Optics, Forces and Development

Session 08

Digital Image Processing II

Image Analysis, Part 1

Jorge Jara
DCC, SCIAN-Lab
jjaraw@gmail.com

Temario

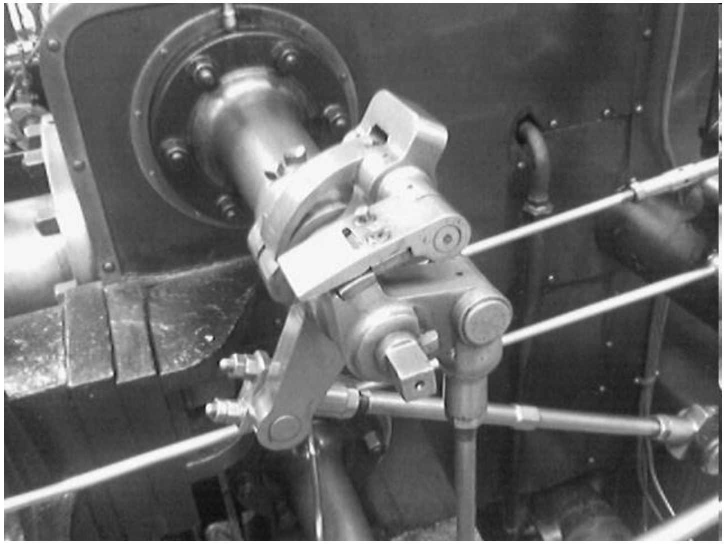
1. Segmentación - métodos (filtros) básicos
   – Umbrales
   – Basados en convolución matricial
   – Morfológicos
   – Filtros de Fourier

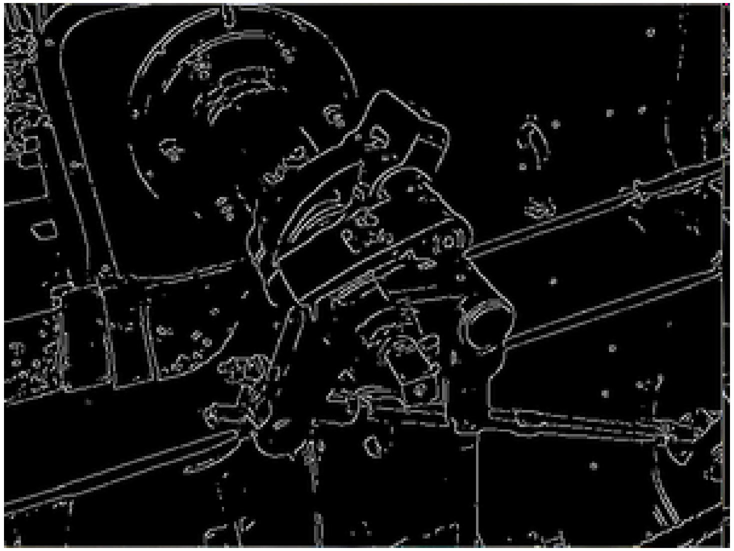
2. Segmentación - métodos avanzados
   – Ajuste de formas (*pattern matching*)
   – Modelos deformables o contornos activos
     • Paramétricos
     • Implícitos (sesión del sábado)

Segmentación – métodos básicos

- Filtros

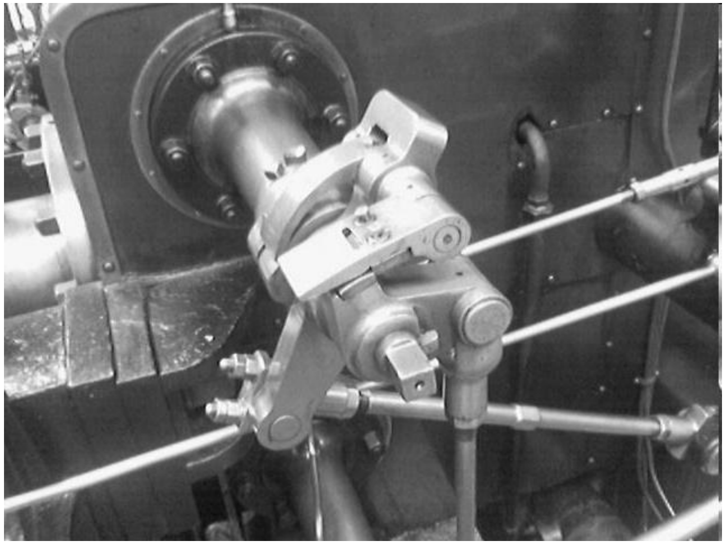
imagen → filtro → imagen filtrada

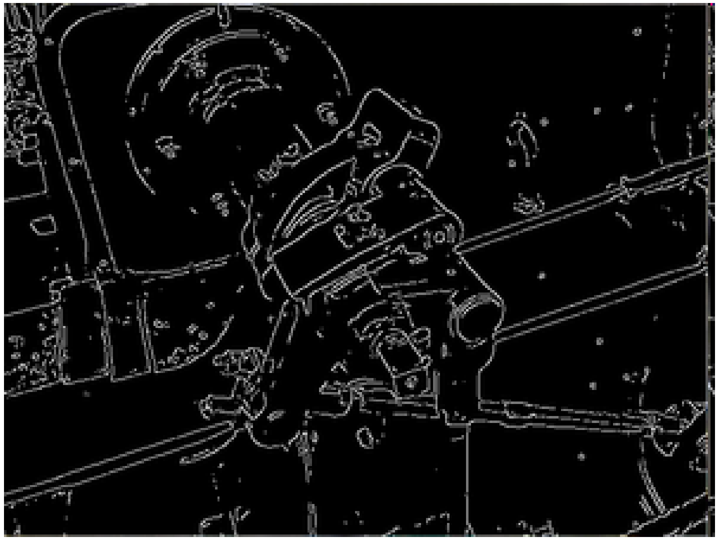


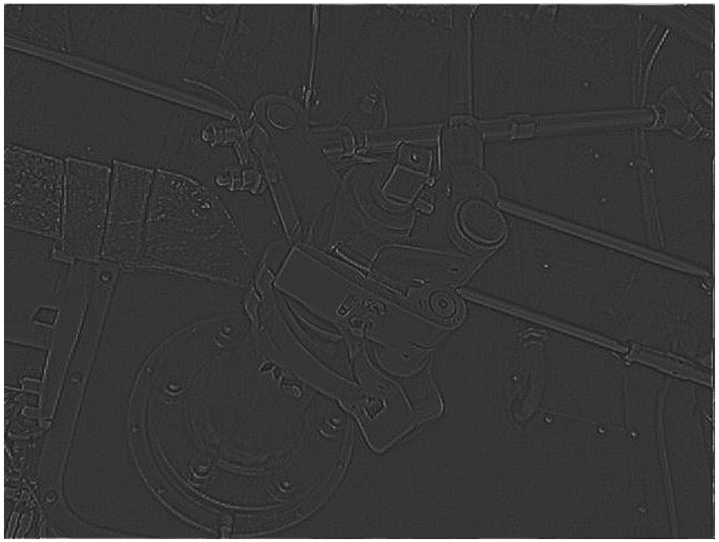


Imágenes de Wikipedia

- Ejemplos de filtros







También pueden definirse mediante su función de transferencia…
•Banda(s) de frecuencia
•Respuesta al impulso

Segmentación – métodos básicos

Umbrales
- Otsu

Basados en convolución
- Gradiente (Sobel, Roberts, …)
- Laplace
- Gausiano

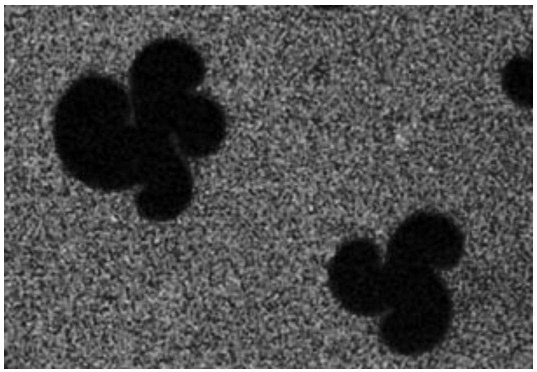
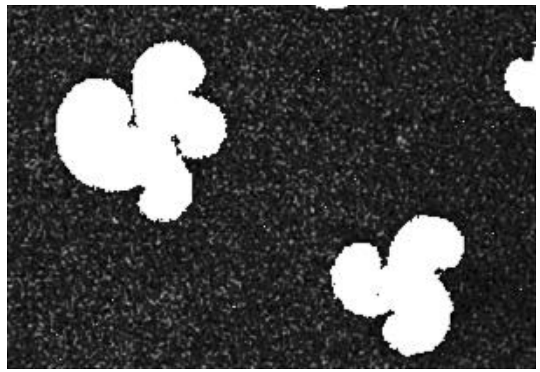
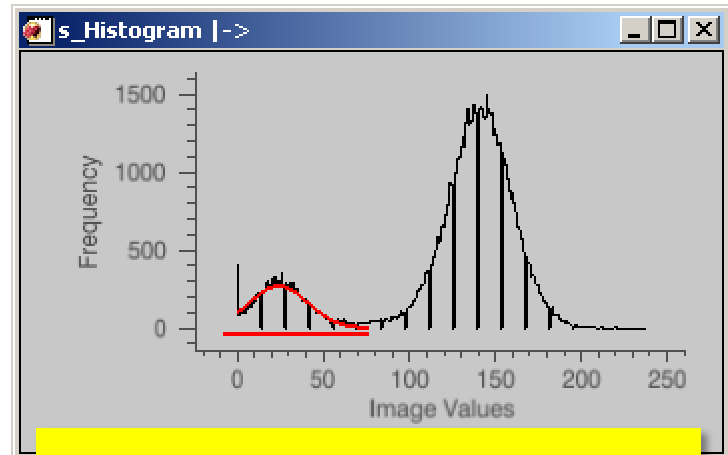
Morfológicos
- Morfología matemática
- Tamaño
- Adelgazamiento (*thinning*, *skeletonization*) *

Operaciones aritméticas-lógicas
- AND, OR, XOR
- Unión, intersección

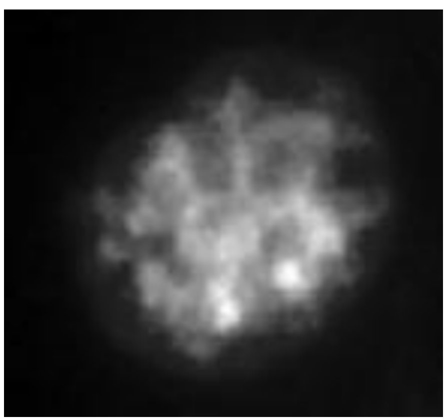
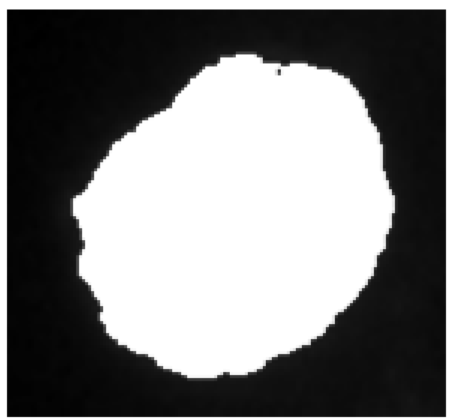
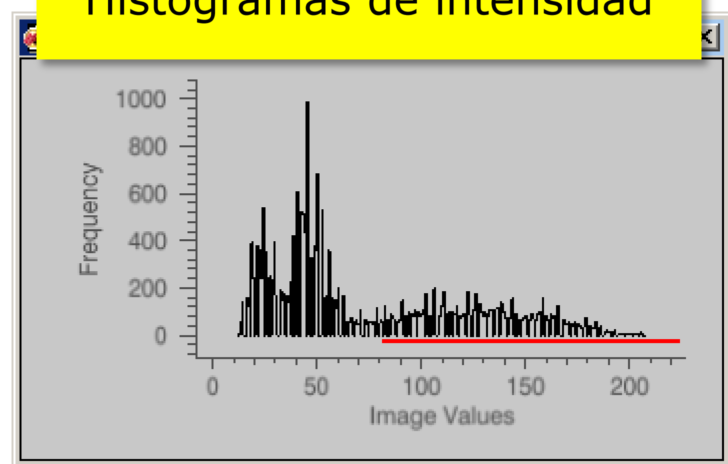
Y un largo etc.

Segmentación – métodos básicos

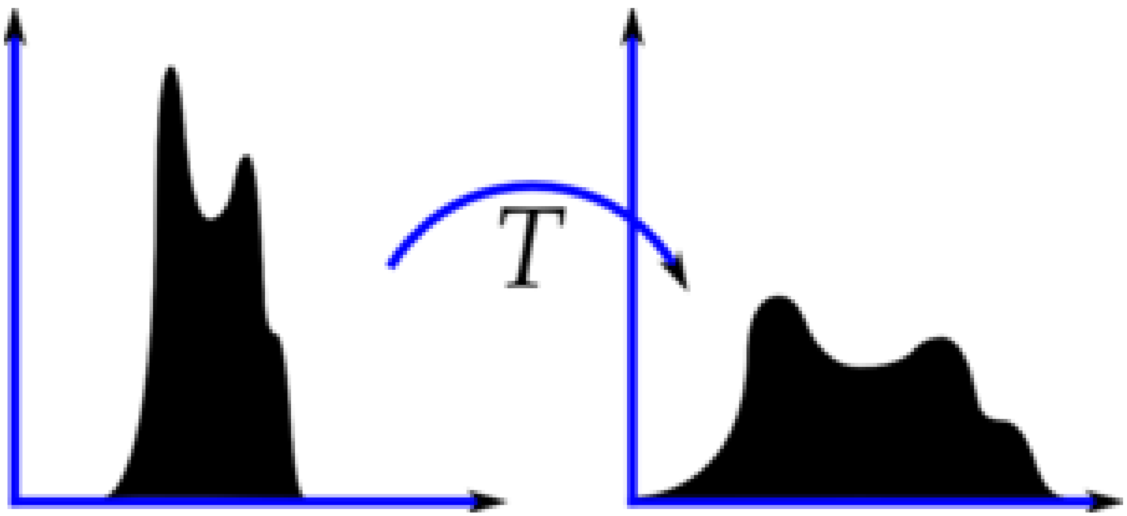
- Filtro de umbral
  segmentación: ROIs (blanco) / fondo (negro)

Histogramas de intensidad

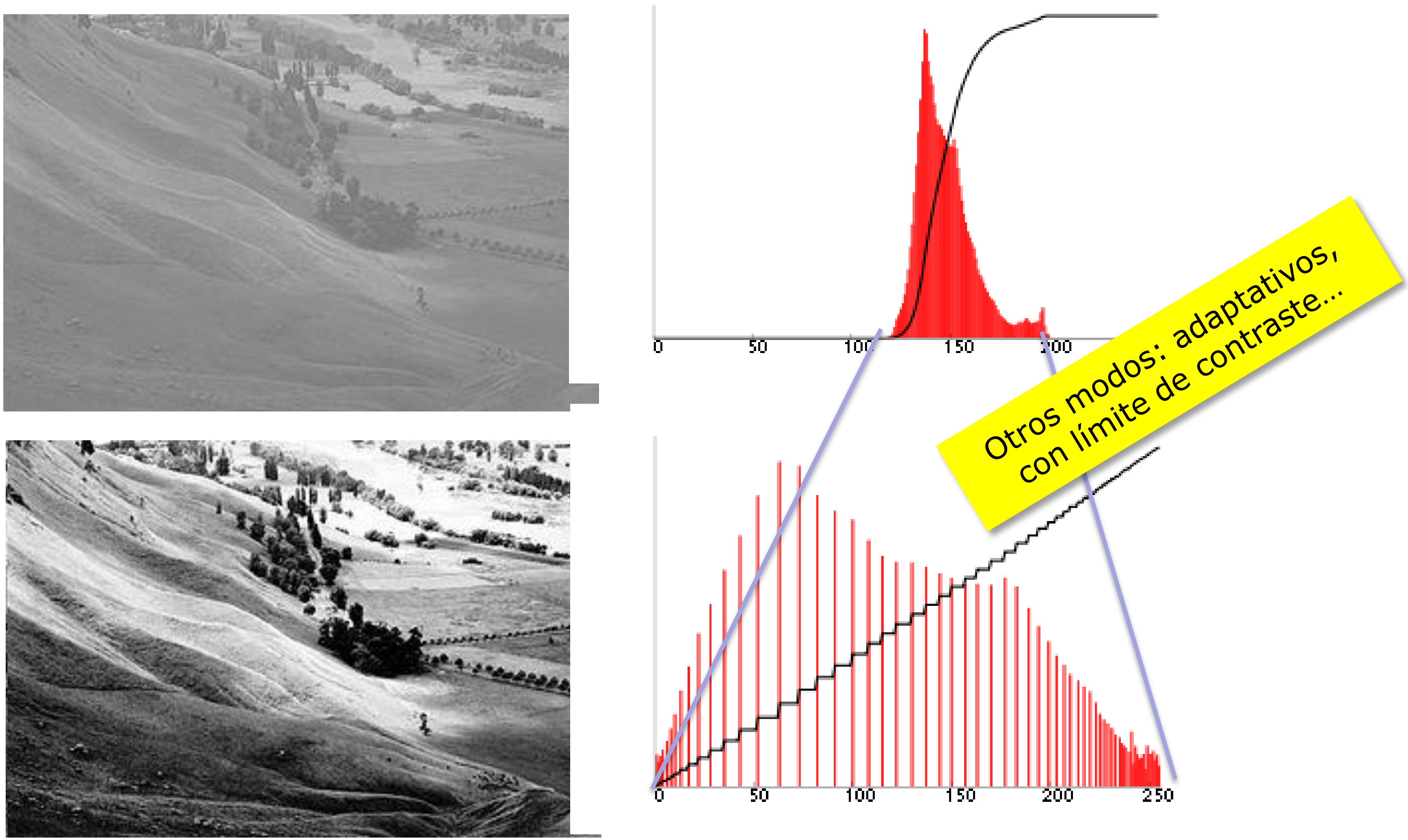
- Ecualización de histograma



http://en.wikipedia.org/wiki/Histogram_equalization

Segmentación – métodos básicos

- Ecualización de histograma básica



http://en.wikipedia.org/wiki/Histogram_equalization

- Umbral de Otsu
  - Idea: separar los píxeles de una imagen en dos conjuntos, con la varianza de intensidad mínima dentro de cada clase

$$\omega_1(t)=\sum_{i=0}^{t} p(i)$$

$$\mu_1(t)=\sum_{i=0}^{t} p(i)x(i)$$

$$\omega_2(t)=\sum_{i=t+1}^{i\max} p(i)$$

$$\mu_1(t)=\sum_{i=t+1}^{i\max} p(i)x(i)$$

$$\min\ \sigma_w^2(t)=\omega_1(t)\sigma_1^2(t)+\omega_2(t)\sigma_2^2(t)$$

$$t:\text{threshold},\quad \omega_i:\text{probability of class } i$$

Algorithm

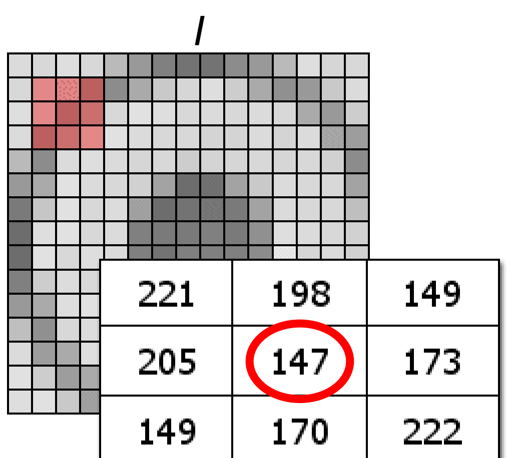
1. Compute histogram and probabilities of each intensity level
2. Set up initial $\omega_i(0)$ and $\mu_i(0)$
3. Step through all possible thresholds $t=1\ldots$ maximum intensity
   1. Update ω_i and μ_i
   2. Compute $\sigma_b^2(t)$
4. Desired threshold corresponds to the maximum $\sigma_b^2(t)$
5. You can compute two maximums (and two corresponding thresholds). $\sigma_{b1}^2(t)$ is the greater max and $\sigma_{b2}^2(t)$ is the greater or equal maximum
6. Desired threshold = $\dfrac{threshold_1+threshold_2}{2}$

N. Otsu (1979).

- Convolución
  - Muchos filtros se basan en esta operación
    http://en.wikipedia.org/wiki/Convolution
    http://mathworld.wolfram.com/Convolution.html
- La convolución matricial es una operación **similar**, realizada entre dos matrices
  - la imagen, *I*
  - un *kernel*, *K*

K

-1	0	1
-2	0	2
-1	0	1



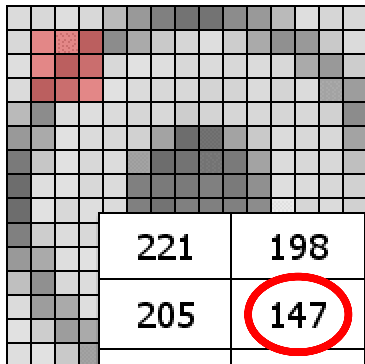
$$
\begin{aligned}
(K \otimes I)_{i,j} = & \ (-1 \cdot 221) \\
& + (\ 0 \cdot 198) \\
& + (\ 1 \cdot 149) \\
& + (\ 0 \cdot 205) \\
& + (\ 0 \cdot \mathbf{147}) \\
& + (\ 0 \cdot 173) \\
& + (-1 \cdot 149) \\
& + (\ 0 \cdot 170) \\
& + (\ 1 \cdot 222) = -63
\end{aligned}
$$

Adaptado de James Matthews, 2002
http://www.generation5.org/content/2002/convolution.asp

K

-1	0	1
-2	0	2
-1	0	1

I



221	198	149
205	147	173
149	170	222

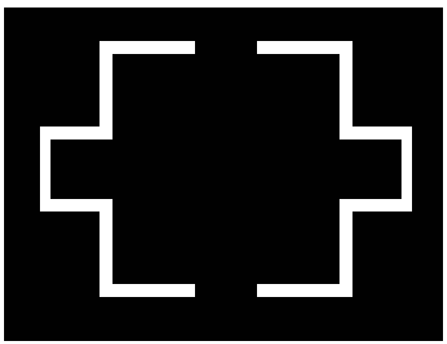
$$
\begin{aligned}
(K \otimes I)_{i,j} &= (-1 * 222) \\
&+ (\ 0 * 170) \\
&+ (\ 1 * 149) \\
&+ (-2 * 173) \\
&+ (\ 0 * \mathbf{147}) \\
&+ (\ 2 * 205) \\
&+ (-1 * 149) \\
&+ (\ 0 * 198) \\
&+ (\ 1 * 221) = +63
\end{aligned}
$$

La convolución matricial no siempre se implementa del mismo modo, lo que puede arrojar resultados distintos

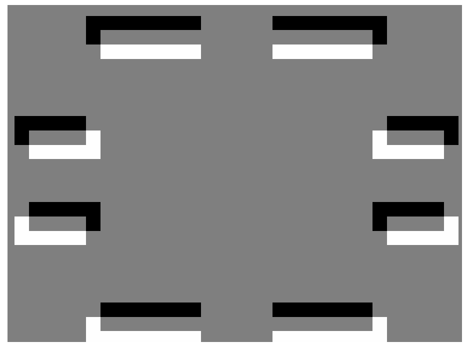
James Matthews, 2002
http://www.generation5.org/content/2002/convolution.asp

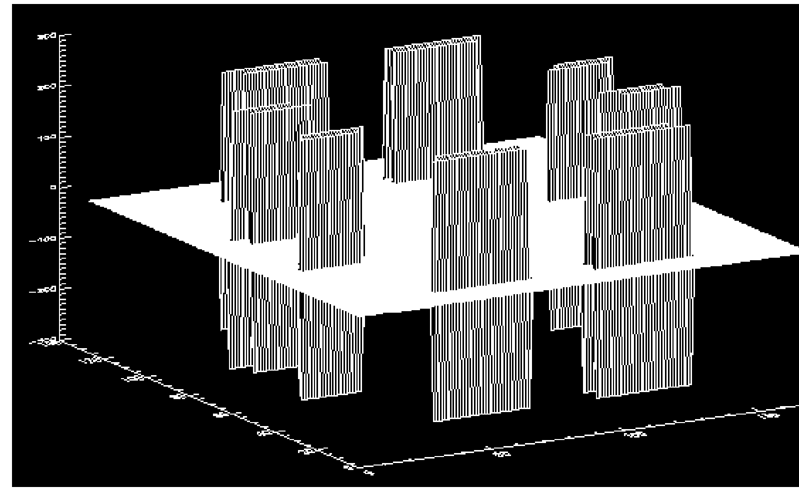
Segmentación – métodos básicos

- Gradientes de intensidad (aproximación discreta)

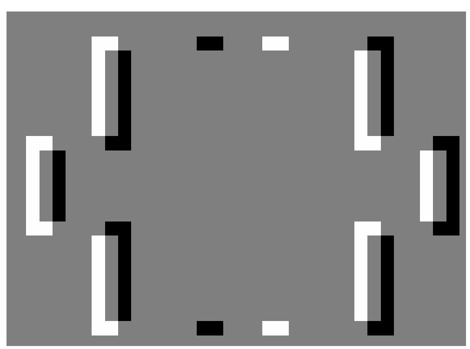


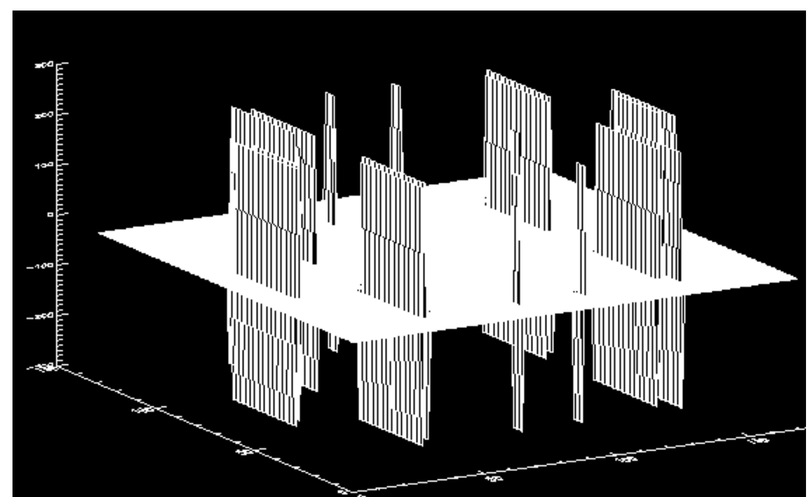
$$\frac{\partial I}{\partial x} \approx$$



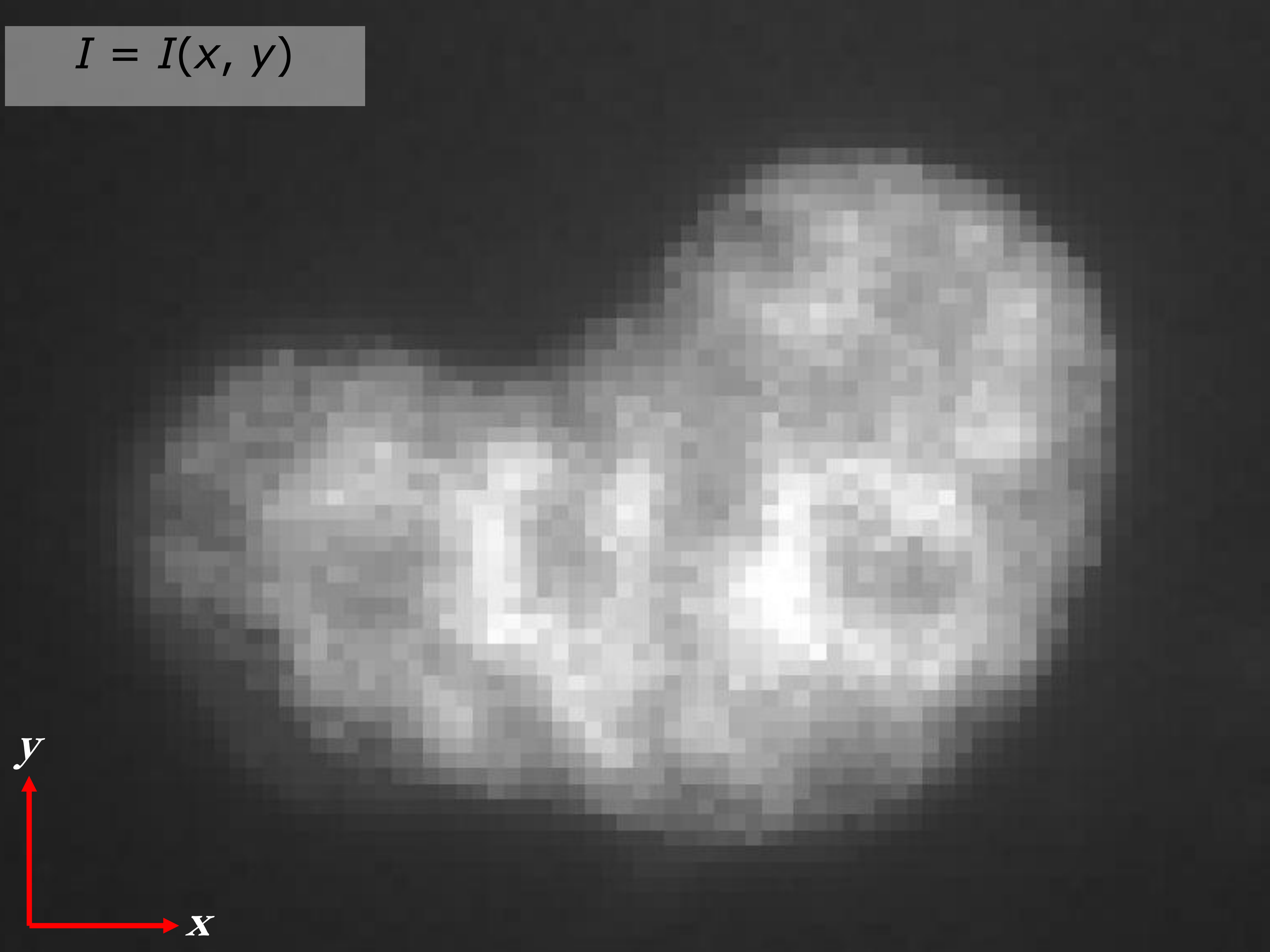


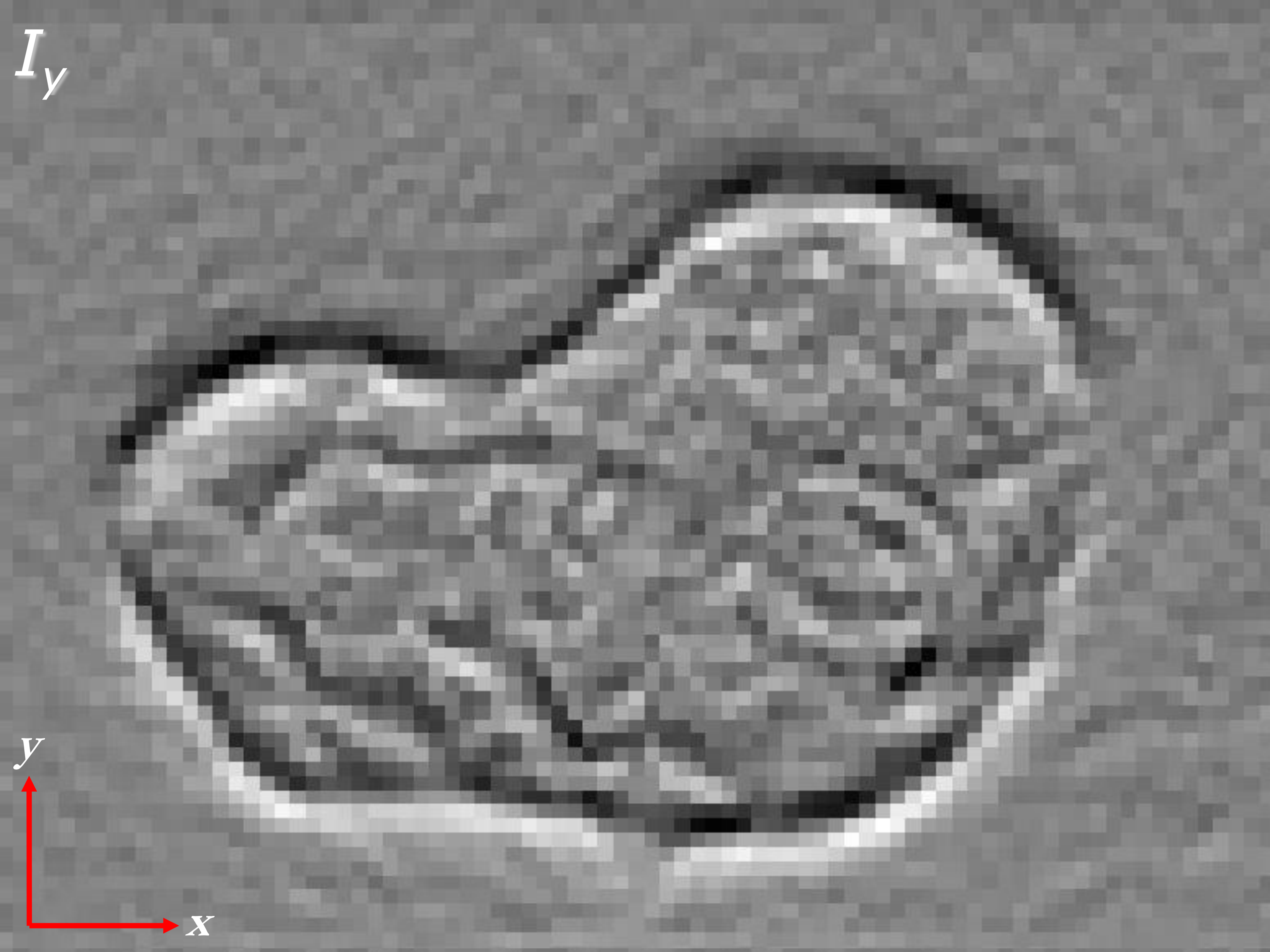
$$\frac{\partial I}{\partial y} \approx$$

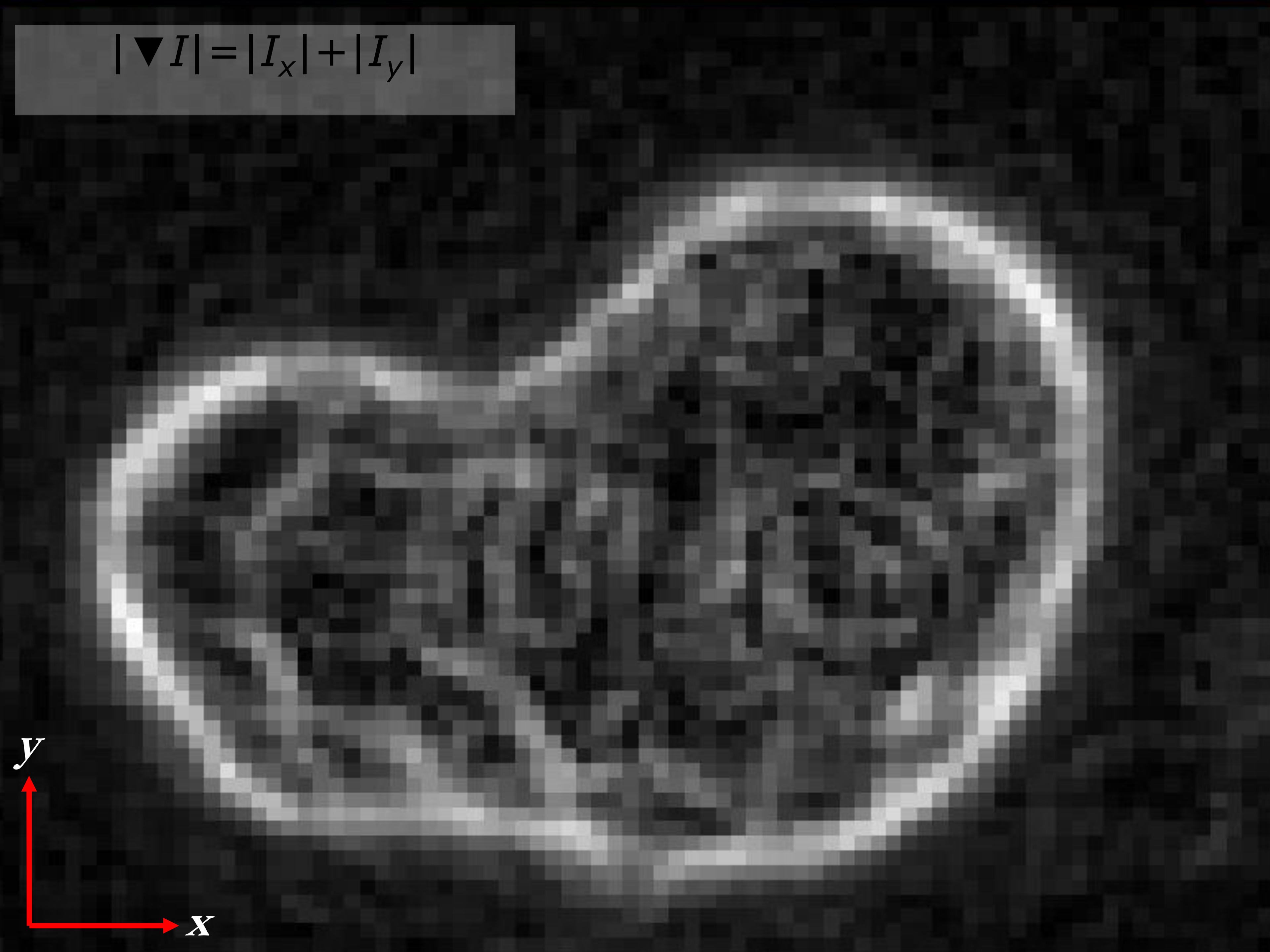




0	+255	-255

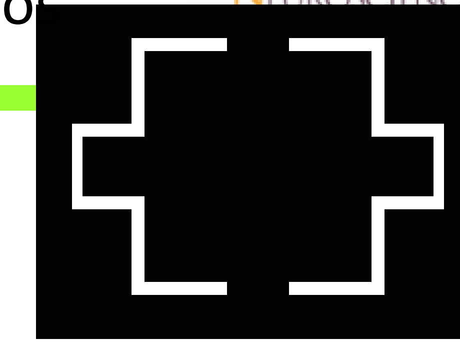

$I = I(x, y)$
y
x


I_y
y
x


$|\nabla I| = |I_x| + |I_y|$
y
x

Segmentación – métodos básicos

- Gradientes de intensidad (aprox. discreta)

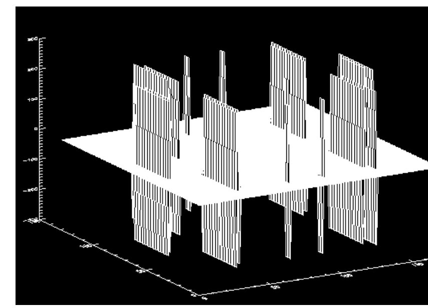


$$I = I(x,y)$$

$$\frac{\partial I}{\partial x} \approx \frac{I(x+\Delta x, y) - I(x,y)}{\Delta x} = Kx \otimes I$$

$\Delta x = 1\,\text{pixel}$

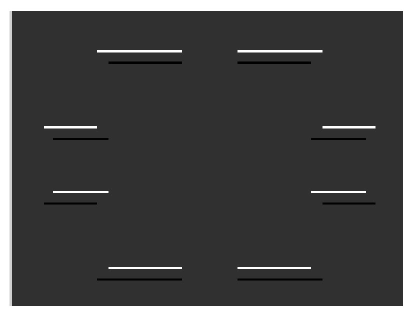
$$Kx = \begin{Bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{Bmatrix}$$

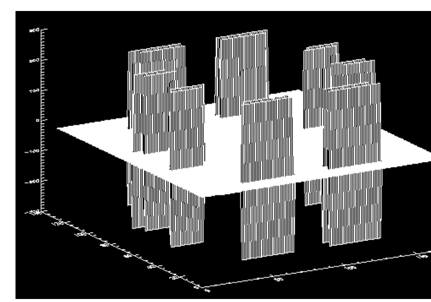


$$\frac{\partial I}{\partial y} \approx \frac{I(x, y+\Delta y) - I(x,y)}{\Delta y} = Ky \otimes I$$

$\Delta y = 1\,\text{pixel}$

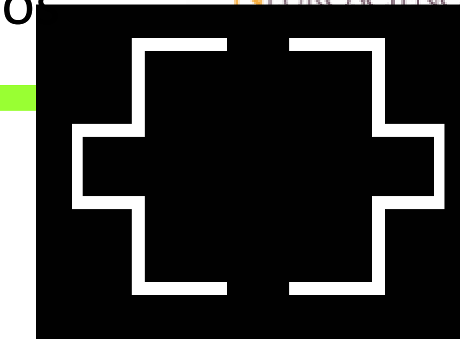
$$Ky = \begin{Bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{Bmatrix}$$





0 +255 -255

Segmentación – métodos básicos



$$I = I(x, y)$$

- Kernels…

Laplace

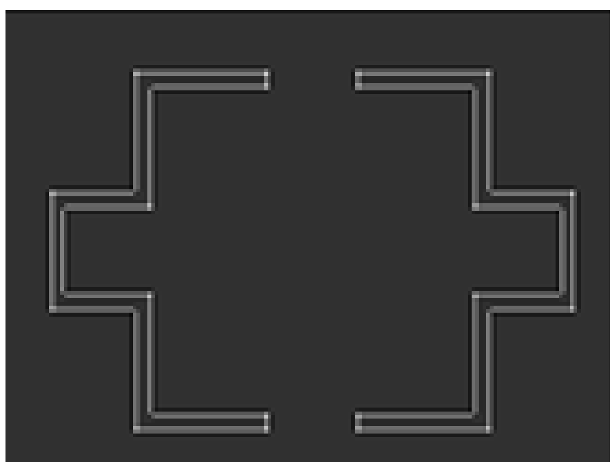
$$\nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

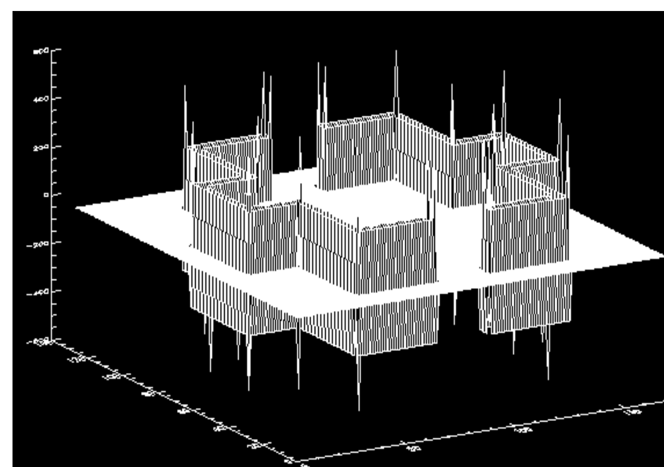
$$\nabla^2 I \approx \frac{f(x+\Delta x, y) - 2f(x,y) + f(x-\Delta x, y)}{(\Delta x)^2} + \frac{f(x, y+\Delta y) - 2f(x,y) + f(x, y-\Delta y)}{(\Delta y)^2}$$

$$\nabla^2 I \approx \frac{f(x+\Delta x, y) + f(x, y+\Delta y) - 4f(x,y) + f(x-\Delta x, y) + f(x, y-\Delta y)}{(\Delta x)^2} = K_L \otimes I$$

$$\Delta x = \Delta y = 1\,\text{pixel}$$

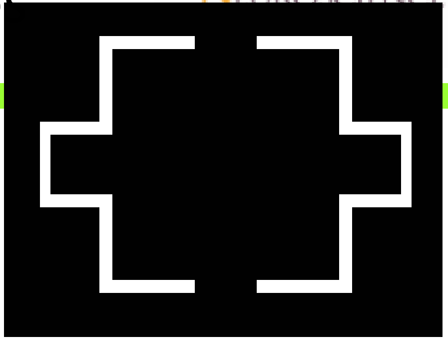
$$K_L = \begin{Bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{Bmatrix}$$





Segmentación – métodos básicos

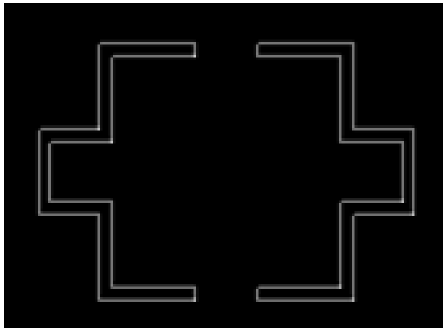
- Kernels…



$$I = I(x, y)$$

Un mapa de aristas o bordes (*edgemap*) considera transiciones de intensidad como bordes

$$f = \sqrt{(Kx \otimes I)^2 + (Ky \otimes I)^2}$$



Filtro de Sobel

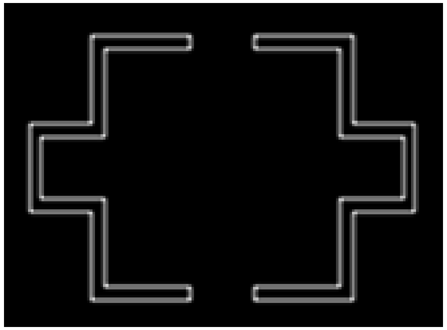
(notar grosor en el maps de bordes) *f*.

¿Cómo son…

$$Sx \otimes I\ ?$$

$$Sy \otimes I\ ?$$

$$\underset{Sx}{\begin{Bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{Bmatrix}} \qquad \underset{Sy}{\begin{Bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{Bmatrix}}$$



$$f_{Sobel} = \sqrt{(Sx \otimes I)^2 + (Sy \otimes I)^2}$$

Segmentación – métodos básicos

- Filtros basados en morfología
  - Un ejemplo: selección por tamaño

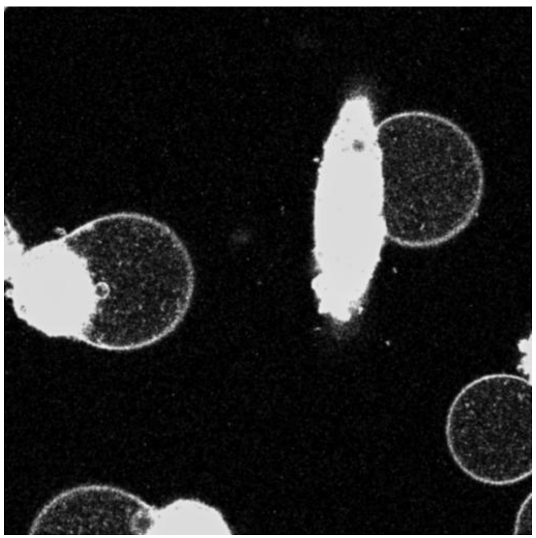
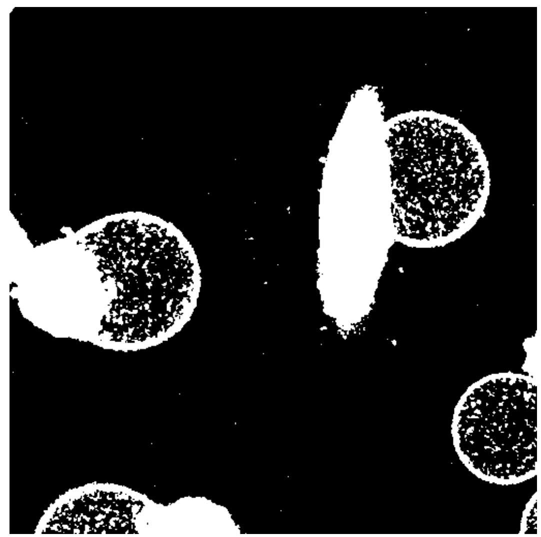
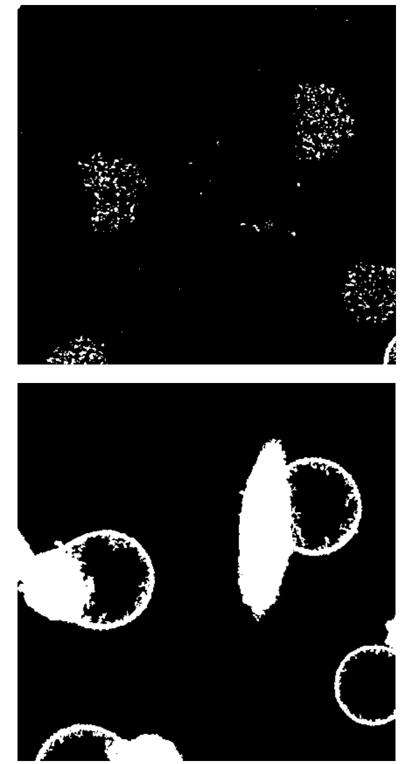


Imagen original
(escala de grises)



Umbral



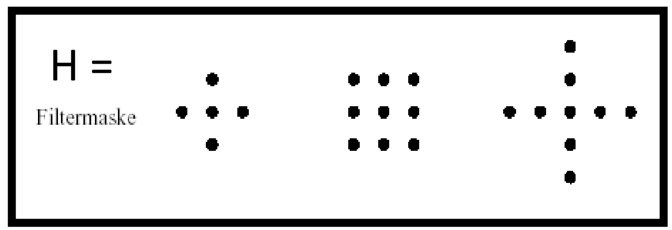
Selección por tamaño

¿Cómo se definiría un algoritmo para selección por tamaño?

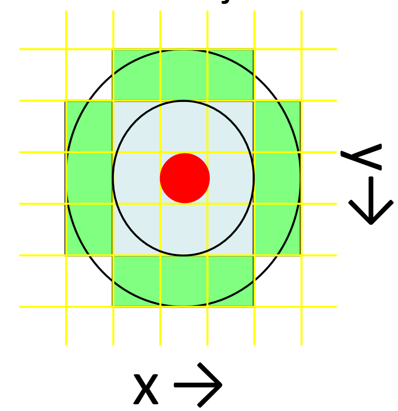
Segmentación – métodos básicos

Más filtros morfológicos :

- Elemento estructurante, plantilla o máscara H...
- introducción de reglas adicionales.



H (x_i, y_j, r):



Filtros de polinomios : $y(m,n) = \bar{h}_1[x(m,n)] + \bar{h}_2[x(m,n)],$

$$\bar{h}_1[x(m,n)] = \sum_{\substack{p=0 \\ (p,q)\neq(0,0)}}^{P-1} \sum_{q=0}^{Q-1} a(p,q) \cdot x(m-p, n-q)$$

$$\bar{h}_2[x(m,n)] = \sum_{\substack{p=0 \\ (p,q)\neq(0,0)}}^{P-1} \sum_{q=0}^{Q-1} \sum_{\substack{k=0 \\ (k,l)\neq(0,0)}}^{P-1} \sum_{l=0}^{Q-1} b(p,q,k,l) \cdot x(m-p, n-q) \cdot x(m-k, n-l)$$

- Filtros basados en morfología
  - **Morfología matemática**. Se usa este término para referirse una familia de operaciones basadas en máscaras y ciertas operaciones básicas

Operaciones de Minkowski

Adición ...dilatación: $A \oplus S = \{(m,n) \mid [S+(m,n)] \cap A \neq 0\}$

Substracción ...erosión: $A(-)S = \{(m,n) \mid [S+(m,n)] \subseteq A \neq 0\}$

...apertura (*opening*): $A \circ S = (A \ominus S) \oplus S,$

...cierre (*closing*): $A \bullet S = (A \oplus S) \ominus S,$

- A: imagen
- S: elemento estructurante

Segmentación – métodos básicos

Adición — dilatación : $A \oplus S = \{(m,n) | [S + (m,n)] \cap A \neq 0\}$.

Substracción — erosión : $A \ominus S = \{(m,n) | [S + (m,n)] \subseteq A \neq 0\}$

cierre : $A \circ S = (A \ominus S) \oplus S$,

apertura : $A \bullet S = (A \oplus S) \ominus S$,

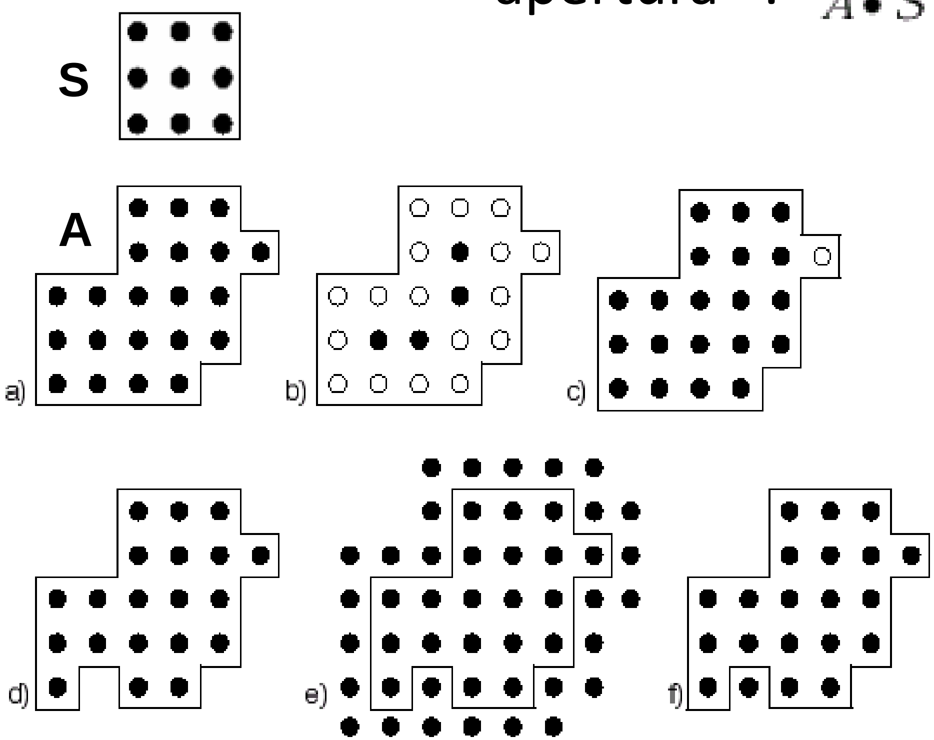
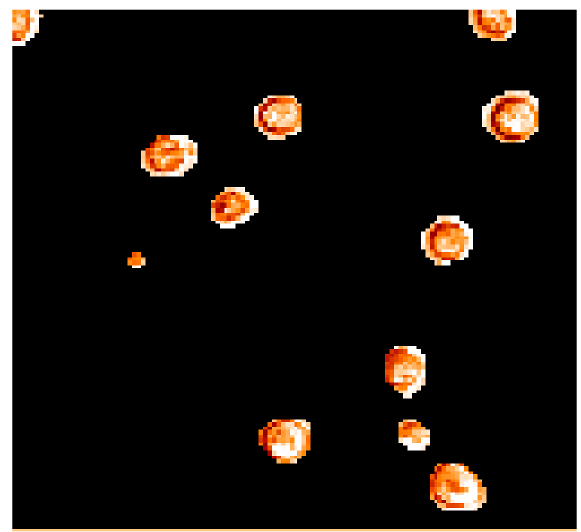


Abb. 2.5. **a** Originalform **b** erodiert **c** opening (Dilatation von b)
d Originalform **e** dilatiert **f** closing (Erosion von e)

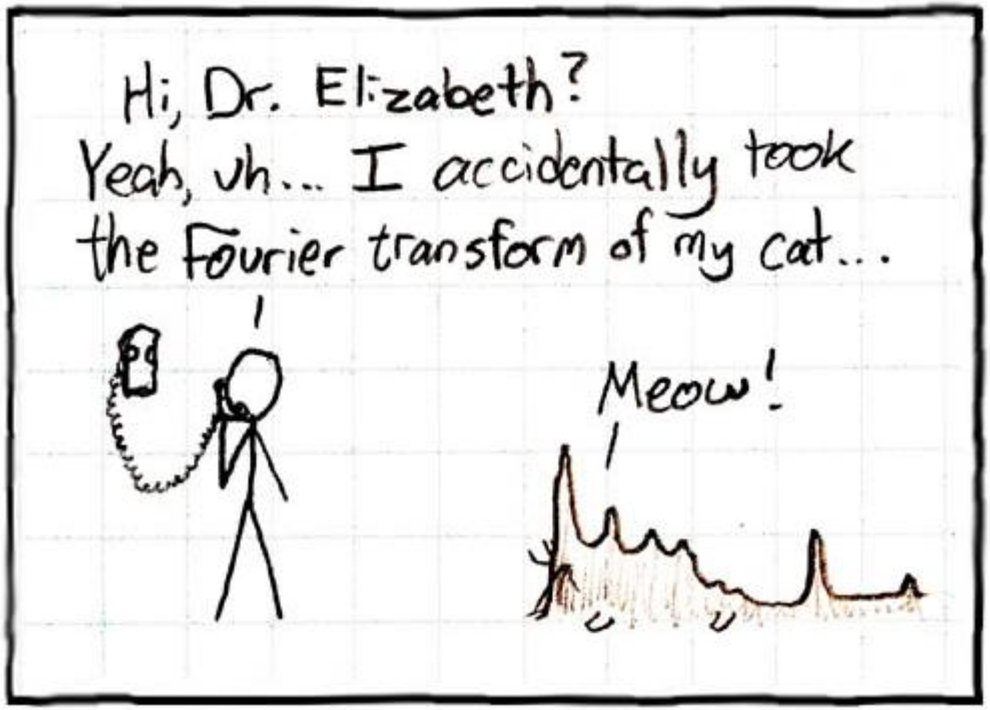


- Transformaciones de dominio… frecuencia (Fourier)

Jean-Baptiste Joseph Fourier

http://es.wikipedia.org/wiki/Jean-Baptiste_Joseph_Fourier

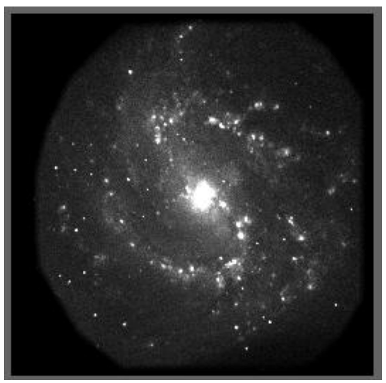
Jean-Baptiste-Joseph Fourier (21 de marzo 1768 en Auxerre - 16 de mayo 1830 en París), matemático y físico francés conocido por sus trabajos sobre la descomposición de funciones periódicas en series trigonométricas convergentes llamadas Series de Fourier, método con el cual consiguió resolver la ecuación del calor. La transformada de Fourier recibe su nombre en su honor. Fue el primero en dar una explicación científica al efecto invernadero en un tratado. Se le dedicó un asteroide que lleva su nombre y que fue descubierto en 1992.



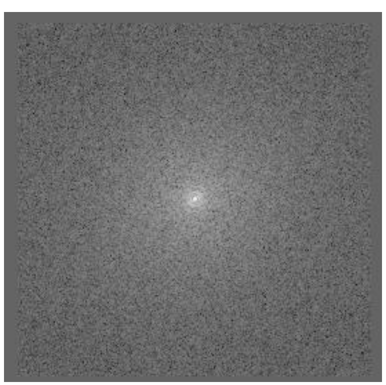
http://xkcd.org/26/

Jean-Baptiste Joseph Fourier

Jean-Baptiste-Joseph Fourier (21 de marzo 1768 en Auxerre - 16 de mayo 1830 en París), matemático y físico francés conocido por sus trabajos sobre la descomposición de funciones periódicas en series trigonométricas convergentes llamadas Series de Fourier, método con el cual consiguió resolver la ecuación del calor. La transformada de Fourier recibe su nombre en su honor. Fue el primero en dar una explicación científica al efecto invernadero en un tratado. Se le dedicó un asteroide que lleva su nombre y que fue descubierto en 1992.



$$F(k,l) = \frac{1}{N^2} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} f(a,b)\, e^{-\imath 2\pi \left(\frac{ka}{N} + \frac{lb}{N}\right)}$$



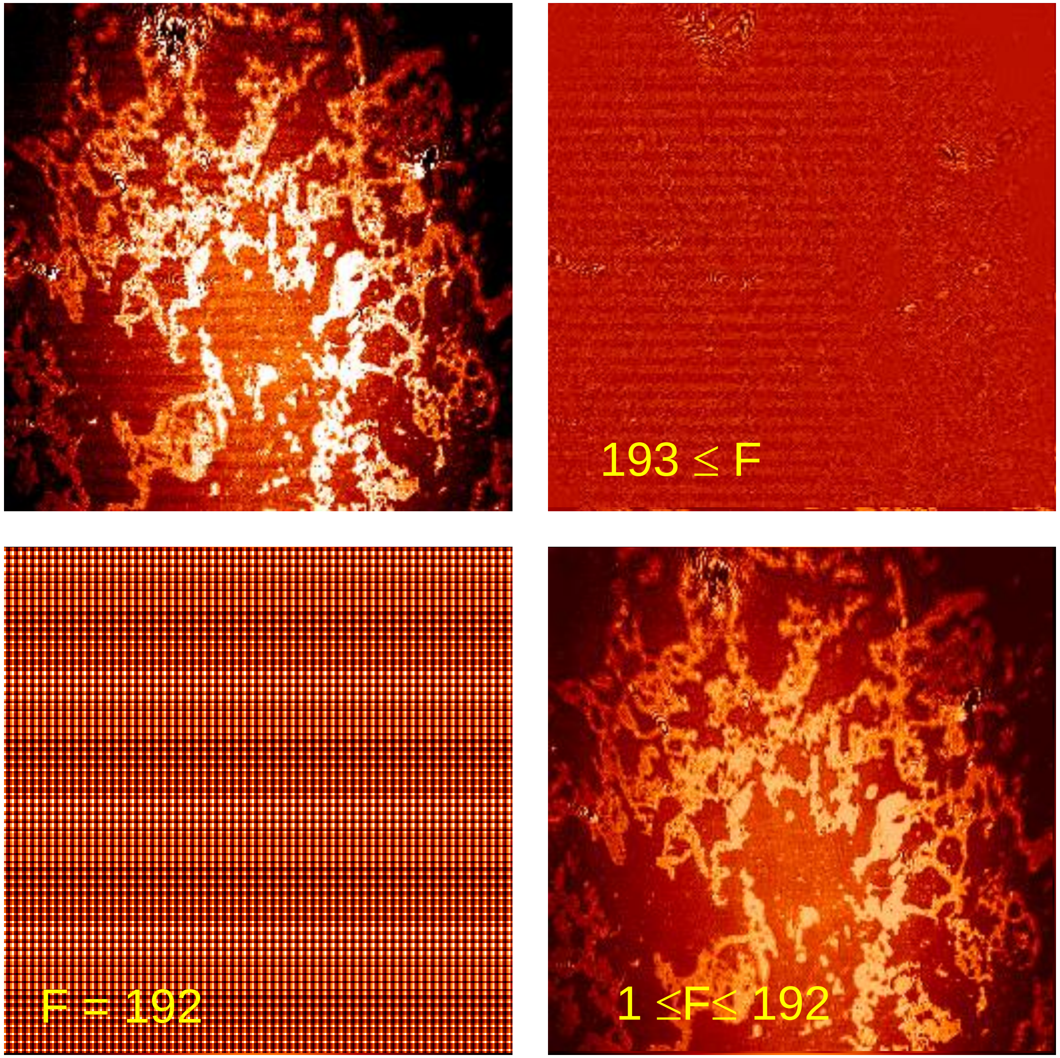
$$f(a,b) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} F(k,l)\, e^{\imath 2\pi \left(\frac{ka}{N} + \frac{lb}{N}\right)}$$

Segmentación – métodos básicos

Filtros locales:
- Lineales
- No lineales

Filtros globales:
- Análisis de Fourier
- Análisis de Wavelet
- ...

Filtros especiales:
- Análisis adaptativo
- ...

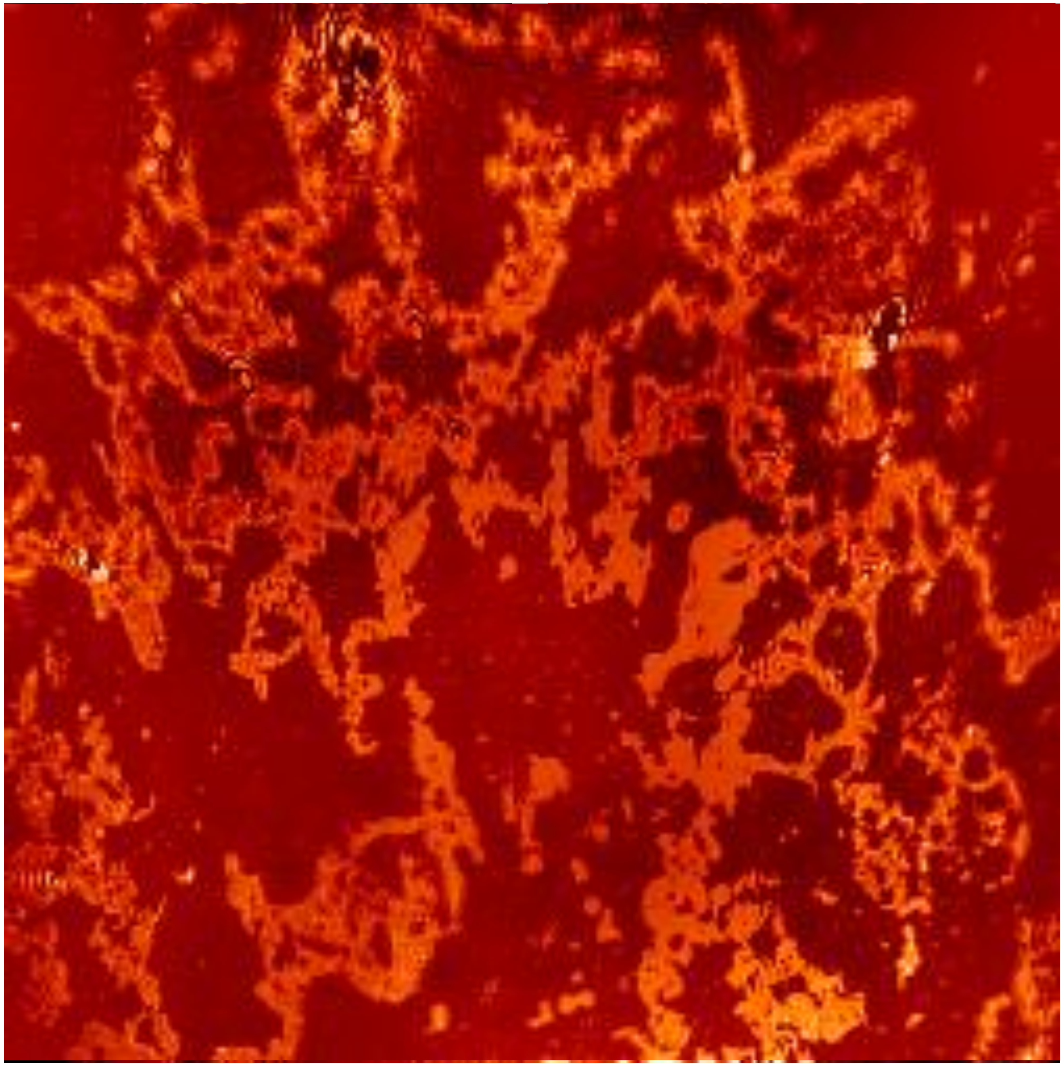


Segmentación – métodos básicos

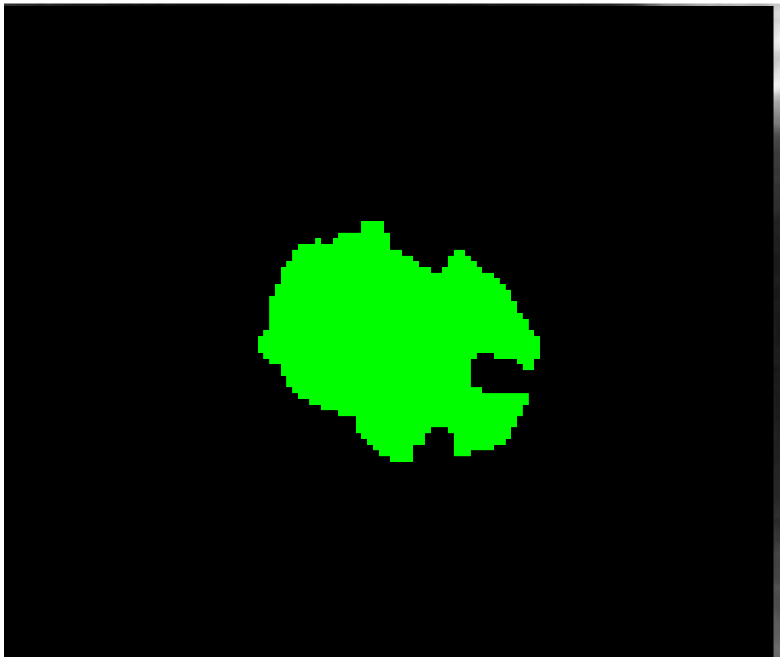
Filtros locales:
- Lineales
- No lineales

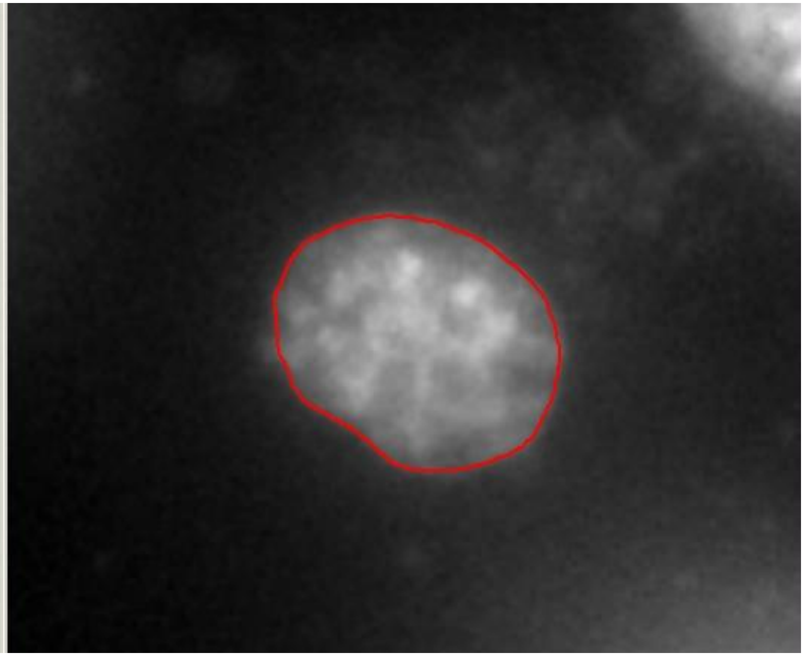
Filtros globales:
- Análisis de Fourier
- Análisis de Wavelet
- ...

Filtros especiales:
- Análisis adaptativo
- ...



- Aun así…





- Modelos de optimización: minimizan/maximizan una función de penalización/similaridad
  - Interest features detection (e.g. edges, points)
    - SIFT
  - Template matching
    - Hough transform
  - Pixel clustering
    - Statistics, probabilities
    - Graph cuts
    - Machine learning (support vector machine, neural networks…)
  - Differential equations, calculus of variations
    - Contour properties
    - Region properties

This can be a VERY long list…

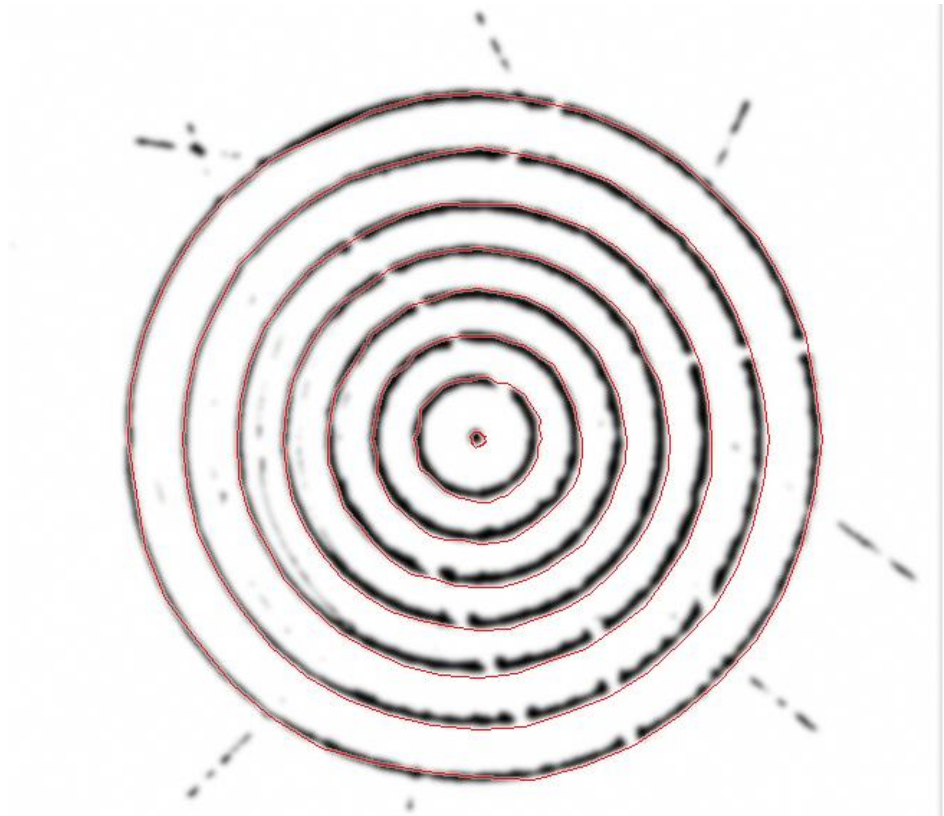
- Template matching
  - “Classic model” Hough transform
  - Applyes to circles, line segments and a variety of shapes

Hough P (1959)

If we can detect edges we can approximate a circle (center, radius)

n connected edges and ***m*** (=***n*** or <> ***n***) circles

A test is performed to determine the circles with “best fit”



Aguilar P, Hitschfeld N (DCC, 2010)

- Variational methods
  - Based on energy minimization, defining integral models
  - Idea: to include desirable features on segmented images (like homogeneus regions, short or smooth ROI boundaries)
  - Optimum solutions found by partial differential equations
  - Examples: Mumford-Shah, Ambrosio-Tortorelli, Chan-Vese

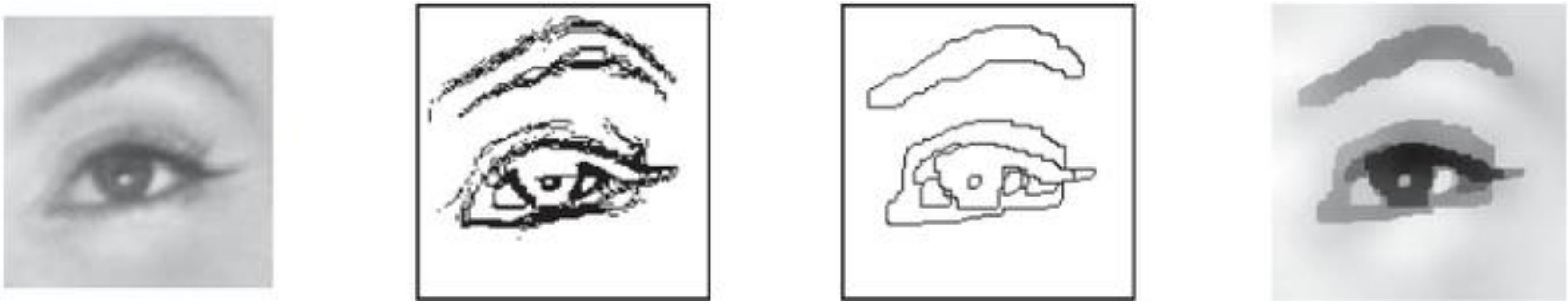


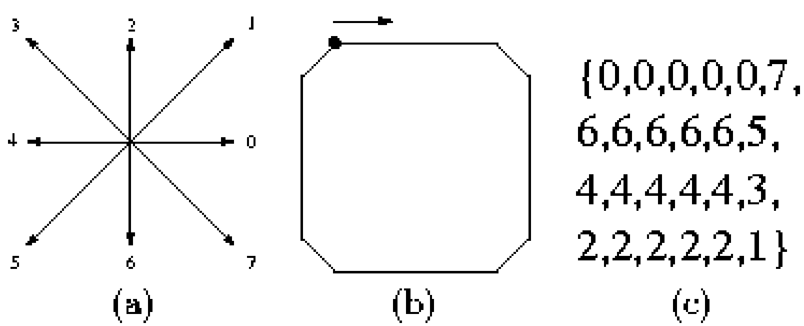
image I | main discontinuties in I | ROI boundaries B | piecewise smooth image J

$$E[J,B] = C \int d\vec{x}(I(\vec{x}) - J(\vec{x}))^2 + A \int_{D/B} \vec{\nabla} J(\vec{x}) \cdot \vec{\nabla} J(\vec{x}) d\vec{x} + B \int_B ds$$

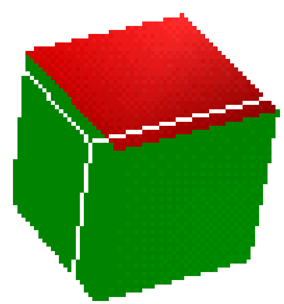
The Mumford & Shah functional (1989)

Boundary model construction

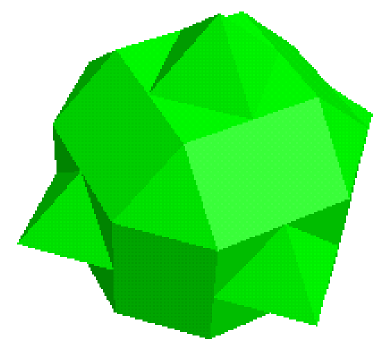
- 2D Freeman chain code
- 3D mesh models (voxel based): *marching cubes* (surface meshes), *tethraedra* (volume meshes)



2D polygon chain code

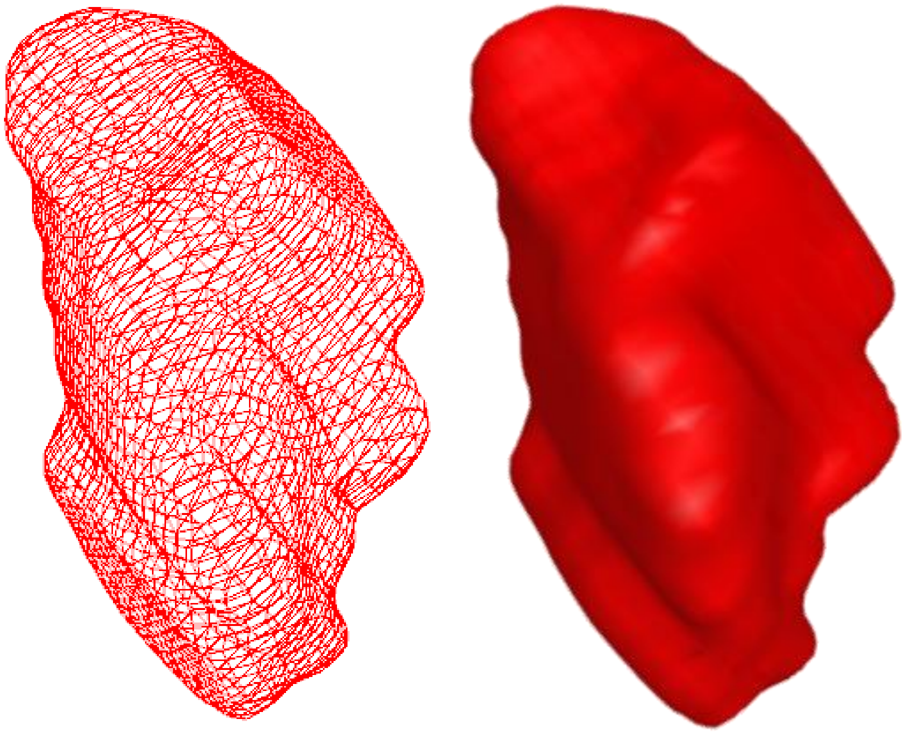


Voxel ("3D pixel") model

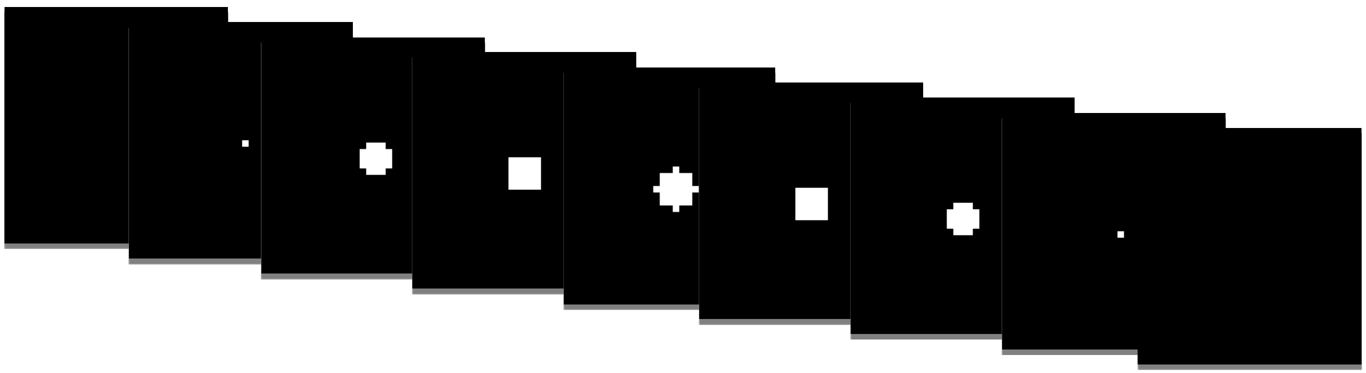


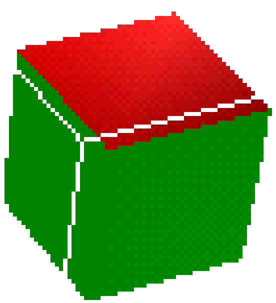
3D polygon mesh

- A typical 3D surface mesh model is formed by:
  - Nodes or vertices
  - Polygons
- Other models
  - Polynomial surfaces (splines, Bézier, NURBS, …)
  - Primitives composition

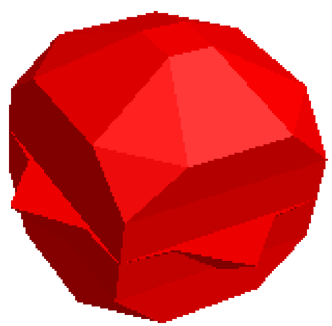


- Surface mesh model construction
  - Many approaches for construction
  - Many approaches for modeling

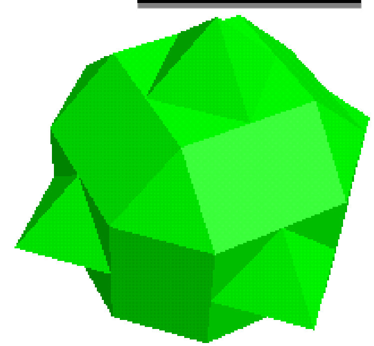




Voxel ("3D pixel") model



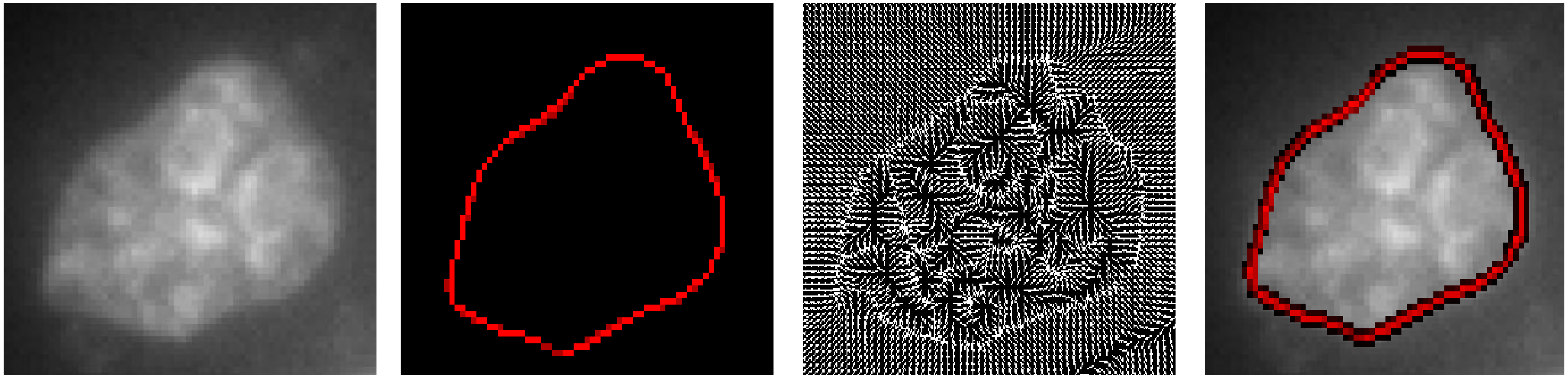
Triangle mesh



Triangular and quarilateral mesh

Segmentación – métodos avanzados

- Active contour models
  - Optimization of different properties



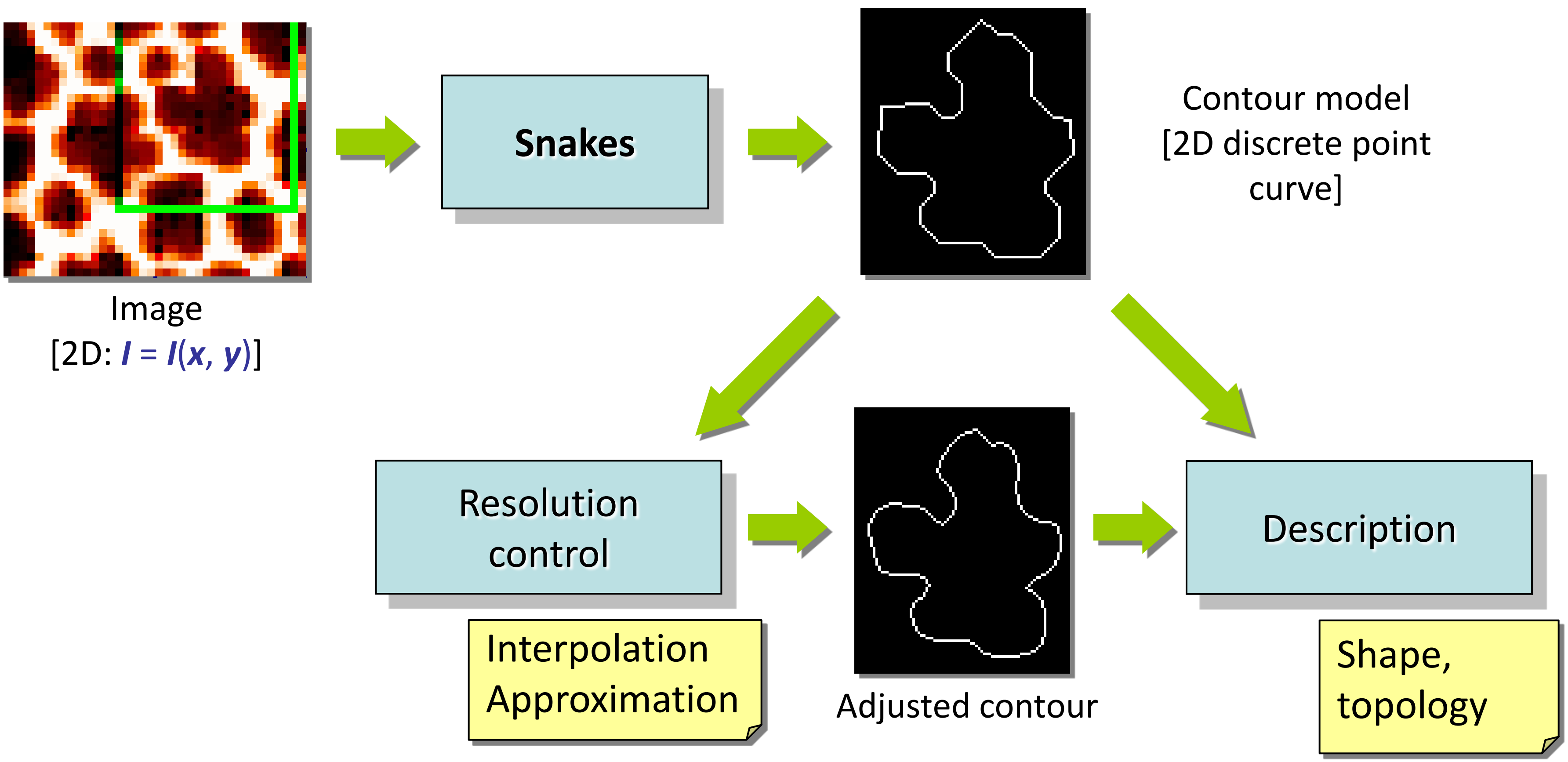
input
image
+initial guess

contour C(s)
- elasticity
  (contraction)
- rigidity
  (bending, cornering)

force field
- repulsion
- attraction

output: force balance
minimal energy

- 2D active contours or *snakes*



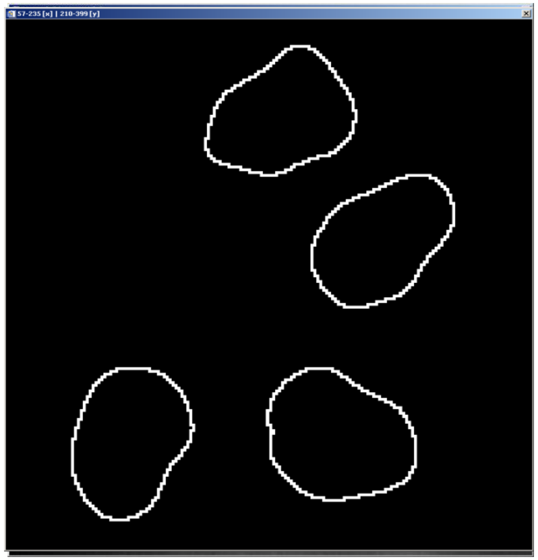
1 contour function → 1 ROI

– 2D parametric curve

$C = C(s) = [x(s), y(s)]$
$s \in [0, 1]$ (arbitrary length)

– 2D discretization

$C = \{[x_i , y_i]; i = 0..n\}$



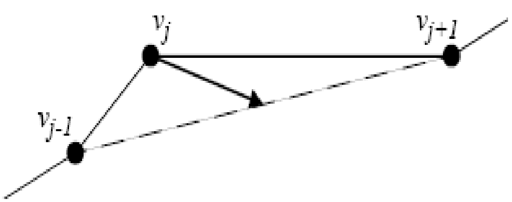
- Snakes: optimization derived from a variational approach
  - Minimization of an integral functional
    “a snake minimizes its energy”

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

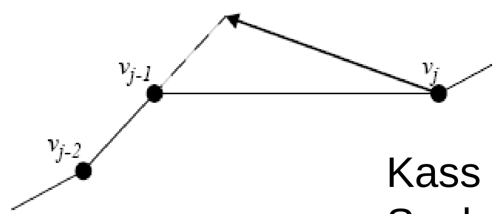
Internal energy, contour dependant

External energy, image dependant

Elasticity term
(coefficient α)

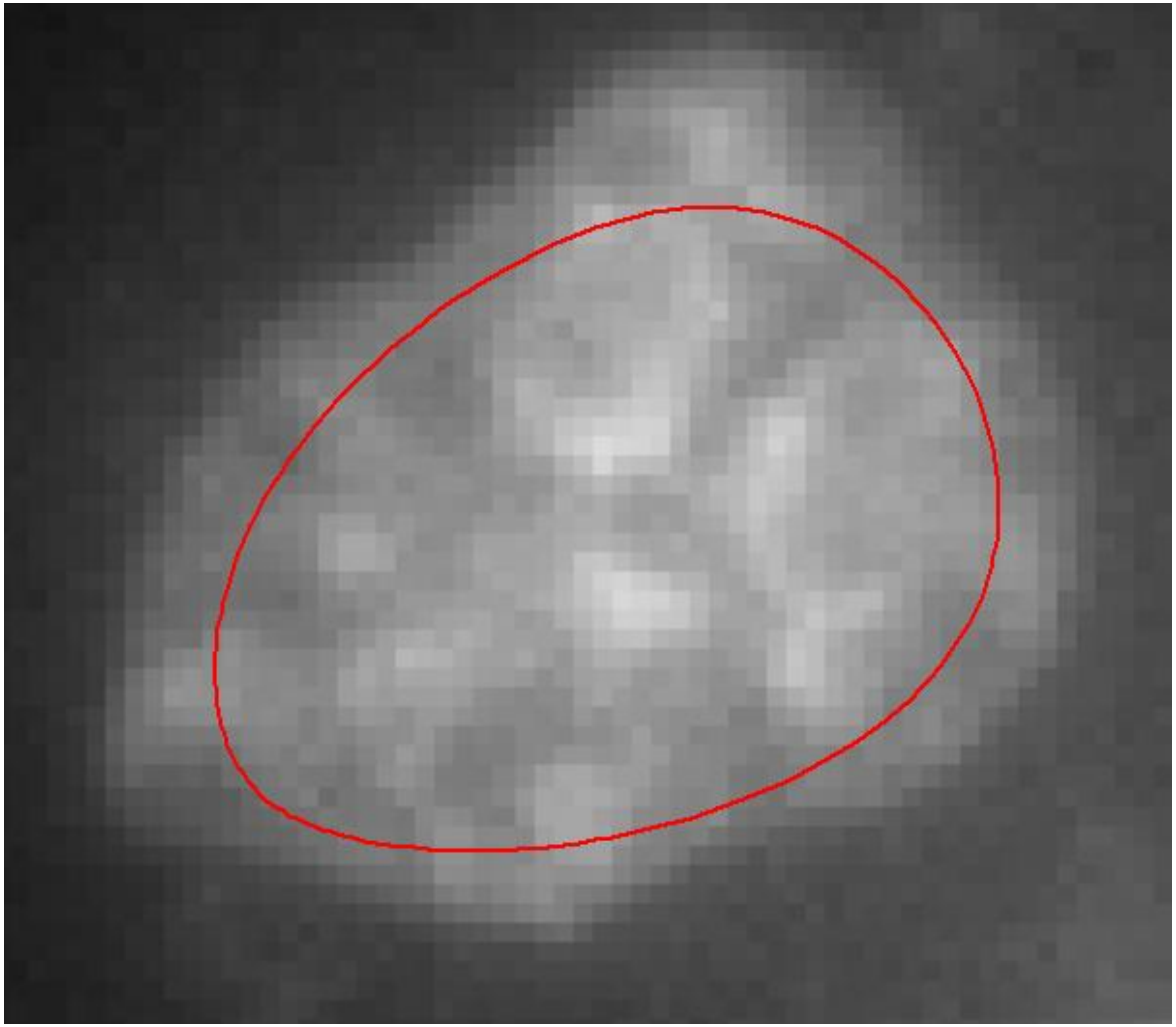


Rigidity term
(coefficient β)

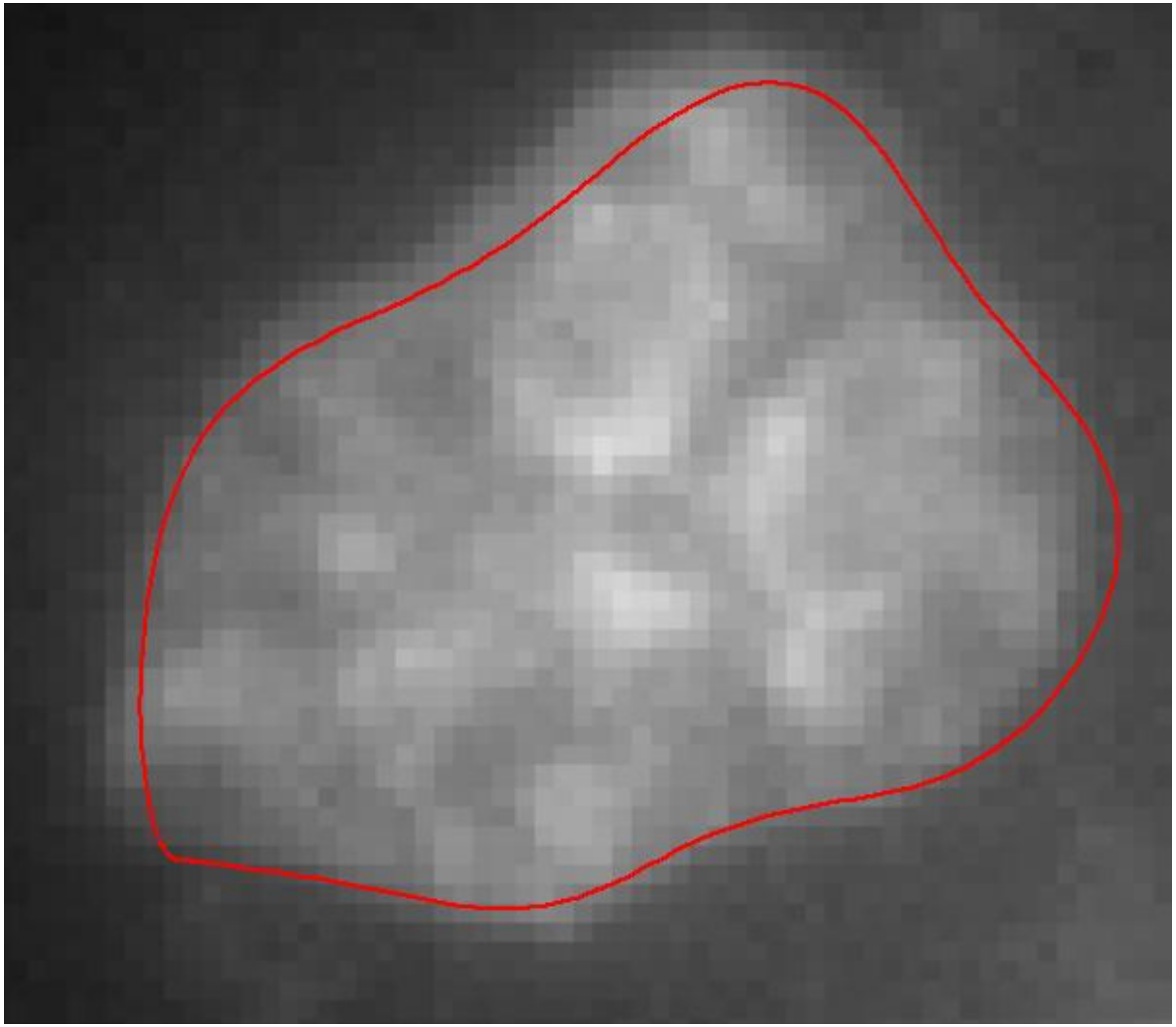


Kass et al (1988)
Snakes: active contour models
Int. J. of Computer Vision 1(4): 321-331

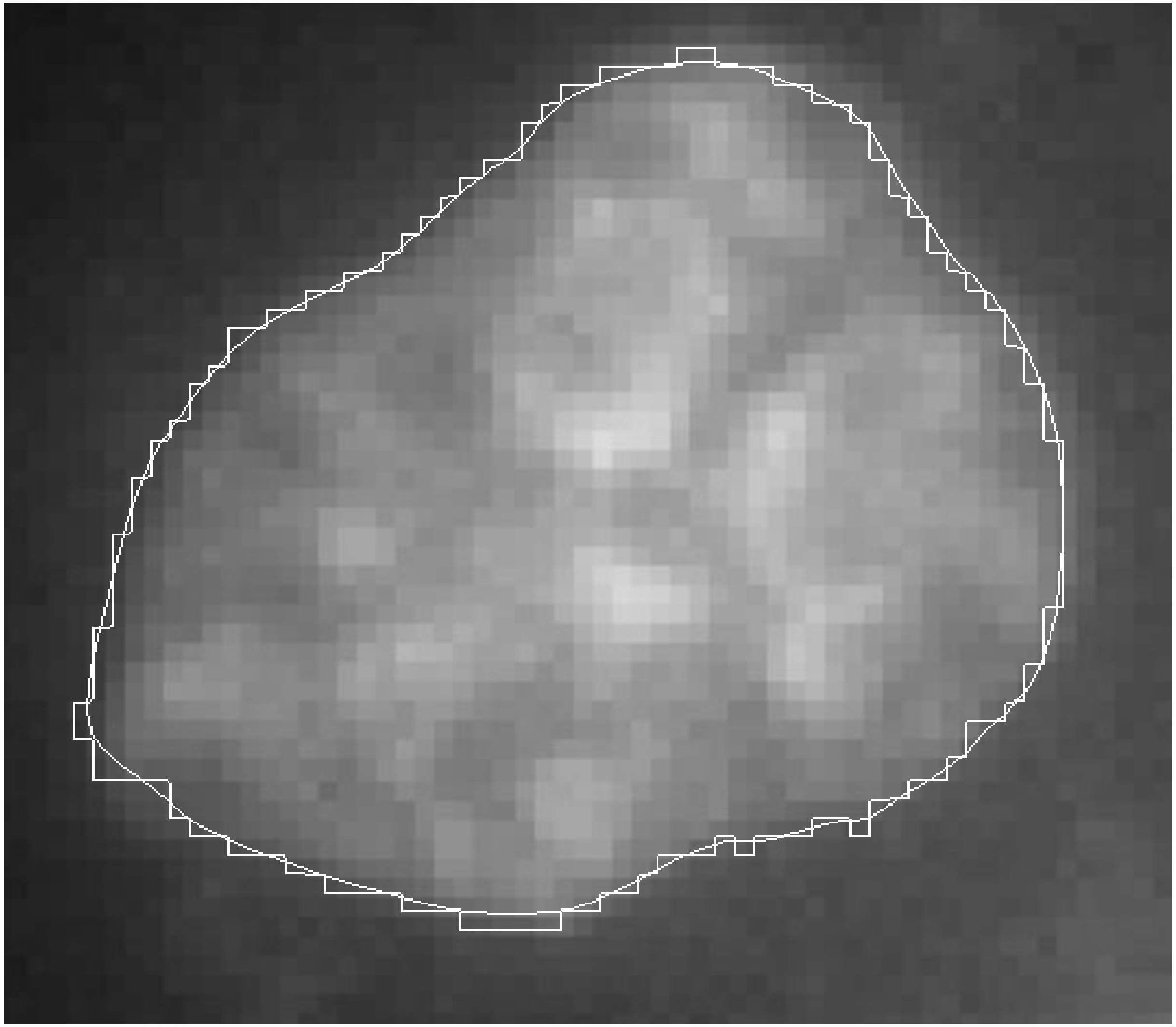
Elasticity - **α**



Rigidity - β



Resolution (f)



Snakes – formulation

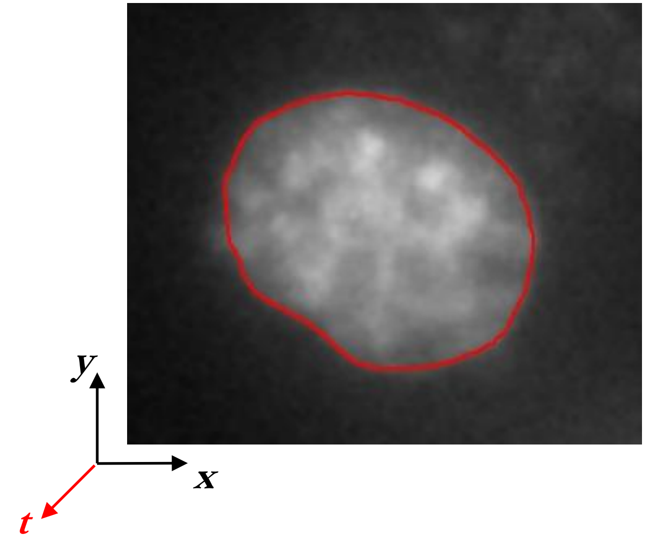
– Energy minimization condition: Euler-Lagrange equation

$$\alpha \frac{\partial^2 C(s)}{\partial s^2} - \beta \frac{\partial^4 C(s)}{\partial s^4} - \nabla E_{ext}[C(s)] = 0$$

Internal **forces** External **force**

Snakes – numerical solution

- Dynamic formulation
  - *C*(*s*) … *C*(*s*, *t*)
  - Time equilibrium leads to solution



$$\frac{\partial C(s,t)}{\partial t} = 0$$

$$\frac{\partial C(s,t)}{\partial t} = \alpha \frac{\partial^2 C(s,t)}{\partial s^2} - \beta \frac{\partial^4 C(s,t)}{\partial s^4} - \nabla E_{ext}[C(s,t)]$$

Time evolution

Internal forces

External force

Snakes – numerical solution

- Parameters (values can be adjusted)
  - α (elasticity), β (rigidity)

$$\alpha \frac{\partial^2 C}{\partial s^2} - \beta \frac{\partial^4 C}{\partial s^4} - \nabla E_{ext}[C] = 0$$

$$\frac{\partial C(s,t)}{\partial t} = 0$$

$$A_{n*n} C - \nabla E_{ext}[C] = 0$$

$$A C_t - \kappa \nabla E_{ext}(C_{t-1}) = -\gamma (C_t - C_{t-1})$$

  - κ (external force), γ (viscosity)

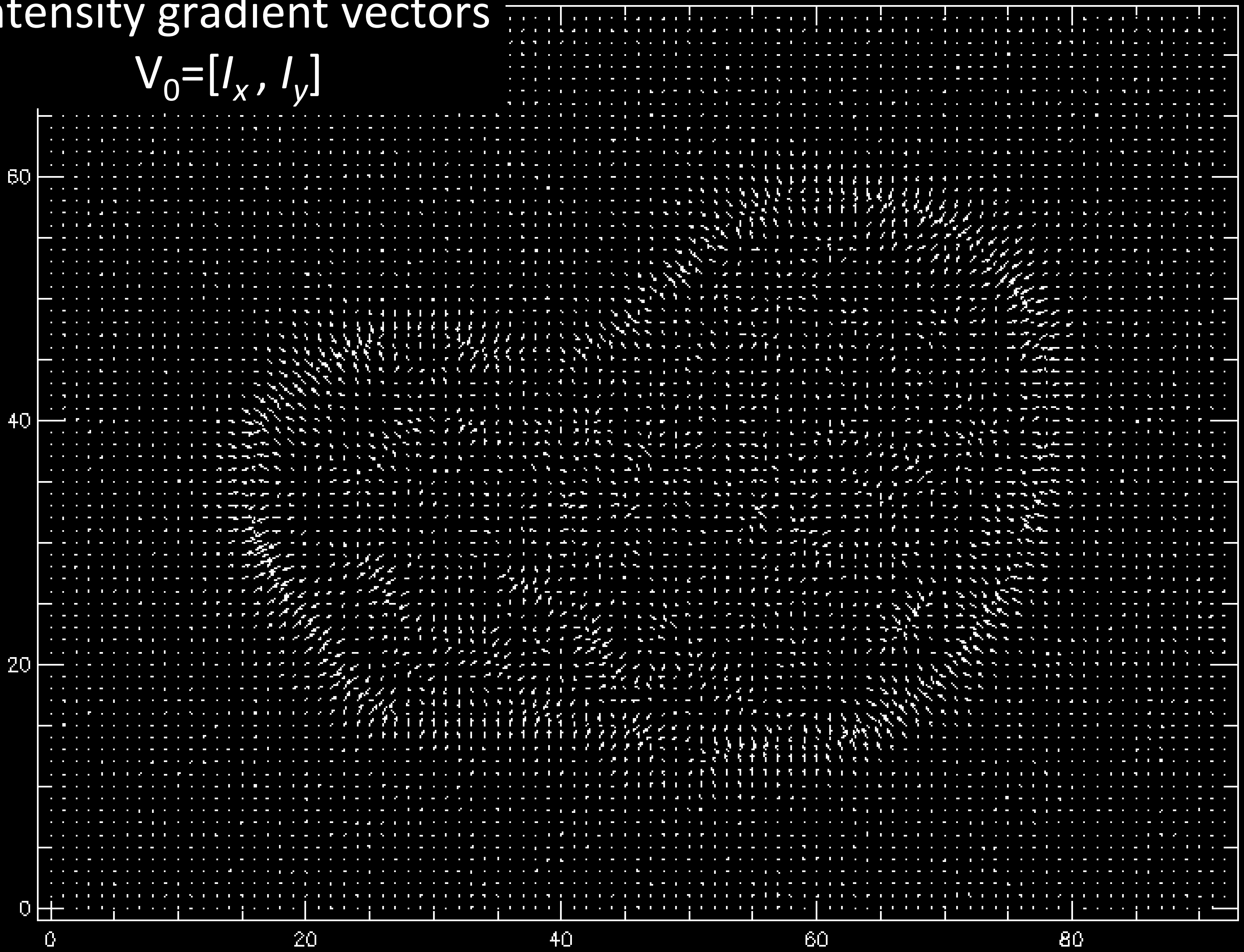
Iterative scheme, solved computationally

$$C_t = (A + \gamma I)^{-1} - (C_{t-1} + \kappa \nabla E_{ext}(C_{t-1}))$$

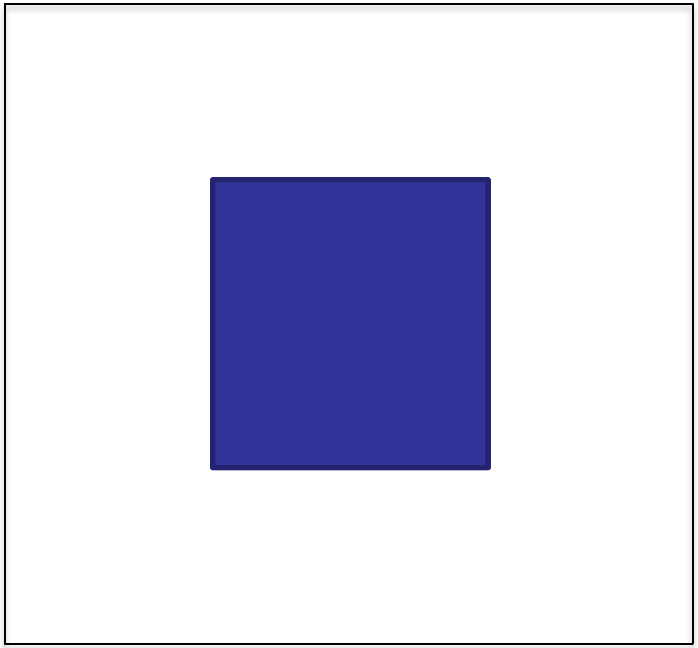
Matrix inversion, array data structures, adaptive algorithms, …

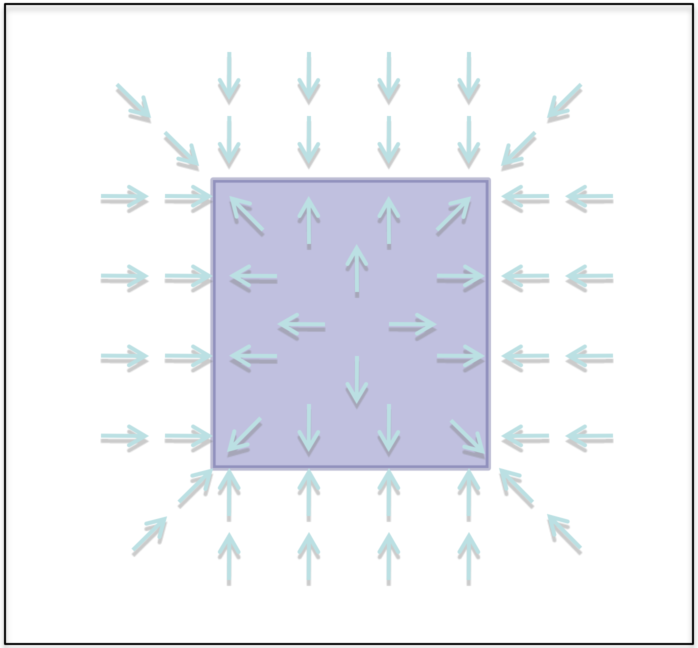
Intensity gradient vectors

$V_0=[I_x , I_y]$



Attraction fields can be constructed from the intensity gradients





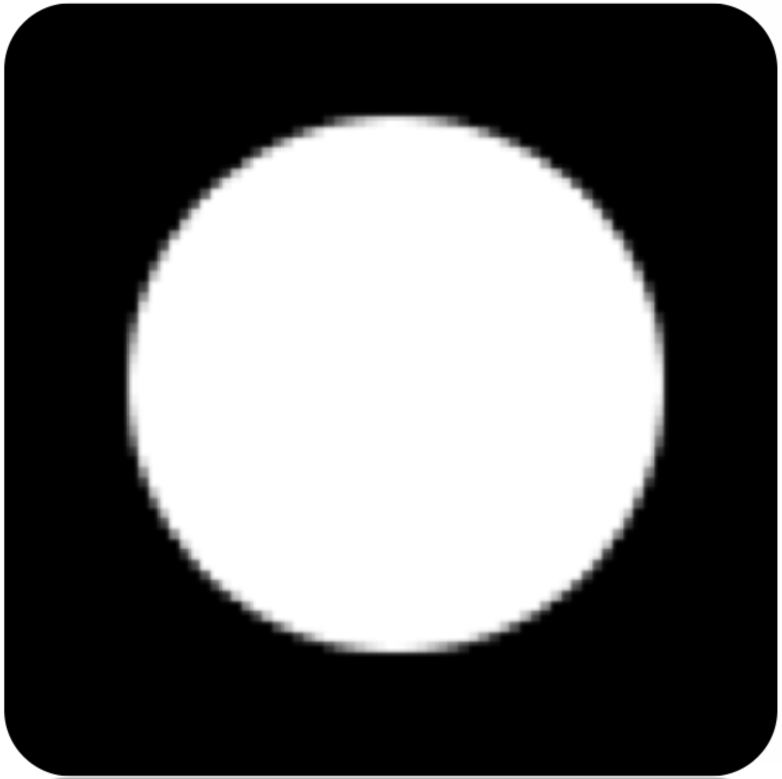
$$\underbrace{I(x,y)}_{\text{Intensidad de Imagen}}$$

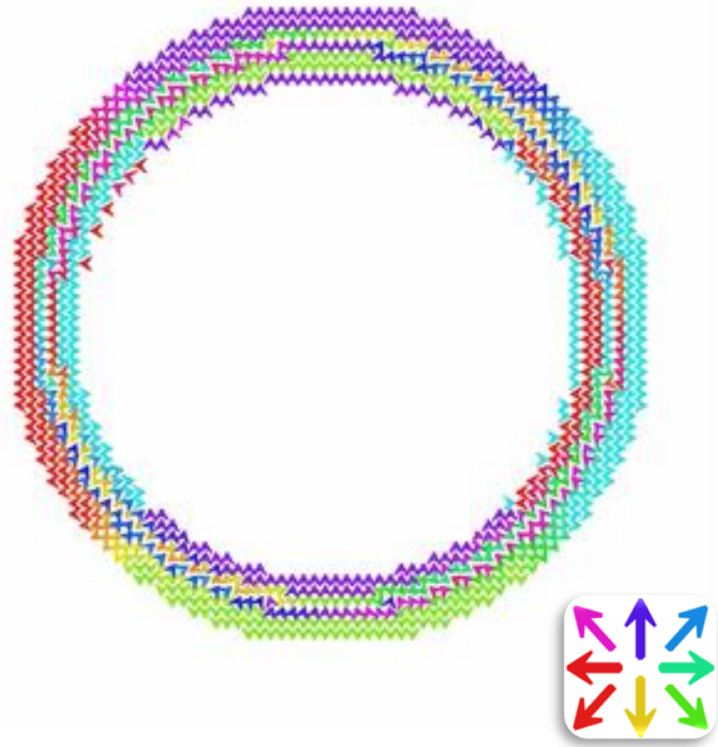
$$\nabla \left|\nabla I(x,y)\right|^2$$

$$\nabla \left|G_\sigma * \nabla I(x,y)\right|^2$$

Image force fields (external forces in active contour models)

Gradient vector flow (GVF)
Generalized GVF (GGVF)





Xu & Prince (1998) Active Contours and Gradient Vector Flow http://iacl.ece.jhu.edu/projects/gvf
Xu & Prince (1998) Generalized gradient vector flow external forces for active contours

GVF image force field

Defined by an energy functional to be minimized

$$\textit{Let}$$

$$V(x,y) = [u,v] \qquad f = |G_\sigma * \nabla I|$$

$$u = u(x,y)$$

$$v = v(x,y)$$

$$\min_{(u,v)\in C^2(\Omega;\mathbb{R}^2)} \int_\Omega \underbrace{g(|\nabla f|)\left(|\nabla u|^2 + |\nabla v|^2\right)}_{\text{Término de Difusión}} + \underbrace{h(|\nabla f|)\left(|u - f_x|^2 + |v - f_y|^2\right)}_{\text{Término de Borde}} \, dx$$

$$\begin{aligned} g\Delta u - h(u - f_x) &= 0 \\ g\Delta v - h(v - f_y) &= 0 \end{aligned} \qquad \Downarrow \qquad \begin{aligned} g\Delta u - h(u - f_x) &= \frac{\partial u}{\partial t} \\ g\Delta v - h(v - f_y) &= \frac{\partial v}{\partial t} \end{aligned}$$

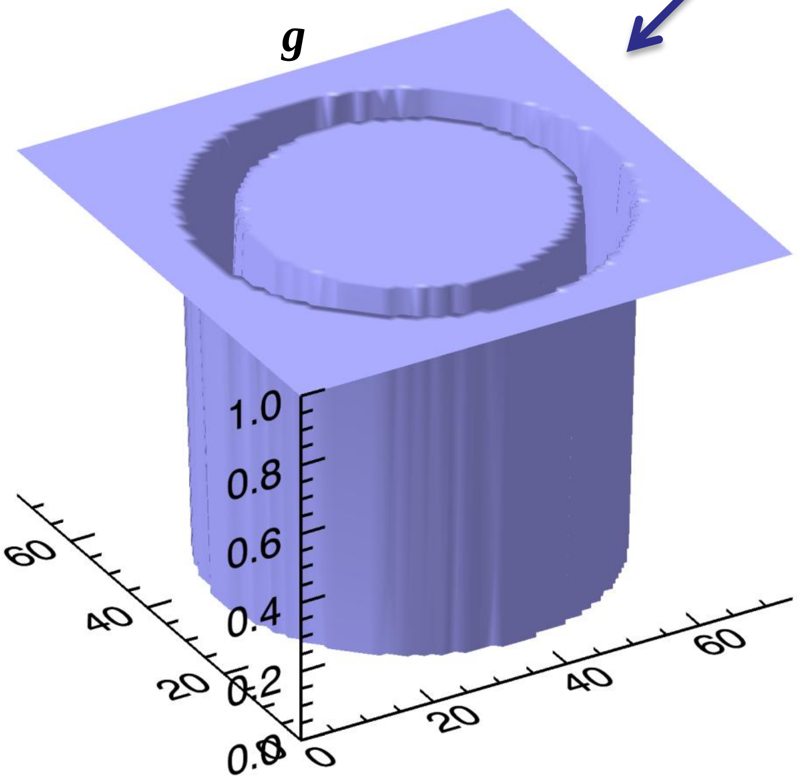
$$\begin{aligned} \nabla g \cdot \nabla u + g\Delta u - h(u - f_x) &= 0 \\ \nabla g \cdot \nabla v + g\Delta v - h(v - f_y) &= 0 \end{aligned}$$

The GVF (GGVF) energy minimization

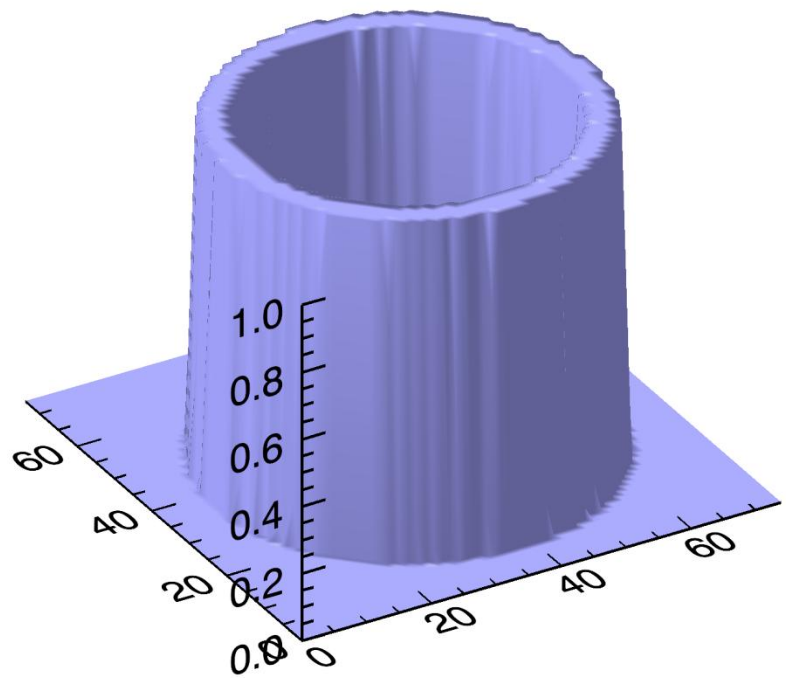
$$f = |G_\sigma * \nabla I|$$

$$\min_{(u,v)\in C^2(\Omega;\mathbb{R}^2)} \int_\Omega \underbrace{g(|\nabla f|)\left(|\nabla u|^2 + |\nabla v|^2\right)}_{\text{Término de Difusión}} + \underbrace{h(|\nabla f|)\left(|u - f_x|^2 + |v - f_y|^2\right)}_{\text{Término de Borde}}\, dx$$

g



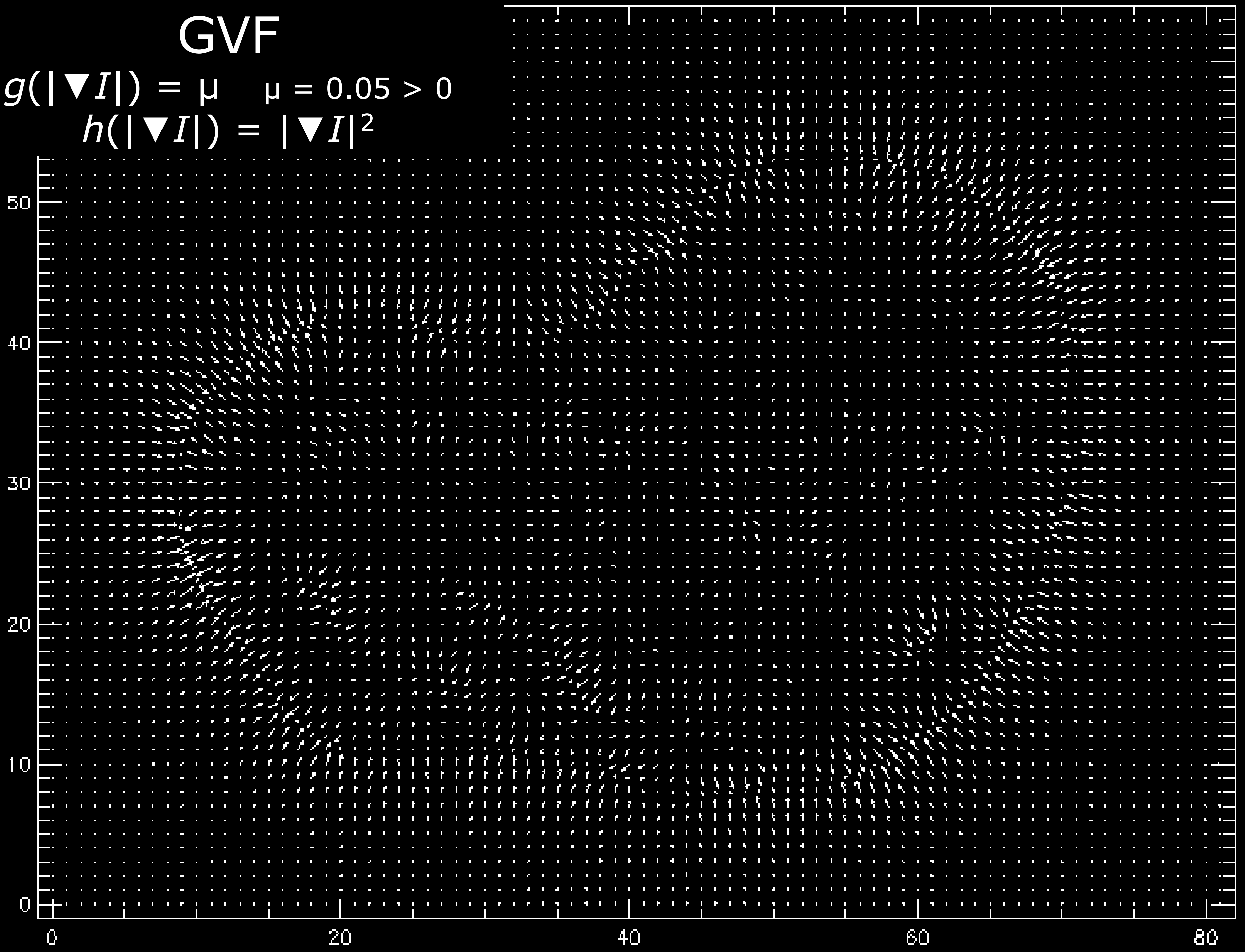
h



GVF

$g(|\nabla I|) = \mu \quad \mu = 0.05 > 0$

$h(|\nabla I|) = |\nabla I|^2$

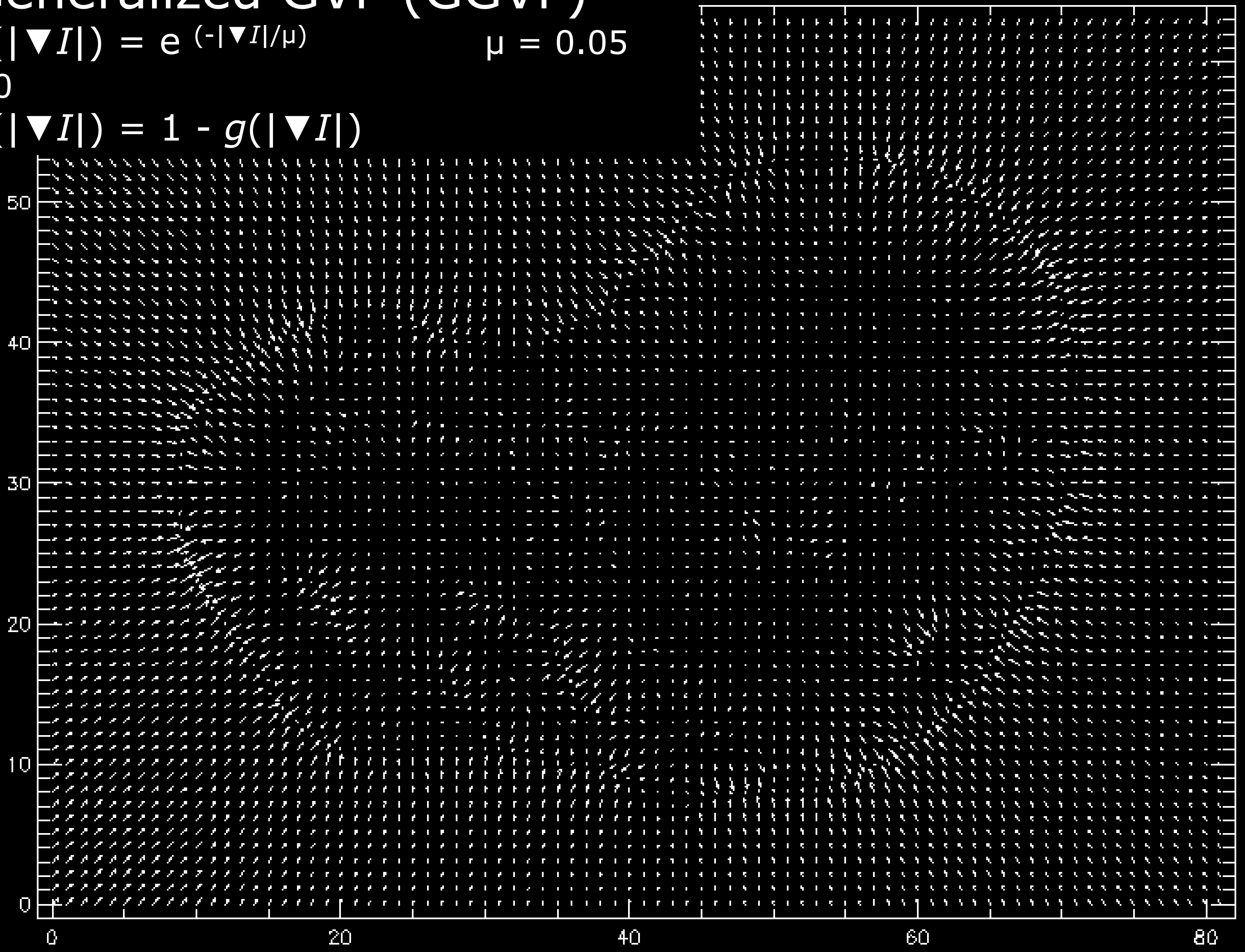


Generalized GVF (GGVF)

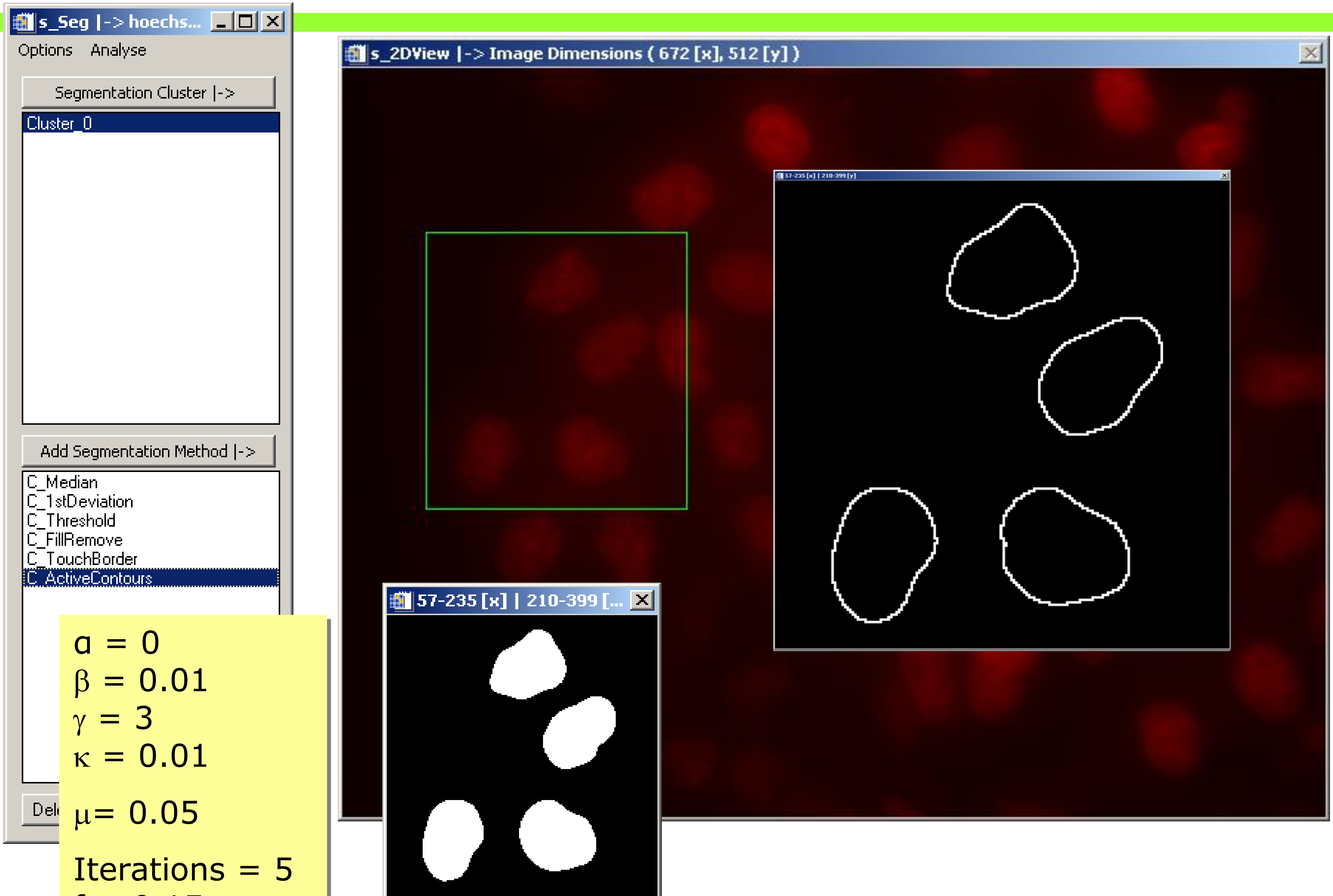
$g(|\nabla I|) = e^{(-|\nabla I|/\mu)}$ $\mu = 0.05$

>0

$h(|\nabla I|) = 1 - g(|\nabla I|)$



Segmentación – métodos avanzados

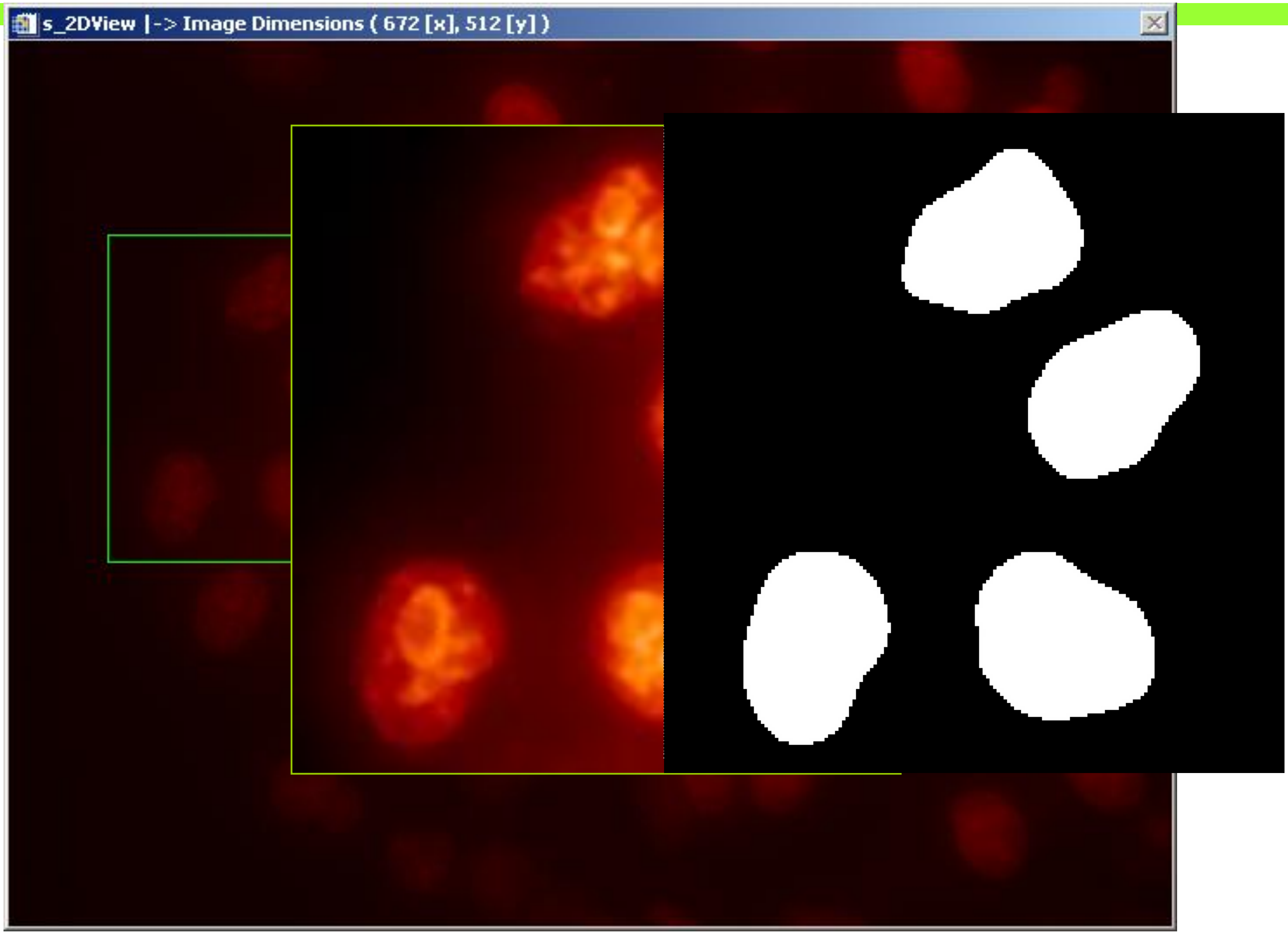


$\alpha = 0$
$\beta = 0.01$
$\gamma = 3$
$\kappa = 0.01$

$\mu = 0.05$

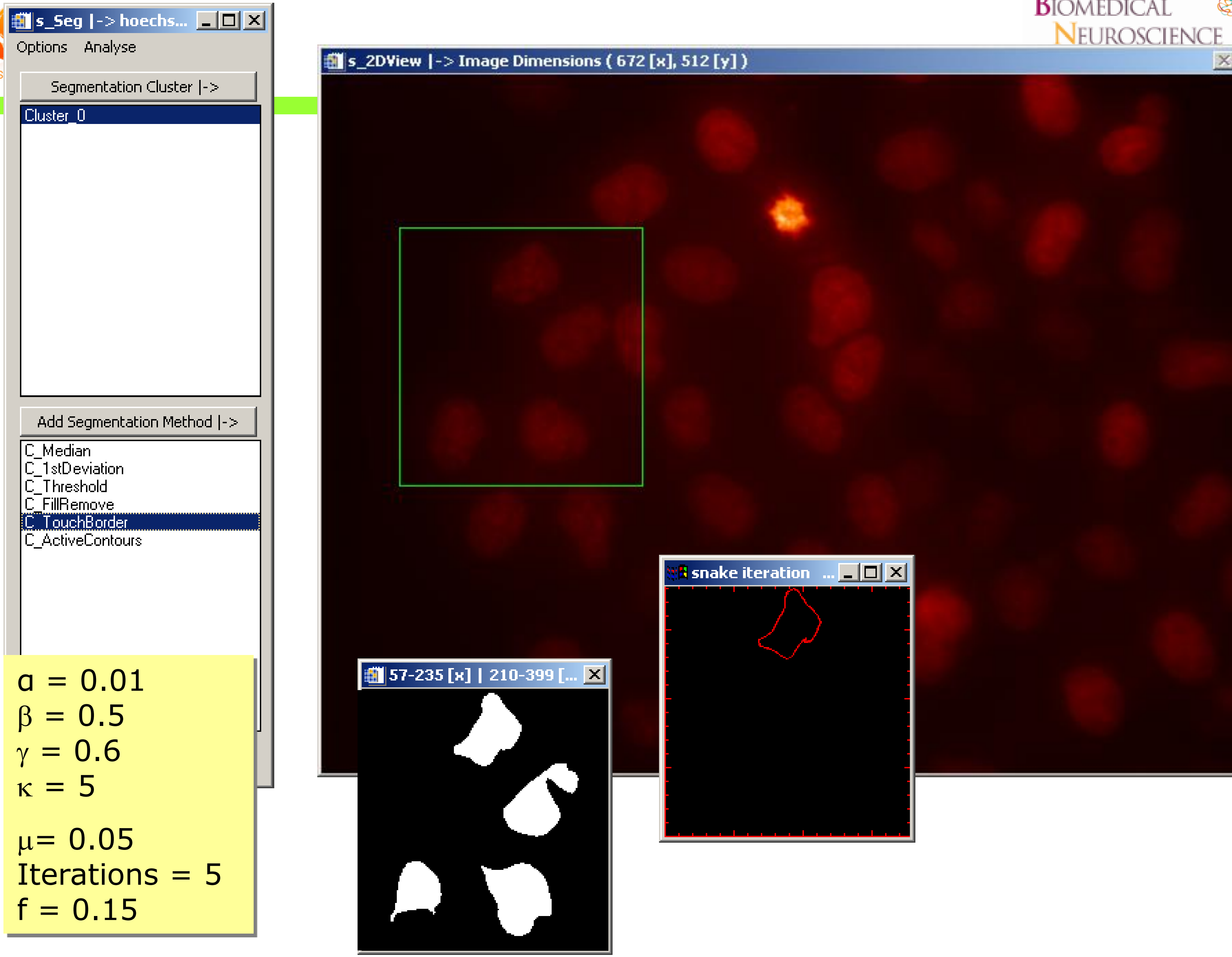
Iterations = 5
f = 0.15

Segmentación – métodos avanzados



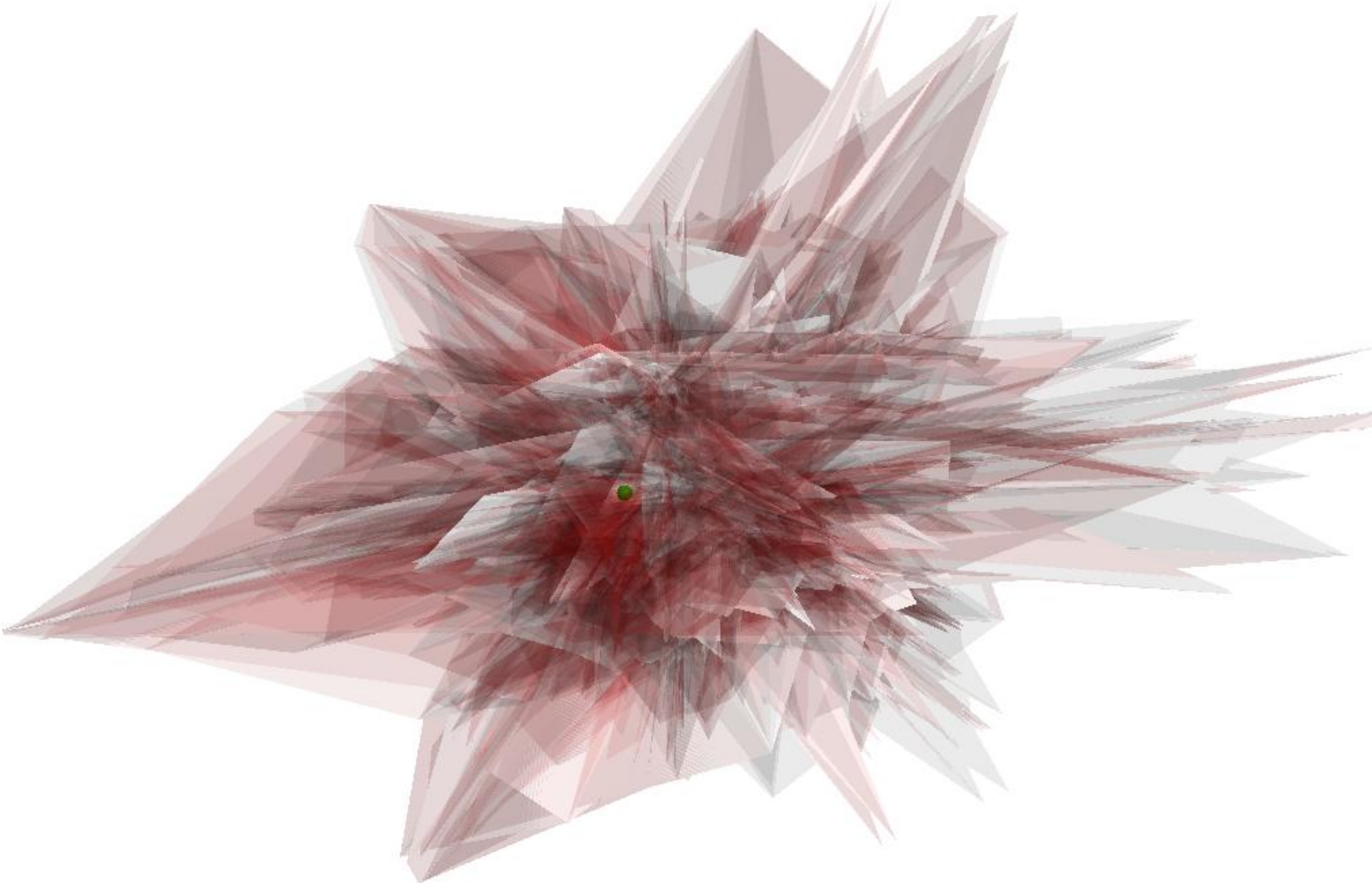
Segmentación – métodos avanzados

- Some problems with the snakes method
  - The quantity of contours must not change, given the 1-1 correspondence of the model
    - Dependecy on the initial guess
  - Local optima for the functional minimization
    - A bad initialization of the snakes can lead to unstability and undesired results
    - Control of the snakes parameters is required



$\alpha = 0.01$
$\beta = 0.5$
$\gamma = 0.6$
$\kappa = 5$

$\mu = 0.05$
Iterations = 5
f = 0.15

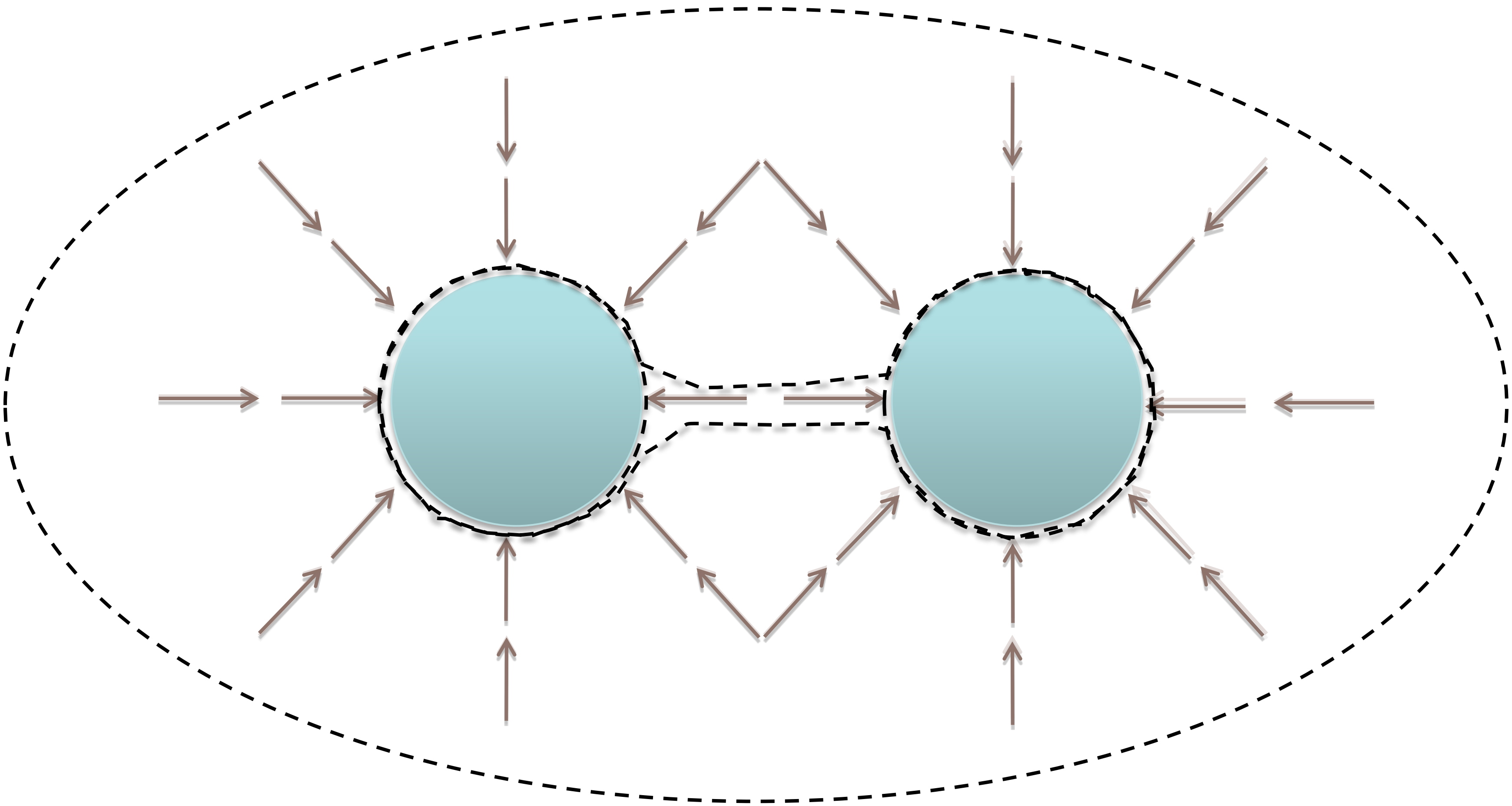


Segmentación – métodos avanzados

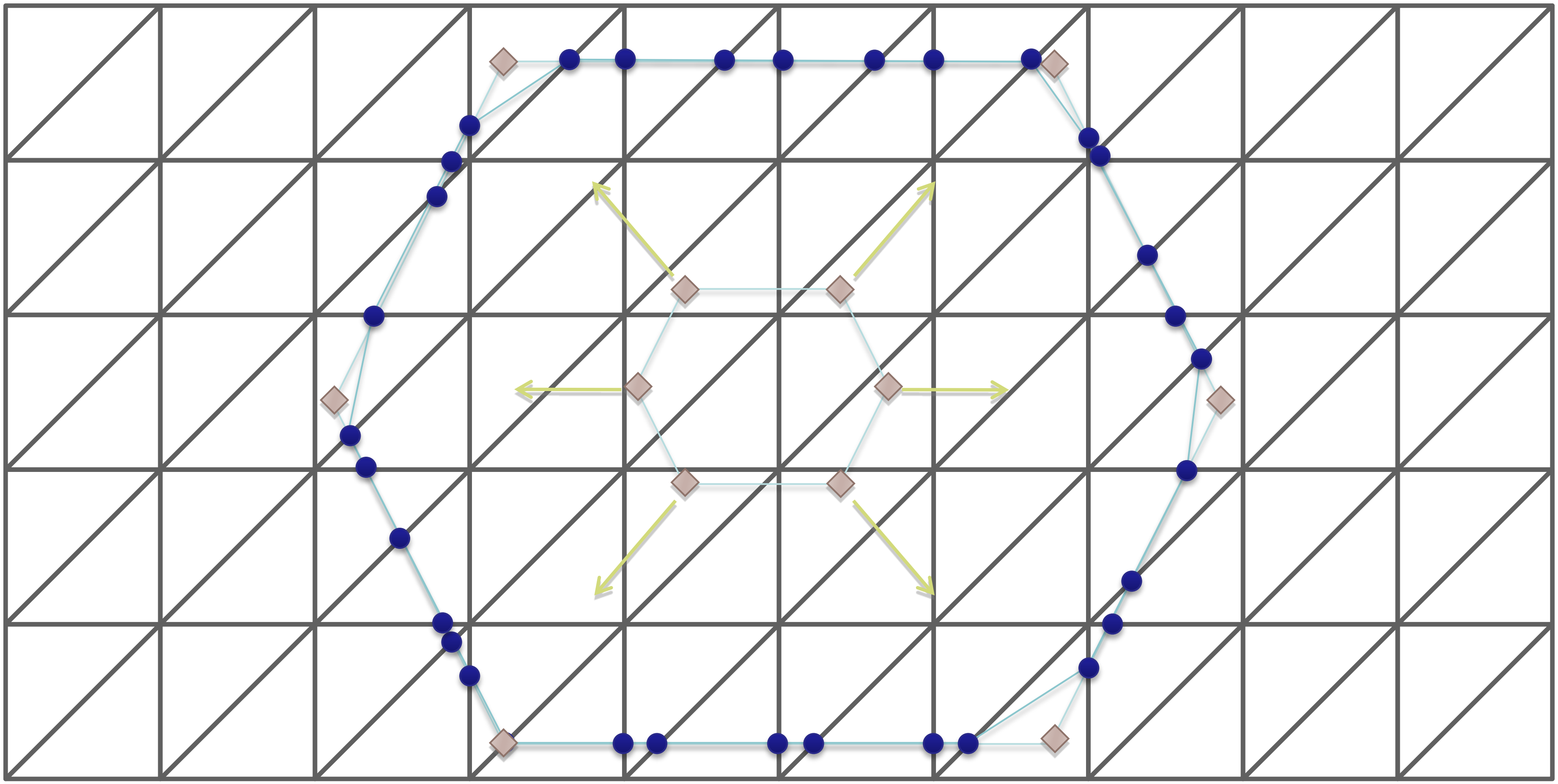
- Alternatives…
  - Arbitrary contour initialization… automation
  - Inference (machine learning approaches)

- Topology adaptive snakes (or surfaces) McInerney & Terzopoulos, 1995 (2D), 2002 (3D)

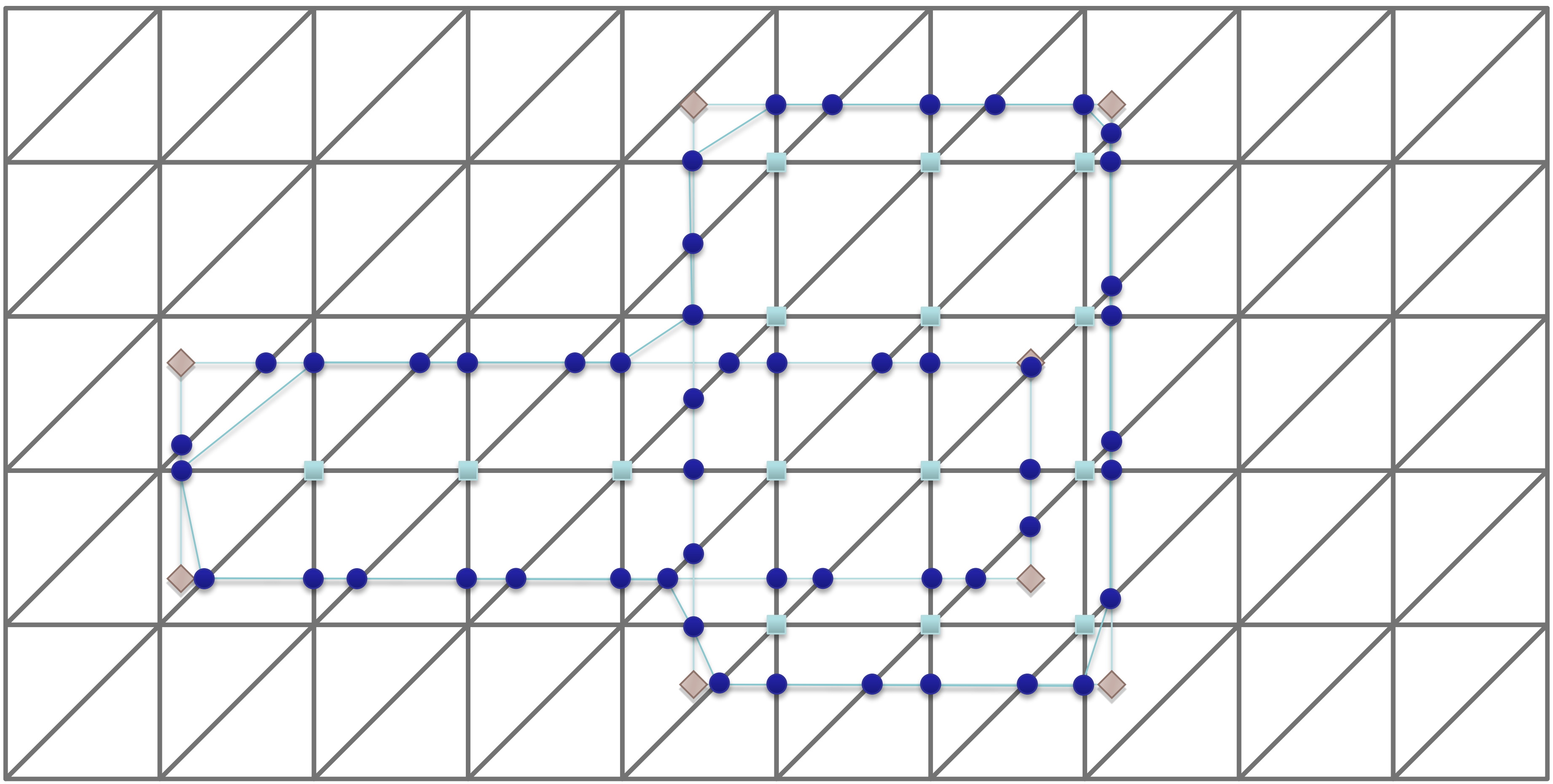
Topological changes handling: splitting, merging, collapse



Between deformations, the contours are re-parametrized by using a grid to detect topology changes



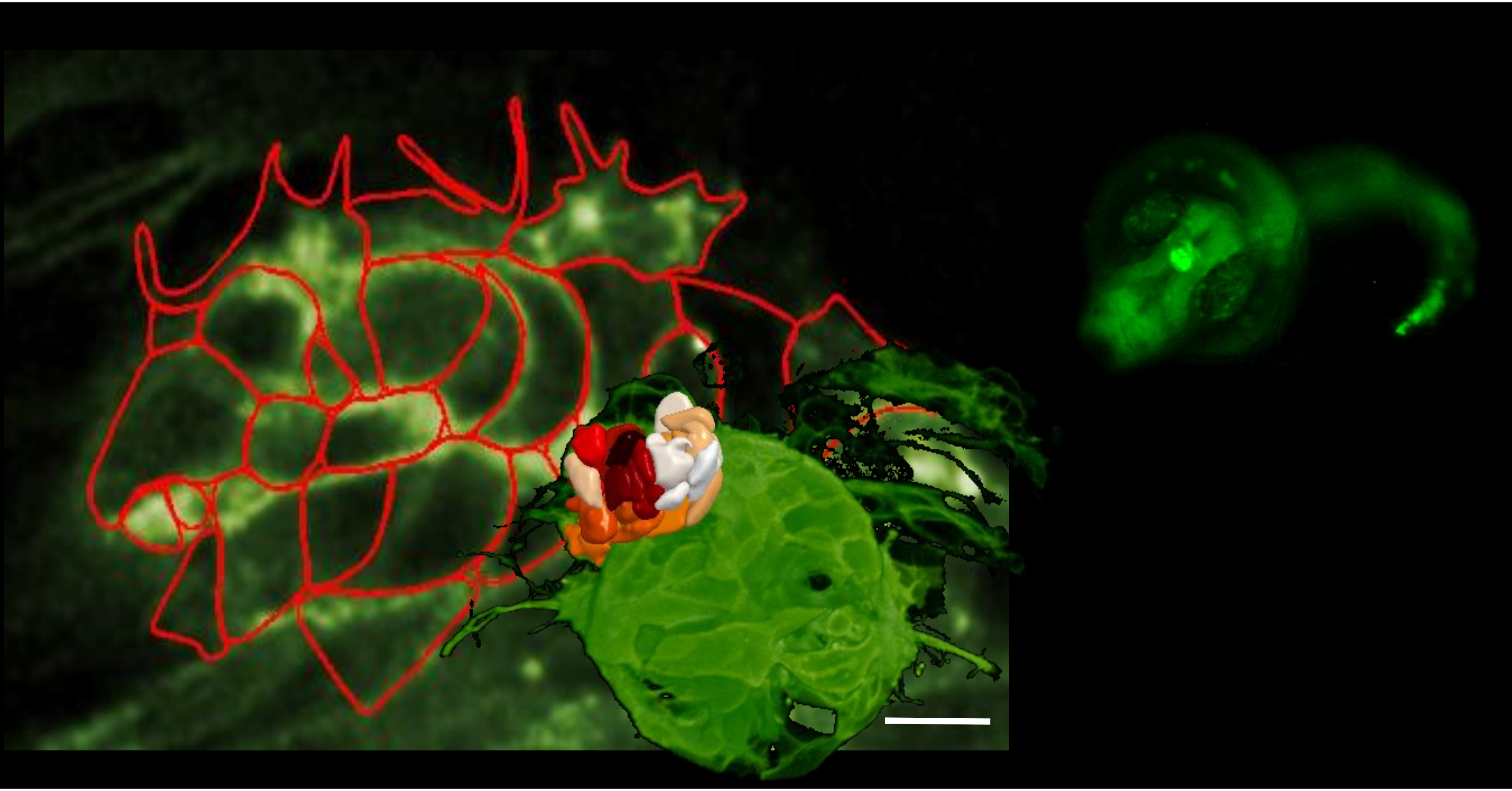
This is an algorithmic approach, "additional" to the mathematical optimization model. The computation becomes more intensive



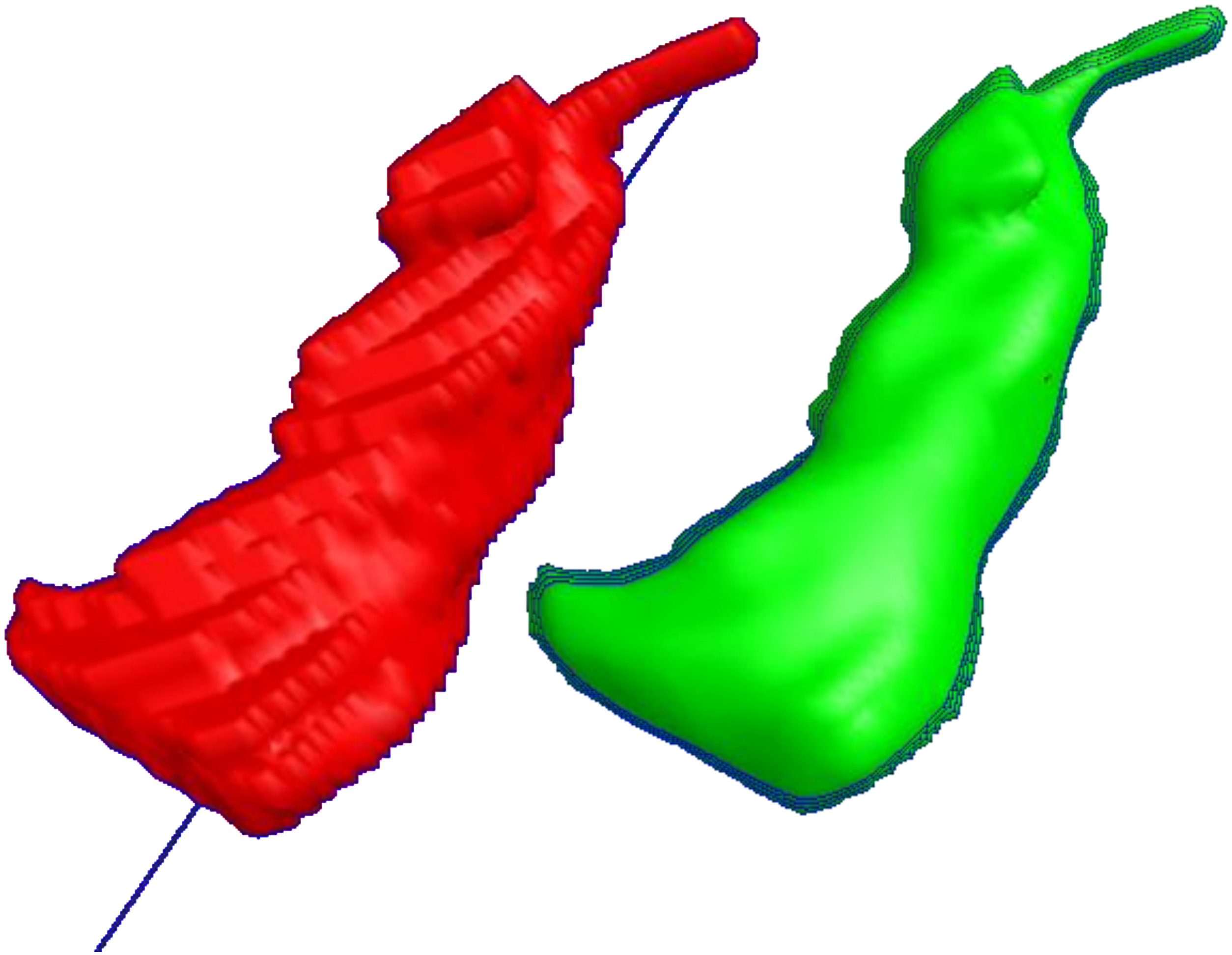
t-snakes movie time!

F Olmos (2009)

Hardest cases: initialization by manual ROI sketching



- Sample 3D ROI



Referencias

- John Russ
  The image processing handbook, 4th ed.
  CRC press, 2002

- Nixon, Aguado
  Feature extraction & image processing, 1st|2nd|3rd ed.
  Academic Press, 2002|2008|2012

- David Marr
  Vision
  MIT press, 1982