Lecture 19-Snake Contour

Yuyao Zhang PhD
zhangyy8@shanghaitech.edu.cn
SIST Building 2 302-F

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.

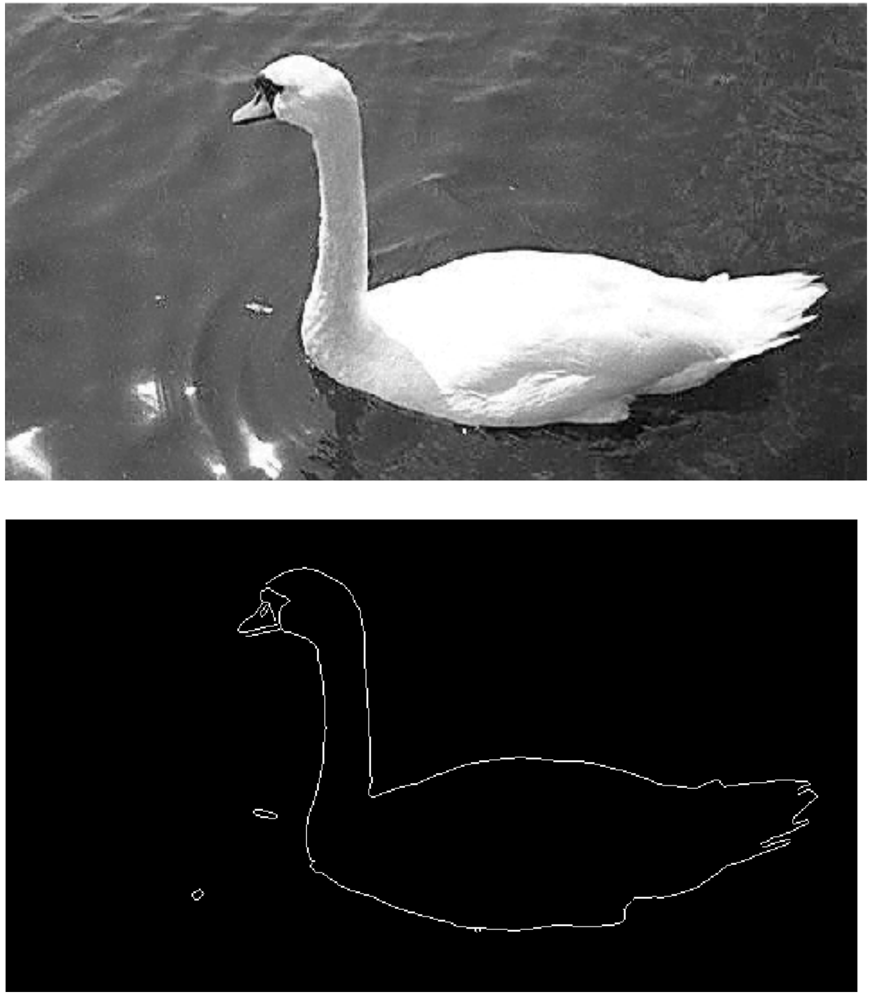
For Information on SNAKEs

- 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.

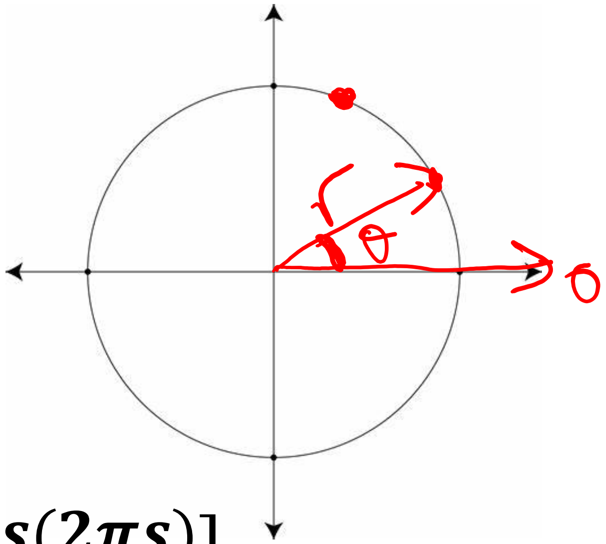
Parametric Curves

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

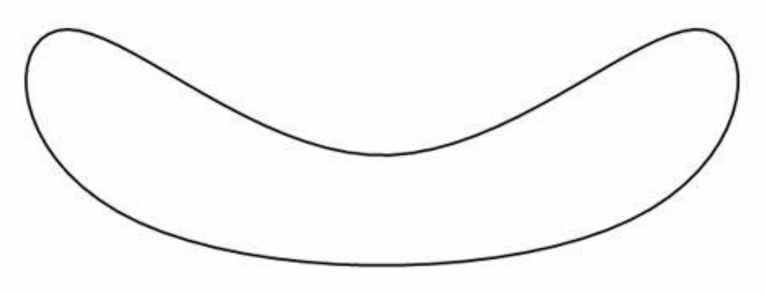
$x^2+y^2=r^2$

$\frac{x^2}{a}+\frac{y^2}{b}=r^2$

- E.g.



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

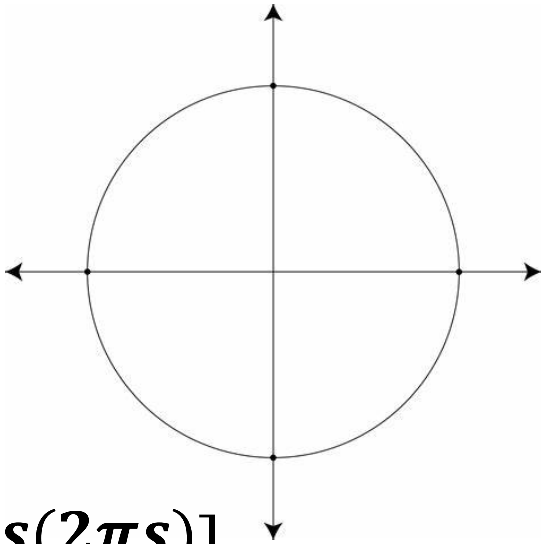


??

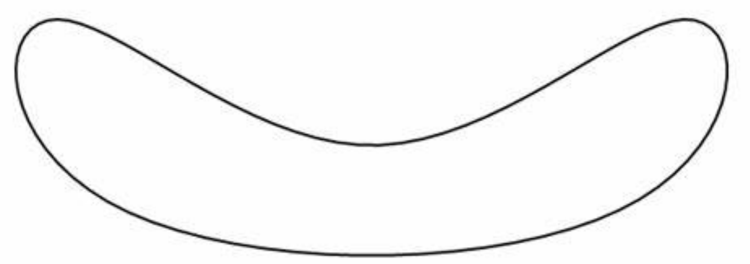
Parametric Curves

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

- E.g.



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

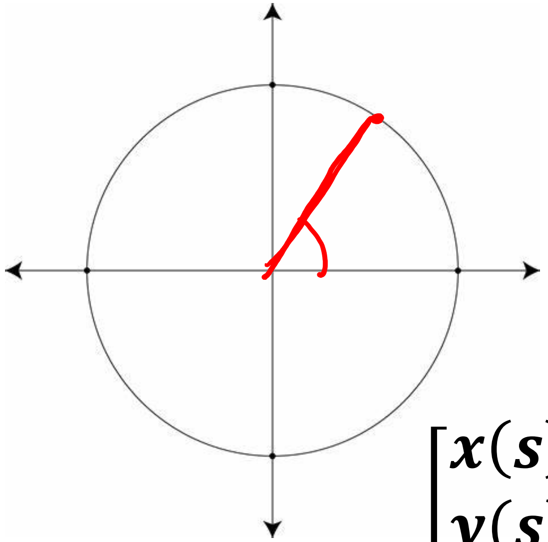


??

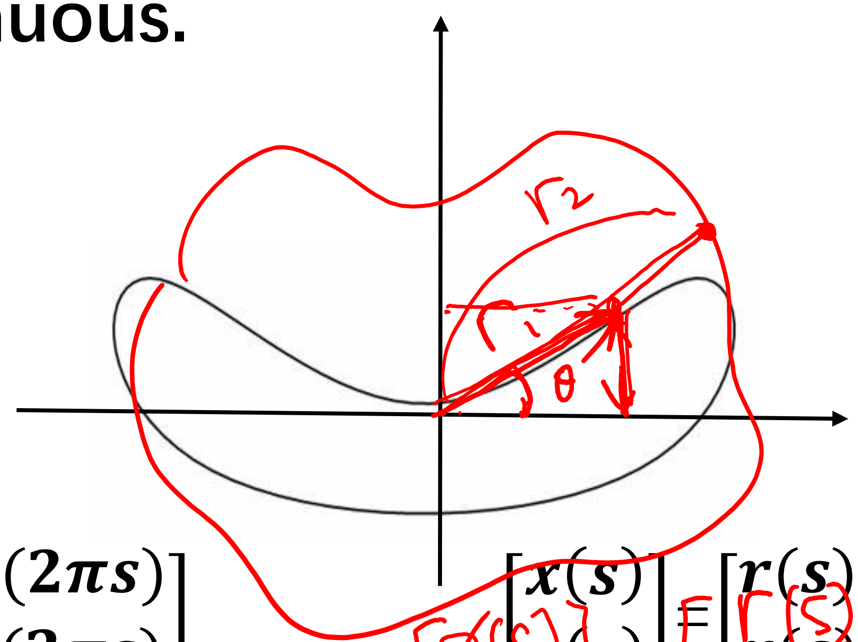
Parametric Curves

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

➢E.g.



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



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

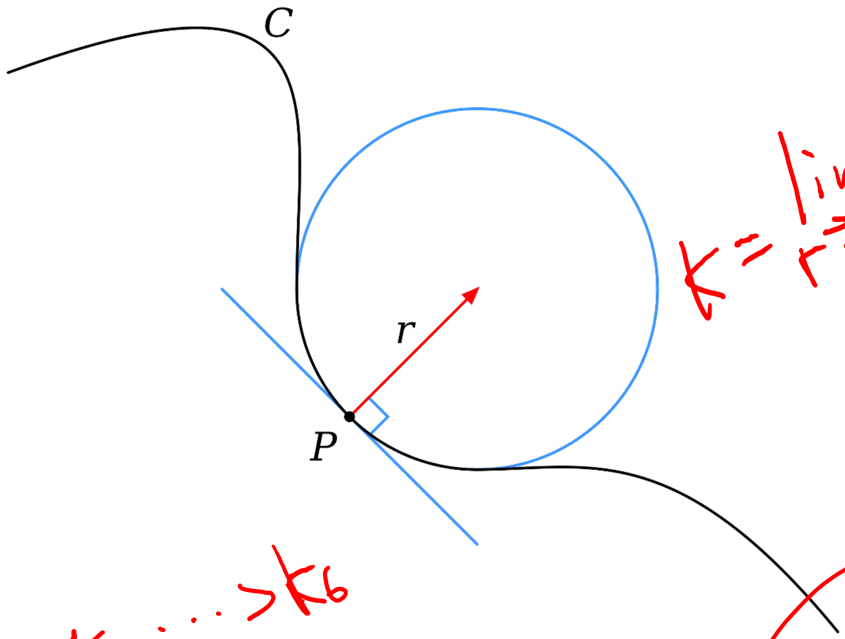
$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} r(s) \cos 2\pi s \\ r(s) \sin 2\pi s \end{bmatrix}$$

C

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

Curvature

$C(s)$



$K = \lim_{r \to \infty} \frac{1}{r} = 0$

- 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.

6

5

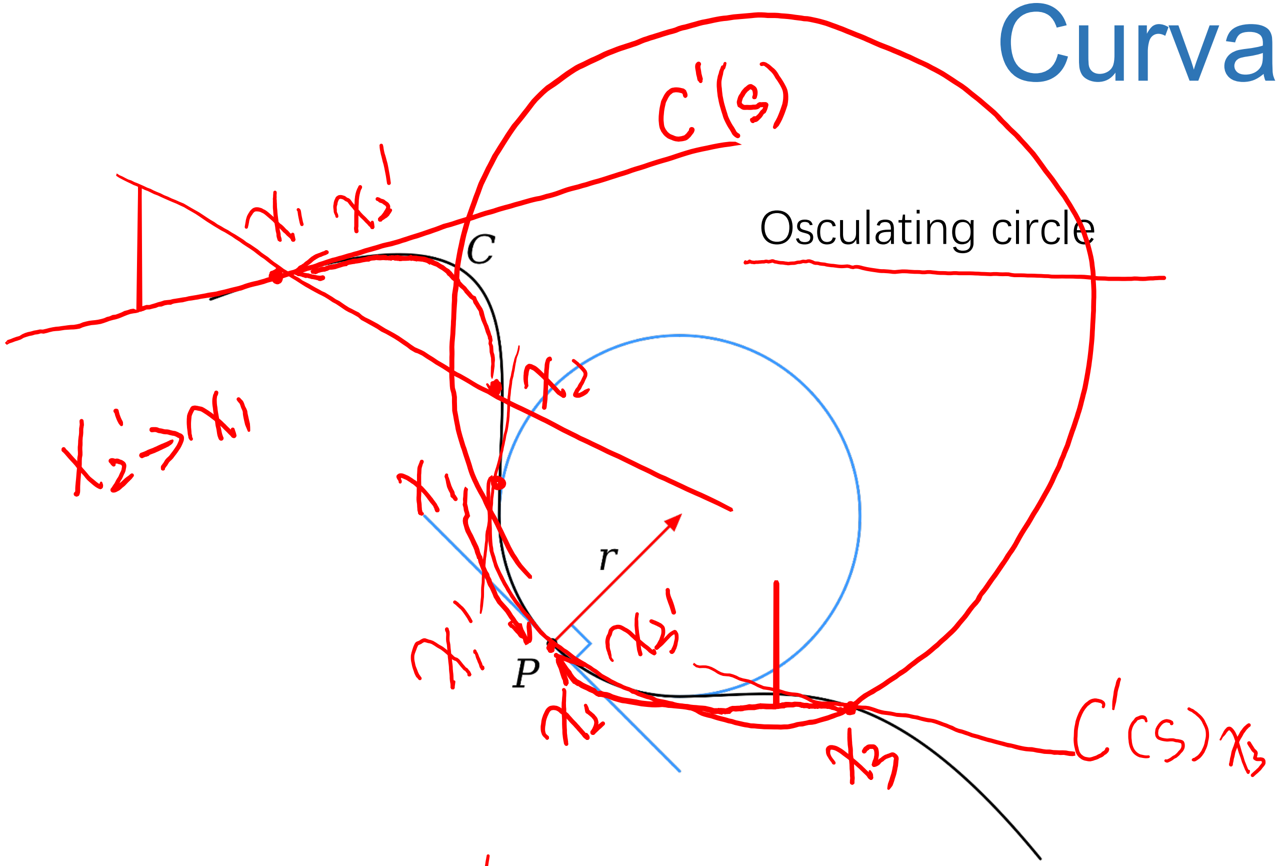
- It is natural to define the curvature of a straight line to be constantly zero. The curvature of a circle of radius r should be large if r is small and small if r is large. Thus the curvature of a circle is defined to be the reciprocal of the radius.

$K_1 > K_2 \cdots > K_6$

1 2 3 4

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

Curvature



$C'(s)_{x_3} \to C'(s)_{x_2}$

$C'(s)_{x_1} \to C'(s)_{x_2}$

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

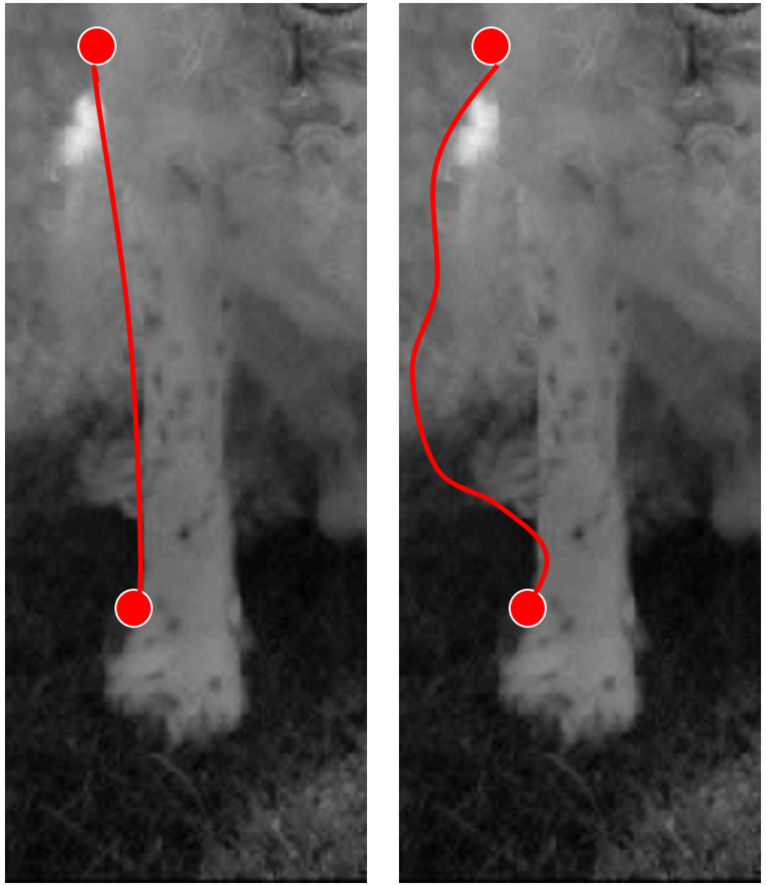
$C''(s)$

- 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 of plane curves

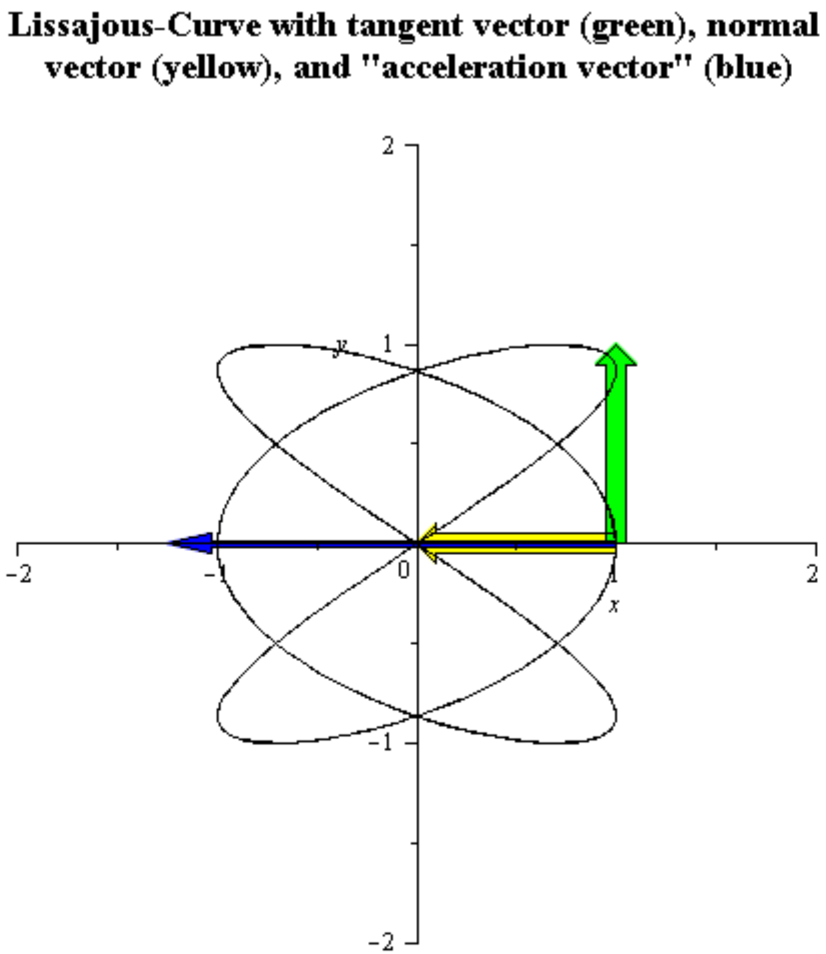
- How do we decide how good a path is? Which of two paths is better?

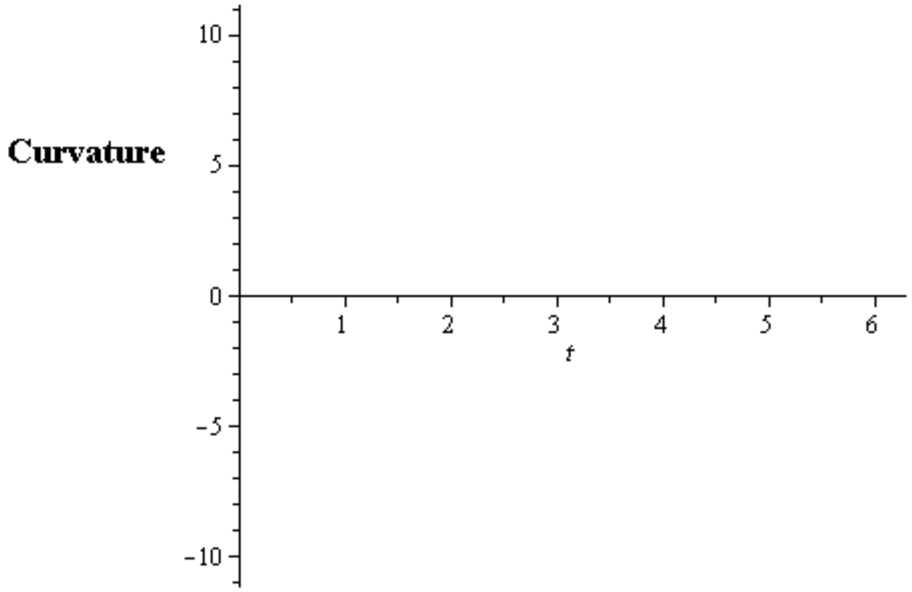


- $\boldsymbol{T(s) = C'(s)}$ is considered as the velocity vector or the unit tangent vector of the curve $\boldsymbol{C(s)}$.
- $\boldsymbol{\kappa(s) = \frac{1}{R(s)} = C''(s)}$ is the curvature of curve $\boldsymbol{C(s)}$.

上海科技大学 ShanghaiTech University

Curvature of plane curves





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.

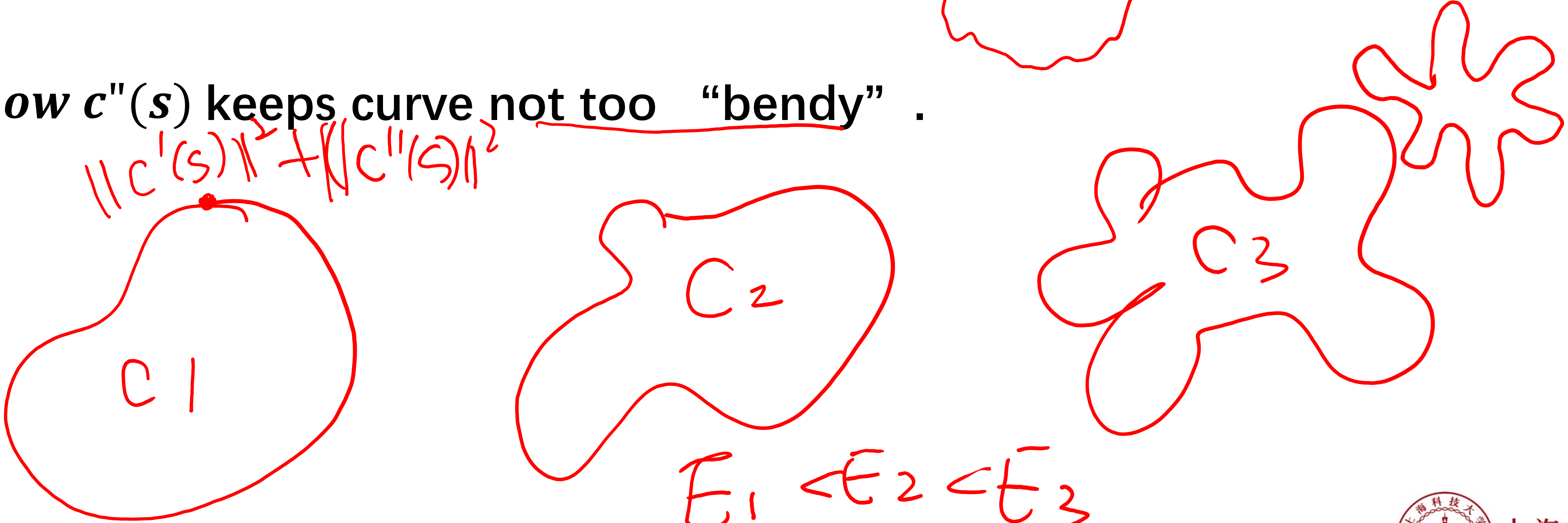
Energy for shape of the curve

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

$s \in [0, 1]$ $2\pi s$

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

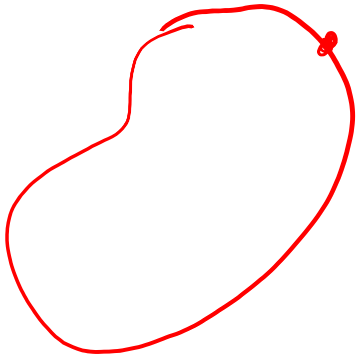
$\|c'(s)\|^2 + \|c''(s)\|^2$



$E_1 < E_2 < E_3$

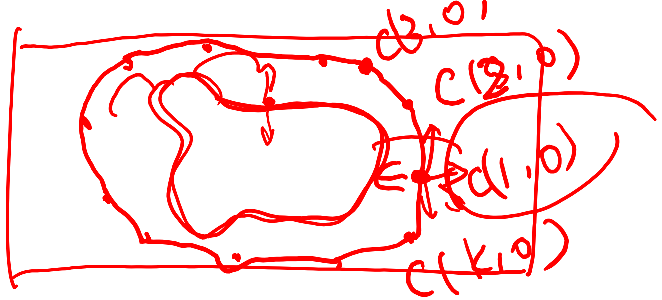
Energy for image intensities

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



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

- ✓ *No edge, then* $\nabla I = 0$, $E_{external}(c) = 0$
- ✓ *Big edge, then* ∇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 $\boldsymbol{E}(\boldsymbol{C})$: $\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{C}}$.
  4. A snake minimize $\boldsymbol{E}(\boldsymbol{C})$ must satisfy the Euler equation: $-\nabla \boldsymbol{E}_{\boldsymbol{external}}(\boldsymbol{C}) = 0$.

- **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:
- $\vec{V}(x,y) = \begin{bmatrix} V_x(x,y) \\ V_y(x,y) \end{bmatrix}$ vector field of the curve.
- $\vec{e}(x,y) = \begin{bmatrix} e_x(x,y) \\ e_y(x,y) \end{bmatrix}$ vector field of the edge map in an image.

G(x,y)

- $\vec{V}(x,y)$ is defined to minimize:在此处键入公式。

How similar $\vec{V}$ and ∇e

smoothness

$\nabla^2 \vec{V}$

min

- $\iint \mu\left[\left(\frac{\partial V_x}{\partial x}\right)^2 + \left(\frac{\partial V_x}{\partial y}\right)^2 + \left(\frac{\partial V_y}{\partial x}\right)^2 + \left(\frac{\partial V_y}{\partial y}\right)^2\right] + \|\nabla e\|^2 \|\vec{V} - \vec{e}\|^2 \; dxdy$

weight of edge map

- 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.