MAIA

Computer Aided Diagnosis (CAD)

Deformable Models Segmentation

Active contours

- The next slices are copied directly from Chumming Li (Vanderbilt University):

  http://www.engr.uconn.edu/~cmli/

Universitat de Girona

Outline

- Curve evolution and level set methods:
  - Curve Evolution: from snake to general curve evolution
  - Level Set Methods: basic concepts and methods

- Numerical issues:
  - Difference scheme: upwind scheme, ENO interpolation…
  - Reinitialisation, velocity extension …

Advantages of Active Contours and Level Sets

- Nice representation of object boundary:
- Smooth and closed, good for shape analysis and recognition and other applications.
- Sub-pixel accuracy.
- Can incorporate various information such as shape prior and motion.
- Mature mathematical tools can be used: calculus of variations, PDE, differential geometry …

Universitat de Girona

Differential Geometry of Curves

- A planar curve C is a function $\mathcal{C}:[0,1] \rightarrow \Re^2$ where

$$\mathcal{C}(p) = (x(p), y(p))$$

  - Tangent vector

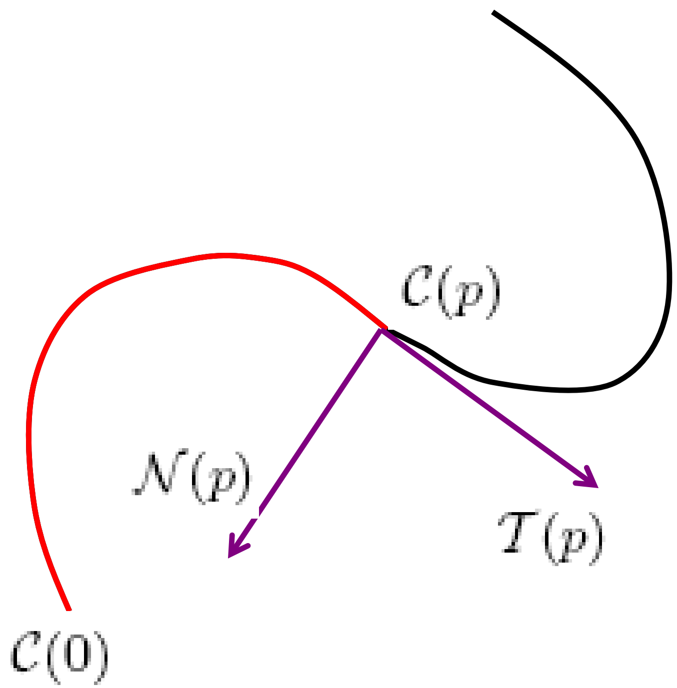
$$\mathcal{T}(p) = \mathcal{C}'(p) = (x'(p), y'(p))$$

  - Normal vector

$$\mathcal{N}(p) = \left(\frac{-y'(p)}{\sqrt{x'(p)^2 + y'(p)^2}}, \frac{x'(p)}{\sqrt{x'(p)^2 + y'(p)^2}}\right)$$

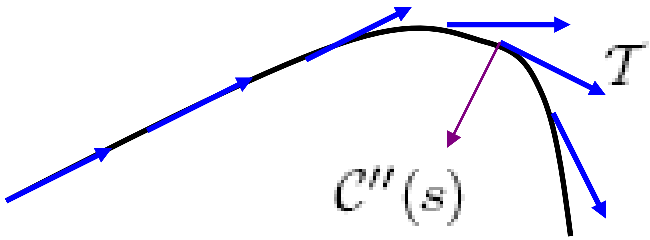
  - Arc length between $\mathcal{C}(0)$ and $\mathcal{C}(p)$

$$s(p) = \int_0^p |\mathcal{C}'(s)| ds$$



Curvature

- Use arc length s as the parameter.

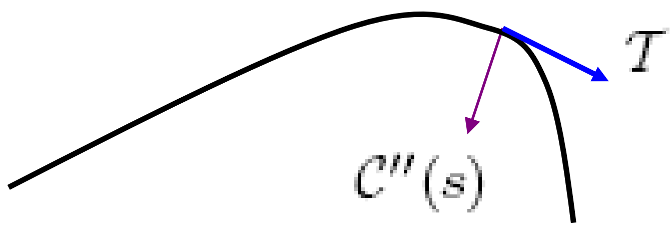


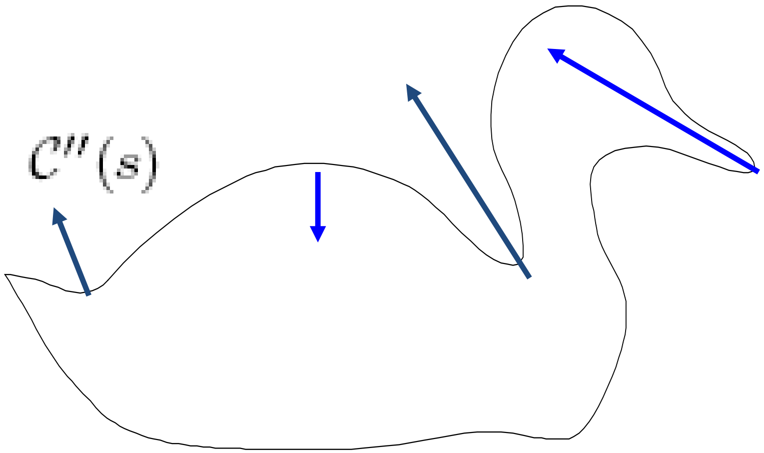
- Then
  - The tangent vector $T(s) = C'(s)$ is a unit vector
  - The unit vector T defines the direction of the curve
  - Curvature describes how fast the curve changes its direction.
  - Take derivative again: $C''(s) = T'(s)$
  - Note that $C'(s) \cdot C'(s) \equiv 1 \Longrightarrow C''(s) \cdot C'(s) = 0$

Universitat de Girona

Curvature (Cont'd)

- Both $\mathcal{C}''(s)$ and $\mathcal{N}$ are orthogonal to $\mathcal{T}$
- Therefore,
  - $\mathcal{C}''(s)$ is parallel to $\mathcal{N}$
  - Actually, $\mathcal{C}''(s) = \kappa(s)\mathcal{N}(s)$





- For a general parameterisation $\mathcal{C}(p) = (x(p), y(p))$

the curvature can be expressed as:

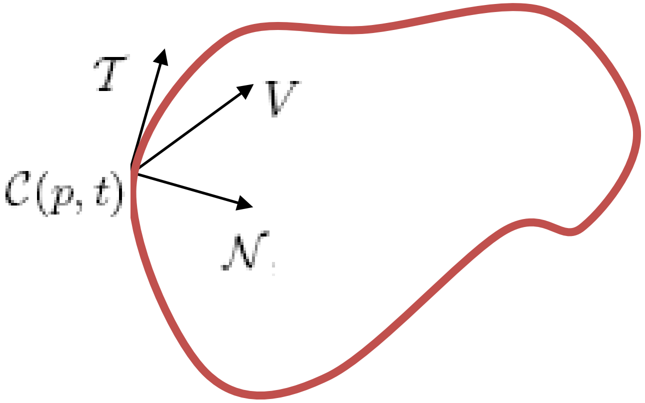
$$\kappa(p) = \frac{x'(p)y''(p) - y'(p)x''(p)}{(x'(p)^2 + y'(p)^2)^{3/2}}$$

Dynamic Curves

- A dynamic curve is a time dependent curve

$$\mathcal{C}(p,t) = (x(p,t), y(p,t))$$

- The motion of the curve is governed by a curve evolution equation: $$\frac{\partial \mathcal{C}(p,t)}{\partial t} = V(p,t)$$



- The tangent vector $\mathcal{T}$ and the normal vector $\mathcal{N}$ forms a basis of $\Re^2$

$$\frac{\partial \mathcal{C}(p,t)}{\partial t} = \alpha \mathcal{T} + \beta \mathcal{N}$$

Universitat de Girona

Geometric Curve Evolution

- Let β be an intrinsic quantity. If $C(p,t)$ evolves according to

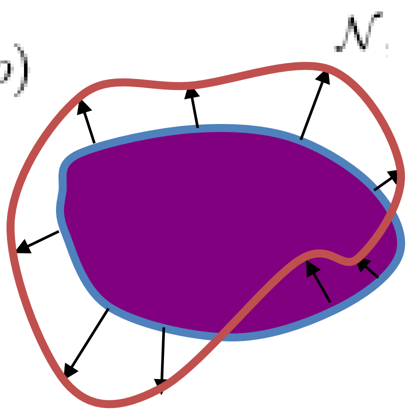
$$\frac{\partial C(p,t)}{\partial t} = \alpha T + \beta N$$

- Then, there exists another parameterization $\bar{C}(q,t)$ of $C(p,t)$ such that $\bar{C}(q,t)$ is solution of:

$$\frac{\partial \bar{C}(q,t)}{\partial t} = \bar{\beta} N$$

where $\bar{\beta} = \beta$ at the same point

$\bar{C}(q) = C(p)$



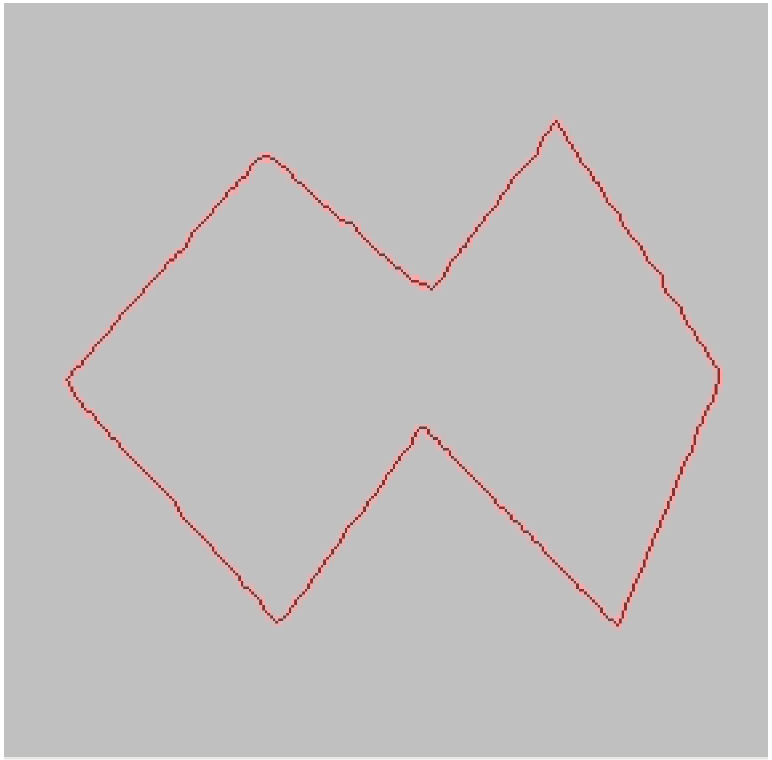
- Therefore, the general geometric curve is:

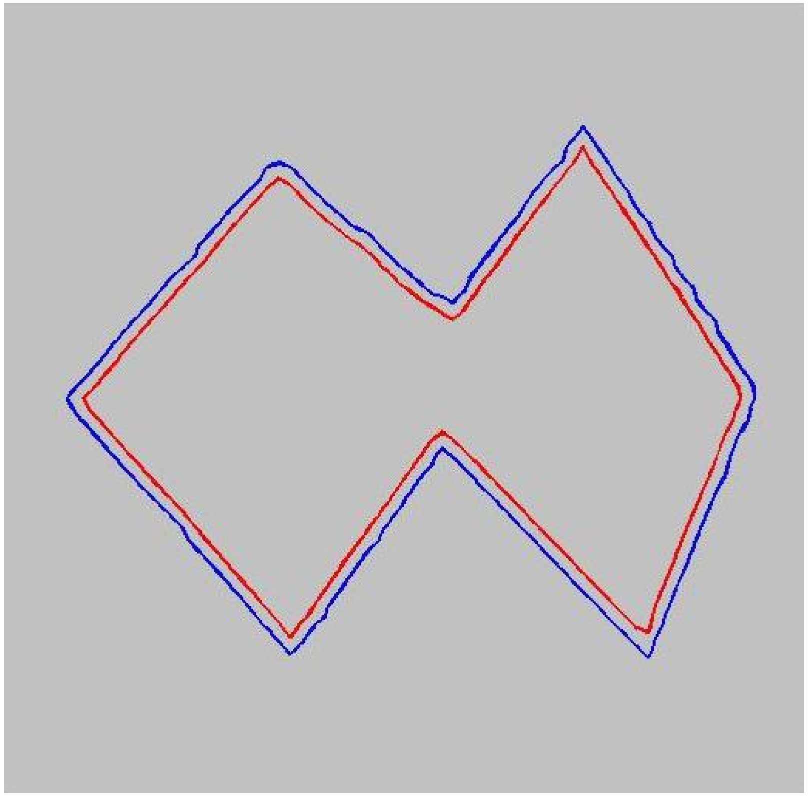
$$\frac{\partial C}{\partial t} = F N$$

Curve Motion

- Constant Speed Motion (Area decreasing/increasing)

$$\frac{\partial \mathcal{C}}{\partial t} = c\mathcal{N}$$

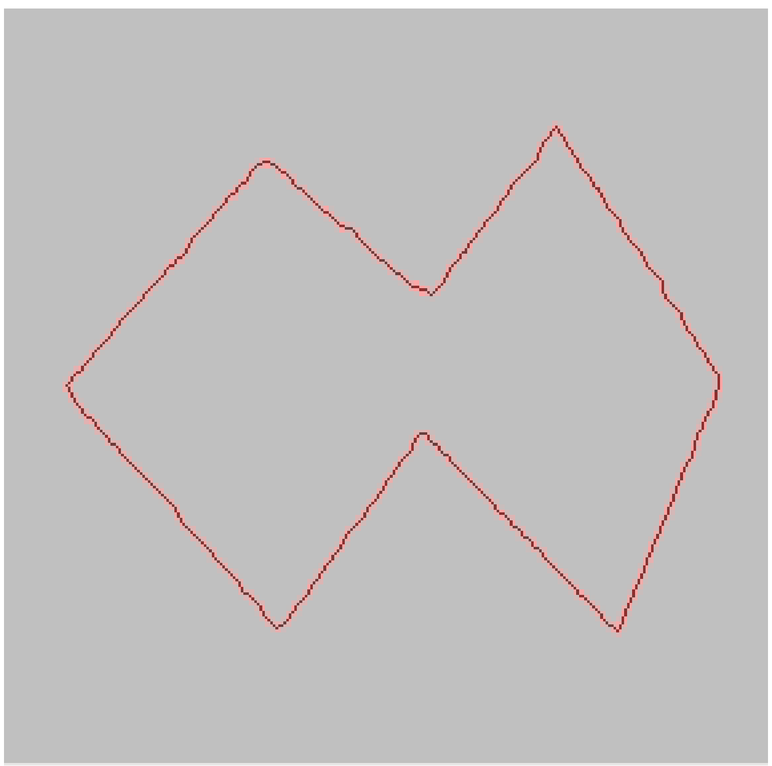


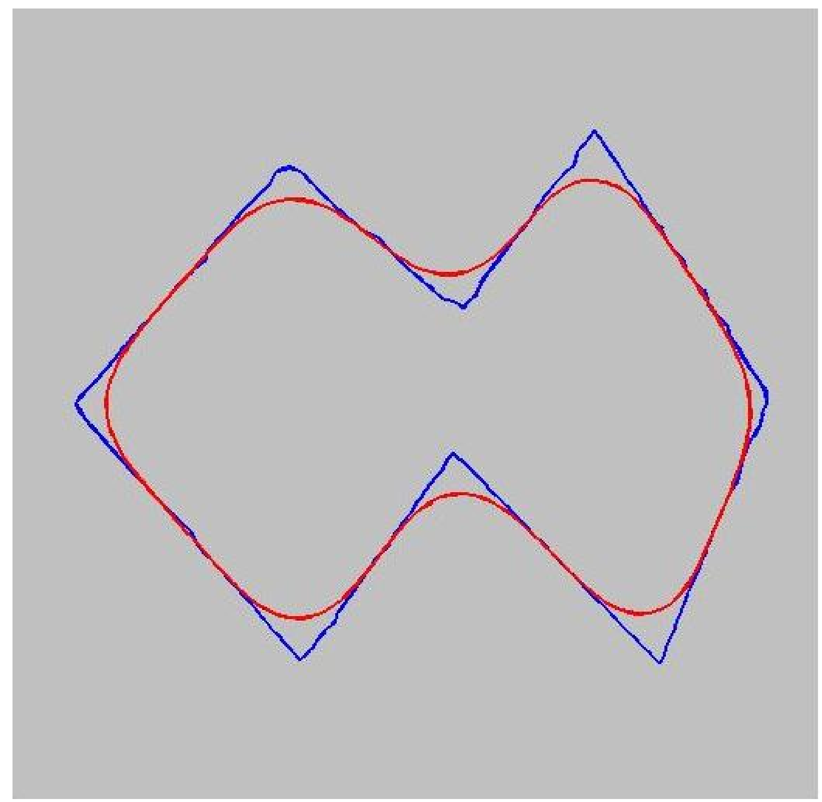


Curve Motion

- Mean curvature motion (Length shortening flow)

$$\frac{\partial \mathcal{C}}{\partial t} = \kappa \mathcal{N}$$

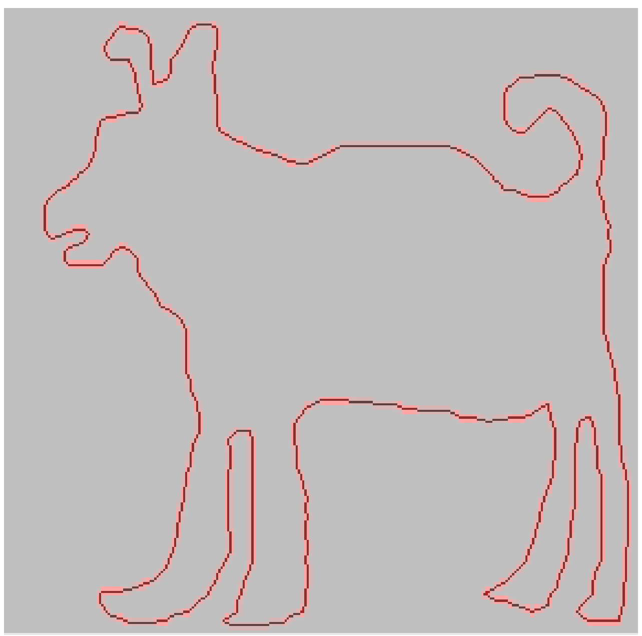


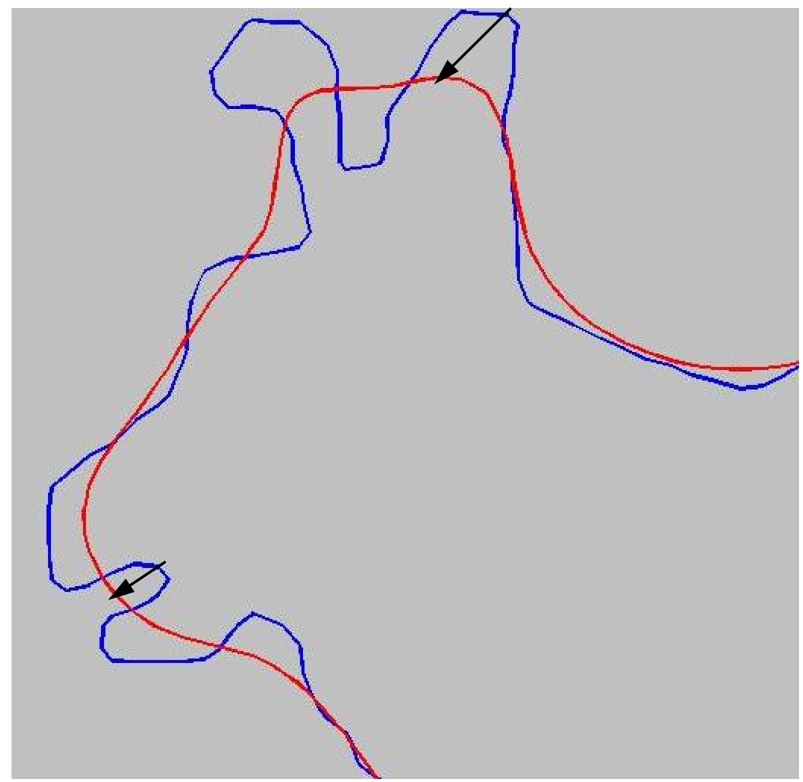


Curve Motion

- Mean curvature motion minimises the arc length of the contour

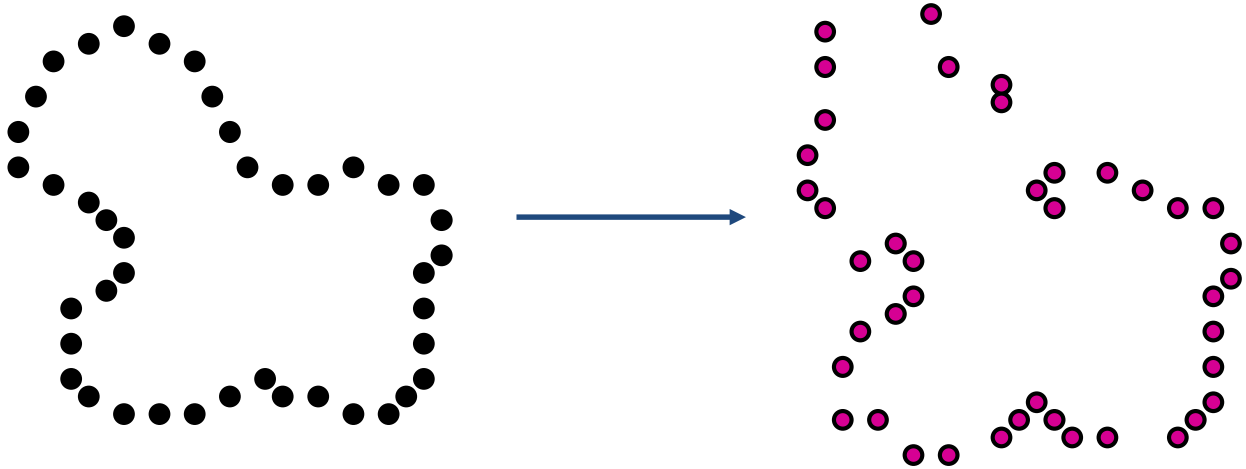
$$\frac{\partial \mathcal{C}}{\partial t} = \kappa \mathcal{N}$$



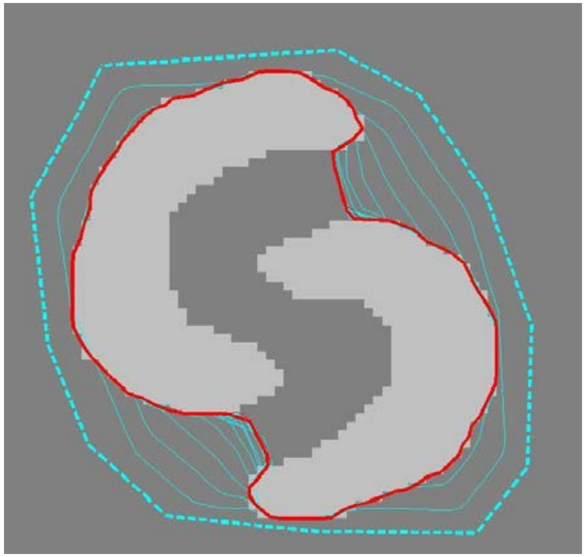


Difficulties of Parameterised Curve Evolution

- Re-parameterisation during evolution: very difficult for 3D surface

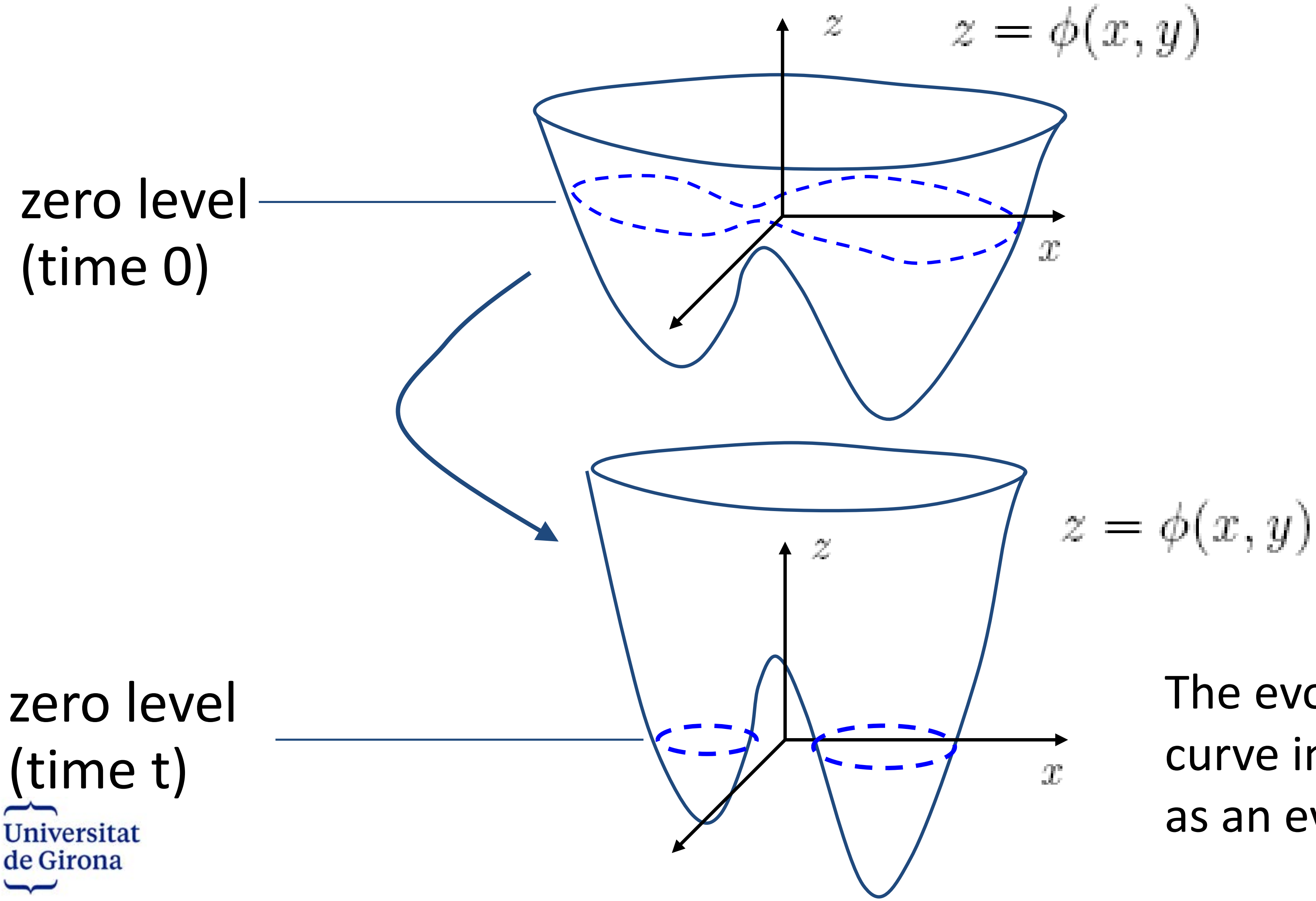


- Cannot handle topological changes



Level Set Representation of Curves

- Level set representation (or addressing the problem in a higher dimension)



The evolution of the curve in time is seen as an evolving surface

Universitat de Girona

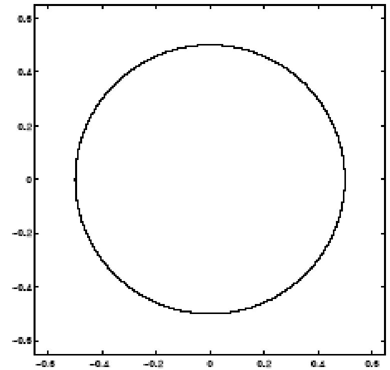
Advantages of Level Set Methods

- All computation is carried out on a fixed grid!!
- So, there is no need for re-parameterization,
- and, topological changes are handled naturally.

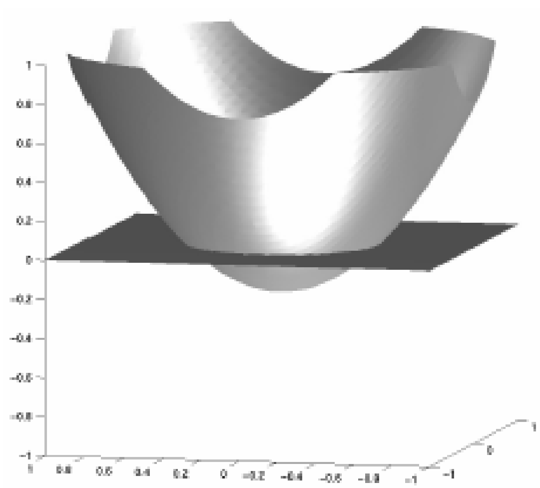
Level Set Functions

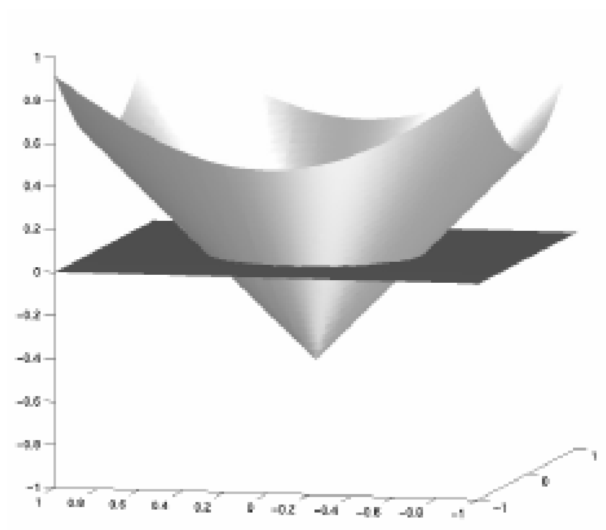
- The speed function defines the evolution of the curve:

contour



Level set functions with the same zero level set

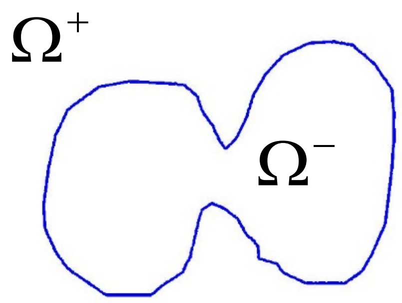




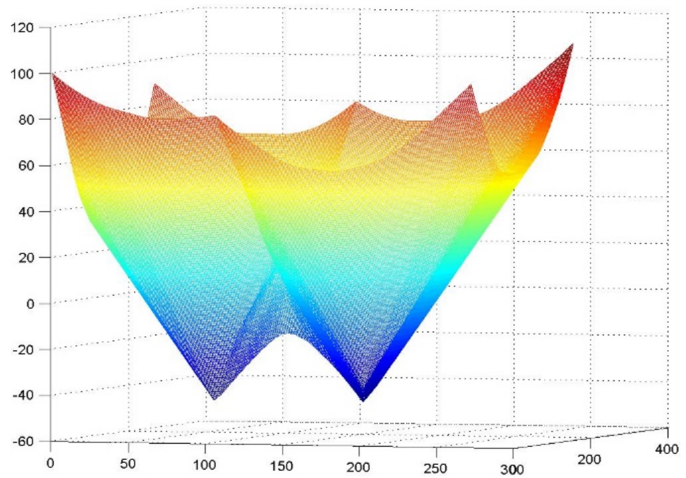
Signed Distance Function

- External pixels are assigned with a positive value, while internal pixels are negative:

Contour C



Signed distance function



$$\phi(\mathrm{x}) = \begin{cases} dist(\mathrm{x}, C) & \text{if x is outside } C \\ 0 & \mathrm{x} \in C \\ -dist(\mathrm{x}, C) & \text{if x is inside } C \end{cases}$$

Useful Calculus Facts in Level Set Formulation

- If a curve $\mathcal{C}(p) = (x(p), y(p))$ is a level set of a function $\phi(x, y)$ then, the normal vector and the curvature are computed from the embedding function $\phi(x, y)$

$$\mathcal{N} = -\frac{\nabla \phi}{|\nabla \phi|} \qquad \kappa = \operatorname{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)$$

Universitat de Girona

From Curve Evolution to Level Set Evolution

- Curve evolution $\dfrac{\partial \mathcal{C}}{\partial t} = F\mathcal{N}$
- where *F* is the speed function, *N* is normal vector to the curve *C*.
- Embed the dynamic curve $\mathcal{C}(p,t)$ as the zero level set of a time dependent function $\phi(\mathbf{x},t)$, i.e. $\phi(\mathcal{C}(p,t),t) = 0$
- Take derivative with respect to time $\dfrac{\partial \phi}{\partial t} + \nabla\phi \cdot \dfrac{\partial \mathcal{C}}{\partial t} = 0$
- After some changes…

$$\mathcal{N} = -\frac{\nabla\phi}{|\nabla\phi|} \qquad \frac{\partial \phi}{\partial t} = -\nabla\phi \cdot \frac{\partial \mathcal{C}}{\partial t} = -\nabla\phi \cdot F\mathcal{N} = -\nabla\phi \cdot \left(-F\frac{\nabla\phi}{|\nabla\phi|}\right)$$

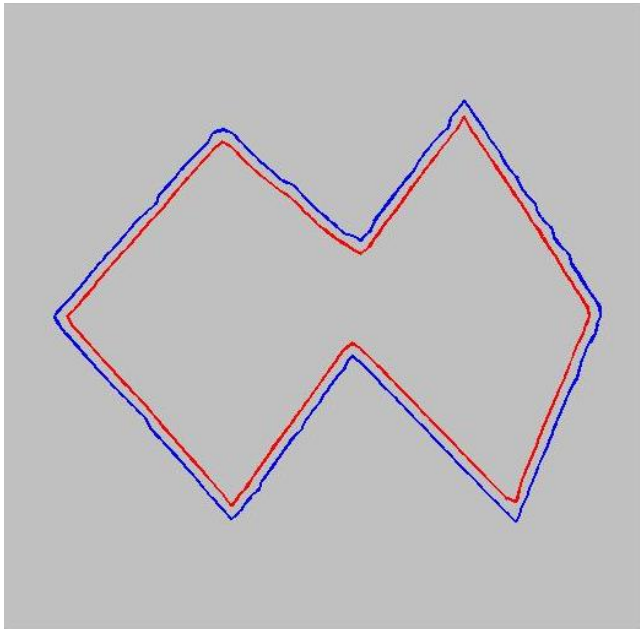
- Level set evolution $\dfrac{\partial \phi}{\partial t} = F|\nabla\phi|$

Universitat de Girona

Special Cases

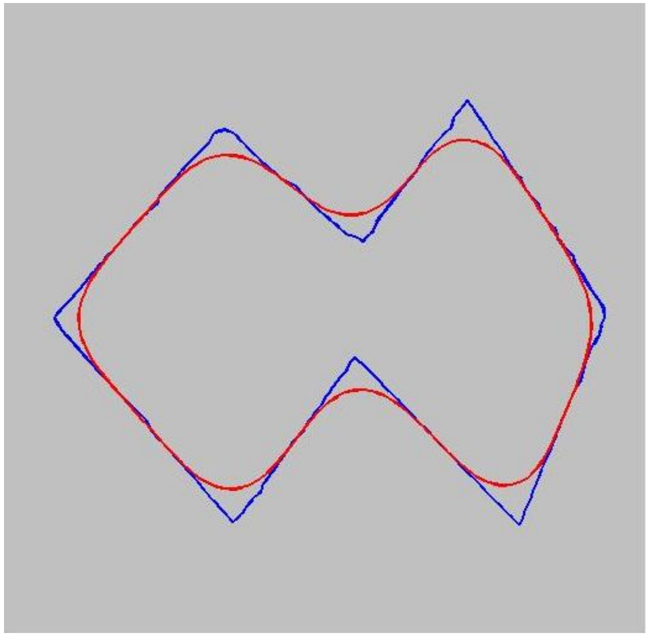
Constant Speed Motion

$$\frac{\partial \mathcal{C}}{\partial t} = c\mathcal{N} \longrightarrow \frac{\partial \phi}{\partial t} = c|\nabla \phi|$$



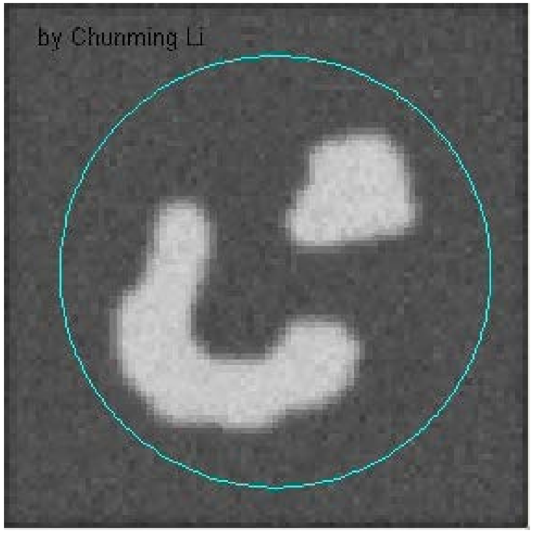
Mean curvature flow

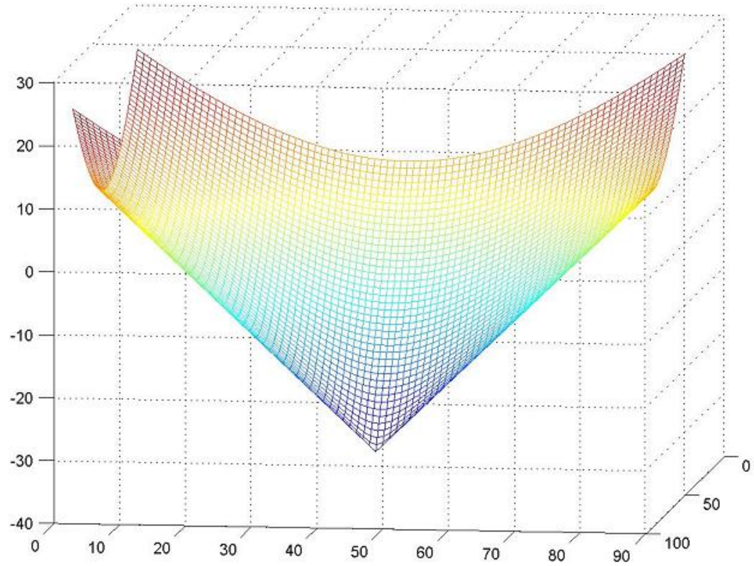
$$\frac{\partial \mathcal{C}}{\partial t} = \kappa\mathcal{N} \longrightarrow \frac{\partial \phi}{\partial t} = \mathrm{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)|\nabla \phi|$$

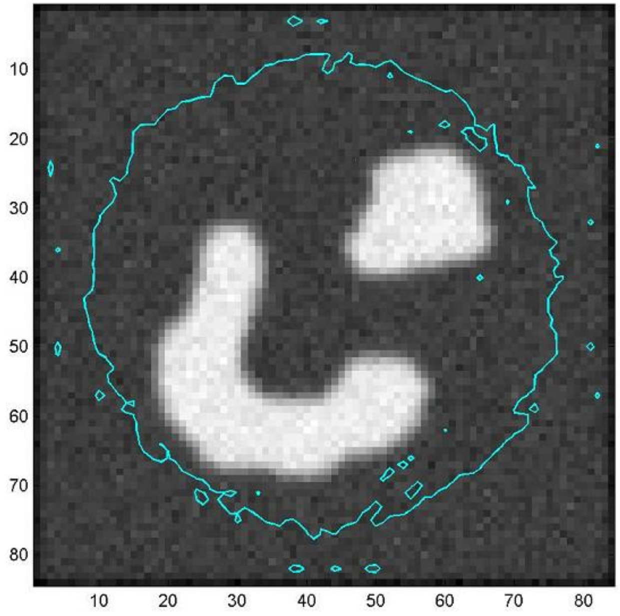


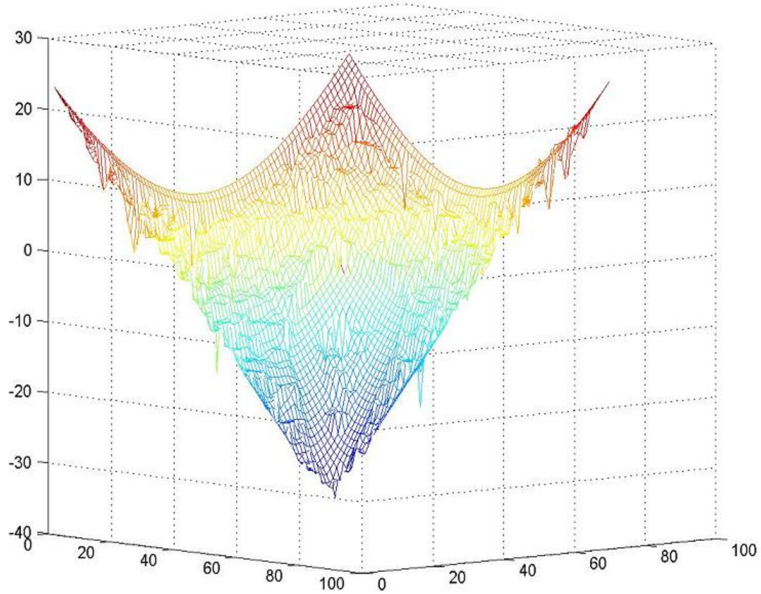
Universitat de Girona

Problem: curve degradation after some iterations

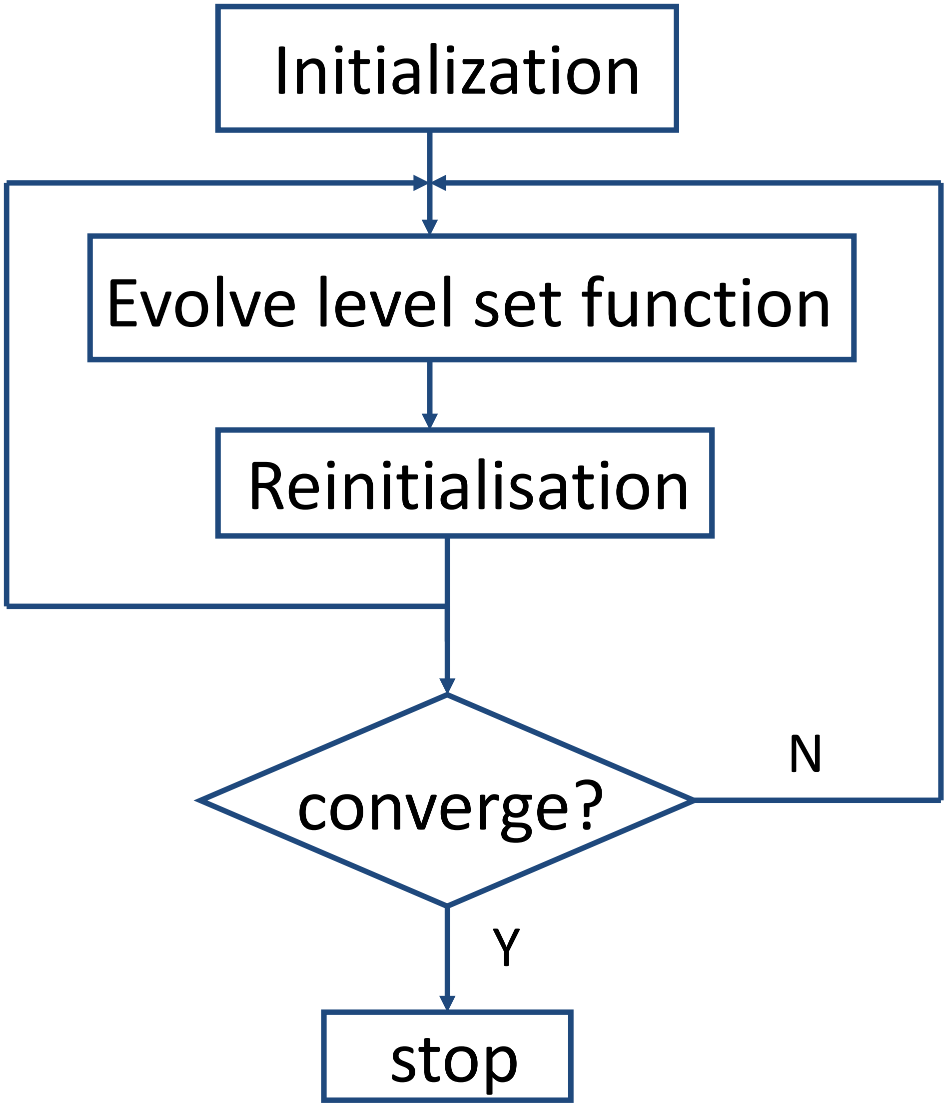








Implementation of Standard Level Set Methods



Explicit Euler Scheme

- Consider general evolution equation:

Approximate temporal derivatives by forward difference

$$\frac{\partial \phi}{\partial t} = L(\phi, \frac{\partial \phi}{\partial x}, \frac{\partial \phi}{\partial y}, \ldots)$$

Approximate spatial derivatives by certain difference scheme

$$\frac{\phi_{i,j}^{n+1} - \phi_{i,j}^{n}}{\Delta t} = L(\phi_{i,j}^{n}, D^x \phi_{i,j}^{n}, D^y \phi_{i,j}^{n}, \ldots)$$

- Update equation at each iteration:

$$\phi_{i,j}^{n+1} = \phi_{i,j}^{n} + \Delta t L(\phi_{i,j}^{n}, D^x \phi_{i,j}^{n}, D^y \phi_{i,j}^{n}, \ldots)$$

Basic Finite Difference Scheme for Spatial Variable

- Level sets are implemented using the up-wind scheme:

- Backward difference

$$D_{ij}^{-x} = \frac{\phi_{ij}^{n} - \phi_{i-1,j}^{n}}{\Delta x} \qquad D_{ij}^{-y} = \frac{\phi_{ij}^{n} - \phi_{i,j-1}^{n}}{\Delta y}$$

- Forward difference

$$D_{ij}^{+x} = \frac{\phi_{i+1,j}^{n} - \phi_{i,j}^{n}}{\Delta x} \qquad D_{ij}^{+y} = \frac{\phi_{i,j+1}^{n} - \phi_{i,j}^{n}}{\Delta x}$$

- Central difference

$$D_{ij}^{0x} = \frac{\phi_{i+1,j}^{n} - \phi_{i-1,j}^{n}}{2\Delta x} \qquad D_{ij}^{0y} = \frac{\phi_{i,j+1}^{n} - \phi_{i,j-1}^{n}}{2\Delta y}$$

Universitat de Girona

Mean Curvature Motion

- Mean curvature motion

$$\frac{\partial \phi}{\partial t} = \mathrm{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)|\nabla \phi|$$

- Update equation:

$$\phi_{i,j}^{n+1} = \phi_{i,j}^{n} + \triangle t K_{i,j}^{n}\sqrt{(D^{0x}\phi_{i,j}^{n})^2 + (D^{0y}\phi_{i,j}^{n})^2}$$

where:

$$K_{i,j}^{n} = D^{-x}\left(\frac{D^{+x}\phi_{i,j}^{n}}{\sqrt{(D^{+x}\phi_{i,j}^{n})^2 + (D^{0y}\phi_{i,j}^{n})^2}}\right) + D^{-y}\left(\frac{D^{+y}\phi_{i,j}^{n}}{\sqrt{(D^{0x}\phi_{i,j}^{n})^2 + (D^{+y}\phi_{i,j}^{n})^2}}\right)$$

Motion in Normal Direction

- Motion in normal direction:

$$\frac{\partial \phi}{\partial t} = g|\nabla \phi|$$

- Right hand side is approximated by:

$$g|\nabla \phi| \approx \max(g_{i,j}, 0)\, \nabla^{+} + \min(g_{i,j}, 0) \nabla^{-}$$

where

$$\nabla^{+} = [\max(D_{ij}^{-x}, 0)^2 + \min(D_{ij}^{+x}, 0)^2 + \max(D_{ij}^{-y}, 0)^2 + \min(D_{ij}^{+y}, 0)^2]^{1/2}$$

$$\nabla^{-} = [\max(D_{ij}^{+x}, 0)^2 + \min(D_{ij}^{-x}, 0)^2 + \max(D_{ij}^{+y}, 0)^2 + \min(D_{ij}^{-y}, 0)^2]^{1/2}$$

- Hence, the update equation is:

$$\phi_{i,j}^{n+1} = \phi_{i,j}^{n} + \Delta t(\max(g_{i,j}, 0)\, \nabla^{+} + \min(g_{i,j}, 0) \nabla^{-})$$

Geodesic Active Contour

- The geodesic active contour is defined as (the last term stands for other terms)

$$\frac{\partial \phi}{\partial t} = g|\nabla \phi| \mathrm{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right) + cg|\nabla \phi| + \langle \nabla g, \nabla \phi \rangle$$

- With update equation:

$$\begin{aligned}\phi_{i,j}^{n+1} = \phi_{i,j}^{n} + \triangle t[g_{i,j} K_{i,j}^{n} \sqrt{(D^{0x}\phi_{i,j}^{n})^2 + (D^{0y}\phi_{i,j}^{n})^2} \\ + c(\max(g_{i,j}, 0) \bigtriangledown^{+} + \min(g_{i,j}, 0) \bigtriangledown^{-}) \\ + (\max((g_x)_{i,j}, 0) D_{ij}^{-x} + \min((g_x)_{i,j}, 0) D_{ij}^{+x} \\ + \max((g_y)_{i,j}, 0) D_{ij}^{-y} + \min((g_y)_{i,j}, 0) D_{ij}^{+y})]\end{aligned}$$

General Evolution Equation

- In general, the level set evolution is defined as:

$$\frac{\partial \phi}{\partial t} = F|\nabla \phi|$$

$$F = \alpha \mathrm{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right) + g + V \cdot \frac{\nabla \phi}{|\nabla \phi|}$$

$$F = F_{curv} + F_{prop} + F_{adv}$$

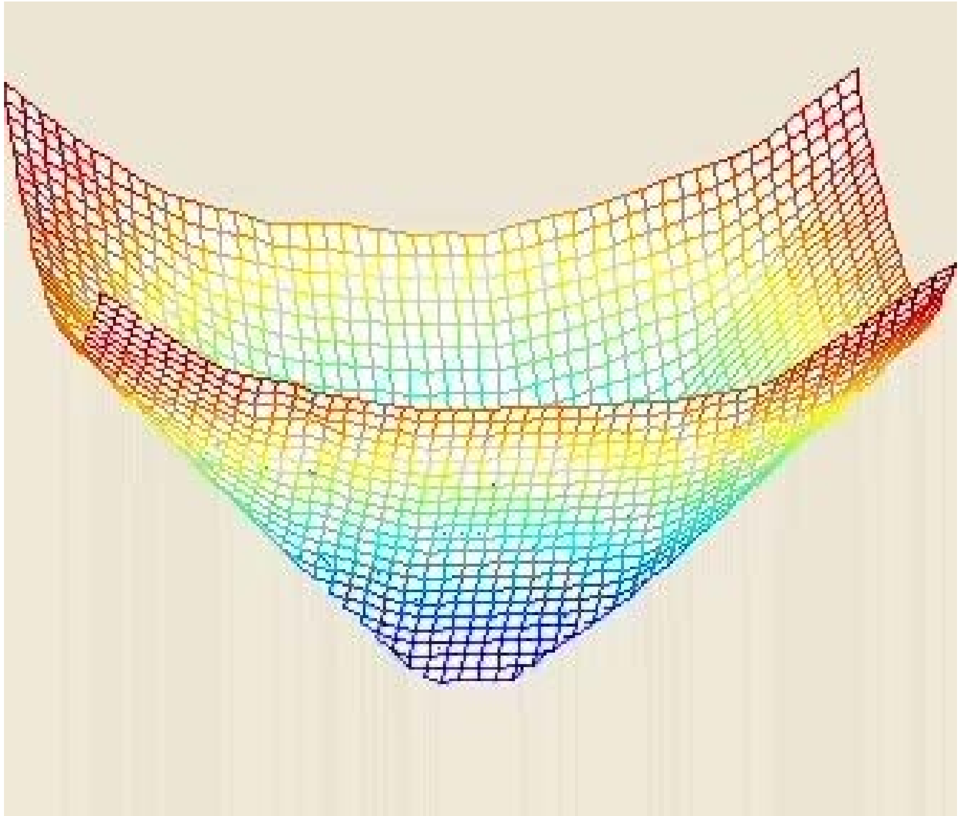
- For a stable evolution, the time step and spatial step must satisfy the CFL (Courant–Friedrichs–Lewy) condition:

$$\triangle t \leq \frac{\min\{\triangle x, \triangle y\}}{\max |F_{i,j}|}$$

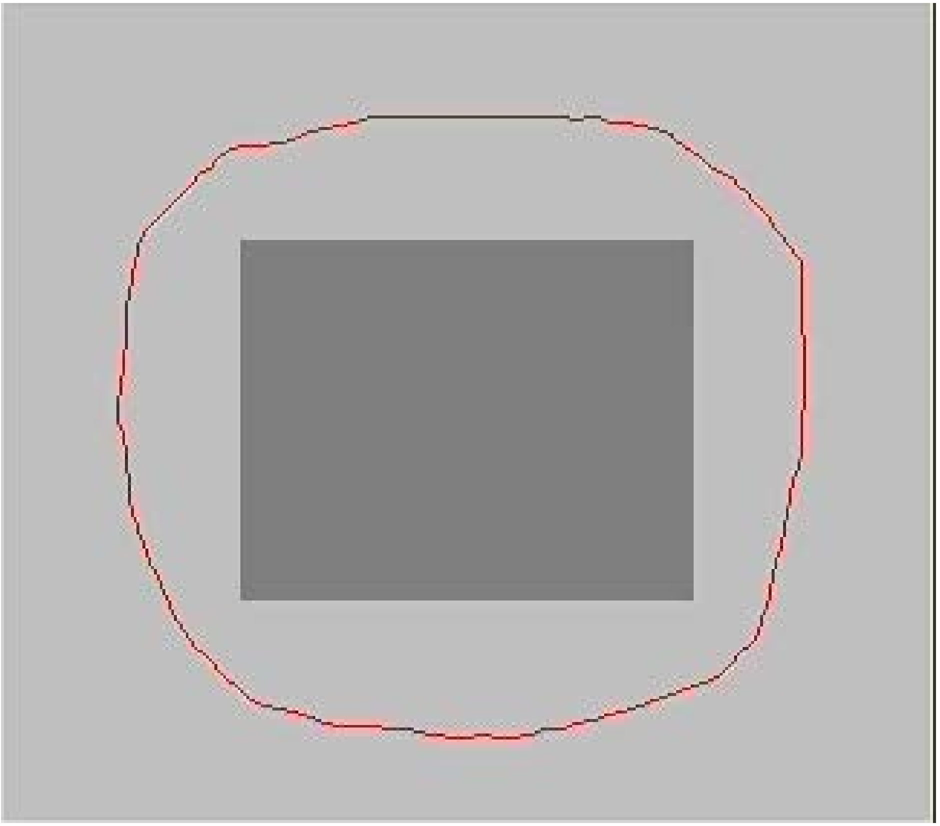
Unstable Evolution in Standard Level Set Methods

- Unstable evolution introduce noise in the curve evolution:

Evolution of level set function



Evolution of zero level set



Reinitialisation (Redistance)

- Reinitialisation: periodically stop the evolution and repair the degraded level set function as a signed distance function.
- Solve to steady state:

$$\frac{\partial \phi}{\partial t} = \mathrm{sign}(\phi_0)(1 - |\nabla \phi|)$$

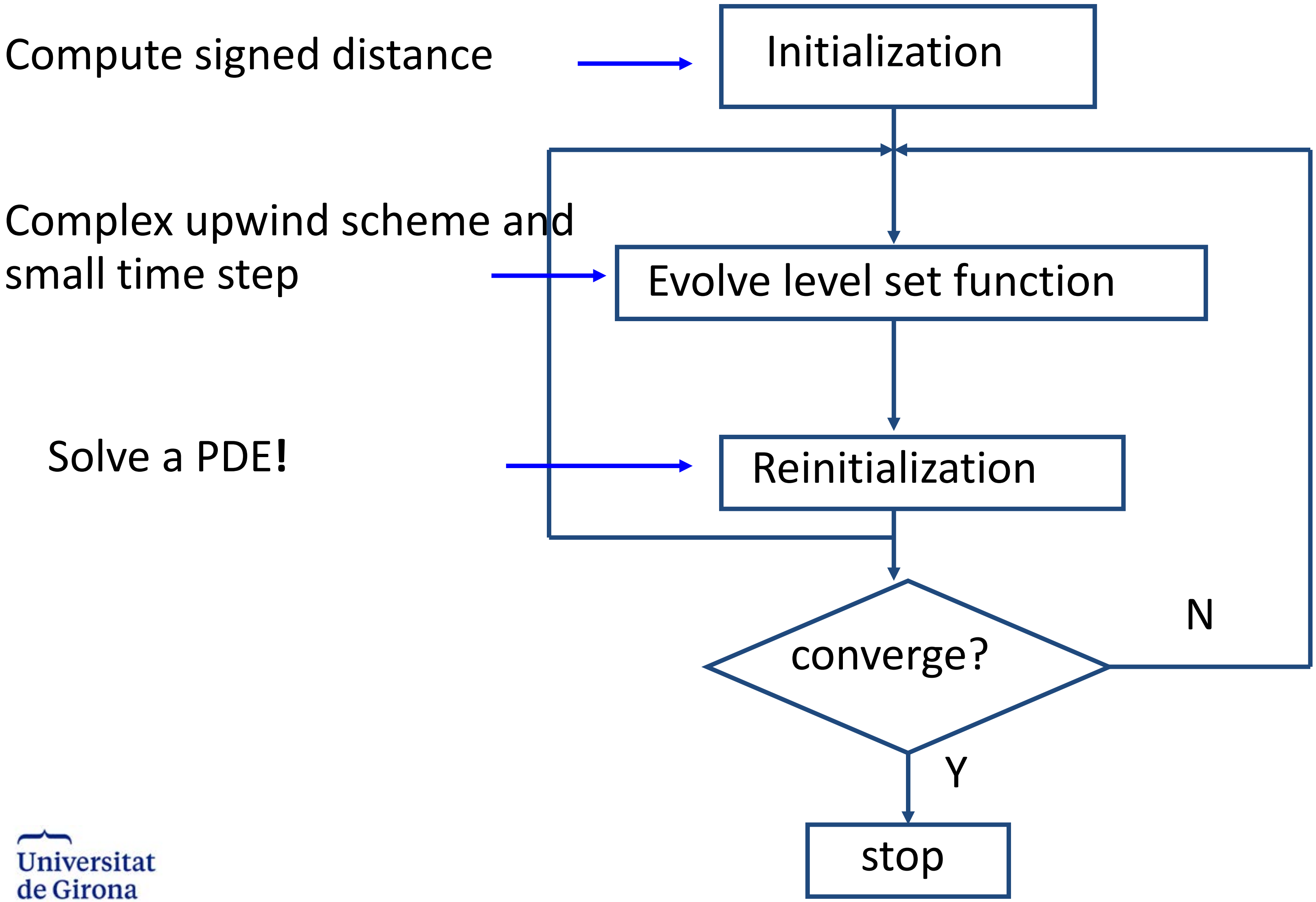
Reinitialisation equation

$$\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x})$$

- Drawbacks of Reinitialisation:
  - Still a serious problem: when and how to reinitialize?
  - Computationally expensive

Summary of Standard Level Set Methods

- In summary,



Level Set Evolution without Reinitialization (Li et al, 2005)

- Goal: Find a level set evolution algorithm that can simultaneously move the zero level set while maintaining the signed distance profile throughout the entire evolution.

- Characteristics of signed distance function:

$$|\nabla \phi| = 1 \iff \text{signed distance function + constant}$$

- Deviation from a signed distance function

$$\mathcal{P}(\phi) \triangleq \int_{\Omega} \frac{1}{2}(|\nabla \phi| - 1)^2 dxdy$$

Region-based level sets methods

- Until now, the level sets where guided using only gradient information
- The level sets can be guided without the gradient. How? By comparing the characteristics of the inner & outer regions
- The main approach was the Chan & Vese paper: active contours without edges, which is based on the Mumfor-Shah functional

Mumford-Shah Functional

$$F^{\mathrm{MS}}(u, C) = \mu \cdot \mathrm{Length}(C) + \lambda \int_{\Omega} |u_0(x, y) - u(x, y)|^2 \, dx \, dy + \int_{\Omega \setminus C} |\nabla u(x, y)|^2 \, dx \, dy$$

Regularization term   Data fidelity term   Smoothing term

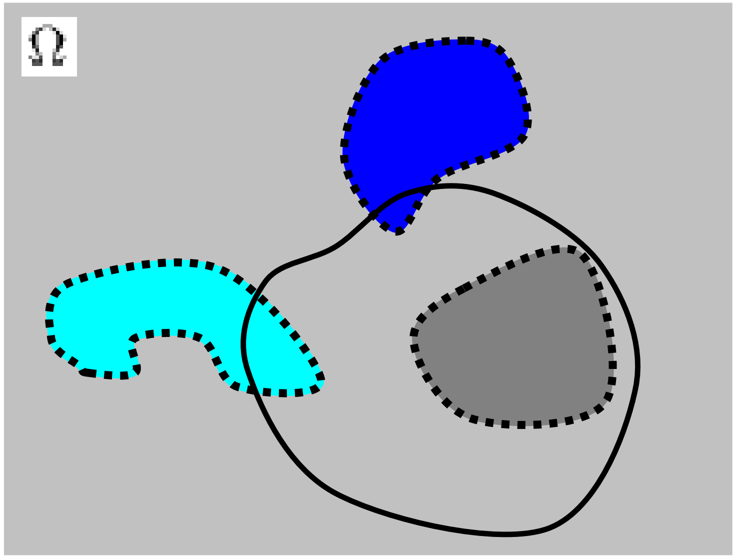
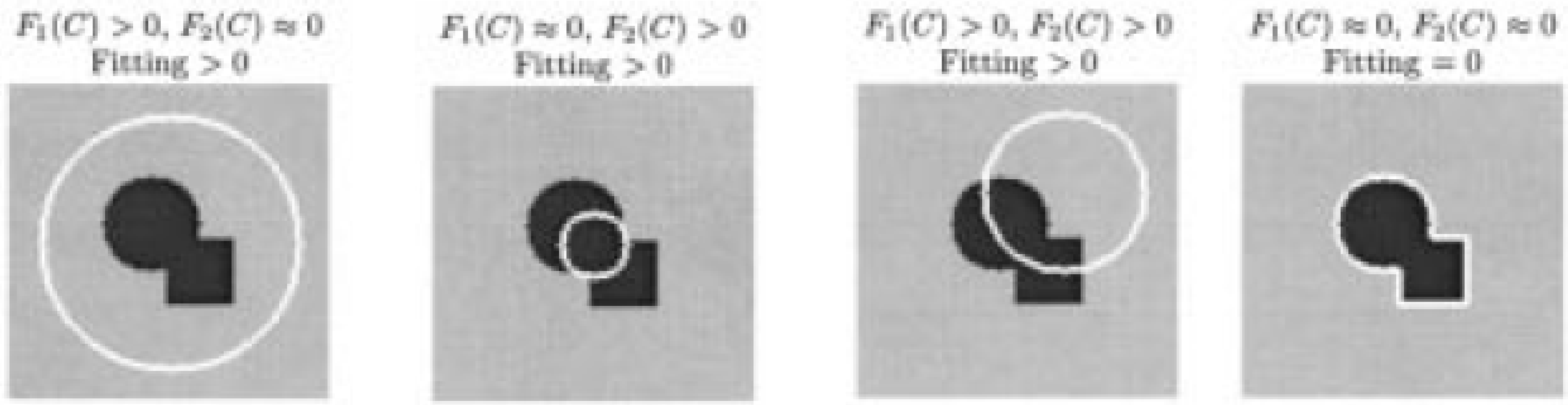


image u_0.

Universitat de Girona

Active Contours without Edges
(Chan & Vese 2001)

$$F_1(C) + F_2(C) = \int_{inside(C)} |u_0(x, y) - c_1|^2 \, dx \, dy + \int_{outside(C)} |u_0(x, y) - c_2|^2 \, dx \, dy$$



Active Contours without Edges

$$\begin{aligned} E^{CV}(\phi, c_1, c_2) &= \mu \int \delta(\phi(x,y))|\nabla\phi(x,y)|dxdy \\ &+ \int_\Omega H(\phi(x,y))|I(x,y) - c_1|^2 dxdy \\ &+ \int_\Omega (1 - H(\phi(x,y)))|I(x,y) - c_2|^2 dxdy \end{aligned}$$

$$\frac{\partial\phi}{\partial t} = \delta(\phi)[\mu \mathrm{div}(\frac{\nabla\phi}{|\nabla\phi|}) - (I - c_1)^2 + (I - c_2)^2]$$

$$c_1 = \frac{\int_\Omega I(x,y)H(\phi(x,y))dxdy}{\int_\Omega H(\phi(x,y))dxdy}$$

$$c_2 = \frac{\int_\Omega I(x,y)[1 - H(\phi(x,y))]dxdy}{\int_\Omega [1 - H(\phi(x,y))]dxdy}$$

Universitat de Girona

Bibliography

- Level sets:
  - Works of Osher & Sethian. Many papers & free reports. Also, see the webpages (https://math.berkeley.edu/~sethian/ http://www.math.ucla.edu/~sjo/)
  - Paper of Caselles: “Geodesic active contours” => showing the equivalence of snakes (Kass) & level sets

- Improvements:
  - Region based: Chan & Vese
  - No parameterisation: see the work of Chumming Li (http://www.imagecomputing.org/~cmli/DRLSE/)

Universitat de Girona