Image segmentation

Thierry Pécot
Associate Researcher
CZI Imaging Scientist

Outline

Intensity thresholding

- **Estimate** an **intensity threshold** such that all pixels with an intensity **superior** (*resp.* **inferior**) to the threshold are part of the **object** component (*resp.* **background** component)

Intensity thresholding

- **Estimate** an **intensity threshold** such that all pixels with an intensity **superior** (*resp.* **inferior**) to the threshold are part of the **object** component (*resp.* **background** component)
- Such methods are based on the **histogram distribution**

Intensity thresholding

- **Estimate** an **intensity threshold** such that all pixels with an intensity **superior** (*resp.* **inferior**) to the threshold are part of the **object** component (*resp.* **background** component)
- Such methods are based on the **histogram distribution**
- **Global** and **local** thresholding methods

Intensity thresholding

- **Estimate** an **intensity threshold** such that all pixels with an intensity **superior** (*resp.* **inferior**) to the threshold are part of the **object** component (*resp.* **background** component)
- Such methods are based on the **histogram distribution**
- **Global** and **local** thresholding methods
- First example: the **global mean thresholding method**:

$$T = \frac{\sum_{i \in \Omega} I_i}{|\Omega|}, \tag{1}$$

where Ω is the set of pixels in the image and I_i is the intensity observed at pixel $i \in \Omega$

Intensity thresholding

- **Estimate** an **intensity threshold** such that all pixels with an intensity **superior** (*resp.* **inferior**) to the threshold are part of the **object** component (*resp.* **background** component)
- Such methods are based on the **histogram distribution**
- **Global** and **local** thresholding methods
- First example: the **global mean thresholding method**:

$$T = \frac{\sum_{i \in \Omega} I_i}{|\Omega|}, \tag{1}$$

  where Ω is the set of pixels in the image and I_i is the intensity observed at pixel $i \in \Omega$
- Second example: the **local mean thresholding method**:

$$T_s = \frac{\sum_{i \in \Omega_s} I_i}{|\Omega_s|}, \tag{2}$$

  where T_s is the local threshold estimated at pixel s, Ω_s is a a local neighborhood defined around pixel s and I_i is the intensity observed at pixel $i \in \Omega_s$

Intensity thresholding

Advantages:

- **No parameters** for global thresholding (but at least **local neighborhood size** for local thresholding)
- **Many global** and **local** thresholding **methods** exist in Fiji

Drawbacks:

- Sensitivity to **noise**
- Sensitivity to **intensity variations**

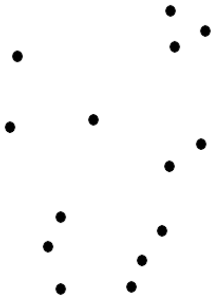
Outline

Image clustering

Image clustering: **classifying** or **grouping** objects based on intensity

Image clustering

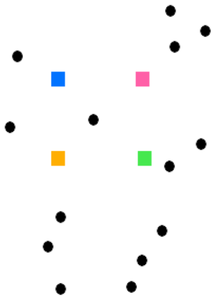
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

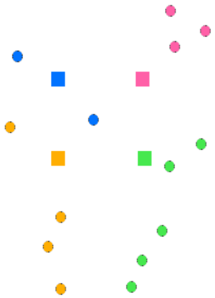
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

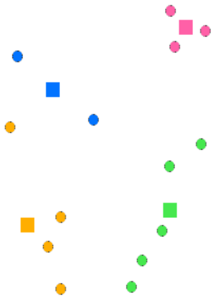
k-means clustering algorithm:



- K $=$ **4 clusters**
- **Euclidean** distance

Image clustering

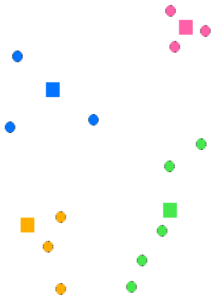
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

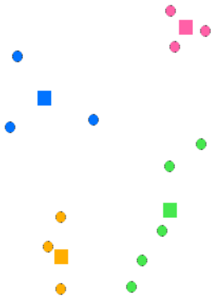
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

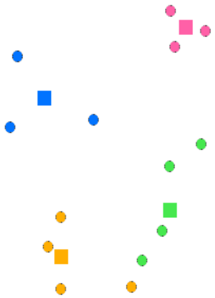
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

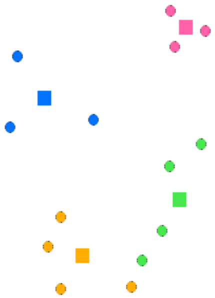
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

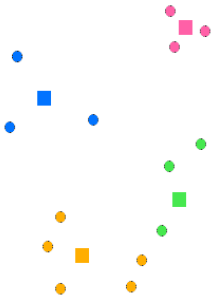
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

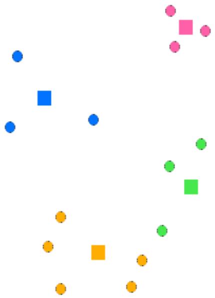
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

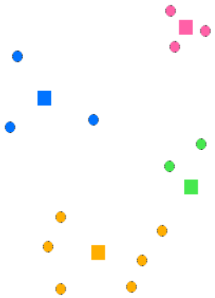
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

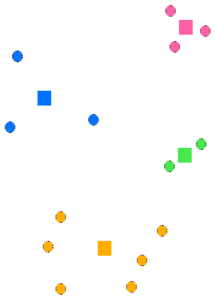
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

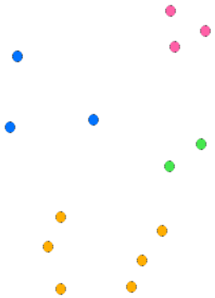
k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

k-means clustering algorithm:



- K = **4 clusters**
- **Euclidean** distance

Image clustering

Advantages:

- Any **number** of **categories** (not limited to object/background)
- **Gray level** and **color** images (or any number of channels)

Drawbacks:

- Need to define a **number** of categories
- Sensitivity to **initialization**
- Sensitivity to **local minima**

Outline

Region growing

Region growing algorithms have **two** main steps:

- **Seed** initialization
- **Expansion** through **similarity** criteria (average intensity, variance, texture, ...)

Region growing

Region growing algorithms have **two** main steps:

- **Seed** initialization
- **Expansion** through **similarity** criteria (average intensity, variance, texture, ...)

Example: **Active contours** approach

Energy based on **partial differential equations**:

$$E(\mathcal{C}) = \sum_{0}^{1} (E_{\text{int}}(\mathcal{C}(q)) + (E_{\text{ext}}(\mathcal{C}(q)))dq \tag{3}$$

where E_{int} controls the **smoothness** of the contours
E_{ext} is responsible for **attracting** the contours towards **objects**

Region growing

Advantages:

- **Flexibility** of criteria (possibility to have several criteria)

Drawbacks:

- **Parameterization**
- Need for **seed initialization**
- **Slow**

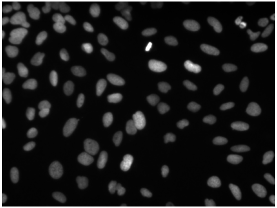
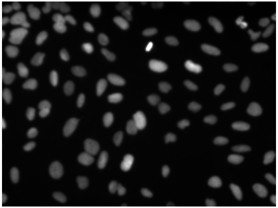
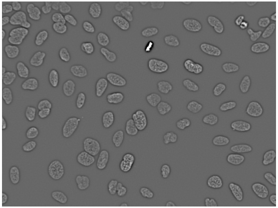
Outline

Frequency decomposition methods

Frequency decomposition methods methods (DoG, Fourier transform, ...) identify a specific **scale** in the image

Frequency decomposition methods

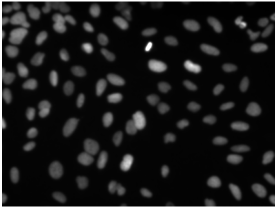
DoG (Difference of Gaussians):

Gaussian kernel $= 1$	Gaussian kernel $= 2$	Difference
		

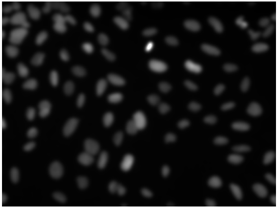
Frequency decomposition methods

DoG (Difference of Gaussians):

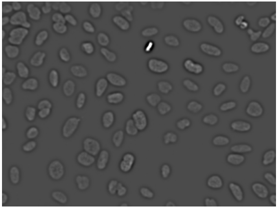
Gaussian kernel = 2



Gaussian kernel = 4



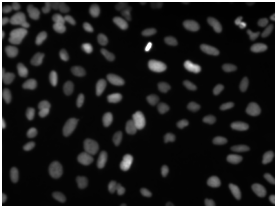
Difference



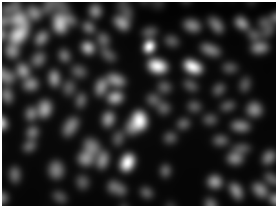
Frequency decomposition methods

DoG (Difference of Gaussians):

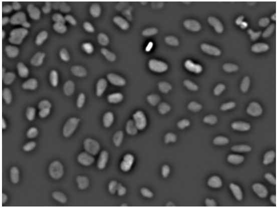
Gaussian kernel $= 2$



Gaussian kernel $= 10$



Difference



Frequency decomposition methods

Advantages:

- **Accurate** when objects are the **same size**
- **Robust** to noise

Drawbacks:

- Need for **specific scale** that is **different enough** from the rest of the image

Outline

Machine learning

Contrary to **traditional approaches** where each **single step** is fully **determined** by the developer, machine learning algorithms are **trained to "learn"**:

- A **training dataset** is defined such that input images and outputs (detection, classification, decision, ...) are **known**
- The **training process** for the algorithm consists in finding how to estimate the outputs given the input images
- Once the **algorithm** is **trained**, other images can be processed

Machine learning

Advantages:

- Potentially solves **complex problems**

Drawbacks:

- Need for **large training datasets** if using deep learning
- Need to define **image features** if using traditional machine learning algorithms
- Need for **optimization** (complex for non specialists)

Practice

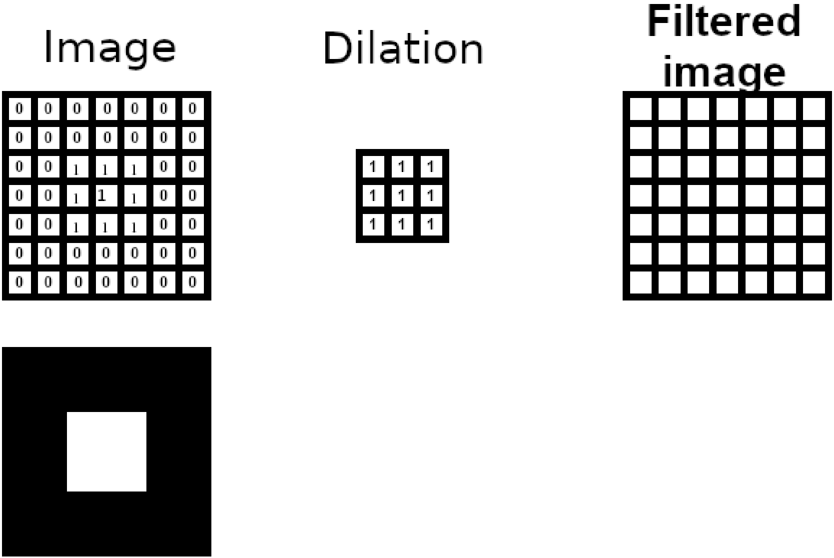
https://youtu.be/5YLf0RZukA8

Outline

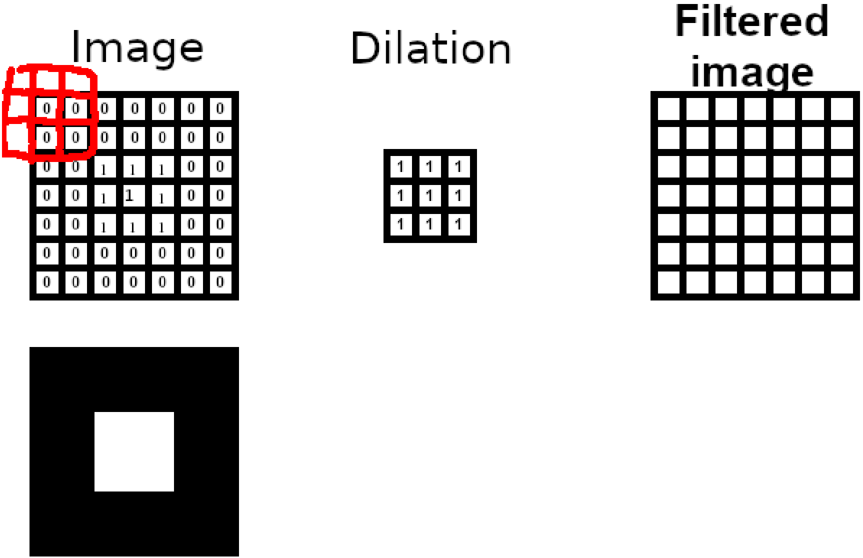
Morphological operations

Morphological operations correspond to **image filtering** applied to **binary images**

Morphological operations



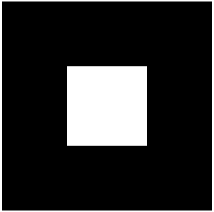
Morphological operations



Morphological operations

Image

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0



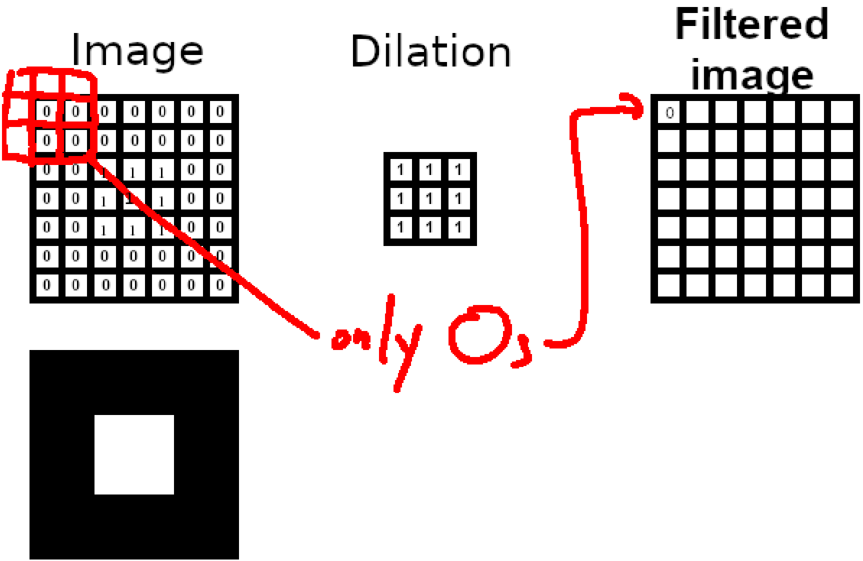
Dilation

1	1	1
1	1	1
1	1	1

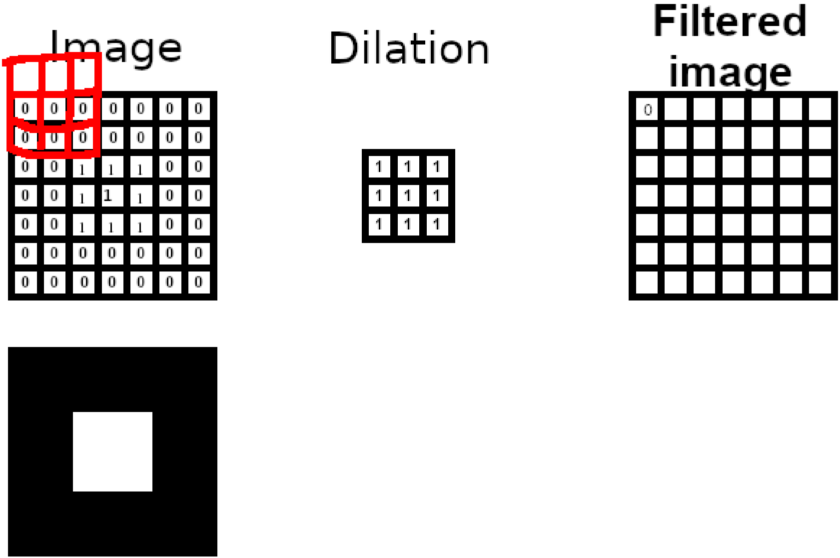
only 0s

Filtered image

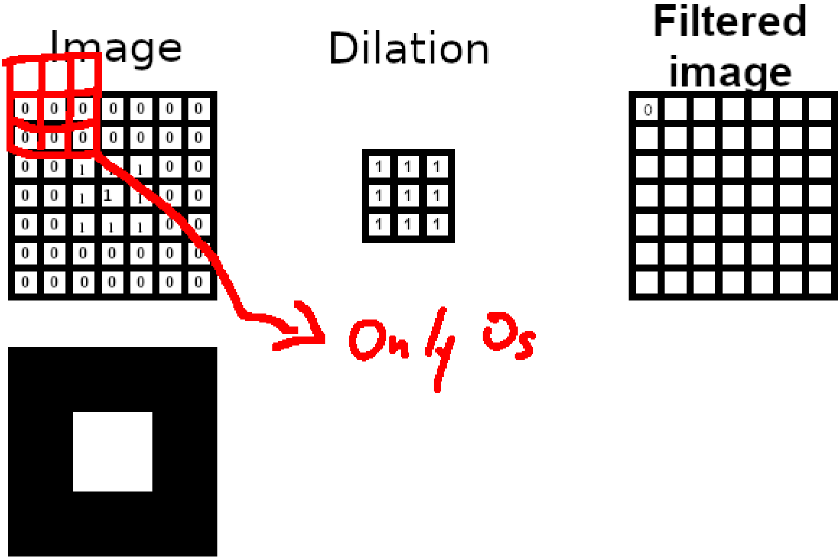
Morphological operations



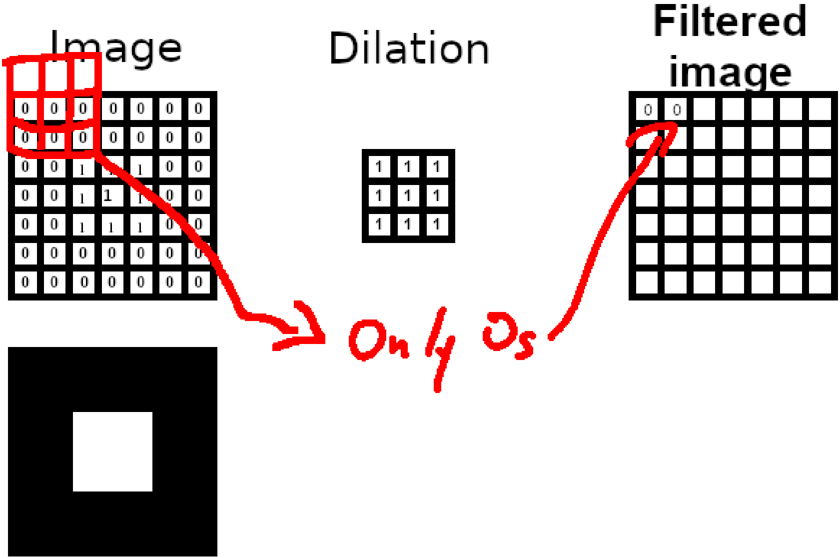
Morphological operations



Morphological operations

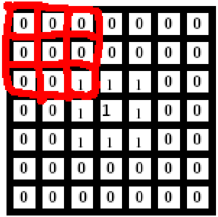


Morphological operations

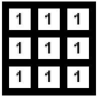


Morphological operations

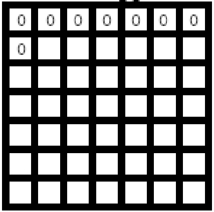
Image

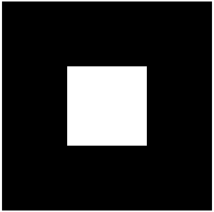


Dilation

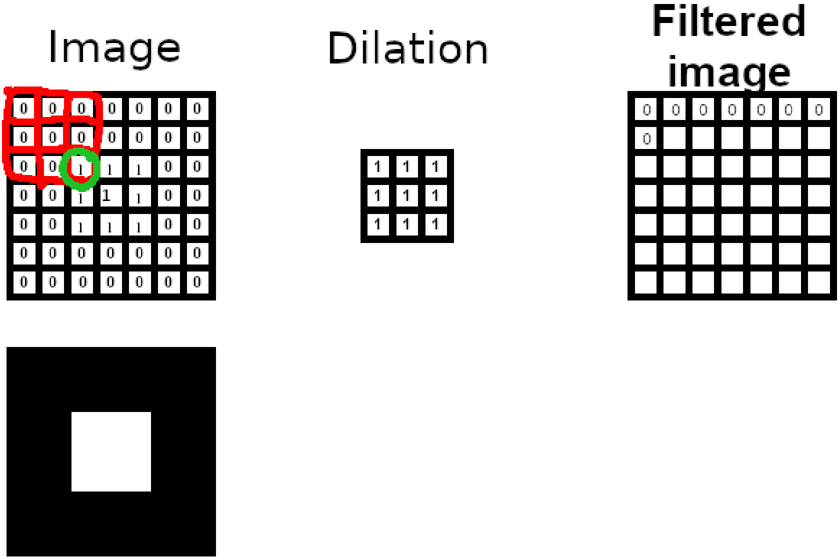


Filtered image

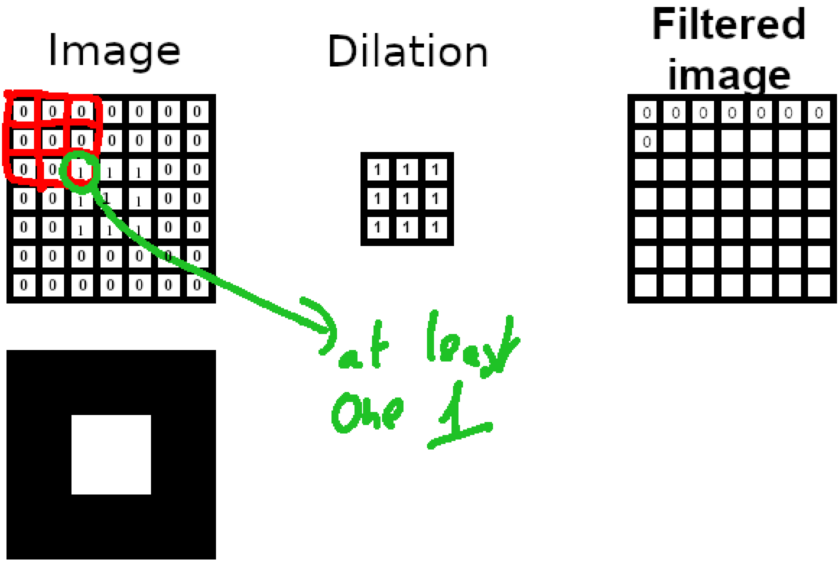




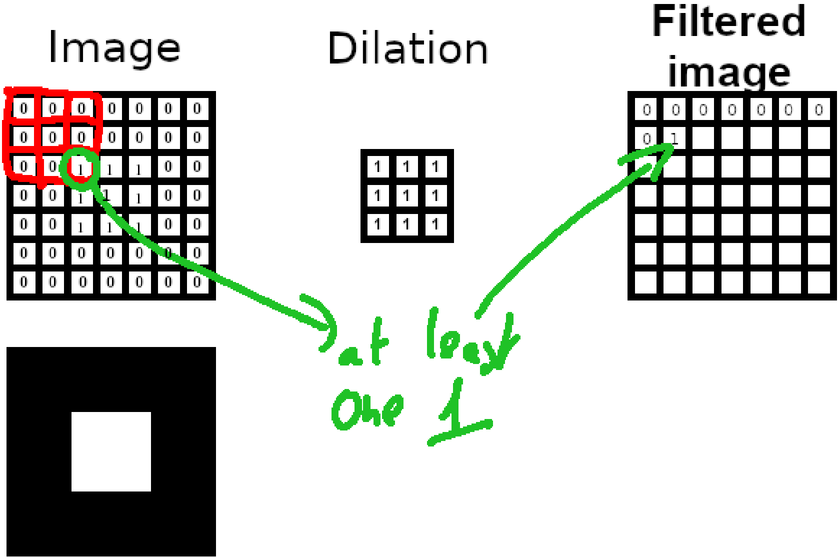
Morphological operations



Morphological operations



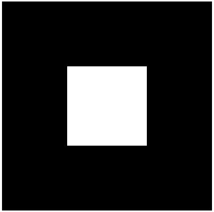
Morphological operations



Morphological operations

Image

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0



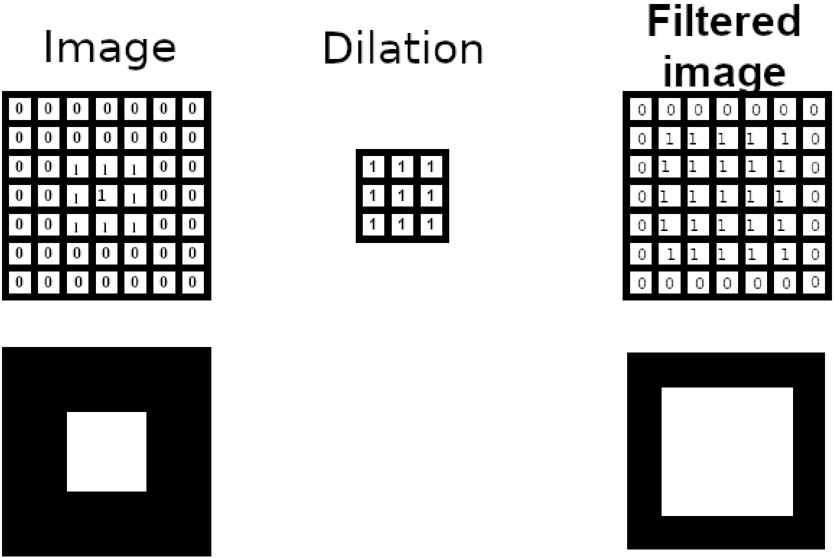
Dilation

1	1	1
1	1	1
1	1	1

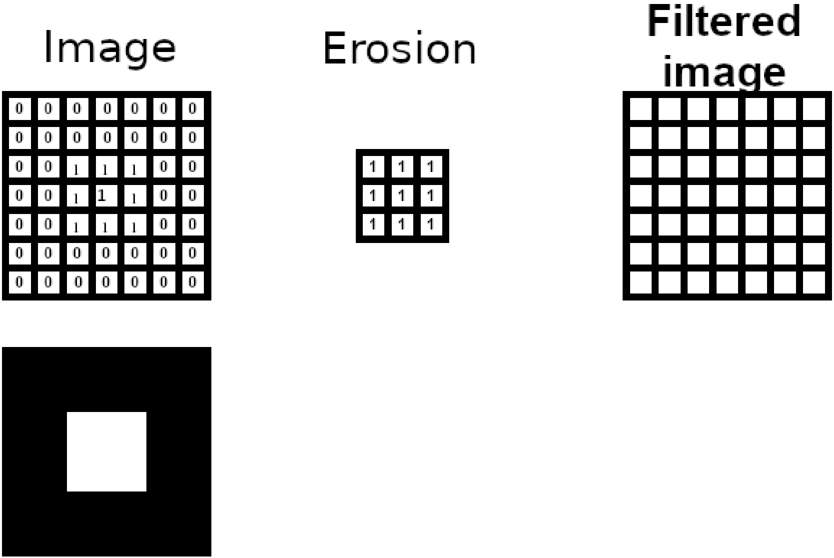
Filtered image

0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0

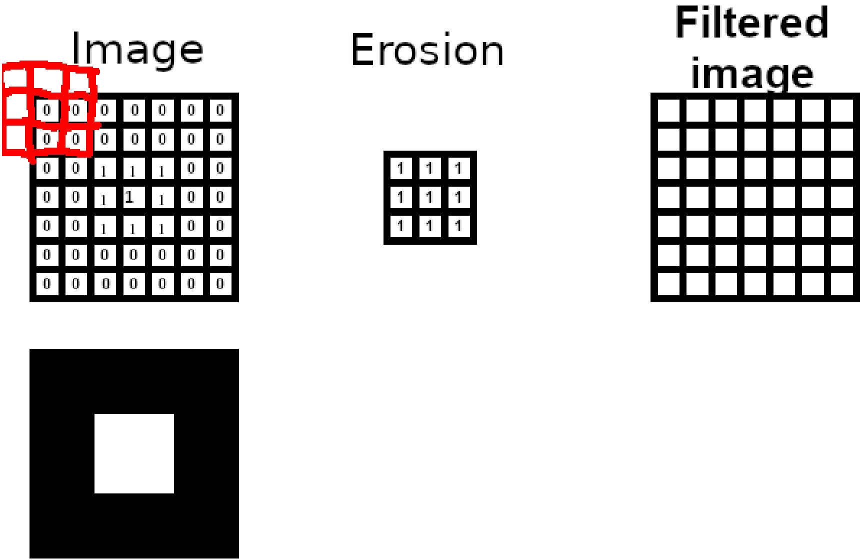
Morphological operations



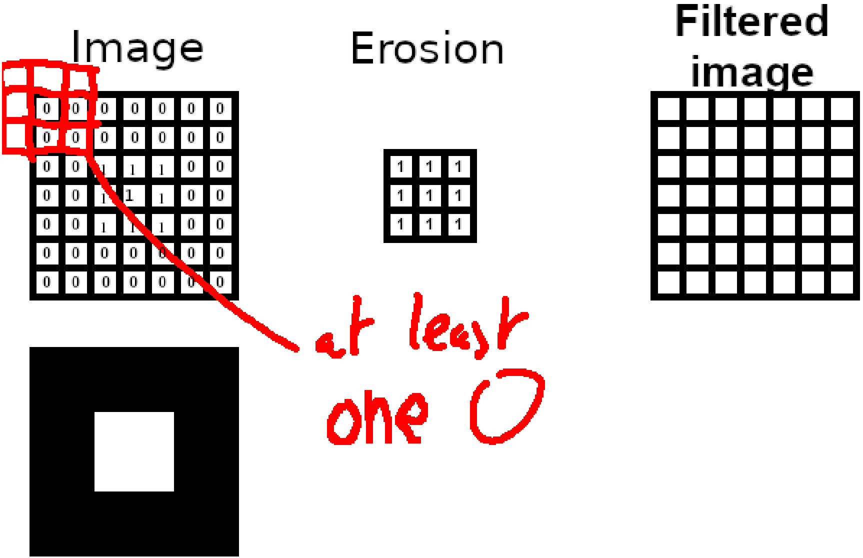
Morphological operations



Morphological operations

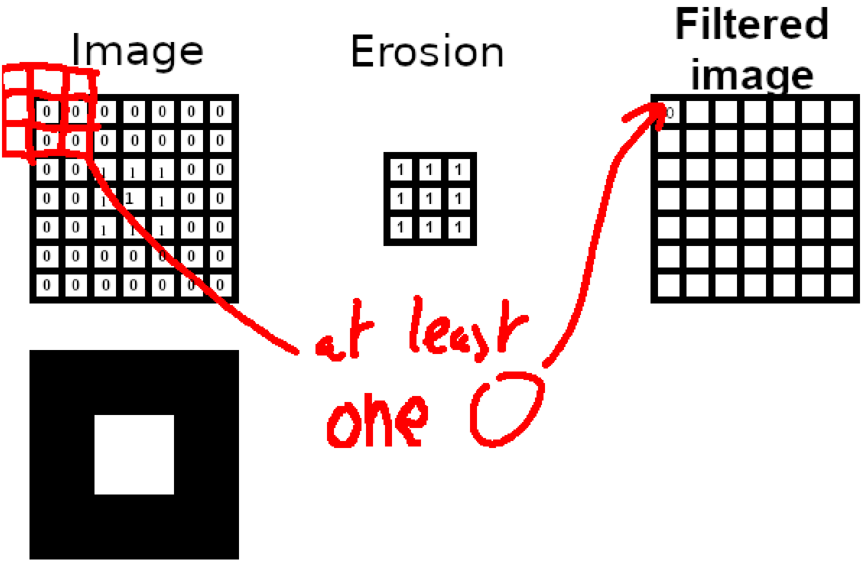


Morphological operations

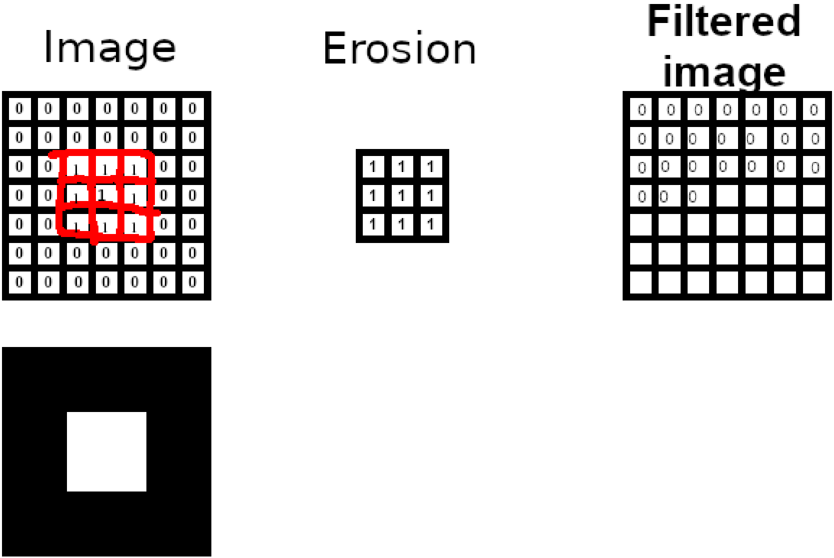


at least one 0

Morphological operations



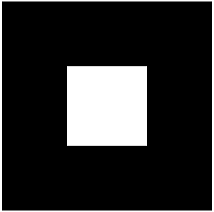
Morphological operations



Morphological operations

Image

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0



Erosion

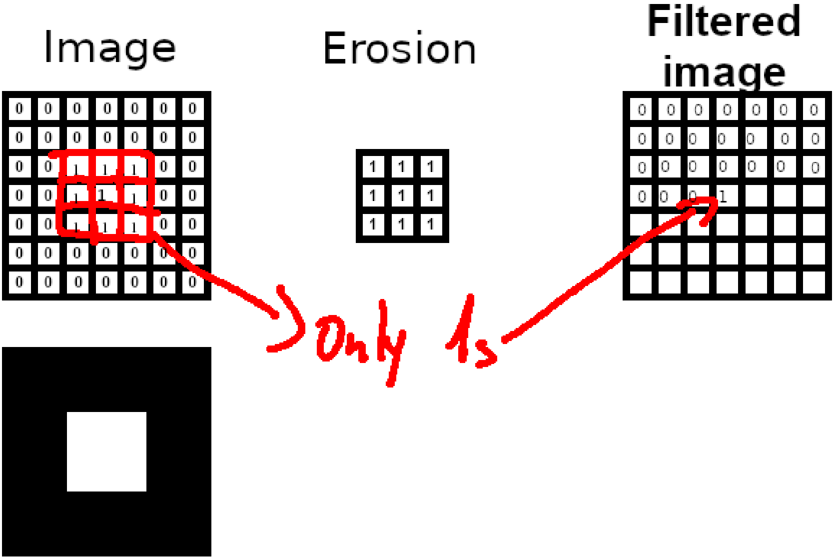
1	1	1
1	1	1
1	1	1

Only 1s

Filtered image

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0				

Morphological operations



Morphological operations

Image

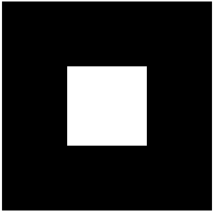
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

Erosion

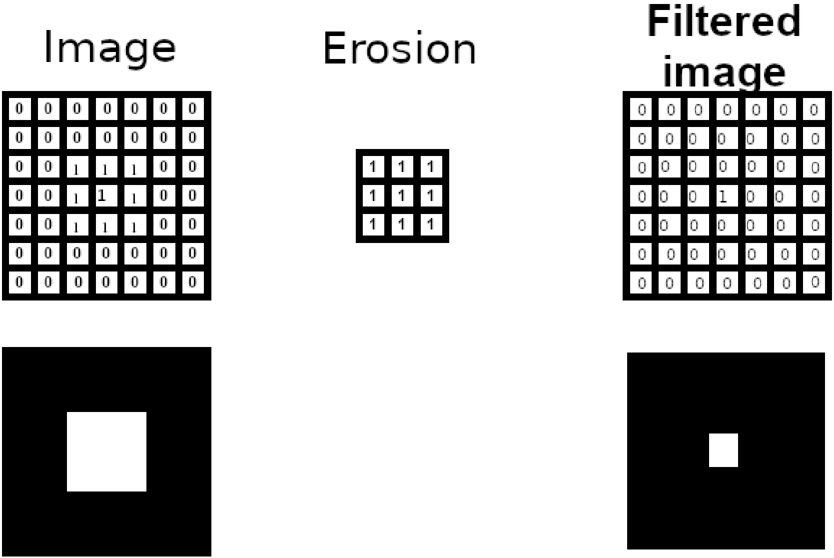
1	1	1
1	1	1
1	1	1

Filtered image

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

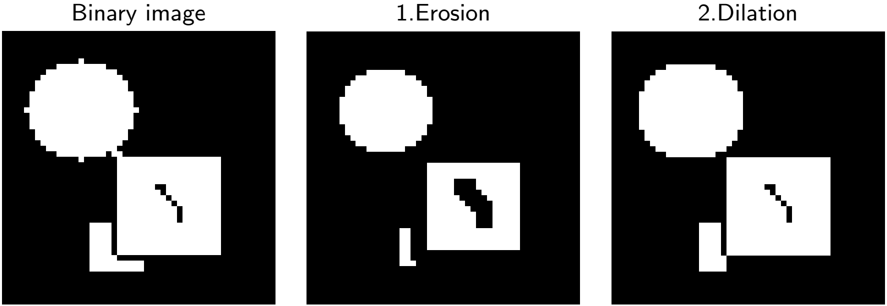


Morphological operations



Opening

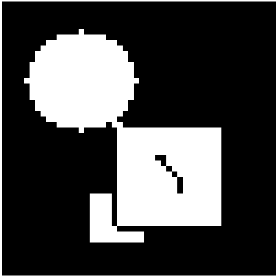
Opening: first **erosion** then **dilation**



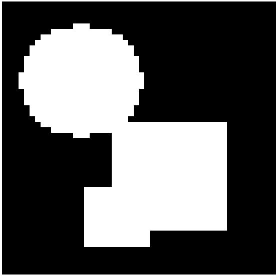
Closing

Closing: first **dilation** then **erosion**

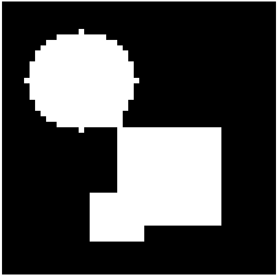
Binary image



1.Dilation

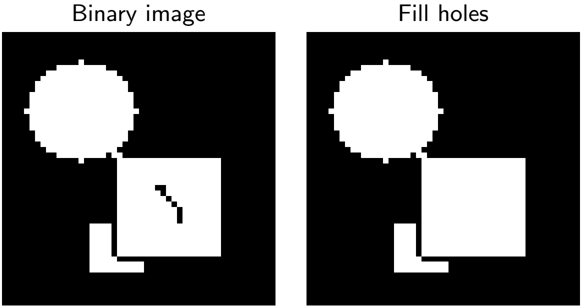


2.Erosion



Fill holes

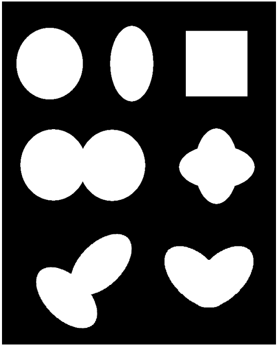
Fill holes: swap regions in the **background** component **to the object** component when **completeley surrounded** by pixels belonging to the object component



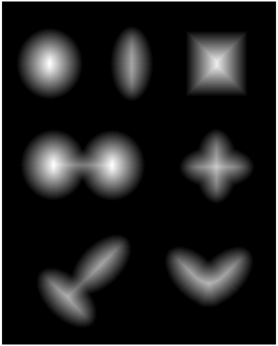
Distance map

Distance map: distance to the **closest** pixel in the **background** component for each pixel in the object component

Binary image

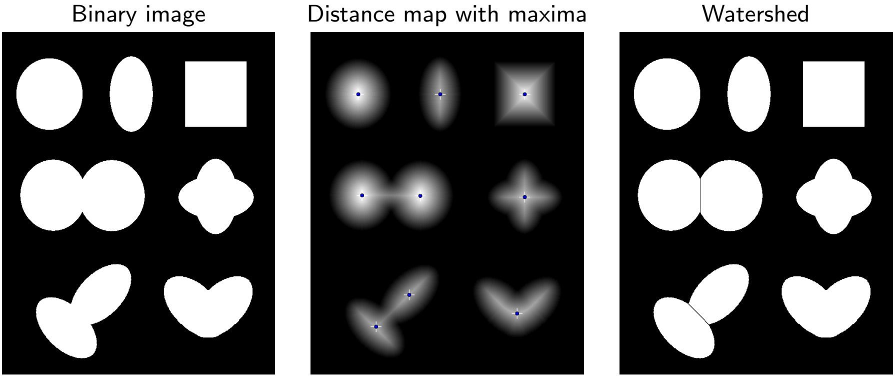


Distance map



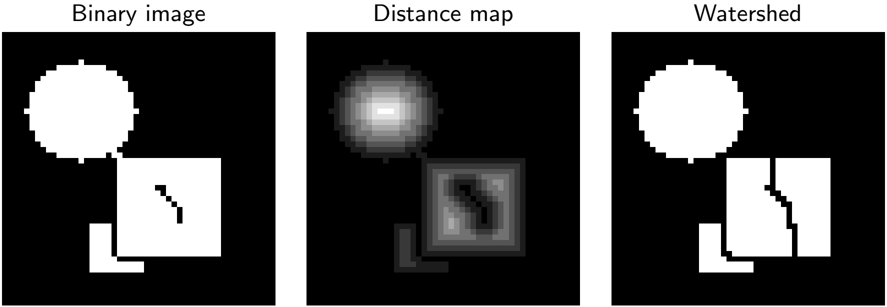
Watershed

Watershed: object separation based on the **local maxima** in the **distance map**



Watershed

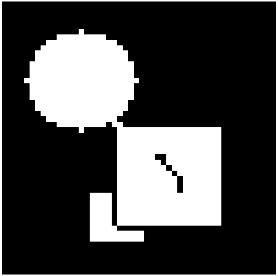
Watershed: object separation based on the **local maxima** in the **distance map**



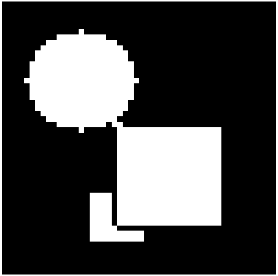
Watershed

Watershed: object separation based on the **local maxima** in the **distance map**

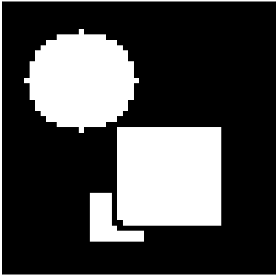
Binary image



1.Fill holes



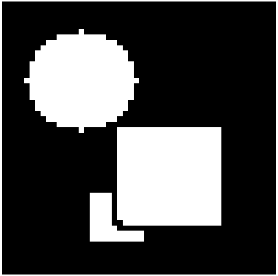
2.Watershed



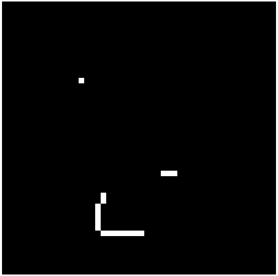
Skeletonization

Skeletonization: single pixel wide **skeletons** obtained by **iteratively removing** pixels located at the object **edge**

Binary image



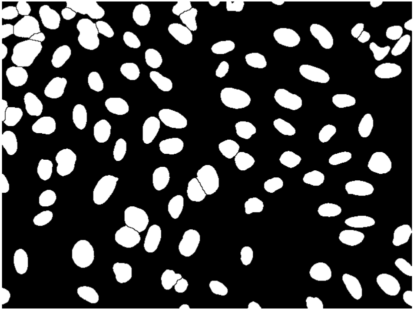
Skeletons



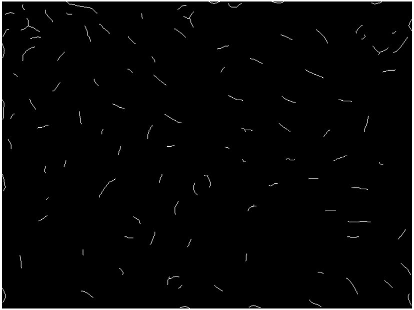
Skeletonization

Skeletonization: single pixel wide **skeletons** obtained by **iteratively removing** pixels located at the object **edge**

Binary image



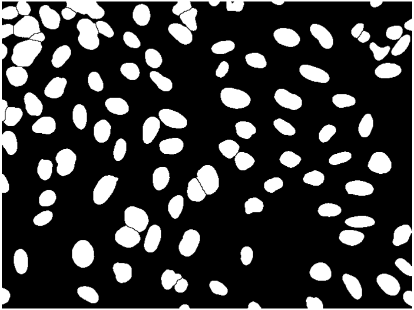
Skeletons



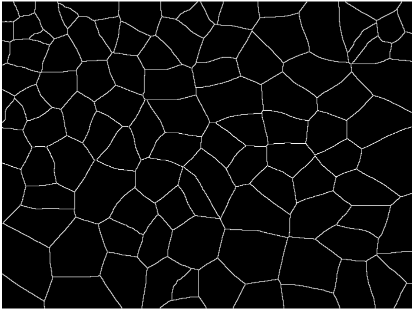
Voronoi

Voronoi tessellation: identification of points that are **equidistant** from at least two **separated objects**

Binary image



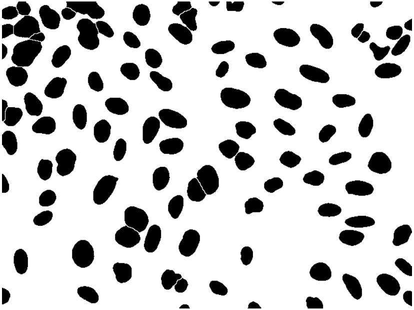
Voronoi



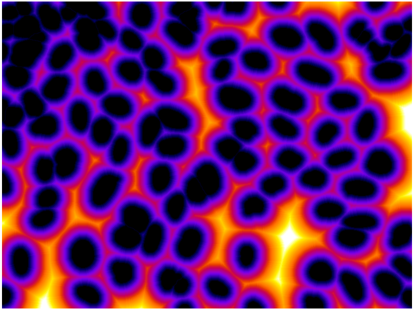
Distance map

Distance map applied to background gives the **closest distance** from each pixel in the background **to objects**

Binary image



Distance map



Practice

https://youtu.be/OfDEtmyztNE