

# 第10章 形状分析

---

10.1 关于形状的讨论

10.2 平面形状的分类

10.3 形状特性的描述

10.4 基于技术的描述

10.5 拓扑结构的描述



# 10.1 关于形状的讨论

## 什么是形状?

- 许多人都知道，但没人能全面定义的概念
  - 读书辨字时，主要是形状信息在起作用
  - 但用语言来解释形状是比较难的
- 几个看起来简单，但很难回答的问题
  - (1) 什么是形状?
  - (2) 什么是客观世界中一个物体的形状?
  - (3) 什么是图象中一个区域的形状?



# 10.1 关于形状的讨论

## 什么是形状?

□ 字典中关于形状的几个定义:

- (1) 形状是由轮廓或外形所确定的外观
- (2) 形状是具有形体或图案的东西
- (3) 形状是实际物体或几何图案的一个性质，该性质依赖于组成该物体或图案的轮廓或表面的所有点间的相对位置



# 10.1 关于形状的讨论

## 什么是形状?

- 形状的定义：
  - 一个目标的形状就是该目标边界上所有点组成的模式
- 形状可定义为“连通的点集合”
- 一般考虑形状时，均考虑“单个”且“完整”的目标。“单个”和“完整”均可用连通的数学概念来描述



# 10.1 关于形状的讨论

## 形状研究的内容

### (1) 预处理

- 采集图象，存储图象，消除噪声，分割目标

### (2) 形状表达和描述

### (3) 形状分类

- 对给定形状的目标确定它是否属于某个预先定义类别（有监督分类）
- 对预先没有分类的形状如何定义或辨识其中的类别（无监督分类或聚类）



# 10.1 关于形状的讨论

## 形状分析的方法

- 描述常采用的三类方法
  - 特征的方法
  - 形状变换的方法
  - 基于关系的方法
- 描述符
  - 一个形状性质可用基于不同的理论技术的描述符来描述
  - 借助同一种理论技术也可以获得不同的描述符以刻画目标形状的不同性质

## 10.2 平面形状的分类

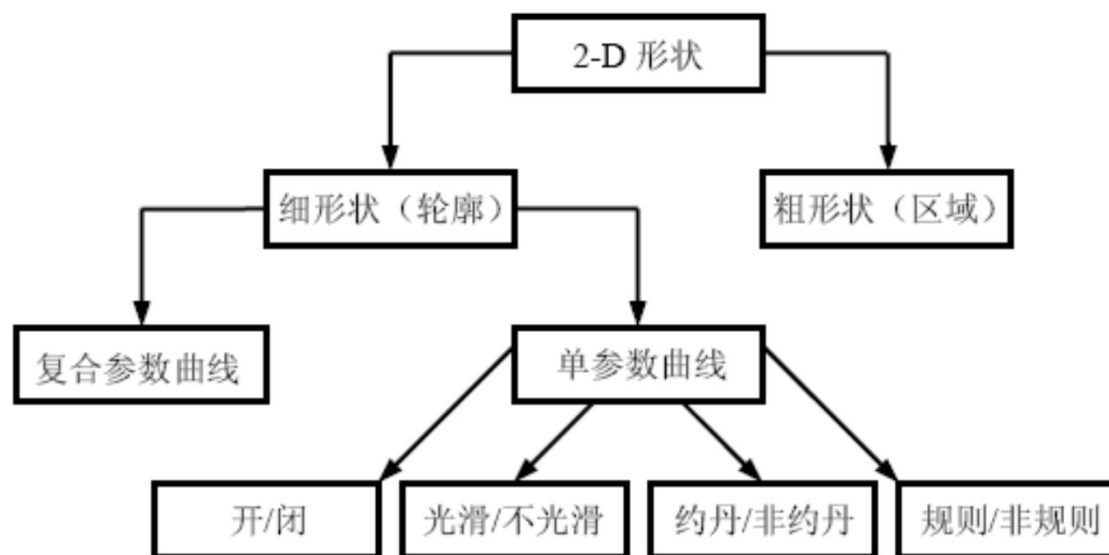


图 12.2.1 平面 (2-D) 形状的一个可能的分类图

光滑：各阶导数均存在

Jordan：无自交叉

复合参数：多个单参数

规则曲线：??



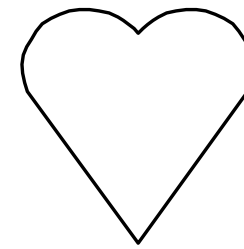
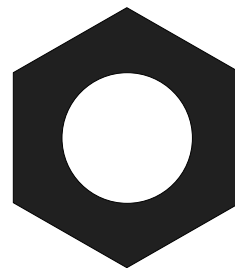
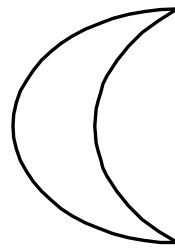
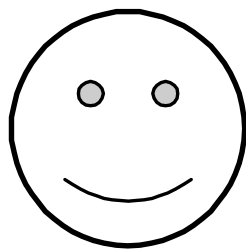
## 10.2 平面形状的分类

### (1) 粗和细形状

粗形状指包括内部的区域

细形状指没有充满的区域

2-D目标的外形 (silhouette)





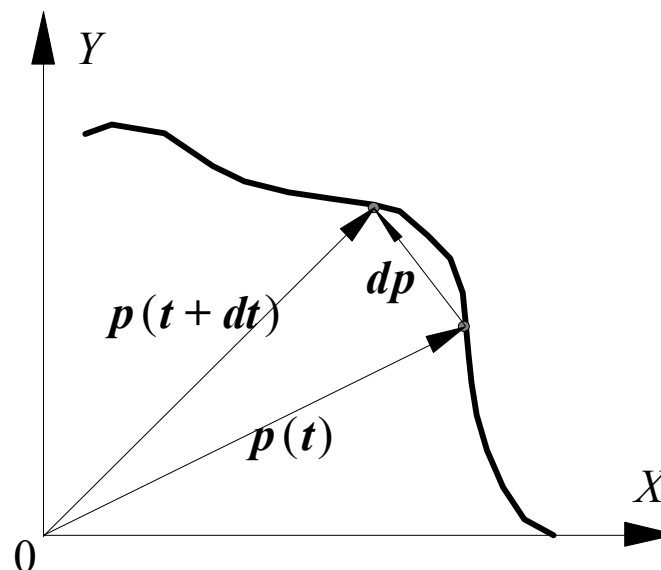


## 10.2 平面形状的分类

### (2) 参数曲线

- 点在2-D空间  
移动得到的轨迹
- 位置矢量的集合

参数为  $t$  时的点速度



$$\mathbf{p}'(t) = \frac{d\mathbf{p}}{dt} = \lim_{dt \rightarrow 0} \frac{\mathbf{p}(t+dt) - \mathbf{p}(t)}{dt} = \begin{bmatrix} \lim_{dt \rightarrow 0} \frac{x(t+dt) - x(t)}{dt} \\ \lim_{dt \rightarrow 0} \frac{y(t+dt) - y(t)}{dt} \end{bmatrix} = \begin{bmatrix} x'(t) \\ y'(t) \end{bmatrix}$$



## 10.2 平面形状的分类

### (3) 规则曲线 (**regular**正则)

- 如果一条参数曲线的速度永远不为零，则称该曲线为规则曲线
- 规则曲线速度的一个重要性质：
  - ✓ 各点的速度矢量都与曲线在该点相切
- 归一化以使沿曲线的切向矢量为单位大小

$$\alpha(t) = \frac{\mathbf{p}'(t)}{\|\mathbf{p}'(t)\|} \Rightarrow \|\alpha(t)\| = 1$$



## 10.3 形状特性的描述

### 形状和尺寸

任何目标均可用它的形状和尺寸来描述。

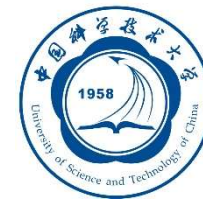
形状性质与尺寸性质不相关。

描述微结构的形状参数应具有一些共性：

- 1) 定量描述能力
- 2) 对形状变化比较敏感

10.3.1 形状紧凑性描述

10.3.2 形状复杂性描述



## 10.3.1 形状紧凑性描述

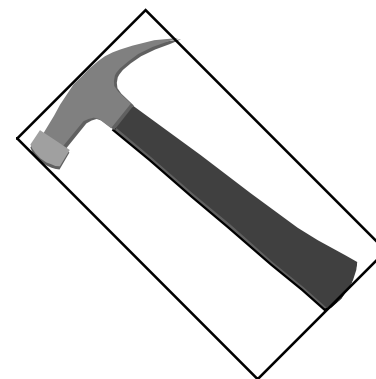
对应目标的几何参数，所以均与尺度有关

### 1. 外观比

- 外观比（aspect ratio）常用来描述塑性形变后目标的形状（细长程度）
- 可借助目标围盒定义

$$R = \frac{L}{W}$$

$L$ 和 $W$ 分别是目标围盒的长和宽





## 10.3.1 形状紧凑性描述

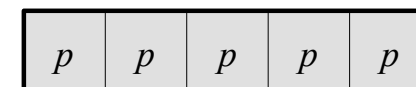
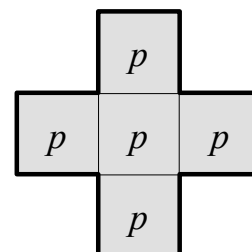
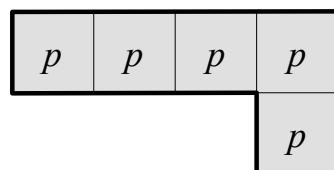
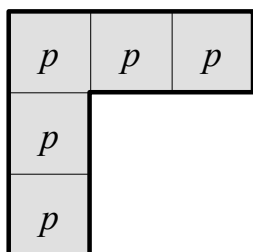
### 2、形状因子(form factor)

基于周长和面积  $F = \frac{\|B\|^2}{4\pi A}$

$F$  的值当区域为圆时达到最小 (1)

没有量纲，所以对尺度变化不敏感

问题：形状不同，形状因子可能相同



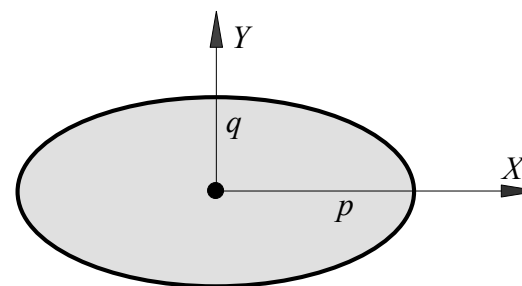
## 10.3.1 形状紧凑性描述

### 3、 偏心率(eccentricity)

描述了区域的紧凑性（伸长情况elongation）



利用整个区域的所有像素



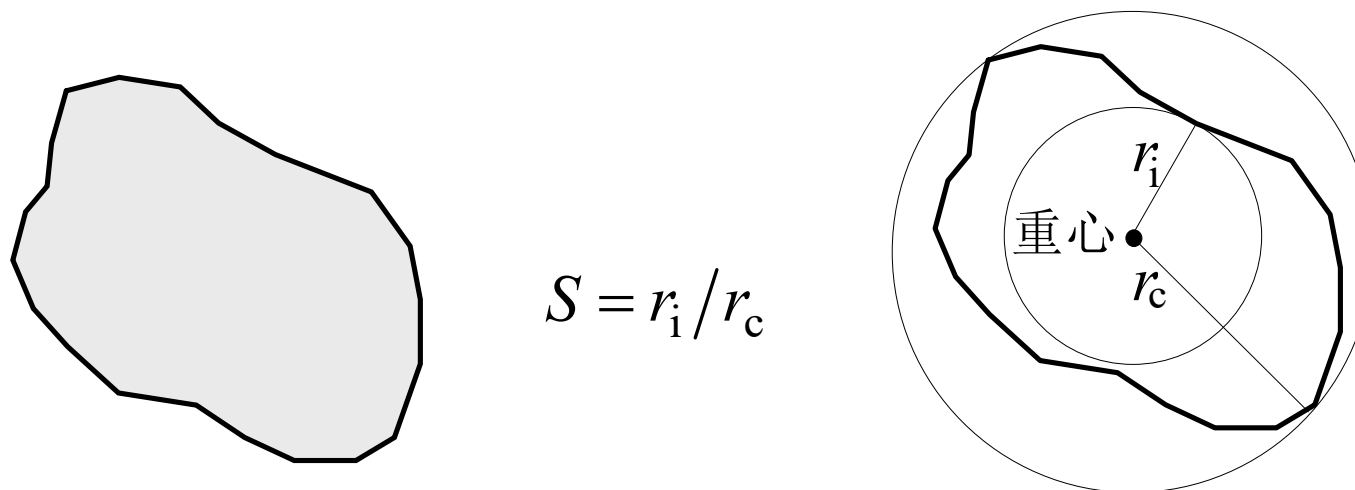
$$E = p / q$$

当区域为圆时E的值达到最小

## 10.3.1 形状紧凑性描述

### 4、球状性(sphericity)

原本指3-D目标的表面积和体积的比值  
基于区域的内切圆和外接圆（圆心为重心）



区域为圆时  $S$  值达到最大 (1)

## 10.3.1 形状紧凑性描述

### 4、球状性(sphericity)

内切圆和外接圆的圆心不在一处

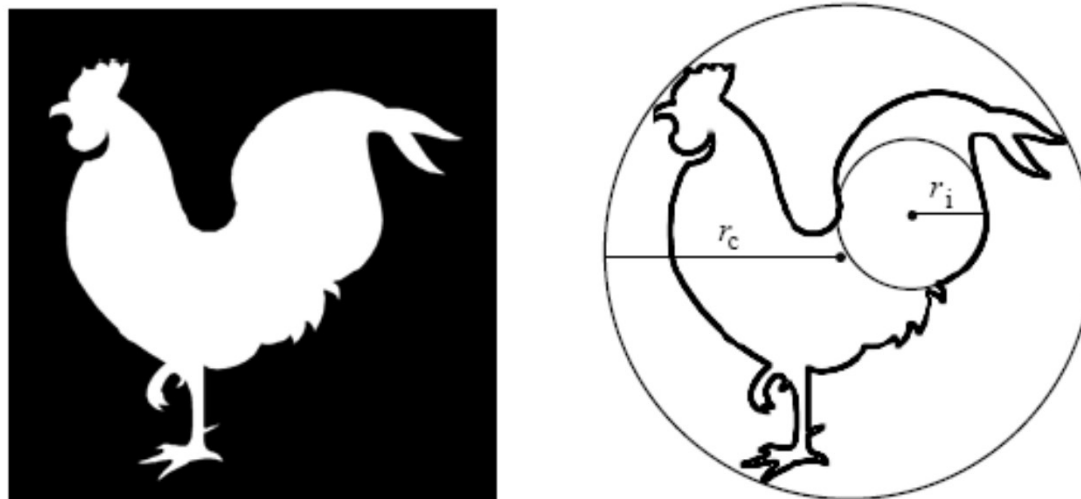


图 12.3.5 球状性定义示意图（内切圆和外接圆的圆心不在一处）



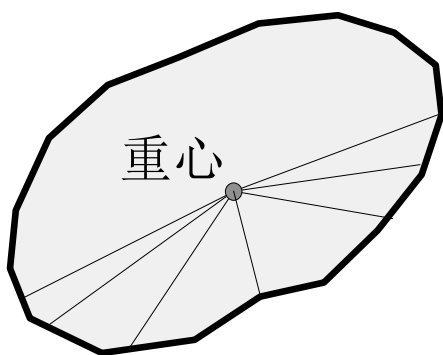


## 10.3.1 形状紧凑性描述

### 5、圆形性(circularity)

利用所有轮廓点

$$C = \frac{\mu_R}{\sigma_R}$$



$$\mu_R = \frac{1}{K} \sum_{k=0}^{K-1} \| (x_k, y_k) - (\bar{x}, \bar{y}) \|$$

$$\sigma_R^2 = \frac{1}{K} \sum_{k=0}^{K-1} \left[ \| (x_k, y_k) - (\bar{x}, \bar{y}) \| - \mu_R \right]^2$$

区域趋向圆时  $C$  值趋于无穷



## 10.3.1 形状紧凑性描述

### 区域描述符示例

几个典型的简单物体

物 体	$R$	$F$	$E$	$S$	$C$
正方形（边长为 1）	1	$4/\pi$	1	$\sqrt{2}/2$	9.102
正六边形（边长为 1）	1.155	1.103	1.010	0.866	22.613
正八边形（边长为 1）	1	1.055	1	0.924	41.616
长为 2 宽为 1 的长方形	2	1.432	2	0.447	3.965
长轴为 2 短轴为 1 的椭圆	2	1.190	2	0.500	4.412

R: 外观比; F: 形状因子; E: 偏心率; S: 球状性; C: 圆形性

## 10.3.1 形状紧凑性描述

### 几个描述符的比较

- 外观比：
  - 比较容易计算
  - 但不适合用来描述非规则性
- 形状因子：
  - 对非规则性比较敏感
  - 对形状伸长度方面不如外观比敏感
- 球状性：
  - 对伸长度和不规则性都比较敏感

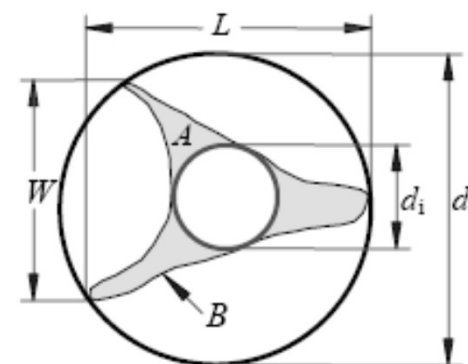


图 12.3.6 几个常用形状参数的示意



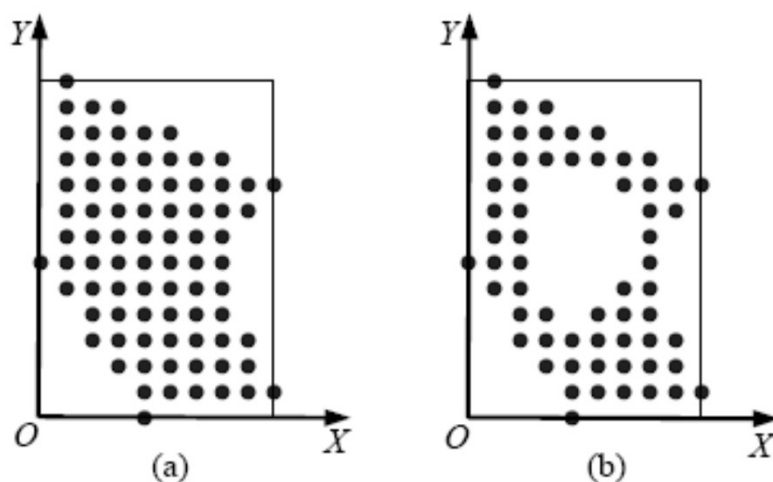
## 10.3.2 形状复杂性描述

- (1) 细度比例 (thinness ratio) : 形状因子的倒数,  
即  $4\pi(A/B^2)$
- (2) 面积周长比:  $A/B$
- (3) 矩形度 (rectangularity) : 定义为  $A/A_{\text{MER}}$ , 其中  
 $A_{\text{MER}}$  代表围盒面积。
- (4) 与边界的平均距离 (mean distance to the  
boundary) : 定义为  $A / \mu_R^2$
- (5) 轮廓温度 (temperature) : 由热力学原理得来,  
定义为  $T = \log_2[(2B)/(B-H)]$   
其中  $H$  为目标凸包的周长。

## 10.3.2 形状复杂性描述

### □ 饱和度 (saturation)

- ◆ 饱和度在一定意义下反映了目标的紧凑性 (紧致性)
- ◆ 它考虑的是目标在其围盒中的充满程度
- ◆ 具体可用属于目标的像素数与整个围盒所包含的像素数之比来计算



$$a = 57.8\% \quad b = 45\%$$

图 12.3.11 目标的饱和度



## 10.4 基于技术的描述

---

### 相关的描述符

由一种表达技术衍生出来的描述符

由基本描述符推导出来的描述符

10.4.1 基于多边形表达的形状描述

10.4.2 基于曲率的形状描述

10.4.3 基于变换的形状描述



## 10.4.1 基于多边形表达的形状描述

### 1. 直接计算的特征

可直接从多边形表达轮廓算出的特征：

- (1) 角点或顶点的个数
- (2) 角度和边的统计量，如均值，中值，方差
- (3) 最长边和最短边的长度，它们的长度比和它们间的角度
- (4) 最大内角与所有内角和的比值
- (5) 各个内角的绝对差的均值



## 10.4.1 基于多边形表达的形状描述

### 2. 形状数的比较

两个形状  $A$  和  $B$  之间的相似度  $k$  是这两个形状数之间的最大公共形状数

如果  $S_4(A) = S_4(B)$ ,  $S_6(A) = S_6(B)$ , ...,  $S_k(A) = S_k(B)$ ,  $S_{k+2}(A) \neq S_{k+2}(B)$ , ..., 则  $A$  和  $B$  的相似度就是  $k$

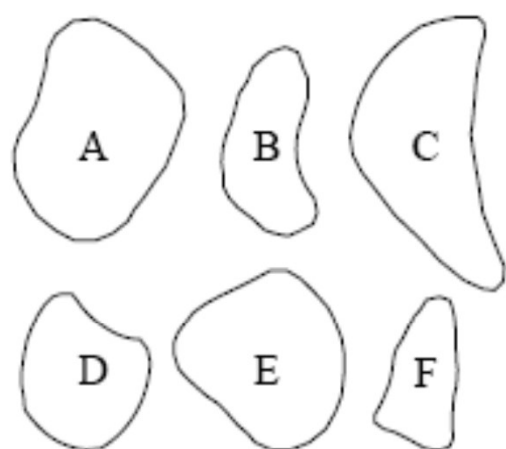
两个形状间的距离：它们相似度的倒数：

$$D(A, B) = 1/k$$

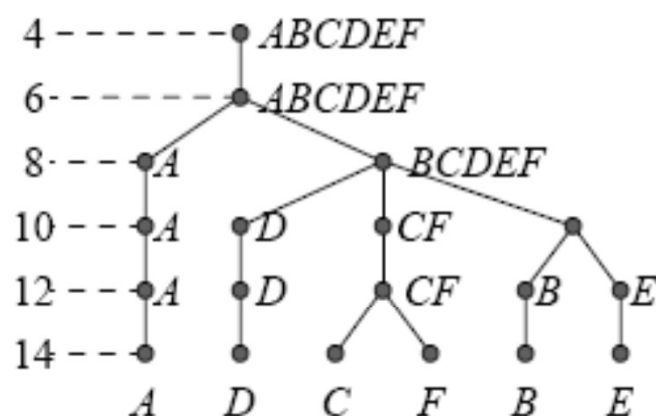


# 10.4.1 基于多边形表达的形状描述

## 2. 形状数的比较



(a)

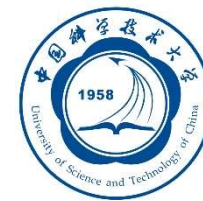


(b)

	A	B	C	D	E	F
A	$\infty$	6	6	6	6	6
B		$\infty$	8	8	10	8
C			$\infty$	8	8	12
D				$\infty$	8	8
E					$\infty$	8
F						$\infty$

(c)

图 12.4.1 形状数比较示例



## 10.4.1 基于多边形表达的形状描述

### 3. 区域的标记

对区域中所有像素沿不同方向进行投影

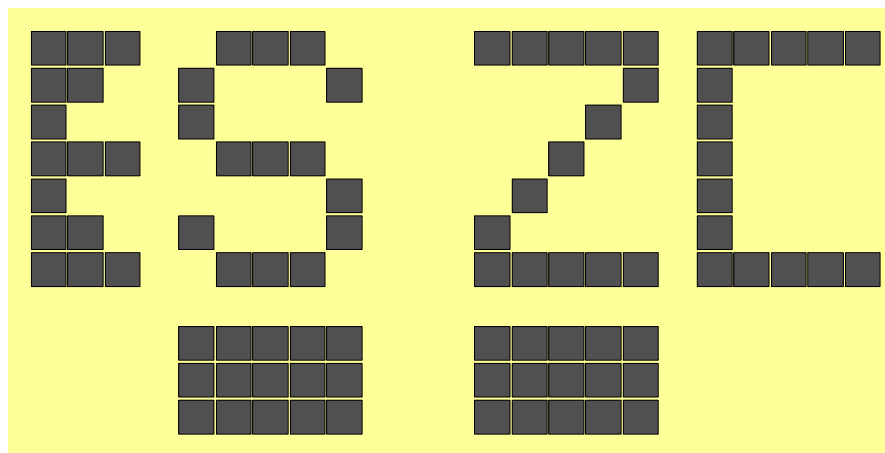
点阵表达的字母（多边形逼近后的结果）

垂直投影：

得到相同的结果

水平投影：

得到不同的结果





## 10.4.2 基于曲率的形状描述

### 1. 曲率与几何特征

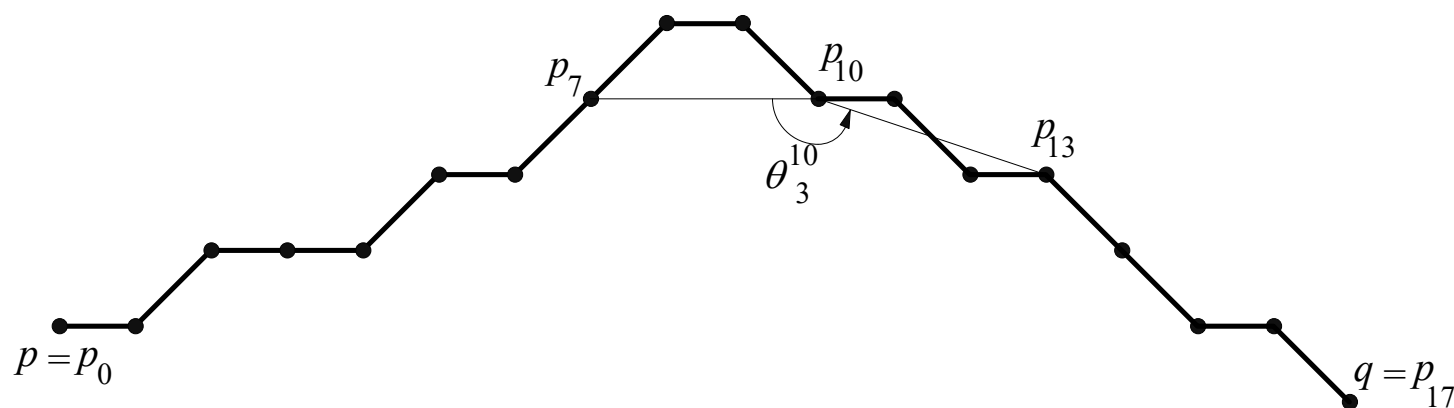
曲率	几何特征
连续零曲率	直线段
连续非零曲率	圆弧段
局部最大曲率绝对值	(一般) 角点
局部最大曲率正值	凸角点
局部最大曲率负值	凹角点
曲率过零点	拐点
大曲率平均绝对值或平方值	形状复杂性, 与弯曲能有关



## 10.4.2 基于曲率的形状描述

### 2. 离散曲率

给定一个离散点集合  $P=\{p_i\}_{i=0, \dots, n}$ ，它定义了一条数字曲线，在点  $p_i \in P$  处的  $k$ -阶曲率  $\rho_k(p_i) = 1 - |\cos \theta_k^i|$ ，其中  $\theta_k^i = \text{angle}(p_{i-k}, p_i, p_{i+k})$  是两个线段  $[p_{i-k}, p_i]$  和  $[p_i, p_{i+k}]$  之间的夹角，而  $k \in \{i, \dots, n-i\}$ 。





## 10.4.2 基于曲率的形状描述

### 2. 离散曲率

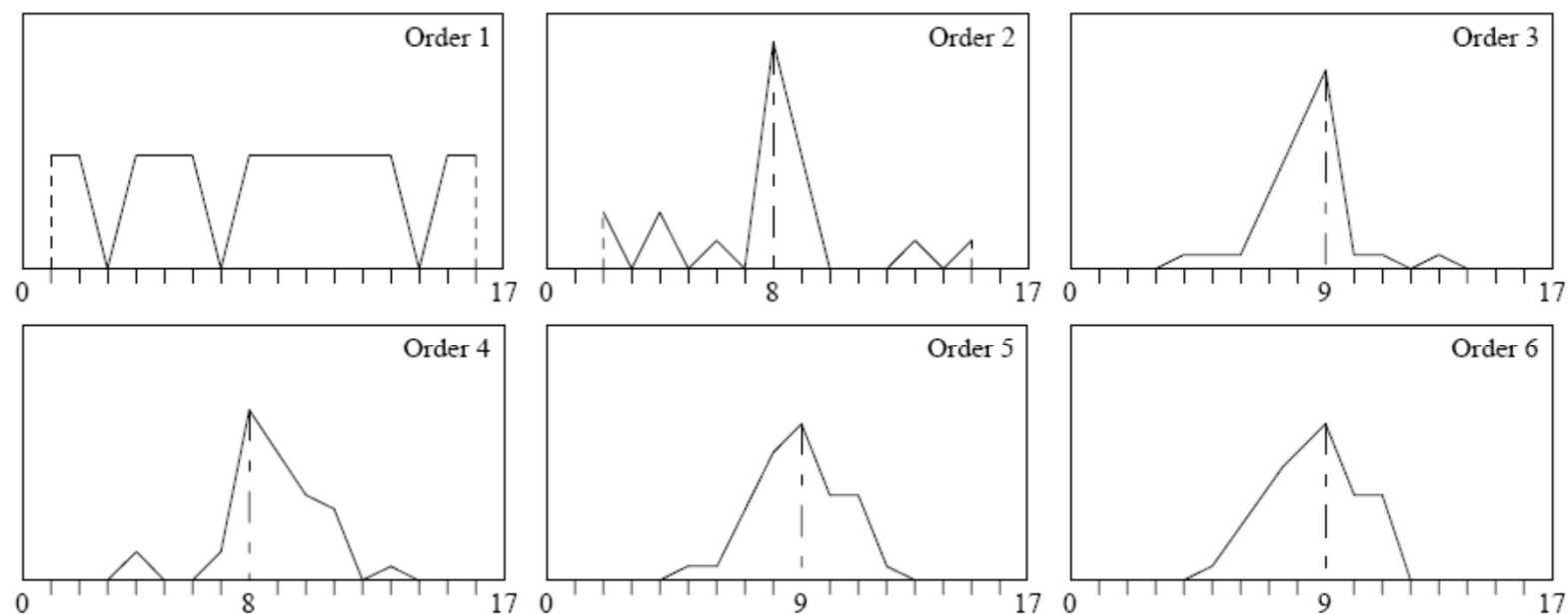


图 12.4.4 由图 12.4.3 得到的各阶曲率



## 10.4.2 基于曲率的形状描述

### 3. 离散曲率的计算

$$k(t) = \frac{x'(t)y''(t) - x''(t)y'(t)}{(x'(t)^2 + y'(t)^2)^{3/2}}$$

- (1) 先对  $x(t)$  和  $y(t)$  进行插值再求导数  
用有限差分的方法

$$\begin{aligned}x'(n) &= x(n) - x(n-1) \\ y'(n) &= y(n) - y(n-1) \\ x''(n) &= x'(n) - x'(n-1) \\ y''(n) &= y'(n) - y'(n-1)\end{aligned}$$

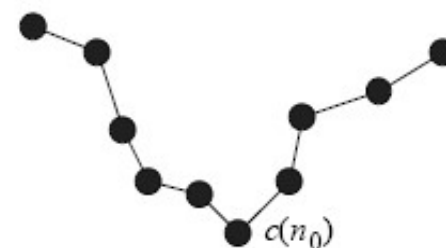


图 12.4.5 基于插值的曲率计算



## 10.4.2 基于曲率的形状描述

### 3. 离散曲率的计算

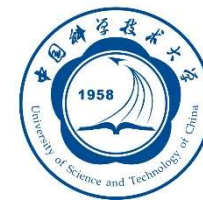
(1) 先对 $x(t)$ 和 $y(t)$ 进行插值再求导数

用B样条逼近轮廓

在点A ( $t=0$ ) 和点B ( $t=1$ ) 间的轮廓用下列多项式逼近

$$\begin{aligned}x(t) &= a_1 t^3 + b_1 t^2 + c_1 t + d_1 \\y(t) &= a_2 t^3 + b_2 t^2 + c_2 t + d_2 \\k &= 2 \frac{c_1 b_2 - c_2 b_1}{(c_1^2 + c_2^2)^{3/2}}\end{aligned}$$

$$\begin{aligned}b_1 &= \frac{1}{12}[(x_{n-2} + x_{n+2}) + 2(x_{n-1} + x_{n+1}) - 6x_n] \\b_2 &= \frac{1}{12}[(y_{n-2} + y_{n+2}) + 2(y_{n-1} + y_{n+1}) - 6y_n] \\c_1 &= \frac{1}{12}[(x_{n+2} - x_{n-2}) + 4(x_{n-1} + x_{n+1})] \\c_2 &= \frac{1}{12}[(y_{n+2} - y_{n-2}) + 4(y_{n-1} + y_{n+1})]\end{aligned}$$



## 10.4.2 基于曲率的形状描述

### 3. 离散曲率的计算

(2) 根据矢量间的夹角来定义等价的曲率测度

$$\mathbf{u}_i(n) = [x(n) - x(n-i), y(n) - y(n-i)]$$

$$\mathbf{v}_i(n) = [x(n) - x(n+i), y(n) - y(n+i)]$$

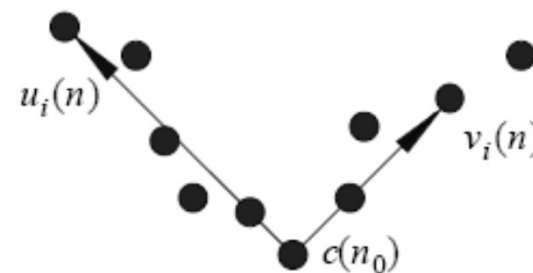


图 12.4.6 基于角度的曲率计算





## 10.4.2 基于曲率的形状描述

### 4. 基于曲率的描述符

- (1) 曲率的统计值：平均值，方差，熵
- (2) 曲率最大、最小点，拐点
- (3) 弯曲能

将给定曲线弯曲成所需形状而需要的能量  
设曲线长度为 $L$ ，在其上一点 $k$ 的曲率为 $k(t)$

$$BE = \sum_{t=1}^L k^2(t)$$



## 10.4.3 基于变换的形状描述

从1个封闭边界可得到1个复数序列

$$s(k) = u(k) + jv(k) \quad k = 0, 1, \dots, N-1$$

将序列进行傅里叶变换

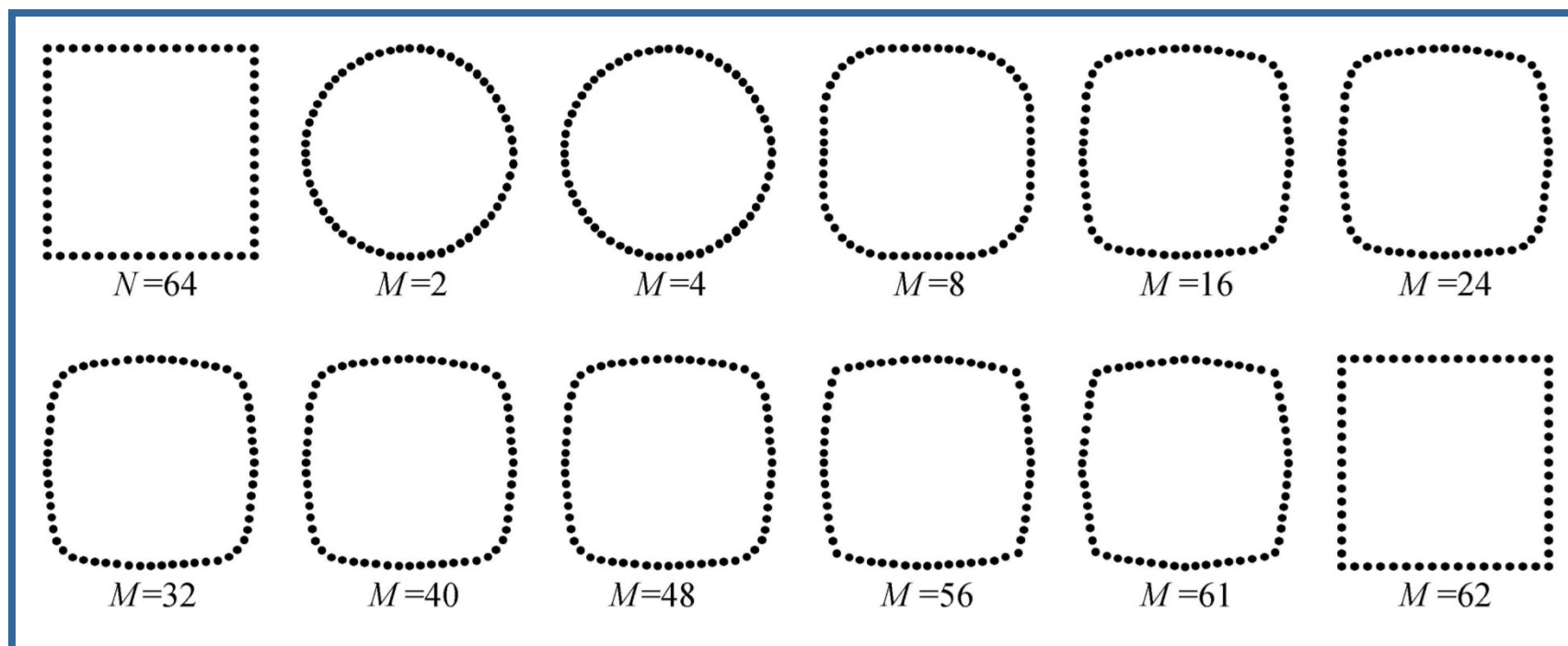
$$S(w) = \frac{1}{N} \sum_{k=0}^{N-1} s(k) \exp[-j2\pi wk / N] \quad w = 0, 1, \dots, N-1$$

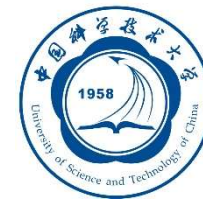
取傅里叶变换系数表达轮廓

$$\hat{s}(k) = \sum_{w=0}^{M-1} S(w) \exp[j2\pi wk / N] \quad k = 0, 1, \dots, N-1$$

## 10.4.3 基于变换的形状描述

少数傅氏描述子能反映边缘的大致情况。  
一些低频系数能够反映大体形状；  
加上更多的高频系数项可以精确反映形状特征。





## 10.5 拓扑结构描述

### 交叉数(Crossing Number)

考虑像素  $p$  的8个邻域像素  $q_i$  ( $i = 0, \dots, 7$ )

$S_4(p)$ : 在 $p$ 的8-邻域中4-连通组元的数目

$$S_4(p) = \prod_{i=0}^7 q_i + \frac{1}{2} \sum_{i=0}^7 |q_{i+1} - q_i|$$

### 连接数(Connectivity Number)

$C_8(p)$ : 在 $p$ 的8-邻域中8-连通组元的数目

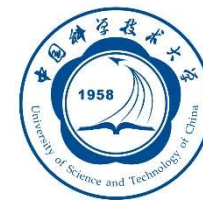
$$C_8(p) = q_0 q_2 q_4 q_6 + \sum_{i=0}^3 (\bar{q}_{2i} - \bar{q}_{2i} \bar{q}_{2i+1} \bar{q}_{2i+2})$$



## 10.5 拓扑结构描述

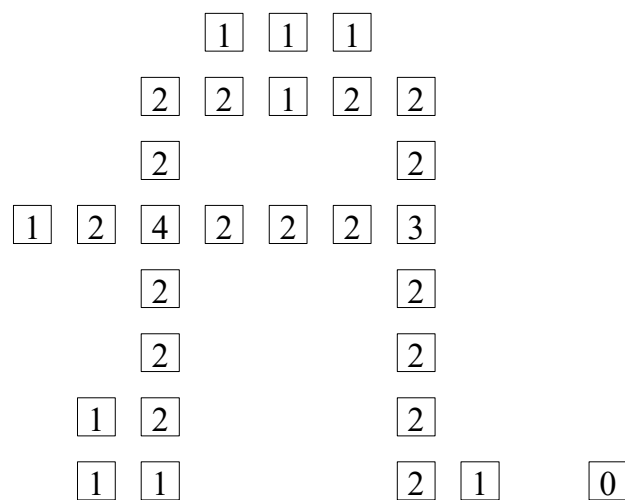
区分4-连通组元  $F$  中的各个像素  $p$ :

- (1) 如果  $S_4(p) = 0$ , 则  $p$  是一个孤立点  
(即  $F = \{p\}$ )
- (2) 如果  $S_4(p) = 1$ , 则  $p$  或者是一个边界点或者是一个内部点
- (3) 如果  $S_4(p) = 2$ , 则  $p$  对保持  $F$  的4-连通是必不可少的一个点
- (4) 如果  $S_4(p) = 3$ , 则  $p$  是一个分叉点
- (5) 如果  $S_4(p) = 4$ , 则  $p$  是一个交叉点

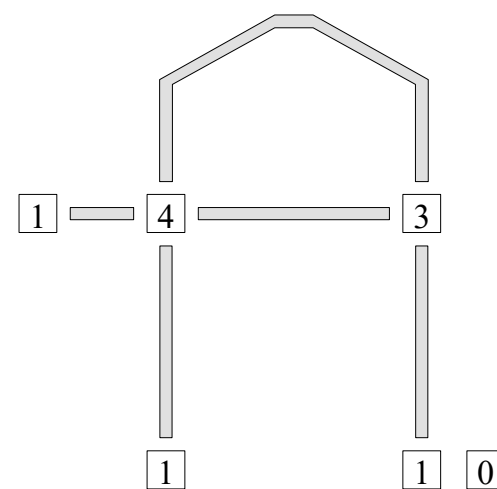


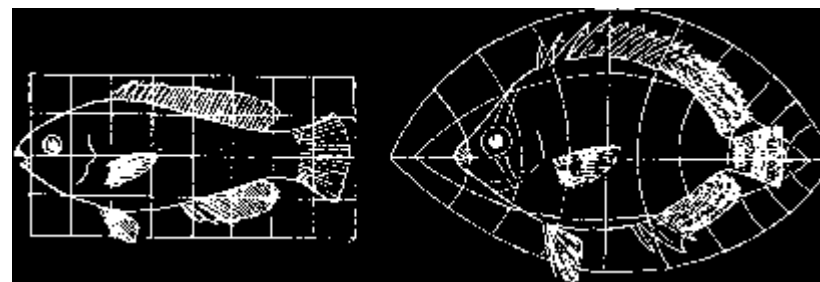
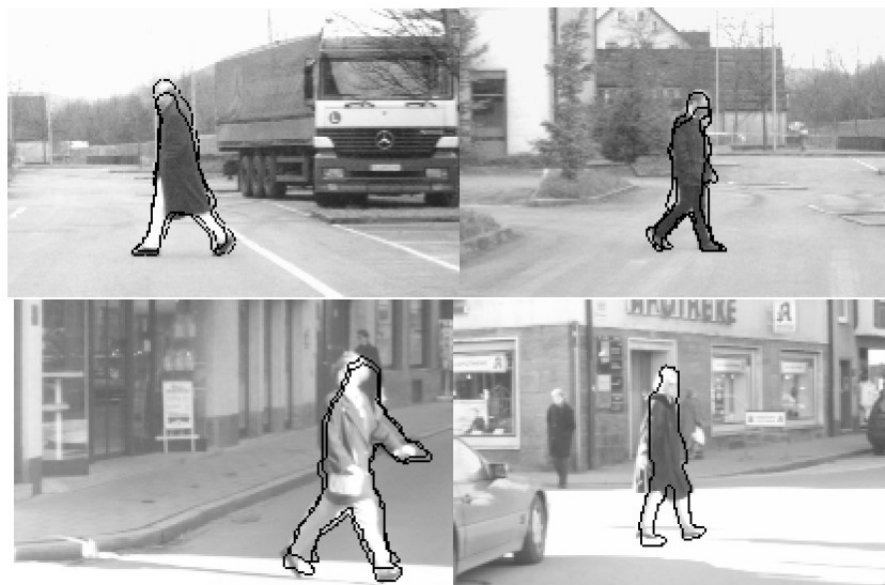
## 10.5 拓扑结构描述

连通区域图

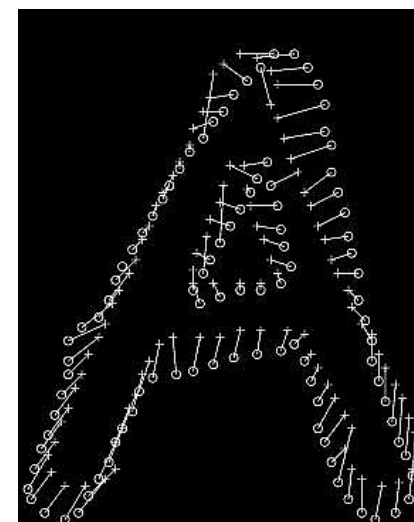
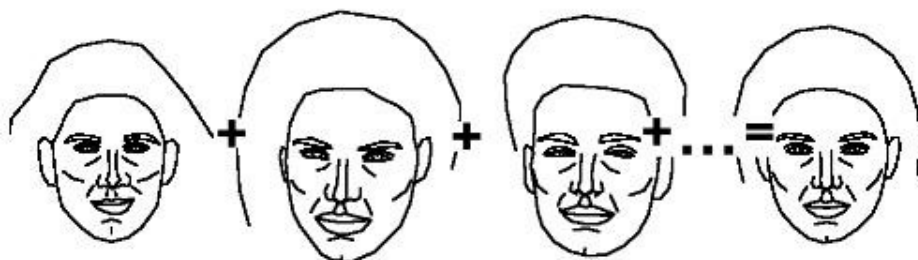


拓扑结构图

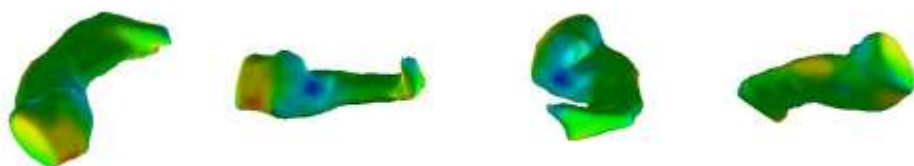




## Step into The Modern Computer Vision: Shape Matching

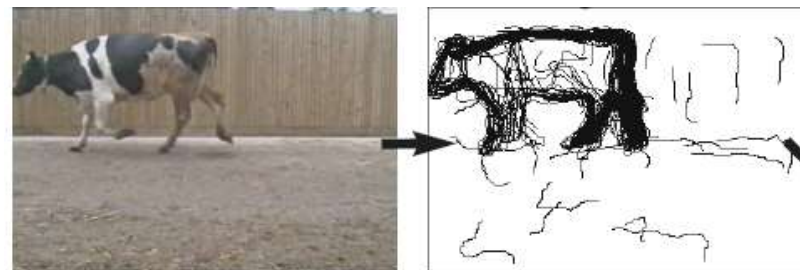


# Role of shape



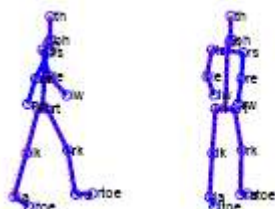
## Analysis of anatomical structures

Figure from Grimson & Golland



## Recognition, detection

Fig from Opelt et al.

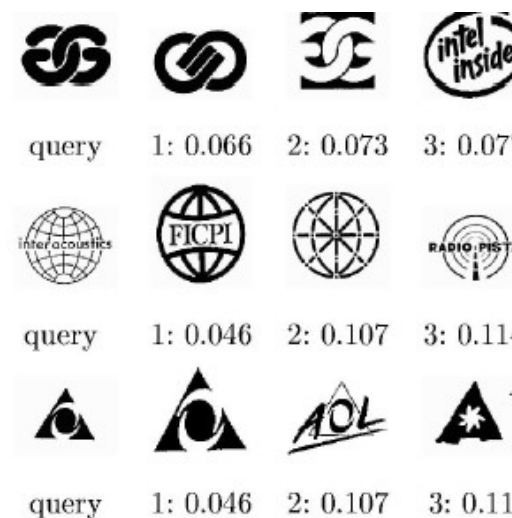


## Pose



## Morphology

<http://usuarios.lycos.es/lawebdelosfosiles/i>



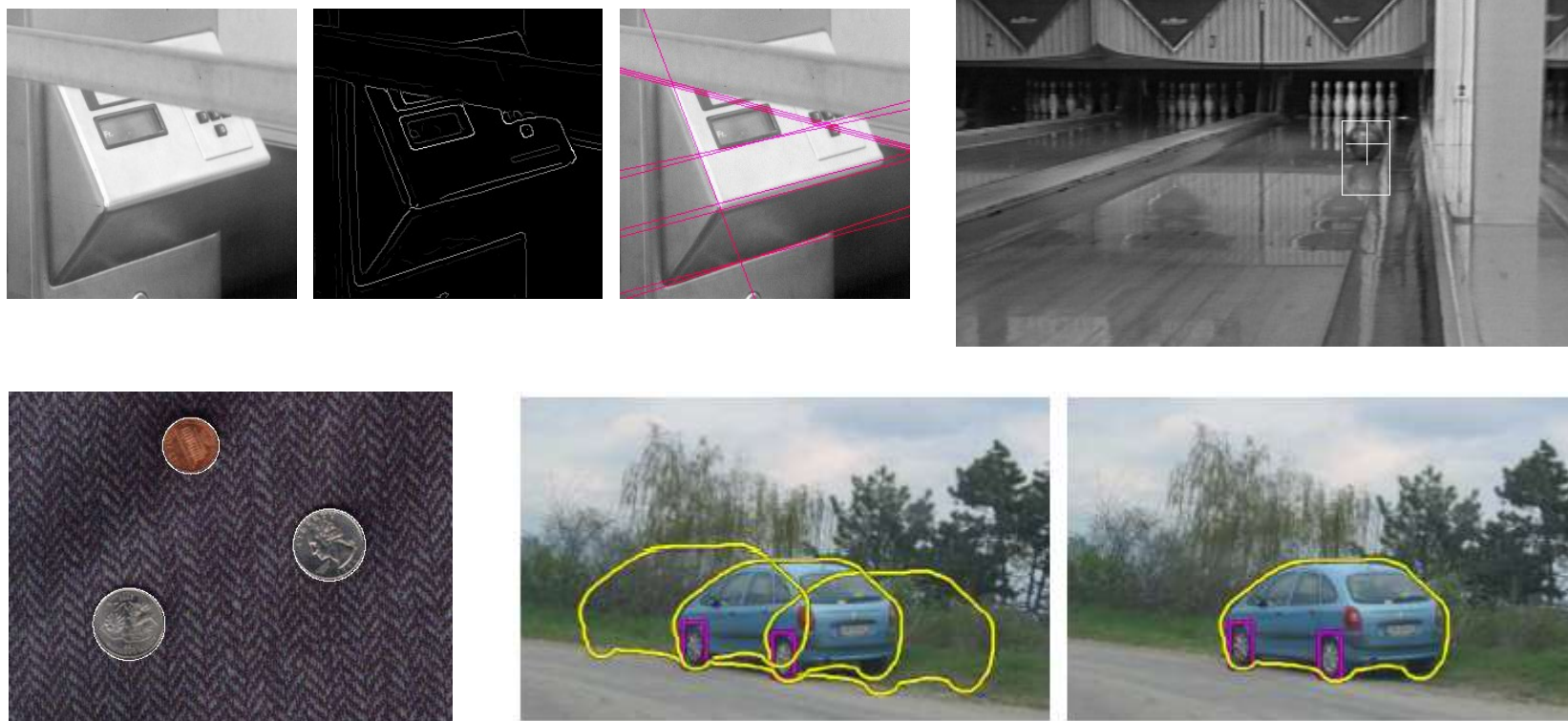
## Characteristic feature

Fig from Belongie et al.



# Fitting

- Associate a model with observed features



For example, the model could be a line, a circle, or an arbitrary shape.



# Outline

---

- Chamfer Distance: global model
- Shape Context: local matching



# Chamfer distance

- Average distance to nearest feature

$$D_{chamfer}(T, I) \equiv \frac{1}{|T|} \sum_{t \in T} d_I(t)$$

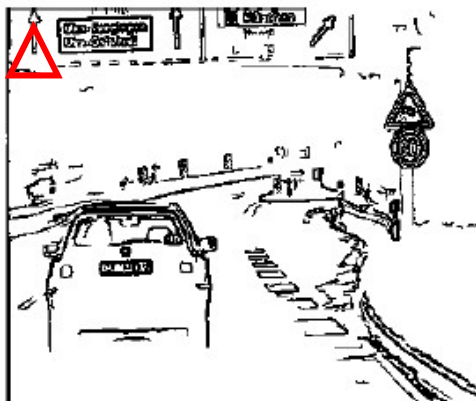
- $T$ : template shape  $\rightarrow$  a set of points
- $I$ : image to search  $\rightarrow$  a set of points
- $d_I(t)$ : min distance for point  $t$  to some point in  $I$

# Chamfer distance

- Average distance to nearest feature

$$D_{chamfer}(T, I) \equiv \frac{1}{|T|} \sum_{t \in T} d_I(t)$$

*How is the measure different than just filtering with a mask having the shape points?*



**Edge image**

*How expensive is a naïve implementation?*

# Distance transform

*Image features (2D)*


*Distance Transform*

1	0	1	2	3	4	3	2
1	0	1	2	3	3	2	1
1	0	1	2	3	2	1	0
1	0	0	1	2	1	0	1
2	1	1	2	1	0	1	2
3	2	2	2	1	0	1	2
4	3	3	2	1	0	1	2
5	4	4	3	2	1	0	1

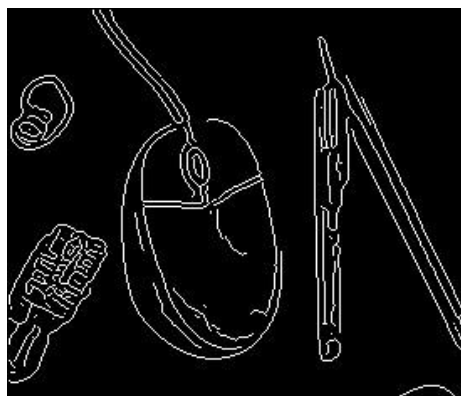
**Distance Transform** is a function  $D(\cdot)$  that for each image pixel  $p$  assigns a non-negative number  $D(p)$  corresponding to distance from  $p$  to the nearest **feature** in the image  $I$

Features could be edge points, foreground points, ...

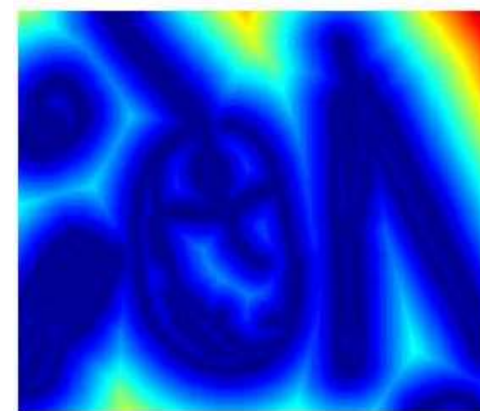
# Distance transform



original



edges



distance transform

Value at  $(x,y)$  tells how far that position is from the nearest edge point (or other binary image structure)

**>> help bwdist**



# Distance transform (1D)

Two pass  $O(n)$  algorithm for 1D  $L_1$  norm

1. Initialize: For all  $j$   
 $D[j] \leftarrow 1_P[j]$  // 0 if  $j$  is in  $\mathbf{P}$ , infinity otherwise
2. Forward: For  $j$  from 1 up to  $n-1$   
 $D[j] \leftarrow \min(D[j], D[j-1]+1)$
3. Backward: For  $j$  from  $n-2$  down to 0  
 $D[j] \leftarrow \min(D[j], D[j+1]+1)$

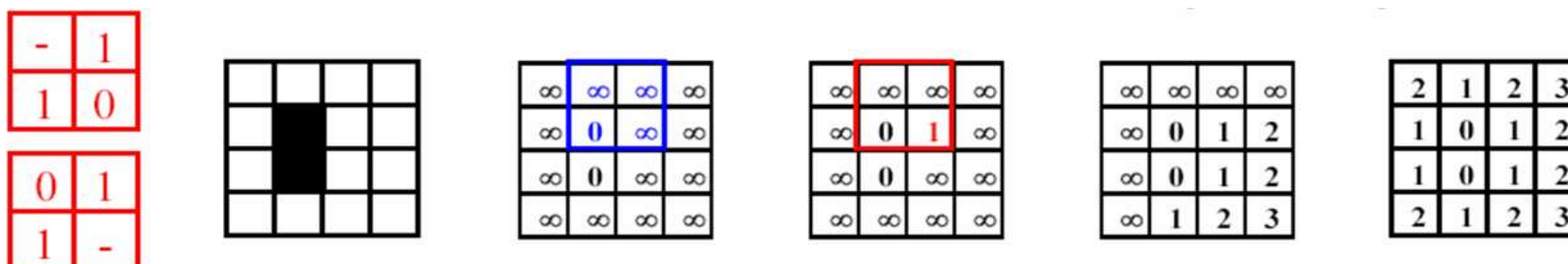
$\infty$	0	$\infty$	0	$\infty$	$\infty$	$\infty$	0	$\infty$
----------	---	----------	---	----------	----------	----------	---	----------

$\infty$	0	1	0	1	2	3	0	1
----------	---	---	---	---	---	---	---	---



# Distance Transform (2D)

- 2D case analogous to 1D
  - Initialization
  - Forward and backward pass
    - Fwd pass finds closest above and to left
    - Bwd pass finds closest below and to right

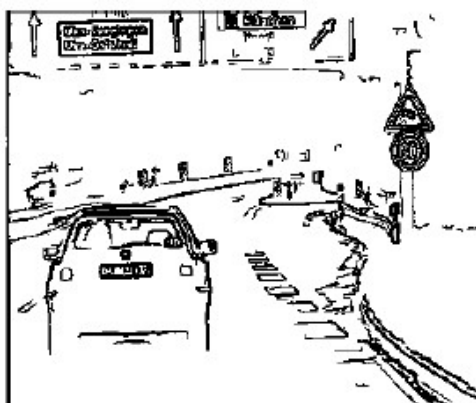




# Chamfer distance

- Average distance to nearest feature

$$D_{chamfer}(T, I) \equiv \frac{1}{|T|} \sum_{t \in T} d_I(t)$$

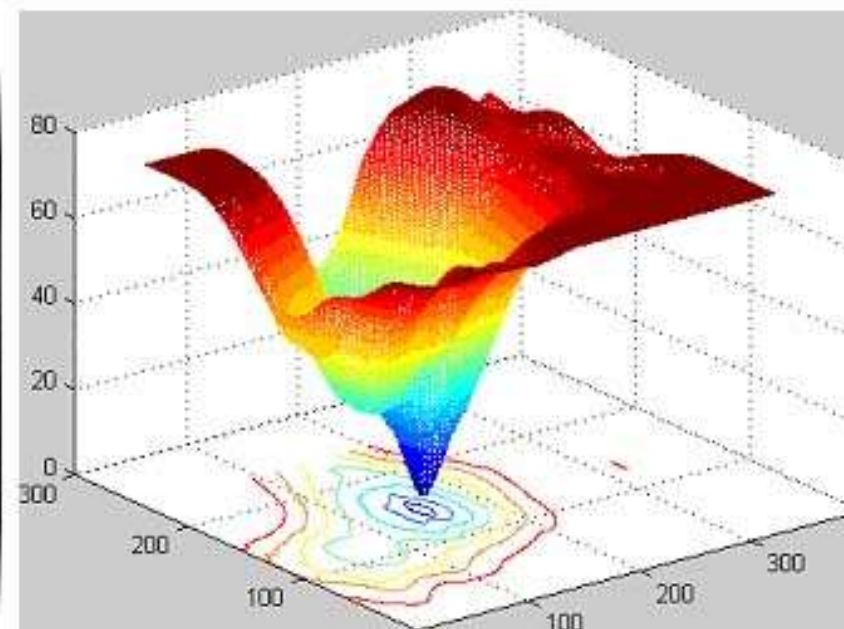


**Edge image**



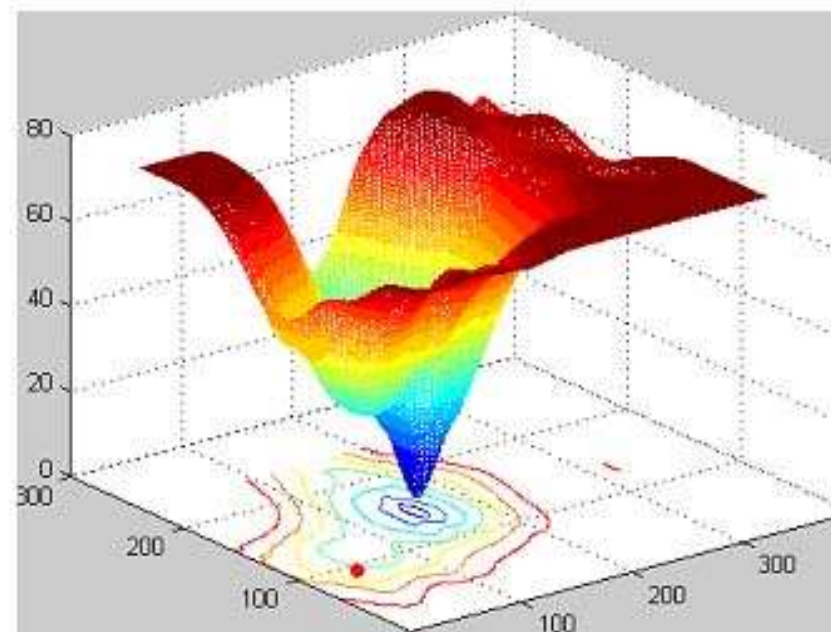
**Distance  
transform image**

# Chamfer Matching

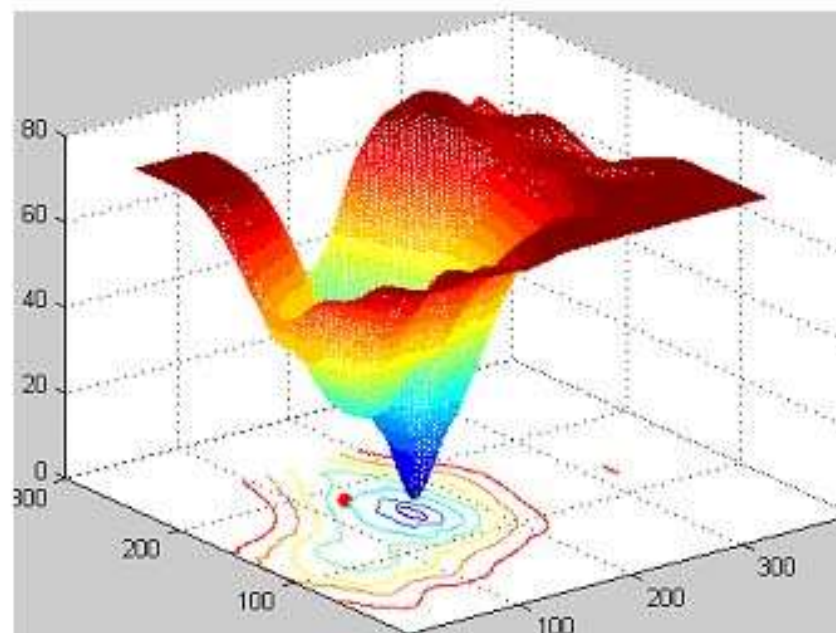
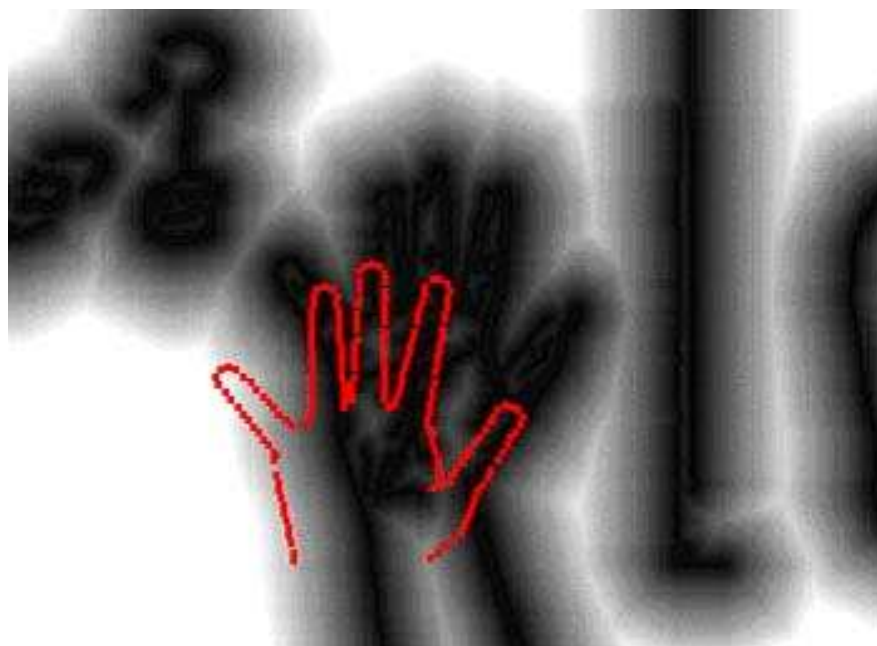


- Distance image provides a smooth cost function
- Efficient searching techniques can be used to find correct template

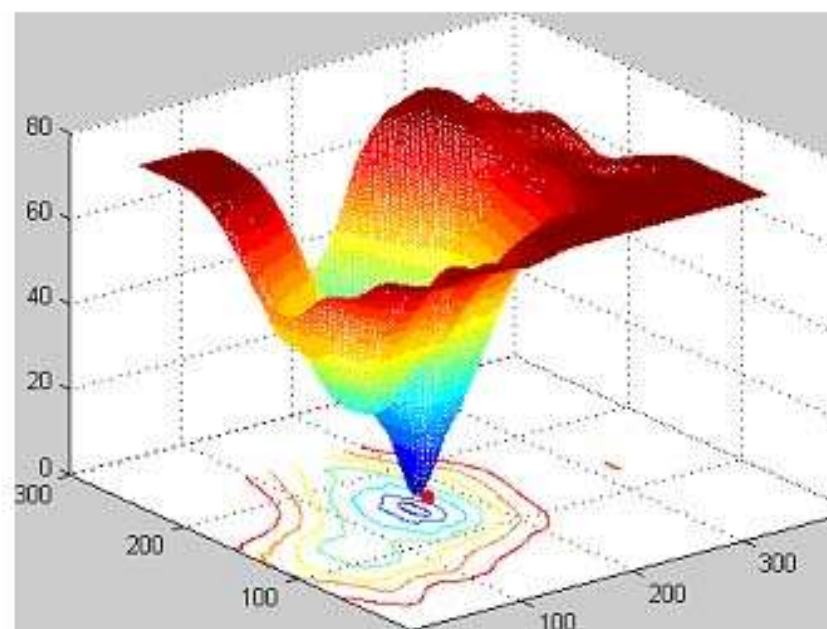
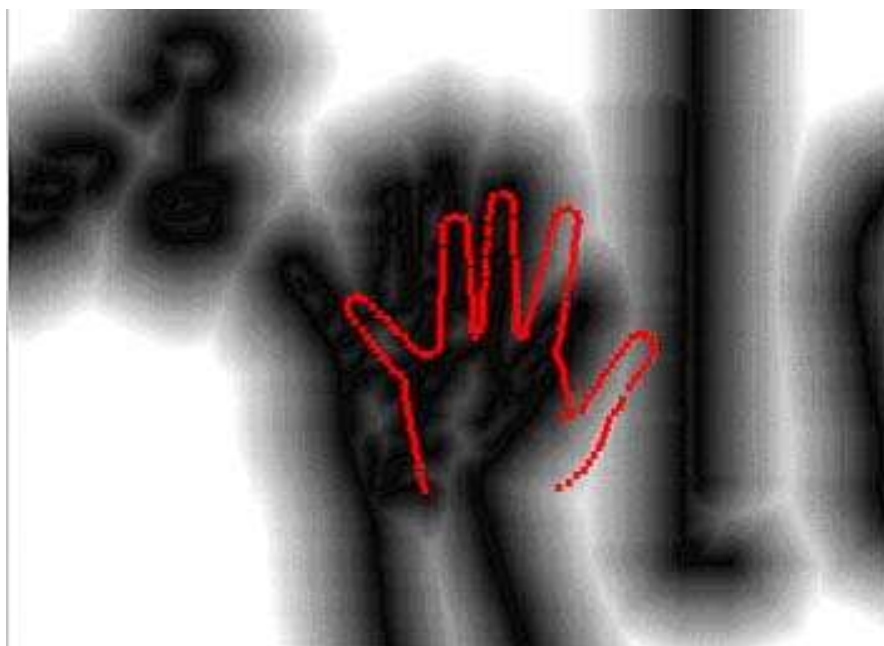
# Chamfer Matching



# Chamfer Matching

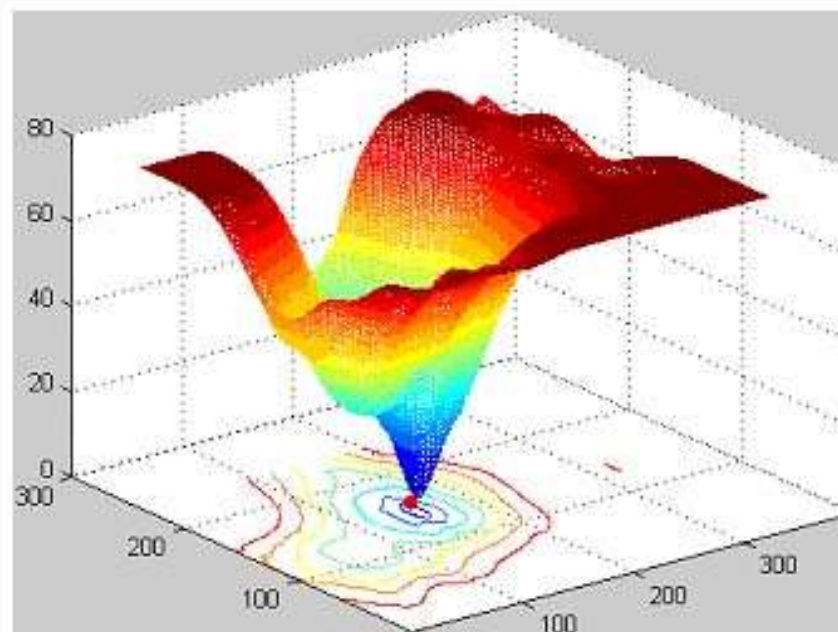
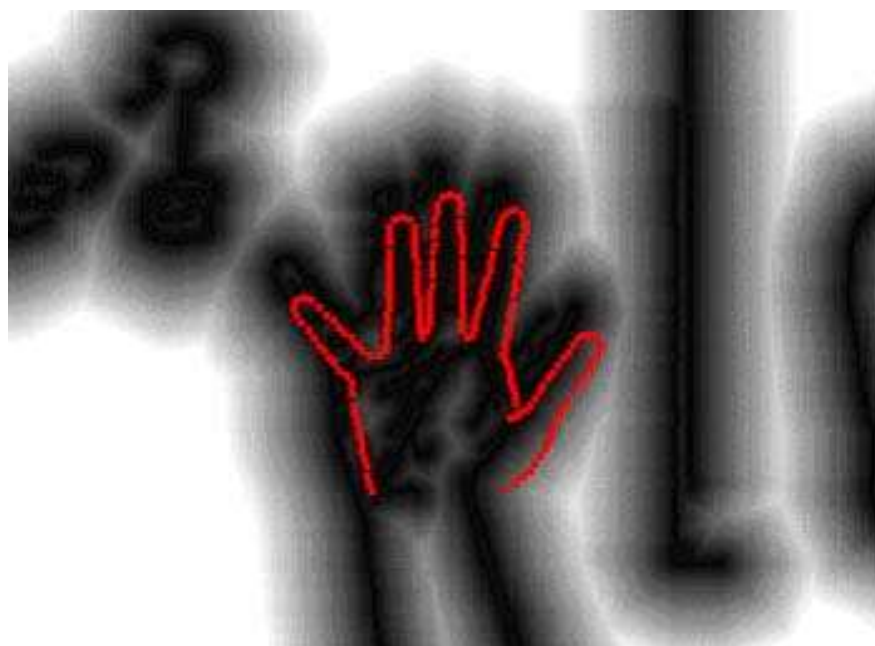


# Chamfer Matching

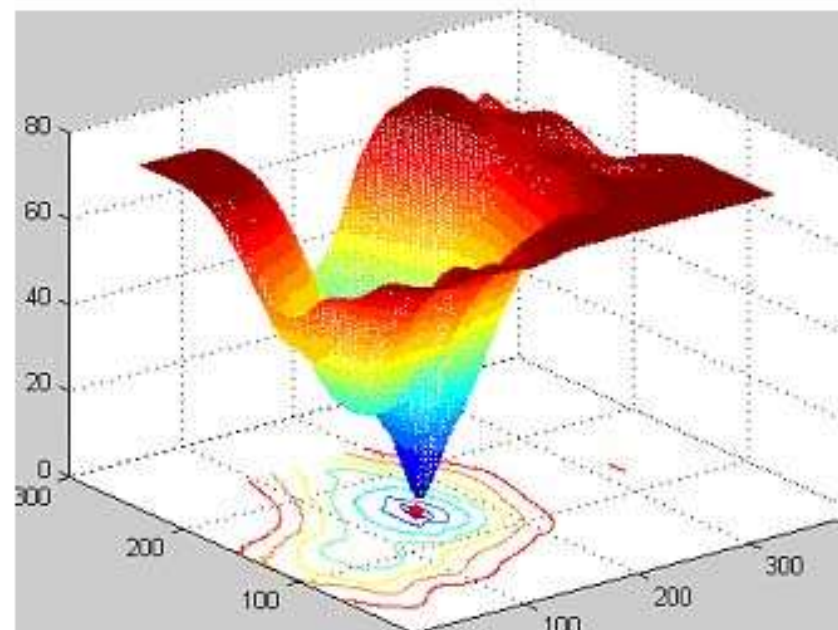




# Chamfer Matching



# Chamfer Matching





# Discussion

---

- Chamfer Matching
  - ◆ Variant to scale and rotation
  - ◆ Sensitive to small shape changes
  - ◆ Need large number of template shapes

But

- ◆ Robust to clutter
- ◆ Computationally cheap

Way out:

- ◆ Multiscale matching
- ◆ Hierarchical model organization





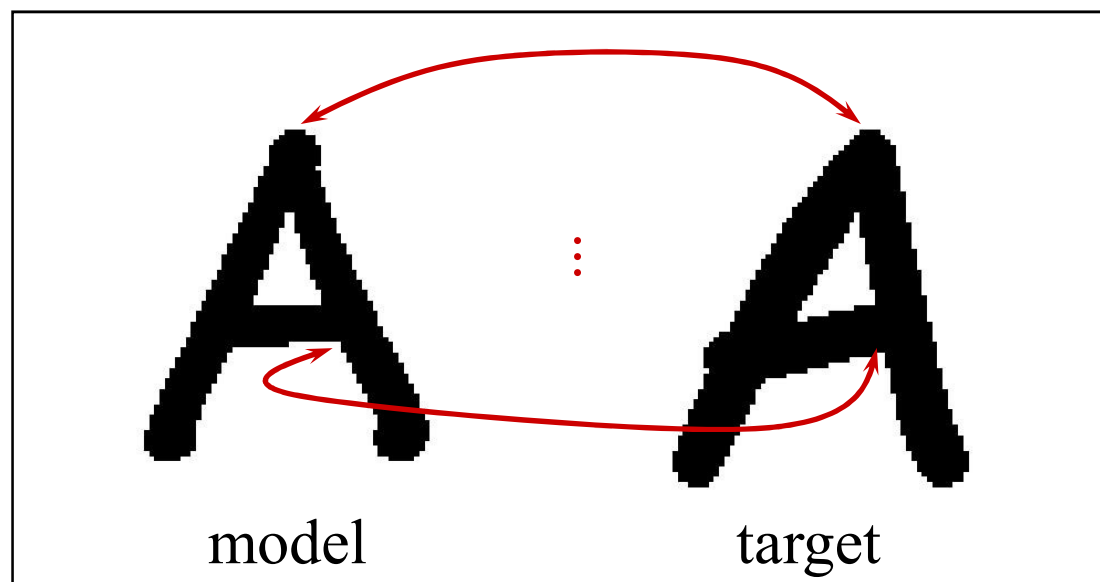
# Outline

---

- Chamfer Distance: global model
- Shape Context: local matching

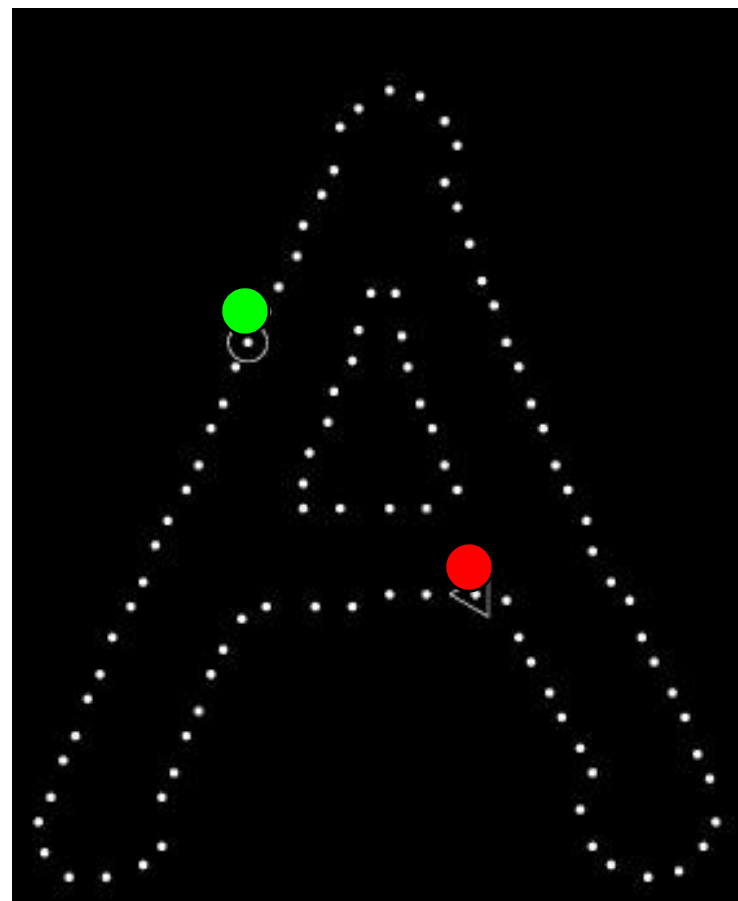
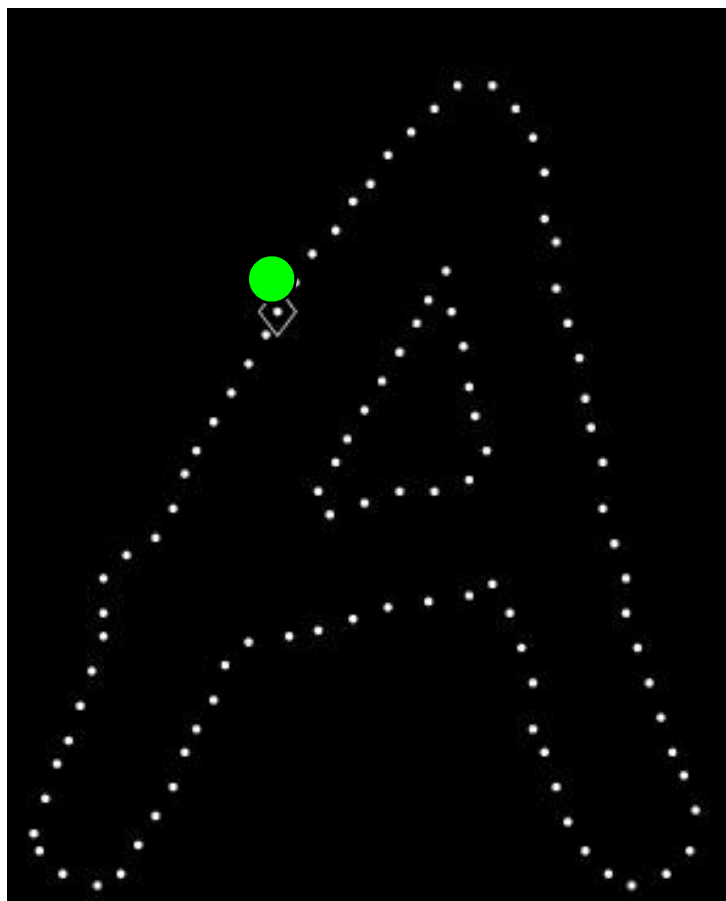
# Matching Framework

---



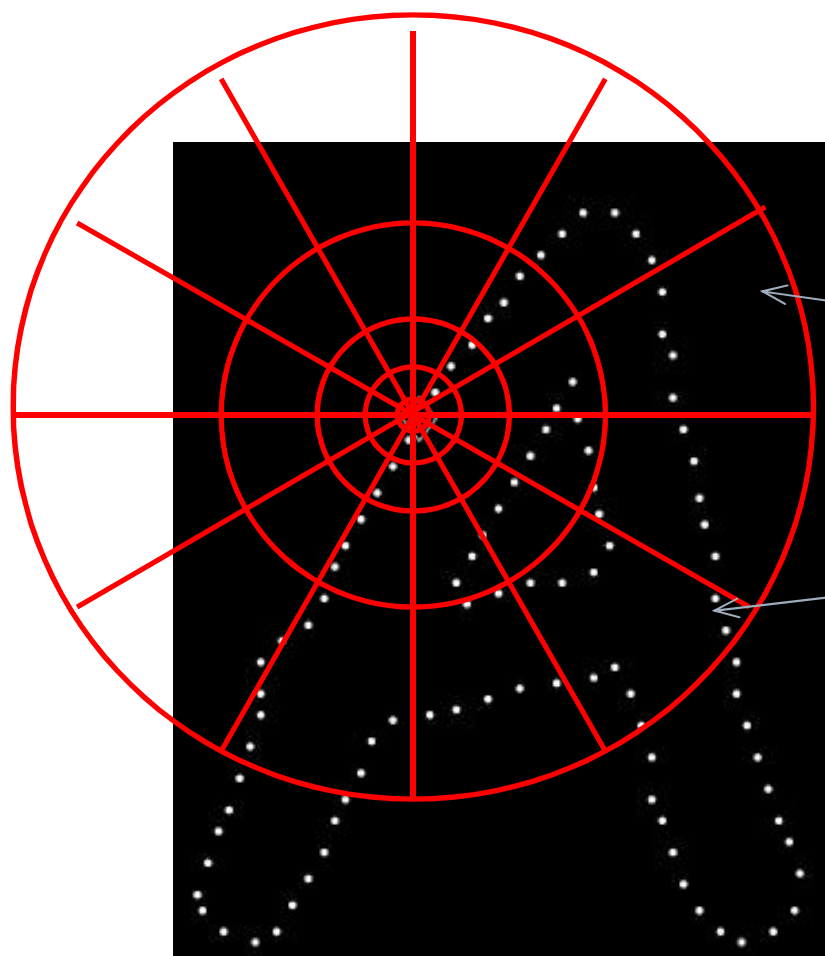
- Find correspondences between points on shape
- Estimate transformation
- Measure similarity

# Comparing shapes



What points on these two sampled contours are most similar? How do you know?

# Shape Context



Count the number of points  
inside each bin, e.g.:

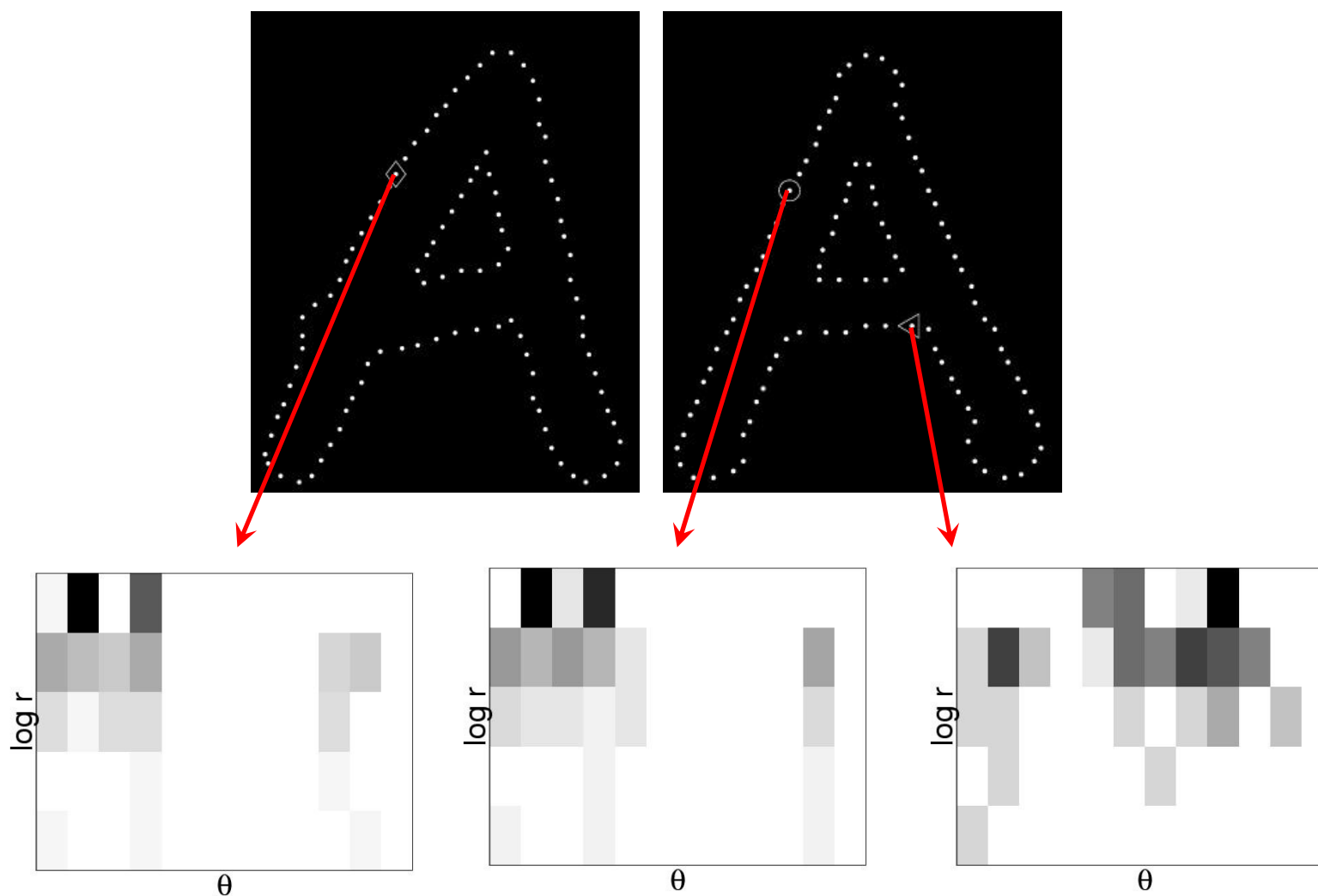
Count = 4

⋮

Count = 10

Compact representation of  
distribution of points  
relative to each point

# Shape Context



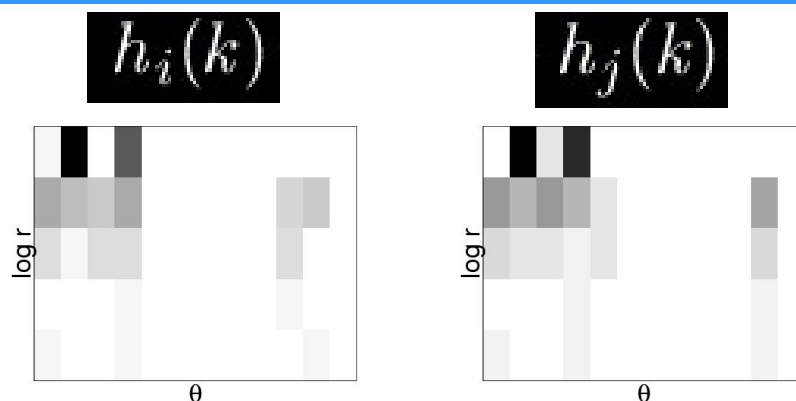


# Shape Contexts

---

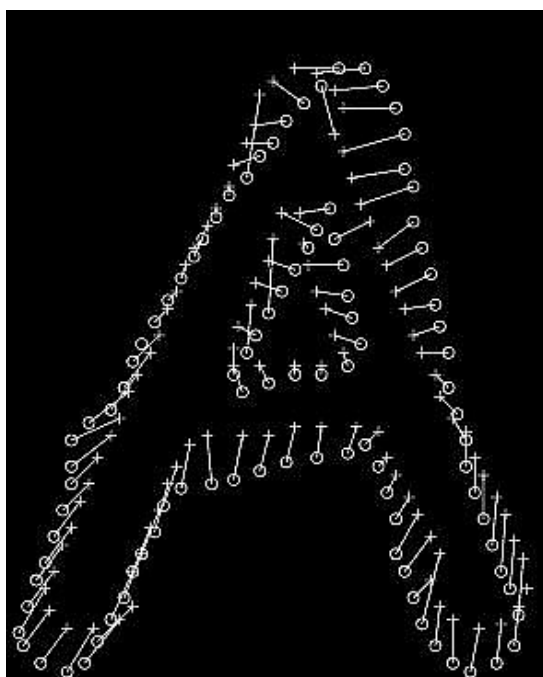
- ❑ Invariant under translation and scale
- ❑ Can be made invariant to rotation by using local tangent orientation frame
- ❑ Tolerant to small affine distortion
  - ❑ Log-polar bins make spatial blur proportional to  $r$
  - ❑ Cf. Spin Images (Johnson & Hebert) - range image registration

# Comparing Shape Contexts



Compute matching costs using Chi Squared distance

$$C_{ij} = \frac{1}{2} \sum_{k=1}^K \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)}$$



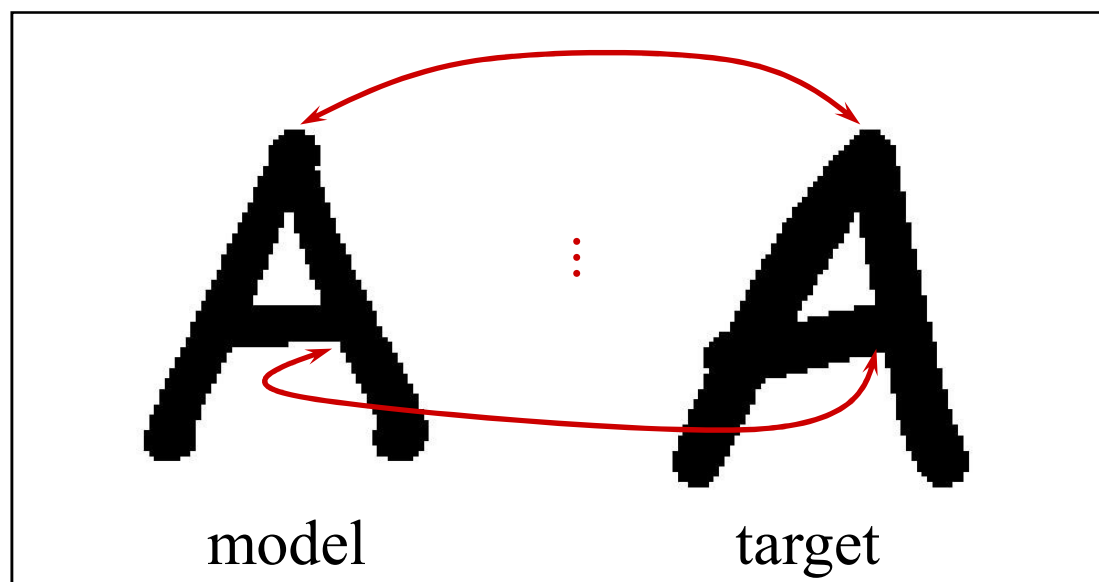
Recover correspondences by solving linear assignment problem with costs  $C_{ij}$

$$H(\pi) = \sum_i C(p_i, q_{\pi(i)})$$

Bipartite graph matching: Hungarian algorithm

# Matching Framework

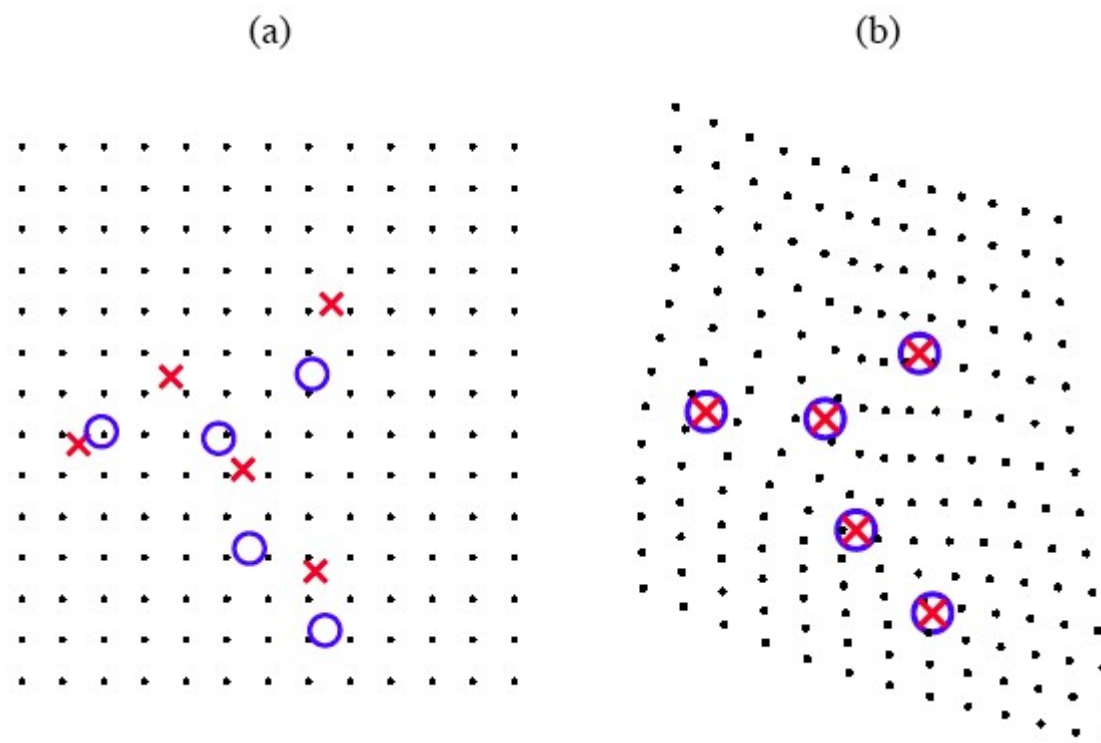
---



- Find correspondences between points on shape
- **Estimate transformation**
- Measure similarity



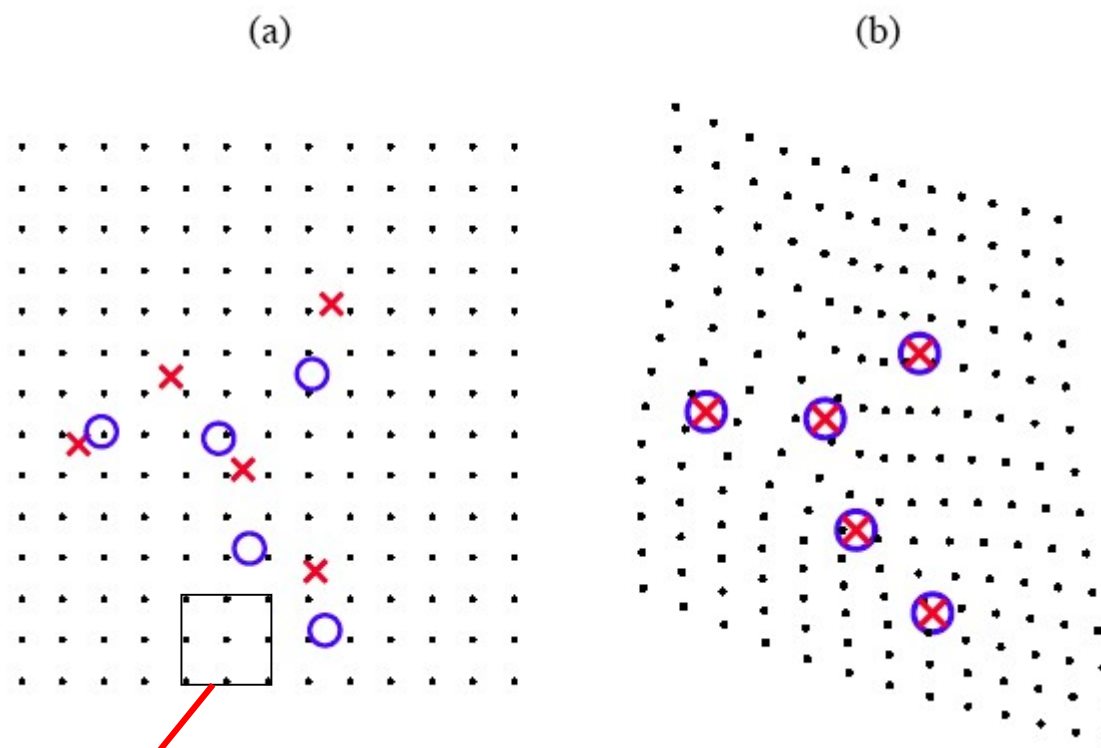
# Thin Plate Spline Model



We would need two functions:

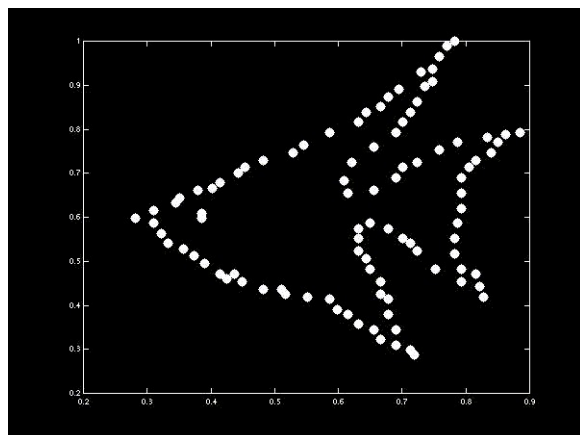
- 1) **`tps_model = est_tps( source_pts, target_pts );`**
- 2) **`morphed_im = morph(im_source, tps_model);`**

# Thin Plate Spline Model

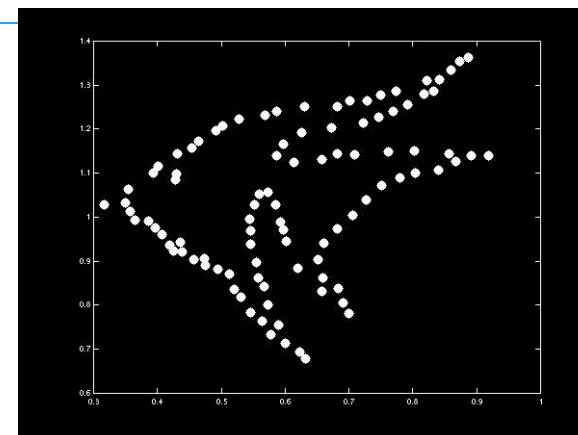


$$f(x, y) = a_1 + a_x x + a_y y + \sum_{i=1}^n w_i U(\|(x_i, y_i) - (x, y)\|)$$

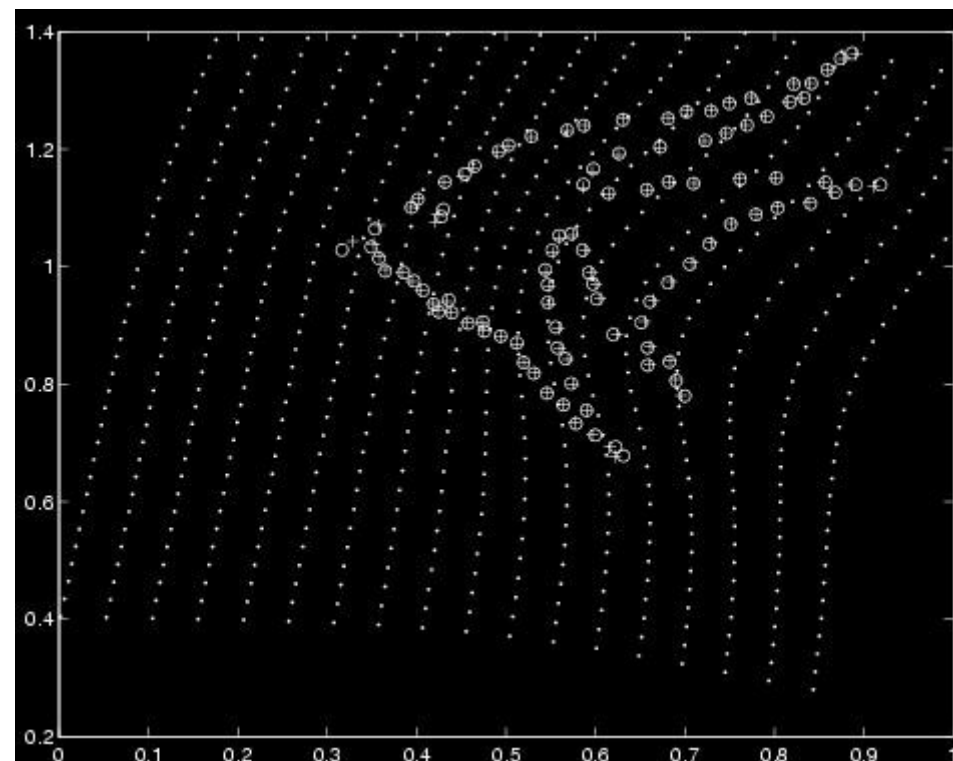
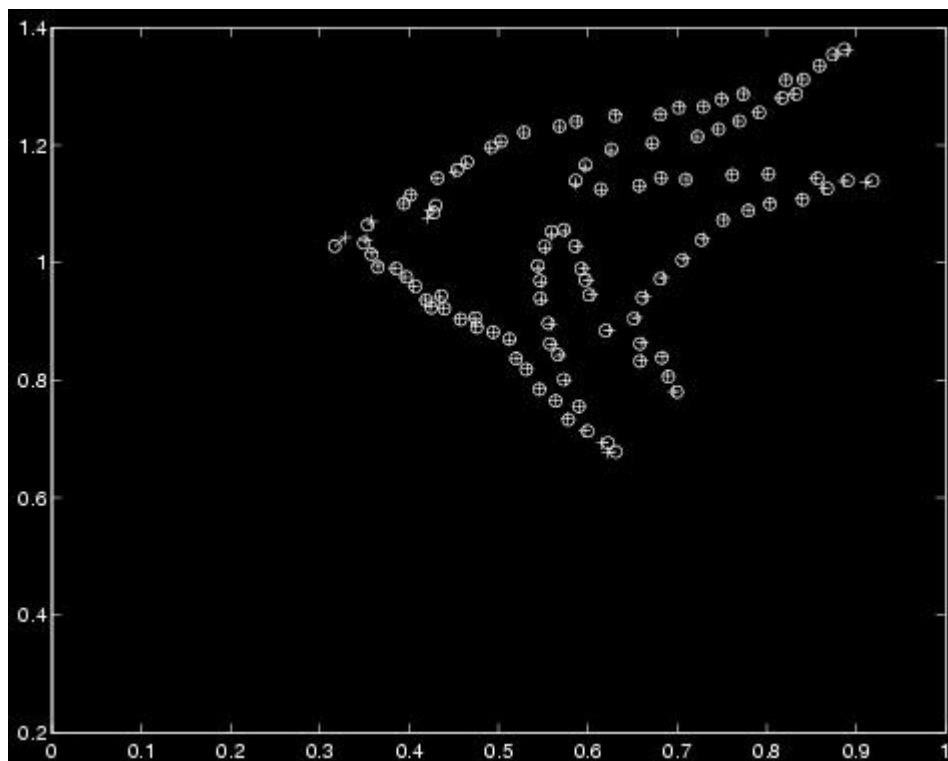
# Matching Example



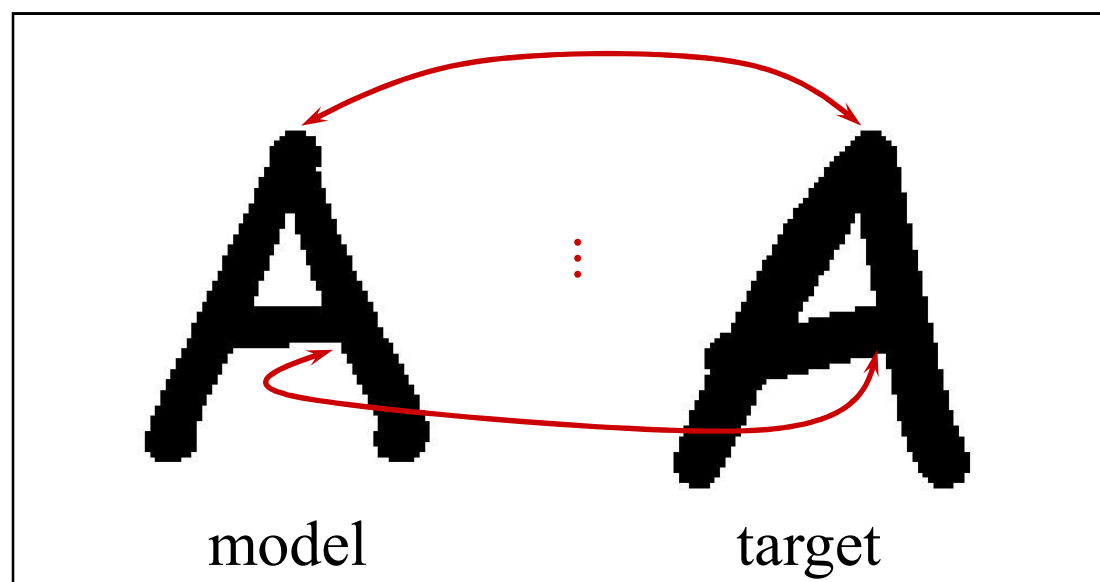
model



target



# Matching Framework



- Find correspondences between points on shape
- Estimate transformation
- Measure similarity

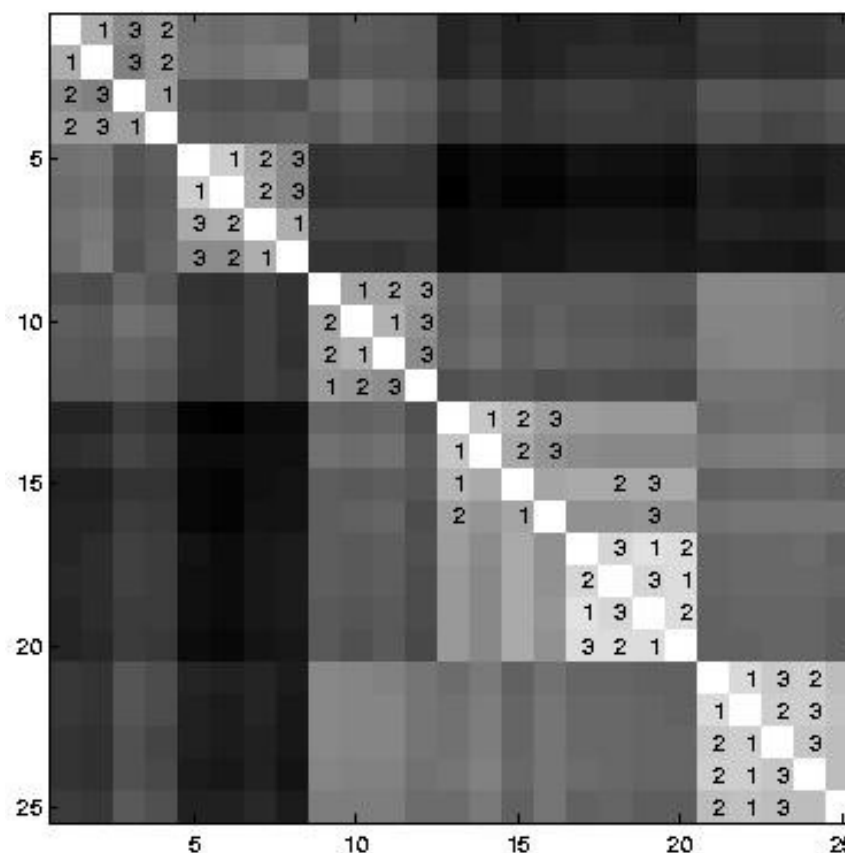
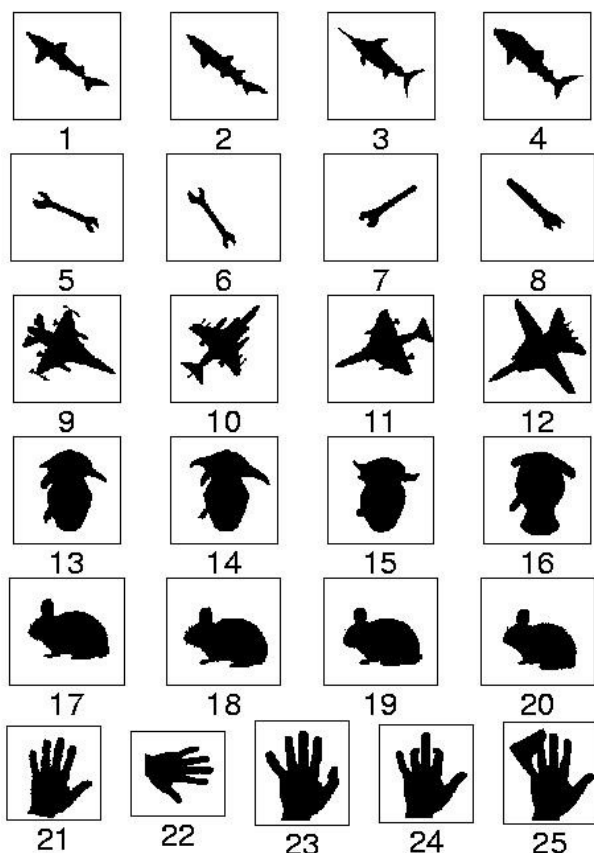


# Terms in Similarity Score

---























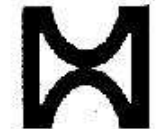
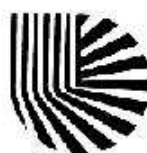








- ☐ Shape Context difference
- ☐ Local Image appearance difference
  - Orientation
  - Gray-level correlation in Gaussian window
  - ... (many more possible)
- ☐ Bending energy

# Application



S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts”, IEEE TPAMI 2002.

# Trademark retrieval

							
query	1: 0.086	2: 0.108	3: 0.109	query	1: 0.117	2: 0.121	3: 0.129
							
query	1: 0.066	2: 0.073	3: 0.077	query	1: 0.096	2: 0.147	3: 0.153
							
query	1: 0.046	2: 0.107	3: 0.114	query	1: 0.078	2: 0.116	3: 0.122
							
query	1: 0.046	2: 0.107	3: 0.114	query	1: 0.092	2: 0.10	3: 0.102