Active Contours (Snakes)

Dmitry Sorokin

Laboratory of Mathematical Methods of Image Processing
Faculty of Computational Mathematics and Cybernetics
Lomonosov Moscow State University

Spring semester 2021

Course outline

- Active Contours (Snakes)
- Level Set Methods: Introduction and Fast Marching Algorithm
- Numerical Schemes for Level Set Methods and Geodesic Active Contours
- Segmentation using Level Set Methods: Region Based Active Contours

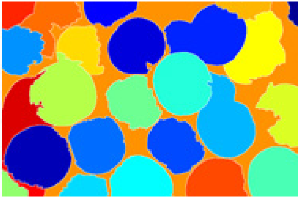
Contents

Contents

Contents

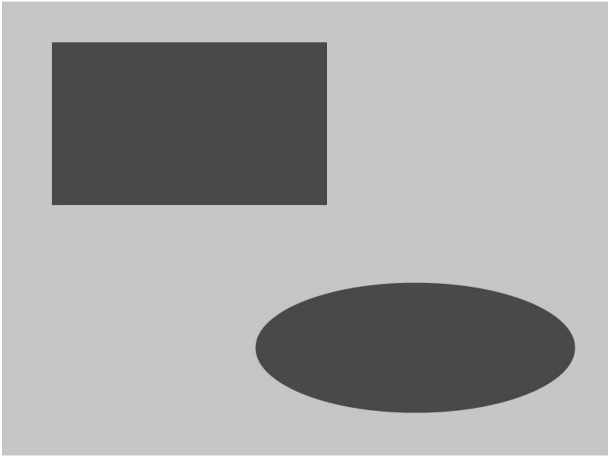
What is segmentation?

- Partition of the image domain into connected regions $X_1, ..., X_n$.

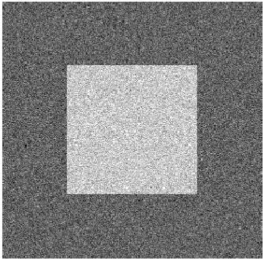


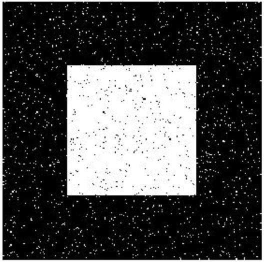
- In the ideal case, every region X_i represents an object in the real world.
- One of the most difficult areas in image analysis: illumination differences, occlusions, lack of a priori knowledge
- No general method exists.

How to segment this image?

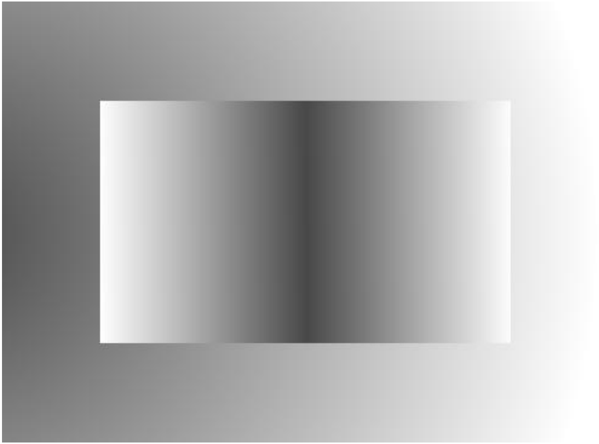


How to segment this image?





How to segment this image?



How to segment this image?

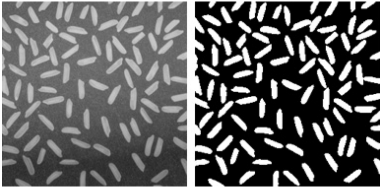
Frequent Assumptions

- Region Based Segmentation: Pixels that belong to the same segment have similar grey values.
- Edge Based Segmentation: There is a jump in the grey values between two adjacent regions. Example: Zero crossings of the Laplacian yield an edge based segmentation with closed contours as segment boundaries.
- Texture Segmentation: Segmenting textures requires a preprocessing step: computation of a suitable texture descriptor. The goal is to achieve almost homogeneous descriptor values within each segment.
- Machine learning: Segmentation principle is derived directly from images during training stage. Training dataset is required. Segmentation quality depends a lot on the training dataset quality.

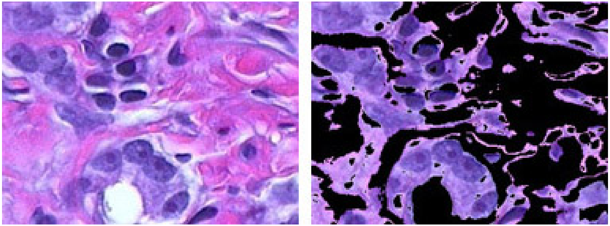
Contents

Classical methods

- Thresholding
  - Simplest method
  - No spatial context, choice of threshold

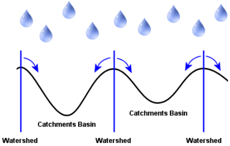


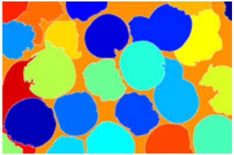
- Color-based Segmentation (e.g. K-means)
  - Uses color information
  - No spatial context again



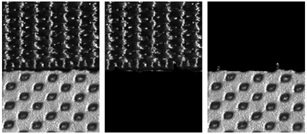
Classical methods

- Watershed algorithm
  - Need to compute gradient magnitude
  - Number of objects corresponds to the number of minima.



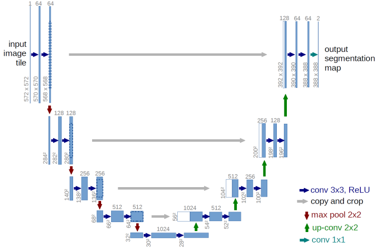
- Texture methods
  - Right choice of texture descriptors (homogeneous descriptor values within each segment)

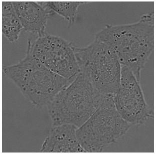


Contents

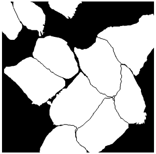
Machine learning

- Classic machine learning
  - SVM, Boosting, Random Forests etc.
  - Since 2012 - mostly Convolutional Neural Networks aka Deep Learning
- Example
  - U-net





→



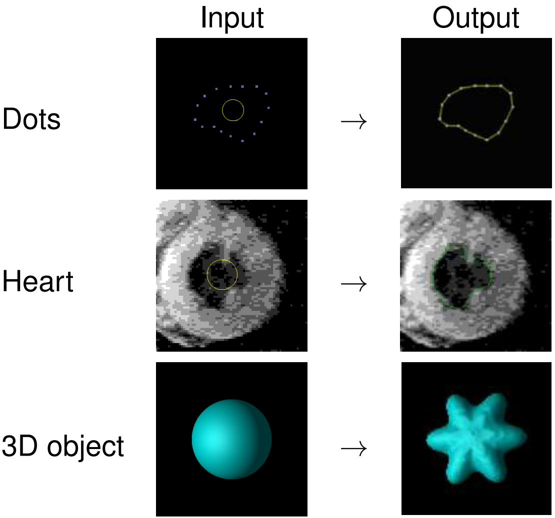
Contents

Energy-based Approaches

- Idea
  - Contour (i.e, curve or surface) with minimal energy is usually searched.
  - Energy is typically composed of two terms:

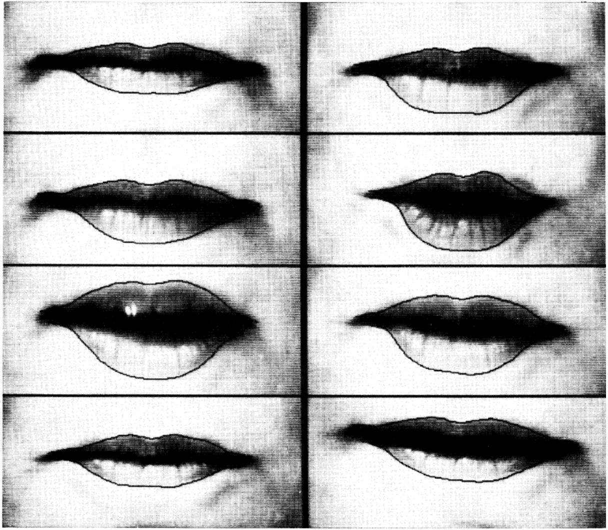
    Internal energy - includes shape constraints
    External energy - includes image data constraints
- Energy minimization often leads to contour evolution driven by external and internal forces.
- The approaches usually suppose that we have a good initial contour close to a state of minimal energy.

Snakes, Motivation



Source: `http://www.iacl.ece.jhu.edu/static/gvf/`

Snakes, Motivation



Source: [Kass et. al. 1987]

Contents

Contents

Active Contour Model

- [Kass et. al 1987]
- The deformable contour (snake) is a mapping:

$$\mathcal{C}(s) : [0, 1] \to \mathbb{R}^2, \quad s \mapsto \mathcal{C}(s) = (x(s), y(s))^T.$$

- We define energy functional of the contour as

$$E_{snake}(\mathcal{C}) = \int_0^1 E_{int}(\mathcal{C}(s)) + E_{ext}(\mathcal{C}(s))ds,$$

where $E_{int}(\mathcal{C}(s))$ is internal energy defined as

$$E_{int}(\mathcal{C}(s)) = \alpha(s)|\mathcal{C}'(s)|^2 + \beta(s)|\mathcal{C}''(s)|^2$$

and $E_{ext}(\mathcal{C}(s))$ is external energy defined as

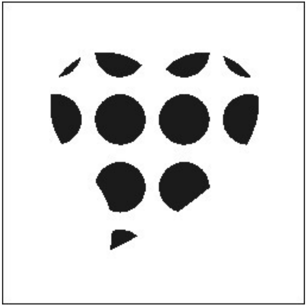
$$E_{ext}(\mathcal{C}(s)) = P(\mathcal{C}(s)),$$

where P is the potential associated to the external forces.

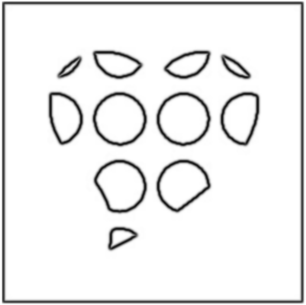
External Energy/Potential: Examples

- Edges
  $P_{edge}(x, y) = -|\nabla f(x, y)|^2$
  or better
  $P_{edge}(x, y) = -|\nabla(G_\sigma(x, y) * f(x, y))|^2$
- Lines (high intensity)
  $P_{line}(x, y) = -f(x, y)$
  or better
  $P_{line}(x, y) = -G_\sigma(x, y) * f(x, y)$
- Combination
  $P(x, y) = -w_{line}P_{line} - w_{edge}P_{edge}$
- Any other task specific [Kondratiev et al., ICPR2016]
- The potential field can be static as well as dynamic.

How to define potential image?



Original image



Edges

Contour Energy Minimization

- We need to minimize $E_{snake}(\mathcal{C})$

$$E_{snake}(\mathcal{C}) = \int_0^1 \alpha(s)|\mathcal{C}'(s)|^2 + \beta(s)|\mathcal{C}''(s)|^2 + P(\mathcal{C}(s))ds =$$

$$= \int_0^1 E(\mathcal{C}(s), \mathcal{C}'(s), \mathcal{C}''(s))ds,$$

- A local minima of the energy functional $E_{snake}(\mathcal{C})$ satisfies necessarily the Euler-Lagrange equation

$$\frac{\partial E}{\partial \mathcal{C}} - \frac{d}{ds}\frac{\partial E}{\partial \mathcal{C}'} + \frac{d^2}{ds^2}\frac{\partial E}{\partial \mathcal{C}''} = 0.$$

Condition for Minima

- Assuming $\alpha(s) = \alpha$ and $\beta(s) = \beta$ we get:

$$\alpha \mathcal{C}'' - \beta \mathcal{C}'''' - \nabla P = 0$$

- We can perceive this equation as a force balance equation

$$F_{int} + F_{ext} = 0$$

where $F_{int} = \alpha \mathcal{C}''(s) - \beta \mathcal{C}''''(s)$ and $F_{ext} = -\nabla P$.

Numerical Solution of Force Balance

- The equation

$$-\alpha\mathcal{C}(s)'' + \beta\mathcal{C}(s)'''' - F_{ext}(\mathcal{C}(s)) = 0$$

can be discretized using finite differences in space (step h)

$$-\frac{a}{h^2}(\mathcal{C}_{i-1} - 2\mathcal{C}_i + \mathcal{C}_{i+1})$$

$$+\frac{b}{h^4}(\mathcal{C}_{i-2} - 4\mathcal{C}_{i-1} + 6\mathcal{C}_i - 4\mathcal{C}_{i+1} + \mathcal{C}_{i+2})$$

$$-(F_1(\mathcal{C}_i), F_2(\mathcal{C}_i)) = 0$$

where $\mathcal{C}_i = \mathcal{C}(ih)$, $a = \alpha(ih)$, $b = \beta(ih)$.

Matrix Form

- This can be written in the matrix form

$$AX = F,$$

where A is a pentadiagonal matrix and X and F consist of curve points $\mathcal{C}_i = (x_i, y_i)$ and forces at these points $F(\mathcal{C}_i) = (F_x(\mathcal{C}_i), F_y(\mathcal{C}_i))$.

$$A = \begin{pmatrix} 2a+6b & -a-4b & b & 0 & \cdots & 0 & b & -a-4b \\ -a-4b & 2a+6b & -a-4b & b & 0 & \cdots & 0 & b \\ b & -a-4b & 2a+6b & -a-4b & b & 0 & \cdots & 0 \\ 0 & b & -a-4b & 2a+6b & -a-4b & & & \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ b & 0 & \cdots & 0 & b & -a-4b & 2a+6b & -a-4b \\ -a-4b & b & 0 & \cdots & 0 & b & -a-4b & 2a+6b \end{pmatrix}$$

$$X = \begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_N & y_N \end{pmatrix}, F = \begin{pmatrix} F_x(x_1, y_1) & F_y(x_1, y_1) \\ F_x(x_2, y_2) & F_y(x_2, y_2) \\ \vdots & \vdots \\ F_x(x_N, y_N) & F_y(x_N, y_N) \end{pmatrix}.$$

Motion Equation

- The energy has many local minima of E. But we are interested in finding a good contour in a given area.
- We suppose we have a rough estimate of the curve and find the curve with (local) minimal energy by solving the associated evolution equation

$$\frac{\partial \mathcal{C}}{\partial t} = F_{int}(\mathcal{C}) + F_{ext}(\mathcal{C})$$

  with initial condition

$$\mathcal{C}(s, 0) = \mathcal{C}_0(s)$$

  and periodic boudary conditions.
- We find a solution of the static problem when the solution $\mathcal{C}(., t)$ stabilizes in t. Then the term $\frac{\partial \mathcal{C}}{\partial t}$ tends to 0 and we achieve a solution of the static problem.

Discrete Motion Equation

- For the evolution equation

$$\frac{\partial \mathcal{C}}{\partial t} = F_{int}(\mathcal{C}) + F_{ext}(\mathcal{C})\,,$$

$$\mathcal{C}(s,0) = \mathcal{C}_0(s)$$

- We use implicit Euler method (time step τ)

$$\frac{\mathcal{C}^{t+1} - \mathcal{C}^{t}}{\tau} = F_{int}(\mathcal{C}^{t+1}) + F_{ext}(\mathcal{C}^{t+1})$$

assuming that F_{ext} is constant in one time step

$$\frac{\mathcal{C}^{t+1} - \mathcal{C}^{t}}{\tau} = F_{int}(\mathcal{C}^{t+1}) + F_{ext}(\mathcal{C}^{t})$$

Discrete Motion Equation

- In matrix form:

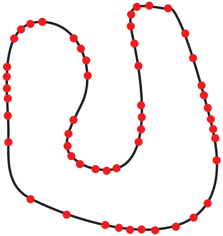
$$\frac{X^{t+1} - X^t}{\tau} = -AX^{t+1} + F(X^t)$$

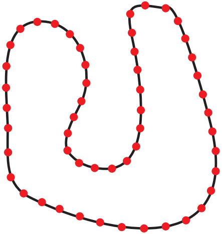
  i.e.,

$$(I + \tau A)X^{t+1} = X^t + \tau F(X^t)$$

  where I is the identity matrix.
- Thus, we obtain a linear system and we have to solve a *pentadiagonal banded symmetric positive system*.
- $(I + \tau A)^{-1}$ can be computed using a LU decomposition only once if the α, β remain constant through time.
- We stop iterating when the difference between iterations is small enough.
- Reparametrization must be performed regularly!

Reparametrization, Example





Before reparametrization   After reparametrization

Contents

Instability Due to Image Forces

- [Cohen 1991]
- Let us examine the effect of the image force $F_{ext} = -\nabla P$. The direction of F_{ext} implies steepest descent in P, which is natural since we want to get a minimum of P. Equilibrium is achieved at points where P is a minimum in the direction normal to the curve.
- We see from $(I + \tau A)X^{t+1} = (X^t + \tau F(X^t))$ that the position at time $t + 1$, X^{t+1}, is obtained after moving X^t along vector $\tau F(X^t)$ and then solving the system.
- Therefore:

Time Discretization

- If $\tau F(X^t)$ is too large the point x^t can be moved too far across the desired minimum and never come back.
- This type of instability can be suppressed by manual tuning of the time step, or
- by normalizing the forces, taking $F_{ext} = -k\nabla P/|\nabla P|$, where τk is on the order of the pixel size. When a point of the curve is close to an edge point, it is attracted to the edge and stabilizes there if there is no conflict with the smoothing process.

Space Discretization

- The force F_{ext} is known only on a discrete grid corresponding to the image.
- We can use bilinear interpolation of F_{ext} at non-integer positions.

Balloons/Pressure Forces. Motivation

- If the curve is not close enough to an edge, it is not attracted by it.
- If the curve is not submitted by any forces, it shrinks on itself.
- Often, due to noise, some isolated points are gradient maxima and can stop the curve when it passes by.

Balloons/Pressure Forces

- To balance this we can add another force. We consider our curve as a "balloon" (in 2D) that we inflate. The external force F becomes

$$F = k_1 \mathbf{n}(s) - k_2 \frac{\nabla P}{|\nabla P|},$$

  where $\mathbf{n}(s)$ is the normal unitary vector to the curve at point $\mathcal{C}(s)$ and k_1 is the amplitude of this force.
- If we change the sign of k_1 or the orientation of the curve, it will have an effect of *deflation* instead of inflation.
- k_1 and k_2 are chosen such that they are of the same order, which is smaller than a pixel size, and k_2 is slightly larger then k_1 so that an edge point can stop the inflation force.

Elasticity and Stiffness Coefficients

- The coefficients of elasticity and stiffness have great importance for the behavior of the curve along time iterations.
- Elasticity is **the ability of hard materials to return to the initial state** in case of elastic deformation
- Stiffness is **the ability of hard materials to resist the applied force** in case of elastic deformation

- If α and β are around unity, the internal energy has a major influence and the image forces have small effect. In this case the curve is only regularized.
- We obtain good results when the parameters are of the order of h^2 for α and h^4 for β, where h is the space discretization step.

Snake Parameters Summary

- The snake evolution equation:

$$\frac{\partial \mathcal{C}}{\partial t} = \alpha \mathcal{C}''(s) - \beta \mathcal{C}''''(s) - k_1 \mathbf{n}(s) - k_2 \frac{\nabla P(\mathcal{C}(s))}{|\nabla P(\mathcal{C}(s))|}$$

  where

$$P(\mathcal{C}(s)) = -w_{line}(G_\sigma * f(\mathcal{C}(s))) + w_{edge}|\nabla(G_\sigma * f(\mathcal{C}(s)))|^2$$

- In matrix form:

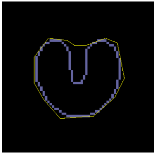
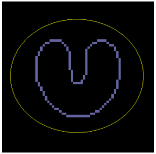
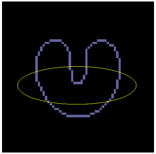
$$(I + \tau A)X^{t+1} = X^t + \tau F(X^t)$$

- Parameter choice (better but not obligatory):
  - α is of the order of h^2 and β is of the order of h^4
  - k_1 sign controls inflate or deflate
  - $|k_1| < |k_2| < 1$
  - τ controls the snake speed

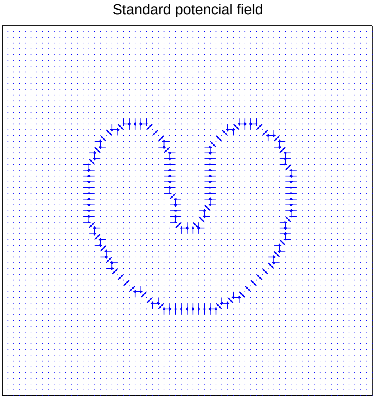
Contents

GVF Snakes: Motivation

- [Xu and Prince 1997,1998]

	Input		Output
Snakes		→	
GVF Snakes		→	
GVF Snakes		→	

Traditional External Forces



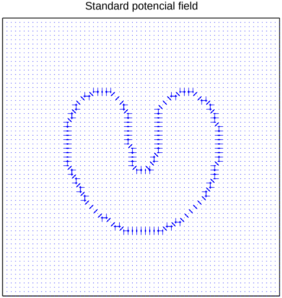
Traditional Potential Field

- Standard potential field for the image $f(x, y)$ looks like:

$$P(x, y) = -w_{line}(G_\sigma * f(x, y)) + w_{edge}|\nabla(G_\sigma * f(x, y))|^2$$

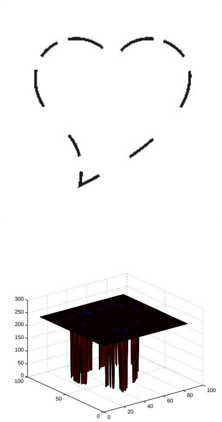
- Properties of external forces $F_{ext}(x, y) = \nabla P(x, y)$
  - ∇P points toward the edges and normal to the edges.
  - ∇P generally has large magnitudes only in the immediate vicinity of the edges.
  - In homogenous regions, where $I(x, y)$ is nearly constant, ∇P is nearly zero.

Traditional External Forces

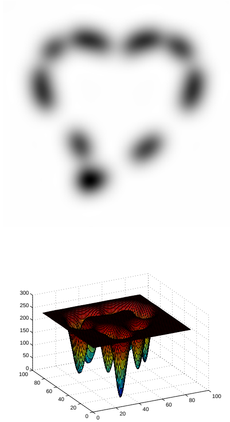


Smoothed External Forces

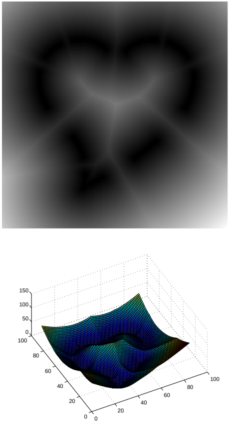
Smoothing is a good idea, but not perfect:



original edges



Gaussian ($\sigma = 10$)



Distance function

GVF Snakes

- The main idea is to compute a new static external force field $F_{ext} = \mathbf{g}(x, y)$, so called Gradient Vector Flow (GVF) field.
- Corresponding dynamic snake equation is

$$\mathcal{C}_t(s,t) = -\alpha\mathcal{C}''(s,t) + \beta\mathcal{C}''''(s,t) - \mathbf{g}$$

- It is solved numerically by discretization and iteration, in identical fashion to the traditional snake.

Gradient Vector Flow

- The gradient vector flow field is the field $\mathbf{g}(x, y) = [g_x(x, y), g_y(x, y)]$ that minimizes the energy functional

$$\varepsilon = \int\int |\nabla P|^2 |\mathbf{g} - \nabla P|^2 + \mu \left(\frac{\partial g_x}{\partial x}^2 + \frac{\partial g_x}{\partial y}^2 + \frac{\partial g_y}{\partial x}^2 + \frac{\partial g_y}{\partial y}^2 \right) dxdy$$

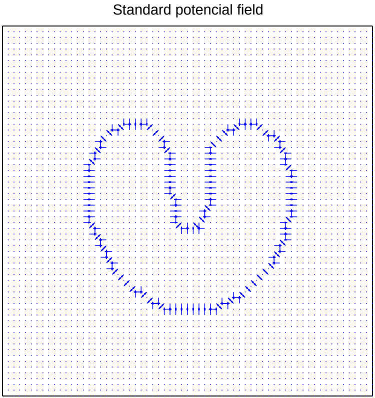
- The parameter μ is a regularization parameter controling the smoothness of the solution (more noise, increase μ)
- GVF field can be found by solving the following Euler equations

$$\mu \nabla^2 g_x - \left(g_x - \frac{\partial P}{\partial x} \right) \left(\frac{\partial P}{\partial x}^2 + \frac{\partial P}{\partial y}^2 \right) = 0$$

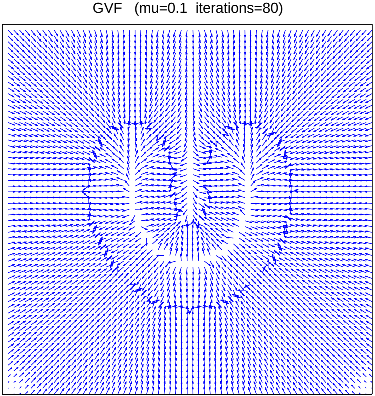
$$\mu \nabla^2 g_y - \left(g_y - \frac{\partial P}{\partial y} \right) \left(\frac{\partial P}{\partial x}^2 + \frac{\partial P}{\partial y}^2 \right) = 0$$

- These equations are solved numerically.

Traditional External Forces

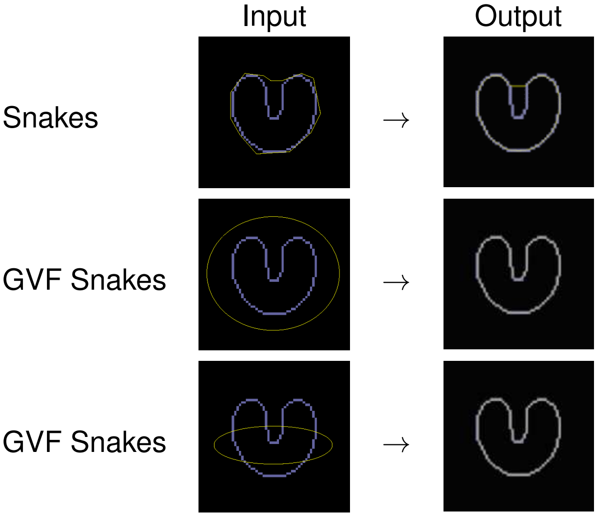


Gradient Vector Flow (Example)



GVF Snakes: Motivation

- [Xu and Prince 1997,1998]



Contents

Deformable Surface

- Snakes as well as GVF can be generalized to 3D space.
- However, the generalization is not straightforward nor easy.
- Implementation of snakes must be completely rewritten.
- The next two slides are only for impression.

Deformable Surface, Definition

- The deformable surface model is a mapping:

$$\mathbf{v}(s,r) : \Omega = [0,1] \times [0,1] \to \mathbb{R}^3$$

$$(s,r) \mapsto \mathbf{v}(s,r) = (x(s,r), y(s,r), z(s,r))$$

- The energy functional is defined

$$E(\mathbf{v}) = \int_\Omega E_{int}(\mathbf{v}(s,r)) + E_{ext}(\mathbf{v}(s,r)) ds dr,$$

where $E_{int}(\mathbf{v}(s,r))$ is internal energy defined as

$$E_{int}(\mathbf{v}) = w_{10} \left| \frac{\partial \mathbf{v}}{\partial s} \right|^2 + w_{01} \left| \frac{\partial \mathbf{v}}{\partial r} \right|^2$$

$$+2w_{11} \left| \frac{\partial^2 \mathbf{v}}{\partial s \partial r} \right|^2 + w_{20} \left| \frac{\partial^2 \mathbf{v}}{\partial s^2} \right|^2 + w_{02} \left| \frac{\partial^2 \mathbf{v}}{\partial r^2} \right|^2 ds\, dr,$$

and $E_{ext}(\mathbf{v}(s,r)) = P(s,r)$ is external (potential) energy.

Motion Equation

- Euler equation (local minima condition)

$$-\frac{\partial}{\partial s}\left(w_{10}\frac{\partial \mathbf{v}}{\partial s}\right)-\frac{\partial}{\partial r}\left(w_{01}\frac{\partial \mathbf{v}}{\partial r}\right)+2\frac{\partial^2}{\partial s\partial r}\left(w_{11}\frac{\partial^2 \mathbf{v}}{\partial s\partial r}\right)$$

$$+\frac{\partial^2}{\partial s^2}\left(w_{20}\frac{\partial^2 \mathbf{v}}{\partial s^2}\right)+\frac{\partial^2}{\partial r^2}\left(w_{02}\frac{\partial^2 \mathbf{v}}{\partial r^2}\right)+\nabla P(\mathbf{v}(s,r))=0$$

- Associated motion equation (snake analogy)

$$\gamma\frac{\partial \mathbf{v}}{\partial t}-\frac{\partial}{\partial s}\left(w_{10}\frac{\partial \mathbf{v}}{\partial s}\right)-\frac{\partial}{\partial r}\left(w_{01}\frac{\partial \mathbf{v}}{\partial r}\right)+2\frac{\partial^2}{\partial s\partial r}\left(w_{11}\frac{\partial^2 \mathbf{v}}{\partial s\partial r}\right)$$

$$+\frac{\partial^2}{\partial s^2}\left(w_{20}\frac{\partial^2 \mathbf{v}}{\partial s^2}\right)+\frac{\partial^2}{\partial r^2}\left(w_{02}\frac{\partial^2 \mathbf{v}}{\partial r^2}\right)+\nabla P(\mathbf{v}(s,r))=0,$$

Numerical Solution

- Numerical solution using finite-difference method is time consuming.
- Different method, e.g. finite element method (FEM), needs to be applied.
- GVF field has to be computed in 3D in advance.

Contents

Summary

- Snake is parametric curve, which changes its shape under the influence of internal and external forces (minimizes own energy)
- Initial model must be close to the expected result
  - Remedy: balloon force, gradient vector flow
- External forces must be appropriately defined to detect objects
- Relatively fast computation
- Preserves topology of contour, but the contour may cross
- Topology changes are problematic
- Can be generalized to 3D (however, generalization is not straightforward)
- Snake can be represented by B-splines (B-snakes).

References

- M. Kass, A. Witkin, and D. Terzopoulos, Snakes: Active Contour Models, Int. J Computer Vision, vol. 1. no. 4., 1988.
- L. D. Cohen, On Active Contour Models and Balloons, Computer Vision, Graphics, and Image Processing: Image Understanding, vol. 53, no. 2, 1991
- L. D. Cohen, and I. Cohen, Finite-Element Methods for Active Contour Models and Balloons for 2-D and 3-D Images, IEEE T. PAMI, Vol. 15, No. 11, 1993
- C. Xu and J. L. Prince, Gradient Vector Flow: A new External Force for Snakes, CVPR'97
- C. Xu and J. L. Prince, Snakes, Shapes, and Gradient Vector Flow, IEEE TIP, Vol. 7, No. 3, 1998
- `http://iacl.ece.jhu.edu/projects/gvf/`