

# Programming Assignment 1: Region Growing

January 25, 2021

*Region growing* is an image segmentation algorithm. It examines neighboring pixels of initial seed points and determines whether the pixel neighbors should be added to the region. The region is iteratively grown by comparing all unallocated neighboring pixels to the region. The difference between a pixel's intensity value and the region's mean, is used as a measure of similarity. The pixel with the smallest difference measured this way is allocated to the respective region. This process stops when the intensity difference between region mean and new pixel become larger than a certain threshold ( $t$ ). Pixels adjacent relationship is defined by 4-connected or 8-connected neighborhood. Figure 1 shows examples of neighborhood of the black pixels in the center. The left image illustrates 8-connected neighborhood (eight gray pixels). The right image shows 4-connected neighborhood (four gray pixels).

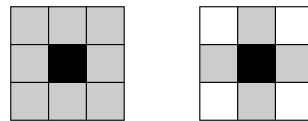


Figure 1: Examples of 8-connected and 4-connected neighborhood.

In this project, you will implement a special case of region growing for binary images (pixel values 0 or 255). You will use 4-connected neighborhood for pixels adjacent relationship. The basic idea of the algorithm is:

1. Select a set of seed points based on pixel intensity values (i.e., 0 or 255) to grow from.
2. The criterion to add a pixel to a region is the same pixel value. That is, we keep examining the *adjacent pixels* of seed points. If they have the same intensity value as the seed points, we place them in the same region.
3. The above process is repeated until there is no pixel to be added.

The images are presented in raw pgm format. The name “PGM” is an acronym derived from “Portable Gray Map.” Each PGM image consists of the following:

- A “magic number” for identifying the file type. A raw pgm image’s magic number is the two characters “P2”.
- Whitespace.
- A width, formatted as ASCII characters in decimal.
- Whitespace.

- A height, again in ASCII decimal.
- Whitespace.
- The maximum gray value (Maxval), again in ASCII decimal. Must be less than 65536, and more than zero.
- Newline or other single whitespace character.
- A raster of Height rows, in order from top to bottom. Each row consists of Width gray values, in order from left to right. Each gray value is a number from 0 through Maxval, with 0 being black and Maxval being white. Each gray value is represented in ASCII.

Your program (`growregion`) should run in command line. It takes two arguments: name of the image file, and the gray-scale value of the region of interest. It prints out your name, name of input image, and a list of comma separated numbers with the first one being the number of regions followed by the size of each region (number of pixels) in ascending order. For example, the following command performs region growing on image `test0.pgm`:

```
java growregion test0.pgm 255
```

It returns sizes of the 16 regions that have pixel intensity 255:

```
Yixin Chen
```

```
test0.pgm
```

```
16,2272,2273,2274,2274,2275,2276,2276,5945,5946,5946,5946,5947,5947,5948,5949,5954
```

If the gray-scale value of the region of interest is changed to 0, the following command:

```
java growregion test0.pgm 0
```

returns

```
Yixin Chen
```

```
test0.pgm
```

```
14,636,637,638,638,3316,3320,3320,3321,3322,3323,3323,3324,3324,148110
```

Please submit a report, JAVA source code and executable to Blackboard before **Monday, Feb 8th, 2021**. In the report, please provide a convincing explanation of your method and rationale for design and implementation decisions.