Automatic Pattern Recognition of Binary Image Objects using Mathematical Morphology.

Emerson Carlos Pedrino

and

Valentin Obac Roda

Departamento de Engenharia Elétrica
Escola de Engenharia de São Carlos - Universidade de São Paulo
Avenida Trabalhador São-Carlense, 400
13566-590 São Carlos (SP) – Brazil
Tel: (55 16) 3373 9363
FAX: (55 16) 3373 9372
ecpedrin@sel.eesc.usp.br
valentin@sel.eesc.usp.br

Abstract. In this paper, we present a methodology for automatic pattern recognition of binary images objects using morphological operators and genetic programming. The results are expressed in terms of the basic morphological operators and logical operators. Genetic Programming (GP) is based on concepts of genetics and Darwin’s principle of natural selection to genetically breed and evolve computer programs to solve a wide variety of problems. GP is a relatively new branch of evolutionary computation and it is gradually consolidating as a promising methodology to be used in applications involving pattern recognition. Mathematical morphology is based on the set theory (complete lattice) where the notion of order is very important. This processing technique has proved to be a powerful tool for many computer vision tasks. An example of application is presented and the result is compared with other methods in the literature.
1 Introduction

Morphological image processing is a nonlinear branch in image processing developed by Matheron and Serra in the 1960’s, based on geometry and the mathematical theory of order [1-2]. Morphological image processing has proved to be a powerful tool for binary and grayscale image computer vision processing tasks, such as edge detection, noise suppression, skeletonization, segmentation, pattern recognition and enhancement.

The design of morphological procedures is not a trivial task in practice. It is necessary some expert knowledge to properly select the structuring element and the morphological operators to solve a certain problem. In the literature there are several approaches using automatic programming to overcome these difficulties [3-4], however, they present several drawbacks as a limited number of operators, only regular forms of structuring elements, only morphological instructions, to name just a few.

Genetic programming (GP) is the most popular technique for automatic programming nowadays and may provide a better context for the automatic generation of morphological procedures [5]. GP is a branch of evolutionary computation and artificial intelligence, based on concepts of genetics and Darwin’s principle of natural selection to genetically breed and evolve computer programs to solve problems.

Genetic Programming is the extension of the genetic algorithms [6] into the space of programs. That is, the objects that constitute the population are not fixed-length character strings that encode possible solutions to a certain problem. They are programs (expressed as parse trees) that are the candidate solutions to the problem. For example, the simple program “min(x*2,x+2*y)” is illustrated in figure 1. The programs in the population are elements from the function set and the terminal set, which will represent the solution of problems in the domain of interest. In GP, the crossover operator is implemented by taking randomly selected subtrees in the individuals (selected according to fitness) and exchanging them.

![Parse Tree for “min(x*2,x+2*y)”](image)

Figure 1. Parse Tree for “min(x*2,x+2*y)”.

There are few applications of GP for the automatic construction of morphological operators in the literature [7]. We developed a linear genetic programming approach for the automatic construction of morphological and logical operators, generating a toolbox named morph_gen for the Matlab program. The developed toolbox can be used for the design of non linear filters, image segmentation and pattern recognition of binary im-
ages. An Example of application is presented and the result is compared with other approaches.

This article is organized as a brief review of the basic concepts of morphological operations and genetic programming, section 1; a detailed description of the developed algorithm, section 2; results and application examples are presented in section 3; section 4 presents the conclusions.

2 Automatic construction of morphological operators

The proposed algorithm developed in this paper for automatic construction of morphological operators uses a linear genetic programming approach that is a variant of the GP algorithm that acts on linear genomes. The developed algorithm operates with two input images, an original image and an image containing only features of interest which should be extracted from the original image. The genetic procedure looks for operator sequences in the space of mathematical morphology algorithms that allows extracting the features of interest from the original image. The operators are predefined procedures from a database that work with particular types of structuring elements having different shapes and sizes. It is also possible to include new operators in the database when necessary. The program output is a linear structure containing the best individual of the final population. The output result from one operator is used as input to subsequent operator and so on. The genetic algorithm parameters are supplied by the user using a graphical user interface (GUI). The main parameters are: tree depth, number of chromosomes, number of generations, crossover rate, mutation rate, error and certain kinds of operators suited to a particular problem. It has been used for the problems the mean absolute error (MAE) as a fitness measure. For example, the fitness function using MAE error was calculated as follows:

\[
\text{d}(a,b) = \frac{1}{XY} \sum_{i=1}^{X} \sum_{j=1}^{Y} |a(i,j) - b(i,j)|
\]  

In equation 1, \( a \) is the resulting image evaluated by a particular chromosome (program) and \( b \) is the goal image. The chromosomes are encoded as variable binary chains. The main steps of the proposed algorithm are illustrated in figure 2.
The genetic parameters and the images are supplied by the user; the initial population of programs is randomly generated. Since the chromosomes are encoded as binary chains, for example, if the user has selected the instructions: and (AND logic), sto (STORE), ero (EROSION) and cpl (COMPLEMENT), the first operator will be coded as “002”, the second as “012”, the third as “102” and the last one as “112”. If the tree depth chosen was four, for example, the chromosome: “000110112” could be created.

After evaluation of each chromosome in a generation, a fitness value is assigned to each one. The selection method used for genetic operators was the tournament selection. In crossover operation, morphological operators are randomly selected and exchanged between parents chromosomes. The mutation operation replaces a randomly selected instruction by another in the range of morphological algorithms space. The reproduction operator copies a single parent into the new generation according to its fitness value.

3 Results and application examples

In this section some results using the developed algorithm are presented. In figure 3 it is presented an original image and the goal image containing heads extracted from a fragment of a music score. The genetic procedure found the following best program to extract heads: “dil_dd_3->dil_q_3->do_nothing->ero_q_3-
The genetic parameters chosen for this task were: 50 generations, 25 chromosomes, tree of depth 6, crossover rate of 90%, mutation rate of 20% and reproduction rate of 20%. The MAE error found was less than 0.7%. The training time was less than 71 seconds and execution time was less than 0.02 seconds. This procedure was applied to image of figure 4 producing a very good result. The operator \textit{dil}_{dd}_3 is a dilation by a 3x3 diagonal structuring element. Operators \textit{dil}_{q}_3 and \textit{ero}_{q}_3 are a dilation and an erosion by a 3x3 square structuring element. The do\_nothing operator gives a relaxation for the tree depth chosen by the user.

![Figure 3. Procedure created automatically for head extraction.](image)

![Figure 4. Example of head extraction.](image)
Comparing the results with those obtained from [8] [9], our implementation presented improvements in fitness, processing time and programming flexibility.

Table 1 summarizes all the results obtained.

### Table 1. Summary of results.

<table>
<thead>
<tr>
<th>Example</th>
<th>Generations</th>
<th>chromosomes</th>
<th>Tree depth</th>
<th>Cross rate</th>
<th>Mut rate</th>
<th>Repr rate</th>
<th>MAE error</th>
<th>Training time</th>
<th>Execution time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head extraction</td>
<td>50</td>
<td>25</td>
<td>6</td>
<td>90%</td>
<td>20%</td>
<td>20%</td>
<td>0.7%</td>
<td>71s</td>
<td>0.02s</td>
</tr>
</tbody>
</table>

4 Conclusions

In this paper a method for automatic pattern recognition of binary image objects using morphological operators and a linear genetic programming approach was presented. An application example has been presented where the solution has been expressed in terms of the basic morphological operators, dilation and erosion, in conjunction with other instructions. Comparing with other methods described in the literature, the developed methodology presents many advantages such as an improvement in processing time, fitness and flexibility in relation to program size (variable) and types of operators. The developed method can be used as a guide to morphological design.

References