UNIT-III

Active Contours

-> Active Contours and active surfaces are means of model-driven segmentation. Their use enforces closed and smooth boundaries for each segmentation irrespective of the image content

-> Data driven approaches: Objects in an image appear homogeneous

-> Model-driven approaches: Ideal object boundary are predicted.

Frameworks for Snakes

-> A higher level process or a user initializes any curve close to object boundary.

-> The snake then starts deforming and moving towards the desired object boundary.

Deformable Models

-> Deformable models are curves or surfaces defined within an image domain that can move under the influence of internal forces, which are defined within the curve or surface itself, and external forces, which are computed from the image data.

Active Contour Modeling

→ The contour is defined in (x, y) plane of an image as a parametric curve.

$$v(s) = (x(s), y(s)) \quad 0 \leq s \leq 1$$

→ Contour is said to possess an energy which is defined as the sum of three energy terms

$$E_{snake} = E_{internal} + E_{external} + E_{constraint}$$

The energy terms are defined cleverly in a way such that the final position of the contour will have a minimum energy (E_{min})

The energy function of the snake is the sum of its external energy and internal energy or

$$E^{*}_{snak} = \int_0^1 E_{snake}(V(s))ds = \int_0^1 (E_{internal}(V(s)) + E_{image}(V(s)) + E_{con}(V(s)))ds$$

Internal energy

The internal energy of the snake is composed of the continuity of the contour E_{cont} and the smoothness of the contour E_{cur}.

$$E_{internal} = E_{cont} + E_{curv}$$

This can be expanded as

$$E_{internal} = \frac{1}{2}\left(\alpha(s)|V_s(s)|^2\right) + \frac{1}{2}\left(\beta(s)|V_{ss}(s)|^2\right)$$

$$= \frac{1}{2}\left(\alpha(s)\left\|\frac{d\bar{v}}{ds}(s)\right\|^2 + \beta(s)\left\|\frac{d^2v}{ds^2}(s)\right\|^2\right)$$

→ The smoothness energy at contour point $V(s)$ could be evaluated as

$$E_{in}(V(s)) = \alpha(s)\left|\frac{dv}{ds}\right|^2 + \beta(s)\left|\frac{d^2v}{d^2s}\right|^2$$

External energy

External energy of a contour point $v(x,y)$ could be

$$E_{ex}(v) = -|\nabla I(v)|_2 = -|\nabla I(x,y)|_2$$

External energy term for the whole snake is

$$E_{ex} = \int_0^1 E_{ex}(v(s))\,ds$$

$$E_{ex} = \sum_{i=0}^{n-1} E_{ex}(v_i)$$

Basic Elastic Snake

The total energy of a basic elastic snake is

$$\rightarrow E = \alpha \cdot \int_0^1 \left| \frac{dv}{ds} \right|^2 ds - \int_0^1 \left| \nabla I(v(s)) \right|^2 ds$$

$$E = \alpha \cdot \sum_{i=0}^{n-1} \left| v_{i+1} - v_i \right|^2 - \sum_{i=0}^{n-1} \left| \nabla I(v_i) \right|^2$$

Gradient Descent

Ex: minimization of function of 2 variables

$$= - \sum_{i=0}^{n-1} \left| I_x(x_i, y_i) \right|^2 + \left| I_y(x_i, y_i) \right|^2$$

$$+ \alpha \cdot \sum_{i=0}^{n-1} (x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2$$

Problem with Snakes

* depends on number and spacing of control points
* Snake may over-smooth the boundary
* Initilization is crucial.

Level Sets

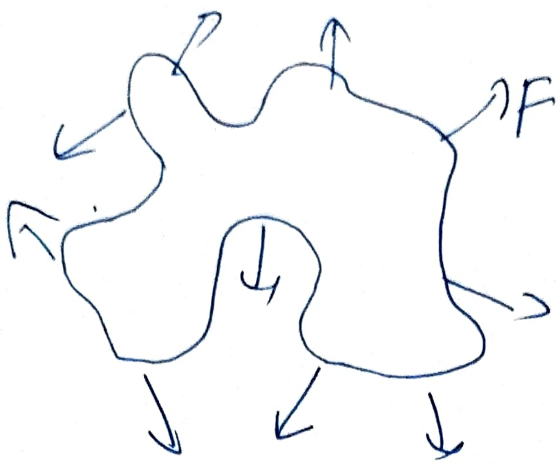
→ A limitation of active contours based on parametric curves of the form (snakes, b-snakes ...) is that it is challenging to change the topology of the curve as it evolves.

* An alternative representation for such closed contours is to use level sets (LS).

→ LS evolve to fit and track objects of interest by modifying the underlying embedding function instead of curve function(s).

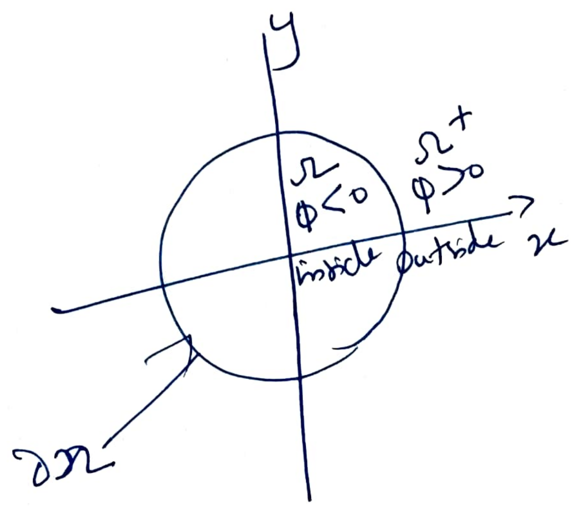
Evolving Curves & Surfaces

* Propogate curve according to speed function $V = Fn$
* F depends on space, time & curve itself.
* Surfaces in three dimensions



Describe curve as Level sets of function.

$\phi(x,y) = x^2 + y^2 - 1 = 0$



$\phi = x^2 + y^2 - 1 = 0$

Split & Merge

For any region R, its internal difference is defined as the largest edge weight in the region's minimum spanning tree

$$Dif(R1, R2) = \min_{e=(v1,v2)|v_1 \in R, v_2 \in R_2} w(e)$$

Their algorithm merges any two adjacent regions whose difference is smaller than the minimum internal difference of these two regions,

$$M\,Int(R_1, R_2) = \min(Int(R_1) + T(R_1), Int(R_2) + T(R_2)),$$

$$\sigma^+_{local} = \min(\nabla_i^+, \Delta_j^+),$$

$$\sigma^-_{local} = \frac{\Delta_i^- + \Delta_j^-}{2}$$

where $\Delta_i^- = \Sigma_k (T_{ik} \Delta_{ik}) / \Sigma_k (T_{ik})$ and

T_{ik} is the boundary length between regions

R_i and $\frac{\Sigma_j \in C P_{ij}}{\Sigma_j \in V P_{ij}} > \varphi,$