Image segmentation - introduction

The goal of segmentation is to meaningfully simplify the representation of images so that we can group together points with certain properties into individual classes. The simplest methods take into account only the values of the points and not their positions. In that case, we are talking about **thresholding**. If we threshold into two classes (usually object(s) in foreground and background), the result is a binary image. Of course, we can threshold to several classes as well. For grayscale images we threshold based on the brightness value, for color images we can do it based on the color value. Also, thresholding methods can be improved by taking into account the positions of the points, so the thresholding based methods begin to merge with other segmentation methods.

The basic division of thresholding methods is as follows:

- classical methods(e.g.. K-means, OTSU, ...)
- methods using AI (so-called semantic thresholding)

According to the way thresholding is applied, we could divide the methods into:

- global thresholding
- local thresholding (here we also use information about the location of points)

When thresholding, it is often clear how to choose the threshold when looking at the histogram. For example, in Fig. 1 we see histograms that have clearly separated areas of values and thus we can reliably place the threshold values. The situation can easily get worse, when there is noise in the image, the scene is unevenly lit (brightness fluctuations), objects reflect light, etc. The result may be the overlap of histograms. An example of such a situation in the presence of noise is shown in Fig. 2.

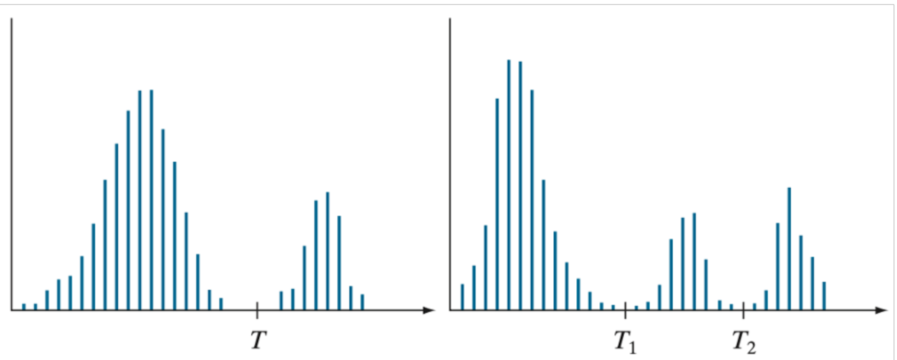

Fig. 1 Examples of image histograms, where it is trivial to choose the threshold value T, or multiple thresholds T_1, T_2

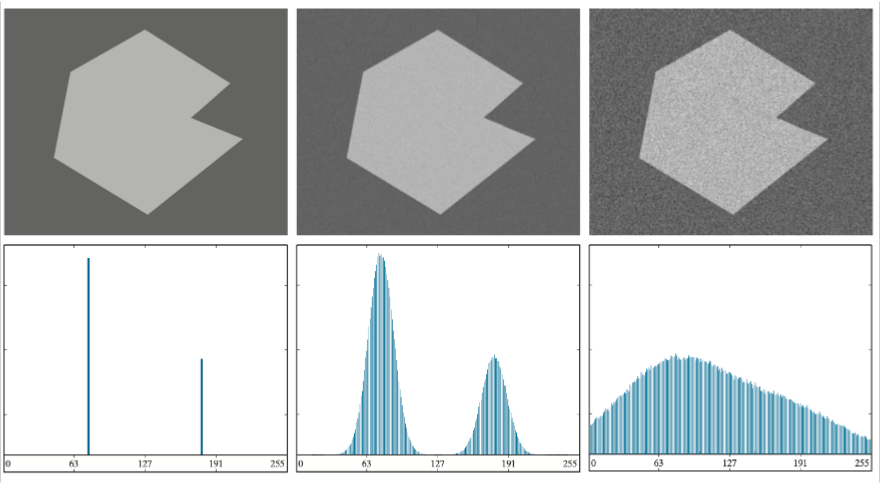

Fig. 2 Examples of images and their histograms, where we see how the histograms of the object and the background merge due to the presence of increasingly strong noise so the situation for the threshold choice gets worse

In the following parts of the text, there are presented the selected segmentation methods. For most methods, examples of their usage in the MATLAB environment are also given.

Information about segmentation in the MATLAB environment can be found in [1]. There are various interactive applications (e.g. colorThresholder, imageSegmenter, volumeSegmenter), a number of functions for performing segmentation (e.g. graythresh, multithresh), displaying and evaluating results, and many others. There are also many examples.

Information on image segmentation can be found in [2] and, together with a number of practical demonstrations in the MATLAB environment, in [3].

References

[1] Mathworks, Image Segmentation, online: `https://www.mathworks.com/help/images/image-segmentation.html`

[2] Gonzalez, R., C., Woods, E., W., Digital Image Processing, Global Edition, 4th edition, Pearson 2018, ISBN 10: 1-292-22304-9

[3] Gonzalez, R., C., Woods, E., W., Eddings, S., L., Digital Image Processing using MATLAB, Gatesmark Publishing, ISBN-10: 0-9820854-1-9