Video Compression by Memetic Algorithm

Pooja Nagpal
M.Tech Student,CE deptt.
Yadavindra College of Engineering
Guru Khashi Campus,Talwandi Sabo.

Seema Baghla
Assistant Professor, CE deptt.
Yadavindra College of Engineering
Guru Khashi Campus, Talwandi Sabo

***Abstract*—Memetic Algorithm by hybridization of Standard Particle Swarm Optimization and Global Local Best Particle Swarm Optimization is proposed in this paper. This technique is used to reduce number of computations of video compression by maintaining same or better quality of video. In the proposed technique, the position equation of Standard Particle Swarm Optimization is modified and used as step size equation to find best matching block in current frame. To achieve adaptive step size, time varying inertia weight is used instead of constant inertia weight for getting true motion vector dynamically. The time varying inertia weight is based up on previous motion vectors. The step size equation is used to predict best matching macro block in the reference frame with respect to macro block in the current frame for which motion vector is found. The result of proposed technique is compared with existing block matching algorithms. The performance of Memetic Algorithm is good as compared to existing algorithms in terms number of computations and accuracy.**

Keywords- Memetic Algorithm (MA); Standard Particle Swarm Optimization (PSO); Global Local Best Particle Swarm Optimization (GLBest PSO); Video Compression; Motion Vectors; Number of Computations; Peak Signal to Noise ratio (PSNR).

I. Introduction

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.

Motion estimation has been popularly used in video signal processing, which is a fundamental component of video compression. In motion estimation, computational complexity varies from 70 percent to 90 percent for all video compression. The exhaustive search (ES) or full search algorithm gives the highest peak signal to noise ratio amongst any block-matching algorithm but requires more computational time [1]. To reduce the computational time of exhaustive search method, many other methods are proposed i.e. Simple and Efficient Search (SES)[1], Three Step Search (TSS)[2], New Three Step Search (NTSS)[2], Four step Search (4SS)[3], Diamond Search (DS)[4], Adaptive Road Pattern Search (ARPS)[5], Novel Cross Diamond search [6], New Cross-Diamond Search algorithm [7], Adaptive Block Matching Algorithm [8], Efficient Block Matching Motion Estimation [9], Content Adaptive Video Compression [10] and Fast motion estimation algorithm [11] . Soft computing tool such as Genetic Algorithm (GA) has also been used for fast motion estimation [12].

Traditional fast block matching algorithms are easily trapped 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. In this paper, we propose a Memetic Algorithm to reduce number of computations of video compression by maintaining same or better quality of video. The paper is divided into five sections. The review of Particle Swarm Optimization techniques is discussed in section 2. The proposed Memetic Algorithm for video compression is discussed in section 3. Section 4 provides experimental results comparing MA with other methods for video compression. The conclusion is given in section 5.

II. Particle Swarm Optimization

The Particle Swarm Optimizer (PSO) is a population-based optimization method developed by Eberhart and Kennedy in 1995[13]. PSO is inspired by social behavior of bird flocking or fish schooling. In PSO, a particle is defined as a moving point in hyperspace. It follows the optimization process by means of local best (*Lbest*), global best (*Gbest*), particle displacement or position and particle velocity. In PSO, particle changes their positions by flying around in a multi-dimensional search space until computational limitations are exceeded. The two updating fundamental equations in a PSO are velocity and position equations. The particle velocity is expressed as Eq (1) and the particle position is expressed as Eq. (2)

$$V_{K+1}^{i} = W * V_{k}^{i} + C_1 r_1 (Lbest - S_{k}^{i}) + C_2 r_2 (Gbest - S_{k}^{i}) \qquad (1)$$

$$S_{k+1}^{i} = S_{k}^{i} + V_{k+1}^{i} \qquad (2)$$

Where, *V*= Particle Velocity
S= Particle Position
Lbest = Local best
Gbest = Global best
W = Inertia weight
C_1 and C_2 are acceleration constant
r_1 and r_2 are random values [0 1]
k = Current iteration

i = Particle number

In first parts, *W* plays the role of balancing the global search and local search. Second and third parts contribute to the change of the velocity. The second part of Eq. (1) is the ‘cognition’ part, which represents the personal thinking of the particle itself. The third part of Eq (1) is ‘social part’, which represents the collaboration among the particles. Without the first part of Eq. (1), all the particles will tend to move toward the same position. By adding the first part, the particle has a tendency to expand the search space, that is, they have ability to explore new area. Therefore, they acquire a global search capability by adding the first parts.

In GLBestPSO [14], inertia weight (*w*) and acceleration co-efficient (*c*) are proposed in terms of global best and local best position of the particles as given in Eq. (3) and (4). The modified velocity equation for the GLBest PSO is given in Eq. (5).

GLBest PSO is given in Eq. (5).

$$w_i = \left(1.1 - \frac{gbest_i}{pbest_i}\right) \quad (3)$$

$$c_i = \left(1 + \frac{gbest_i}{pbest_i}\right) \quad (4)$$

$$v_i(t) = w * v_i(t-1) + c_i * r(t) * (pbest_i + gbest_i - 2x_i(t)) \quad (5)$$

III. Memetic Algorithm For Video Compression

Memetic Algorithm is developed by hybridization of Standard Particle Swarm Optimization and Global Local Best Particle Swarm Optimization. This technique is used to reduce number of computations of video compression by maintaining the same or better quality of video.

We have modified velocity and position equations of PSO to achieve step size for video compression, which is used to predict best matching macro block in the reference frame with respect to macro block in the current frame for which motion vector is found.

To get the step size, the velocity and position equations of PSO are modified as given below. The velocity equation is expressed as Eq. (6).

$$V(t) = W*C*r \quad (6)$$

Where *W* is the inertia weight, *C* is the acceleration constant, *r* is random number between 0 to 1 and *t* is generation number. To get the adaptive step size, the time varying inertia weight (*W*) is used instead of constant inertia weight similar to GLBestPSO for getting the true motion vector dynamically. The time varying inertia weight is based up on previous motion vectors as given in Eq. (7)

$$W = (1.1 - Gbest + Pbest) \quad (7)$$

$$Gbest = X + Y$$
$$Pbest = X - Y$$

Where, *X* and *Y* is the x and y coordinates of the predicted motion vector. The velocity term in Eq. (6) is added with previous motion vector to predict the next best matching block as given in Eq (8)

$$S(t+1) = S(t) + V(t) \quad (8)$$

In Memetic Algorithm, a search is made in an earlier frame of the sequence over a random area of the frame. The search is for the best matching block viz. the position that minimizes a distortion measured between the two sets of pixels comprising the blocks. The relative displacement between the two blocks is taken to be the motion vector. Usually the macro block is taken as a square of side consists of 16 pixels. The compression ration is 128:1 or 256:2. The each block size of 16 x 16 is compressed into two pixels which are nothing but motion vectors.

In Memetic Algorithm, five swarms are used to find best matching block. The initial position of block to be searched in reference frame is the predicted motion vector as expressed in Eq. (8). In Memetic Algorithm, the number of generations is taken as 2. The cost required for finding best matching block in the reference frame is ten blocks, which is less than existing methods.

The mean absolute difference (MAD) is taken as objective function or cost function in Memetic Algorithm and is expressed as in Eq. (9).

$$MAD = \frac{1}{MN}[\sum_{P=1}^{M}\sum_{Q=1}^{N}|CurrentBlock(P,Q) - \mathrm{Re}\,ferenceBlock(P,Q)|] \quad (9)$$

Where, *M* = Number of rows in the frame and N = Number of columns in the frame. The objective quality obtained by Memetic Algorithm has been measured by the peak signal-to-noise ratio (PSNR), which is commonly used in the objective quality comparison. The performance of the proposed method is evaluated by following Eq (10)

$$PSNR = 10\log_{10}\frac{255^2}{\frac{1}{MN}\sum_{P,Q=1}^{M,N}(OrigionalFrame(P,Q) - CompensatedFrame(P,Q))^2} \quad (10)$$

A further small improvement in the Memetic Algorithm is to check for Zero Motion Prejudgment (ZMP). If current macroblock matches with macroblock in the reference frame i.e. cost is zero then motion vector are directly stored as zero motion vector instead of gaining the motion vector through Memetic Algorithm. The zero motion prejudgment saves considerable amount of computational time.

Zero Motion Prejudgment is the procedure to find the static macro blocks which contains zero motion. In real world video sequences more than 70% of the MBs are static which do not need the remaining search. So, significant reduction of computation is possible if we predict the static macro blocks by ZMP procedure before starting motion estimation procedure and the remaining search will be faster and saves memory. We first calculate the matching error (MAD) between the macro block in the current frame and the macro block at the same location in the reference frame and then compare it to a predetermined threshold. If the matching error is smaller than predetermined threshold we consider this macro block static which do not need any further motion estimation, and return a [0, 0] as its motion vector (MV).

IV. Experimental Results And Discussions

The presented method Memetic Algorithm has been tested for two videos sequences. The performance of the proposed method is measured in terms of the *PSNR* and percentage of saving number of computations of video compression. To test the efficiency of the proposed algorithm with existing methods, the algorithms are executed in single machine. Video sequence with distance of two frames between current frame and reference frame are used to generate frame-by-frame results of the Memetic Algorithm. The performance of Memetic Algorithm is compared with other existing methods such as ES, SESTSS, TSS, NTSS and 4SS and the results are presented in Table 1 and Table 2. Figure 1 to Figure 2 shows the comparison of *PSNR* with other existing methods for two video sequences respectively. Figure 3 to Figure 4 shows the comparison of number of computations with other existing methods for two video sequences respectively. The speed of Memetic Algorithm is found to be faster than that of already published methods and *PSNR* is close to published methods as shown in Table 3 and Table 4.

The Memetic Algorithm saves number of computations up to 95.36% to 94.95 % with PSNR degradation of -0.1222 to -0.0537 as compared to ES. Similarly, Memetic Algorithm saves number of computations up to 58.95% to 35.73% with PSNR gain of +1.08 to +0.03as compared to SESTSS, TSS, NTSS and 4SS.

TABLE I. Comparison of psnr

Sequence	ES	SESTSS	TSS	NTSS	4SS	Memetic
Video 1	32.3762	31.1686	31.4379	32.2046	31.8304	32.2540
Video 2	37.2776	37.1786	37.2081	37.1939	37.1480	37.2239

TABLE II. Comparison of number of computations

Sequence	ES	SESTSS	TSS	NTSS	4SS	Memetic
Video 1	207.4109	23.6349	16.2827	19.7889	18.7977	10.4642
Video 2	209.8687	23.7264	16.8776	18.7248	17.3848	9.7377

TABLE III. PSNR Gain By Memetic Algorithm Over Existing Methods

Sequence	ES	SESTSS	TSS	NTSS	4SS
Video 1	**-0.1222**	1.0854	0.8161	**0.0494**	0.4236
Video 2	**-0.0537**	0.0453	0.0158	**0.03**	0.0759

TABLE IV. Number Of Computations Saving By Memetic Algorithm Over Existing Methods

Sequence	ES	SESTSS	TSS	NTSS	4SS
Video 1	**94.95485**	55.72564	**35.73425**	47.12086	44.33255
Video 2	**95.3601**	58.95838	**42.304**	47.99571	43.98728

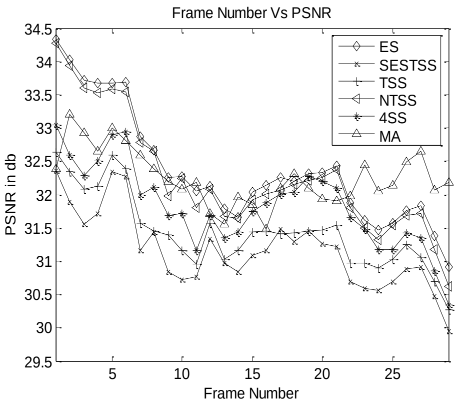

Figure1. Comparison of PSNR for Video 1

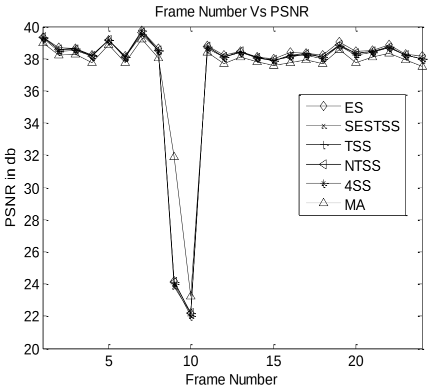

Figure2. Comparison of PSNR for Video 2

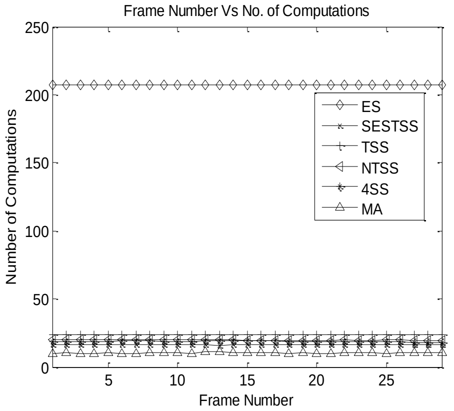

Figure3. Comparison for Number of Computations for Video 1

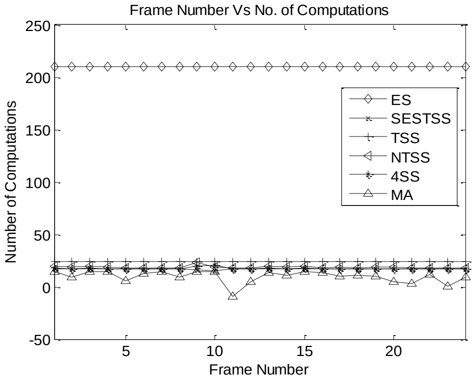

Figure 4 Comparison for Number of Computations for Video 2

V. Conclusion

The soft computing tool such as Memetic Algorithm is used for video compression. The performance of Memetic Algorithm is good as compared to existing algorithms except ES for video compression because it find best matching block with less computational cost by maintaining same accuracy of video. Memetic Algorithm is faster and accurate tool for video compression. The results shows promising improvement in terms of accuracy, while drastically reducing the number of computations. The step size equation of MA predicts the best matching block with less computational requirement. Memetic Algorithm uses only two-steps for video compression. An approximately more than 96% of computational time saving in the motion estimation coding is achieved as compared to ES algorithm. This saving comes with less degradation in the PSNR. This technique can easily used as block matching motion estimation algorithm or video compression algorithm.

References

[1] 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

[2] 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.

[3] 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

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

[5] 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, July 2008, pp.900-909.

[6] 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. 18, no.6, June 2008, pp. 735-745.

[7] 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

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

[9] 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.

[10] Viet-Anh Nguyen, Yap-peng Tan, "Efficient Block Matching Motion Estimation Based on Integral Frame Attributes", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.16, no. 2, March 2006, pp.375-385.

[11] 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.

[12] C.W. Lam., L.M. Po and C.H. Cheung, "A New Cross- Diamond Search Algorithm foe Fast Block Matching Motion Estimation", *2003 IEEE International Conference on Neural Networks and Signal Processing*, Nanjing, China, Dec. 2003, pp.1262-1265.

[13] Yao Nie, Kai-Khuang Ma, "Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation", *IEEE Trans. Image Processing*, vol.11, no.12,December 2002, pp. 1442-1448.

[14] Chun-Ho Cheung, Lai-Man Po, "A Novel Cross Diamond Search Algorithm for Fast Block Estimation", *IEEE Trans. Circuits and Systems*, vol.11, no.12, December 2002, pp. 1442-1448.

[15] Shan Zhu, Kai-Khuang Ma, "A New Diamond Search Algorithm for Fast Matching Motion Estimation", *IEEE Trans. Image Processing*, vol.9, no.2, February 2000, pp.287-290.

[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, June 2000, pp.114-117.

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

[18] Lai-Man Po., Wing –Chung Ma, "A Novel Four- Step Search Algorithm for Fast Block Motion Estimation", *IEEE Trans. Circuits and Systems for Video Technology*, vol.6, no. 3, June 1996, pp. 313-317 .

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

[20] Renxiang Li, Bing Zeng, and Ming L. Liou, "A New Three- Step Search Algorithm for Block Motion Estimation", *IEEE Trans. Circuits and Systems for Video Technology*, vol. 4, no. 4, August 1994, pp. 438-442.

AUTHORS PROFILE

Pooja Nagpal 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 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.