FOUR-STEP GENETIC SEARCH FOR BLOCK MOTION ESTIMATION

Man F. So and Angus Wu

Electronic Design Automation Center, Department of Electronic Engineering
City University of Hong Kong, Hong Kong, PRC
E-mail: {mfso,eeawueed}@ee.cityu.edu.hk

ABSTRACT

Genetic Algorithms (GA) is well known for searching global maxima and minima[1, 2]. In general, number of search points required by GA for searching global extreme is much lower than the exhausted search. GA has been applied to Block Matching Algorithm (BMA) and demonstrates positively its capability in BMA. The mean square error (MSE) performance of GA based BMA is close to full search (FS). However, the disadvantage of GA is the computational requirement for practical use. A four-step genetic search algorithm is proposed to BMA. The proposed method takes advantage of GA and 4SS. The simulation result shows the proposed method has similar performance to full-search (FS) in terms of MSE. In addition, number of search points required by the proposed algorithm is approximately equal to 14% of FS and closed to three-step search (3SS). The speed up ratio between proposed algorithm and FS is 5.6 times.

1. INTRODUCTION

The block matching algorithm (BMA) is a powerful algorithm for image sequence compression. It has been widely used for different video compression standards, such as H.261[3], MPEG-1[4] and MPEG-2[5]. The commonly used distortion measurement is mean absolute error (MAE). The pixel intensity $F_t(m,n)$ represents the co-ordinate (m,n) of the upper left pixel of the current block. Similarly, the pixel intensity $F_{t-1}(m,n)$ represents the co-ordinate (m,n) of the upper left pixel of the previous block. We assume the block of MxM pixels and the maximum displacement offset is l pixels. The MAE between the block (m,n) of current frame and block (m+u,n+v) of previous frame can be express by the following equation:

$$MAE_{m,n}(u,v) = \frac{1}{M^2}\sum_{i=0}^{M-1}\sum_{j=0}^{M-1}|F_t(i+m,j+n) - F_{t-1}(i+m+u,j+n+v)|$$

$$-l \leq u,v \leq l$$

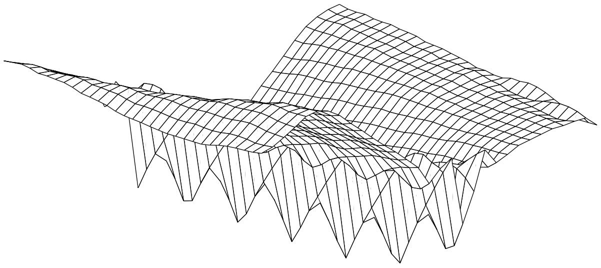

Figure 1: Typical plot of MAE value within search area

The motion vector (u,v) is determined by the minimum MAE of the same block (m,n). Full-search (FS) searches all possible candidate block within the search area to find the block with minimum MAE. The total search points for FS is $(2l+1)^2$. If the maximum displacement l is 7, then the total search number is 255. The computational requirement of FS is too high for real time application. As a result, there are several fast algorithms have been proposed for BMA, for instance, three-step search (3SS)[6], new three-step search (N3SS)[7] and four-step search (4SS)[8]. These fast algorithms use uniform search pattern to reduce the search points. They assume distortion measurement, such as MAE, decreases monotonically as the search point get closer to the global minimum. For the maximum displacement l is 7, the maximum search points for 3SS is 25, N3SS is 33 and 4SS is 27. Although the computational cost of these algorithms are lower than FS, these fast algorithms in general performs worse than FS. As shown in figure 1, there are several minimum points within the search area. The assumption of these fast algorithms does not hold. In Chow and Liou's work, it points out that these algorithms may be trapped by number of local minimum[9]. These fast algorithms find the motion vector with local minimum MAE instead of global minimum MAE. Since the distribution of distortion measurement is not uniform, FS is still needed for high quality image compression. Alternatively, genetic algorithm (GA) is a well known algorithm for locating global maxima or minima in any arbitrary search space. Moreover, the computational cost of GA is lower than exhausted search. Recently, GA has been applied to BMA[9, 10] with encouraging result, in particularly, lightweight genetic search algorithm (LGSA) [10]. However, GA requires high computational overheads. GA based BMA suffers from a high computational cost. A four-step genetic search algorithm is proposed to BMA. The proposed method takes advantage of GA and 4SS.

2. FOUR-STEP GENETIC SEARCH FOR BLOCK MATCHING ALGORITHM

The proposed four-step genetic search for BMA combines GA and 4SS. Figure 2 shows the block diagram of the proposed algorithm. The following section discusses the outline and advantages of GA, 4SS and the proposed algorithm. Although exhausted search can find global maximum and minimum in any arbitrary search space (S), the efficiency of exhausted search in terms of computational requirement may be too low for practical use. Moreover, some search space are too large to search every element. In general, GA can search individual points in any arbitrary search space (S) with much less search steps. These individual points are represented by binary string. They are named as chromosome (C). In each generation, there will be a number of chromosomes (N). Each chromosome will be evaluated by a fitness function (f). A chromosome with higher fitness value means it is strong. The probability for that chromosome survival will be high. On the other hand, a chromosome with lower fitness value means the probability of that chromosome will survive for next generation is low. It will be replaced by a strong chromosome. Finally, chromosomes in each generation will become better. A set of genetic operator, namely crossover and mutation operators, will be used for generating new chromosomes . After several generations, the chromosome with the highest fitness value will be the best chromosome to represent the extreme within the search space. In addition, the search will stop if any chromosome's fitness value meet the stopping criteria.

The search area of the first three search steps of 4SS is a 5×5 square. In each search step, nine search points will be used. The search point with minimum MAE will be next step's search center. If the center of the search area with minimum MAE, the search step will be directly set to the forth step. Since part of current step's search points overlap previous step's search points, some searching points can be skipped[8]. The search area for the forth step of 4SS is a 3×3 square. Number of search points of the forth step is the same as previous step. Furthermore, the block movement of image sequence is small and gentle[7]. The performance of 4SS is better than 3SS and N3SS in terms of computational efficiency and MAE. Since 4SS uses smaller search area than 3SS and N3SS, 4SS can locate small movement more accurately. In addition, half-way stop technique is being used in 4SS; it only needs small number of search points to locate small movement motion vector.

Assume the maximum motion offset l is 7 pixels. Total generations of proposed algorithm is four. In the first three generations, the mutation step size is 2. In the forth generation, the mutation step size is 1. The mutation step size of four-step genetic search is the the same as the search step size of 4SS. It covers all possible motion vector within 7 pixels. Because of the distribution of distortion measurement is too complex to model and the timing constraint, GA will be chosen as a searching tool. Moreover, the nature of motion vector distribution is strong biased on small movement. Not only more search points is selected in the central region, but also the mutation step is small. Small search step can locate small movement block more accurate. Since part of current step's search points overlap previous step's search points, some searching points can be omitted. Hence, the motion vector can be computed faster. The proposed algorithm takes advantages of GA and 4SS. The new algorithm can search motion vector faster with better quality. The search will stop after four generations or the MAE of any chromosome is equal to zero.

In this application, the search space S consists set of 2-D motion vectors. X_i and Y_i represent x and y component of motion vector respectively. In each generation, the ith chromosome C_i is presented in the following manner:

$$C_i = \{\ X_i,\ Y_i\ \}$$
$$X_i = (x_{i,k-1},\ \cdots,\ x_{i,0}) \text{ and } Y_i = (y_{i,k-1},\ \cdots,\ y_{i,0})$$

where the chromosome $X_{i,n}$, $Y_{i,n}$ are decimal coded chromosomes and k is the length of the chromosome

Each chromosome C represents a possible motion vector. The following are summaries the proposed four-step genetic search for Block Matching Algorithm:

1. Initial Population – Due to the fact that most of the block motion is very gentle and small[7], the initial population of N chromosomes (search points) is selected nearly the central region randomly. Each chromosome represents a motion vector. The length of chromosome is mainly depended on the maximum displacement of block. If the maximum displacement is l pixels then the length of chromosome is $\lceil log_2 2l \rceil$.

2. Evaluation – The fitness value of each chromosome is evaluated. There are n chromosomes will be selected from N chromosomes for reproductions. The fitness function is defined as:

   $$\mathrm{f}_i(\mathrm{C}_i) = \begin{cases} MAE(\mathrm{C}_i) & , \quad MAE(\mathrm{C}_i) \le MAE(C_n) \\ MAE(\mathrm{C}_n) & , \quad MAE(\mathrm{C}_i) \ge MAE(C_n) \end{cases}$$

   where C_n is the nth minimum MAE of N chromosomes and $0 \le i \le N$ - 1

   Since the fitness value is defined by chromosome's MAE, the chromosome with smaller fitness value means it is better chromosome to represent the true motion vector.

3. Reproduction – In each generation, there are n chromosomes will be selected for reproduction. There are totally n discrete interval for reproduction decision criteria. The range of each interval r_i is calculated by the following expression:

   $$\mathrm{r}_i = \left[\ \sum_{k=0}^{i-1} \mathrm{f}_k, \sum_{k=0}^{i} \mathrm{f}_k\ \right)$$

   where $[\ and\ )$ detones closing and opening boundaries, f_k is the kth chromosomes, $0 \le$ i $\le$ n-1

   Reproduction index t_i is generated randomly, $t_i\ \epsilon\ [\ 0\ , \sum_{k=0}^{n-1} \mathrm{f}_k\ )$. The chromosome C_j will be selected for reproduction if t_i falls in the interval r_j, that is, $t_i \epsilon r_j$. N chromosomes will be reproduced from these n chromosomes. The new reproduced chromosomes will be transferred to the mutation pool. This method is similar to weighted roulette wheel method[1].

4. Mutation – The mutation pool consists of N new reproduced chromosomes. Each chromosome C will be mutated by a pair of mutation operators. They are performed on X and Y component of chromosome C. The size of mutation step depends on the current generation. For the first three generation, *size* is 2. The size of forth *step* is 1.There are eight different pairs of mutation operators. p is a operator identity number between 0 and 7. p is determined by the different chromosome number n. The mathematical relationship between p and n is $p = n \; mod \; 8$. The mutation operation and operator pairs are defined as:

$$x_{mut} = (-1)^i * \lceil \tfrac{j}{4} \rceil \;\; , \;\; y_{mut} = (-1)^k * \lceil \tfrac{l}{4} \rceil$$
$$i = (p+2) \; div \; 4 \;\; , \;\; j = (p+2) \; mod \; 4$$
$$k = p \; div \; 4 \;\; , \;\; l = p \; mod \; 4$$
$$X_n^{'} = X_n + x_{mut} * size \; , \; Y_n^{'} = Y_n + y_{mut} * size$$

5. Selection – The motion vector is the chromosome with the lowest fitness value within the population. In each generation, N chromosomes are selected within $2N$ chromosomes. Part of the chromosomes with smaller fitness value will be selected as members of next generation. The weaker chromosomes will be eliminated.

6. We repeat step 2 - 5 for 4 generations. The chromosome with the minimum fitness value is the motion vector of the block.

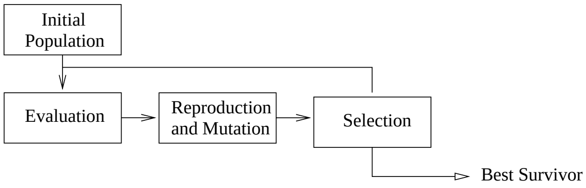

Figure 2: Diagram of the proposed Four Step Genetic Search

3. SIMULATION RESULT

The proposed algorithm (4GS), FS, 3SS, 4SS and LGSA are simulated. The simulation results are summarized in Table 1, 2 and 3. Figure 3 and 4 show the MSE comparison of Table Tennis and Football sequence. In the simulation, the block size is 16×16 pixels, and the maximum motion offset is 7 pixels. These algorithms are simulated on a Intel 586 PC. Different types of image sequence have been tested. The length and dimension of each image sequence are summarized in Table 4. They are Football, Garden, Miss American, Mobile, and Table Tennis. Only luminance components will be used for testing. The pixel are uniformly quantized into 8 bits. Table 1 shows the number of search points for each sequence. Both 4GS and LGSA need more search points than 3SS and 4SS. They use 2 to 44 % more search points than 3SS. The new method needs less number of search points than LGSA. It only takes 2 to 27 % more search points than the conventional 3SS. On the other hand, LGSA takes 28 to 44 % more search points than 3SS. In general, the computational cost of 4GS is lower than LGSA. In addition, 4GS takes 39% to 44% more search points than 4SS. However, the search number of the proposed algorithm is much less than FS. The proposed algorithm only takes approximate 14 % search points compared to FS.

Table 2 shows search time of different algorithms. LGSA and 4GA take longer search time than 3SS to compute motion vector. Amount of search time is increased by 25 % to 60%. On the other hand, the computation time of LGSA and 4GA are much less than FS. They save approximate 80% to 85% of computation time. Moreover, the search time of 4GS is shorter than LGSA by 6.7% to 13.4%.

Table 3 shows the performance (in terms of mean square error) of different algorithms. In general, 4GS and LGSA perform better than 3SS and 4SS. However, the performance of the genetic algorithm based BMA is depending on nature of image sequence. LGSA performs better with Football, Mobile, and Table Tennis sequence. The proposed algorithm performs better with Garden, Miss American and Saleman sequence. The average mean square error(MSE) of estimated sequence by LGSA and 4GS are very close to each other. Nevertheless, the proposed 4GS and LGSA both perform comparatively to FS. Figure 4 and 5 shows the MSE of 4GS and LGSA are close to FS.

4. CONCLUSION

In conclusion, a four-step GA-based block matching algorithm is proposed. Because of the nature of GA, the proposed algorithm can avoid the trap of local minimum. The motion vector with global minimum MAE can be found with less search points. Number of search points required by the proposed algorithm is approximately equal to 14% of FS. From the experimental result, the new method performs better than 3SS and 4SS whilst the computational requirement is less than LGSA and much efficient than FS. The proposed algorithm is much faster than FS. The speed up ratio is from 5.6 to 6 times. The new algorithm can search motion vector with excellent quality and performs faster. As a result, the proposed algorithm is suitable for H.261, MPEG-1 and MPEG-2. Since the new algorithm with regularity, it is also suitable for hardware implementation.

5. REFERENCES

[1] J. H. Holland. "Adaption in Natural and Artifical Systems". University Michigan Press, Ann Arbor, 1975.

[2] Zbigniew Michalewicz. "Genetic Algorithms + Data Structures = Evolution Programs". Springer-Verlag, 2nd, extended edition, 1994.

[3] CCITT SGXV. "Description of reference model 8 (RM8)". Specialists Group on Coding for Visual Tele-

phony, Document 525, Working Party XV/4, Specialists Group on Coding for Visual Telephony, June 1989.

[4] ISO/IEC 11172-2 (MPEG-1 Video). "Information technology – Coding of moving pictures and associated audio for digitial stroage media at up about 1.5 Mbit/s: Video". 1993.

[5] ISO/IEC 13818-2 — ITU-T H.262 (MPEG-2 Video). "Information technology – Generic coding of moving pictures and assoicated audio information: Video". 1995.

[6] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro. "Motion compression Interframe coding for video conferenceing". In in Proc. NTC 81, pages C9.6.1–9.6.5, New Orleans, LA,, Nov/Dec 1981.

[7] R. Li, B. Zeng, and M.L. Liou. "A new three-step search algorithm for block motion estimation". IEEE Transactions on Circuits and System For Video Technology, 4(4):438–442, Aug. 1994.

[8] Lai-Man Po and Wing-Chung Ma. "A Novel Four-Step Search Algorithm for Fast Block Motion Estimation". IEEE Transactions on Circuits and System for Video Technology, 6(3):313–317, June 1996.

[9] Keith Hung-Kei Chow and Ming L. Liou. "Genetic Motion Search Algorithm for Video Compression". IEEE Transactions on Circuits and System For Video Technology, 3(6):440–445, Dec 1993.

[10] Chun-Hung Lin and Ja-Ling Wu. "Genetic Block Matching Algorithm for Video Coding". Proceeding of Multmedia' 96, pages 544–547, 1996.

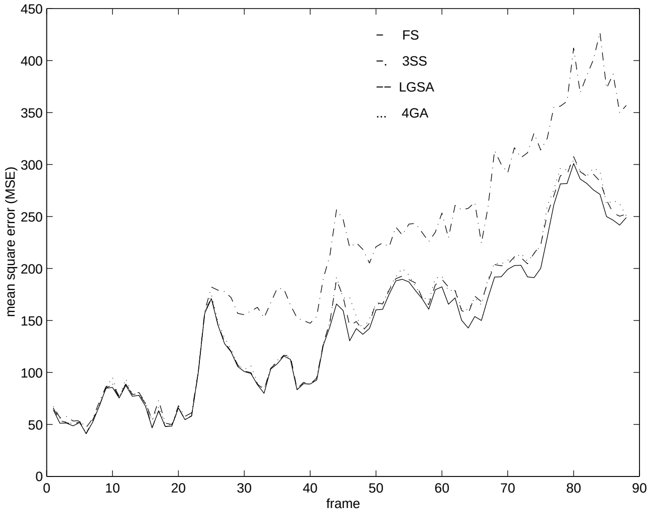

Figure 3: Comparison of Table tennis sequence

	FS	3SS	4SS	LGSA	4GS
Football	756.57	91.9	69.41	130.64	114.68
Garden	949.93	116.28	95.43	181.48	167.29
Missa	741.94	92.55	73.50	146.14	130.99
Mobile	1689.94	205.29	166.44	318.28	289.06
Saleman	913.94	110.41	79.07	158.43	137.14
Tennis	760.29	92.24	76.02	139.01	127.41

Table 1: Summary of Search time

	FS	3SS	4SS	LGSA	4GS
Football	68153	7744.6	5727.8	9912.4	7949.2
Garden	66676	7658.2	6179.4	10789	9261.2
Missa	51076	5962.7	4638.2	8572.6	7142.9
Mobile	347686	39121	31091	54855	46293
Saleman	82688	9348	6522.1	12011	9524.6
Tennis	68153	7771.2	6317.9	10598	8997.2

Table 2: Summary of average search number

	FS	3SS	4SS	LGSA	4GS
Football	177.67	199.31	193.96	180.16	181.92
Garden	361.84	425.31	397.09	365.92	365.46
Missa	9.98	11.15	10.95	10.08	10.06
Mobile	40.48	49.50	43.32	40.64	40.66
Saleman	19.77	20.66	20.41	19.98	19.88
Tennis	175.97	235.56	204.37	181.68	184.34

Table 3: Summary of average MSE

	Number of frames	Dimension
Football	89	352x240
Garden	114	352x240
Missa	116	256x256
Mobile	39	720x576
Saleman	89	352x288
Tennis	89	352x240

Table 4: Summary of sequence information

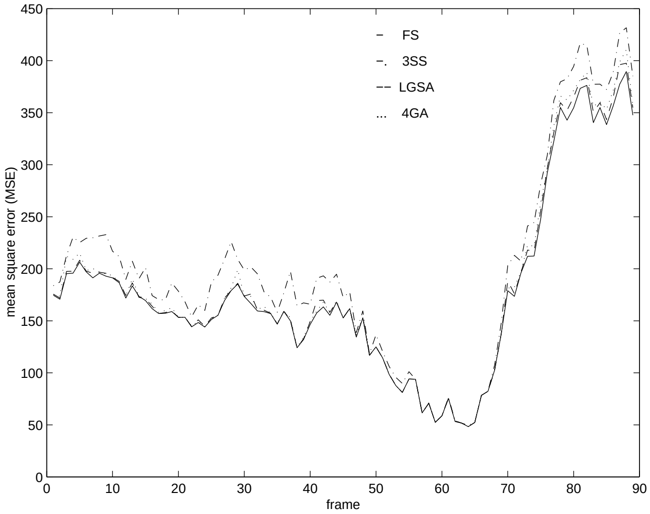

Figure 4: Comparison of Football sequence