NAMAL INSTITUTE

AI Project Report

Template Matching through Evolutionary Algorithm

Muhammad Zeeshan Ahmed
Haris Ali
Sana Tariq
Manzoor Ul Haq

Department of Computer Science

November 9, 2020

Abstract

This project is based on three steps: Natural phenomenon , theory and computational model. These three steps follow the "Evolutionary theory".// The theory says that life on earth is related to each other and modifications occurs according to environment change and best-fitters to the environment remains and make the population. So whenever a model is created it comes from the idea of above three steps. When a natural phenomenon occurs it is observed and theory is made. To test the theory a model is generated, after the generation of model it is tested and then launched. The idea of the project is to determine a small image from a large image using some functions and logic based on best-fit. We gives the small and large image as input and using some functions we iterate some values and gathers those best match value to find the smaller image.

Contents

1 Natural Phenomenon

Natural phenomenon is that which takes place in reality and here the natural phenomenon we are taking is totally based on evolutionary theory. Evolutionary theory says that ”All life on the Earth is connected and related to each other and modifications of population taking place through process of natural selection, where some traits were favored in and environment over others”. Natural selection can change a species in minor aspects. For example, natural selection can cause a population to change colour or size over generations. Means that organisms change over time to better adapt to its environment that helps them to survive and have more offspring.

2 Theory

Theory extracted from natural phenomenon is that population exists every where with some variation in it and by doing process of natural selection, we will have the fittest individuals. New population will be generated by cross of best fit individuals and this population also have variation in it. So, this process is continuously taking place and changing population over generations according to environment.

3 Computational Model

Computational model is taking two template images as input in which one of the image is small part of the large image and finds the coordinates of smaller image within the large image.

Coordinates of large image collectively is a population then each coordinate or point is individual itself and every coordinate or point shows variation with other coordinates or points.

Our model will choose random population of some size from given large image in first iteration and place the small image on each individual to have the fittest one or simply max fitness value. If it fails to find the max fitness value in first iteration then it will arrange the population according to the fitness values from highest to lowest. After that it will take first two individuals from arranged population to cross to produce their children and apply mutation to make variation in them. Now these children will take place of their parents and then next two individuals from population will be taken and their children take place of their parents and so on till new population is produced which is basically the new generation from the random population. After this it will be in a loop which again rearrange the population according to fitness values from highest to lowest and take parents to produce children that take place of them. And goes on until it reaches to max fitness value or fittest individual.

4 Application

4.1 Libraries

Libraries that am using in this application are as follow:

4.1.1 scipy

Using ”stats”, one of the module in scipy that has a built-in function to find correlation value between two arrays which we are using as fitness value. Higher the correlation value, higher will be the fitness value and vice versa. Images are actually 2D arrays that are behind the template. One of the array is of small image and the other one is of the same size on some individual from population.

4.1.2 numpy

Using numpy, we are finding max index from list and generating a random number or list of random numbers.

4.1.3 matplotlib

We are using three modules of library ”matplotlib” which are as below:
1) pyplot
pyplot is used to show a graph in new window.
2) image
image is used to store image in a variable and that variable is actually the 2D array of that image.
3) patches
patches is used to draw a rectangle on individual having max fit value to detect the small image within large image.

4.2 Functions

Functions that are used in implementing this application.

4.2.1 slice(x, y)

In this function, we are slicing the large image at different individuals(coordinates) from population of same size as of small image by taking coordinates as parameters.

4.2.2 bestfit(xlist, ylist)

This function takes list of coordinates as I have stored x coordinates in one list and y coordinates in other list. So, this takes two list and returns the same list but arranging them according to the fitness values at each point. Points having highest fitness value are on the top of the list when it is returned. Along with this two, it also returns the list of fitness values accordingly arranged as other two lists.

4.2.3 binary(decimalNumber)

This function simply takes a decimal number and returns its binary form.

4.2.4 invert(binaryNumber)

This function takes a binary number and flip any random bit and then returns modified binary number.

4.2.5 concatenate($\mathbf{P}_x, P_y$)

This function takes coordinates as input and concatenate them and returns concatenated number and length of x and y coordinates. Because when we have to split x and y coordinate, we are required to have their lengths.

4.2.6 replaceBit($\mathbf{P1}_concatenatedNumber, P2_concatenatedNumber$)

Its functionality is to generate a random index from given any of two numbers in input and replace all bits from that index to the end of that number with the same bits of other one and then returns the modified numbers.

4.2.7 crossRelation($\mathbf{P1}_x, P1_y, P2_x, P2_y$)

In this function, we are calling the ”concatenate” function twice that concatenate the x and y coordinates of first point and store the resulting number in one variable. Similarly, concatenate second point and store the resulting number in other variable. Then, we call the function ”replaceBit” which will modify the numbers that we get from ”concatenate” functions by replacing their bits. This function returns the modified number returned from ”replaceBit”, length of x coordinate of first point and length of x coordinate of second point.

4.2.8 mutation(P1, P2, $\mathbf{lenP1}_x, lenP2_x$)

This function takes two concatenated points returned from ”crossRelation” and and calls ”invert” function on them. After that it will separate the x and y coordinates of both inverted numbers and returns x, y coordinates of both inverted numbers.

4.2.9 loop(xlist, ylist)

This function takes population(list of coordinates) and arrange them according to the fitness value by calling "bestfit". After that it will check the condition if top value in fitness value list is equal to the max fitness value which can be 1, 0.9 or 0.8 etc because our fitness is calculated on basis of correlation and correlation values are in between -1 to 1 then it will return that point on which max fitness value is found. Otherwise, it goes in loop by crossRelation among first two individuals from population and applying mutation on them to have two new points which takes place of previous from which they are generated. When the whole population is replaced by new population that means next generation is started and same work will also be done by them and this will where we find that individual having max fitness value.

4.2.10 main()

Main function is generating random population on which "bestfit" is to be called. After generating list of coordinates, it will call the loop function and when it returns the point where max fitness value is achieved, this function specify the position of small image within large image by drawing rectangle on that.

Whole procedure in simple words is that we are first taking random population which is set of coordinates from large image and finding out the best fit individual by matching the corelation value to 1.0. If not found then, we will move on to produce new generation after cross over which replaces bits of two coordinates and mutation process in which any single bit of both coordinates are flipped and loop starts which again and again producing generations untill we have a best fit individual means threshold of 1.0.