Lecture 16-Snake Contour

Yuyao Zhang, Xiran Cai PhD
zhangyy8@shanghaitech.edu.cn caixr@shanghaitech.edu.cn
SIST Building 2 302-F/302-C
Course piazza link：
piazza.com/shanghaitech.edu.cn/spring2021/cs270spring2021

上海科技大学
ShanghaiTech University

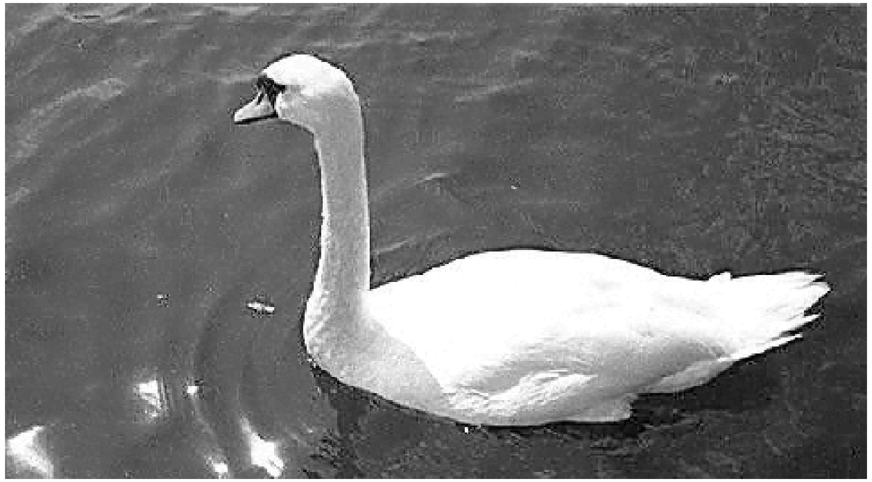
Active Contours (SNAKES)

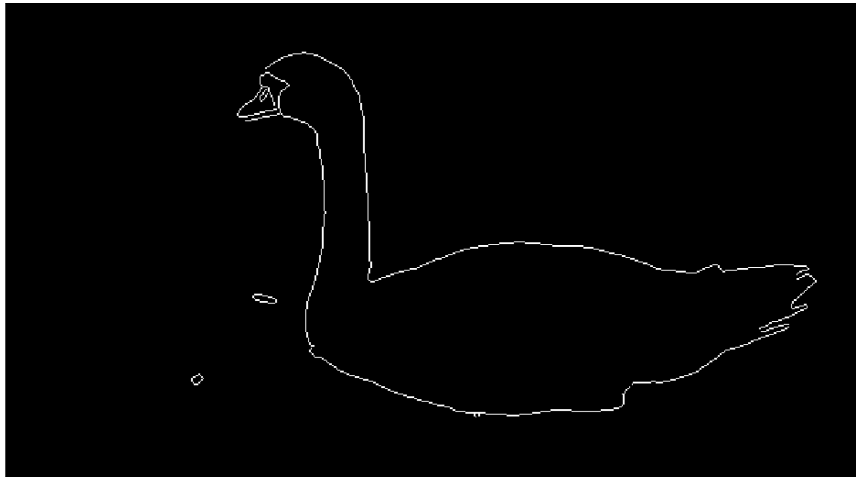
- Back to boundary detection
  - This time using perceptual grouping.
- This is non-parametric
  - We’re not looking for a contour of a specific shape.
  - Just a good contour.

For Information on SNAKEs

- Kass, Witkin and Terzopoulos, IJCV.
- "Dynamic Programming for Detecting, Tracking, and Matching Deformable Contours", by Geiger, Gupta, Costa, and Vlontzos, IEEE Trans. PAMI 17(3)294-302, 1995
- E. N. Mortensen and W. A. Barrett, Intelligent Scissors for Image Composition, in ACM Computer Graphics (SIGGRAPH `95), pp. 191-198, 1995

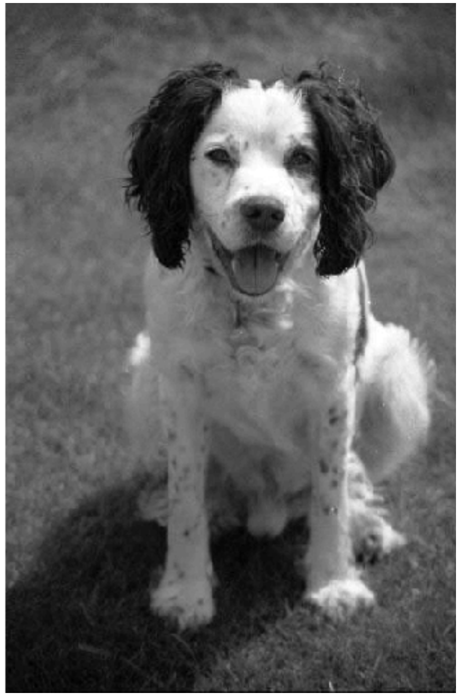
Boundary following





Sometimes edge detectors find the boundary pretty well.

Sometimes it's not good enough.



Improve Boundary Detection

➢ **Idea: segment using curves, not pixels.**

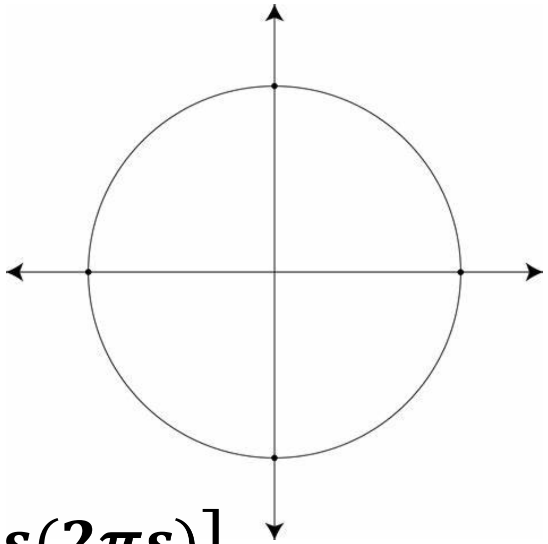
➢ **We want a segmentation curve that**

- **1) Conforms to image edges.**
- **2) Generates a smooth and varying curve.**

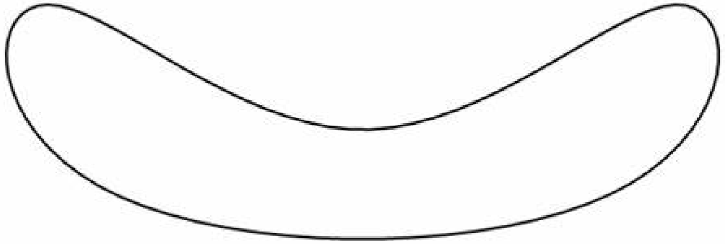
Parametric Curves

- Consider $\begin{bmatrix} x(s) \\ y(s) \end{bmatrix}$ $s \in [0, 1]$ continuous.

- E.g.



$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} r\cos(2\pi s) \\ r\sin(2\pi s) \end{bmatrix}$$

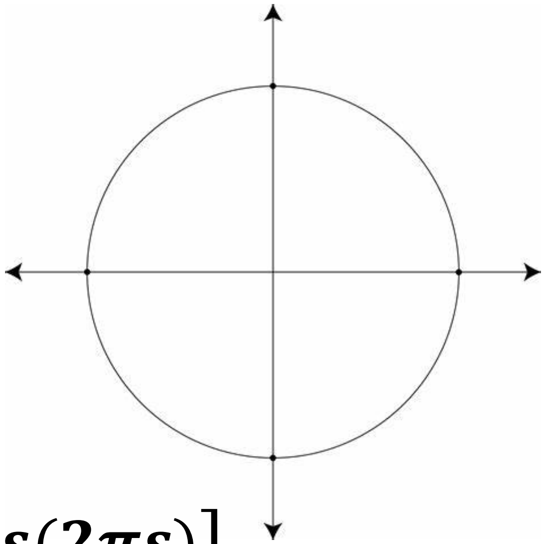


??

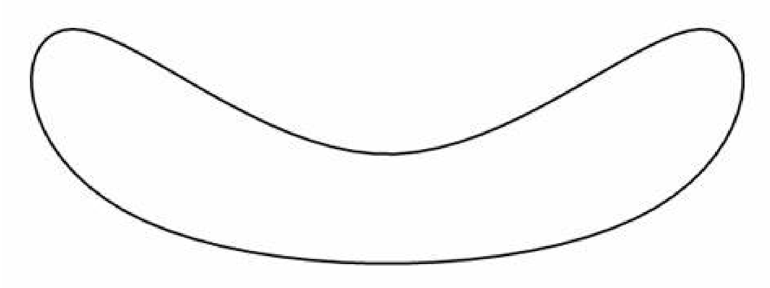
Parametric Curves

- Consider $\begin{bmatrix} x(s) \\ y(s) \end{bmatrix}$ $s \in [0, 1]$ continuous.

- E.g.



$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} r\cos(2\pi s) \\ r\sin(2\pi s) \end{bmatrix}$$

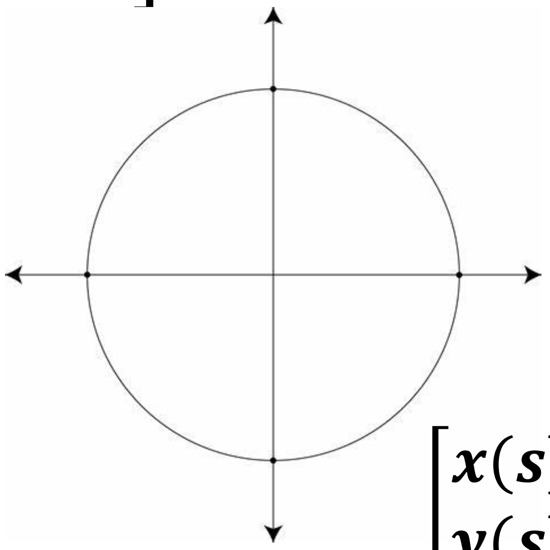


??

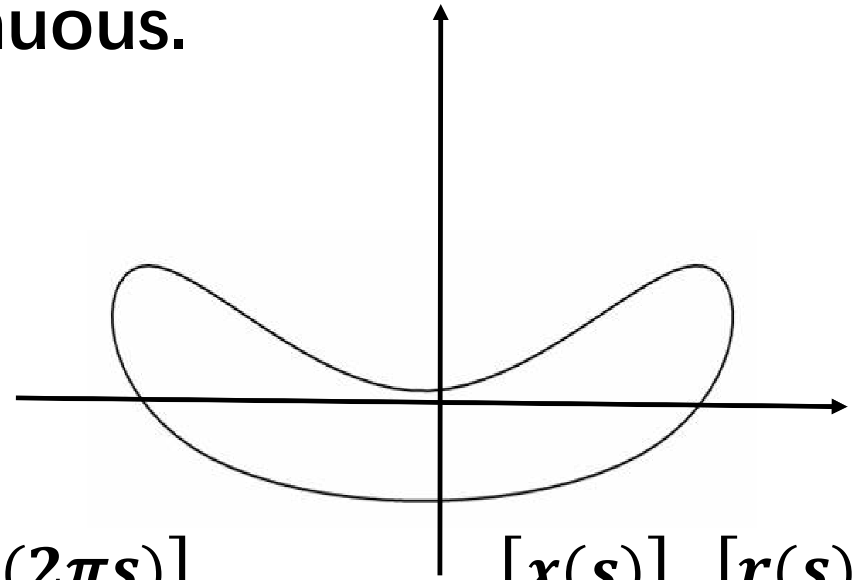
Parametric Curves

- Consider $\begin{bmatrix} x(s) \\ y(s) \end{bmatrix}$ $s \in [0,1]$ continuous.

- E.g.



$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} rcos(2\pi s) \\ rsin(2\pi s) \end{bmatrix}$$

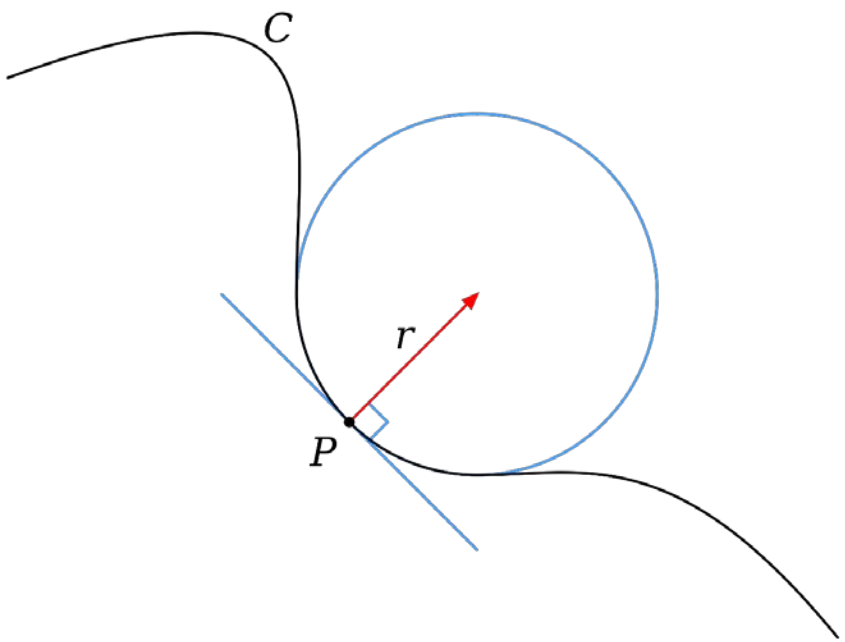


$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} r(s)cos(2\pi s) \\ r(s)sin(2\pi s) \end{bmatrix}$$

C

- We define a curve using $C(s) = [x(s), y(s)]$.

Curvature

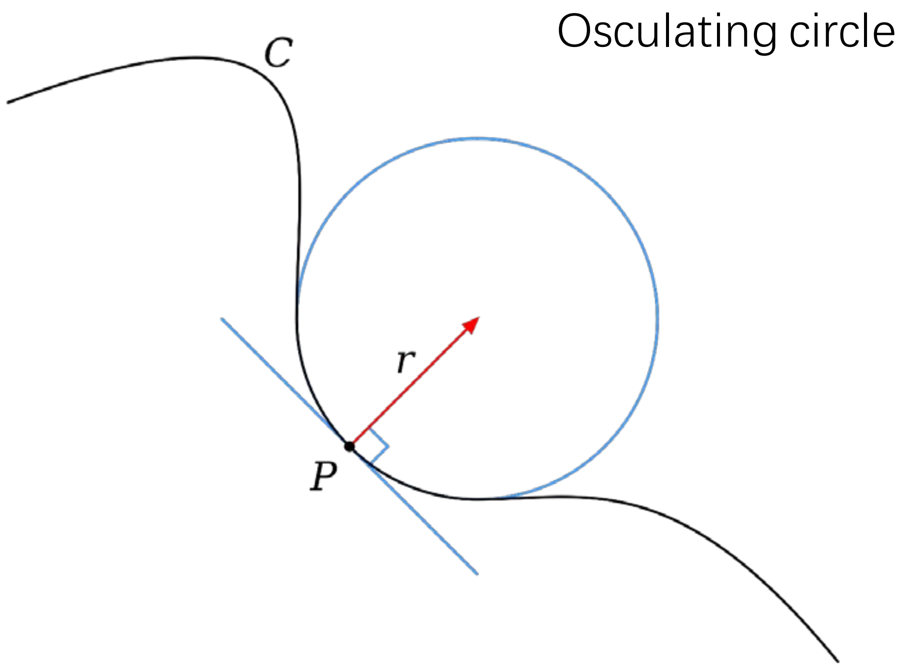


Curvature $K(s) = \frac{1}{R(s)}$

- Let C be a plane curve (the precise technical assumptions are given below). The curvature of C at a point is a measure of how sensitive its tangent line is to moving the point to other nearby points.

- It is natural to define the curvature of a straight line to be constantly zero. The curvature of a circle of radius r should be large if r is small and small if r is large. Thus the curvature of a circle is defined to be the reciprocal of the radius.

Curvature



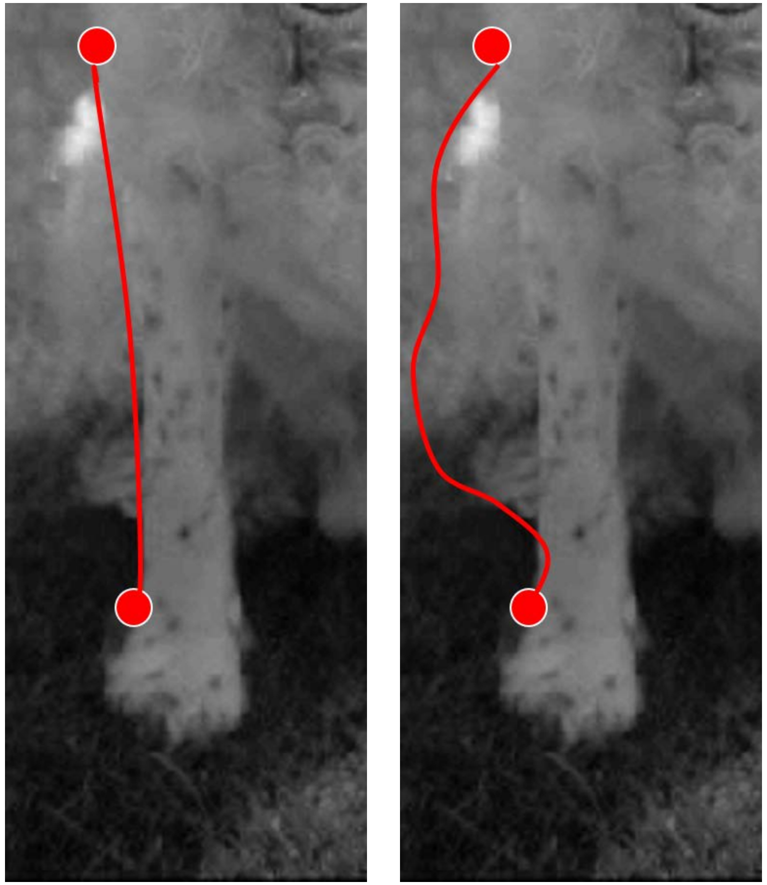
Curvature $K(s) = \frac{1}{R(s)}$

- Let C be a plane curve (the precise technical assumptions are given below). The curvature of C at a point is a measure of how sensitive its tangent line is to moving the point to other nearby points.
- It is natural to define the curvature of a straight line to be constantly zero. The curvature of a circle of radius r should be large if r is small and small if r is large. Thus the curvature of a circle is defined to be the reciprocal of the radius.

上海科技大学 ShanghaiTech University

Curvature of plane curves

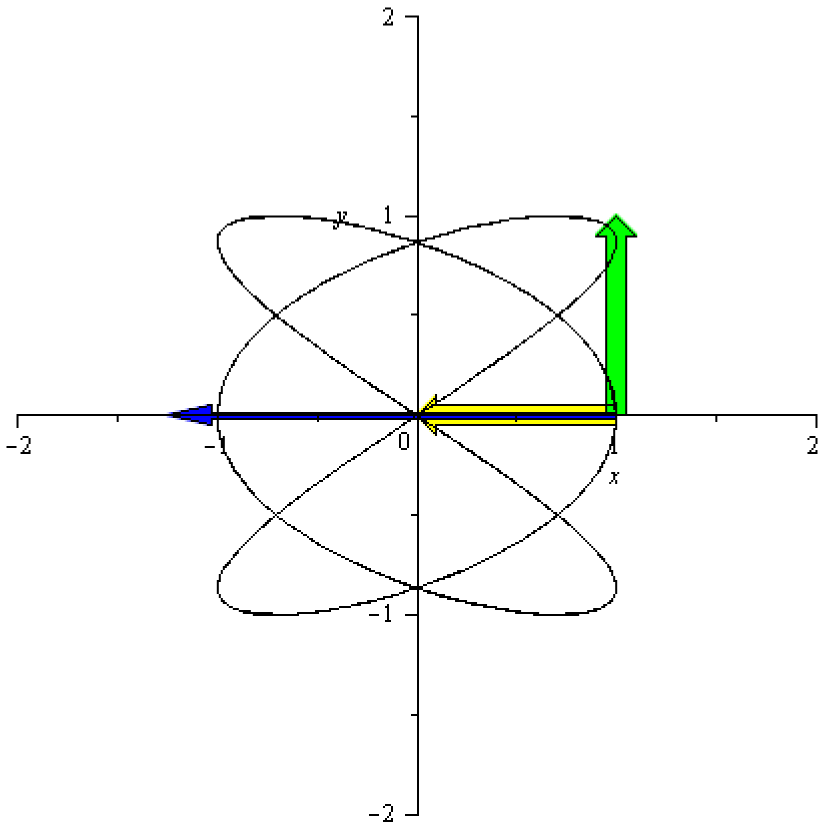
- How do we decide how good a path is? Which of two paths is better?

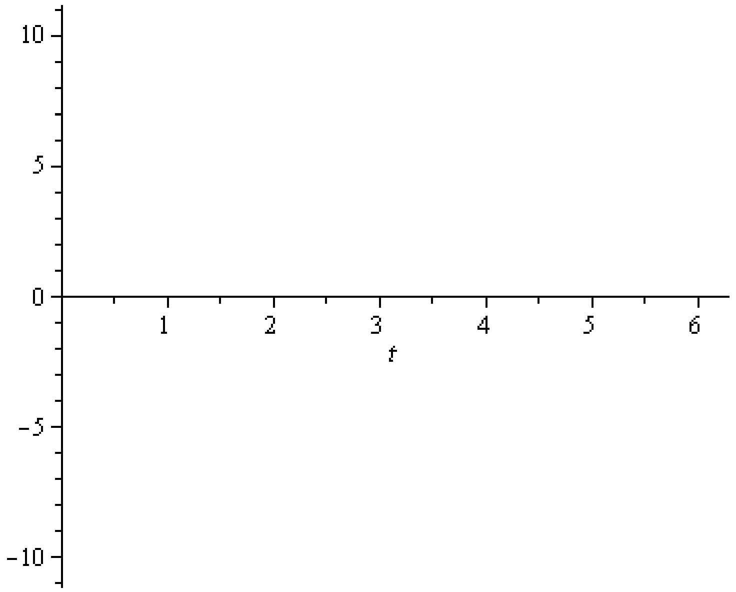


- $T(s) = C'(s)$ is considered as the velocity vector or the unit tangent vector of the curve $C(s)$.
- $\kappa(s) = \frac{1}{R(s)} = C''(s)$ is the curvature of curve $C(s)$.

Curvature of plane curves

Lissajous-Curve with tangent vector (green), normal vector (yellow), and "acceleration vector" (blue)





Find energy for the curve

- Idea: we want to define an energy function $\boldsymbol{E(c)}$ that matches our intuition about what makes a good segmentation.

- Curve will iteratively evolve to reduce/ minimize $\boldsymbol{E(c)}$.

- $\boldsymbol{E(c) = E_{internal}(c) + E_{external}(c)}$.

  - ✓ $\boldsymbol{E_{internal}(c)}$ depends only on the shape of the curve.

  - ✓ $\boldsymbol{E_{external}(c)}$ depends on image intensities.

Energy for shape of the curve

- $\boldsymbol{E_{internal}(c) = \int_0^1 \alpha\|c'(s)\|^2 + \beta\|c''(s)\|^2\, ds}$.
  - ✓ $\boldsymbol{Low\ c'(s)}$ keeps curve not too “stretchy”.
  - ✓ $\boldsymbol{Low\ c''(s)}$ keeps curve not too “bendy”.

Energy for image intensities

- $E_{external}(c) = \int_0^1 -\|\nabla I(c(s))\|^2 ds$

$$= \int_0^1 -\{\left[\frac{\partial I}{\partial x}\big(x(s), y(s)\big)\right]^2 + \left[\frac{\partial I}{\partial y}\big(x(s), y(s)\big)\right]^2\}ds$$

- ✓ *No edge, then* $\nabla I = 0$, $E_{external}(c) = 0$
- ✓ *Big edge, then* $\nabla I =$ *big positive value*, $E_{external}(c) =$ *big negative value*

How to minimize E(c)?

- **Requires: variational calculus. (变分微积分)**

  1. In practice for digital images, we solve the problem by creating a curve $C(s,t)$. Where t represents the iteration.
  2. Curve approximated by k discrete points $(\boldsymbol{x_i}, \boldsymbol{y_i})$.
  3. Then we step $C(s,t-1)$ to $C(s,t)$ by taking a step along gradient of $\boldsymbol{E}(\boldsymbol{C})$: $\frac{\boldsymbol{\partial E}}{\boldsymbol{\partial C}}$.
  4. A snake minimize $\boldsymbol{E}(\boldsymbol{C})$ must satisfy the Euler equation: $-\boldsymbol{\nabla E_{external}}(\boldsymbol{c})$ = 0.

- **Result: Curve inches along until points around perimeter stop changing.**

Try this

- Launch “snake.m”.
- Load image “circle”. Click the button “Set new points”, initialize starting points.
- Click the button “start” to start.

Problem with basic snake (E_{ext})

- Contour never “sees” strong edges that are far away.
- Small gradient: Snake gets hung up.
- When there is no gradient for external Energy, then only internal Energy working.
- Can not work for outer boundary.

Gradient vector flow (GVF)

- **Idea: instead of using exactly the image gradient, create a new vector field over image plane:**
- $\vec{\boldsymbol{V}}(\boldsymbol{x},\boldsymbol{y}) = \begin{bmatrix} \boldsymbol{V}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y}) \\ \boldsymbol{V}_{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{y}) \end{bmatrix}$ **vector field of the curve.**
- $\vec{\boldsymbol{e}}(\boldsymbol{x},\boldsymbol{y}) = \begin{bmatrix} \boldsymbol{e}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y}) \\ \boldsymbol{e}_{\boldsymbol{y}}(\boldsymbol{x},\boldsymbol{y}) \end{bmatrix}$ **vector field of the edge map in an image.**
- $\vec{\boldsymbol{V}}(\boldsymbol{x},\boldsymbol{y})$ **is defined to minimize:在此处键入公式。**
- $\iint \mu[\left(\frac{\partial \boldsymbol{V}_{\boldsymbol{x}}}{\partial \boldsymbol{x}}\right)^2 + \left(\frac{\partial \boldsymbol{V}_{\boldsymbol{x}}}{\partial \boldsymbol{y}}\right)^2 + \left(\frac{\partial \boldsymbol{V}_{\boldsymbol{y}}}{\partial \boldsymbol{x}}\right)^2 + \left(\frac{\partial \boldsymbol{V}_{\boldsymbol{y}}}{\partial \boldsymbol{y}}\right)^2] + \ \|\nabla e\|^2 \left\|\vec{\boldsymbol{V}} - \vec{\boldsymbol{e}}\right\|^2 \ dxdy$
- **Intuition:** ∇e **is big: gradient is large,** $\vec{V}$ **follows edge gradient faithfully;** ∇e **is small: gradient is small, follows along to be as smooth as possible, trades off smooth vs how faithful.**

C. Xu and J.L. Prince, ``Gradient Vector Flow: A New External Force for Snakes,'' Proc. IEEE Conf. on Comp. Vis. Patt. Recog. (CVPR), Los Alamitos: Comp. Soc. Press, pp. 66-71, June 1997

Extensions

- Active shape models.
- Active appearance models.
- Level sets.
- FAST: FMRIB's Automated Segmentation Tool.

Discussion

- Try your own image with convolutional snake and GVF snake using the provided implementation or looking for better demon online.
- Find out what snake can do and what snake can not.