**NUS-ISS**
*Vision Systems*

# Module 6 - Building vision system using machine learning (2) - Segmentation and tracking

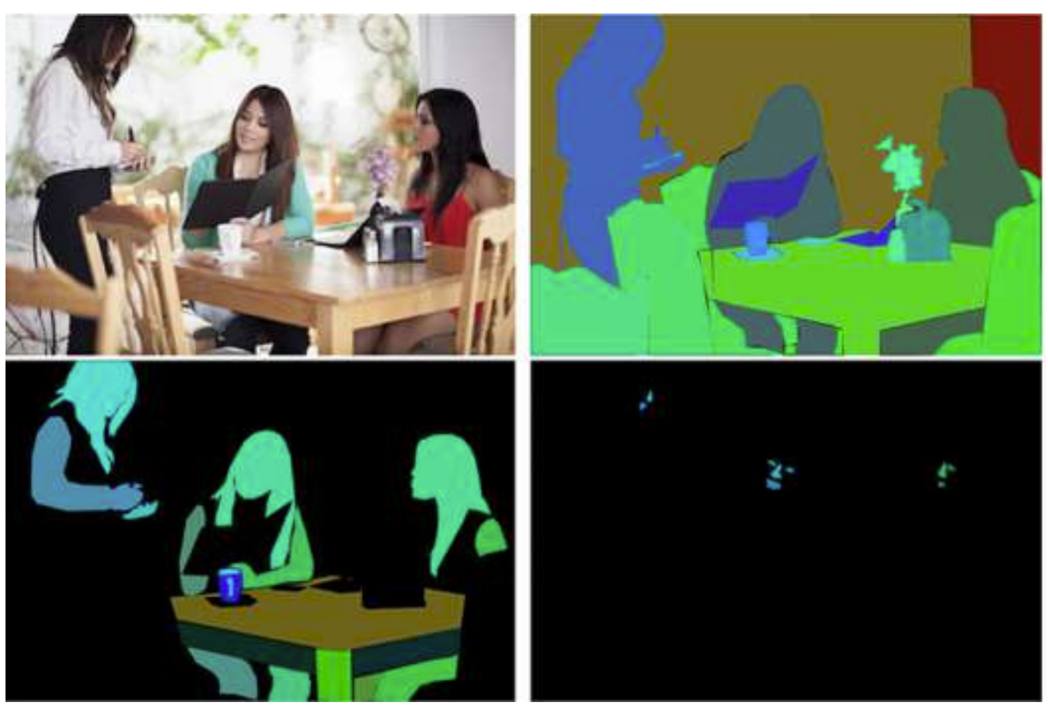
Dr. Tan Jen Hong
Lecturer & Consultant
Institute of System Science
National University of Singapore
issjht@nus.edu.sg

# Learning objectives

- Understand image segmentation
- Perform segmentation using morphological geometric active contour
- Perform segmentation using morphological geometric active contour without edges
- Perform morphological operation

# Image segmentation

- Image segmentation: paritioning an image into multiple segments
- Each segment is a set of pixels, to represent particular item(s) in an image
- Typically used to locate objects or their boundaries in an image
- Precise definition: assign a label to every pixel in an image



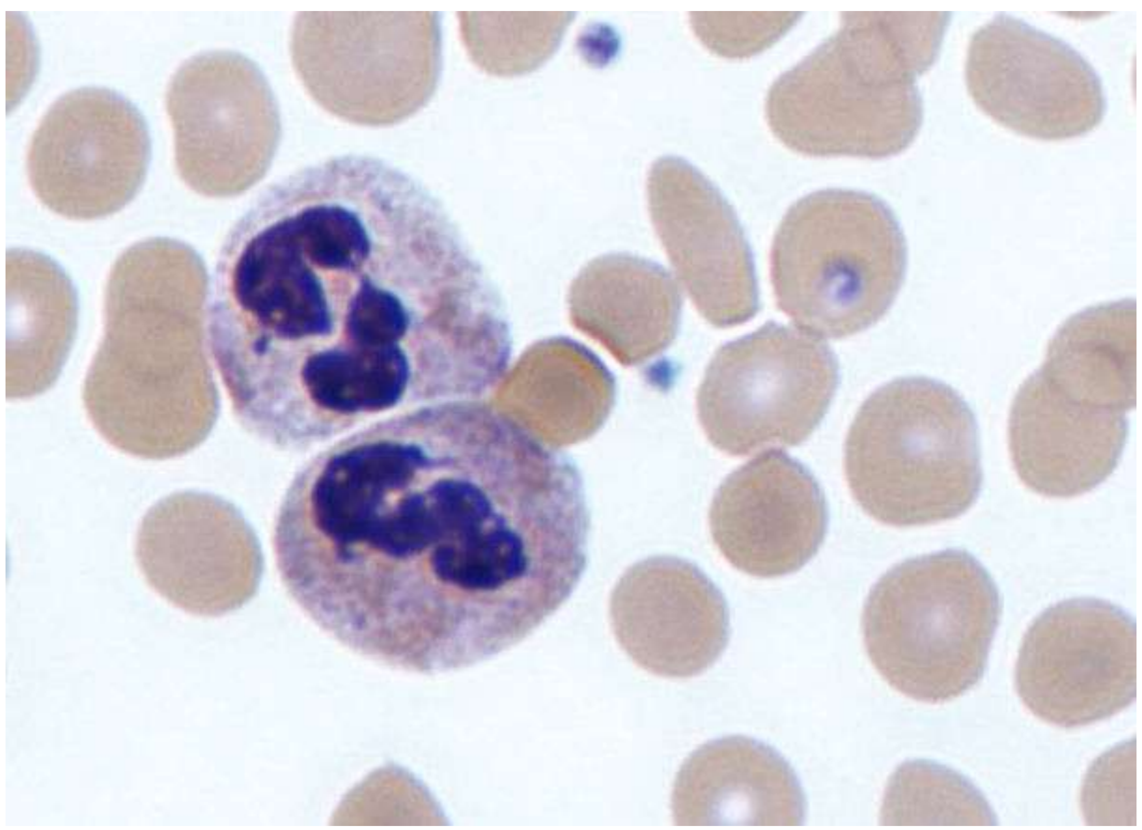
Source: https://towardsdatascience.com/paper-summary-recent-progress-in-semantic-image-segmentation-d7b93ee1b705

## Why the need of advance technique?

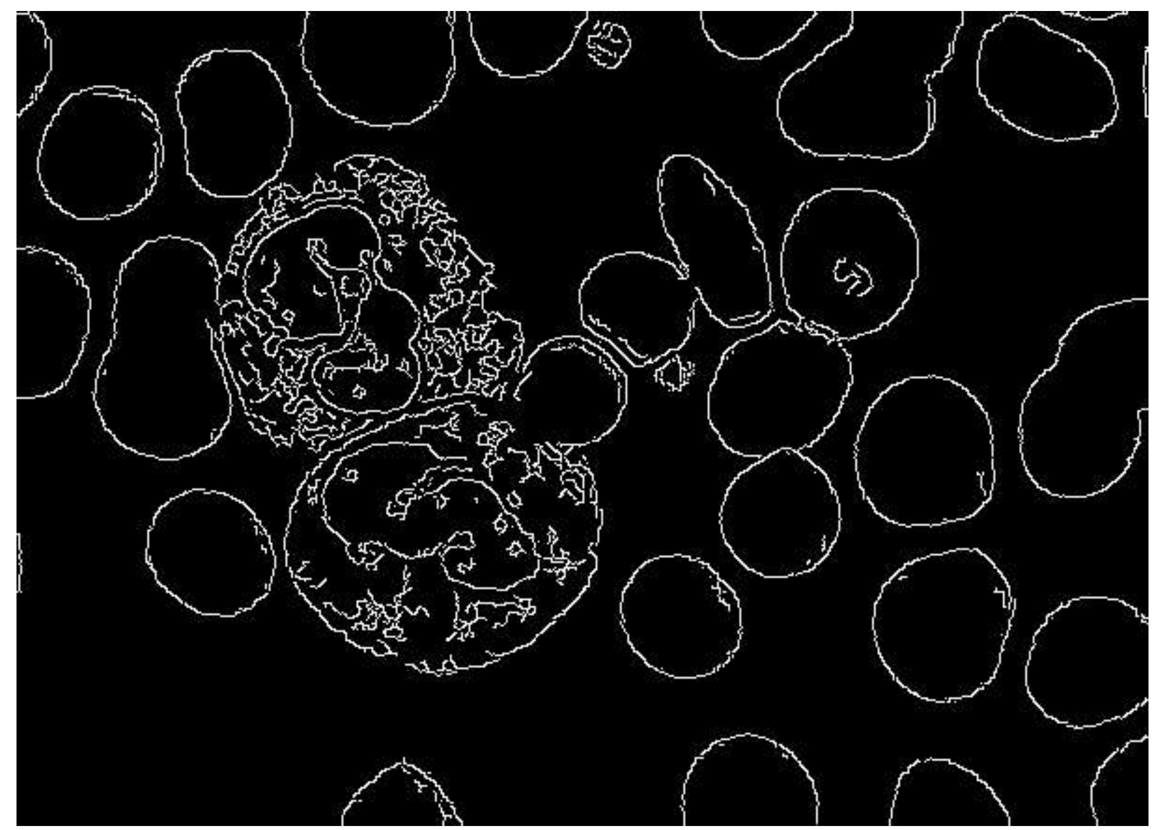
- Edge detection doesn't work nearly all of the time

- Threshoding doesn't work nearly all of the time

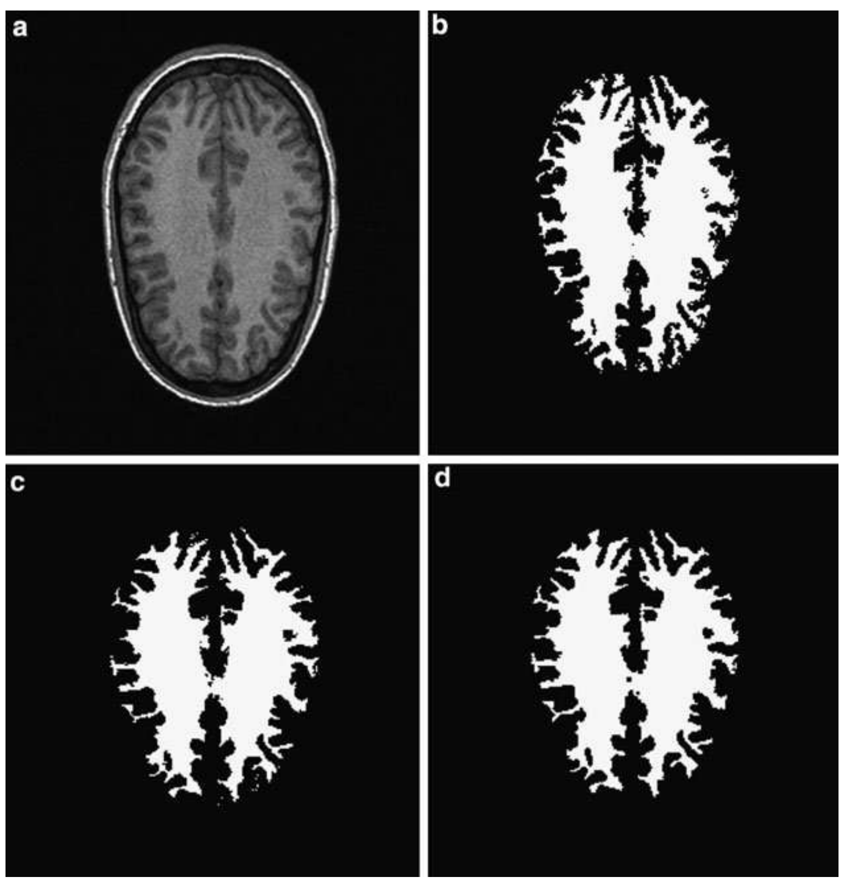
nuc



canny edge detection



# Image segmentation
## Methods



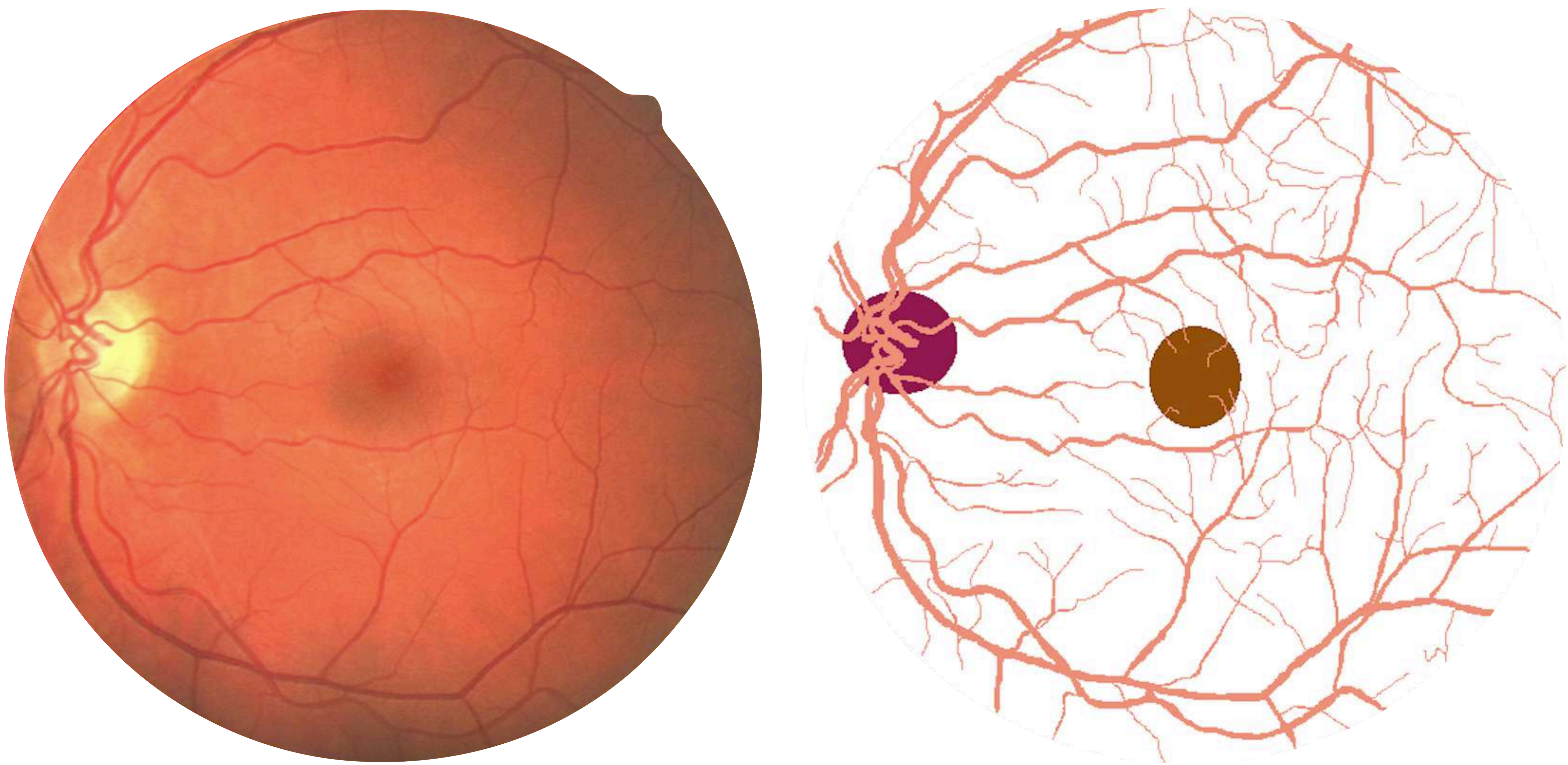

Source: https://www.researchgate.net/publication/226478233_Cerebral_White_Matter_Segmentation_using_Probabilistic_Graph_Cut_Algorithm/figures?lo=1

- Many methods were proposed to perform segmentation in literature:

  Region growing
  Watershed
  k-means clustering
  Level-set
  Fast marching
  Mumford-Shah model
  Random walker
  Markov random fields
  Simulated annealing
  Expectation maximization
  Active contour
  Active shape model
  Active appearance model
  ...

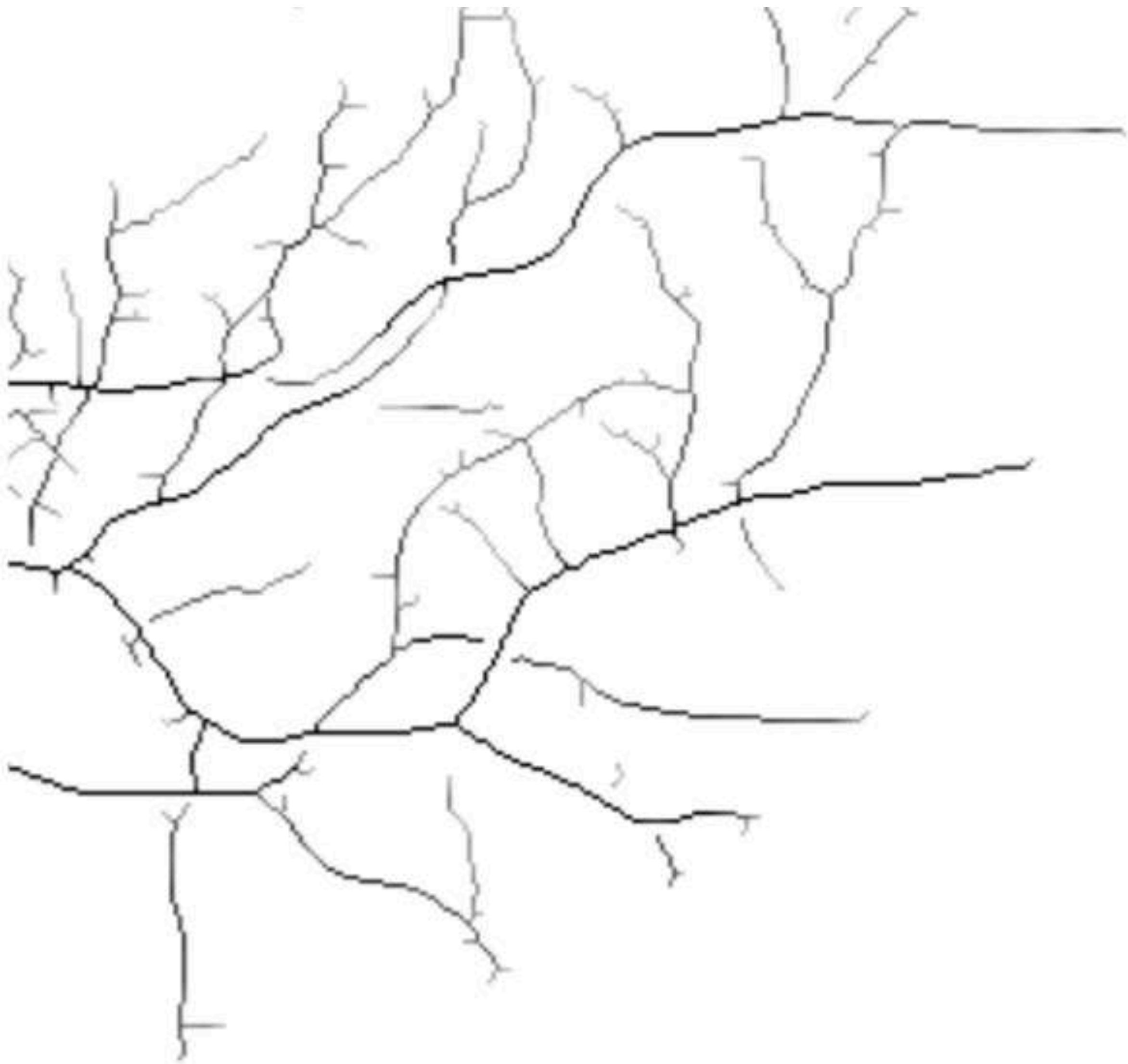
- In literature, terms often encountered: automated, semi-automated

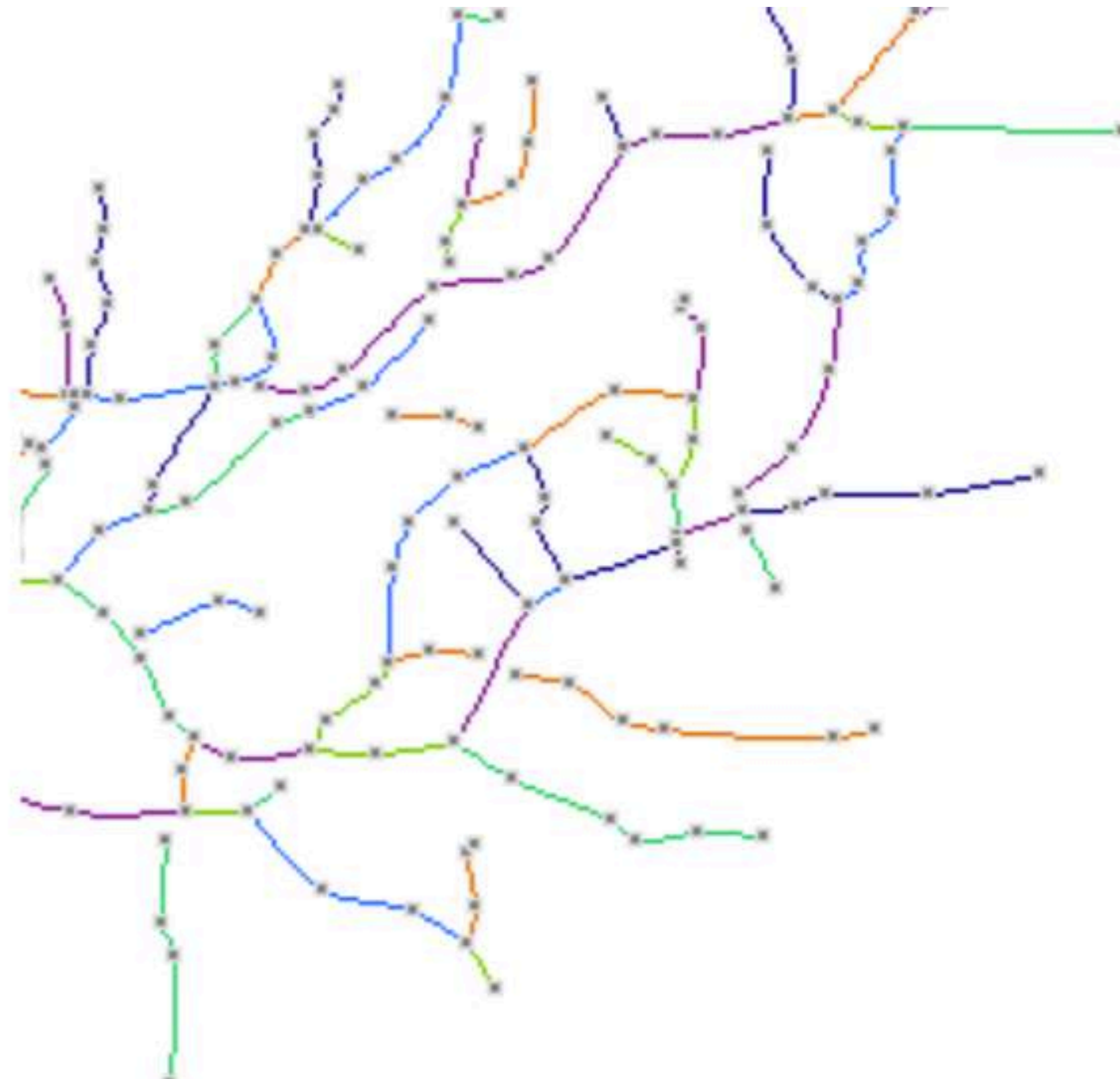
# Image segmentation
## Retinal images



# Image segmentation

## Blood vessels

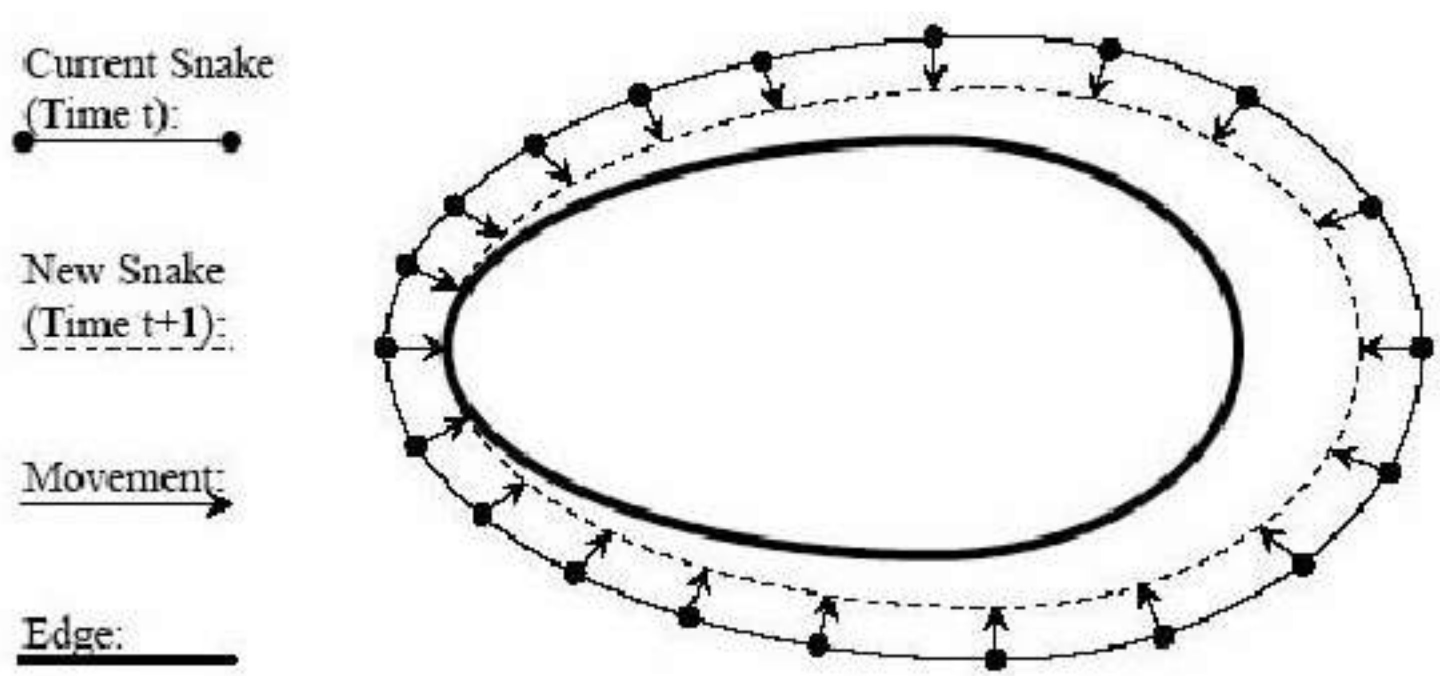




Automated extraction of retinal vasculature

# Active contour model
Snakes

- Active contour model, also called snakes in literature
- Introduced by Michael Kass, Andrew Witkin and Demetri Terzopoulos around 1980
- One of the power solutions for object tracking, shape recognition, segmentation, edge detection and stero matching in the early days




Source: https://www.researchgate.net/publication/226478233_Cerebral_White_Matter_Segmentation_using_Probabilistic_Graph_Cut_Algorithm/figures?lo=1

# Active contour model

## Snakes

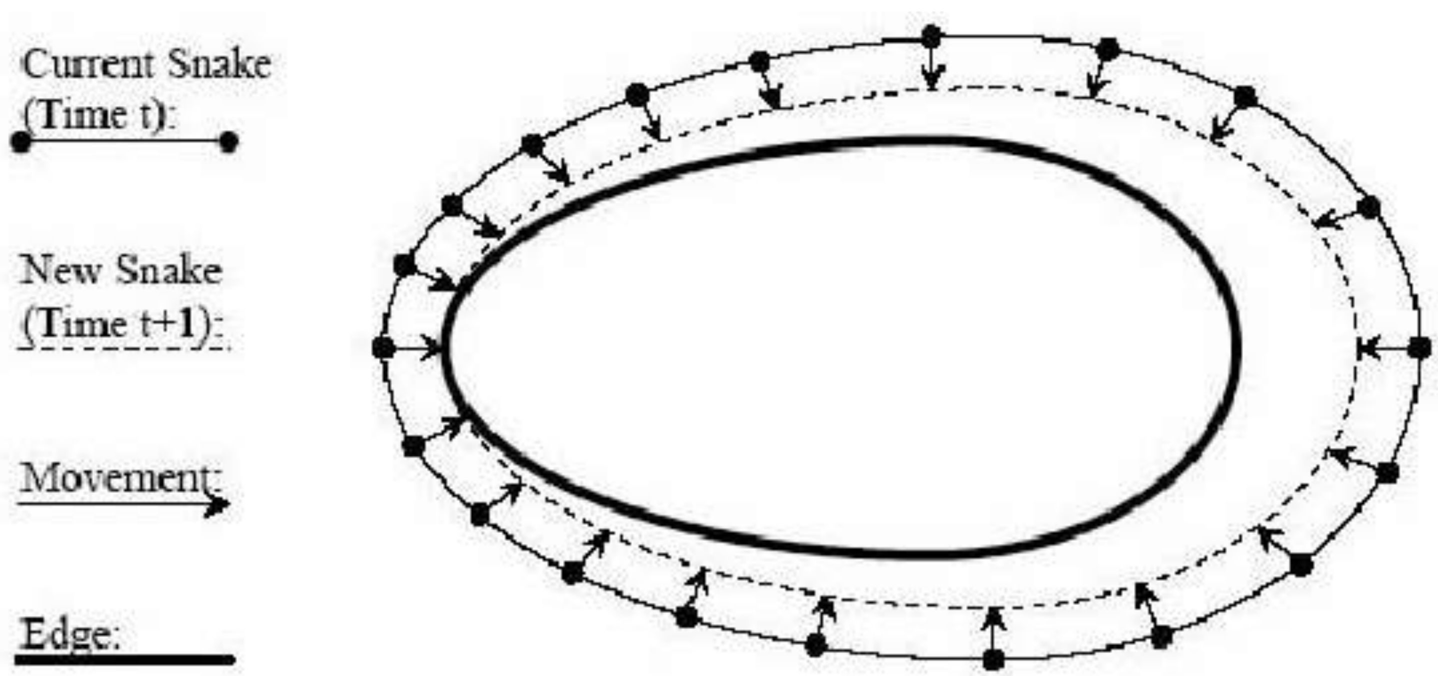
- As the contour moves, it tries to minimize the energy

$$E = \int_0^1 \left[ \frac{1}{2} \left( \alpha \left| \mathbf{x}'(s) \right|^2 + \beta \left| \mathbf{x}''(s) \right|^2 \right) + E_{ext} \left( \mathbf{x}(s) \right) \right] ds$$

the energy derived from the length / tension of the contour

the energy derived from the smoothness of the contour

the energy derived from the image of interest




Source: https://www.researchgate.net/publication/226478233_Cerebral_White_Matter_Segmentation_using_Probabilistic_Graph_Cut_Algorithm/figures?lo=1

# Active contour model

## Snakes

- As the contour moves, it tries to minimize the energy

$$E = \int_0^1 \left[ \frac{1}{2} \left( \alpha \left| \mathbf{x}'(s) \right|^2 + \beta \left| \mathbf{x}''(s) \right|^2 \right) + E_{ext} \left( \mathbf{x}(s) \right) \right] ds$$

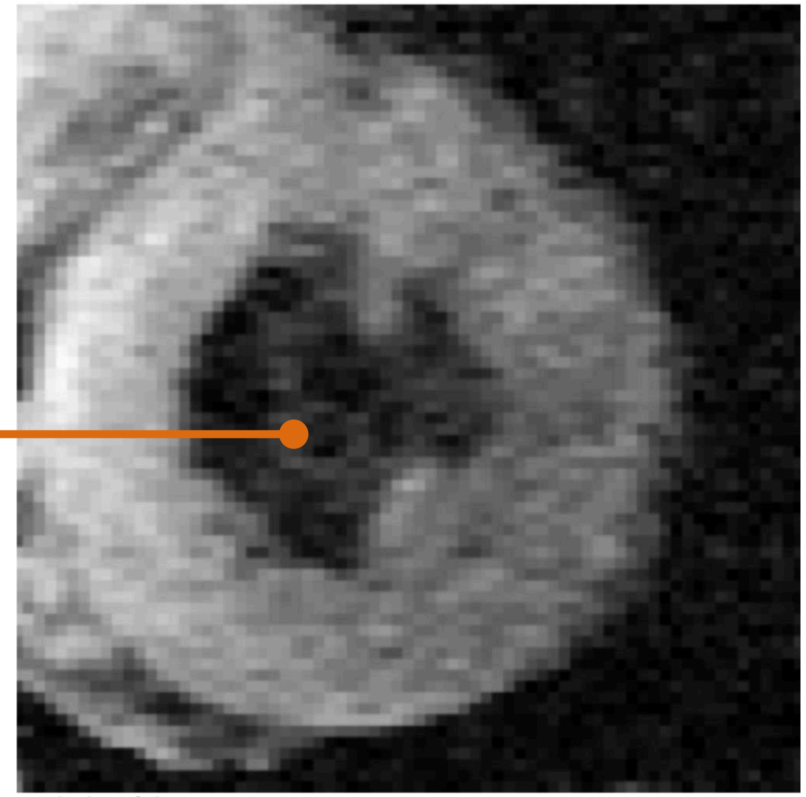
the energy derived from the length / tension of the contour

the energy derived from the smoothness of the contour

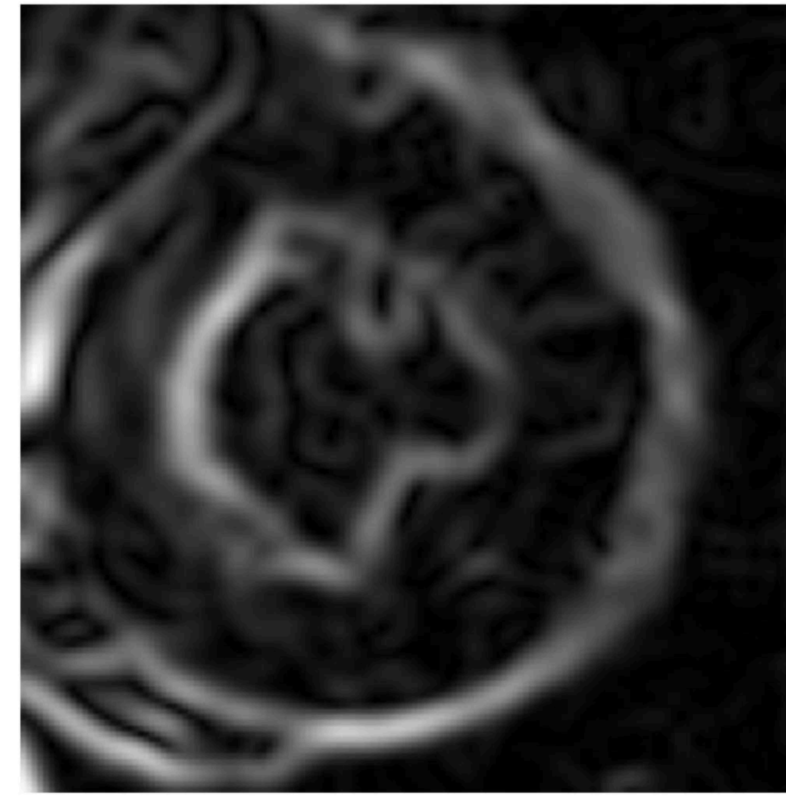
the energy derived from the image of interest

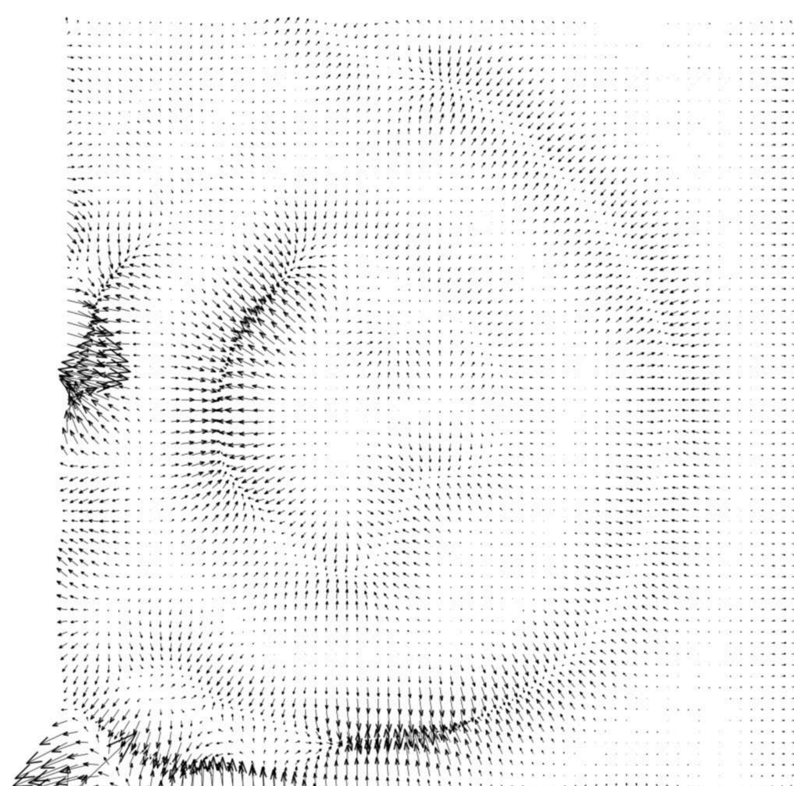
# Active contour model

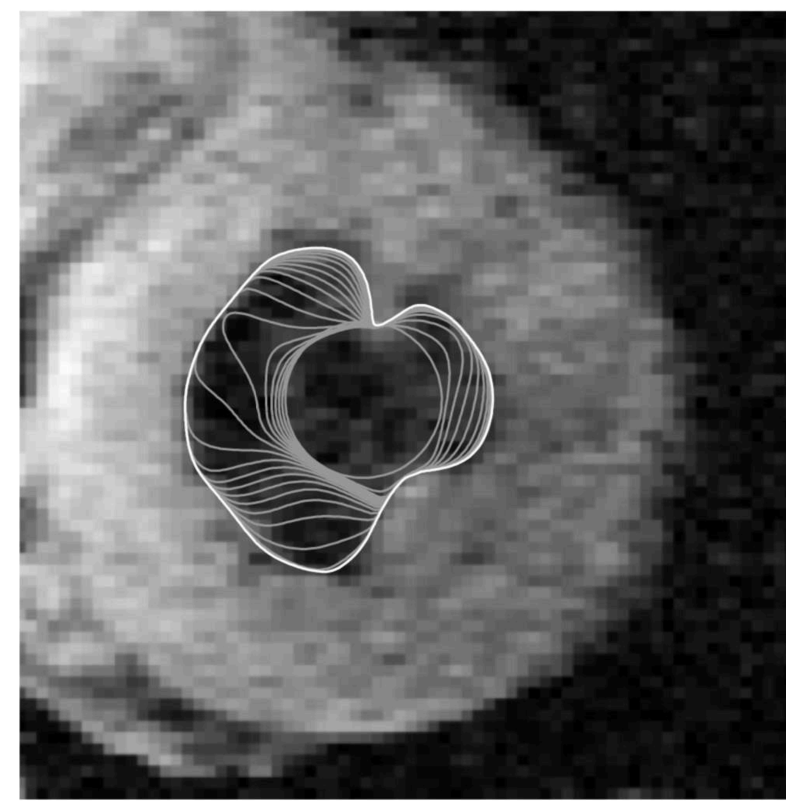
Snakes on MRI of left ventrical of a human heart



original image


edge map

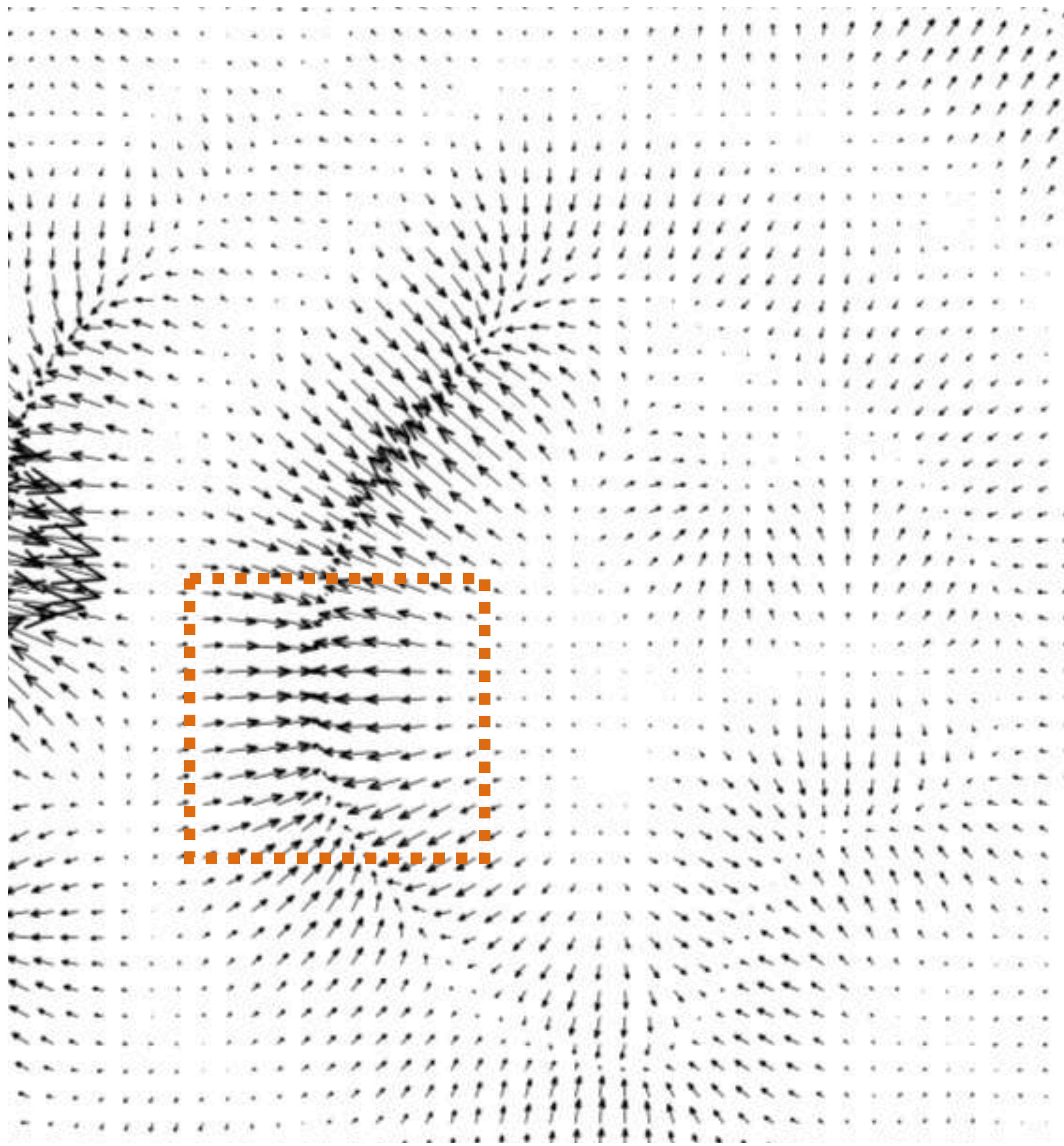

vector flow


movement of contour

Source:http://iacl.ece.jhu.edu/pubs/p087c.pdf

# Active contour model

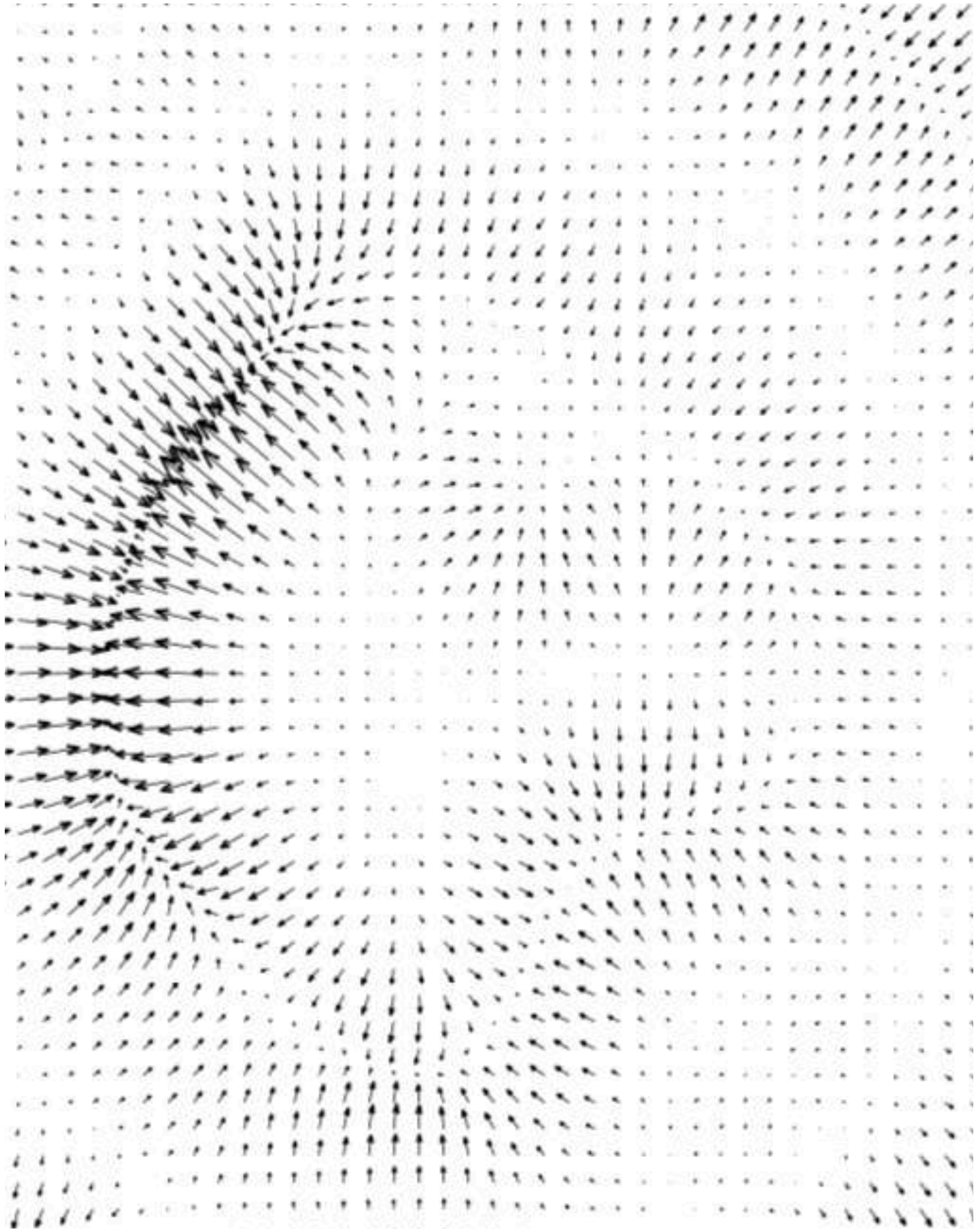
gradient field

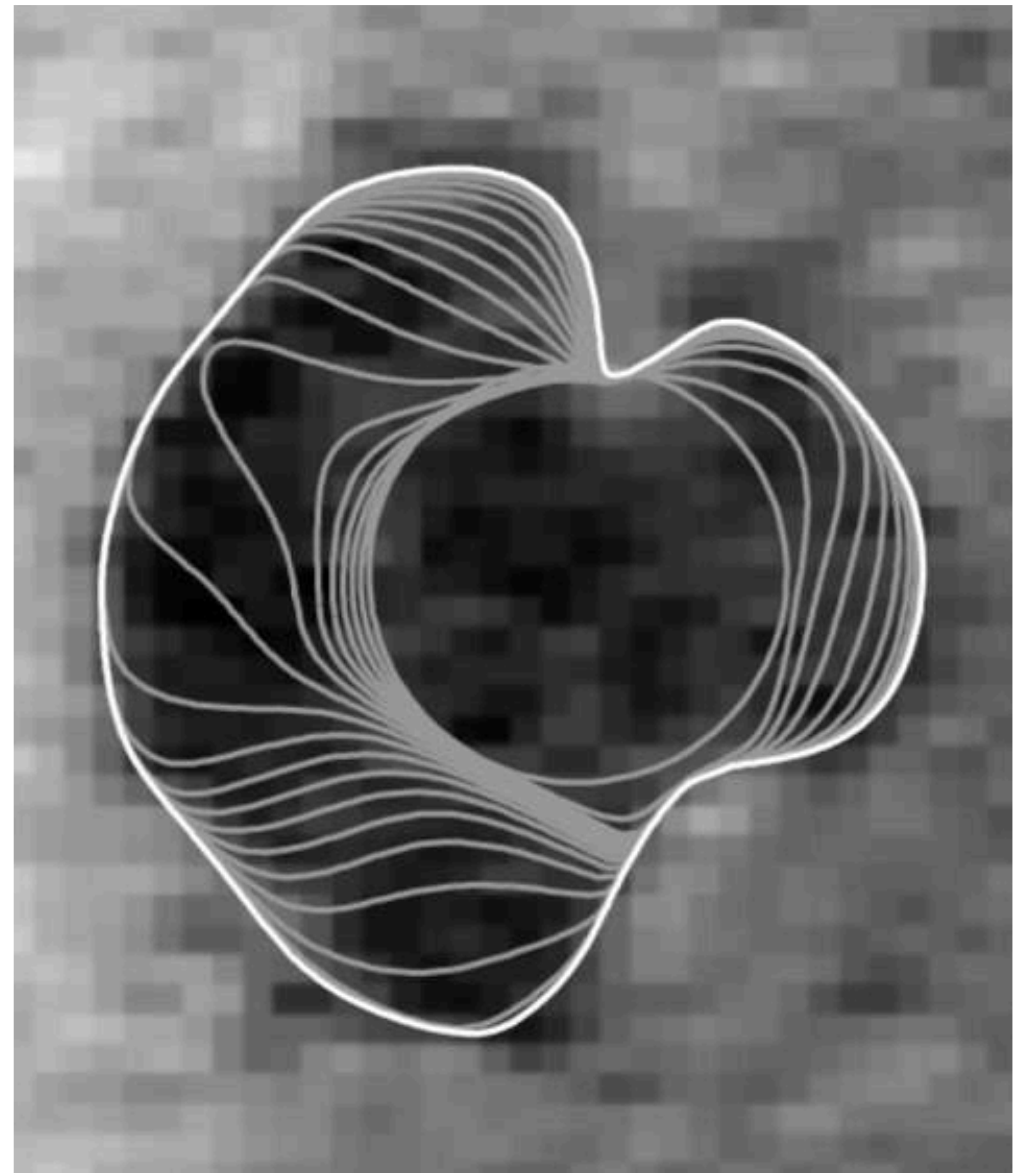


Source:http://iacl.ece.jhu.edu/pubs/p087c.pdf

# Active contour model

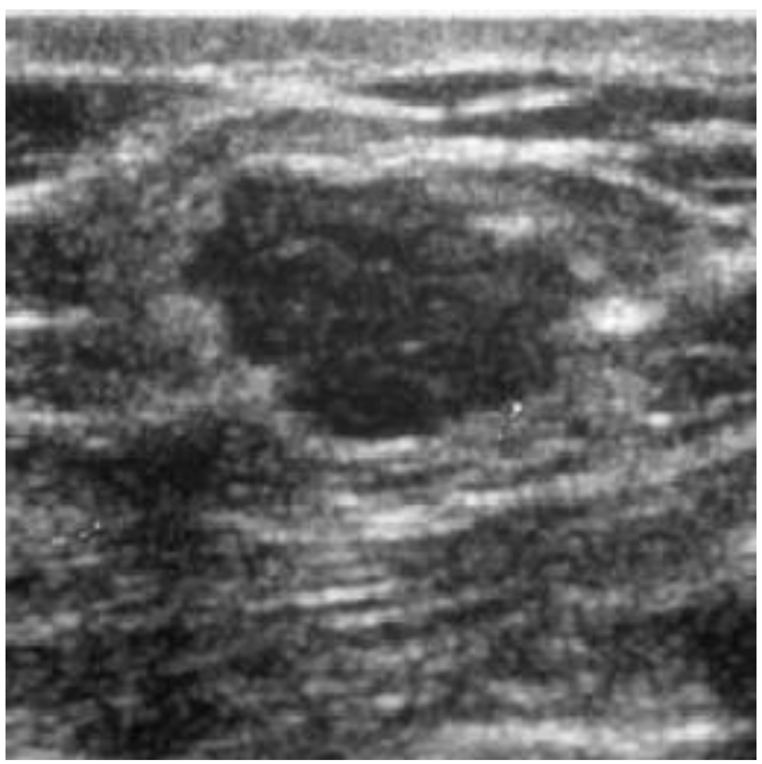
## Snakes





Source:http://iacl.ece.jhu.edu/pubs/p087c.pdf

## Problem



- Object of interest: a nodule
- No clear boundary between the object of interest and its surrounding
- Image is noisy;  thresholding unlikely to work
- Nodule has no sensible shape, description or appearance profile; not possible to use Viola-Jones algorithm

# Morphological snake

Geometric active contour

- Segmenting process: continuous calculating below expression

$$\frac{\partial u}{\partial t} = \underbrace{g(I)|\nabla u|\,\mathrm{div}\left(\frac{\nabla u}{|\nabla u|}\right)}_{\text{smoothing force}} + \underbrace{g(I)|\nabla u|\nu}_{\text{balloon force}} + \underbrace{\nabla g(I)\nabla u}_{\text{image attraction force}}$$

IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 1

## A morphological approach to curvature-based evolution of curves and surfaces

Pablo Márquez-Neila, Luis Baumela and Luis Alvarez

**Abstract**—We introduce new results connecting differential and morphological operators that provide a formal and theoretically grounded approach for stable and fast contour evolution. Contour evolution algorithms have been extensively used for boundary detection and tracking in computer vision. The standard solution based on partial differential equations and level-sets requires the use of numerical methods of integration that are costly computationally and may have stability issues. We present a morphological approach to contour evolution based on a new curvature morphological operator valid for surfaces of any dimension. We approximate the numerical solution of the curve evolution PDE by the successive application of a set of morphological operators defined on a binary level-set and with equivalent infinitesimal behavior. These operators are very fast, do not suffer numerical stability issues and do not degrade the level set function, so there is no need of re-initializing it. Moreover, their implementation is much easier since they do not require the use of sophisticated numerical algorithms. We validate the approach providing a morphological implementation of the *Geodesic Active Contours*, the *Active Contours Without Borders* and *Turpopixels*. In the experiments conducted the morphological implementations converge to solutions equivalent to those achieved by traditional numerical solutions, but with significant gains in simplicity, speed and stability.

**Index Terms**—Computer vision, Mathematical Morphology, Curve Evolution, Level-Sets, Morphological Snakes

### 1 Introduction

Active contours or snakes are one of the most widely used computer vision tools [1], [2]. Although they provide a unified account of a number of visual problems, including detection of edge and subjective contours [2] and stereo matching [3], they have been extensively used for object boundary detection and tracking [1], [2], [4], [5], [6], [7], [8], [9] as well as segmenting 2D [10], [11], [12], [13], [14] and tensor images [15]. Recent results have shown that they can achieve robust tracking performance over long and challenging sequences with dramatic changes in target shape and appearance [16], [17] as well as overlaps, partial occlusions and poor image contrast [18]. It has also been used for oversegmentation [19]. These tasks are formulated in variational terms, where an image induces an energy functional on a curve or surface. Minimizing the functional in a steepest descent manner evolves the surface towards a local minimum that represents the solution of the problem.

Despite its great success, the original parametric active contour approach depends on the parametrization of the contour and cannot naturally handle changes in the topology of the curve. These issues were addressed in subsequent approaches such as the *Geodesic Active Contour* (GAC) [20], [21] and the *Active Contours Without Edges* (ACWE) [10], [11]. In the GAC the energy functional is a geodesic in a Riemannian manifold with a metric induced by image features, in its simplest case, the target borders. The ACWE does not need well defined borders and it is less sensitive to the initial configuration and to the model parameters. Both approaches are based on the level-set formulation [22], [23]. In this case the curve is evolved by propagating an interface represented by the zero level-set of a smooth function, using a time-dependent partial differential equation (PDE). The solution to this PDE is costly computationally, and in the case of the simplest finite-difference explicit numerical scheme, it has stability constraints on the size of the time step. Absolutely stable solutions to the GAC model improve the stability by combining a semi-implicit discretization with an additive operator splitting (AOS) [24], [25]. Level-set solutions typically develop steep or flat gradients that yield inaccuracies in the numerical approximation [26]. This is usually solved by periodically re-initializing the level-set function as a distance to the zero level-set, which can also be addressed as a front propagation problem [27]. This again increases the computational cost of the method and reduces the topological flexibility, since it prevents the level-set from creating new contours far away from the initial interface [26]. For the ACWE model, however, this re-initiliazation is optional [10].

Both the stability constraints and the necessity of re-initializing the distance function render traditional level-sets approaches as problematic schemes in time-

- *Pablo Márquez-Neila is with the Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, Spain.*
*E-mail: p.mneila@upm.es*
- *Luis Baumela is with the Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, Spain.*
*E-mail: lbaumela@fi.upm.es*
- *Luis Alvarez is with the Departamento de Informática y Sistemas, Universidad de las Palmas de Gran Canaria, Spain.*
*E-mail: lalvarez@dis.ulpgc.es*

- This expression has three components: smoothing force, balloon force, image attraction force

- Balloon force: decides if the curve is going to inflate or deflate

- *v* is the parameter to play with

# Morphological snake
For nodule

- Let’s take a look at the final output first

# Morphological snake

Geometric active contour

- Read image, extract first channel

```
> import cv2
> import numpy as np
> import matplotlib.pyplot as plt

> mam = cv2.imread('mama07ORI.bmp')
> mam = mam[:,:,0]/255.0
```

- Calculate inverse Gaussian gradient:

```
> import morphsnakes as ms
> invg= ms.inverse_gaussian_gradient(mam,
                                      alpha=1000,
                                      sigma=5.48)
```

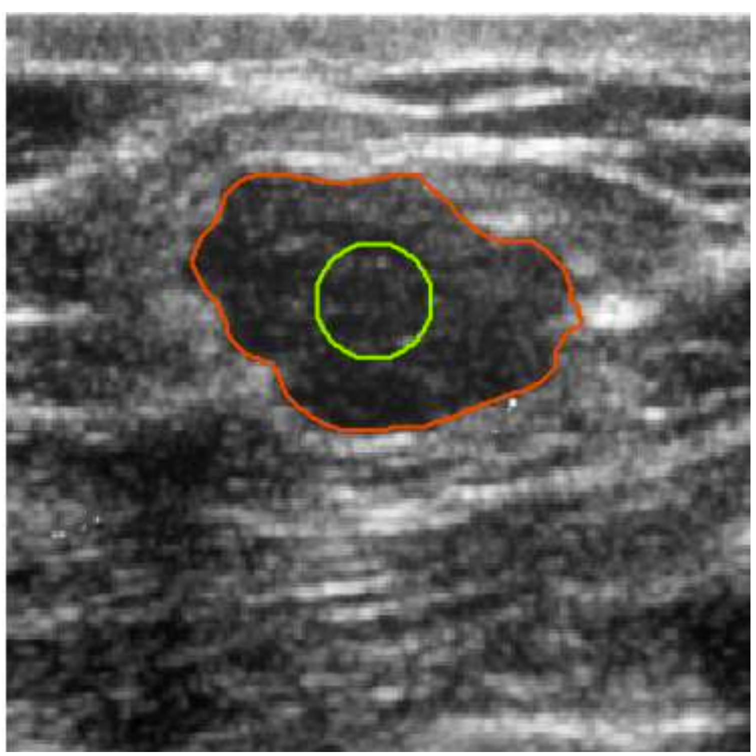
control the steepness in the transition between flat areas and borders (alpha)

standard deviation of the Gaussian filter (sigma)

- Create initial level set:

```
> ls0 = ms.circle_level_set(mam.shape,
                            (100, 126),
                            20)
```

image shape; center (y, x); radius



# Morphological snake

## Geometric active contour

- Create a visualization callback

```
> callback = ms.visual2d(mam)
```

- Perform segmentation

```
> lsf = ms.morphological_geodesic_active_contour(invg,
                                                 iterations=45,
                                                 init_level_set=ls0,
                                                 smoothing=1,
                                                 threshold=0.31,
                                                 balloon=1,
                                                 iter_callback=callback)
```

smoothing: number of times smoothing operator is applied per iteration

threshold: areas of image with value smaller than threshold will be considered borders

balloon: balloon force; positive for inflation, negative for deflation

lsf

- To plot correctly using 'cv2plt':

```
> cv2plt(lsf*255)
```

# Morphological snake

Active contour without edges

- Read image

```
> lak = cv2.imread('lakes3.jpg')
> lakg= cv2.cvtColor(lak,
                     cv2.COLOR_BGR2GRAY)/255.0
```

- Create initial level set:

```
> ls0 = ms.circle_level_set(lakg.shape,   image shape
                            (80, 170),    center (y, x)
                     radius 25)
```

- Create a visualization callback

```
> callback = ms.visual2d(lak)
```

lake3.jpg

# Morphological snake

For lake

- Let's take a look at the final output first

# Morphological snake

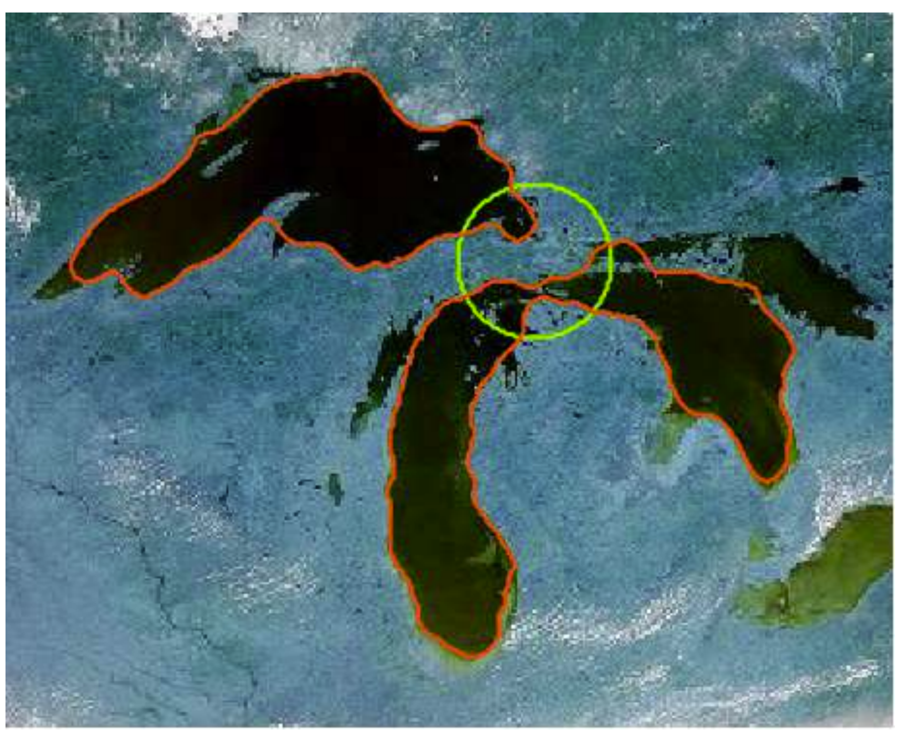
Active contour without edges

- Perform segmentation

```
> lsf = ms.morphological_chan_vese(lakg,
                                   iterations=200,
                                   init_level_set=ls0,
                                   smoothing=3,
                                   lambda1=1,
                                   lambda2=1,
                                   iter_callback=callback)
```

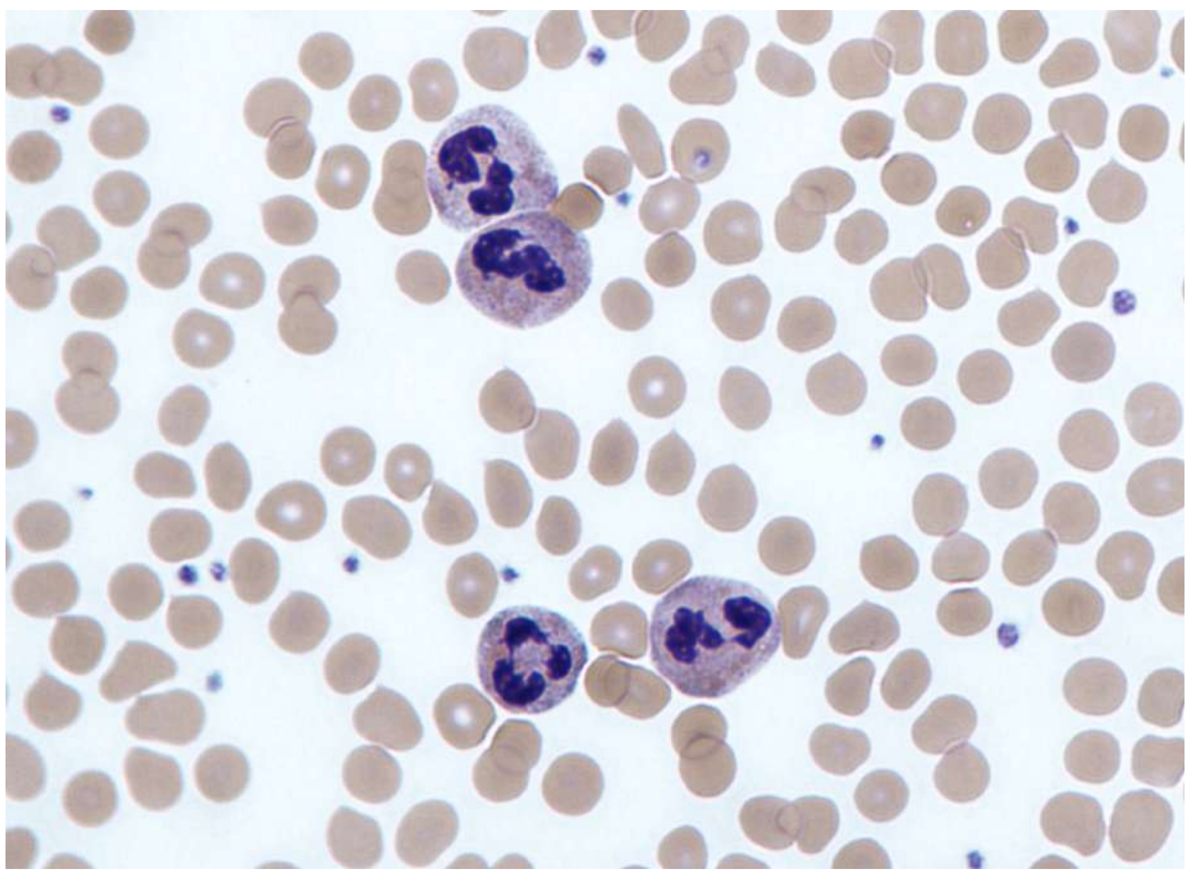
weight parameter for outer region: `lambda1=1`

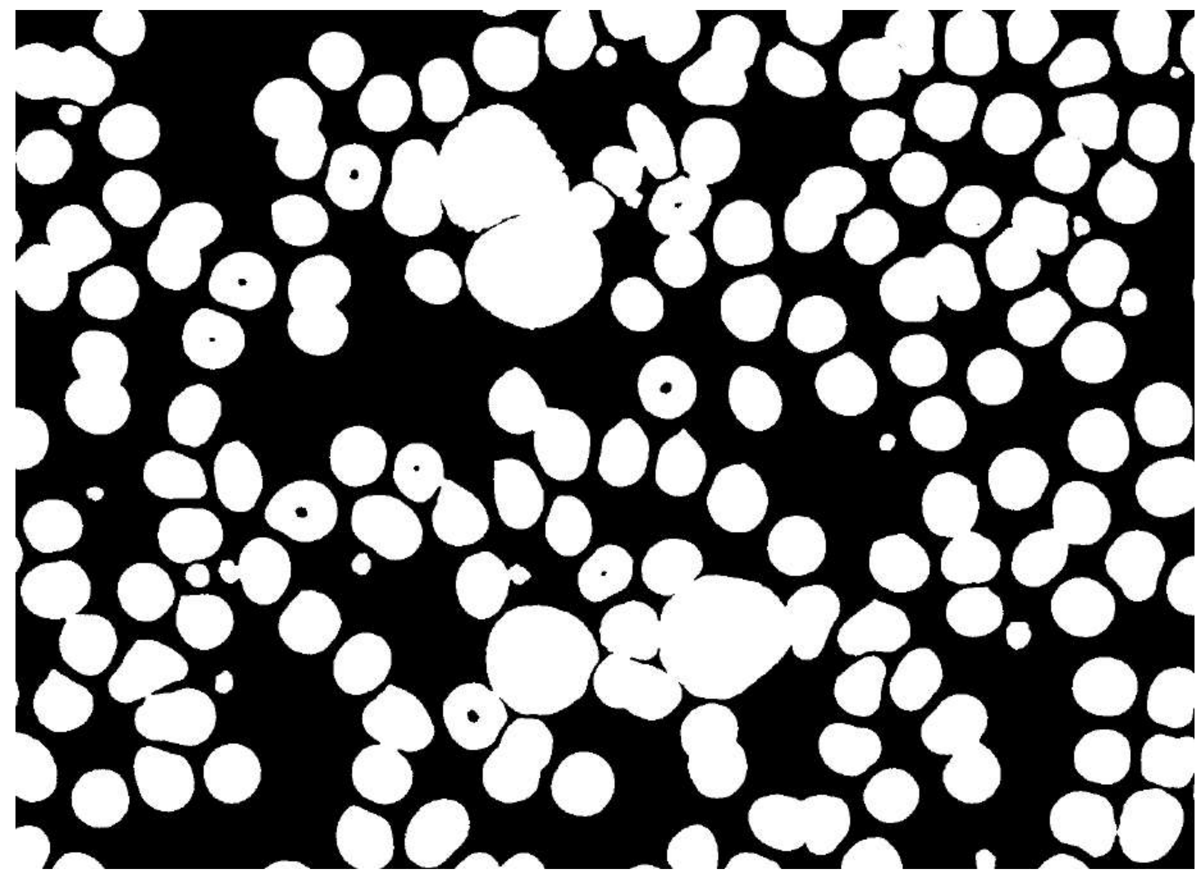
weight parameter for inner region: `lambda2=1`



# Morphological snake
On a previous problem

- Holes in red blood cells; how to avoid?





# Morphological snake

The code

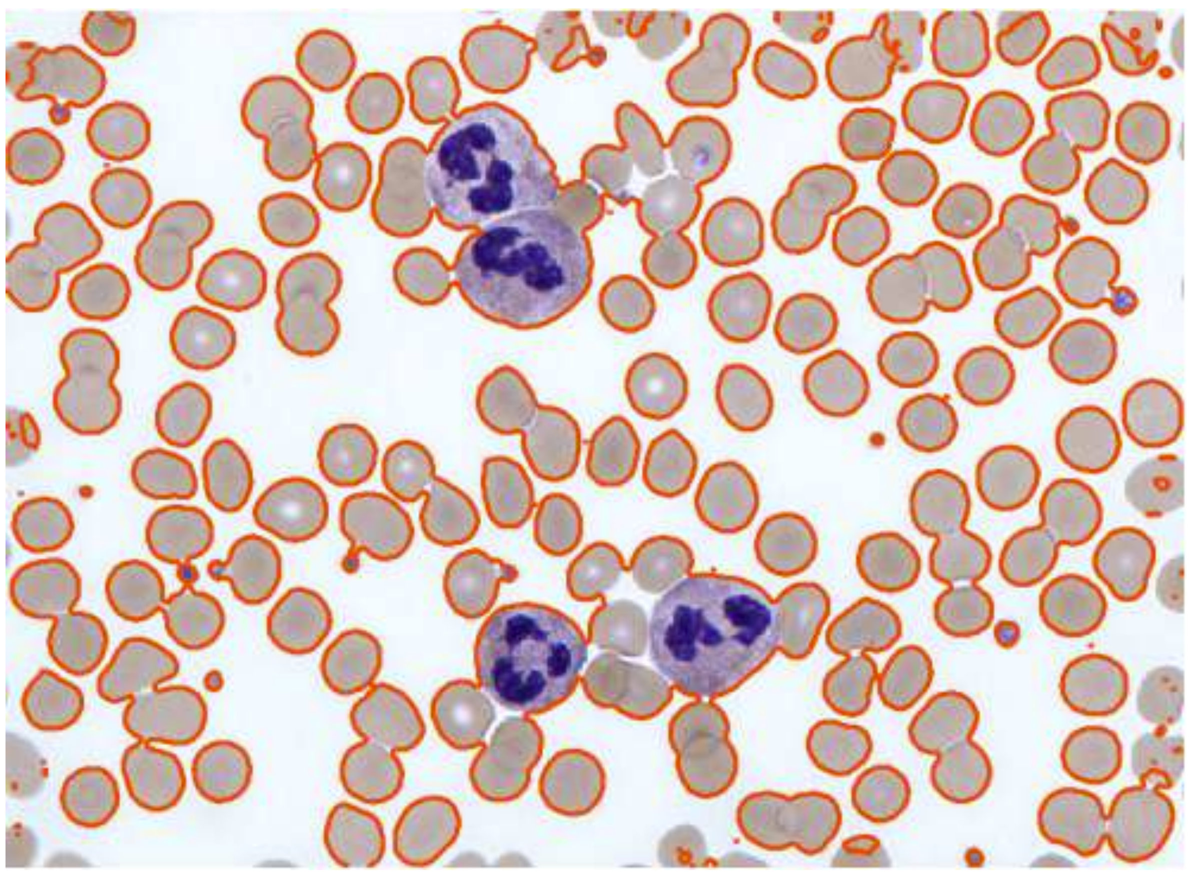
```
> neu      = cv2.imread('neu.jpg')
> neu      = cv2.resize(neu,(728,530),interpolation=cv2.INTER_LINEAR)

> neug     = cv2.cvtColor(neu,cv2.COLOR_BGR2GRAY)/255.0
> invg     = ms.inverse_gaussian_gradient(neug, alpha=700, sigma=1)

> ls0      = ms.circle_level_set(neug.shape, (265, 364), 450)

> callback    = ms.visual2d(cv2.cvtColor(neu,cv2.COLOR_BGR2RGB))
> lsf         = ms.morphological_geodesic_active_contour(invg,
                                                       iterations=400,
                                                       init_level_set=ls0,
                                                       smoothing=1,
                                                       threshold=0.5,
                                                       balloon=-1,
                                                       iter_callback=callback)
```

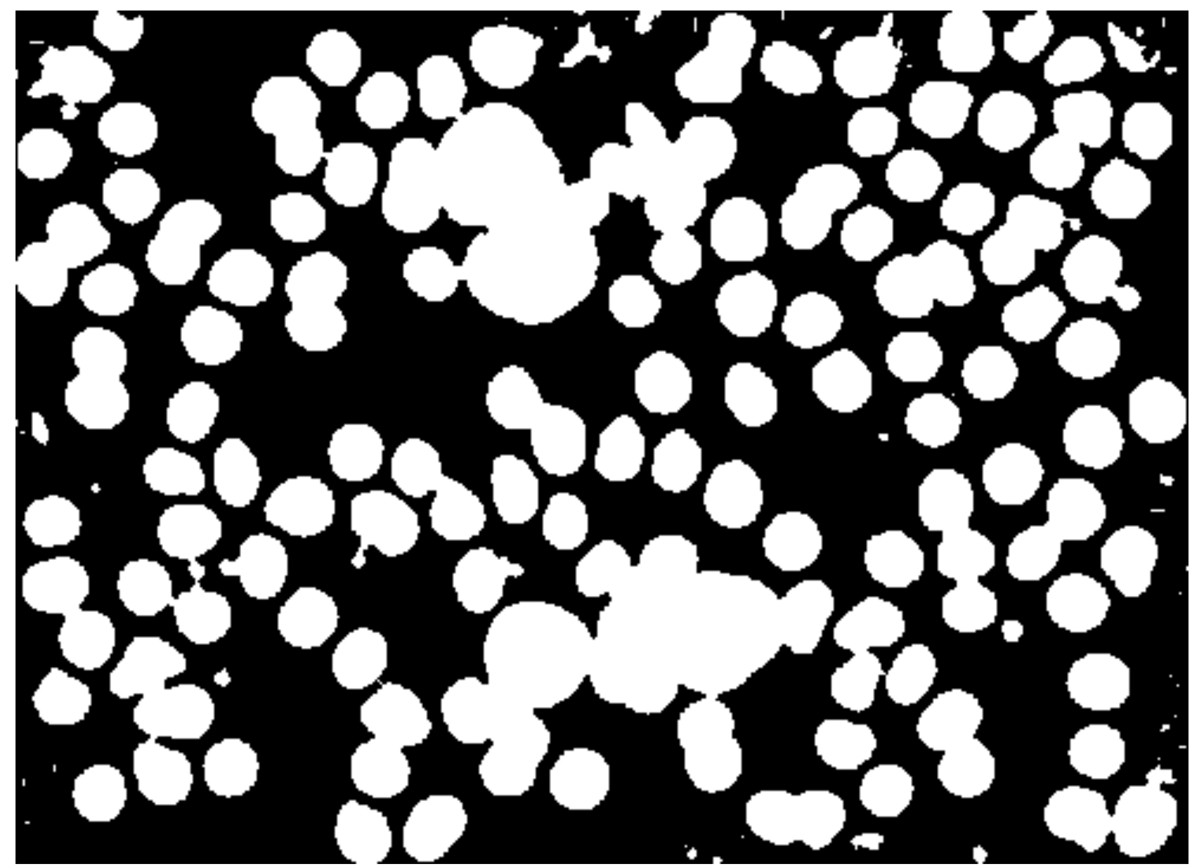
Negative value for balloon force to shrink contour

# Morphological snake

On a previous problem

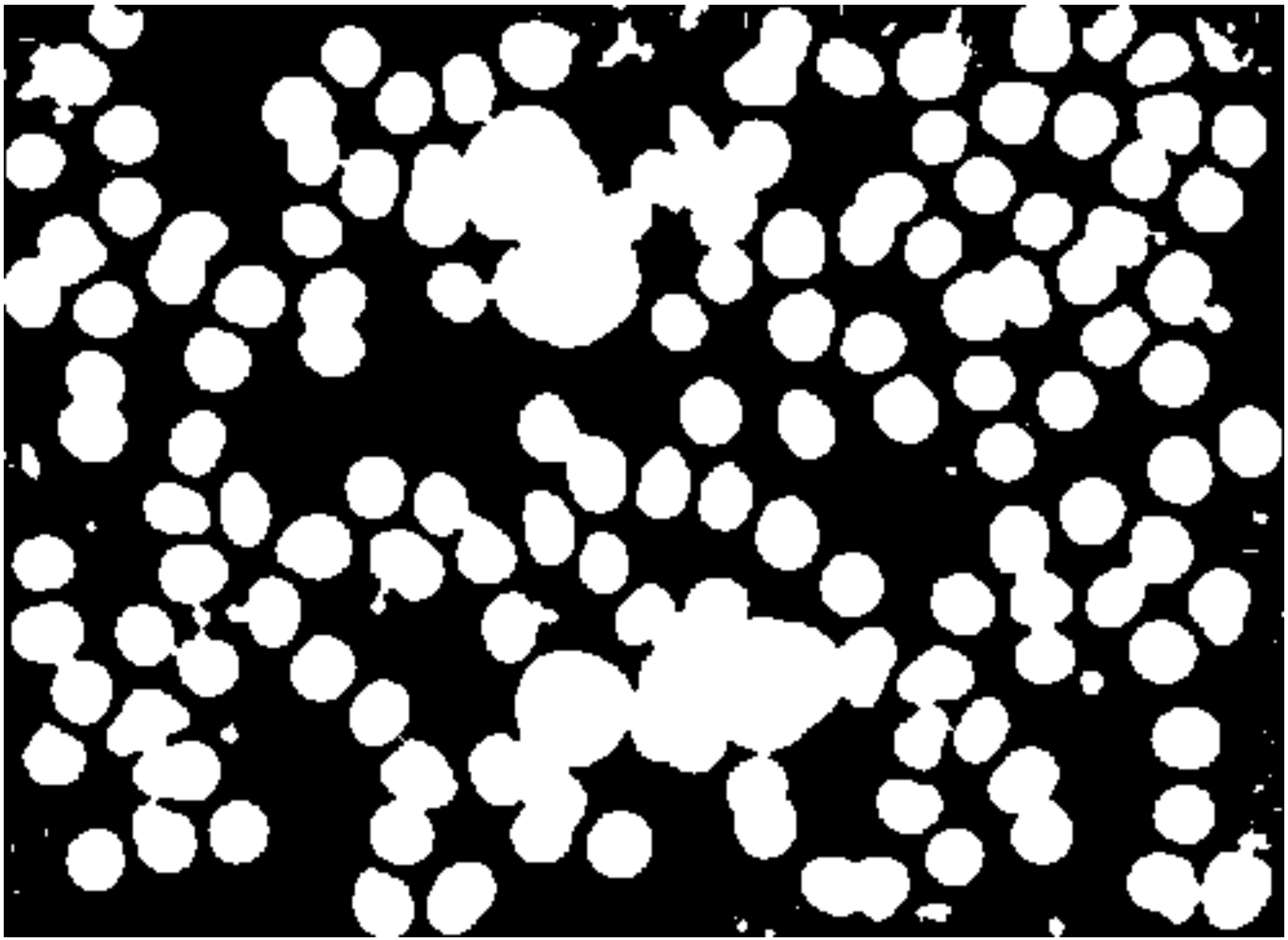
- No holes in red blood cells, but some red blood cells are not captured
- Reality: no single best solution, often need to combine two or more steps/methods





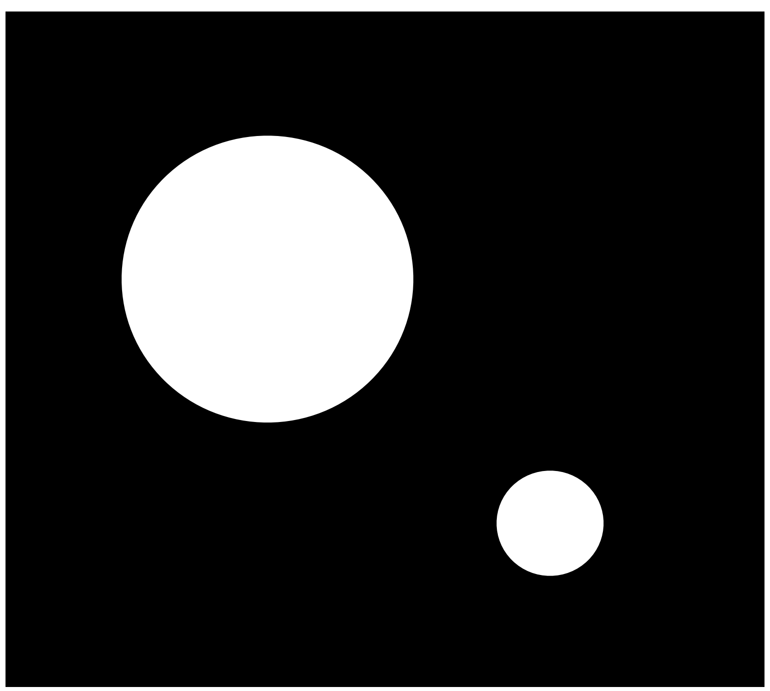
# How to remove noise

On a previous problem



## Removing noise
A simple example

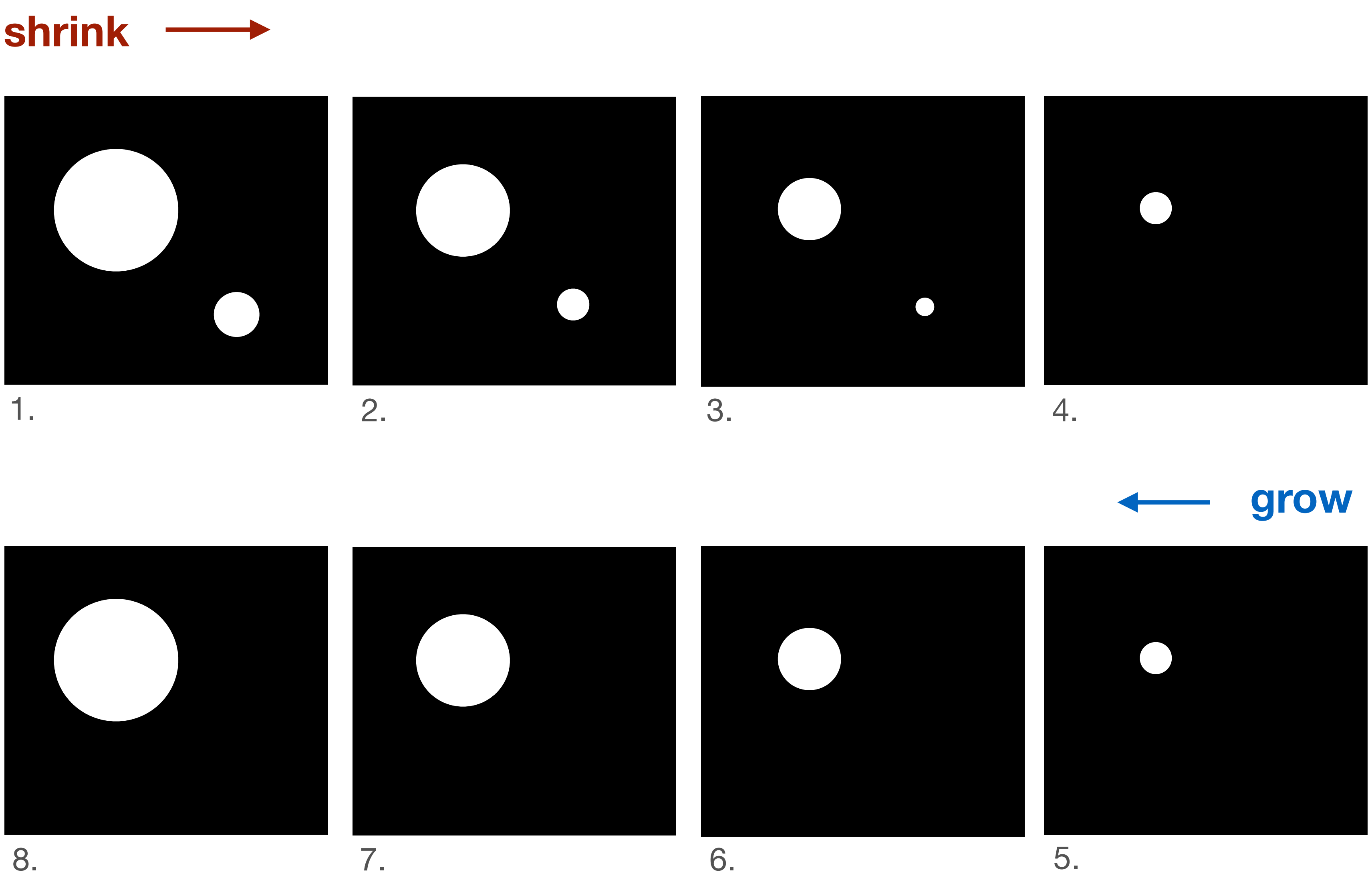
- Assume the bottom left is an image.
- Problem: How can we remove the small circle (considered as noise) without affecting a lot on the big circle (the area of interest)



## Removing noise

A simple example

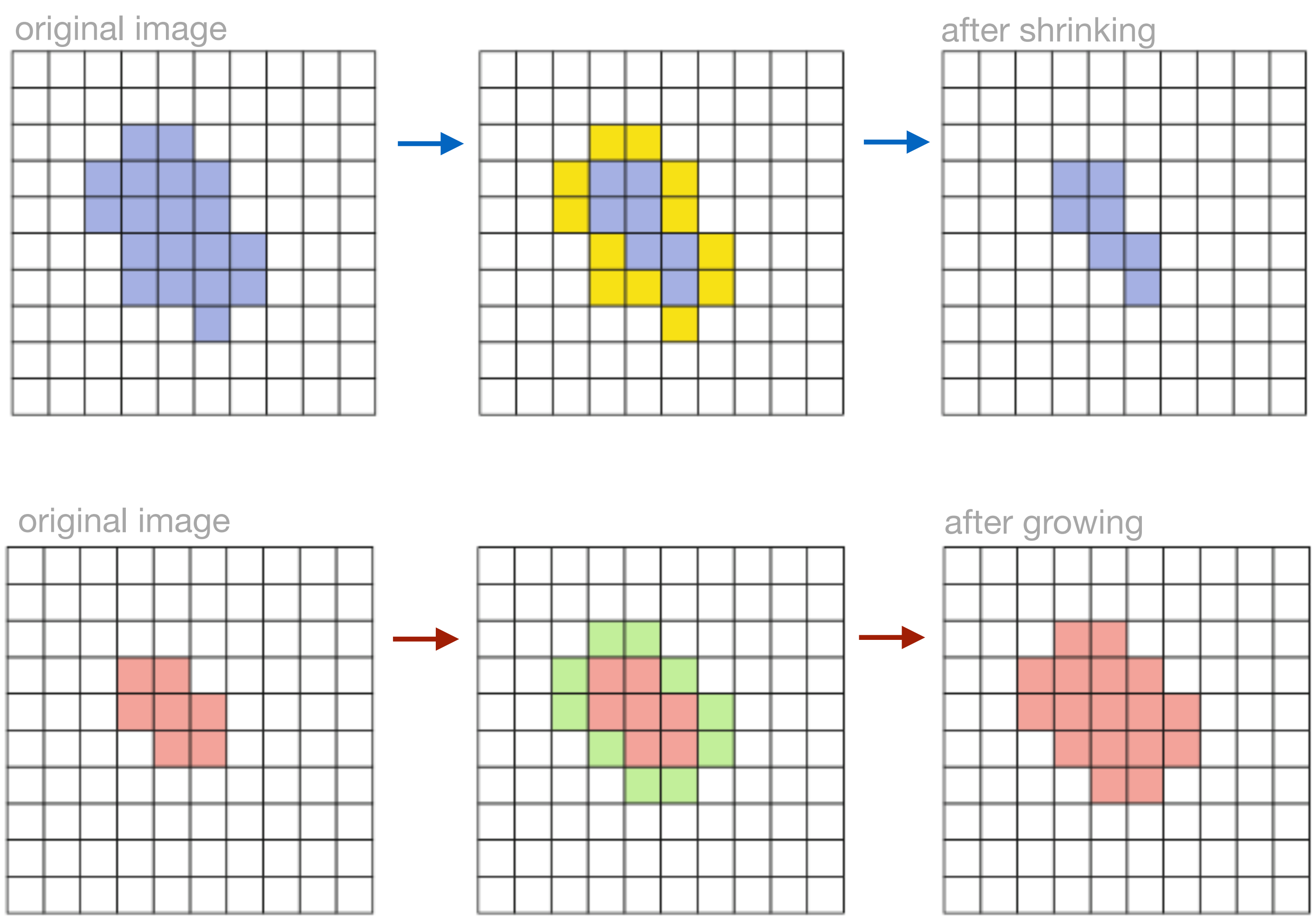
- The strategy: shrink all the regions in the image to the extent that the small circle vanishes, then grow back the remaining region



# Morphological operation

Shrinking, growing

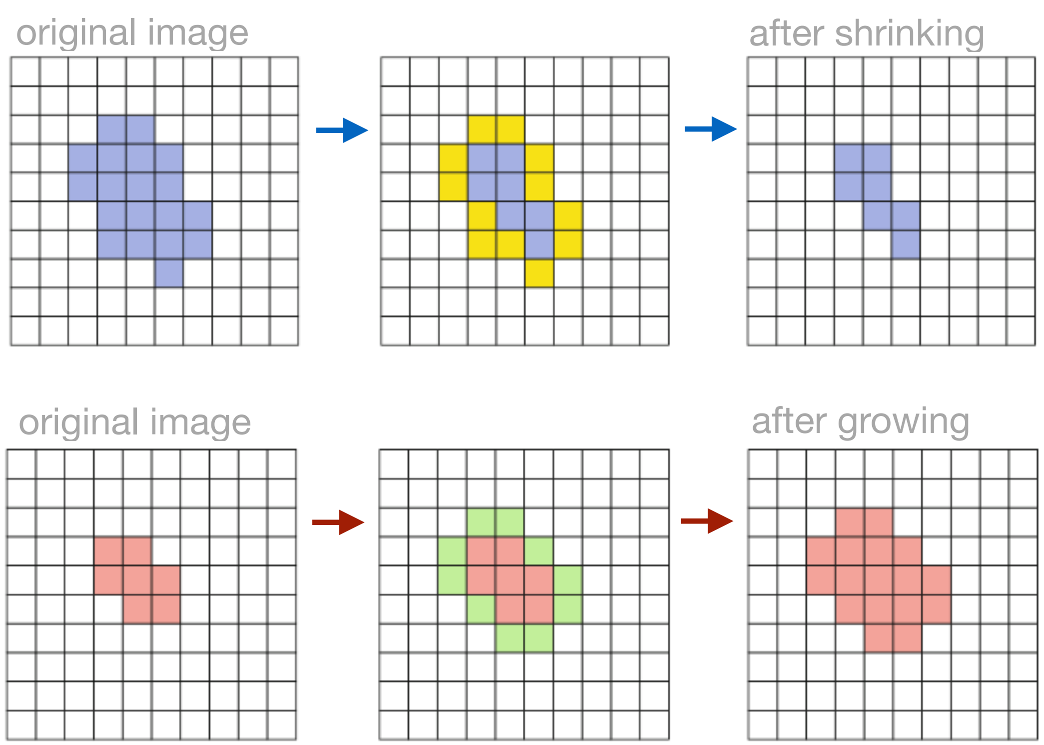
- Shrinking is performed by removing a layer of border pixels
- Growing is performed by attaching a layer of pixels




Source: Digital Image Processing by Burger and Burge, 2016

# Morphological operation

Erosion, dilation

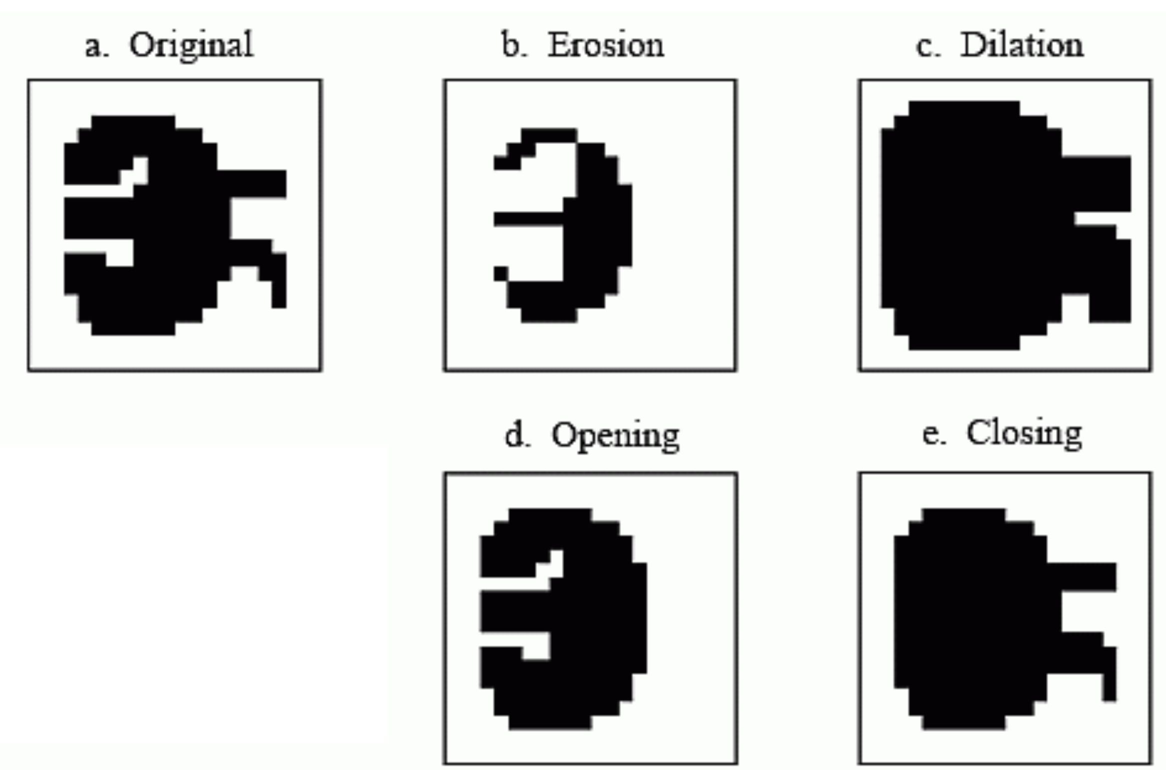



Source: Digital Image Processing by Burger and Burge, 2016

- Both the shrinking and growing are achieve through morphological operation
- Specifically, erosion is the morphological operation we use to shrink areas in image
- Dilation is the morphological operation we use to grow areas in image
- Morphological operation involves kernel / structuring element; the size and the shape of kernel affects the rate and the style of shrinking and growing

# Morphological operation

## Composite morphological operation




Source: http://www.dspguide.com/ch25/4.htm

- Dilation and erosion are often used together
- If an erosion is followed by a dilation using the same kernel / structuring element, this composite morphological operation is called opening
- If a dilation is followed by an erosion using the same kernel / structuring element, this composite morphological operation is called closing
- In reality, the use of morphological operations can be pretty ad hoc, i.e. each application requires a custom solution by trial-and-error
- It is usually more of an art than a science

# Morphological operation

Five main composite types

- In opencv, there are quite a few composite morphological operation ready to use

Opening

Closing

Gradient

Top hat

Black hat

# Morphological operation

## Hat and gradient

- Top hat composite morphological operation is defined as the difference between input image and its opening

- Black hat composite morphological operation is defined as the difference between input image and its closing

- Gradient composite morphological operation is defined as the difference between the dilation and the erosion of a given image

Top hat

Black hat

Gradient

# How to remove noise

Solution

- Use morphological operation: opening

- First, create kernel

```
> kernel  = np.ones((4,4),np.uint8)
```

- Perform opening

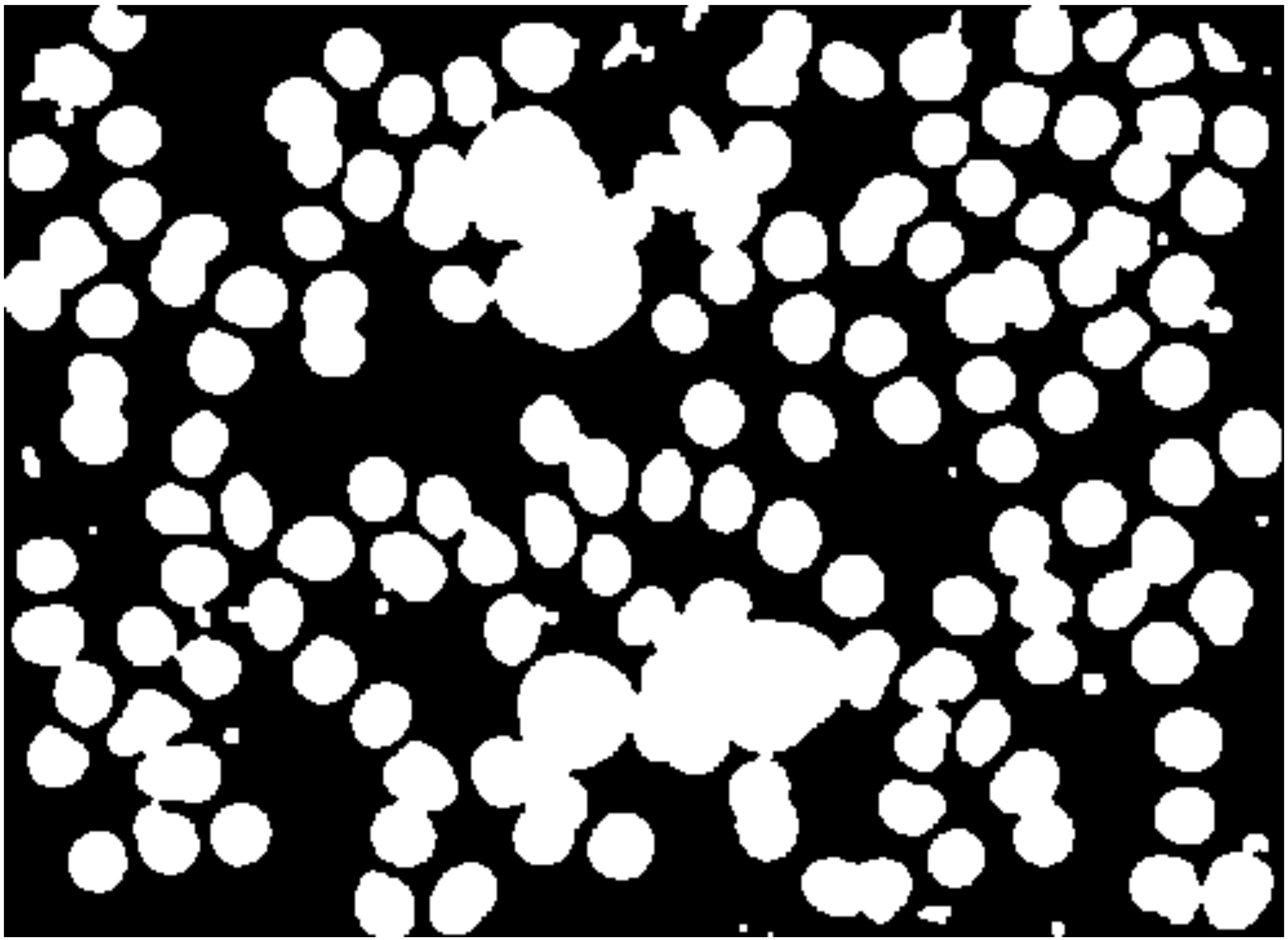
```
> final   = cv2.morphologyEx(np.uint8(lsf),
             operation type  cv2.MORPH_OPEN,
                             kernel)
```

- Available operation type:

cv2.MORPH_OPEN
cv2.MORPH_CLOSE
cv2.MORPH_GRADIENT
cv2.MORPH_TOPHAT
cv2.MORPH_BLACKHAT

# How to remove noise

## Outcome



# First, a workshop, then a challenge

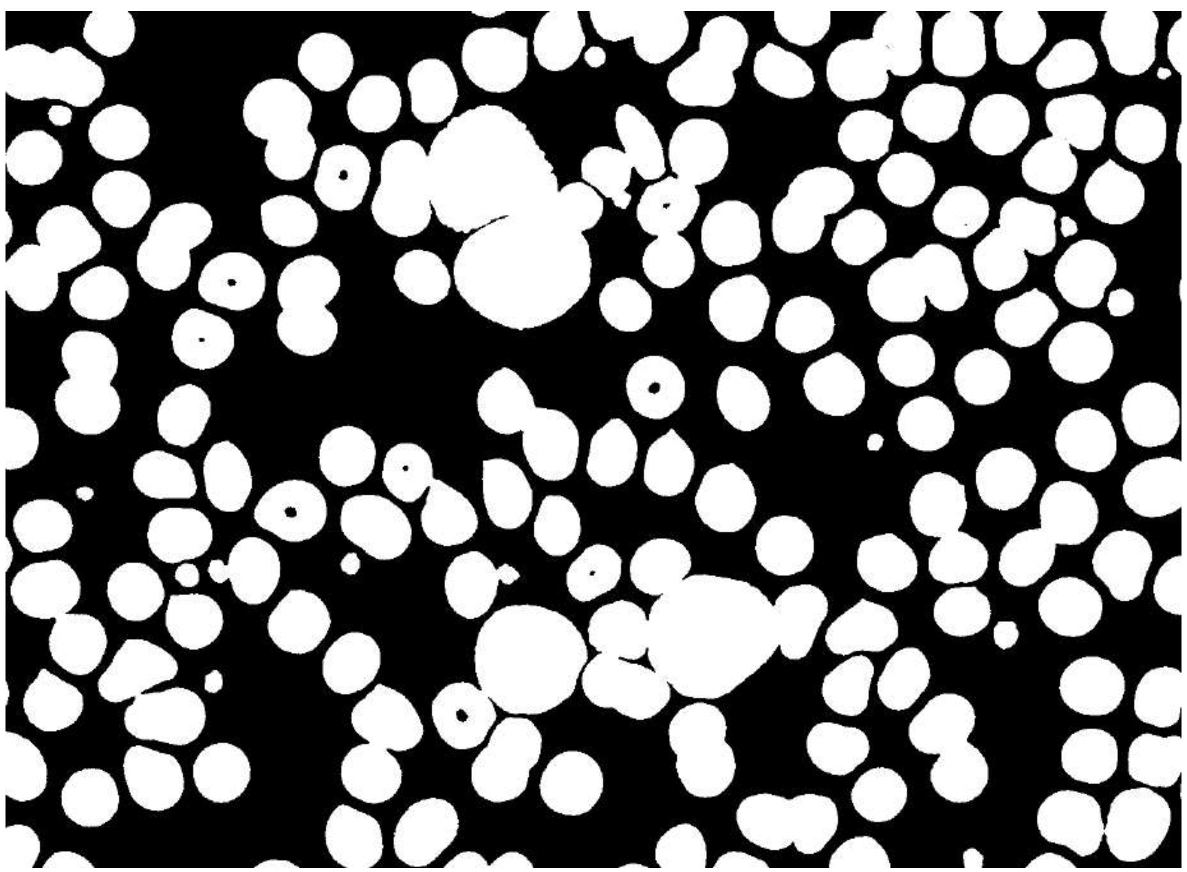
## Challenge

More than a technique

- The segmentation by thresholding gets holes in some cells; the segmentation by snake miss certain cells around the border

- How about combine both techniques to get better output ?

Segmentation by thresholding



Segmentation by morphological snake