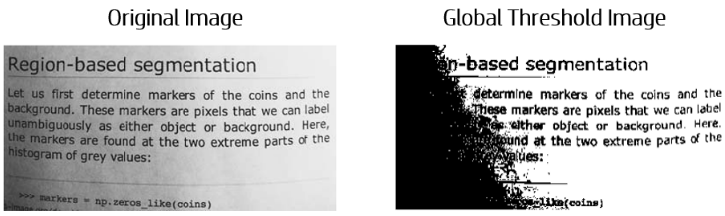
29 Adaptive Mean Thresholding

21 points

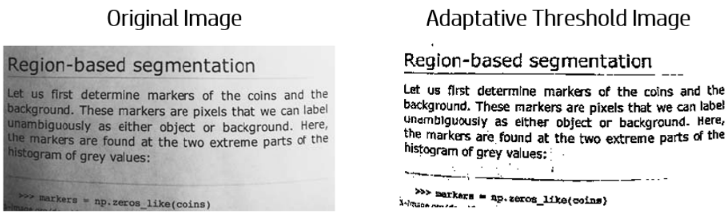
Introduction

The simplest thresholding methods replace each pixel in an image with a black pixel (value = 0) if the image intensity is less than some threshold value, or a white pixel (value = 1) if the image intensity is greater than that threshold value. This process is also called binarization of an image.

However, applying the same threshold value for the whole image is a very crude method and only works well when the image has no shadows or bright regions. The following is an example of the result of applying a basic thresholding algorithm vs the original image.



It is possible to improve the performance of thresholding by using Adaptive Thresholding methods. See the following example and note the difference between global and adaptive thresholding.



The objective of this problem is to implement an Adaptive Mean Thresholding algorithm. Don't panic! The principle of this algorithm is very easy to understand! Adaptive Mean Thresholding computes a threshold value per each pixel of the image instead of computing a single threshold value for all the pixels.

This individual thresholding value is obtained by calculating the mean of a NxN matrix (also called the 'window') whose center is a certain pixel. In the following example, we have centered the 3x3 window on the pixel colored light green. The pixels inside the 'window' are colored blue.

Image Size = 7 x 7

125	125	75	125	255	75	45
100	100	75	50	40	25	0
100	150	175	190	225	245	255
255	255	250	245	255	255	255
250	245	240	235	225	220	215
200	150	200	100	150	100	50
50	25	20	0	25	25	0

Window Size = 3 x 3

If the pixel value is greater than or equal to the obtained threshold, that pixel is assigned the value 1. If not, that pixel is assigned the value 0. See the following example:

225	220	215
150	100	50
25	25	0

$$1)\ mean\ threshold = \frac{225 + 220 + 215 + 150 + 100 + 50 + 25 + 25 + 0}{9} = 112.2 = \mathbf{112}$$

$$2)\ pixel\ value = 100$$

$3)\ pixel\ value \geq mean\ threshold?$ → NO

$$4)\ pixel\ value = 0$$

This process is carried out for each pixel in the image. Corner pixels and pixels situated along the sides of the image are missing some neighboring pixels. When placing the window around these pixels, assume that the value of missing neighbors is '0'. For example:

0	0	0					
0	125	125	75	125	255	75	45
0	100	100	75	50	40	25	0
	100	150	175	190	225	245	255
	255	255	250	245	255	255	255
	250	245	240	235	225	220	215
	200	150	200	100	150	100	50
	50	25	20	0	25	25	0

Write a program that applies the Mean Adaptive Threshold method to an input image given the size of the window.

> HINT: Round the computed mean threshold value to the nearest integer before comparing its value with the pixel value!

Input

The number of lines of the input depends on the size of the image, but it always follows the same structure:

- First line is the number of rows of the input image.
- Second line is the number of columns of the input image.
- The following lines correspond to the input image.
- Finally, the last line is the size of the window. This must be an odd value.

Output

The output is the thresholded image.

Example

Input

```
5
5
7 5 8 4 7
3 6 5 5 1
9 3 8 8 9
2 4 9 1 1
4 4 1 4 9
5
```

Output

```
1 1 1 1 1
1 1 1 1 0
1 0 1 1 1
1 1 1 0 0
1 1 0 1 1
```