A Novel Fast Motion Estimation Method Based on Genetic Algorithm

Shen Li, Wei-pu Xu, Hui Wang, Nan-ning Zheng
Institute of Artificial Intelligence and Robotics
Xian Jiaotong University, Xi'an, Shaanxi, 710049, P.R.China
Email: nnzheng@xjtu.edu.cn

Abstract
A novel fast motion-estimation method based on an improved genetic algorithm is presented, in which both objective search and random search derived from genetic mutation are used for searching the global optimum and a threshold selection operator is applied to speeding up the estimation. The selection of initial population based on the coherence between neighboring macroblocks also improves the performance of search. Experimental results demonstrate that this method has very similar performance to that of FS, but just slightly slower than SSS and 2DL. The inherent robustness and high parallelism enable it to be suitable for VLSI implementation of video encoders.

1. Introduction

Motion estimation is an essential component of many video encoding algorithms. The most popular method adopted to estimate the motion between frames is the block matching algorithm (BMA), in which a block of size M*N (usually 16*16) is compared with a corresponding block within a search area in the previous frame. Three main elements—match criterion, search area and search scheme—should be considered in the BMA. The match scheme is the most important, which plays a crucial part in the performance of BMA.

Essentially, the match scheme of BMA is an optimal problem. The full search BMA will always find the optimum motion vector by calculating the difference between every block in the search window from the previous frame and the current block. However, the price paid for this optimum performance is a high computational cost, which prevents it from being applied in most real-time systems. A number of fast search algorithms[4][5][6] have been proposed to greatly reduce the computational complexity by finding a sub-optimum motion. All these algorithms rely on an assumption that the difference measure decreases monotonically as the search position moves closer to the optimum position. Because this assumption usually does not hold, these fast algorithms may only find the local minimum and can not find the global minimum. Thus the quality of encoded video may become much worse.

The genetic algorithm[10][15] is an optimum-searching process based on the laws of natural selection and genetics. It adopts the coding set of parameters instead of the parameters themselves to operate and searches the optimum based on groups of points, not a single point. Moreover, it uses a random, instead of a definite rule to work on the searching process. All of these give it high robustness as well as parallelism, and enable it to be free of the limitation in continuity and single peak requirement. It is effective in solving the problems of searching global optimum, although the computational complexity of genetic algorithm is high.

In the paper, a novel fast motion-estimation method based on an improved genetic algorithm is provided. It not only gets rid of many fast BMAs' defects, but also much faster than common genetic algorithms. In Section II, the improved genetic motion-estimation algorithm is described. Experimental results as well as performance comparison with existing algorithm are presented in Section III.

2. Improved Genetic Fast BMA

To solve an actual problem with genetic algorithm, the parameters of the problem are coded firstly; and then a fitness function should be chosen for determining the winner. Later the initial population are selected and begin to evolve, that is, are processed by the selection operator, crossover operator, mutation operator and other genetic operators. The final winner is the result of the problem when the procedure is broken by the stopping rule. The improved genetic fast BMA is discussed in detail as follows:

1) Determine the coding scheme

The result space should be bi-directionally mapped
into a space of chromosomes. Since the motion vector is represented by \(MF(x, y)\), it is binary encoded into \((x_1, x_2, \ldots, x_n, \text{var}_1, \text{var}_2, \ldots, \text{var}_n)\) where \(x_i, y_i = 0\) or \(1\). \(l = 2n, n = \log_2 s_m + 1\), \(s_m, s_r\) represent the half width and the half height of the search window respectively. Here, \(x_1, y_1\) is specially used for denoting the sign of the vector. (i.e. \(x_i = 0\) denotes positive motion vector; \(x_i = 1\) denotes negative motion vector.)

2) **Define the fitness function**

The fitness function is defined:

\[
f(C_i) = \max\left\{ME(C_i^1), ME(C_i^2), \ldots, ME(C_i^n)\right\} - ME(C_i);
\]

where \(ME(C)\) represents the matching error of the chromosome \(C\). Hence the smaller the matching error, the larger the fitness value.

3) **Chose the genetic operator**

Common genetic algorithms usually choose selection operator, crossover operator and mutation operator for evolution. In this algorithm, only selection operator, crossover operator and a special optimization operator are used according to the character of motion vectors.

- **Selection operator**

  In a general selection process, a chromosome is selected based on a probabilistic scheme, which costs much in computation. To reduce the complexity, a simple "threshold-based" selection mechanism according to the mean fitness of all chromosomes is provided. The chromosomes, whose fitness is higher than the mean fitness, will be copied to the new population. Others will be first acted on by a mutation operator, then enter the new population. Thus the chromosome with a larger fitness, namely, the search point with a smaller matching error will have enough opportunity to survive.

- **Mutation operator**

  After being carefully studied, the genetic mutation is found to have an actual effect in selecting the next search point. If the significant genes of the chromosomes are mutated, the next search point will be far from the center point of the search window, whereas the next will be nearer to the center if the less significant genes is mutated. Owing to its outstanding performance, the mutation operator is well used in searching the global optimum while the crossover operator is removed. Because the horizontal vector \(x\) has nothing to do with the vertical vector \(y\), the mutation in the higher piece of the chromosome and the mutation in the lower will be performed individually, that is, at first two random numbers \(R_x\) and \(R_y\) are generated, then the corresponding \(x_{xR_x}, y_{yR_y}\) will be bitwise reversed and the new chromosome \((x_{xR_x}, x_{yR_y}, \ldots, x_n, y_{R_x}, \ldots, y_n)\) is obtained.

  - **Special optimization operator**

    In case that only the selection operator and the mutation operator above are acted on the population, some dominant chromosomes will be selected each time and have no chance to mutate. Consequently it’s possible that their nearby points, which may have smaller matching error than they, can’t be reached in the evolution. In order to avoid this situation, a special optimization operator is performed on the chromosome whose fitness is the largest, that is, the neighboring eight points of this chromosome are to be searched and the best of the nine points is selected for the next generation.

4) **Select the initial population and determine the population size**

In common genetic algorithms, the initial population are selected in random. Owing to their significant effects on the performance of the search, here they are selected according to a rule that they should be as near as possible to the global optimum. This rule is based on two facts:

- Most of the motion vectors are center-biased.
- It has been applied in many fast BMAs. The center point of the search window and its neighboring eight points can be chosen for parts of the initial population.

- Neighboring motion vectors in one frame are similar.

Movement of the whole picture (i.e. displacement of the camera) and large objects will produce similar motion vectors of adjacent macroblocks. This fact can be viewed from Figure 1, which depicts the motion vector diagram of part of the second frame from Bike sequence.

![Figure 1. Motion vector graph of part of the second frame from BIKE sequence](attachment:image)

According to Figure 2, the motion vector of the current macroblock can be predicted by former neighboring macroblocks, that is \(MV_{pred} = \left(\frac{MV_1 + MV_2 + MV_3}{3}\right)\), where \(MV_{pred}\) represents the predicted motion vector of the current macroblock; \(MV_i\) represents the motion vector of the \(i^{th}\) macroblock; "+" means vector-plus.

67
5) Determine the stopping rule

In this problem, the largest fitness of the chromosomes is available when the matching error is zero. At this time the evolution should be broken for the global optimum has been found. However, in most circumstance the matching error can’t reach zero, the evolution should be stopped under the limitation in the number of generations.

The improved genetic BMA can be summarized as follows.

a Select the initial population

\[ P_0 = \{ C_{i}^{0} \}_{i=0}^{17} \]

where \( C_{i}^{0} = (x_{1i}^{0}, x_{2i}^{0}, ..., x_{ni}^{0}, y_{1i}^{0}, y_{2i}^{0}, ..., y_{mi}^{0}) \);

b Calculate the fitness of each chromosome and the mean fitness of this generation.

c Copy or mutate the chromosome to the next generation according to the rule discussed above

d Stop the evolution if the stopping rule is satisfied.

Otherwise, go to step b, c.

e Convert the strongest chromosome to the search point, and get the result.

3. Experimental results and performance evaluation

Three typical standard sequences are tested separately using the improved genetic BMA and the FS, 3SS, and 2DL. The three sequences are Bike (contains mainly translation and rotation of objects), Garden (contains the global movement of camera, and have abundant details in the area of flowers) and Race (has complex movement). In the experiment, picture format is PAL, 30F/s. Each picture group has 12 frames, the distance between P-frames is 3, search range is selected \((\pm 30) \times (\pm 30)\) for P-frames, \((\pm 7) \times (\pm 7)\) for the first B-frames to search forward, \((\pm 15) \times (\pm 15)\) for the first B-frames to search backward, \((\pm 15) \times (\pm 15)\) for the second B-frames to search forward, \((\pm 7) \times (\pm 7)\) for the second B-frames to search backward. Mean absolute error function is taken as the matching criterion in all tests. In the improved genetic algorithm, population size \(N=18\), the number of the generations \(G=4\). The performance of each algorithm is evaluated by the subjective picture quality, overall signal-to-noise (SNR) between the original and decoded picture and the number of bits required to code the frames. Figure 4-6 shows the results of comparison between different algorithms.

The original and decoded frames of Garden sequence using different searching algorithms are shown as Figure 4. It can be seen that the one with improved genetic BMA outperform the one with 3SS, and is close to the one with FS in picture quality. The white blocks in the picture depicts the corresponding macroblocks are intra-coded instead of using the inter-frame prediction. The more white blocks inside the frame, the more bits are required to coding the frame. Figure 5-6 illustrate the overall SNR between the original and decoded frames and the bits-requirement for encoding the sequences using different algorithms. It is shown that the SNR using the improved genetic BMA is very close to that of FS, and much higher than 3SS and 2DL, and that the bit allocation with our method is more stable and reasonable than that of 3SS and 2DL. In addition, the computational load of our method is only slightly higher than 3SS and about one-fifth that of FS.

4. Conclusion

A novel fast motion-estimation method, the improved genetic BMA, is provided in this paper. It makes full use of the advantage of genetic algorithm to search the global optimum and get rid of its shortcoming of high complexity in computation. Furthermore, the fixed selection of initial population highly improves the performance. Experimental results demonstrate that the improved genetic BMA resembles FS in SNR and bits-requirement, but much faster than FS. Due to its inherited robustness, regularity and high parallelism in architecture, the improved genetic algorithm can be well implemented in real-time video encoders using VLSI technique.
5. Reference


Figure 4 The original and decoded frames of Garden sequence using different searching algorithms

Figure 5. Performance comparison in SNR of various sequences using different algorithms

Figure 6. Performance comparison in bits-requirement of various sequences using different algorithms