Segmentation

Introduction

Image segmentation is a method in which a digital image is broken into various subgroups called image segments, which help reduce the complexity of the image to make processing or analysis of the image simpler. In other words, segmentation involves assigning labels to pixels. All picture elements or pixels belonging to the same category have a common label assigned to them.

An example of different types of image segmentation. | Image: Mrinal Tyagi

Image segmentation consists in partitioning an image f according to a certain criterion. This means that the image is divided into regions R_i that are both mutually disjoint and collectively cover the entire image. Two pixels in the same region satisfy the criterion, but two pixels in two adjacent regions do not.

The figures below show several examples of segmentation.

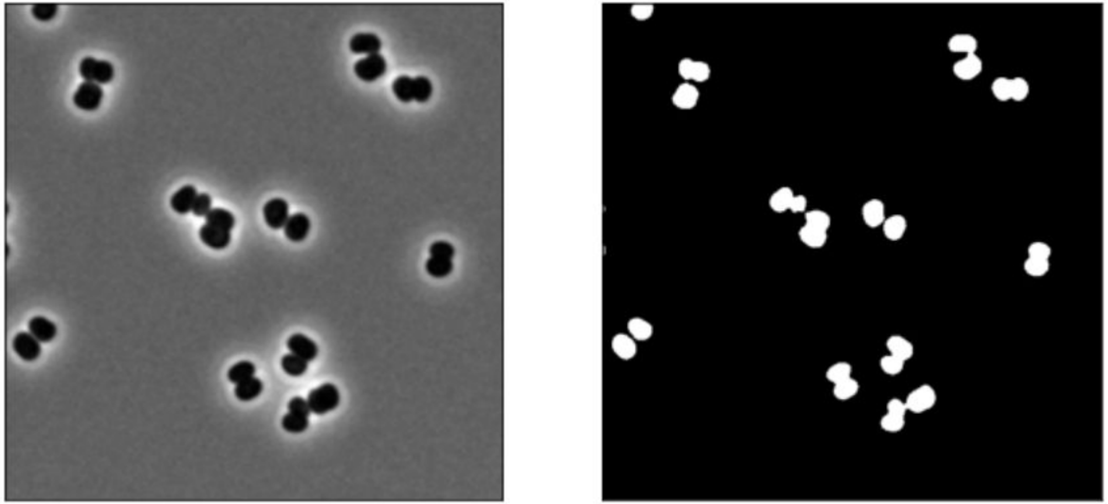


Fig. 48 Example of segmentation on gray levels (left: original image, right: segmentation given as regions).

Fig. 49 Example of segmentation on color (left: original image, right: segmentation given as regions, the colors of regions is the mean color of the pixels in the original image

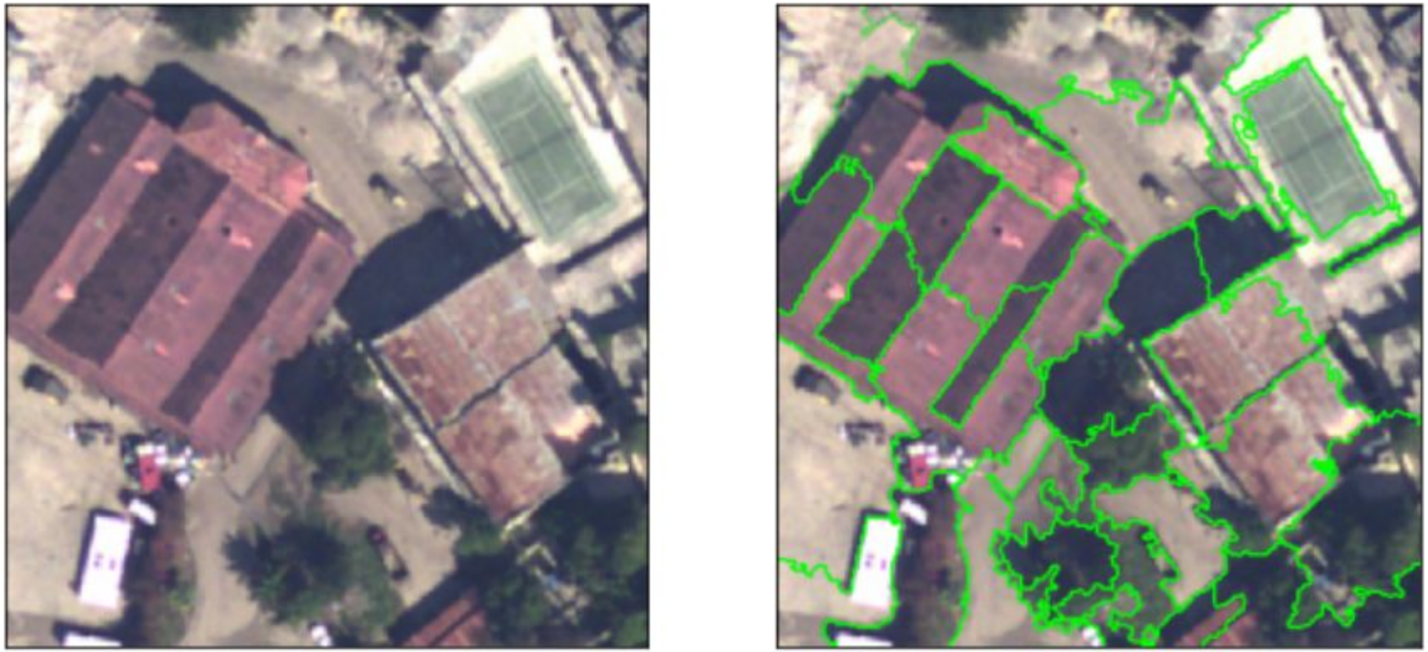


Fig. 50 Example of segmentation on color with a constraint of the region size
(left: original image, right: segmentation given as borders)

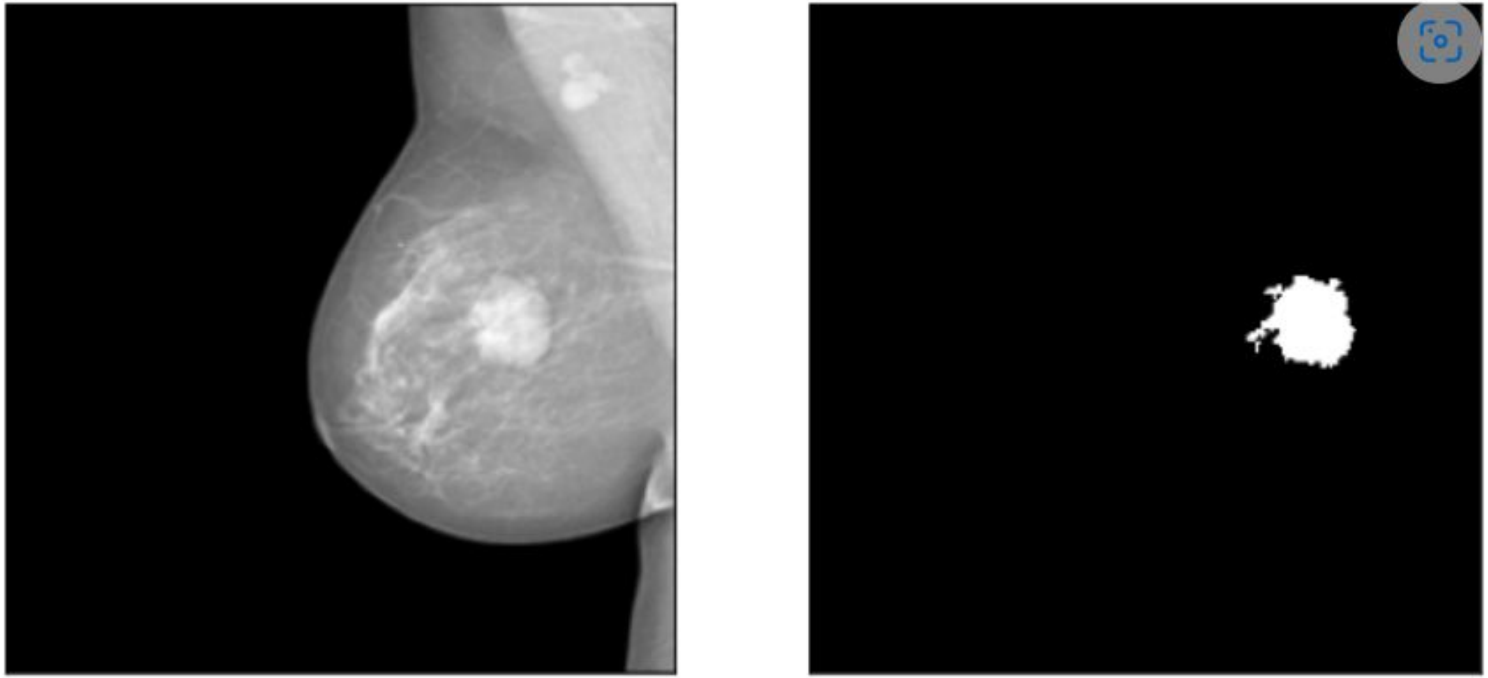


Fig. 51　Example of segmentation on color (left: original image, right: segmentation given as borders)

The result of segmentation is not unique: it depends on the criterion, the segmentation method, the initialization of the method, etc. There are a lot of diverse segmentation methods and this chapter focuses on the more usual ones. Finally, we talk about how to measure the quality of segmentation.

Histogram thresholding

Binary thresholding

A very simple method of segmentation consists in associating with each pixel of the image f a binary number which depends on the intensity of the pixels and on a threshold (French: *seuil*) T:

$$g(m,n) = \begin{cases} 1 & \text{if } f(m,n) \geqslant T, \\ 0 & \text{if } f(m,n) < T \end{cases}$$

This method is called “binarization” (French: *binarisation*). It gives a segmentation into two classes, depending on the intensity of the pixels of a grayscale image (Fig. 52).

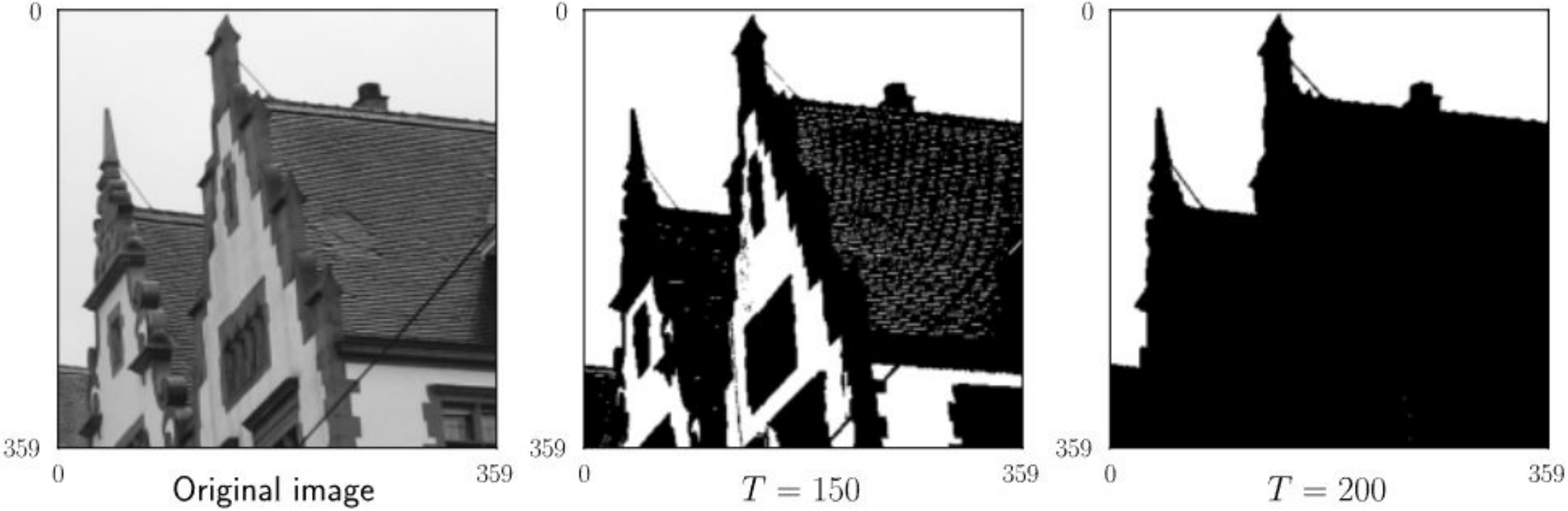


Fig. 51 Example of binarization for two thresholds

As you can see, the segmentation depends on the value of T. Therefore, the histogram is very useful to choose the threshold! As an example, Fig. 53 gives the histogram of the image, with the chosen thresholds.

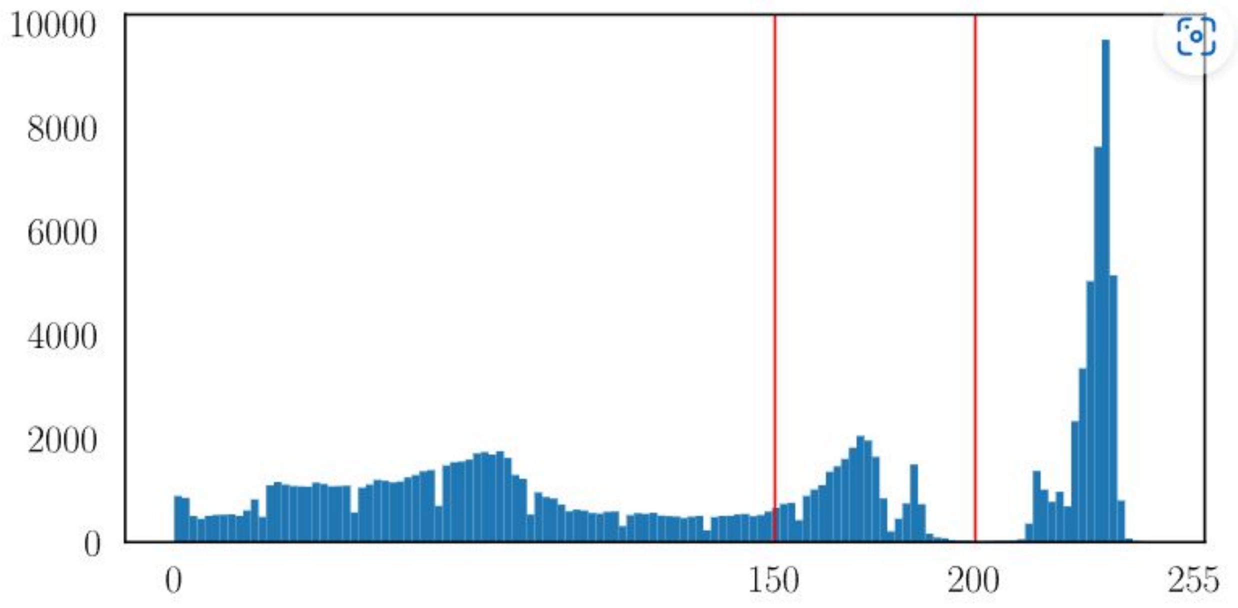


Fig. 53 Histogram of Fig. 52 with the two thresholds.

It would be useful to have an automatic process to define the threshold, whatever the image to segment. Otsu’s method is the most famous automatic method for histogram thresholding.

Otsu’s method

Binarization divides the histogram of the images in two groups, namely class 0 and class 1, as illustrated Fig. 54.

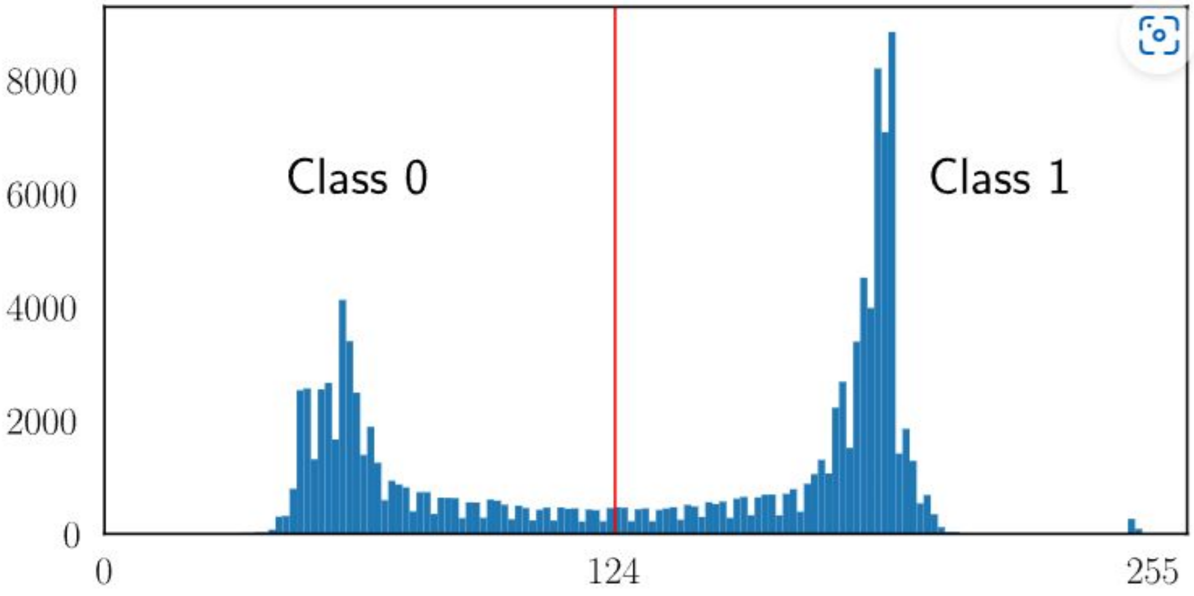


Fig. 54 An histogram and a threshold T.

Each group has a number of pixel $w_i(T)$ with mean $\mu_i(T)$ and variance $\sigma_i^2(T)$, where i is the group index (0 or 1). Otsu's method [Otsu 1979] computes the threshold T which minimizes the intra-class variance $\sigma_w^2(T)$ (also known as the within-class variance), defined as the weighted mean of the variances of each class:

$$\sigma_w^2(T) = w_0(T)\sigma_0^2(T) + w_1(T)\sigma_1^2(T).$$

Considering the intensities to lie in $\{0, \ldots, L-1\}$ and h the histogram, the variables are defined as below.

Class 1

$$w_0(T) = \sum_{i=0}^{T} h(i)$$

$$\mu_0(T) = \frac{1}{w_0(T)} \sum_{i=0}^{T} ih(i)$$

$$\sigma_0^2(T) = \frac{1}{w_0(T)} \sum_{i=0}^{T} (i - \mu_0(T))^2 h(i)$$

Class 2

$$w_1(T) = \sum_{i=T+1}^{L-1} h(i)$$

$$\mu_1(T) = \frac{1}{w_1(T)} \sum_{i=T+1}^{L-1} i h(i)$$

$$\sigma_1^2(T) = \frac{1}{w_1(T)} \sum_{i=T+1}^{L-1} (i - \mu_1(T))^2 h(i)$$

The algorithm to determine the value of T that minimize $\sigma_w^2(T)$ is simple: the intra-class variance $\sigma_w^2(T)$ is calculated for all the thresholds $T = \{0, \ldots, L-1\}$, and the one that gives the lowest value for $\sigma_w^2(T)$ is returned.

An example of Otsu's thresholding is given Fig. 55.

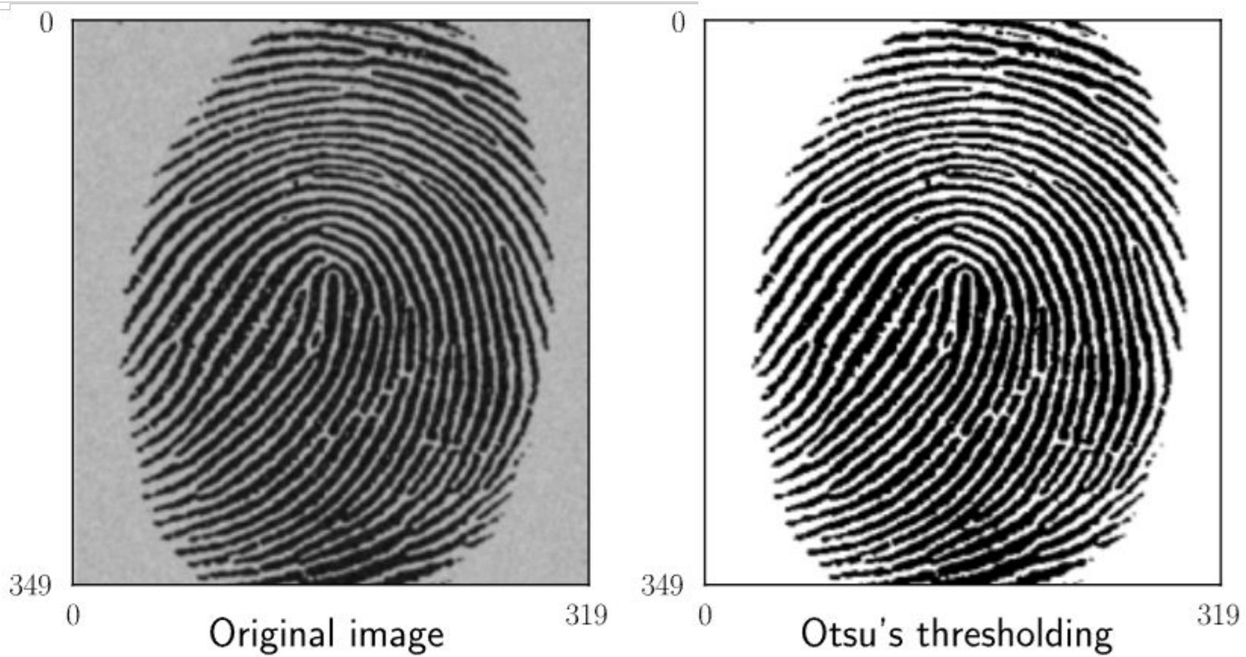


Fig. 55 Result of Otsu's segmentation.

Multiple Threshold

An image can be segmented in more than two classes by defining or computing several thresholds (see Fig. 56). In particular, Otsu's method can be extended to several thresholds, but the computational complexity (hence the computation time) increases greatly with the number of classes!

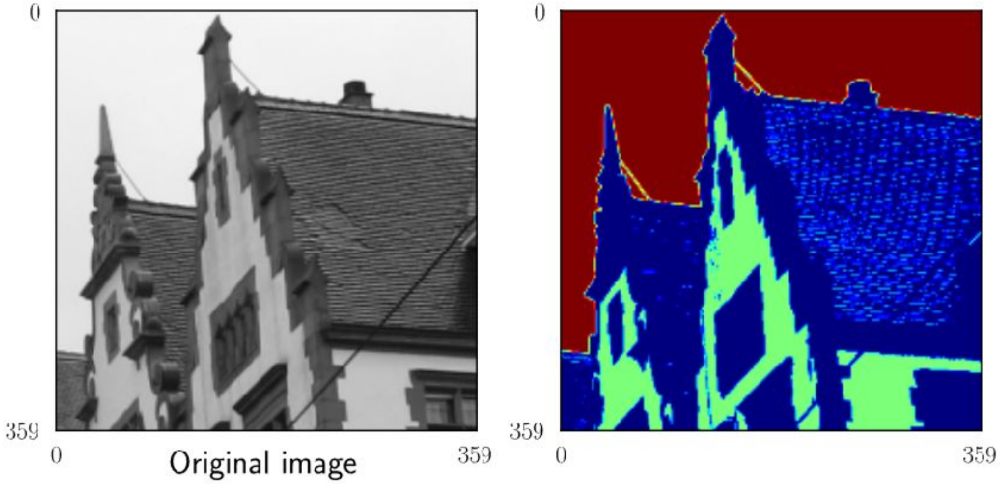


Fig. 56 Several thresholds are applied to an image to get several classes (shown in colors).

Clustering

Thresholding apply well on a grayscale image, for which it is easy to define a threshold from the modes of the histogram. However, this approach cannot be applied on a color or multiband image because there is no histogram. Each pixel in a B-band image can be represented by a point in a B-dimensional space. By doing so, pixels with similar colors form groups in the space, as illustrated in Fig. 57.

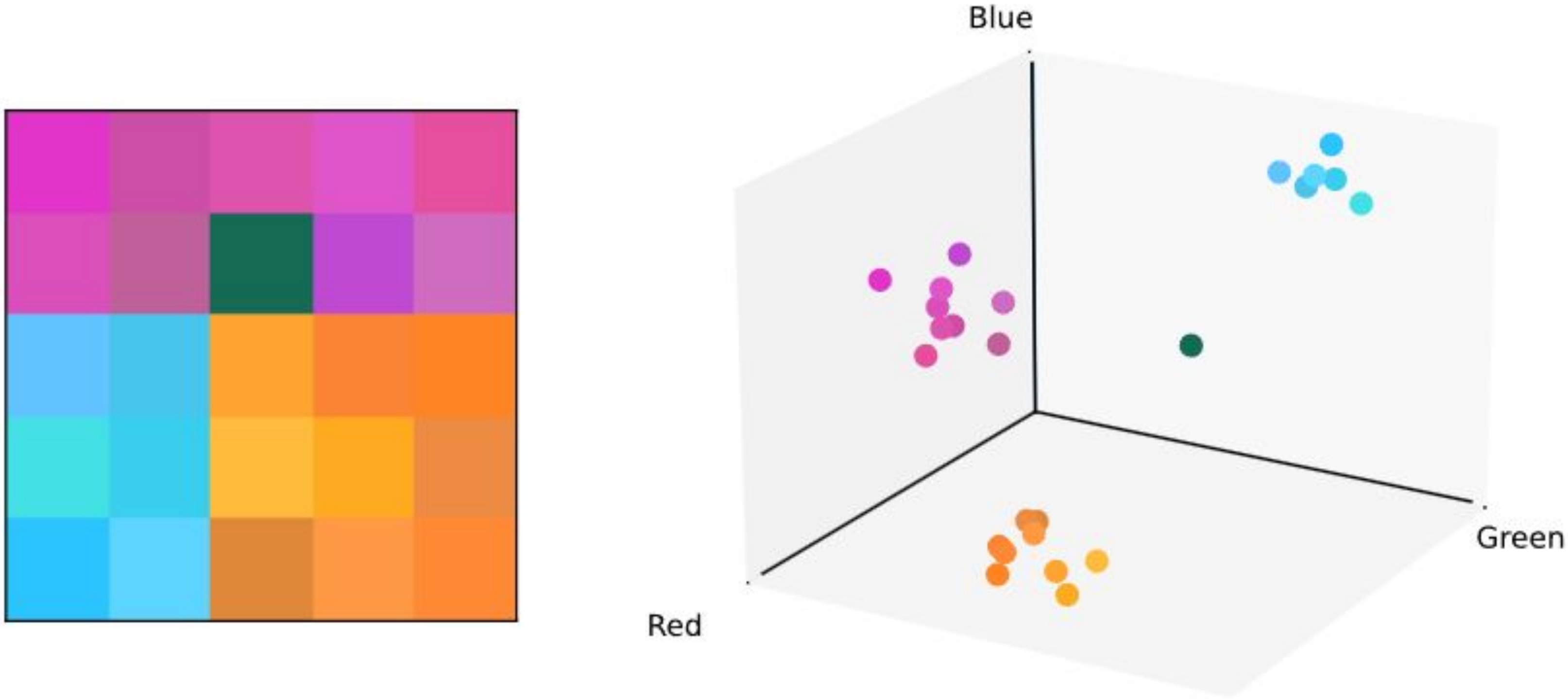


Fig. 57 Showing the pixels of a color image in a B-dimensional space. #

Clustering methods consists in defining groups of pixels. Therefore, all the pixels in the same group define a class in the segmented image.

A classical clustering method for image segmentation is the k-means method (French: *k-moyennes*).

In French, these methods are called *méthodes de classification*, although it would be more precise to call them *méthode de coalescence*.

The k-means algorithm [Steinhaus 1957, MacQueen 1967] is an iterative method that affects every point in the space $\mathbb{R}^B$ to a group (called cluster). The number K of groups is chosen by the user. In the sequel, the centroid defines the center of a group. Its coordinates are the mean of the coordinates of the points in the group.

The algorithm is given below.

Algorithm: K-means

1. Initialize (randomly) the K centroids
2. While the centroids vary:
    1. STEP A For each point:
        1. Calculate the distances from the point to all centroids
        2. Assign the point to the nearest group
    2. STEP B Calculate the centroid of each group

Fig. 58 illustrate this algorithm, in the simple case of an image with two bands (hence the two-dimensional space) segmented into $K = 2$ classes.

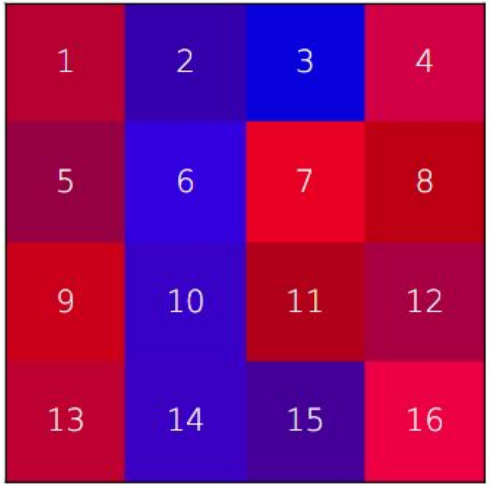
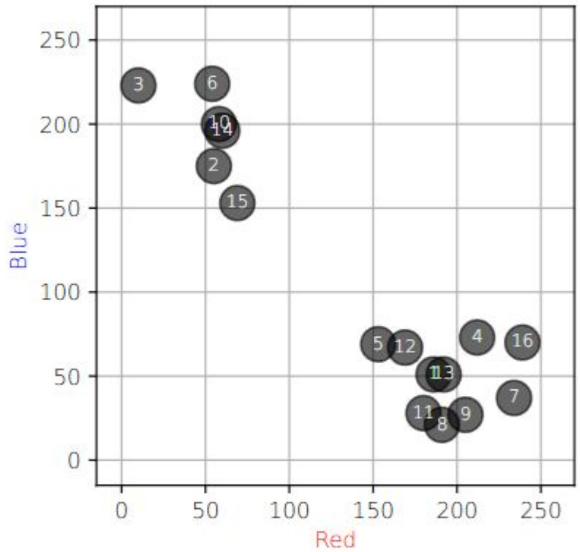


Image with bands red and blue. Numbers are the pixel indices.

186 51	55 175	10 223	212 73
153 69	54 224	234 37	191 21
205 27	58 200	180 28	169 67
192 51	60 196	69 153	239 70

Intensities (red and blue) of the pixels.



Pixels plotted in the (red,blue) space.

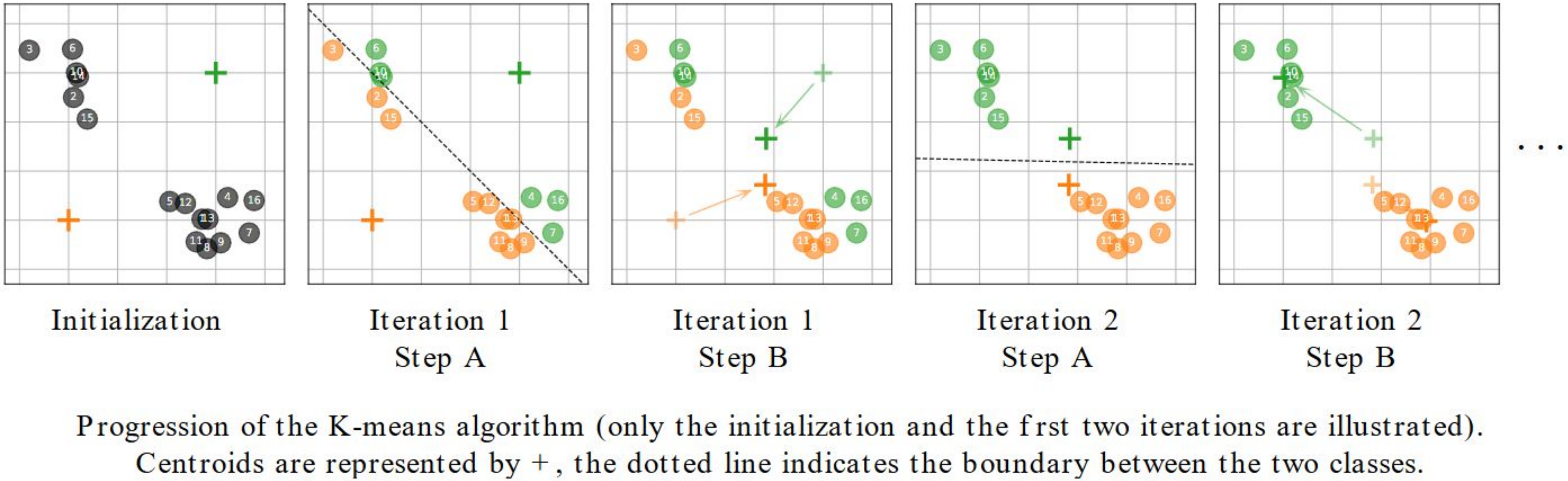


Fig. 58 Illustration of the k-means algorithm (click to enlarge). #

Fig. 59 gives the result of the k-means algorithm on an image.

Fig. 59 Segmentation with the k-means algorithm on the left image (center: $K = 2$ classes, right: $K = 4$ classes).

The pros and cons of the k-means method for image segmentation are listed below.

Advantages

- simplicity
- easy to implement
- generally fast
- works correctly when the clusters are spherical*

Disadvantages

- requires to know the number of classes
- sensitive to initialization
- can be slow in large dimensions
- fails for non-spherical structures*
- sensitive to outliers*

The characteristics above identified with * are now detailed.

Because the k-means algorithm performs the grouping with respect to the distance of the points to the centroids, it assumes that the groups are sphericals. Therefore, the algorithm works well for spherical clusters, but it fails if the clusters are not spherical, as depicted in the images below.

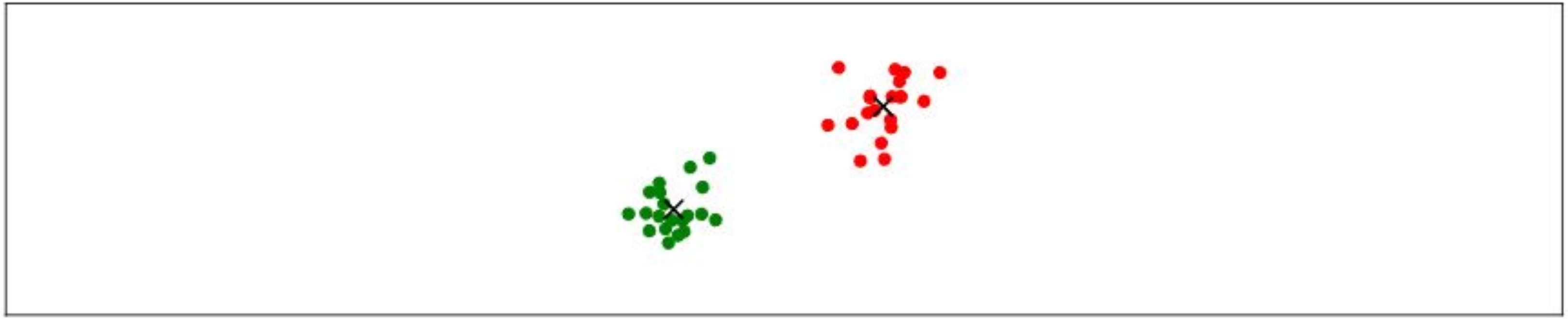


Fig. 60 Spherical clusters: the k-means algorithm works well. The points are depicted by ● whose color correspond to the class, and the centroids are depicted by a black ×. #

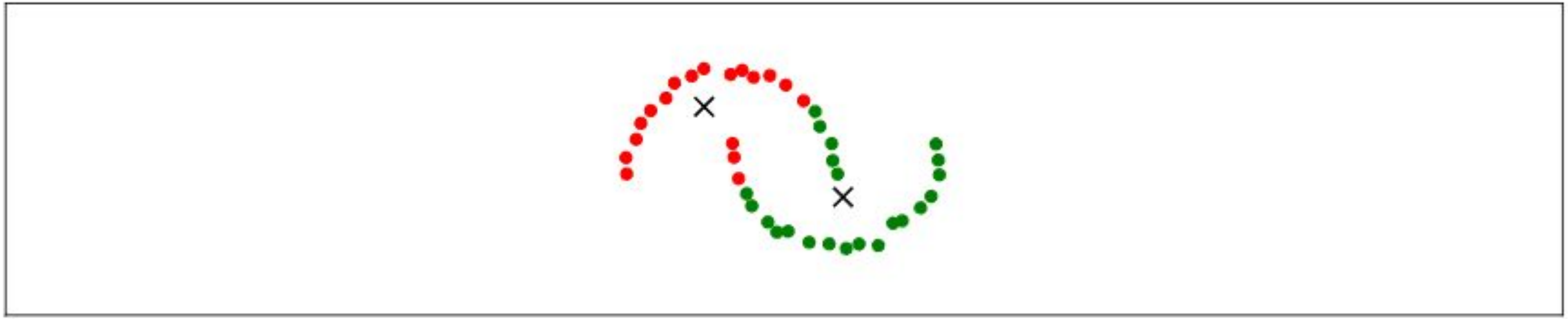


Fig. 61 Non-spherical clusters: the k-means algorithm fails.

In addition to this, the centroids are calculated as the mean of the points in the cluster. But the mean is not a robust estimation and is sensitive to points located far from the group. Thus, the algorithm may fail in the presence of outliers (*valeurs aberrantes*).

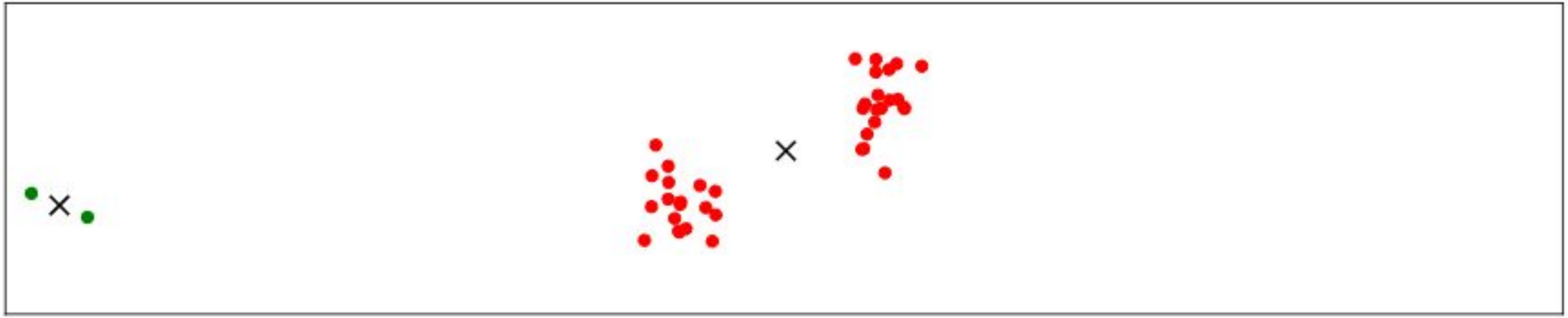


Fig. 62 Presence of outliers: the k-means algorithm fails in this example.

Other clustering methods are available to avoid some of the aforementionned limits. We can cite for example Gaussian mixture modeling or Mean-shift.

Region Growing

Logic behind region growing algorithm is **principle of similarity.**

Principle of similarity states that region is coherent if all pixels of that region are homogeneous.

Similarity of regions is used as the main segmentation criterion in region growing. (choice of criteria affects segmentation results dramatically!)

How to evaluate segmentation?

This chapter has introduced the usual methods of segmentation, but there are many more! No method is the best because the result depends, among other things, on the image itself. Consequently, it is interesting to evaluate, for the type of image to process, the quality of the segmentation. To do that, different criteria can be used, defined below. In addition to the image to be segmented, we also need the expected result, which we call “ground truth” (French: *vérité terrain*).

Imagine that the ground truth and the segmentation are the images in Fig. 63 (binary segmentation).

Fig. 63 Ground truth f^* (left) and segmentation f (right).

Each image has two areas: the segmented object (shown in white) and the background (in black). So, we can define four types of zones (cf. Fig. 64):

In French, we talk with *vrai positifs*, *vrai négatifs*, *faux positifs* and *faux négatifs*.

- the true positives (TP) represent the pixels considered as being in the object and being really in the object,
- conversely, the true negatives (TN) are the pixels outside the object both in the segmentation and the ground truth,
- the false positive (FP) are the pixels considered by the segmentation in the object, but which in reality are not part of it,
- finally, the false negative (FN) are the pixels of the object that the segmentation has classified outside.

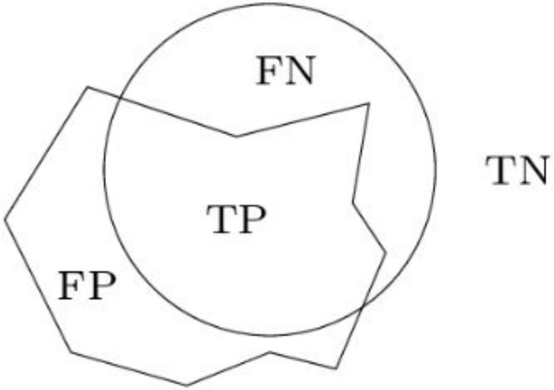


Fig. 64 Definition of true positive (TP), false positive (FP), true negative (VN) and false negative (FN).

From these four quantities, one or the other of the criteria below can be used.

Sensibility (*sensibilité*)

$$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$$

Specificity (*spécificité*)

$$\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}$$

Dice coefficient (*coefficient de Dice*)

$$\frac{2\,\mathrm{TP}}{2\,\mathrm{TP}+\mathrm{FP}+\mathrm{FN}} = \frac{2\,|f\cap f^*|}{|f|+|f^*|}$$

Jaccard coefficient (*coefficient de Jaccard*)

$$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}} = \frac{|f\cap f^*|}{|f\cup f^*|}$$

References

- J.B. MacQueen, “Some Methods for classification and Analysis of Multivariate Observations”, *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, p. 281–297, 1967
- N. Otsu, “A threshold selection method from gray-level histograms”, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no 1, p. 62-66, 1979.
- H. Steinhaus, “Sur la division des corps matériels en parties”, *Bull. Acad. Polon. Sci.*, vol. 4, no 12, p. 801-804, 1957.