NICHING METHODS WITH THE BREEDER GENETIC ALGORITHM FOR GEOMETRIC PRIMITIVE EXTRACTION: PRELIMINARY RESULTS

Diago, L(*) and Ochoa, A.

(*)Center of Bioengineering, CEBIO, ISPJAE - University, 127 s/n Marianao, C-Havana, Cuba.

diago@electrica.ispjae.edu.cu

Institute of Cybernetics Mathematics and Physics, Calle 15 e/ C y D, Vedado, C-Havana, Cuba.

ABSTRACT

Geometric primitive extraction or localization is a widely used method in image recognition tasks. A geometric primitive is a curve or surface, which can be described by an equation with a number of free parameters. A major problem faced in geometric primitive extraction deals with the characteristics of the search space: multimodal, nonlinear, noisy and highly dimensional; and with the possible variations of shapes of primitives. Genetic algorithms (GAs) [1] offer a domain-independent approach suitable for working with such complex problems.

The Breeder Genetic Algorithm (BGA) [8] is a GA inspired by the science of breeding animals. This paper explores the use of a BGA with niching methods [7] to find multiple primitives at the same time. Niching methods extend GAs to domains that require the location and maintenance of multiple solutions. We investigate two types of niching techniques, the so-called sharing and crowding methods. Our simulations have shown a clear superiority of crowding in this application. Therefore, we develop and test this method further to consider the use of truncation selection in the BGA with eye fundus images.

In this paper, a geometric primitive is a curve or surface, which can be described by an equation with a number of free parameters. It has recently been shown that extracting the best geometric primitive from a given geometric data is equivalent to finding the optimum value of a cost function [11]. Geometric data are an unordered list of points in two or three-dimensional Cartesian space, which can be obtained by edge detection process. Once it is understood that primitive extraction is such an optimization problem, the use of GA suggests itself. To use GA, the parameter vector must be encoded in chromosome form. The parameter vector is what we have into the chromosomes to represent the primitive's parameters. In our experiments a chromosome representation of a primitive by a minimal subset of member points is used. Such a subset is the smallest number of points necessary to define a unique instance of a primitive [12].

The Hough Transform (HT) is the most common method of primitive extraction [6]. Roth and Levine [12] mentioned that the main disadvantage of the GA approach relative to the HT is that the GA must be applied repeatedly to the geometric data for each extracted primitive, while the HT extracts all the primitives at once. Our alternative find a number of primitives at the same time by using the idea of niches described in GA literature [7]. We investigate two types of niching techniques, the so-called sharing and crowding methods. Our simulations have shown a clear superiority of crowding in this application. Therefore, we develop and test this method further to consider the use of truncation selection in the BGA with eye fundus images.

1. INTRODUCTION

A genetic algorithm (GA) is an optimization approach based on the evolutionary metaphor [1]. Previous studies on why and how the GA works have proceeded along different lines, most arguments are based on the Schema Theorem [5]. Mhlenbein [8] have criticized this theorem and proposed the Breeder Genetic Algorithm (BGA).

BGA is inspired by the science of breeding animals. In this algorithm, each one of a set of virtual breeders has the task to improve its own subpopulation. Like human breeders a BGA uses truncation selection.

2. A BRIEF DESCRIPTION OF NICHING METHODS FOR GENETIC ALGORITHMS

Users utilize population size (n), crossover probability (pc) and mutation probability (pm) to maintain multiple solutions in the simple GA. Some studies utilize selection
pressure as an additional control parameter [3]. (In fitness-proportionate selection, selection pressure can be controlled through the use of fitness scaling). The expected number of one individual on a new generation is a function of the probability of that individual to be selected and the number of individuals in the current generation. If the relation between the expected number of the best individual and the rest of individuals is high then in the new generation will appear only the best individual, even if it is a poor solution. The evolution stops because the lack of diversity. This is called premature convergence. Niching methods extend the application genetic algorithms to domains that require the location and maintenance of multiple solutions [1].

2.1 Sequential location of niches

Instead of slowing a GA’s convergence, one could encourage it to converge quickly and then start another one. Goldberg suggests restarting GAs that have substantially converged, by reinitializing the population using both randomly generated individuals and the best individuals from converged population [2]. Reinitialization is somewhat similar to using high rate of mutation. Controlling the parameters of convergence of the population we remove the selected primitive from the image and continue searching with the residual data while it is significant [12]. The main disadvantage of this approach is that the GA must be applied repeatedly to the geometric data for each extracted primitive but it offers good results when the number of local optima is not high [9].

2.2 Sharing (SH)

The objective function of an arbitrary individual can be altered in proportion with the vicinity average with other individuals of the population. This strategy is known as sharing. It causes the dispersion of highly similar individual in the search space. It also opposes to the early convergence. The behavior of the function that allows to share the space could be defined as follows [7]:

\[ f'(i) = \frac{f(i)}{\sum_{j=1}^{n} sh(d(i, j))} \]

where:

\[ sh(d) = \begin{cases} 1 - \left( \frac{d}{\sigma} \right)^{\alpha} & \text{if } d \leq \sigma \\ 0 & \text{otherwise} \end{cases} \]

and \( \alpha \) is a constant (typically set to 1) used to regulate the shape of the sharing function. The computation of \( d(i, j) \) is carried out based on the Euclidean distance or Hamming distance by means of a comparison between phenotypes or genotypes of individuals.

2.3 Crowding (CW)

Crowding methods attempt to replace population members in a way that maintain diversity. They insert new elements into a population by replacing similar elements. Two individuals are similar if the distance (either Euclidean or Hamming) between them is equal or less than some value taken as threshold. We remove similar individuals from the population. This is known as crowding with elimination (CE) [7]. Alternatively some authors favor the mating of distant individuals instead of removing similar ones [4]. This is known as crowding with mating restrictions (CMR) [7]. A crossover strategy where new individuals are created only if they are better than their parents is used to favor the formation of niches. This is known as deterministic crowding (DC) [7].

2. GEOMETRIC PRIMITIVE EXTRACTION

In our experiments, the images were passed through an edge detection process using the sobel operator [10] to extract the silhouette of the objects. All the points that belongs to the contour of the image were stored in an array as an ordered pair (x,y) and the chromosome codify a primitive using the index of the points in the array. For primitive extraction, an individual geometric primitive is a population member. The parameter’s vector is encoded in the chromosome by the smallest number of points necessary to define a unique instance of a primitive called minimal subsets. In the case of the circle, as we have three parameters (the radius and the coordinates of it center) then only three points are required to determine its equation. This is the number of genes \( n = 3 \). By randomly selecting a minimal subset from the geometric data, the parameter’s vector must be determined solving an equation system (inversion problem).

The fitness function \( f \) then counts the number of points from the geometric data that fall within a fixed-distance \( d_{th} \) of the geometric primitive:

\[ f(p_1, \ldots, p_n) = \sum_{i=1}^{n} s(r_{p_i}) \]

where:

\[ s(r_p) = \begin{cases} 1 & \text{if } r_p \leq d_{th} \\ 0 & \text{otherwise} \end{cases} \]

and \( r_{p_i} \) is the smallest distance from the point data \( p_i \) to the parameter vector of the selected primitive. The larger the value of this cost function, the more significant the primitive. After each evaluation of the generation, we
applied the sharing function within the fitness function using phenotypes and the euclidean distance. We also used crowding with elimination.

A random sampling of minimal subsets was performed to create the initial population of N individuals. The selection method used was the “truncation selection” with the T% of the population for mating and discrete recombination without mutation was used to create new geometric primitives.

According to the fitness function, primitives with larger radius will be favored. To avoid this we introduce a normalization that take the fitness function as a relation between the number of points describe before and the number of points that must appear in the image for the primitive described by the selected parameter’s vector.

We also introduce a parameter that consider the environment where the geometric primitive will be found. For example, if we look for all primitive with bounded radius we take minimal and maximal radius from the size of the image reducing in this way the search space.

3.1 Crowding with elimination and truncation selection.

Mahfoud [7] studied this crowding technique in presence of proportional selection. As our algorithm uses truncation selection have been necessary the following modification: Instead of apply the elimination in all the generation it is carried out only in the set of selected individuals.

If the number of removed individuals is not bigger than some threshold, then the new generation is generated from the reduced selected set. Otherwise, individuals are added from the best unselected individuals provided that they are different enough to the selected ones.

4. PRELIMINARY RESULTS

We run BGA to find all the geometric primitives present in Figure 1 for different number of individuals in the initial generation (N=40, 60, 80). BGA runs 10 times for each number of individual, used 10 generations and T% = 0.2.

Table I shows the percent of times that BGA found the three primitives (P1, P2, P3) shown in Figure 2 by the two methods (SH and CE).

<table>
<thead>
<tr>
<th>N</th>
<th>SH(%)</th>
<th>CE(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0.9 0.7</td>
<td>0.5 0.9</td>
</tr>
<tr>
<td>60</td>
<td>1 0.9 0.5</td>
<td>1 0.9 0.9</td>
</tr>
<tr>
<td>80</td>
<td>1 1 0.8</td>
<td>1 1 1</td>
</tr>
</tbody>
</table>

Figure 3 shows a group of primitives considered as bad results in the cases that BGA could not find all the primitives.

We have also obtained good results with more complex images (Figure 4) for the objective analysis of the retinal fundus morphology, particularly the disc-cup ratio, widely used in glaucoma diagnosis [9]. We use only 1000 evaluations to extract both primitives.
5. DISCUSSION AND CONCLUSIONS

For better use of the space we have shown this simple example that illustrates a clear superiority of CE over SH (see Table I). CE found all primitives in 10 generations with N=60, and SH found only two. Both methods with N=80 found all the primitives.

For the primitive shown in Figure 1b the contour information is not complete. Sometimes BGA fails (see Figure 3b) because the fitness function presented in this paper has several limitations when the information is not complete.

For the retinal fundus morphology Diago et al [13] used sequential location of niches running GA for the determination of the disc-cup ratio in eye fundus images. They described the disc and the cup by the equation of an ellipse and run GA once with 40 generations of 20 individuals for the disc and again for the cup with the same parameters. We reduce the number of evaluation for the BGA using CE from 1600 to 1000.

REFERENCES

NICHING METHODS WITH THE BREEDER GENETIC ALGORITHM FOR GEOMETRIC PRIMITIVE EXTRACTION: PRELIMINARY RESULTS

ABSTRACT

Geometric primitive extraction or localization is a widely used method in image recognition tasks. A geometric primitive is a curve or surface, which can be described by an equation with a number of free parameters. A major problem faced in geometric primitive extraction deals with the characteristics of the search space: multimodal, nonlineal, noisy and highly dimensional; and with the possible variations of shapes of primitives. Genetic algorithms (GAs) offer a domain-independent approach suitable for working with such complex problems. The Breeder Genetic Algorithm (BGA) is a GA inspired by the science of breeding animals. This paper explores the use of a BGA with niching methods to find multiples primitives at the same time. Niching methods extend GAs to domains that require the location and maintenance of multiple solutions. We investigate two types of niching techniques, the so called sharing and crowding methods. Our simulations have shown a clear superiority of crowding in this application. Therefore we develop and test this method further to consider the use of truncation selection in the BGA with some eye fundus images.