

# Object Features Detection

Jan Gaura

2020-02-13

Today's exercise is focused on implementation of simple object features. This process will take some time and we will spend more time on this topic.

One of the possible ways in detection and classification of objects in images is using object features. Today's lecture will consist of the following steps:

- Thresholding the image,
- indexing objects.

In the end, our task is to label each object in Figure 1.

## Thresholding the Image

The first step is to separate objects in the image  $f$  from the background. In our example image, this task is not so hard, so we can apply the following thresholding criterion to each pixel in image  $f$  to construct a new image  $g$

$$g(x, y) = \begin{cases} 1, & f(x, y) > t \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

where  $t$  is a chosen threshold value. There are many methods to select  $t$ , see the text for this course. However, in the case of our input image, we can set the threshold manually.

After this step, we have a binary image that indicates where are the objects and where is the background.

## Indexing Objects

In order to compute moments and features of the upcoming classification, we need to separate detected objects from the previous step. This process is called indexing and its purpose is to distinguish between the objects in the binary image. The result of such process is shown in Figure 2

We can implement this process using a simple flood fill algorithm. In this algorithm, we go through all the pixels and if it belongs to the objects, we label this pixel with chosen constant and recursively flood fill four neighbours with this constant too. As one might expect, we label each object with different constant.

Value 1 can be changed to 255 to better see the output in an image that is grayscale bit 8 bits depth.

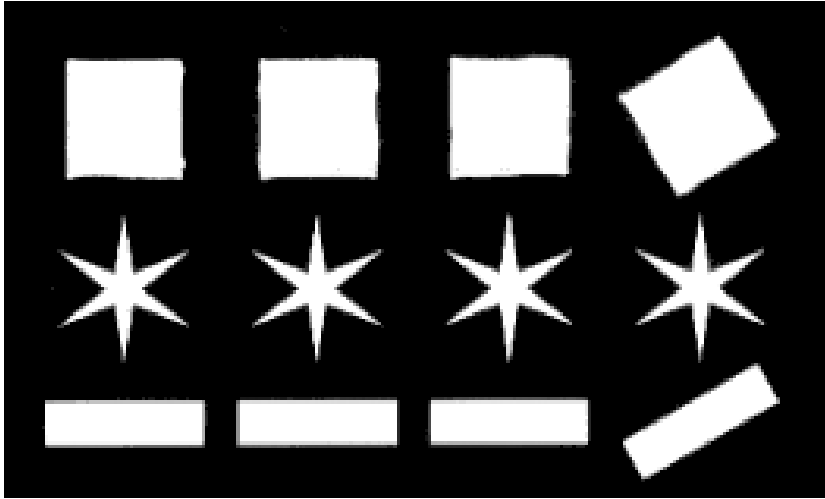


Figure 1: Input image.

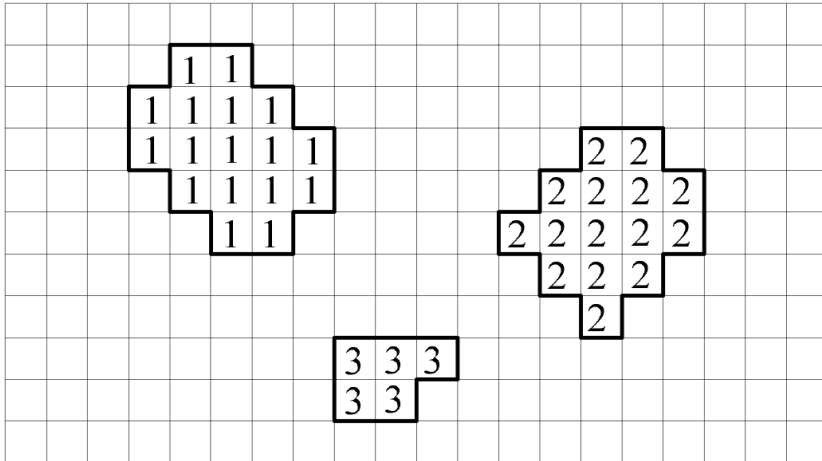


Figure 2: Indexed image.

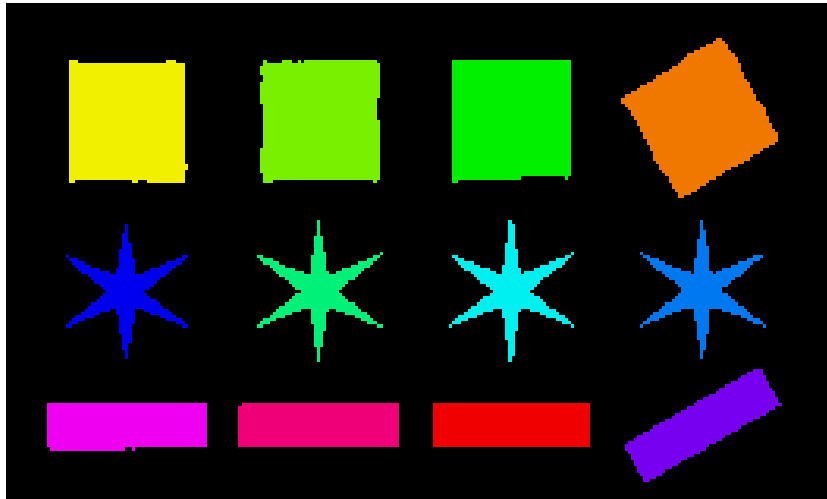


Figure 3: Colorized indexed image for out training image. Each color represents different object index.