Interactive Genetic Algorithms for Facial Composite Generation

Gül Nildem Demir
Department of Computer Engineering, Istanbul Technical University
Maslak, Istanbul, 34469, Turkey
nildemdemir@hotmail.com

ABSTRACT

The aim of this project is to generate a target face using interactive genetic algorithms (IGAs). Genetic algorithms are a variation of evolutionary algorithms based on the Darwinian principles of natural selection. In IGAs, users get involved in the algorithm on fitness evaluation and stopping stages. The described system implements two types of the IGA: the interactive generational GA (IGGA) and the interactive steady state GA (ISSGA).

Categories and Subject Descriptors

I.2.8-Problem Solving, Control Methods and Search

General Terms

Design, Algorithms, Human factor

Keys

Genetic algorithms, interactive genetic algorithms, computerized facial composite generation, active appearance model.

1. INTRODUCTION

Evolutionary algorithms (EAs) are nature based heuristics which help in solving optimization problems. In this study, we used genetic algorithms (GAs), which are a type of EAs, to build a computer based automatic face generation tool.

Face generation becomes an import issue when the face of a suspect needs to be described by the witness. There are a variety of facial composite generation methods to visualize a target face [1]. Our technique is inspired from face generation tools Evo-FIT and Eigen-FIT which are also based on EAs [1].

In this study, we built a system based on the parameter vector of the active appearance model (AAM) [1] and two different IGA approaches to generate the target face. The evolutionary process of the GA is represented through selection, recombination, mutation and replacement operators. In our study the interactive versions of the GAs differ from standard GAs in fitness evaluation and stopping stages where the users get involved in the algorithm. We implemented in our system the following IGAs: Interactive Generational GA (IGGA), where the whole population is replaced at each iteration and the Interactive Steady-State GA (ISSGA) where only one or no individual is replaced.

The paper is organized as follows: Section 2 gives the description of the genetic algorithms and their interactive versions. Section 3 shows the experimental results and section 4 presents the conclusion and future work.

2. INTERACTIVE GENETIC ALGORITHMS FOR FACIAL COMPOSITE GENERATION

2.1 Genetic Algorithms

Genetic algorithms are a variation of evolutionary algorithms [2] [3]. Based on the population replacement strategy, there are two kinds of GAs: The generational GA (GGA) and the steady state GA (SSGA). In the GGA, the whole population is replaced at each generation. In the SSGA, only a few individuals (or none) are replaced at a time. A population is a set of chromosomes and each chromosome represents a possible solution of the algorithm. Chromosomes consist of numeric values called "genes". After the generation of the initial population randomly, the chromosomes in the population undergo an iterative evolutionary process. The completion of each iteration causes the initial population to be replaced by the final mating pool. In our solution we designed the evolutionary process as a composition of the following operators for a population with size N and chromosome length n:

1. Selection: In the GGA, the tournament selection method is used. N chromosome pairs are selected randomly from the parent population. For each pair, the chromosome with the highest fitness value goes to the mating pool. The ith and $(i+1)$th chromosomes in the mating pool become pairs, which will proceed to the cross-over operation. In the SSGA only one random pair is selected for the cross-over operation.
2. Crossover: The crossover mechanism is almost the same in both GAs. For each pair, two points are selected randomly in the range [0, n]. Every gene between these points is exchanged between the parent chromosomes with probability pc. In the GGA, N offspring are formed after the crossover operation. In the SSGA only one offspring is generated.
3. Mutation: The gene values of each offspring chromosome are changed with the probability pm. In both algorithms the Gaussian mutation technique is used: A random variable drawn from a normal distribution with mean 0 and standard deviation (step size) σ is added to each gene value: $x'_i = x_i + N_i(0, 0.1)$, $i \in \{1,...,n\}$ If the new gene

value exceeds the boundary values, the mirroring method is applied

4. Replacement: The previous population is replaced by the offspring. In the GGA the offspring replace the parents. In the SSGA, if the offspring's fitness value is greater than one of its parents, that parent is replaced by the offspring.. In the case of replacement with elitism, the fittest chromosome of the parent population replaces the least fit chromosome of the offspring.

Pseudocode:
```
generate initial population randomly
repeat
  evaluate fitness of individuals
  perform reproduction
    select pairs
    recombine pairs through cross-over
    apply mutation
    replace the initial population
until stopping criteria met
```

2.2 Interactive Genetic Algorithms

In the IGGA, the user assigns a fitness value to each individual by ranking face images corresponding to individuals [1]. In the ISSGA, "replacement of the worst parent" strategy is implemented by having the user select two face images from a set of three faces based on the similarity to the target face [1]. The selected faces are "fitter" than the unselected ones and they replace the parent individuals in the population. In both cases, the user has the ability to stop the algorithm. Therefore, the stopping criterion is the satisfaction of the user with the displayed face images [1]. We have excluded the elitism at the replacement stage.

2.3 Implementation

In our problem, chromosomes represent faces and faces are defined as a set of AAM [4] parameters. These parameters are real numbers in the range [-0.3, 0.3]. The number of AAM parameters used in the model is 17. Therefore each chromosome has n=17 genes represented as real numbers. The crossover probability pc is selected as 0.8 and the mutation probability pm as 1. The step-size (standard deviation) of the Gaussian mutation is taken as 0.1. The population size is selected as 8. The initial population is generated randomly according to a Gaussian distribution with mean 0 and standard deviation 0.1.

3. EXPERIMENTS

For the experiments, we specified a test set of four face images to generate, eight subjects to generate target faces and ten different subjects to identify the generated face images.

Two images of the test set belong to people whose faces are well-known by the subjects and their face images exist in our database. One image belongs to someone, whose face is well-known by the subjects and his/her face image does not exist in the database. The last image belongs to someone, whose face is not well-known by the subjects –a face that is only seen for about 5 minutes- and his/her face image does not exist in the database. Eight subjects ran the IGGA and ISSGA four times each to generate the target face images of the test set. We calculated the average face images for each target face per algorithm. These generated average images are shown to the other group subjects and they are asked to identify the images. Tables 1 and 2 show the experimental results after the runs of IGGA and ISSGA.

Table 1. Experimental results of IGGA

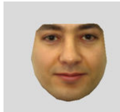
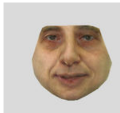
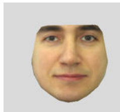
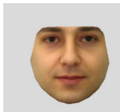
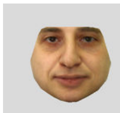
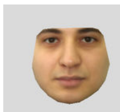
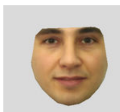
Target face	Average face through IGGA	Ident. ratio	Average iteration number	Av.eva-luated face number
		80%	≈11	≈88
		100%	≈10	≈80
		50%	≈7	≈56
		50%	≈9	≈72

Table 2. Experimental results of ISSGA

Target face	Average face through ISSGA	Ident. ratio	Average Iteration Number	Av.eva-luated face number
		40%	≈37	≈116
		100%	≈32	≈101
		70%	≈34	≈107
		40%	≈39	≈112

As seen above, the identification ratios of all target faces except the second target face with the identification ratio of 100%, are between 40% and %80 for both algorithms. The second face has different characteristics than the others, which makes it easier to recognize. But the rest of the face images are somewhat similar to each other. However, conducted experiments are not enough to draw a general conclusion. We specified a test set of four face images, which is a very small one. Moreover, the number of experimental runs is also very small (one run per target image). We also calculated the average of facial composites generated by the subjects per algorithm. To make the average more meaningful we should have done more experiments.

4. CONCLUSION AND FUTURE WORK

In this work, we have completed the target system successfully. Preliminary results are encouraging, but more experiments are needed. This work is a preliminary study, therefore it aims to give an overall performance analysis and an idea about the applicability of the project. More work has to be done to make the project compete with similar works in the literature. Future works:

- Adding self-adaptive mutation and comparing the Gaussian mutation with self-adaptive mutation,
- Increasing the number of face images in the face database to improve the AAM,
- Conducting more experiments on a bigger test set,
- Making face properties like size, shape, placement editable during run-time and updating the AAM vector respectively,
- Building the models of facial features separately and making a facial feature constant while the other feature models change,
- Adding hair style, moustache, beard,
- Adding the aging effect.

5. ACKNOWLEDGEMENTS

This work is being developed as a senior year graduation project under the supervision of Dr. A. Şima Uyar and Binnur Kurt in the Computer Engineering Department of Istanbul Technical University, Turkey.

6. REFERENCES

[1] Akbal T., Demir G. N., Kanlikilicer E., Kus M. C., Ulu F. H., “Interactive Nature-Inspired Heuristics for Automatic Facial Composite Generation”, Genetic and Evolutionary Computaton Conference Undergraduate Student Workshop, 2006.

[2] Eiben A.E., Smith J.E., Introduction to Evolutionary Computing, Springer, 2003.

[3] Eiben A. E., Schoenauer M., “Evolutionary Computing”, Information Processing Letters, 82(1): 1-6, 2002

[4] Matthews I., Baker S., “Active Appearance Models Revisited,” International Journal of Computer Vision, Vol. 60, No. 2, pp. 135-164, November 2004 .