OBJECT COUNTING

Project 1

05 MART 2019
Dilay Sapmaz 041701032
MUHITTIN GOKMEN

MEF UNIVERSITY
COMPUTER PROGRAMMING
PROGRAMMING STUDIO

Abstract

In this report, the first project of the programming studio course written in the Python software language, "object counting", and the following steps were written.

Problem Definition:

In an image, the code will count how many object on that image. This can be useful for a place that needs to count how many substance or an image.

Solution methods

There are two solution methods:

1)Levialdi Algorithms

2)TSF Algorithms

1)Levialdi Algorithms:

Levialdi's shrinking algorithm shrinks every connected component in a binary image to a single isolated point to the upper right corner of its bounding rectangle and erases it.

2)TSF Algorithms

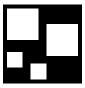
In this approach, the image space is partitioned like a checkerboard into two disjoint sets: 1 and 2. Where 1 and 2 denote the two distinct subfields.

My Work

Images consist of pixels. I need to work with pixels. Firstly, image's pixels' need to convert to binary. And I take gray code function from my instructor. Now the image is gray. It needs to be black and white. Then I convert 0 and 1. Zero means the pixel is black, objects' white parts are 1.

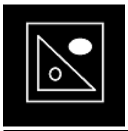
I started with Levialdi. Both algorithms have a while loop with ifs. Then TSF and it's functions. I cannot spend enough time to GUI. GUI needs much attention.

Results:



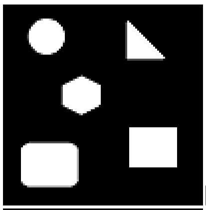
For this image

```
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Levialdi Counter: 4
Levialdi Iterations: 16
TSF counter: 4
TSF iterations: 6

Process finished with exit code 0
```



For this image, output is:

```
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.
Levialdi Counter: 4
Levialdi Iterations: 128
TSF counter: 4
TSF iterations: 65

Process finished with exit code 0
```



For this image, output is:

```
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Levialdi Counter: 5
Levialdi Iterations: 47
TSF counter: 5
TSF iterations: 15

Process finished with exit code 0
```

For this image, output is:

```
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Levialdi Counter: 16
Levialdi Iterations: 84
TSF counter: 16
TSF iterations: 46

Process finished with exit code 0
```

Contribution of this project:

I learn how to write in Python, and it's libraries. I struggle with complex code to write the algorithm. If I write GUI, I learn much more things for this project but the time was not enough.

Conclusions:

Levialdi is faster than TSF but it calculates much iterations. Time depends that