ISSN No. 0976-5697

Volume 2, No. 3, May-June 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Motion Estimation in Video Compression and Soft Computing Techniques

Pooja Nagpal*
M.Tech Student: C.E dept.
Yadavindra College of Engineering
Talwandi Sabo, India
Poojanagpal48@gmail.com

Seema Baghla
Assistant Professor: C.E dept.
Yadavindra College of Engineering
Talwandi Sabo, India
garg_seema238@yahoo.co.in

Abstract: The demand for communications with moving video picture is rapidly increasing Video is required in many remote video conferencing systems, and it is expected that in near future cellular telephone systems will send and receive real-time video. Video compression is a technology for transforming video signals that aims to retain original quality under a number of constraints, e.g. storage constraint, time delay constraint or computation power constraint Block matching motion estimation is the essence of video coding systems. Soft Computing is the fusion of methodologies that were designed to model and enable solutions to real world problems, which are not modeled, or too difficult to model, mathematically This paper gives a review of the different block matching algorithms used for motion estimation in video compression, also introduce the soft computing techniques with its necessity in video compression .

Keywords: Video Compression; Motion Estimation; Particle Swarm Optimization;

I.INTRODUCTION TO VIDEO COMPRESSION

Video Compression refers to reducing the quantity of data used to represent video images and is a straightforward combination of image compression and motion compensation [1]. Video compression is a technology for transforming video signals that aims to retain original quality under a number of constraints, e.g. storage constraint, time delay constraint or computation power constraint. It takes advantage of data redundancy between successive frames to reduce the storage requirement by applying computational resources. Digital video is virtually immune to noise, easier to transmit and is able to provide a more interactive interface to users. In a video scene, data redundancy arises from spatial, temporal and statistical correlation between frames. These correlations are processed separately because of differences in their characteristics. Hybrid video coding architectures have been employed since the first generation of video coding standards, i.e. MPEG.

Motion Estimation is an important part of video compression systems, where the estimated motion vectors are used to produce a motion-compensated prediction of a frame to be coded from a previously coded reference frame [1].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 employed to reduce spatial dependency within a frame in the spatial domain. Entropy coding is used to reduce statistical redundancy over the residue and compression data. This is a lossless compression technique commonly used in file compression.

A. Compression Classification

The compression techniques in general fall into two main categories, entropy encoding and source encoding. The difference is that entropy encoding addresses encoding techniques that consider the nature of data, and the information it represents. On the other hand source encoding is a type of encoding that operates independent on the information that the source data represents. Image compression techniques can further be divided into two other categories, lossy and lossless. Lossless compression addresses the compression techniques designed to reduce coding and interpixel redundancy. Where

as lossy compression refers to reduction of data used to represent the image by reducing the psycho visual redundancy in the image. One could say that entropy encoding usually (although not necessarily) leads to lossy compression and source encoding leads to lossless compression.

B. Organization of the Paper

The rest of the paper is organized as follows: Section 2 describes motion estimation in video compression. The process of video compression and decompression is given in section 3. In section 4, soft computing techniques have been discussed with its necessity in video compression and brief review of different soft computing techniques with applications. The paper is summarized in the last section.

II.MOTION ESTIMATION IN VIDEO COMPRESSION

For motion estimation for each block in the frame, a search is made in previous frame of the sequence over a predefined area of the image [1]. The search is for the best matching block i.e. the position which minimizes a distortion measure between the two sets of pixels comprising the blocks. The relative displacement between the two blocks is taken to be the *motion vector*. The search area for a good macro block match is constrained up to ‘*p*’ pixels on all fours sides of the corresponding macro block in previous frame. This ‘*p*’ is called as the search parameter [1-17]. Larger motions require a larger ‘*p*’ and larger search parameter results in computationally expensive motion estimation. Usually the macro block is taken as side 16 pixels of a square (16 X 16). The matching of one macro block with another is based on the output of a cost function. Typical block sizes are of the order of 16 x 16 pixels, and the maximum displacement might be ±64 pixels from the block's original position [1-5].

The output of the motion-estimation algorithm comprises the motion vector for each block, and the pixel value differences between the blocks in the current frame and the “matched” blocks in the reference frame. The motion estimation algorithm should provide the following features.

I. Reduction in computational complexity
II. Representing true motion (proving good quality)
III. Reduction in bit rate (high compression ratio)

The traditional motion estimation algorithm uses an Exhaustive Search (ES). This is computationally demanding but algorithmically simple and relatively easily implemented in hardware. The exhaustive search (ES) [1] or full search algorithm gives the highest peak signal to noise ratio amongst any block-matching algorithm but requires more computational time. Some fast search algorithms based on square or rectangular pattern and some non-square block or geometric pattern or shapes based have been proposed so far. The rectangular pattern block based fast search algorithms are Three-Step Search (TSS) [2], 2D-logarithmic search (2DLOG) [3], orthogonal search (OSA) [4] , New three-step search (NTSS) [2], Four-step search (4SS) [5] and Block-based gradient descent search (BBGDS) [6] . One problem that occurs with the Three Step Search is that it uses a uniformly allocated checking point pattern in the first step, which becomes inefficient for small motion estimation. Although 2D-logartahamic algorithms require more steps than the Three Step Search, it can be more accurate, especially when the search window is large. Orthogonal search is a hybrid of the Three Step Search and the Two Dimensional Logarithmic Search. It has a vertical stage followed by a horizontal stage for the search for the optimal block. New Three-Step Search (NTSS) [2], Four-step search (4SS) [5] and Block-based gradient descent search (BBGDS) [6] follows center-biased property. The computational complexity of the 4SS is less than that of the TSS, while the performance in terms of quality is as good.

Non rectangular block based motion search algorithm such as the diamond search (DS) [7] and hexagon-based search (HEXBS) [10] algorithm have been proposed in literature. They required much fewer checking points, in contrast to algorithms with limited steps. DS uses two different types of fixed patterns, one is Large Diamond Search Pattern (LDSP) and the other is Small Diamond Search Pattern (SDSP) [7]. The diamond search [7] algorithms are fast search methods as they reduce the number of search points in the process of block motion estimation. The hexagonal search or hexagon-based search [10] has shown the significant improvement over other fast algorithms such as DS. Adaptive Road Pattern Search (ARPS) [11], uses the motion vector of the immediate left macro block to predict its own motion vector. The main advantage of ARPS over DS is that, if the predicted motion vector is (0, 0), it does not waste computational time in doing LDSP rather directly starts using SDSP [11].

In order to overcome the limitation of the traditional fixed-size block motion compensation, Variable-Size Block Motion Compensation (VSBMC) [12-14], Region-Wise block Motion Compensation (RWMC) [15-16] and Quad tree-Structured Variable-Size Block-Matching Motion Estimation [17] techniques are used. The fast motion estimation algorithms such as Adaptive Block Matching Algorithm [18], Efficient Block Matching Motion Estimation [19], Content Adaptive Video Compression [20] and Fast motion estimation algorithm [21] have been proposed by researchers.

III.VIDEO COMPRESSION AND DECOMPRESSION PROCESS

The basic flow of the entire video compression and decompression process is shown in Fig.1. The motion with current frame is estimated with respect to previous frame in the encoding side. A motion compensated image for the current frame is then created that builds up of blocks of images from the previous frame. The motion vectors for blocks used for motion estimation are transmitted, as well as the difference of the compensated image with the current frame is also encoded and are sent. The sent encoded image is then decoded at the decoder and is used as a reference frame for the subsequent frames. The decoder reverses the process and creates a full frame.

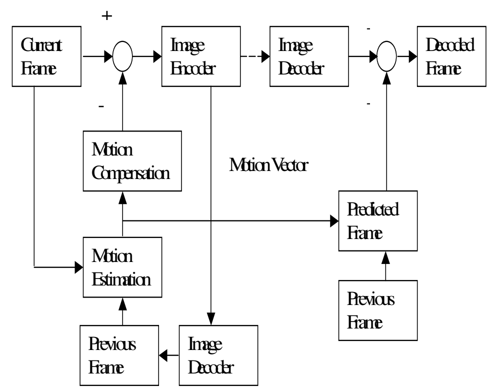

Figure 1. Video Compression and Decompression Process

The three basic steps as given below are used in motion estimation.

- The current frame is divided into macro block.
- Each macro block is compared to a macro block in the reference frame using some error measure and the best matching macro block is selected.
- A motion vectors for each macro block is calculated.

IV.SOFT COMPUTING

Soft Computing is the fusion of methodologies that were designed to model and enable solutions to real world problems, which are not modeled, or too difficult to model, mathematically. In real world, we have many problems which we have had no way to solve analytically, or problems which could be solved theoretically but actually impossible due to its necessity of huge resources and/or enormous time required for computation. For these problems, methods inspired by nature sometimes work very efficiently and effectively. Although the solutions obtained by these methods do not always equal to the mathematically strict solutions, a near optimal solution is sometimes enough in most practical purposes. These biologically inspired methods are called Soft Computing The main goal of Soft Computing is to develop intelligent machines and to solve nonlinear and mathematically unmodelled system problems.

Soft computing techniques resemble biological processes more closely than traditional techniques, which are largely based on formal logical systems such as sentential logic and predicate logic or rely heavily on computer-aided numerical analysis (as in finite element analysis). Soft computing techniques are intended to complement each other. Unlike hard computing schemes, which strive for exactness and full truth, soft computing techniques exploit the given tolerance of imprecision, partial truth, and uncertainty for a particular problem [23].

The applications of Soft Computing have proved two main advantages. First, it made solving nonlinear problems, in which mathematical models are not available, possible. Second, it introduced the human knowledge such as cognition, recognition, understanding, learning, and others into the fields of computing. This resulted in the possibility of constructing intelligent systems such as autonomous self-tuning systems, and automated designed systems.

A. Necessity of Soft Computing for Video Compression

Motion estimation is a multi-step process that encompasses techniques such as motion vector prediction, determination of search range and search patterns and identification of termination criteria. Each of these techniques has several diversions that may suit to a particular set of video characteristics. It would be hard to conceive a universal algorithm that can perform well for all kinds of video contents. However, if important characteristics of a video sequence can be identified and utilized for adjusting various steps of motion estimation, one can design an adjustable algorithm that can tune its parameters to suit the video at hand.

Real videos contain mixture of motions with slow and fast contents. No fixed fast block-matching algorithm can efficiently remove temporal redundancy of video sequences with wide motion contents. Larger motions require a larger search parameter but it makes the process of motion estimation more computationally expensive. Amongst all block-matching algorithms, in order to find the block with minimum distortion, the full search (FS) motion estimation algorithm matches all possible displaced blocks within the search area of the reference frame. Massive computation is required for the implementation of FS. Many fast algorithms have been developed like the TSS, NTSS, 4SS, DS, etc. to reduce computational complexity. These algorithms are faster because only selected possible displaced blocks are matched within the search area in the reference frame to find the block with minimum distortion. It is known that in terms of computation these algorithms are better but the PSNR of these algorithms are low as compared to FS. These are also motion dependent. The cause of failure of these algorithms is initial static search that suffers from limitation to detect true motion vector. The fixed step size algorithms are suitable for small or large motion depending upon the step size. Similarly, the prediction of true motion vector depends upon the step size. Therefore, in any motion estimation algorithm, step size plays vital role for getting true motion vectors without loosing the quality of video. Compared with fixed step-size motion estimation, the adaptive step algorithm improves motion estimation and hence overall video encoding speed.

Therefore, adaptive step size can be for getting true motion vectors. The soft computing technique can be used to achieve adaptive step size for motion estimation.

B. A Basic Review of Soft Computing Techniques

Soft computing also allows complex of methodologies that embraces approximate reasoning, imprecision, uncertainty and partial truth in order to mimic the remarkable human capability of making decisions in real-life, ambiguous environments. Soft computing has therefore become popular in developing systems that encapsulates human expertise. The applications of soft computing covers a wide range of application areas, including optimization, data analysis and data mining, computer graphics and vision, prediction and diagnosis, design, intelligent control, and transportation systems. Computational time is bottleneck of many soft computing techniques. Another bottleneck to soft computing techniques is the problem of creating infeasible solution when computational time required is more. On the other hand, local search algorithms can converge in a few iterations but lack a global perspective. The combination of global and local search procedures must offer the advantages of both soft tools while offsetting their disadvantages. The high computational cost and need for accuracy of global solution have forced the researchers to develop efficient soft computing technique in terms of new, modified or hybrid soft computing techniques.

Soft computing techniques like evolutionary algorithms, bio-inspired algorithms, metaheuristics algorithms, artificial neural networks are extensively used in engineering and science to analyze very complex phenomena for which conventional methods are not suitable. These are stochastic methods that emulate biologic processes or natural phenomena. The capability to find a global optimum, without being trapped in local optima, and the possibility to well face nonlinear and discontinuous problems, with great numbers of variables, are some advantages of these techniques. These methods avoid the need to compute any derivatives.

Many evolutionary algorithms like, Genetic Algorithm (GA)[26-29], Particle Swarm Optimization (PSO) [23-30], Bacterial Foraging Optimization (BFO)[32], Biogeography Based Optimization (BBO)[44], Differential Evolution (DE)[45-46], Ant colony optimization(ACO)[48] etc. are developed to meet the needs of the researchers. Since 1960's Genetic Algorithm (GA) proves its dominant role in the optimization world [33-43]. But the limitation of getting trapped in local minima and three step procedures such as selection, crossover and mutation increase computational time; forced the researchers to search for more efficient optimization techniques [33-39]. The Particle Swarm Optimization (PSO) is a population-based optimization method developed by Eberhart and Kennedy in 1995[23-31]. It was inspired by social behavior of bird flocking or fish schooling. It can handle efficiently arbitrary optimization problems [23-31]. K. M. Passino conceived BFO in 2002[32]. BFO is based on the foraging behavior of Escherichia Coli (E. coli) bacteria present in the human intestine. BBO is a biology inspired optimization technique developed by Dan Simon in 2008[44]. It is inspired by mathematical models of biogeography by Robert MacArthur and Edward Wilson [44]. DE is a method for doing numerical optimization without explicit knowledge of the gradient of the problem to be optimized [45-46]. An Optimization algorithm modeled on the actions of an ant colony is named as ACO [48].

C. Applications of Soft Computing Techniques

- Application of soft computing to handwriting recognition
- Application of soft computing to automotive systems and manufacturing
- Application of soft computing to image processing and data compression
- Application of soft computing to architecture
- Application of soft computing to decision-support systems
- Application of soft computing to power systems
- Neuro fuzzy systems
- Fuzzy logic control

V. SUMMARY

With the increasing popularity of technologies such as Internet streaming video and video conferencing, video compression has became an essential component of broadcast and entertainment media. Motion Estimation (ME) and compensation techniques, which can eliminate temporal redundancy between adjacent frames effectively, have been widely applied to popular video compression coding standards such as MPEG-2, MPEG-4. Traditional fast block matching algorithms are easily traped into the local minima resulting in degradation on video quality to some extent after decoding. Since Evolutionary Computing Techniques are suitable for achieving global optimal solution, these techniques are introduced in this paper to estimate motion. Soft computing is likely to play an especially important role in science and engineering, but eventually its influence may extend much farther.

This paper gives a review to motion estimation in video compression its techniques and also introduce soft computing techniques used for motion estimation. This may help us to predict future trends or behaviors, allowing business to make proactive and knowledge-driven decisions

VI.REFERENCES

[1] Jianhua Lu., and Ming L. Liou, “A Simple and Efficient Search Algorithm for Block Matching Motion Estimation,” *IEEE Transactions Circuits and Systems for Video Technology*, vol.7, no.2, pp. 429- 433, April 1997.

[2] Li R., Zeng, B. and Liou M.L., “A new three-step search algorithm for block motion estimation,” *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 4, pp.438–442, 1994.

[3] A. Puri, H. M. Hang and D. L. Schilling, “An efficient block matching algorithm for motion compensated coding,” *IEEE International Conference, Acoustic, Speech, Signal Processing*, 1987, pp. 1063–1066.

[4] Lai-Man Po and Wing-Chung Ma, “A Novel Four Step Search Algorithm For Fast Block Motion Estimation,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no.3, pp 313-317, June 1996.

[5] L. K. Liu and E. 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. 4, pp. 419–423, Aug. 1996.

[6] S. Zhu and K. K. Ma, “A new diamond search algorithm for fast block matching motion estimation,” *IEEE Transactions Image Processing*, vol. 9, no. 2, pp. 287–290, Feb. 2000.

[7] D. Sheckler, Y. Ozturk and H. Abut, “Variable Size Block Motion Estimation,” Conference on Signals, Systems & Computers, vol.1, 1998, pp. 868 - 872

[8] G.R. Martin, R.A. Packwood and I. Rhee, “Variable Size Block Matching Motion Estimation with Minimal Error,” *W92. Department of Computer Science, University of Warwick*, October 1995.

[9] J. Zhang, M.O. Ahmad and M.N.S. Swamy, “Quad tree structured region- ise motion compensation for video compression,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 5, pp.808-822, Aug. 1999.

[10] S. Calzone, K. Chen, C. Chuang, A. Divakaran, S. Dube, L. Hurd, J. Kari, G. Liang, F. Lin, J. Muller and H.K. Rising, "Video compression by mean-corrected motion compensation of partial quad trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 86-96, Feb. 1997.

[11] I.Rhee, G. R. Martin, S. Muthukrishnan and R. A. Packwood, “Quad tree-Structured Variable-Size Block-Matching Motion Estimation with Minimal Error,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 1, pp. 42-50, Feb. 2000.

[12] Humaira Niasr, Tae-Sun Chol, “An Adaptive Block Motion Estimation Algorithm Based on Spatio Temporal Correlation”, *Digest of Technical papers, International conference on consumer Electronics*, Jan 7-11, 2006, pp.393-394.

[13] Viet-Anh Nguyen and Yap-peng Tan, “Efficient Block Matching Motion Estimation Based on Integral Frame Attributes”, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, pp. 375-385, March 2006.

[14] Jiancong Luo, Ishfog Ahmed, Yong Fang Liang and Vishwanathan Swaminathan, “Motion Estimation for Content adaptive Video Compression,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no.7, pp. 900-909, July 2008.

[15] Chun-Man Mak, Chi keung Fong and Wai Khen Chan, “Fast Motion Estimation For H.264/AVC in Walsh Hadmard Domain,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no.8, pp. 735–745, June 2008.

[16] Shen Li., Weipu Xu, Nanning Zheng, Hui Wang, “A Novel Fast Motion Estimation Method Based on Genetic Algorithm,” *ACTA ELECTRONICA SINICA*, vol.28, no.6, pp.114-117, June 2000.

[17] J. Kennedy and R. Eberhart, “Particle Swarm Optimization,” *IEEE International Conference on Neural Networks*, Perth, Australia. vol. 4, Dec 1995, pp. 1942-1948.

[18] Y.H. Shi and R.C.Eberhart, “Parameter Selection in Particle Swarm Optimization,” *Annual Conference on Evolutionary Computation*, 1999, pp.101-106.

[19] R.C. Eberhart and Y. Shi, “Comparison between genetic algorithm and particle swarm optimization,” *IEEE International Conference Computt.* Anchorage, AK, May 1998, pp. 611-616.

[20] J. F. Schutte, J.A. Reinbolt, B.J. Fregly, R.T. Haftka and A.D. George, “Parallel Global Optimization with the

Particle Swarm Optimization," *International Journal for Numerical Methods in Engineering*, pp. 2296-2315, 2004.

[21] Srinivas Pasupuleti and Roberto Bhattiti, "The Gregarious Particle Swarm Optimizer (G-PSO)," *GECCO 2006*, Seattle, Washington, USA, July 8-12, 2006, pp. 92-98.

[22] Ratnaweera, S. Halgamuge and H. Watson, "Self Organizing Hierarchical Particle Swarm Optimization with time varying acceleration coefficients," *IEEE Transactions on Evolutionary Computation*, vol. 8, pp. 240-255, 2004.

[23] M. Senthil Arumugam, M. V. C. Rao and Aarthi Chandramohan, "A new and improved version of particle swarm optimization algorithm with global-local best parameters," *Journal of Knowledge and Information System (KAIS), Springer*, vol.16, no.3, pp.324-350, 2008.

[24] M. Senthil Arumugam, M. V. C. Rao and Alan W.C. Tan, "A new novel and effective particle swarm optimization like algorithm with extrapolation technique," *International Journal of Applied Soft Computing, Elsevier*, vol. 9, pp. 308-320, 2009

[25] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control System Magazine,* vol. 22, Issue 3, pp. 52-67, June 2002

[26] D. S. Weile and E. Michielssen, "Genetic algorithm optimization applied to electromagnetic: A review," *IEEE Trans. Antennas Propagation*, vol. 45, pp. 343-353, Mar 1997.

[27] Ryosuke Kubota, Takeshi Yamakawa and Keiichi Horio, "Reproduction Strategy Based on Self-Organizing Map for Real coded Genetic Algorithm," *Neural Information Processing- Letters and Reviews*, vol. 5, no. 2, pp. 27-32, Nov 2004.

[28] P. Jog, J. Y. Suh and D. Van Gucht, "Parallel genetic algorithms applied to the traveling salesman problem," *SIAM Journal of Optimization*, vol. 1, pp. 515-529, 1991.

[29] D. S. Weile and E. Michielssen, "Genetic algorithm optimization applied to electromagnetic: A review," *IEEE Trans. Antennas Propagation*, vol. 45, pp. 343-353, Mar 1997.

[30] Dan Simon, "Biogeography-Based Optimization," *IEEE Trans. on Evol. Computer.* vol. 12, no.6, pp. 712-713, 2008.

[31] Swagatam Das and Sudeshna Sil, "Kernel-induced fuzzy clustering of image pixels with an improved differential evolution algorithm," *Information Sciences, Elsevier,* vol. 180, pp. 1237–1256, 2010.

[32] Y. Wang, B. Li and T. Weise, "Estimation of Distribution and Differential Evolution Cooperation for Large Scale Economic Load Dispatch Optimization of Power Systems," *Information Sciences*, *Elsevier,* 2010

[33] C. Twomey T. Stützle, M. Dorigo, M. Manfrin and M. Birattari, "An analysis of communication policies for homogeneous multi-colony ACO algorithms," *Information Sciences, Elsevier,* 2010.

VII .AUTHORS

Pooja received diploma in Computer Engineering in 2004 from Technical Board, Chandigarh. B. Tech degree in Computer Science and Engineering in 2007 under Punjab Technical University,Jalandhar.Since 2008 she is pursuing her M.Tech from Yadavindra College of Engineering, Guru Khashi Campus,Talwandi Sabo,Bathinda(Punjab) Her interests include, Image Processing,Optimization and Swarm Intelligence .She has contributed near about 8 technical papers in various national and international conferences. She is a life member of ISTE.

Seema completed M.Tech. in Computer Engineering from punjabi University Patiala in year 2007and B.Tech. in Computer Science & Engineering from Punjab Technical University, Jalandhar in year 2004 and holds NINTH merit position in University. She is presently working as Assistant Professor in Computer Engineering at Yadavindra College of Engineering, Punjabi University Guru Kashi Campus, Talwandi Sabo (Bathinda). Her research areas include Genetic Algorithms, Optimization, Image processing, motion estimation, Encryption etc.