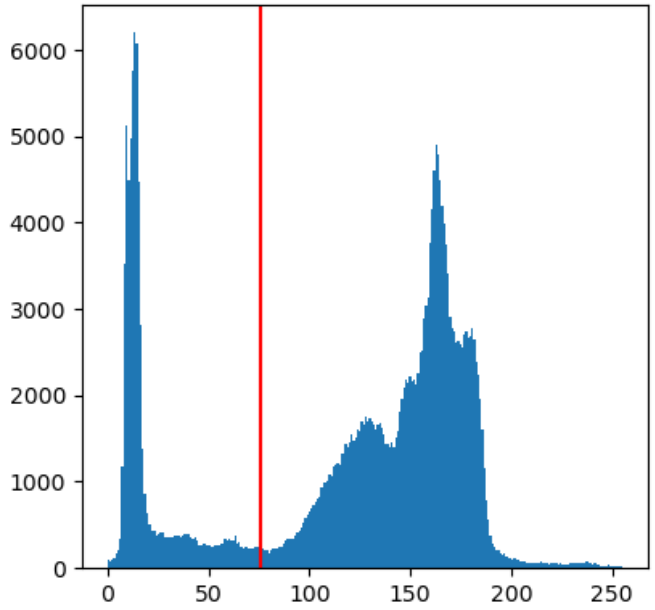
Segmentation by Thresholding

- Use a threshold to separate a whole image into two parts.
- Thresholding is a method of separating pixels into "classes" based on their values.
- The threshold is usually determined from the histogram. For example, for binary segmentation, we can choose the half-way point between the two main peaks.

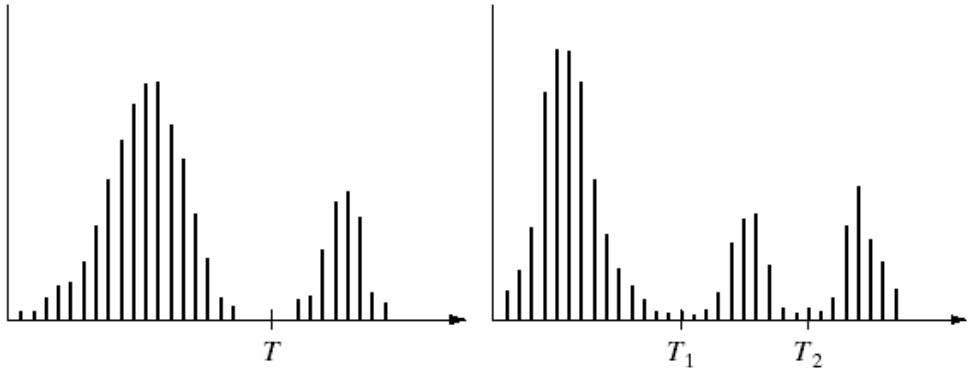


https://scikit-image.org/docs/0.14.x/auto_examples/xx_applications/plot_thresholding.html

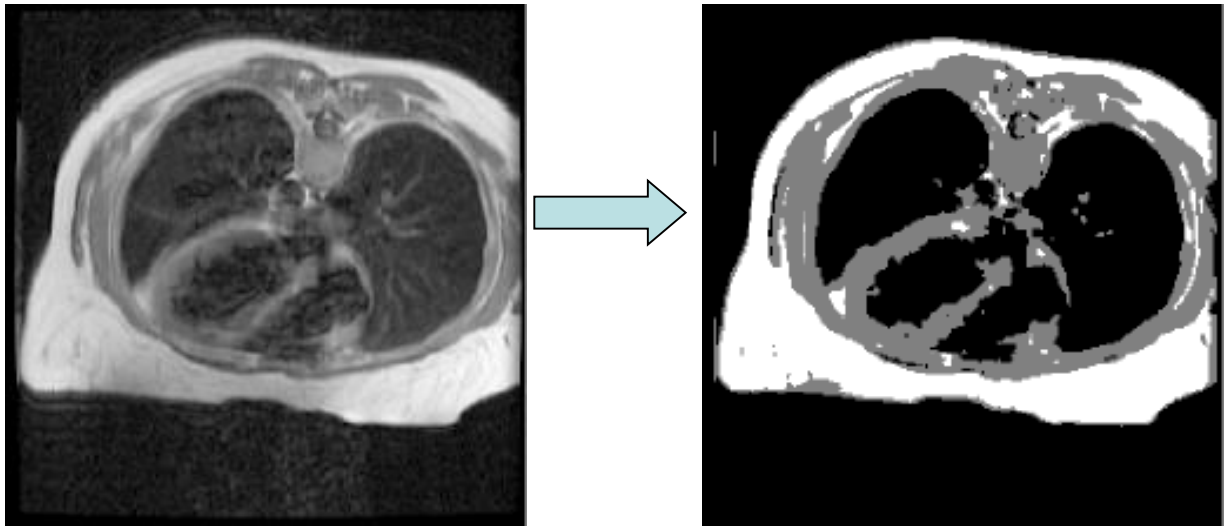
Segmentation by Thresholding

- When appropriate, multiple thresholds can be used to separate the image into more than two parts.

Histograms with two and three peaks:



Example (3 peaks):



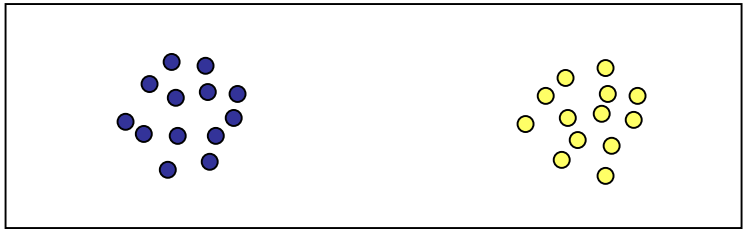
Automatic Selection of Thresholds

- (Basic Global Algorithm) in textbook
- Start with a threshold T that is the average gray level of the image.
- In each iteration:
  - Compute m_1 and m_2, the average gray levels for those pixels with gray levels above and below T, respectively.
  - Replace T with $(m_1+m_2)/2$.
  - When the change of T between iterations is below some small predefined tolerance, terminate the iteration.
- This version is for single-threshold (two-region) cases. It can also be extended to more thresholds.
- This is a special case of clustering based segmentation, to be discussed later.

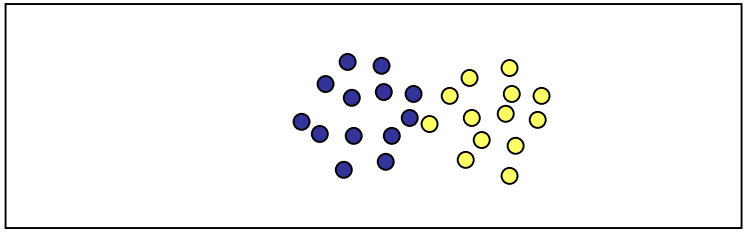
Otsu's Method

The concept of the **separability** between two distributions:

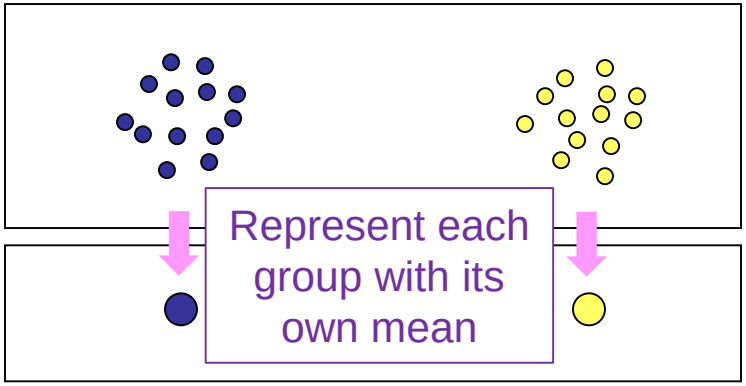
Good separability:



Not-so-good separability



One way to quantify separability: σ_B^2 / σ_G^2



σ_G^2 (global variance)

σ_B^2 (between-group variance)

Otsu's Method: Select a threshold that, when used to separate the pixel values into two groups, gives the maximum separability.

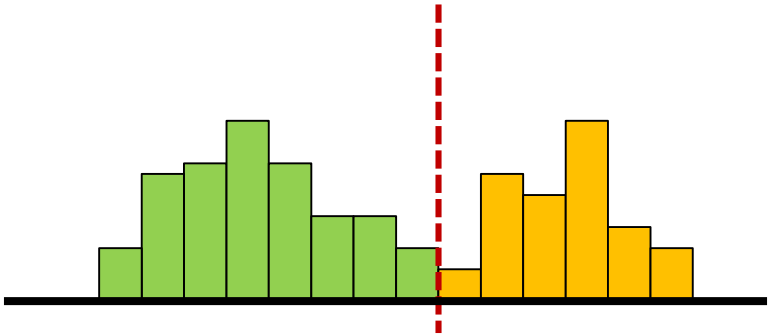
Otsu's Method

Otsu's Method automatically select a threshold to maximize separability:

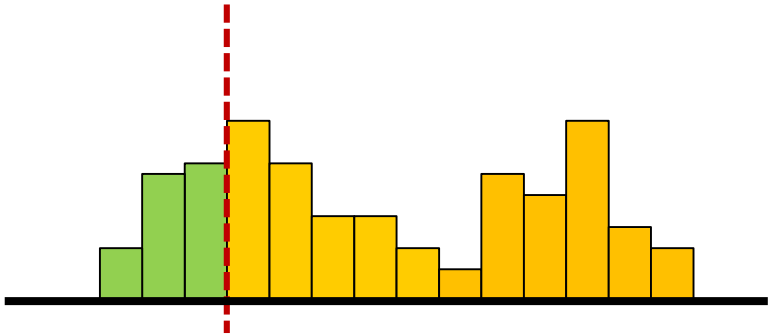
- For each possible threshold, compute the gray-level separability of the two regions:

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_G^2} = \frac{P_1(k)[m_1(k) - m_G]^2 + P_2(k)[m_2(k) - m_G]^2}{\sigma_G^2}$$

- Select the threshold level (k) that maximizes the separability.

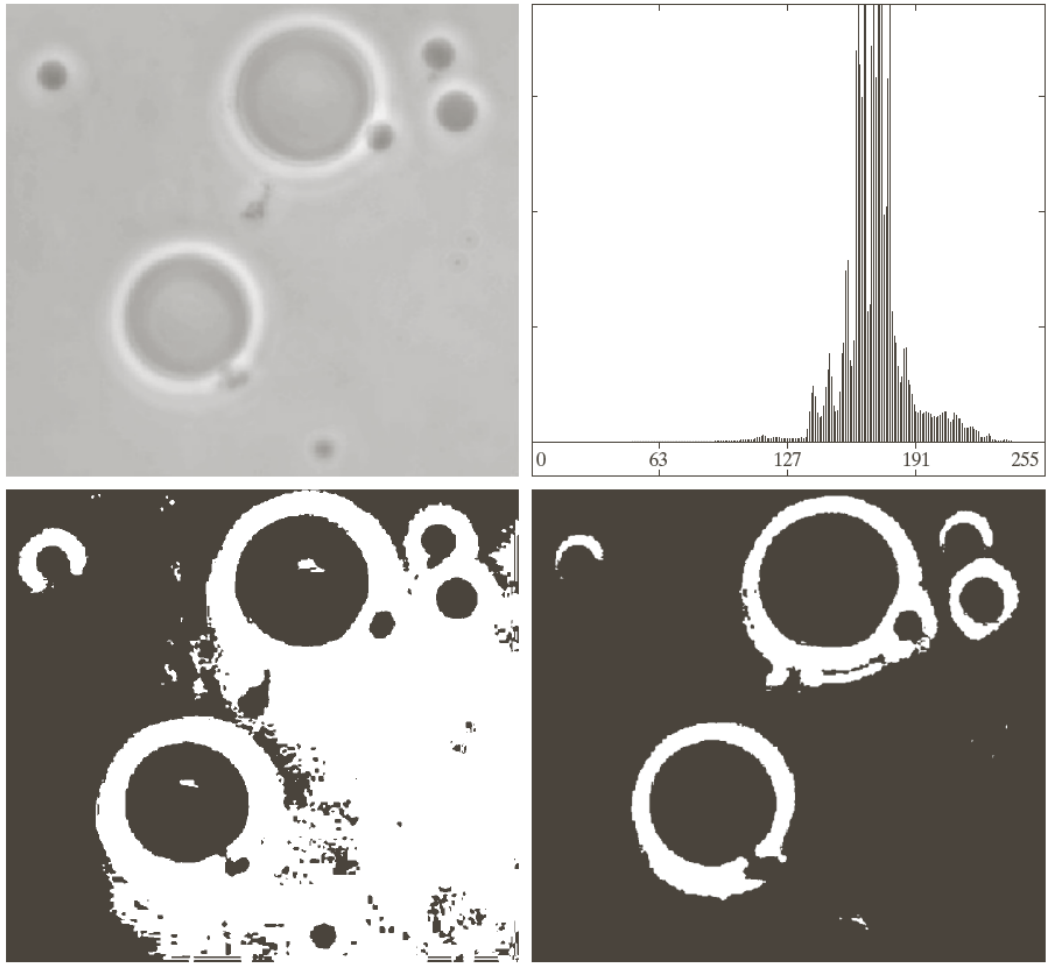


Good separability:



Not-so-good separability

Automatic Selection of Thresholds

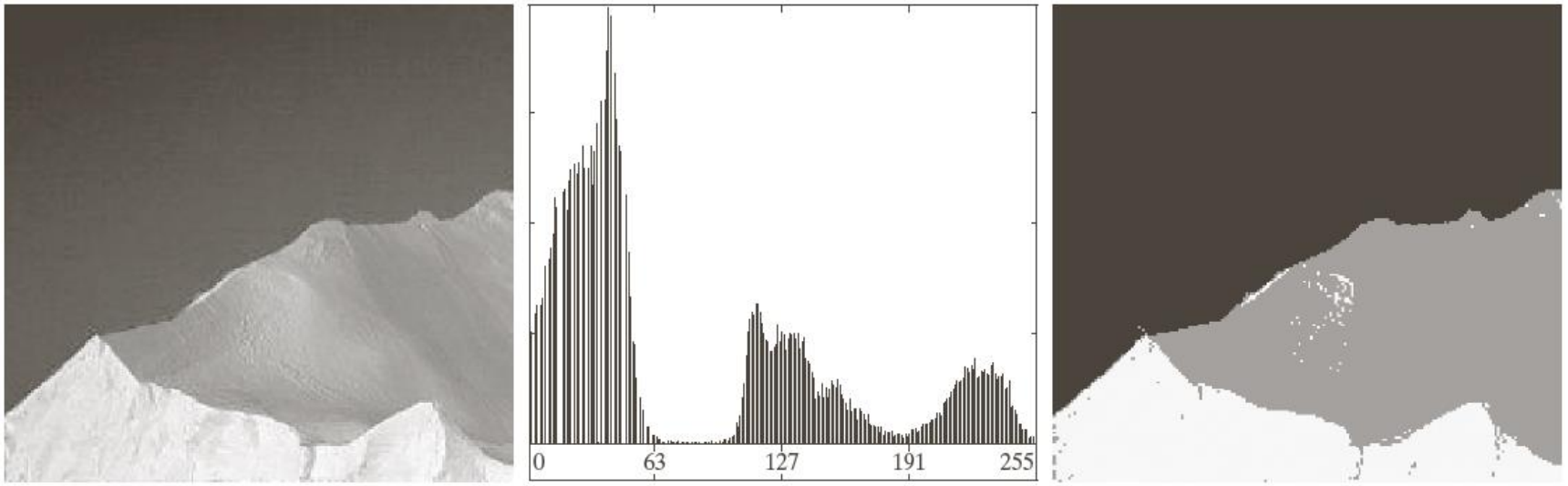


Basic global algorithm

Otsu's method

Automatic Selection of Thresholds

Otsu's method with two thresholds (3 regions):

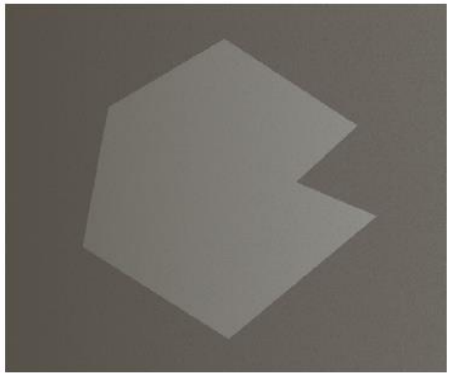


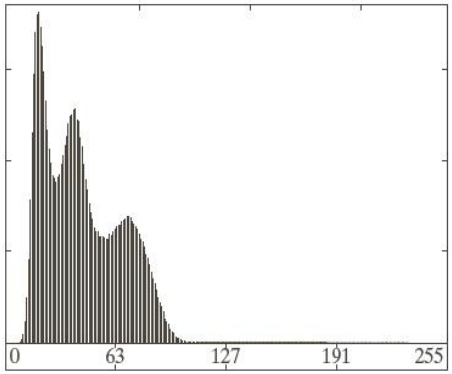
$$\eta(k_1,k_2)=\frac{\sigma_B^2(k_1,k_2)}{\sigma_G^2}=\frac{\sum_{j=1}^{3}P_j(k_1,k_2)[m_j(k_1,k_2)-m_G]^2}{\sigma_G^2}$$

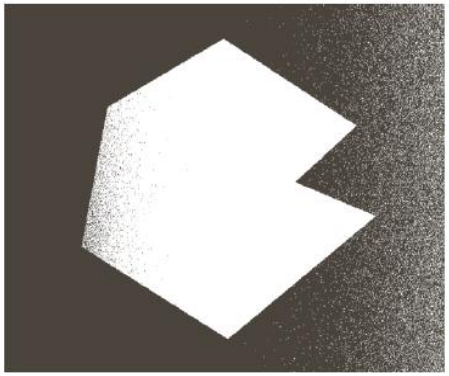
Adaptive Thresholding

In some situations, thresholds need to be selected locally, not globally. The most common cause is the variation of illumination.

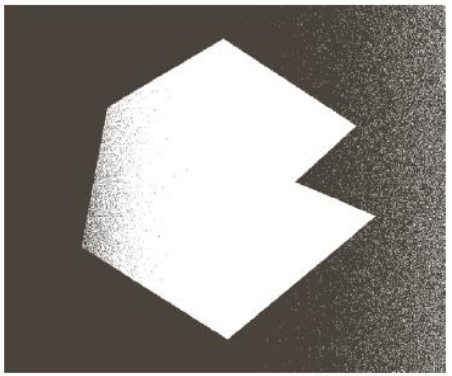
An example of segmentation error that results from global thresholding:







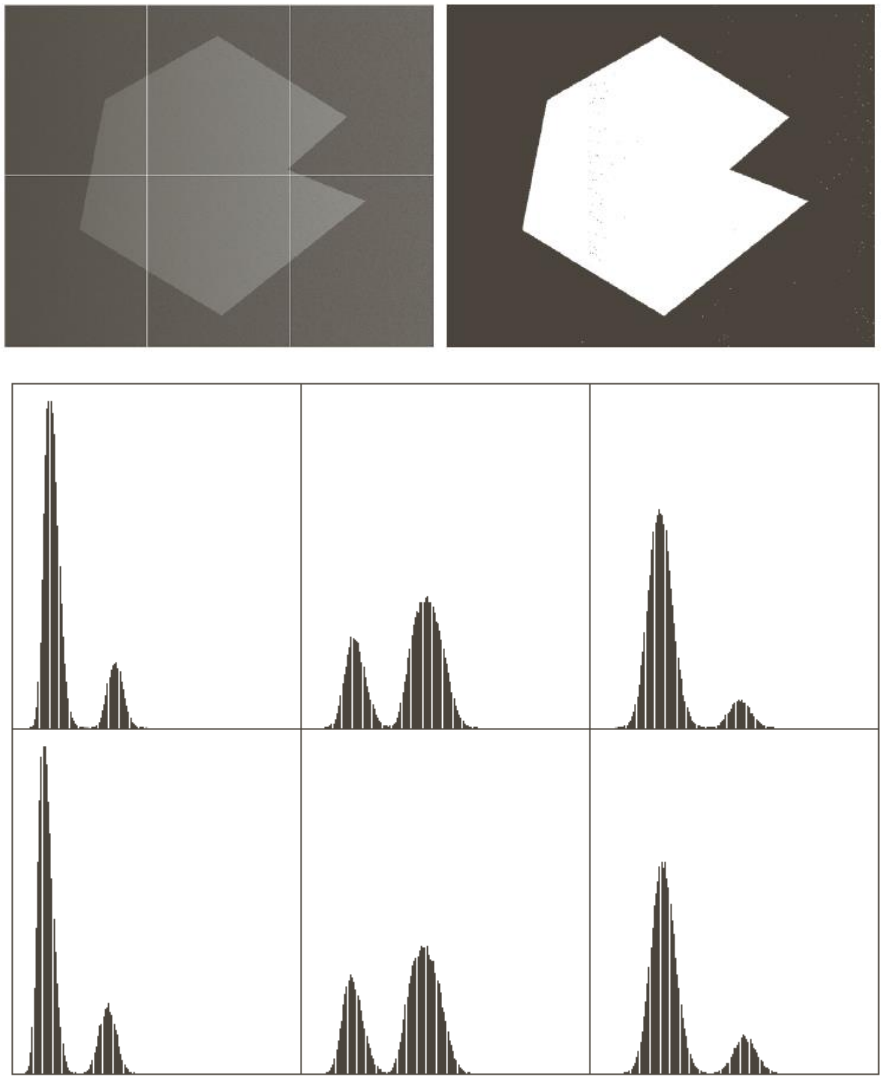
Basic global algorithm



Otsu's method

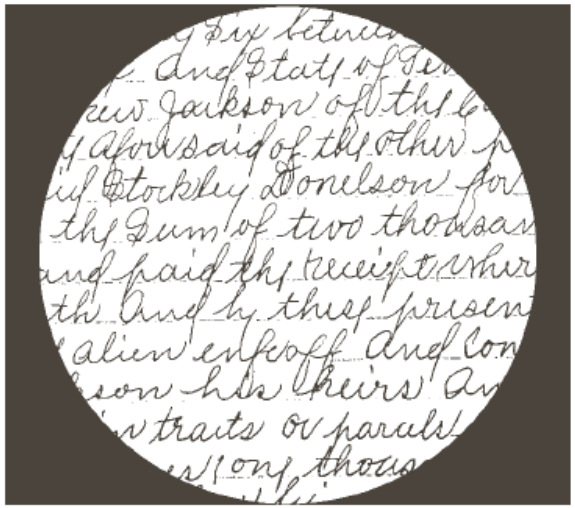
Adaptive Thresholding

The same image, with different thresholds for the 6 regions:

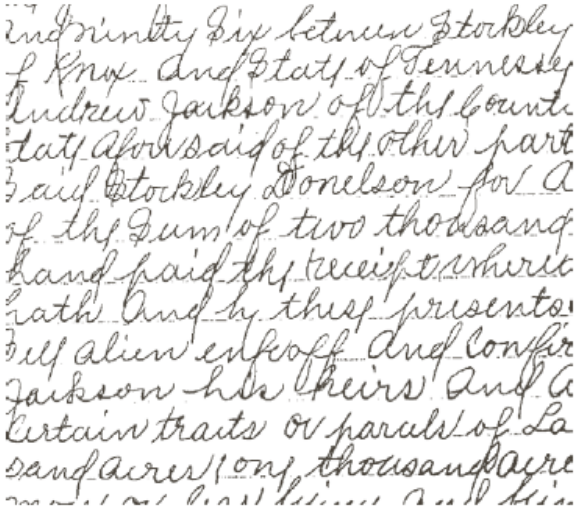


Adaptive Thresholding

global threshold



local threshold



Thresholding by Similarity

Example with skin color segmentation:

Model of Caucasians

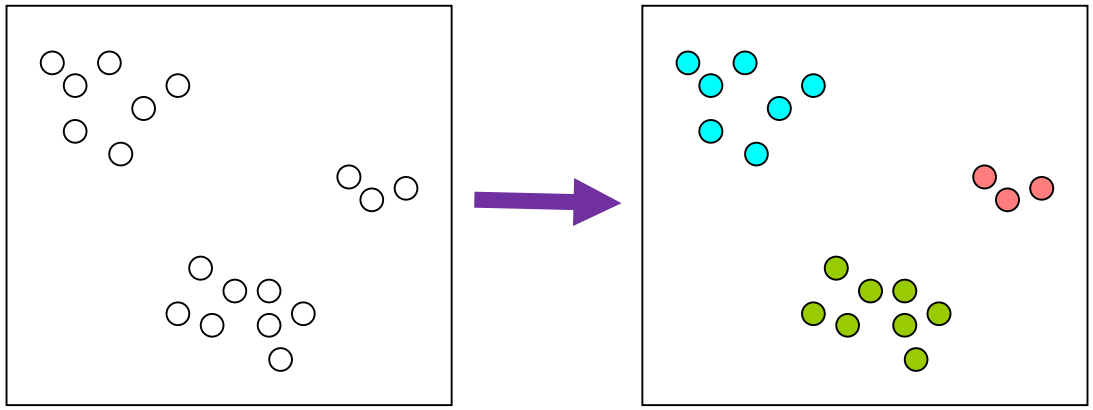
similarity

Model of blacks

thresholding

Segmentation by Clustering

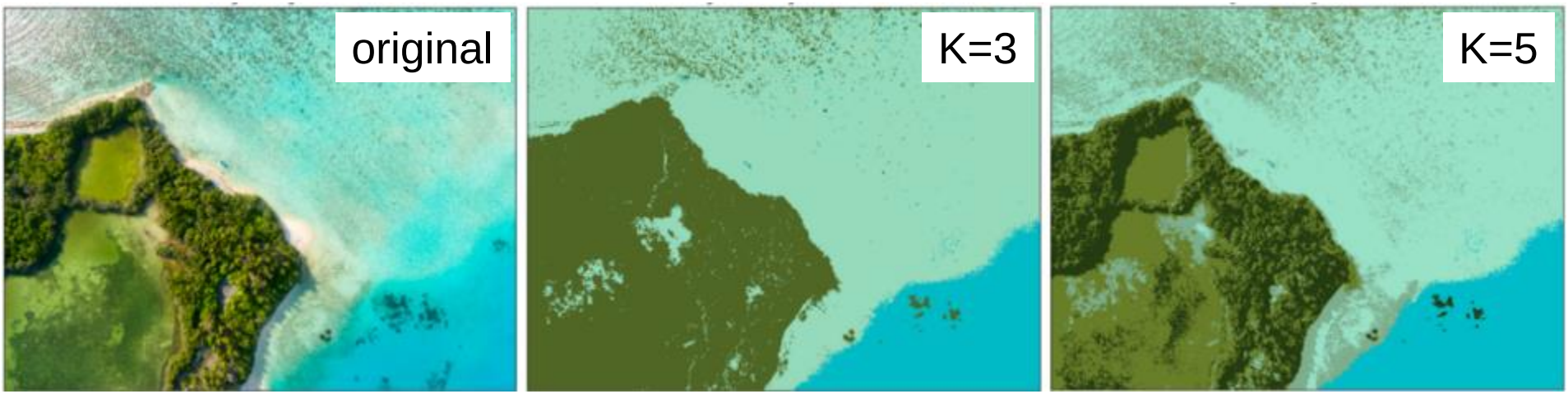
- Clustering is a class of algorithms that group unorganized data into meaningful “clusters”.



- In image segmentation, we can consider the pixels as units of data, and group them into clusters using one of the many available clustering algorithms.
- Each cluster consists of a set of clusters.
- We will show some examples with minimal details about the various clustering algorithms.

Segmentation by Clustering

- **K-means clustering**: A very common type of clustering algorithms with a predefined number (K) of clusters.



- The result images are shown with the mean colors of the clusters.
- The “basic global algorithm” for automatic threshold selection is just 2-means clustering with gray-scale images.

https://www.kdnuggets.com/2019/08/introduction-image-segmentation-k-means-clustering.html

Segmentation by Clustering

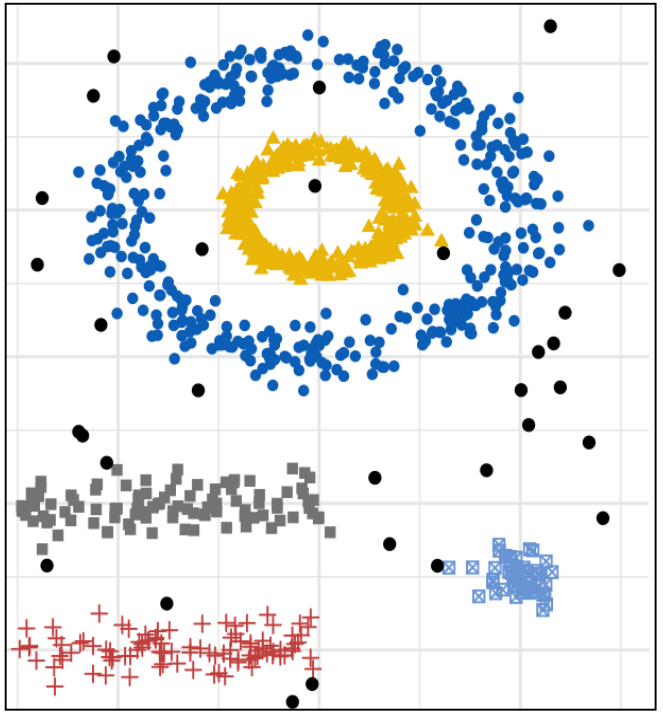
- K-means clustering (more examples):

- Related clustering algorithms with mixed (overlapped) clusters: expectation-maximization, and fuzzy c-means.

https://www.kdnuggets.com/2019/08/introduction-image-segmentation-k-means-clustering.html

Segmentation by Clustering

- **DBSCAN**: A popular density-based clustering algorithm that does not require a predefined number of clusters.

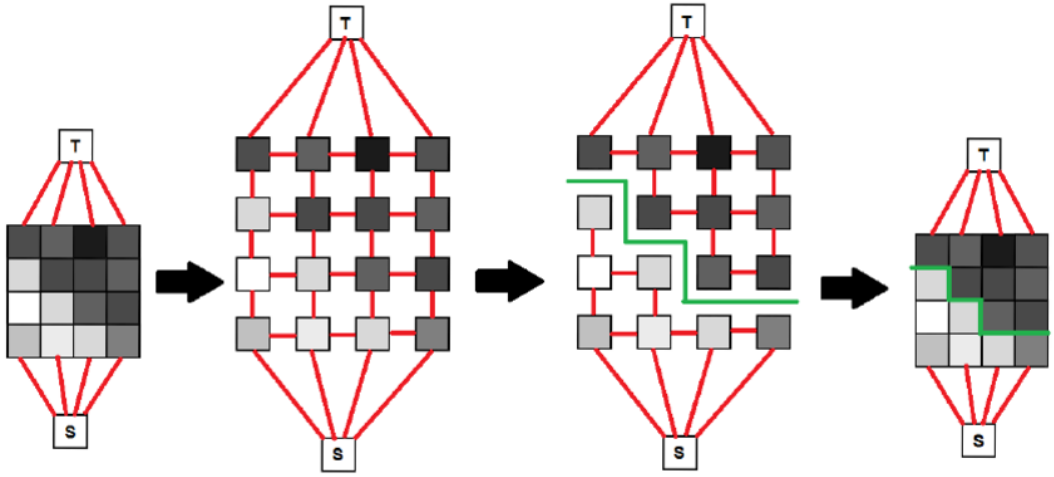


www.datanovia.com

https://www.kdnuggets.com/2019/08/introduction-image-segmentation-k-means-clustering.html

Segmentation by Clustering

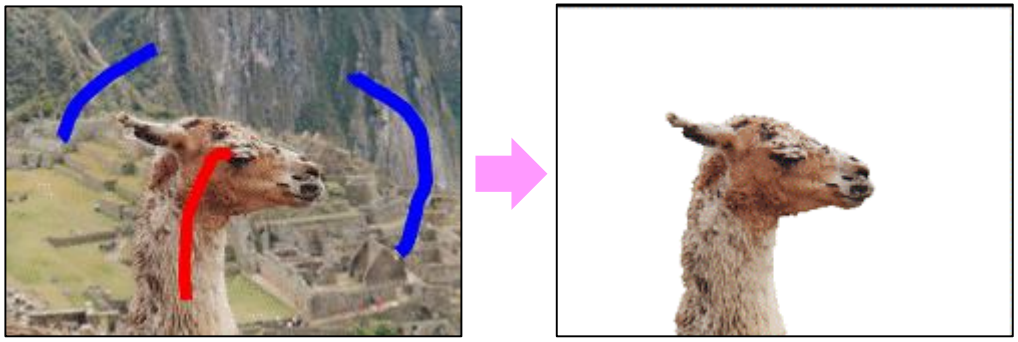
- Graph-cut based image segmentation:
  - Treat pixels as vertices in a graph.
  - One edge per pair of adjacent pixels.
  - Edge weight = Pixel similarity
  - Find the min-cut of the graph.
  - Popular for interactive segmentation with manually assigned sources and sinks.



https://www.researchgate.net/figure/Using-the-max-flow-min-cut-approach-for-image-segmentation-a-Define-set-of-pixels-in_fig5_292335453

Segmentation by Clustering

- Graph-cut based image segmentation: Some examples:

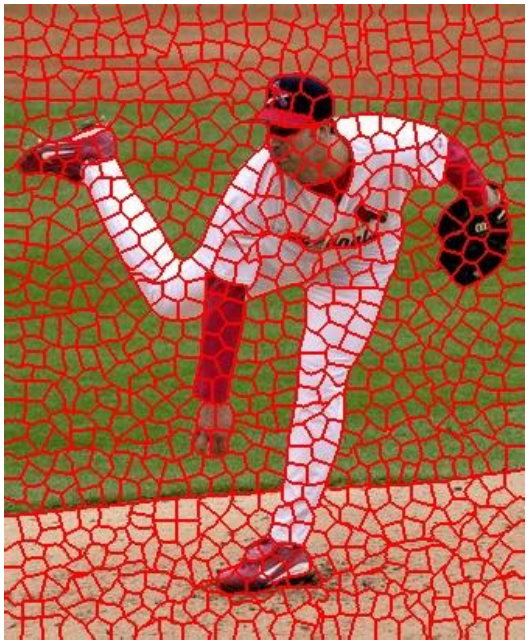


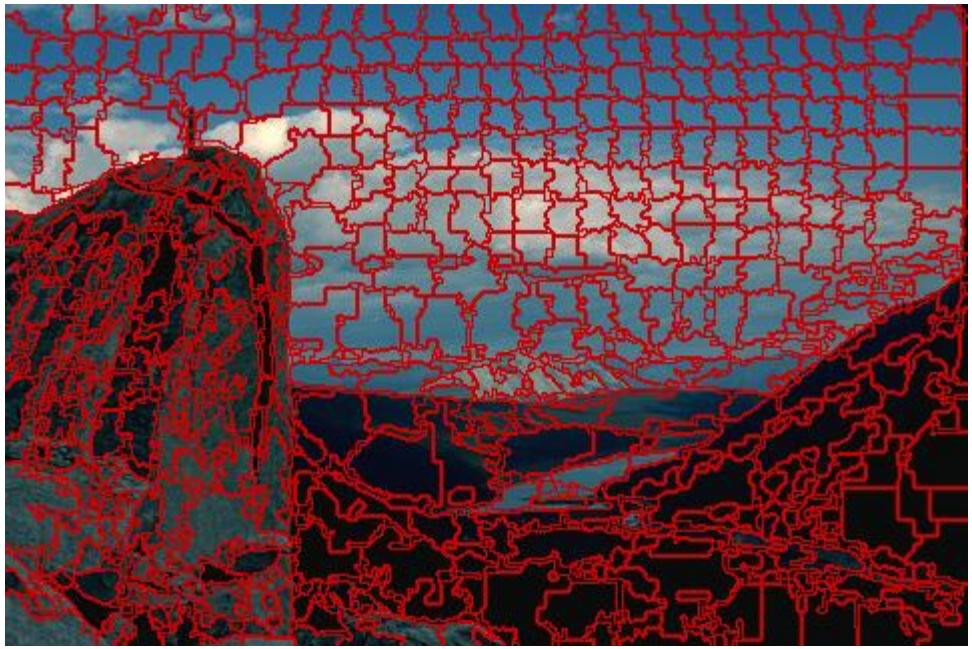
https://www.datasciencecentral.com/profiles/blogs/interactive-image-segmentation-with-graph-cut-in-python

https://filebox.ece.vt.edu/~jbhuang/teaching/ece5554-4554/fa16/hw4.html

Superpixels

- With so many pixels in an image, the segmentation algorithms naturally take a lot of computation.
- **Superpixels** is a representation of image with the unit being homogeneous small regions.
- Superpixels are generated with clustering algorithms.





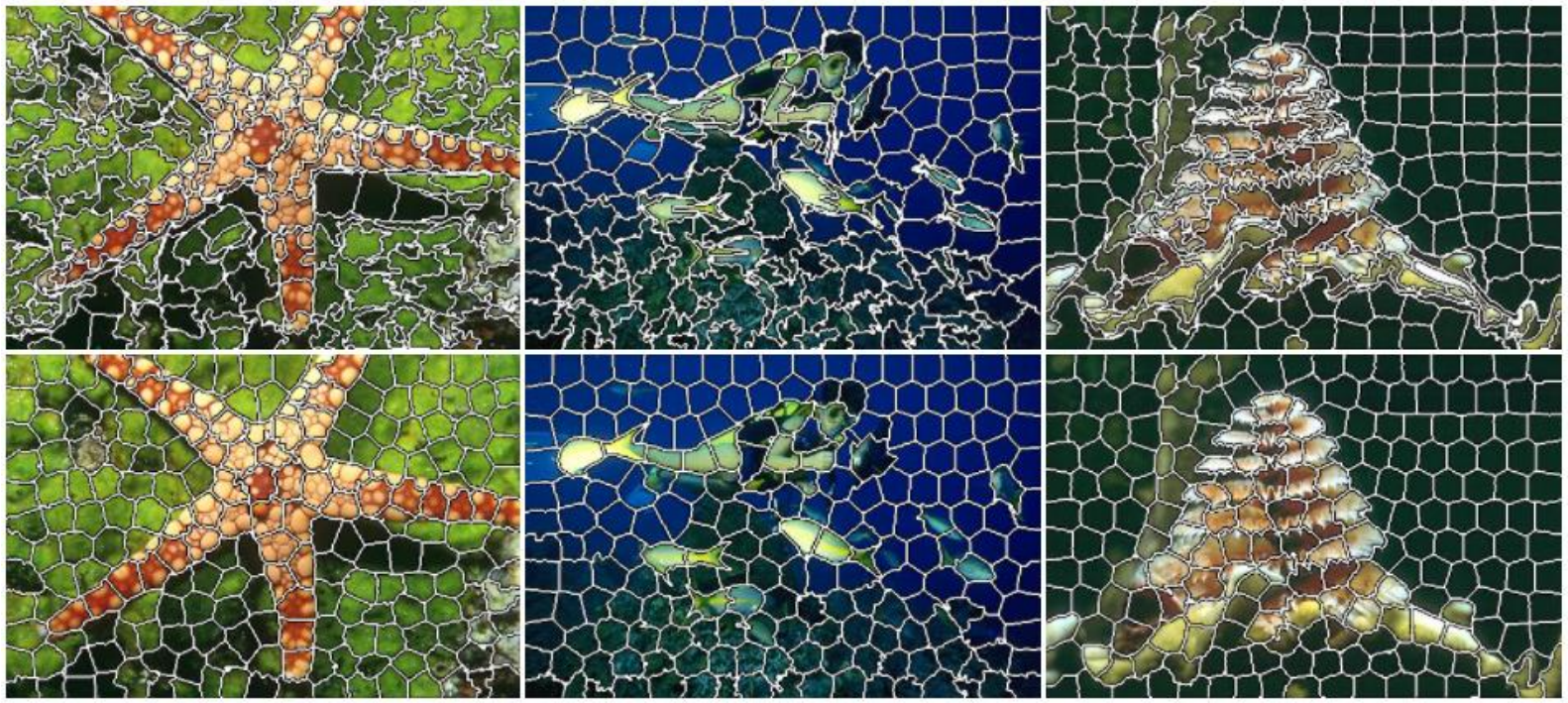
Superpixels

- Since each superpixel needs to be a connected set of pixels, distances among pixels are considered in the clustering process (in addition to pixel values).
- Two main factors affecting superpixel results:
  - Relative importance of pixel values and locations.
  - Parameter controlling the size of number of superpixels.
- Superpixels usually result in over-segmentation of the image. Postprocessing can then be applied to the superpixels (such as superpixel merging, or interactive graph cut, etc.).

https://www.epfl.ch/labs/ivrl/research/slic-superpixels/

SLIC Superpixels

- **SLIC** is probably the most well-known superpixel algorithms.
- SLIC is based on k-means, with k being the number of superpixels.
- Each pixel is defined by five values: three for color (usually RGB) and two for location.



https://www.epfl.ch/labs/ivrl/research/slic-superpixels/