Video Segmentation and Summarization Based on Genetic Algorithm

Yang Xue
College of Physical Science and Information Engineering
Hebei Normal University
Shijiazhuang, China

Wei Zhicheng*
College of Physical Science and Information Engineering
Hebei Normal University
Shijiazhuang, China

***Abstract*—This paper presents a Binary Genetic Algorithms (BGA) based video summarization system. The similarity functions are first defined to evaluate segmentation, which are extremely expensive to be optimized with traditional methods. Then the system employs binary crossover and mutation operators to get the meaningful summary in a video search space. In order to test performance of the BGA method, we first compare the BGA method with Decimal Genetic Algorithms (DGA) method. The obtained results show that it is more quickly to find the best results for BGA than DGA. Second, the BGA method and the uniform approach have been compared. Experimental results show that the BGA method can capture more information than the uniform method and reduce redundancy.**

Keywords-video summarization; keyframe; genetic algorithms; fitness function

I. INTRODUCTION

Video summarization techniques try to provide the user with a summary containing sufficient information to obtain a quick idea of what happens in the video. There are two different kinds of abstracts viz. still image abstract (keyframes) and moving image abstract (video skim) [1]. The former is a presentation of salient images or keyframes, the latter consists of both the collection of image sequences and the corresponding audio abstract. Since the still image abstract is the simplest and most intuitive, it is the field that has attracted by far the most attention.

A simple method for extracting keyframes is selecting them from each shot by measuring low level information such as color, edges or motions. Also, there are other complicated methods existing for providing summarizations. For example, Sundaram and Chang [2] used Kolmogorov complexity to measure the complexity of each shot, and computed the video summarization according to both complexity and additional semantic information. The authors in [3] proposed a statistical framework based on hidden Markov models (HMMs) for video skimming. A hierarchical video structure summarization method using Laplacian Eigenmap is proposed in [4]. These algorithms overcome some technical problems from different angles, but these methods have the disadvantage of the model complexity, huge computation, and poor universality.

To overcome these shortcomings, in this paper, we propose a BGA method to extract keyframes. Genetic Algorithms (GA) has several advantages for video summarization. First, the genetic mechanism is independent on the fitness function and supports different types of evaluation equations. Second, GA is global searching method and naturally suitable for decremental heuristic search. Moreover, GA has a distinct advantage in terms of time.

II. FEATURE EXTRACTION

A video usually have thousands of frames, and a lot of frames may be similar to adjacent ones [5].We should give special attention to those that are not too similar, so we first subsample the video at a lower rate.

In this paper, the original set of frames is called *C*. We pick out the frames from *C* by measuring their differences with color histograms [6]. We reduce the HSV color space with color quantization into 256 colors, which includes 16 levels in H, 4 levels in S, and 4 levels in V. The histogram is defined by

$$H = (h_0, h_1, h_2, \ldots, h_{n-1})$$

Where n is the number of histogram bins. For any two images *i* and *j* we define

$$d(i,j) = \sum_{m=0}^{n-1} \left| H_i(n) - H_j(n) \right|$$
$$dh(i) = d(i-1, i)$$
$$dh(0) = d(1,0)$$

Where H_i is the HSV histogram of image i.

The video is subsampled by taking only those with *dh* greater than one standard deviation from the mean:

$$C' = \left\{ j \in C \,\middle|\, dh(j) > \overline{dh} + \varepsilon_1 \right\} \quad (1)$$

* Corresponding author (Email:weizhicheng@hebtu.edu.cn)
This work is supported by Natural Science Young Foundation of Hebei Department of Education (Grant No. 2010258).

Where ε_1 is a parameter which determine the number of frames in set C'.

On C', we define $Dh(i)$ as we did $dh(i)$ on C.

III. EVALUATION FUNCTIONS

We evaluate an image by its importance. The importance of an image can be got by three factors which are length factor, commonality factor and precedence factor. The three factors will be introduced in detail.

Since the longer shot is more important, one factor to define importance is to use length as a criterion. On the reduced set C', the length factor δ_i of an image i is computed as

$$\delta(i) = \text{number of frames in } C \text{ from } i \text{ to the next frame in } C'.$$

People are always attracted by less common image in a video, if they have no subjective intention. Another factor to define importance is to use commonality, as in Uchihashi and Foote [7]. For weighting each image by its commonality factor P_i, we first define a set R_i as follows

$$R_i = \left\{ j \in C' \middle| d(i,j) < \overline{Dh} + \varepsilon_2 \right\}$$

The parameter ε_2 determine the number of frames in set R_i. Then commonality factor is computed as

$$P_i = |R_i| / |C'|. \qquad (2)$$

In this equation, $|R_i|$ and $|C'|$ denote the number of frames in sets R_i and C' respectively.

In addition, earlier appearing frames are more heavily weighted than later ones in the same set, so we extend this notion of importance by using precedence factor A_i as a criterion [8]. Let $W_i = \left\{ j \in R_i \middle| i \le j \right\}$, then the precedence factor is computed as

$$A_i = |W_i| / |R_i|. \qquad (3)$$

To take the three factors into consideration, we define the evaluation function as follows.

$$f(S_k) = \sum_{i,j \in S_k} d(i,j)(T_i + T_j) \qquad (4)$$

$$T_i = A_i \log(\delta(i)) \log(1/P_i) \qquad (5)$$

Where S_k is a k selected images subset from C'. We take the log of two factors because they can have large variations.

The effect of evaluation function is making keyframes more dissimilar and show the main idea of the original video.

IV. DECIMAL GENETIC ALGORITHM FOR VIDEO SEGMENTATION

GA is a search method used in computing to find exact or approximate solutions to optimization and search problems [9]. Due to the large number of the set C', there is no efficient standard algorithm to optimize (4). But genetic algorithm (GA) is able to search this space effectively.

In this section, we describe a Decimal encoding GA method (DGA) for video summarization. DGA method can be described by specifying the encoding, fitness function, crossover and mutation operations.

Fig. 1 shows the outline of DGA method for video summarization.

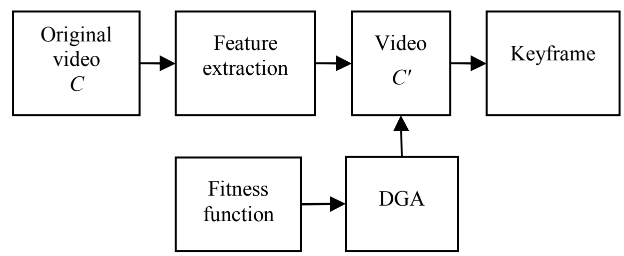


Figure 1. The block diagram of the DGA method for video summarization.

A. Encoding

For the encoding, we take a integer string called a chromosome of dimension D, where D represents the number of keyframes. Given a video with N number of images, Each gene in the chromosome $C_i (i = 1, 2, \cdots D)$ represents corresponding frame in the original video.

B. Fitness Function

In DGA, we take (4) as fitness function. Any well-defined evaluation function can also be used to characterize the desirable properties of summarization.

C. The Crossover and Mutation Operators

The crossover and mutation operators used in the Standard GA (SGA) are possible to produce redundant genes, we proposed novel crossover and mutation operators in this section.

Two chromosomes selected for crossover are merged and sorted to form an intermediate chromosome. Then two child chromosomes are produced by picking odd position genes and even position genes separately from the intermediate chromosome. Fig. 2 shows the crossover operation. As can be seen the offspring are produced without any redundancy due to the merging and odd-even frame position distribution.

5	12	27	31	48

Parent Chromosome 1

7	12	22	34	45

Parent Chromosome 2

5	7	12	12	22	27	31	34	45	48

Intermediate Chromosome

5	12	22	31	45

Child Chromosome 1

7	12	27	34	48

Child Chromosome 2

Figure 2. The DGA crossover operation

Each gene in the chromosome produced by the crossover operator is mutated with a probability p_m. The mutation operator mutates a gene C_i into C_i' as follows [10].

$$C_i' = \begin{cases} rand(1, C_{i+1} - 1) & if \quad i = 1 \\ rand(C'_{i-1} + 1, N) & if \quad i = D \\ rand(C'_{i-1} + 1, C_{i+1} - 1) & otherwise \end{cases} \tag{6}$$

Like the crossover operator, this mutation operator performs without any redundancy in frame numbers. The offspring will be generated by crossover and mutation operations. Then the offspring and the parents compete to survive in the next generation.

V. Binary Genetic Algorithm for Video Segmentation

In this section, we present another GA method for video summarization, Binary encoding Genetic Algorithm (BGA). We also describe it by specifying the encoding, fitness function, crossover, and mutation operations.

A. Encoding

For the encoding, every chromosome is a bianry bit string, like the SGA. The bit position of a chromosome is an index for a image in C'. The length of the chromosome is the number of frames. We use the left position of 1 to denote the segment boundary. For example, a chromosome 01000101001 breaks $C' = \{i_0, i_1, \ldots, i_{10}\}$ into the segments $\{i_0\}$, $\{i_1, i_2, i_3, i_4\}$, $\{i_5, i_6\}$, $\{i_7, i_8, i_9\}$, $\{i_{10}\}$. The number of segments is set to be a fixed constant. In BGA, we take boundary images as the keyframs in the summary.

B. Fitness Function

Any well-defined evaluation function can be used as the fitness function. In this algorithm , we also take (4) as the fitness function.

C. The Crossover and Mutation Operators

The BGA works by randomly selecting chromosomes to reproduce for the next generation. The selection is biasing toward individuals with higher fitness. For the crossover operator, two chromosomes are sliced at the crossing site, and the two tail pieces are swapped and rejoined with the head pieces to produce two offsprings like the SGA. But instead of crossing at a random bit, we randomly select a segment as the crossing site. All positions are selectable with equal probability.

After crossover procedure, a mutation operator is then defined as selecting gene randomly and increasing or reducing the 1's number in order to maintain the fixed number of segments.

VI. Experimental and Results

We illustrate the BGA method by an example of summarizing a video ‘XYang’ with 990 frames. When we calculate the fitness function, the parameter ε_1 in (1) determine the number of frames in set C'. In order to achieve a better tradeoff between performance and real-time, we set the parameter $\varepsilon_1 = 0.05$ after experiments.

The BGA and DGA methods are both applied to the video with k=6, mutation probability p_m=0.2 and crossover probability p_c=0.8. Population size is 50. And generation number is 100.

To show the effectiveness of the method, we do two kinds of comparisons. First, we compare BGA method with the DGA method to get the better one. Second, we compare the BGA approach with the uniform approach.

A. The comparision of BGA and DGA

We have experimented 1000 times using these two methods respectively on the video ‘XYang’. Fig. 3 shows the distributions of the experimental results. It is shown that the BGA and DGA can both find optimal fitness 110.1367. But the BGA is more likely than the DGA to get the optimal result.

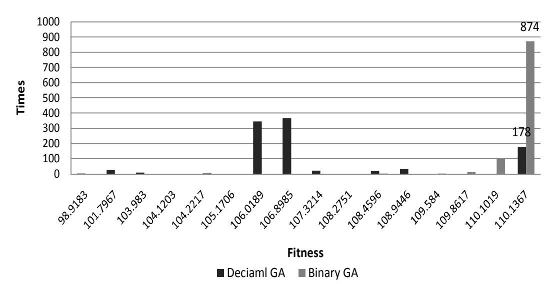


Figure 3. Result distributions of two kinds of genetic algorithms. We experiment 1000 times using BGA and DGA respectively on video ’XYang’. The optimal fitness is 110.1367.

Fig. 4 shows the performance of two approaches of GA for video summarization. According to this figure, we can see that BGA outperforms the DGA in convergence speed.

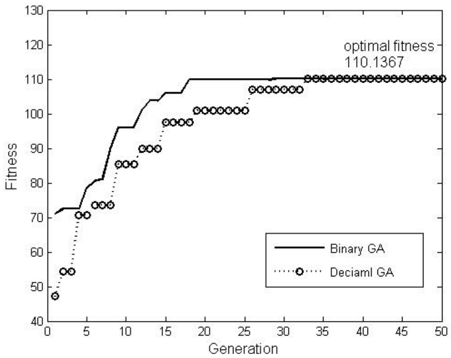


Figure 4. Convergence speed comparison between BGA and DGA

Table I. compares the difference of time and performance between BGA and DGA based on four videos. It is shown that the probability of finding the optimal fitness using BGA is much higher than the DGA with all these videos. It also can be seen that the convergence time of BGA is shorter than DGA if there are few key frames in the still abstract. Unfortunately, if the frame number is increased, the time advantage of BGA will become indistinct.

It has been shown that in the experiments, the BGA uses single-point crossover which is conducive to retaining the good templates. That is to say the short, low-order, highly-fit schemas are easier to be recombined to form more highly fit higher-order schemas in BGA. Meanwhile, the DGA merges two chromosomes and sorts to form an intermediate chromosome (Fig. 2). It's a kind of multi-point crossover. The templates will easily be damaged. Based on the building block hypothesis, the BGA has the advantage of keeping good templates into next generation. The proportion of finding optimal fitness value will be higher. It is consistent with experimental results that BGA method is much better than the DGA.

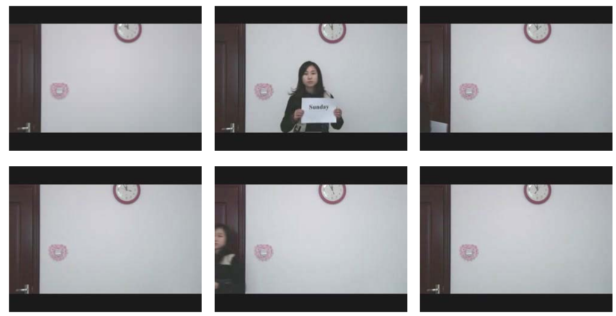
B. *The comparision of BGA and uniform approach*

We also compare the experimental results between BGA and uniform approach. Fig. 5 compares the six key frames of video 'XYang' between BGA approach and uniform approach which is mostly used by current video browsers. It is shown that the BGA based method can grasp the salient information to obtain a quick idea of what happens in the video and reduce redundancy. On the other hand, the uniform method gets too much redundant information. It is hard for people to get the main idea of this video.

VII. Conclusion

This paper describes a BGA based method for video summarization. We pick out the least similar images from original video by measuring the differences with the technique of color histograms. Fitness function is defined for genetic algorithm to search the optimized frames. We first compare the BGA method and DGA method. The experimental results show that BGA outperforms the DGA in both the convergence speed and the possibility to get the global optimum.

a. BGA approach



b. Uniform approach

Figure 5. Keyframes of two methods from video 'XYang'

Furthermore, the performance of BGA and the uniform approach have been compared. Experimental results show that the BGA method can capture optimal results to obtain the main idea of the original video and reduce redundancy.

As future paths of research, we will analyze the effect of parameters and fitness functions. Hybridization of GA with the present video abstraction methods to improve the performance is also worth an investigation.

References

[1] B. T. Truong, S. Venkatesh, "Video Abstraction: a Systematic Review and Classification," ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 3, No. 1, Feb. 2007.

[2] H. Sundaram, S. F. Chang, "Constrained utility maximization for generating visual skims," in Proc. IEEE Workshop Content-Based Access of Image and Video Library, pp. 124-131, 2001.

[3] S. Benini, P. Migliorati, R. Leonardi, "Hidden markov models for video skim generation," in the Eighth Int. Work. on Image Analysis for Multimedia Interactive Services, June. 2007.

[4] M. Jiang, A. H. Sadka, D. Crookes, "Hierarchical video summarization in reference subspace," IEEE Trans. On Consumer Electronics, Vol. 55, No. 3, pp. 1551-1557, 2009.

[5] Z. Z. Tabrizi, B. M. Bidgoli, M. Fathi, "Video Summarization Using Genetic Algorithm and Information Theory," proceedings of the 14th International CSI Computer Conference, Tehran, Iran, pp. 158-163, 2009.

[6] J. S. Boreczky, L. A. Rowe, "Comparison of video shot boundary detection techniques," Storage and Retrieval for Still Images and Video Databases IV, Proc. of SPIE 2670, pp. 170-179, 1996.

[7] S. Uchihashi, J. Foote, "Summarizing Video using A Shot Importance Measure and A Frame-packing Algorithm," Proceedings of ICASSP '99, Vol. 6, pp. 3041-3044, 1999.

[8] C. Patrick, G. Andreas, P. Wolf et. al "A Genetic Algorithm for Video Segmentation and Summarization," Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), New York, Vol. 3, pp. 1329, 2000.

[9] G. R. Harik, F. G. Lobo, D. E. Goldberg, "The compact genetic algorithm," IEEE Transactions on Evolutionary Computation, Vol. 3, No. 4, pp. 287-297, 1999.

[10] D. K. Ashwin Raju, C. Shunmuga Velayutham, "A Study on Genetic Algorithm based Video Abstraction System," Proceedings of NaBIC 2009, Coimbatore, pp. 878-883, 2009.

TABLE I. PERFORMANCE COMPARISON BETWEEN BGA AND DGA

	Video name	**Total frame number**	**Frame number in the abstract**	**Average time (s)**		**The proportion of finding optimal fitness value**	
				BGA	***DGA***	***BGA***	***DGA***
1	XYang	990	6	0.7887	0.8512	87.4%	17.8%
2	Aquarium	803	9	1.4295	1.1194	83.6%	24.6%
3	Downtown	706	9	1.1766	1.0674	86.4%	8.7%
4	Golf	524	6	0.7696	0.8383	99.5%	50.5%