong, "Model-based quarter-pixel motion estimation with low computational complexity," Electronic Letters, 45, June 2009
9. J. Cho, J. W. Suh, G. Jeon and J. Jeong, "Specification of hierarchical-modelbased fast quarter-pixel motion estimation", SPIE Opt. Eng. 49, 067003 (Jun 30, 2010)
10. Junsang Cho et al, "Error Surface-Aware Modeling Algorithm for Quarter-Pixel Motion Estimation", IEEE Transactions on Consumer Electronics, Vol. 58, No. 3, pp 912 – 916
11. Yong-Sheng Chen, Yi-Ping Hung, and Chiou-Shann Fuh, "Fast Block Matching Algorithm Based on the Winner-Update Strategy", IEEE Transactions On Image Processing, Vol. 10, No. 8, Pp 1212- 1222
12. Hui-Yu Huang and Shih-Hsu Chang, "Block motion estimation based on search pattern and Predictor", 2011 IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing (CIMSIVP), pp 47-51.

AUTHOR PROFILE

S. Ashwin is currently pursuing his Master's degree in Embedded Systems from P.S.G. College of Technology, Coimbatore. He obtained his Bachelor's degree in Electronics and Communication Engineering from Sri Krishna College of Engineering and Technology, Coimbatore. His research areas include Image Processing and Wireless Communication Networks

S. Jayanthi Sree is currently working as Assistant Professor in Dr. NGP Institute of Technology, Coimbatore. She obtained her Master's in VLSI Design from GCT, Coimbatore and Bachelor's degree in Electronics and Communication Engineering from Sri Krishna College of Engineering and Technology, Coimbatore. Her research areas include Image Processing and VLSI Design.

S. Aravind Kumar is currently working as Graphics Hardware Engineer in Intel Technologies Pvt Ltd, Bangalore. He obtained his Master's degree in Communication Systems from P.S.G. College of Technology, Coimbatore and Bachelor's degree in Electronics and Communication Engineering from Sri Krishna College of Engineering and Technology, Coimbatore. Her research areas include biomedical image processing and Communication Networks.