第六章：图像识别—目标检测

中国科学技术大学
电子工程与信息科学系

主讲教师：李厚强 (lihq@ustc.edu.cn)
周文罡 (zhwg@ustc.edu.cn)

助教：谢乔康 (xieqiaok@mail.ustc.edu.cn)
周　浩 (zhouh156@mail.ustc.edu.cn)

目标检测

- 简单形状的检测
- 人脸识别与检测
- 目标检测

目标检测

- 简单形状的检测
- 人脸识别与检测
- 目标检测

简单形状的检测

- Hough Transform
- Distance Transform

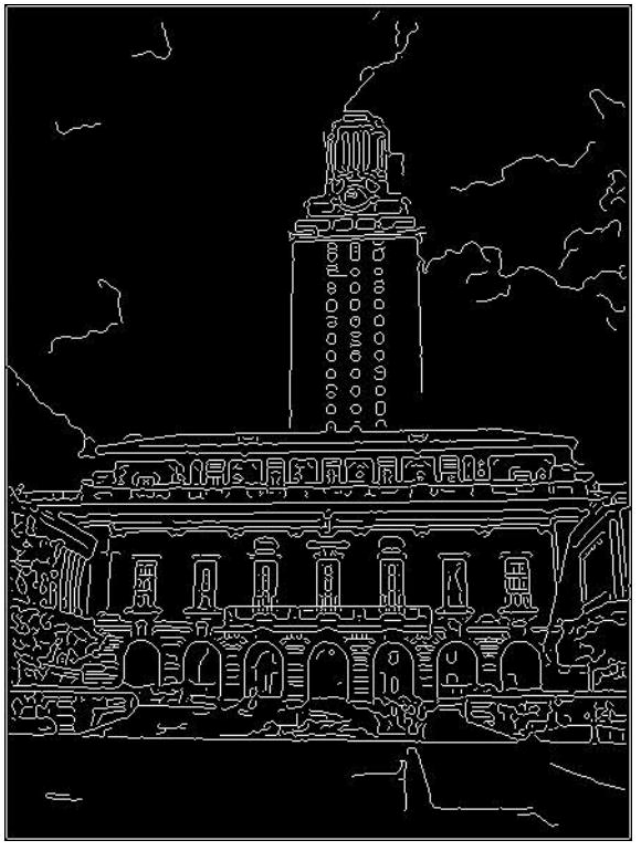
Hough Transform

- 应用场景：直线拟合

如何从边缘检测的结果得到直线拟合的结果？

Hough Transform

- 直线拟合的难点
  - 边缘检测点杂乱且多余
  - 不同的检测点属于不同的直线
  - 部分线段可能漏检
  - 边缘检测点上存在噪声



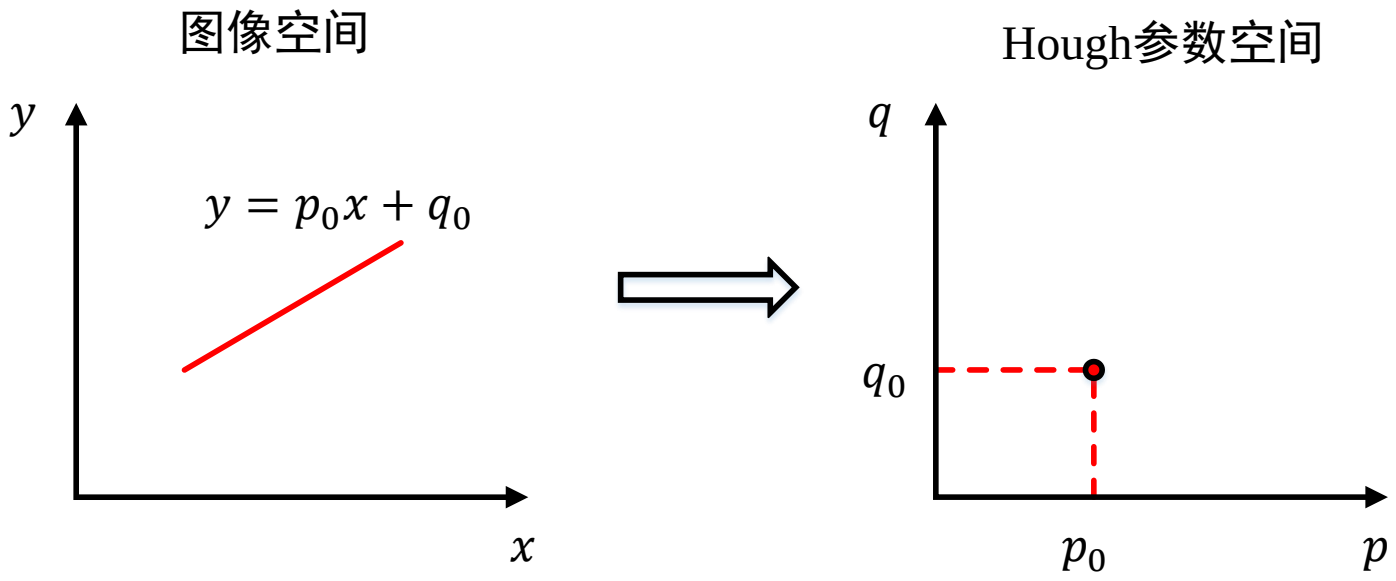
Hough Transform

- Voting schemes
  - Let each feature vote for all the models that are compatible with it
  - Hopefully the noise features will not vote consistently for any single model
  - Missing data doesn't matter as long as there are enough features remaining to agree on a good model

Hough Transform基本原理

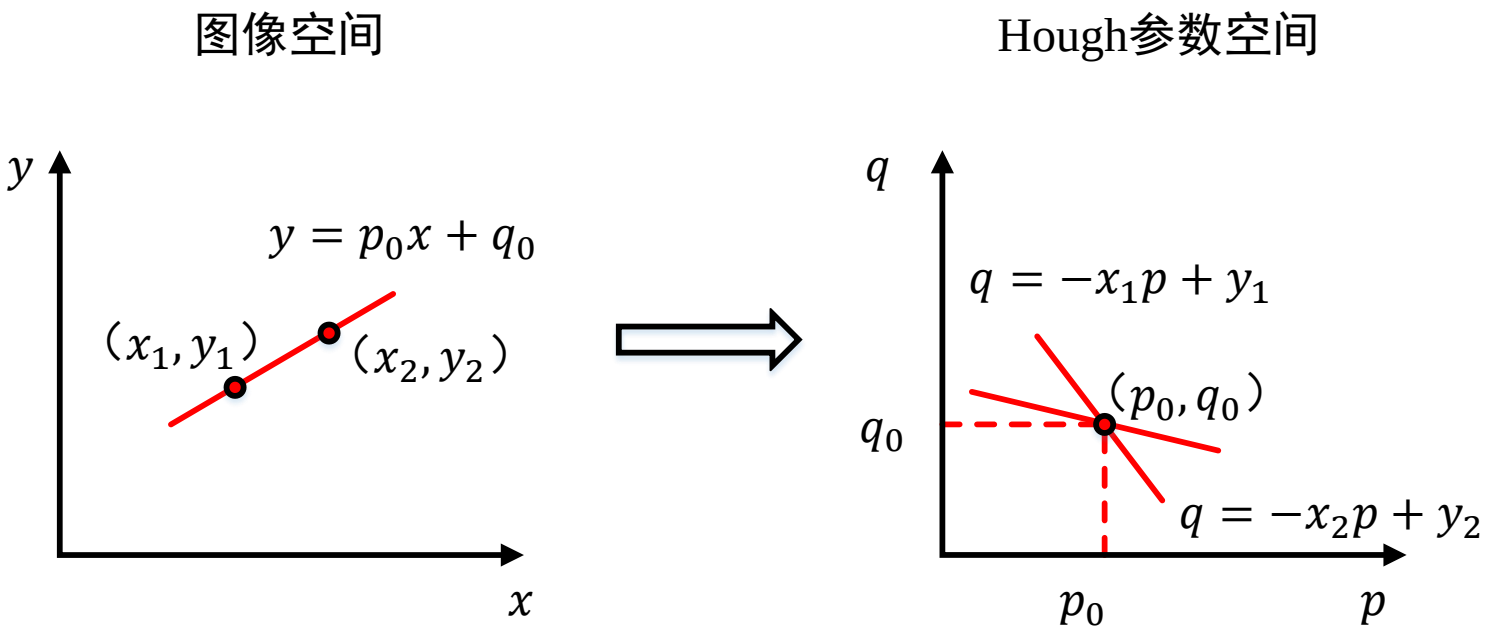
- Hough Transform
  - 图像空间与参数空间之间的一种变换

- Hough参数空间
  - 在图像空间中的一条直线，对应于Hough参数空间中的一个点



直线检测

- 图像空间中同一条直线上的任意两个点，在参数空间中对应于两条相交的直线



直线检测

- 具体方法
  - 将参数空间离散成一个2-D的累加数组$A(p, q)$

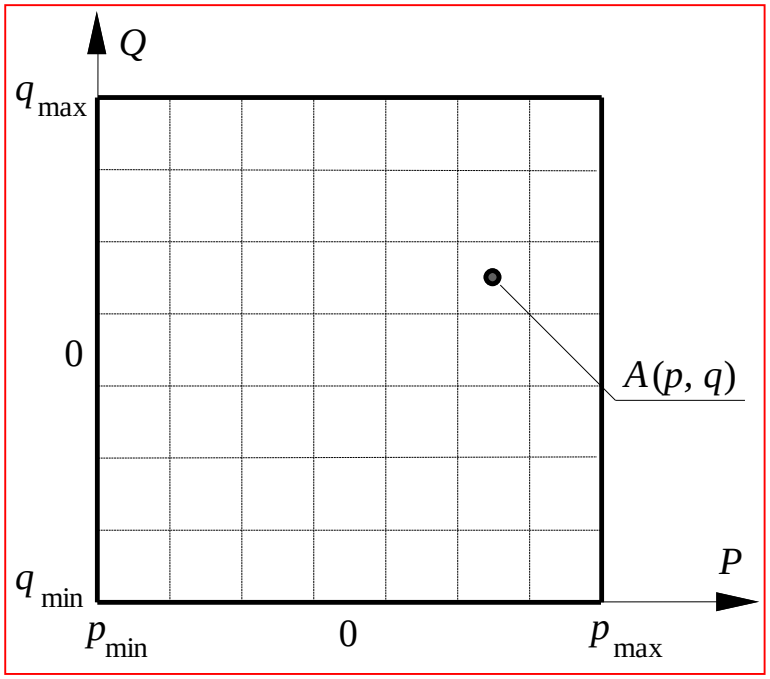
    $p \in [p_{\min}, p_{\max}]$

    $q \in [q_{\min}, q_{\max}]$

  - A(p, q) = A(p, q) + 1

    A(p, q)：共线点数

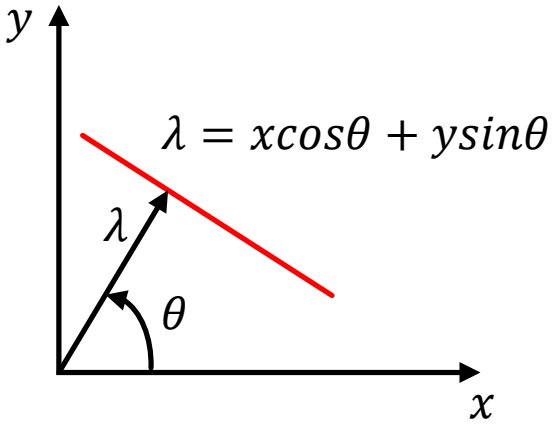
    (p, q)：直线方程参数

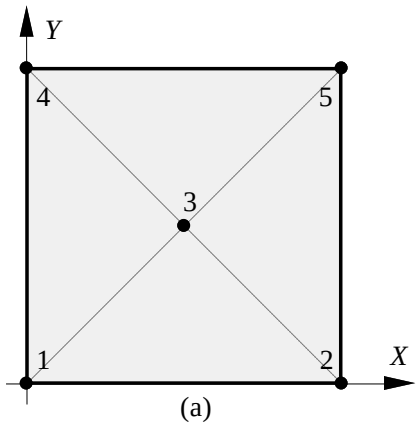


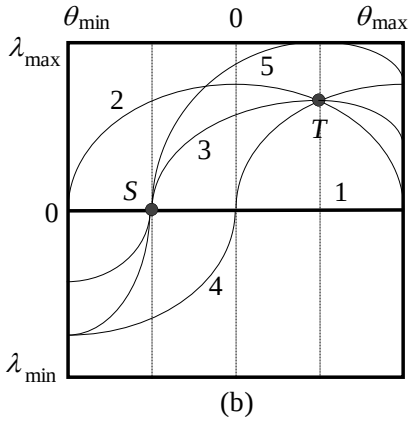
- 潜在的问题
  - $p_{\max}$/ $q_{\max}$为无穷大

直线检测的改进形式

- 直线的极坐标方程
  - $\lambda = xcos\theta + ysin\theta$
  - 参数λ和θ唯一确定一条直线







- X-Y平面的一个点对应参数空间的一条正弦曲线
  - $\lambda = x_0 cos\theta + y_0 sin\theta \leftrightarrow \lambda = Asin(\theta + \alpha)$

$$\text{其中}\alpha = \tan^{-1}\left(\frac{x_0}{y_0}\right), A = \sqrt{x_0^2 + y_0^2}$$

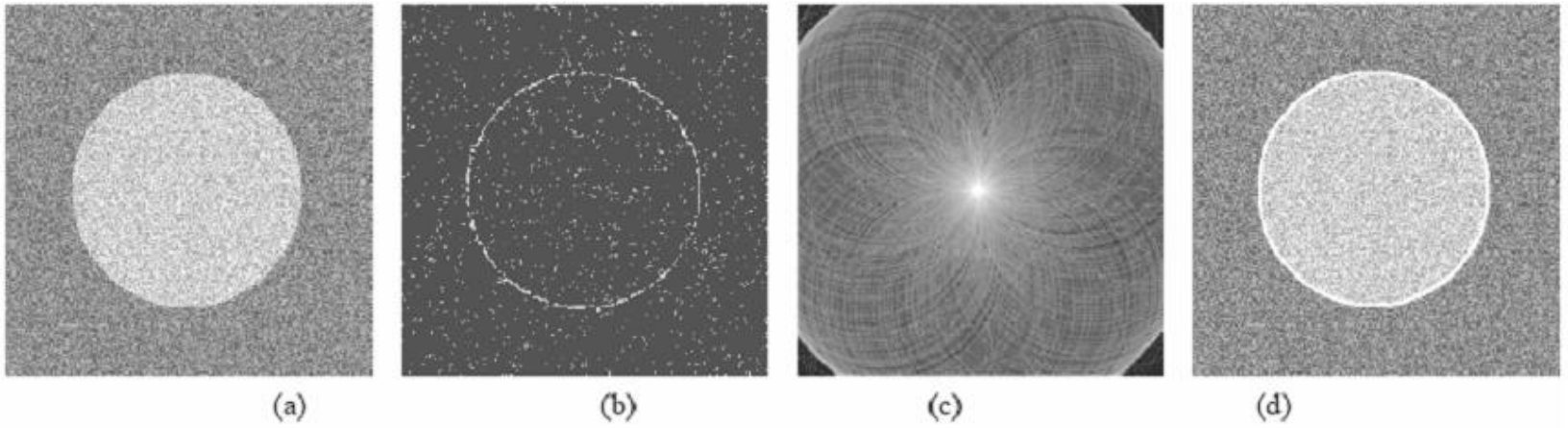
其他形状的检测

- Hough Transform可以检测满足解析式 $f(x, c) = 0$形式的各类曲线并把曲线上的点连接起来

- 示例：以圆周检测为例
  - 圆周方程：$(\mathrm{x} - a)^2 + (y - b)^2 = r^2$
  - 三个参数a, b, r，所以需要在参数空间中建立3-D累加数组，其中的元素可以记为$A(a, b, r)$

圆周检测

示例



图(a)为256x256，灰度256级，叠加随机噪声；

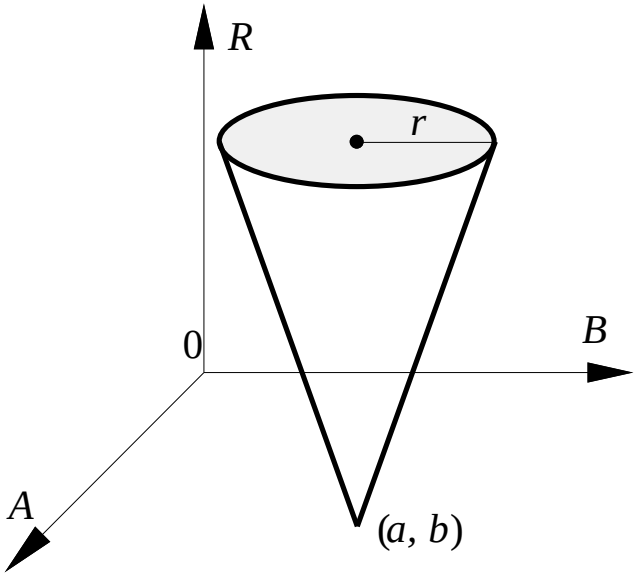
图(b)为求梯度(Sobel算子)取阈值后的结果；

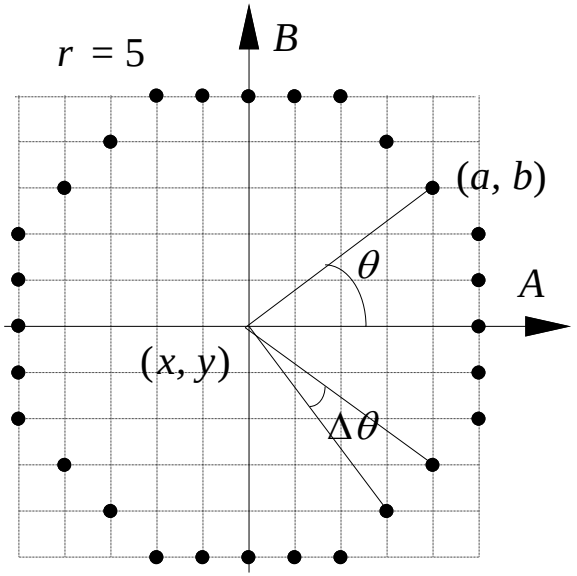
图(c)哈夫变换累计器图；

图(d)为检测出的圆周附加在原图上的效果

圆周检测

- 利用梯度降维
  - 使累加数组的维度减少一维
  - 圆周——圆周对偶性





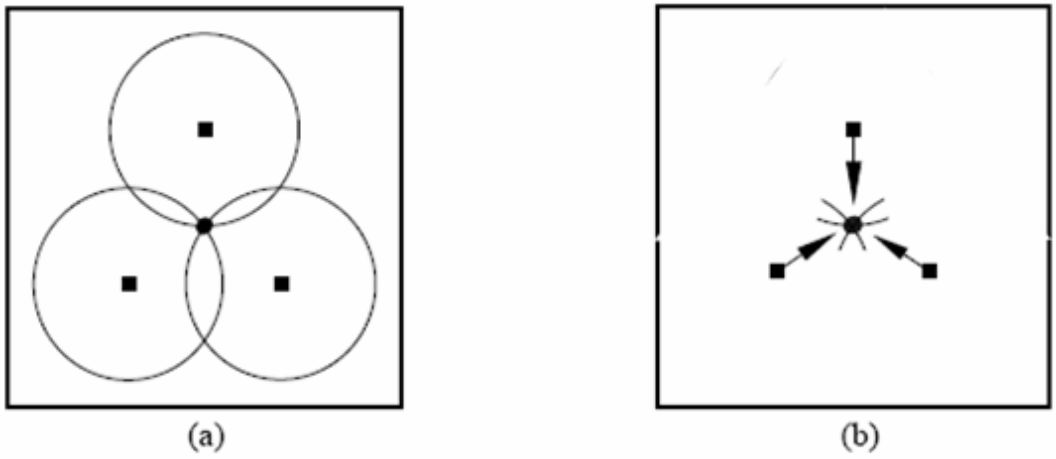
圆周检测

- 利用梯度降维

$$a = x - r\sin\theta$$

$$b = y + r\cos\theta$$

1个3-D累加器数组→2个2-D累加器数组



利用梯度与否 2 种情况下的累加数组示意

利用梯度信息检测椭圆

- 椭圆方程

$$\frac{(x-p)^2}{a^2}+\frac{(y-q)^2}{b^2}=1$$

- 对x求导

$$\frac{(x-p)}{a^2}+\frac{(y-q)}{b^2}\tan\theta=0$$

- 联立得到

$$p=x\pm\frac{a^2\tan\theta}{\sqrt{a^2\tan^2\theta+b^2}} \qquad q=y\pm\frac{b^2}{\sqrt{a^2\tan^2\theta+b^2}}$$

建立2个3-D累加数组$A_x(p,a,b)$和$A_y(p,a,b)$

广义哈夫变换

- 广义Hough变换将一般的模板匹配与Hough变换相结合
- 先对模板与图象上的物点作坐标变换，然后求相关
- 并用类似Hough变换检测物体的表决方法来确定匹配点

广义哈夫变换原理

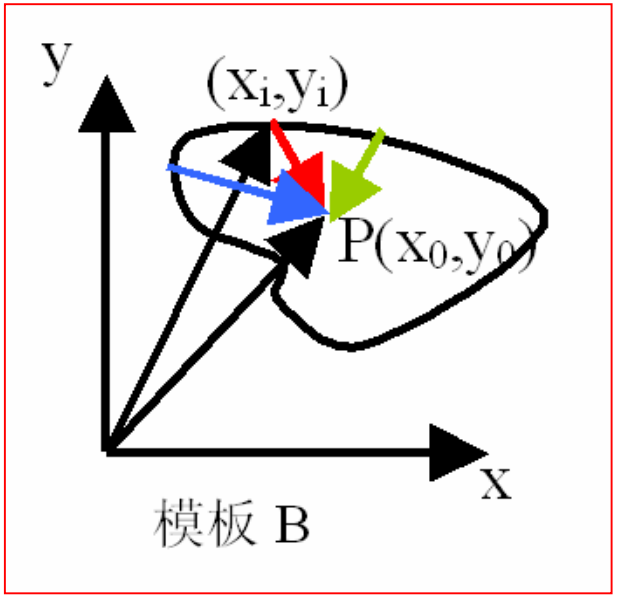
- B为模板物体的一组点集
- P(x0,y0)为一参考点，常把P取为B的中心点。
- 广义Hough变换是一组矢量的集合：B
  - $B = \{(dx_i, dy_i), I = 1, 2, \cdots, m\}$

$$H(B, P)$$

$$\{(dx_i, dy_i), i = 1, 2, \cdots, n\}$$

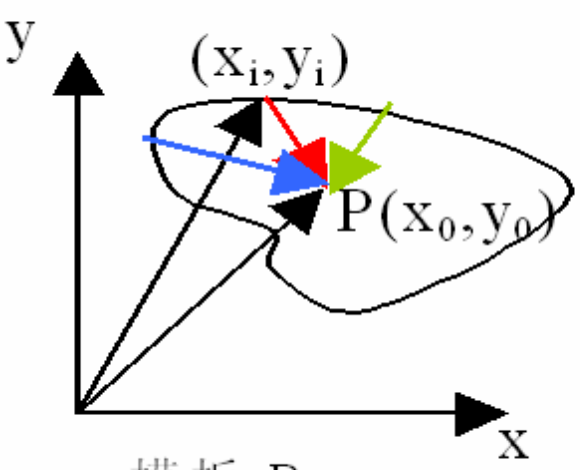
$$dx_i = x_0 - x_i = -(x_i - x_0)$$

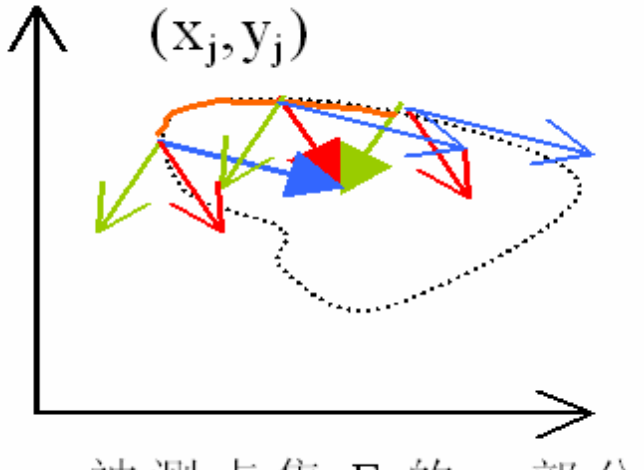
$$dy_i = y_0 - y_i = -(y_i - y_0)$$



广义哈夫变换原理

- 检测时，将待测图象，记为点集E：
  - $E = \{(x_j, y_j), j = 1, 2, \cdots, n\}$
- 将待检测点集与变换后的模板B做相关运算
  - $\forall i, j$,计算$v(i,j) = (dx_i + x_j, dy_i + y_j)$,
  - 如果有很多点$v(i,j)$都对应同一个坐标点，则该点为与B相匹配的形状的中心位置



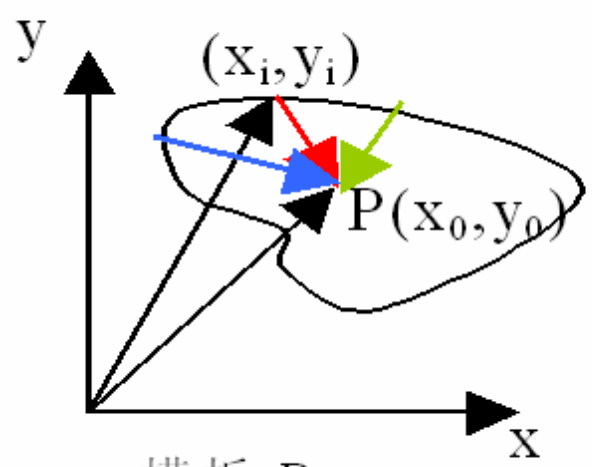


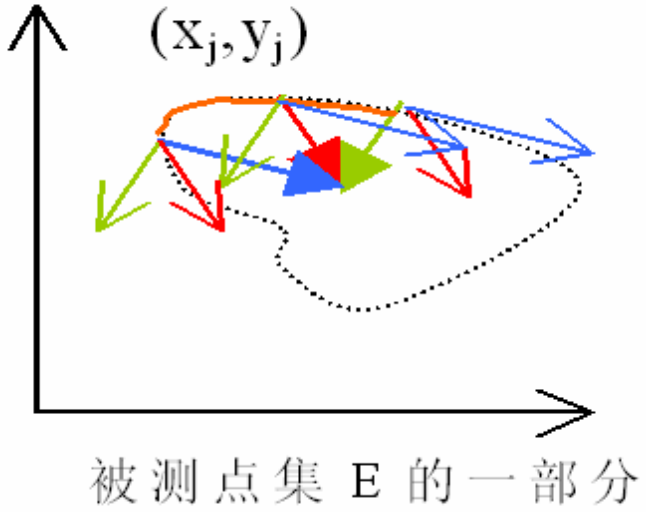
广义哈夫变换原理

检测时，将待测图象，记为点集E：

$$E = \{(x_j, y_j), j = 1, 2, \cdots, m\}$$

将待测的点集E与变换后的模板B作相关运算





广义哈夫变换原理

- 在所需检测的曲线或目标轮廓没有或不易用解析式表达时，可以利用表格来建立曲线或轮廓点与参考点间的关系，从而可继续利用哈夫变换进行检测

建立参考点与轮廓点的联系：

$$p = x + r(\theta) \cdot \cos(\phi)$$
$$q = y + r(\theta) \cdot \sin(\phi)$$

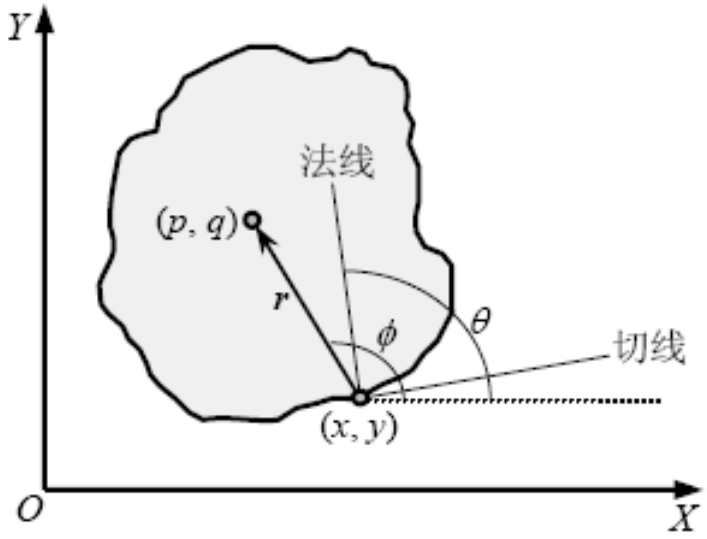


图 6.1.8　建立参考点和轮廓点的对应关系

广义哈夫变换原理

- 已知轮廓形状、朝向和尺度而只需检测位置信息
- 根据θ, r和ϕ 的函数关系作出参考表

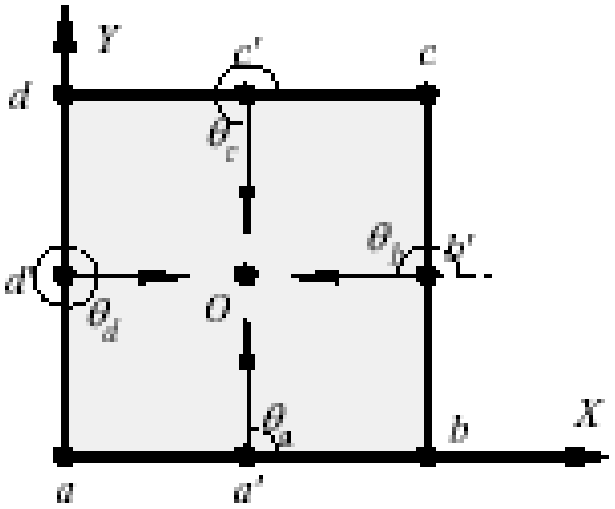
梯度角θ	矢径$r(\theta)$	矢角$\phi(\theta)$
θ_1	$r_1^1,\ r_1^2,\ \cdots,\ r_1^{N_1}$	$\phi_1^1,\ \phi_1^2,\ \cdots,\ \phi_1^{N_1}$
θ_2	$r_2^1,\ r_2^2,\ \cdots,\ r_2^{N_2}$	$\phi_2^1,\ \phi_2^2,\ \cdots,\ \phi_2^{N_2}$
$\cdots$	$\cdots$	$\cdots$
θ_M	$r_M^1,\ r_M^2,\ \cdots,\ r_M^{N_M}$	$\phi_M^1,\ \phi_M^2,\ \cdots,\ \phi_M^{N_M}$

- 给定一个测试点(x', y') 及其梯度角θ'，即可确定一组可能的参考点位置
  - 根据 梯度角θ'，找到表中对应梯度角所在行的矢径和视角序列
  - 基于坐标(x', y')和序列中每一组(r, ϕ)值，反推出形状参考点坐标

$$p' = x' + r(\theta) \cdot \cos(\phi), q' = y' + r(\theta) \cdot \sin(\phi)$$

广义哈夫变换原理

轮廓点	a	a'	b	b'	C	c'	d	d'
矢径 $r(\theta)$	$\sqrt{2}/2$	1/2	$\sqrt{2}/2$	1/2	$\sqrt{2}/2$	1/2	$\sqrt{2}/2$	1/2
矢角 $\phi(\theta)$	1π/4	2π/4	3π/4	4π/4	5π/4	6π/4	7π/4	8π/4

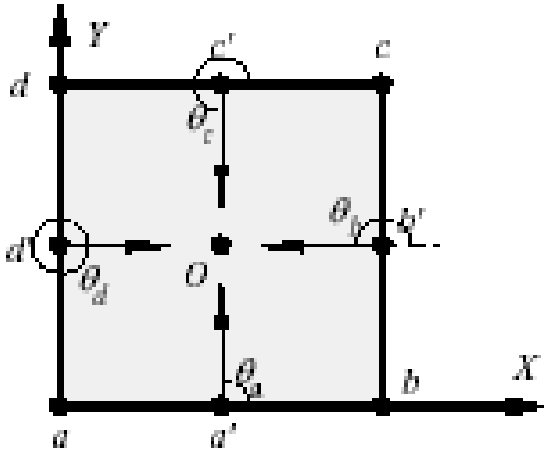


梯度角 θ	矢径 $r(\theta)$		矢角 $\phi(\theta)$	
$\theta_a = \pi/2$	$\sqrt{2}/2$	1/2	π/4	2π/4
$\theta_b = 2\pi/2$	$\sqrt{2}/2$	1/2	3π/4	4π/4
$\theta_c = 3\pi/2$	$\sqrt{2}/2$	1/2	5π/4	6π/4
$\theta_d = 4\pi/2$	$\sqrt{2}/2$	1/2	7π/4	8π/4

广义哈夫变换原理

- 利用正方形上的8个轮廓点判断可能参考点位置(p', q')
- 对每个θ有 2个 r 及2个 ϕ 与之对应

$$p' = x' + r(\theta) \cdot \cos(\phi)$$
$$q' = y' + r(\theta) \cdot \sin(\phi)$$



梯度角	轮廓点	可能参考点		轮廓点	可能参考点	
θ_a	a	O	d'	a'	b'	O
θ_b	b	O	a'	b'	c'	O
θ_c	c	O	b'	c'	d'	O
θ_d	d	O	c'	d'	a'	O

点O出现频率最高

广义Hough变换的性能

- 运算量较小
- 抗干扰性也较强
- 可以求出曲线的某些参数
- 可适用于不规则曲线
- 仍不具有不变性

完整广义哈夫变换

- 轮廓的平移 + 轮廓放缩、旋转
- 累加数组：
- $A(p_{min}: p_{max}, q_{min}: q_{max}, \beta_{min}: \beta_{max}, S_{min}: S_{max})$

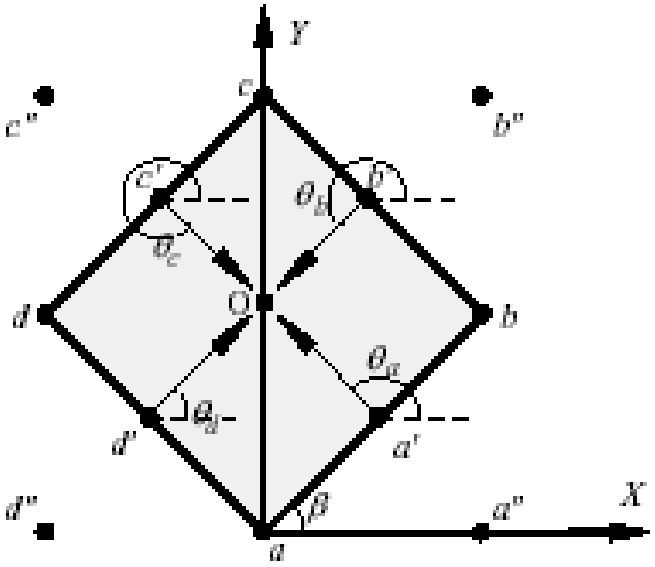
$$p = x + S \times r(\theta) \times \cos[\phi(\theta) + \beta]$$

$$q = y + S \times r(\theta) \times \sin[\phi(\theta) + \beta]$$

- 累加数组的累加：$A(p, q, \beta, S) = A(p, q, \beta, S) + 1$

完整广义哈夫变换

计算示例



原梯度角 θ	新梯度角 θ'	矢径 $r(\theta)$		新矢角 $\phi(\theta)$	
$\theta_a = \pi/2$	$\theta'_a = 3\pi/4$	$\sqrt{2}/2$	1/2	$2\pi/4$	$3\pi/4$
$\theta_b = 2\pi/2$	$\theta'_b = 5\pi/4$	$\sqrt{2}/2$	1/2	$4\pi/4$	$5\pi/4$
$\theta_c = 3\pi/2$	$\theta'_c = 7\pi/4$	$\sqrt{2}/2$	1/2	$6\pi/4$	$7\pi/4$
$\theta_d = 4\pi/2$	$\theta'_d = \pi/4$	$\sqrt{2}/2$	1/2	$8\pi/4$	$1\pi/4$

梯度角	轮廓点	可能参考点		轮廓点	可能参考点	
θ'_a	a	O	d'	a'	b'	O
θ'_b	b	O	a'	b'	c'	O
θ'_c	c	O	b'	c'	d'	O
θ'_d	d	O	c'	d'	a'	O

简单形状的检测

- Hough Transform
- Chamfer Distance

Distance Transform

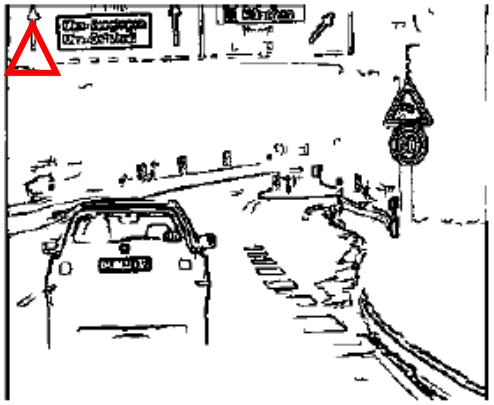
- 应用场景：形状匹配与物体识别

Chamfer Distance

- Average distance to nearest feature

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

  - T是模板形状（数据点的集合）
  - I 是搜索的图片（数据点的集合）
  - $\mathrm{d}_I(t)$是模板中的点t到搜索图片中的点的最小的距离



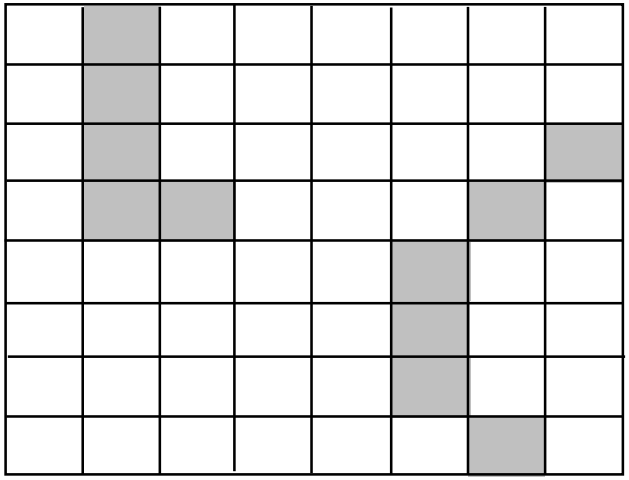
Chamfer Distance

- 定义

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

上面所指的feature可以是边缘点，前景点等等有区分性的点

Image features (2D)



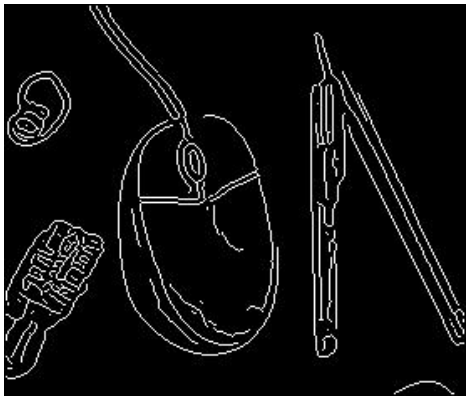
Distance Transform

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

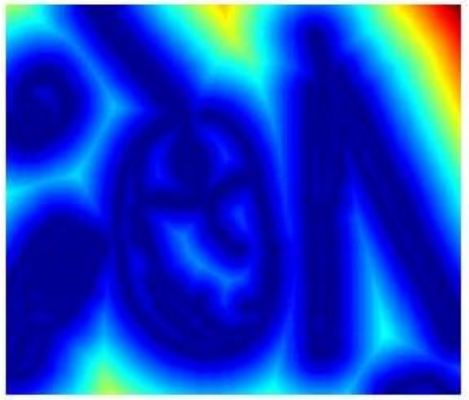
Chamfer Distance

- 距离变换实例
  - 在距离变换的结果中，每个位置的值表示这个位置到最近的边缘点（或者其他二值化的图片结构）的距离

original



edges



distance transform

Chamfer Distance

- 1-D距离变换
  - 1-D L_1范数的距离变换是一个计算复杂度为O(n)的算法
  - 算法步骤
    1. 对图中任意位置的值进行初始化，当j在特征P中时初始化为0，否则初始化为inf。将第 j 个位置的值记为$D[j]$
    2. Forward pass: for j from 1 up to n-1，更新$D[j]$

$$D[j] = \min(D[j], D[j-1] + 1)$$

    3. Backward pass: for j from n-2 down to 0，更新$D[j]$

$$D[j] = \min(D[j], D[j+1] + 1)$$

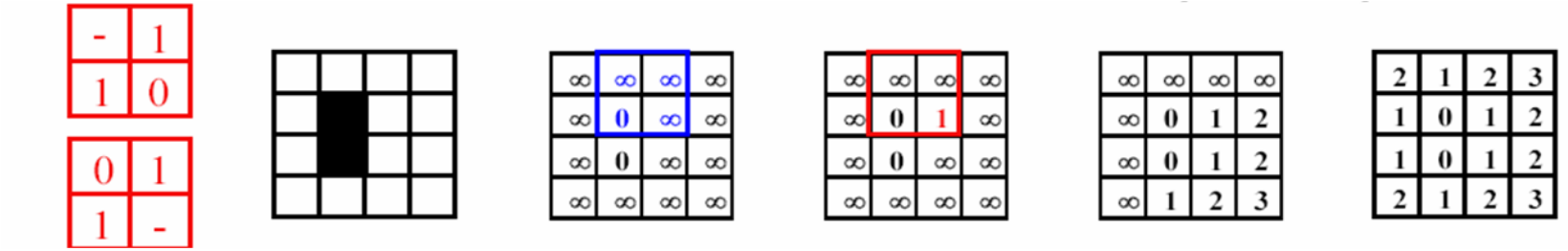
Chamfer Distance

- 2-D距离变换：算法步骤与1-D情况类似
  - 初始化距离矩阵
  - Forward pass从上方和左方找离特征点最近的距离

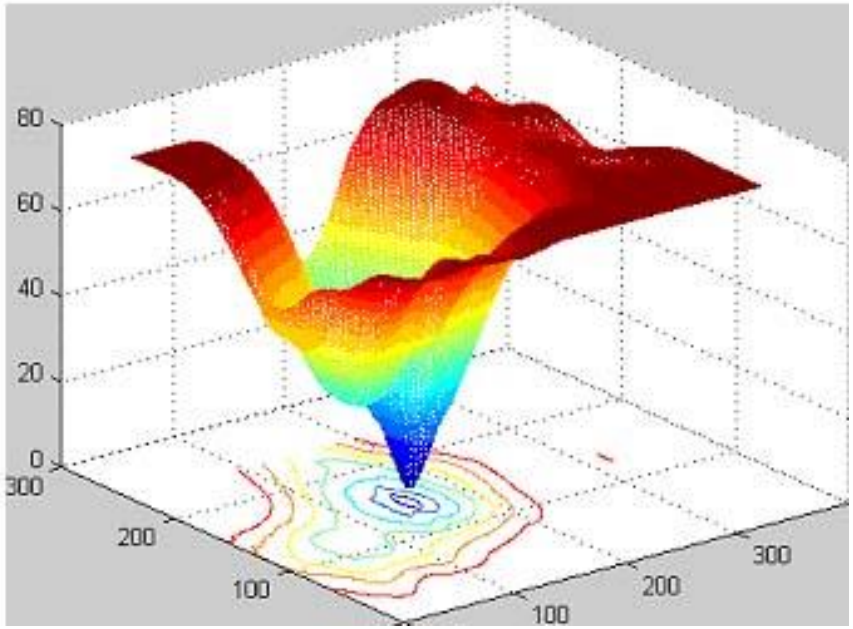
$$D[i,j] = \min(D[i,j], D[i,j-1]+1, D[i-1,j]+1)$$

  - Backward pass从下方和右方找离特征点最近的距离

$$D[i,j] = \min(D[i,j], D[i,j+1]+1, D[i+1,j]+1)$$

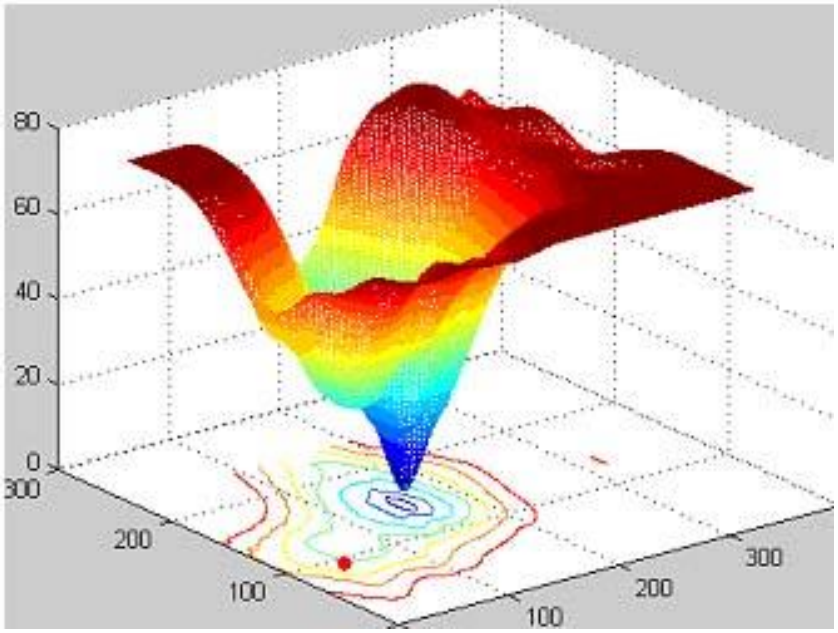


Chamfer Matching

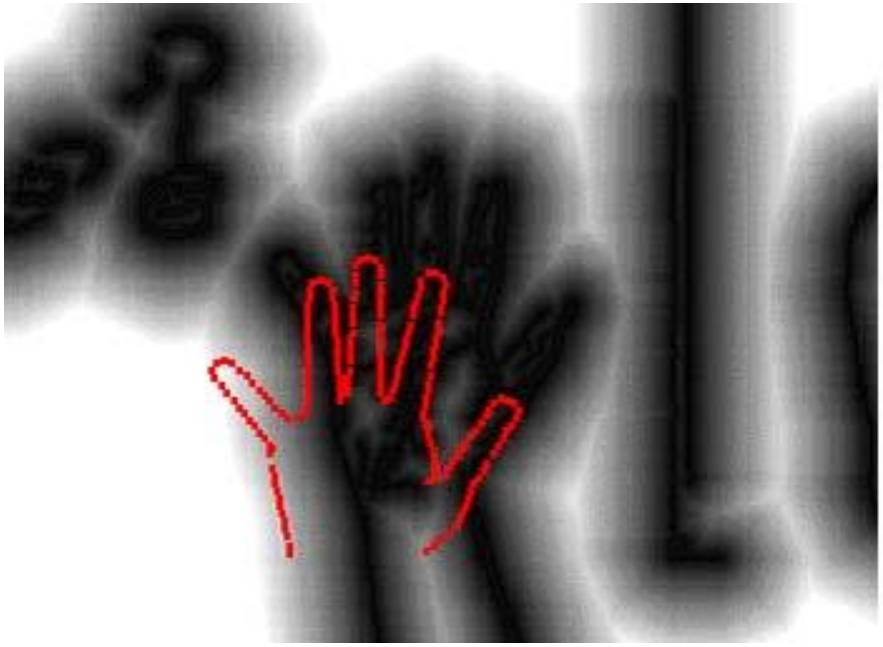


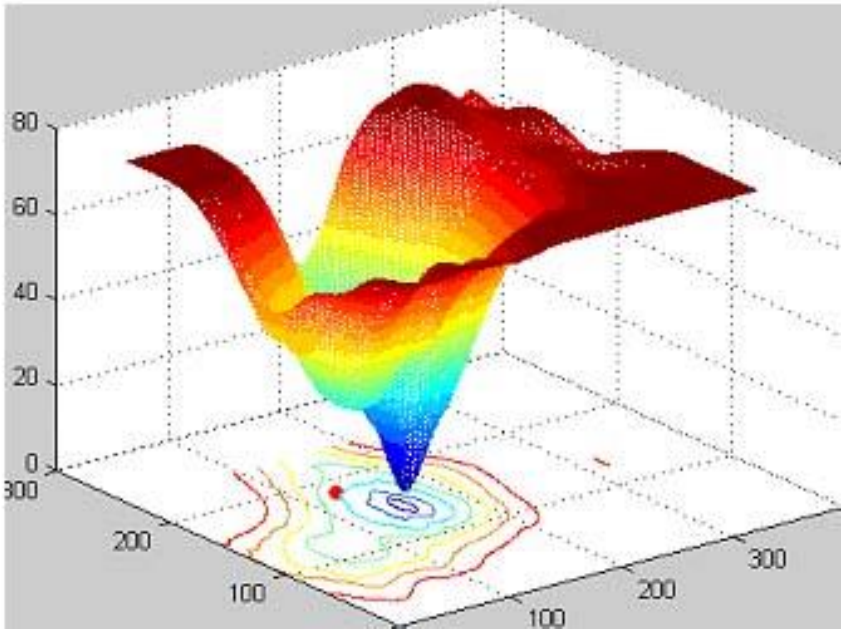
- Distance image provides a smooth cost function
- Efficient searching techniques can be used to find correct template

Chamfer Matching

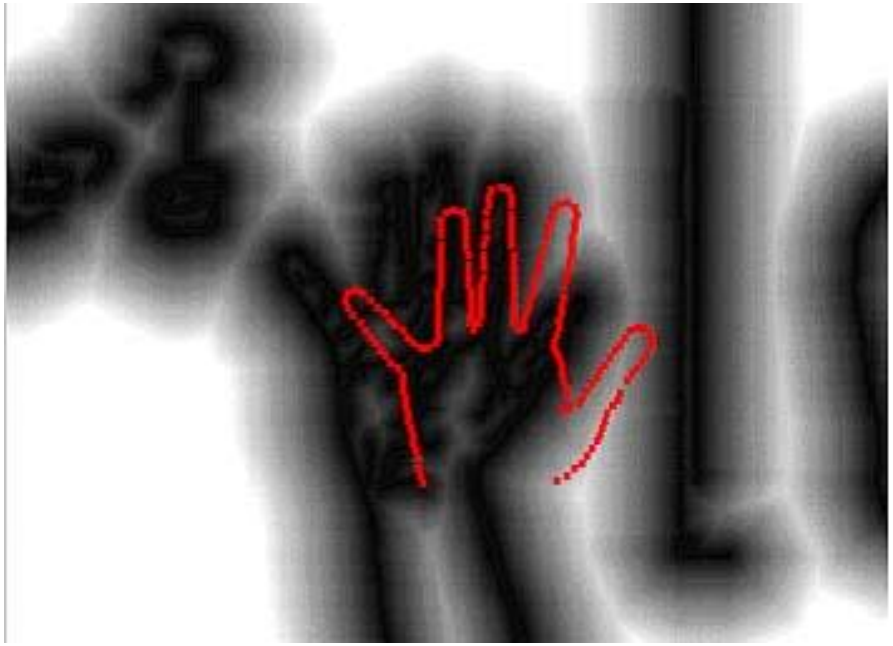


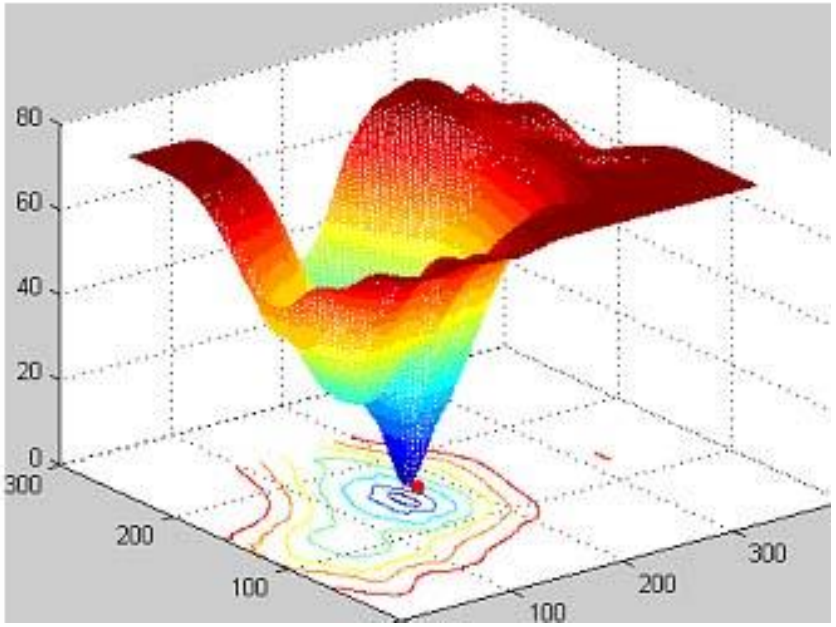
Chamfer Matching



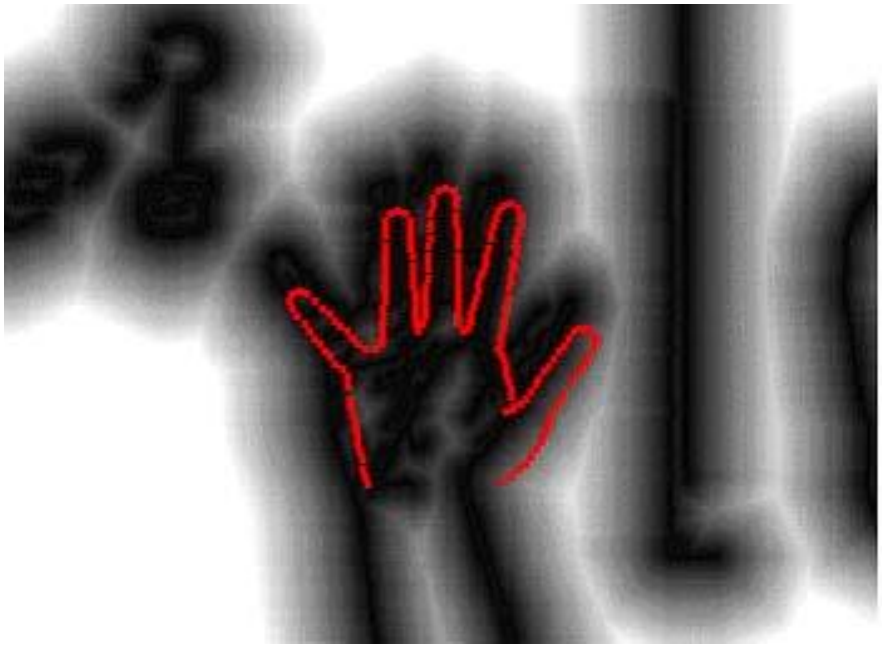


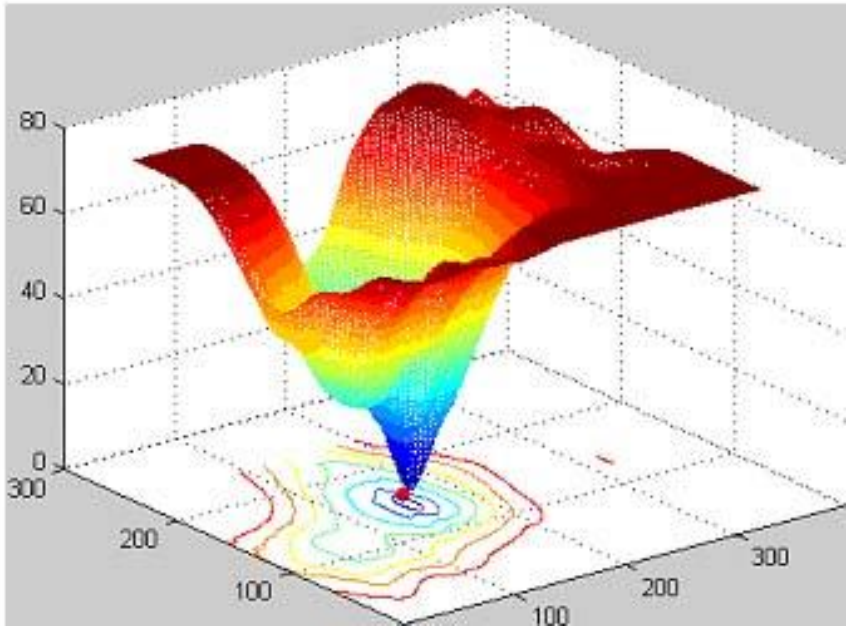
Chamfer Matching



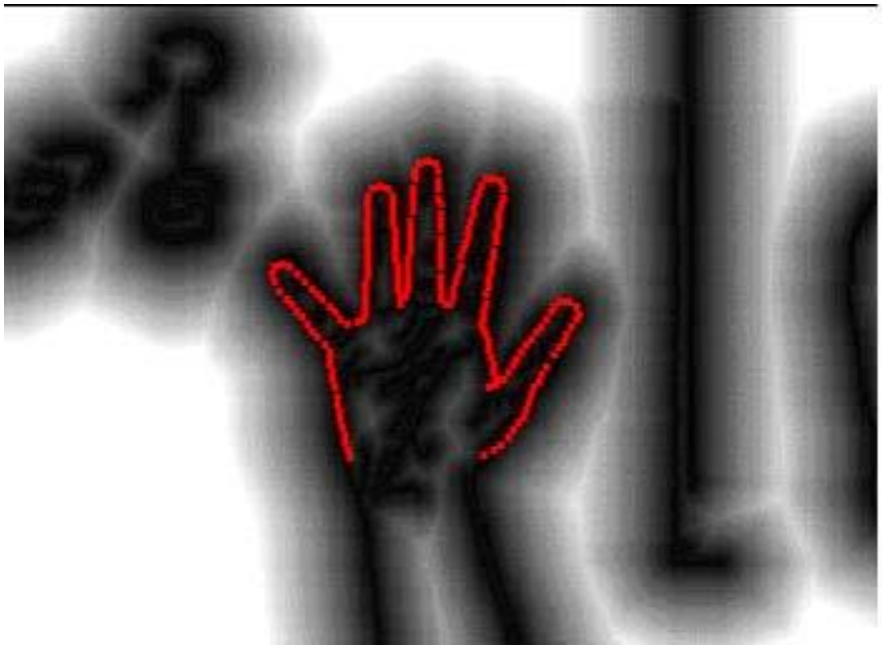


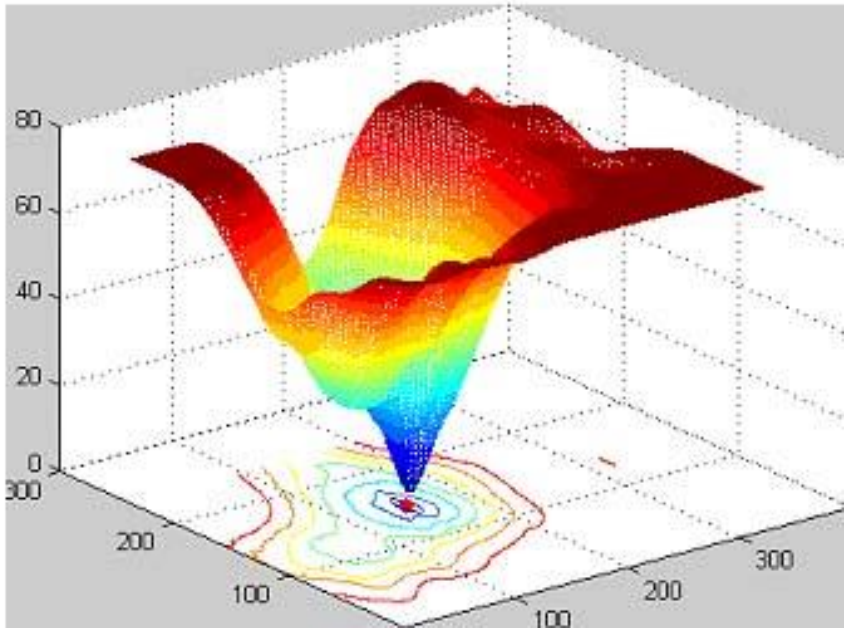
Chamfer Matching





Chamfer Matching





Chamfer Distance

- 优点
  - Robust to clutter
  - Computationally cheap
- 缺点
  - Variant to scale and rotation
  - Sensitive to small shape changes
  - Need large number of template shapes
- 改进方法
  - Multiscale matching
  - Hierarchical model organization

目标检测

- 简单形状的检测
- 人脸检测与识别
  - Viola & Jones Face Detector
  - Eigenface for Face Recognition
- 目标检测

Viola & Jones Detector

- 人脸检测算法面临的挑战
  - 基本思想：将人脸检测问题形式化为滑窗分类问题
  - 人脸可能出现在图中的任意区域，对于一张仅仅100万像素的图片，就需要分类超过100万个滑动窗口（sliding windows）
  - 在每张图片中，相比较于所有可能的位置，仅仅有非常少数量的人脸存在
  - 在实际应用中我们希望虚警的检测能尽可能的少
  - 为了检测速度尽可能的快，我们需要能快速的过滤掉大量不存在人脸的区域

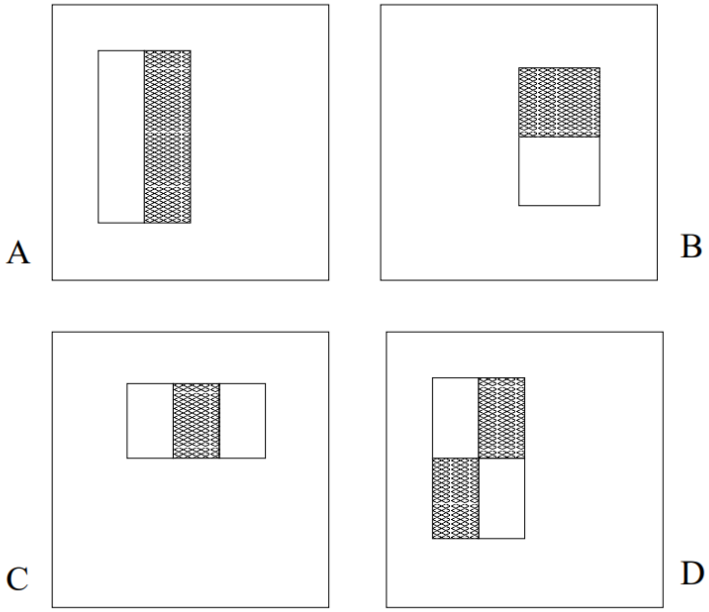
Viola & Jones Detector

- Viola & Jones Detector: 第一个实时的人脸检测算法
  - 弱可学习性等价于强可学习性：考虑一系列“基学习器”，让“后来者”重点关注“先行者”容易出错的部分，然后再将这些基学习器结合起来
  - 速度提升
    - ✓ 利用积分图像提取图像特征值，效率高
    - ✓ 利用adaboost分类器的特征筛选特性，保留最有用特征，减少了检测时的运算复杂度
  - 准确率提升
    - ✓ 将adaboost分类器进行改造，变成级联adaboost分类器，提高了人脸检测的准确率（降低漏检率和误检率）
  - 检测精度评估
    - ✓ 检测率：存在人脸并且被检测出的图像，在所有存在人脸图像中的比例。
    - ✓ 漏检率：存在人脸但是没有检测出的图像，在所有存在人脸图像中的比例。
    - ✓ 误检率：不存在人脸但是检测出存在人脸的图像，在所有不存在人脸图像中的比例。

- P.Viola and M. Jones. *Rapid object detection using a boosted cascade of simple features*. In CVPR 2001.

Viola & Jones Detector

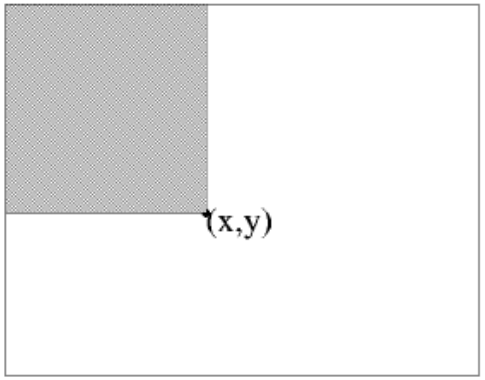
- Haar-like features
  - 与Haar小波类似的矩形状的特征，共四种基本形式

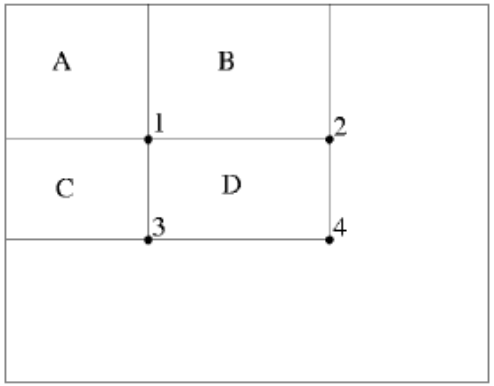


  - 特征的计算：
    - ✓ value = ∑(白色区域的像素值) – ∑(黑色区域的像素值)

Viola & Jones Detector

- 积分图 (Integral Images)
  - 积分图的计算： $I(x,y)=\sum_{i=0}^{x}\sum_{j=0}^{y}f(i,j)$





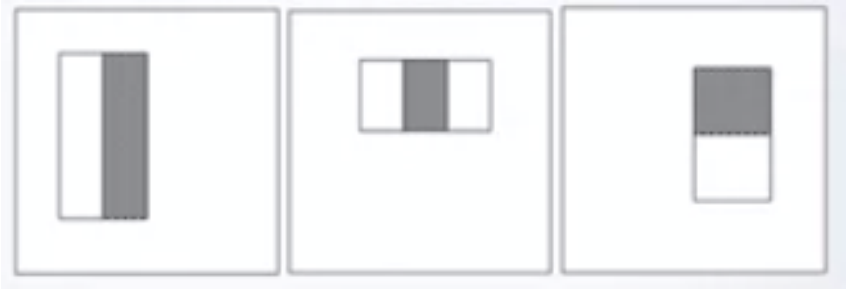
  - 以右图中D区域为例

$$\sum_{(i,j)\epsilon D} f(i,j)=I_1+I_4-(I_2+I_3)$$

  - 积分图的使用避免了重复的像素值求和计算，每个像素点只需计算一次求得各点的积分图即可求得不同区域的Haar-like特征

Viola & Jones Detector

- 面临的问题
  - 当使用一个24×24大小的sliding window时，可能得到的矩形状特征有大约160,000个。



  - 在测试时，我们不可能直接考虑整个特征集合进行检测。
- 通过AdaBoost选择有区分性的特征，并将对应的分类器级联组合起来
  - 让每一级的分类器，都具有非常高的检测率（99.9%,接近100%），同时误检率也保持相当高（大概50%，即FPR）
  - 例如，级联20个adaboost分类器，可实现检测率$(0.999)^{20} = 0.98$，同时误检率为$(0.5)^{20} = 9.5 \cdot 10^{-7}$

Viola & Jones Detector

- 基于AdaBoost的特征选择和分类器训练
  - 基于Haar-like特征，定义一系列弱分类器
  - 对于Adaboost算法的每一个循环：
    1. 初始化正负样本的权重
    2. 在每一个样本上评价每一个特征
    3. 为每个特征选取最好的阈值（分类器）
    4. 选取最好的特征和分类器进行组合
    5. 对样本的权重进行重新分配
  - 计算复杂度为O(TNK)
    - ✓ 其中T为循环次数，N为样本个数，K为特征个数

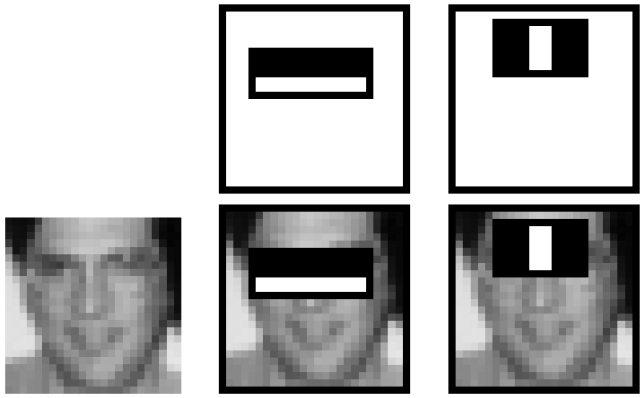
Viola & Jones Detector

- 详细步骤

- Given example images $(x_1, y_1), \ldots, (x_n, y_n)$ where $y_i = 0, 1$ for negative and positive examples respectively.
- Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for $y_i = 0, 1$ respectively, where m and l are the number of negatives and positives respectively.
- For $t = 1, \ldots, T$:
  1. Normalize the weights,
  $$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$$
  so that w_t is a probability distribution.
  2. For each feature, j, train a classifier h_j which is restricted to using a single feature. The error is evaluated with respect to w_t, $\epsilon_j = \sum_i w_i \left| h_j(x_i) - y_i \right|$.
  3. Choose the classifier, h_t, with the lowest error ϵ_t.
  4. Update the weights:
  $$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$$
  where $e_i = 0$ if example x_i is classified correctly, $e_i = 1$ otherwise, and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.
- The final strong classifier is:
$$h(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}$$
where $\alpha_t = \log \frac{1}{\beta_t}$

Viola & Jones Detector

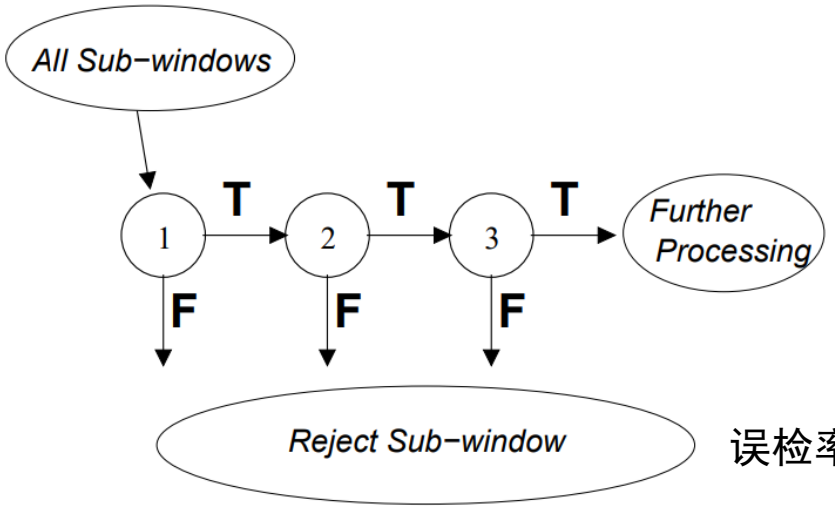
- 特征选择的结果
  - 可视化结果：下图是第一个和第二个被选中的特征



  - 定量的结果：由200个特征构成的正面人脸检测器检出率为95%，而14084个样本中出现1次的虚警。

Viola & Jones Detector

- Attentional cascade
  - We start with simple classifiers which reject many of the negative sub-windows while detecting almost all positive sub-windows
  - Positive response from the first classifier triggers the evaluation of a second (more complex) classifier, and so on
  - A negative outcome at any point leads to the immediate rejection of the sub-window



检测率：$D = \prod_{i=1}^{K} d_i$

误检率：$F = \prod_{i=1}^{K} f_i$,

Viola & Jones Detector

- 效果
  - 在测试时，平均每个window使用了10个特征
  - 在一个700Mhz的奔腾3处理器上，这个算法检测一张384×288的照片中的人脸需要大概0.067秒（15fps），比之前性能相当的方法快了15倍
- 局限性：适用于正脸的检测，对侧脸检测性能不佳

目标检测

- 简单形状的检测
- 人脸检测与识别
  - Viola & Jones Face Detector
  - Eigenface for Face Recognition
- 目标检测

Eigenface 数学基础

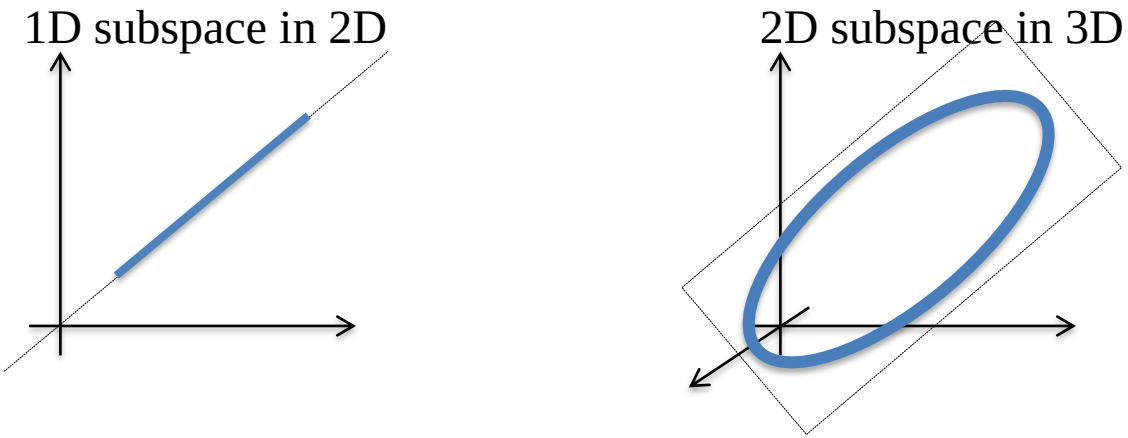
- 奇异值分解（SVD）
  - $A = \mathrm{U}\Sigma\mathrm{V}^{\mathrm{T}}$

$$\underset{U}{\begin{bmatrix}-.39 & -.92\\ -.92 & .39\end{bmatrix}} \times \underset{\Sigma}{\begin{bmatrix}9.51 & 0 & 0\\ 0 & .77 & 0\end{bmatrix}} \times \underset{V^T}{\begin{bmatrix}-.42 & -.57 & -.70\\ .81 & .11 & -.58\\ .41 & -.82 & .41\end{bmatrix}} = \underset{A}{\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\end{bmatrix}}$$

  - 特征值分解与奇异值分解的关系
    - ✓ 对AA^T做特征值分解，特征向量对应上式中U的列向量，特征值的平方根对应上式中的奇异值矩阵的对角元素Σ
    - ✓ 对A^TA做特征值分解，特征向量对应上式中V的列向量
  - 若A是对称阵，则有$A = \Phi\Sigma\Phi^T$，其中Φ为特征向量构成的矩阵

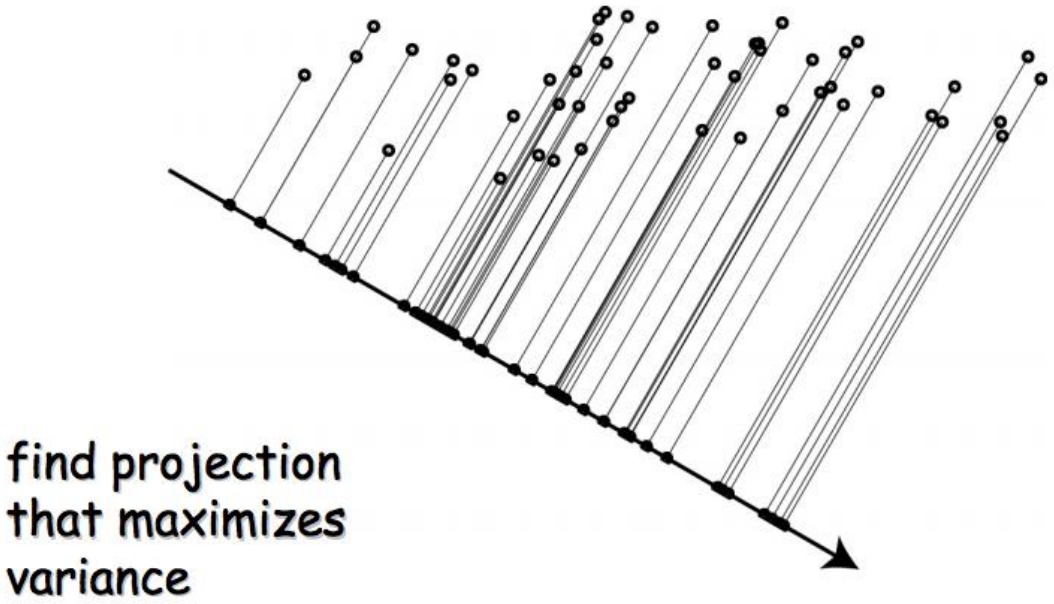
Eigenface 数学基础

- 主成分分析（PCA）基本思想
  - 在子空间中，物体的特征表达比在完整空间中更加扁平（压缩）



Eigenface 数学基础

- 数据降维时如何保留较多的信息
  - 信号处理中认为信号具有较大的方差，噪声有较小的方差



Eigenface 数学基础

- 假设样本x服从高斯分布的数据 $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$中采样得到，则有均值与方差矩阵

$$\boldsymbol{\mu} = \mathrm{E}[\mathbf{X}] = [\mathrm{E}[X_1], \mathrm{E}[X_2], \ldots, \mathrm{E}[X_k]]^{\mathrm{T}}$$

$$\boldsymbol{\Sigma} =: \mathrm{E}[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^{\mathrm{T}}] = [\mathrm{Cov}[X_i, X_j]; 1 \leq i, j \leq k]$$

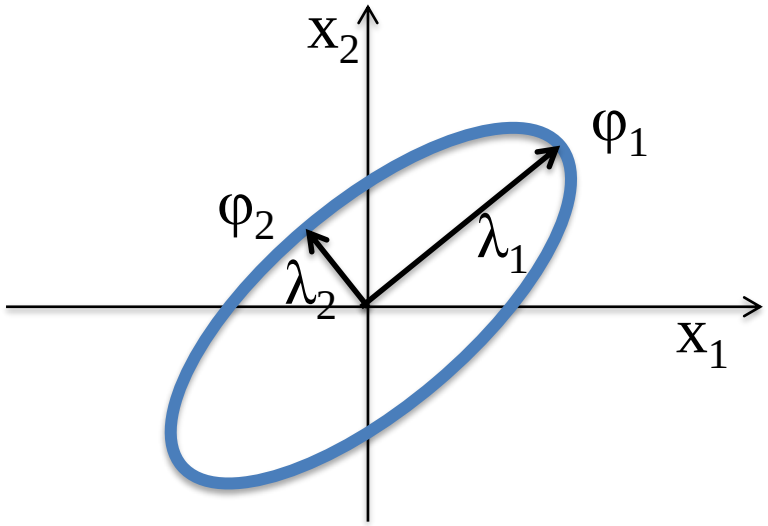
- 高斯分布的协方差矩阵是对称阵，可以表示分解为

$$\boldsymbol{\Sigma} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathrm{T}} = \mathbf{U}\boldsymbol{\Lambda}^{1/2}\left(\mathbf{U}\boldsymbol{\Lambda}^{1/2}\right)^{\mathrm{T}}$$

$$\begin{aligned}\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) &\iff \mathbf{X} \sim \boldsymbol{\mu} + \mathbf{U}\boldsymbol{\Lambda}^{1/2}\mathcal{N}(0, \mathbf{I}) \\ &\iff \mathbf{X} \sim \boldsymbol{\mu} + \mathbf{U}\mathcal{N}(0, \boldsymbol{\Lambda}).\end{aligned}$$

Eigenface 数学基础

- 假设$\mathrm{X}\sim N(\mu,\boldsymbol{\Sigma})$，其协方差矩阵为$\boldsymbol{\Sigma}$
  - 主成分ϕ_i是$\boldsymbol{\Sigma}$的特征向量
  - 主成分ϕ_i对应的长度是$\boldsymbol{\Sigma}$的特征值
  - λ_1与λ_2差别越大，ϕ_1方向上包含x的信息越多



Eigenface 数学基础

PCA

Given sample $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, $x_i \in \mathcal{R}^d$

- compute sample mean: $\hat{\mu} = \frac{1}{n}\sum_i(\mathbf{x}_i)$
- compute sample covariance: $\hat{\Sigma} = \frac{1}{n}\sum_i(\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^T$
- compute eigenvalues and eigenvectors of $\hat{\Sigma}$

$$\hat{\Sigma} = \Phi\Lambda\Phi^T, \ \Lambda = diag(\sigma_1^2, \ldots, \sigma_n^2) \ \ \Phi^T\Phi = I$$

- order eigenvalues $\sigma_1^2 > \ldots > \sigma_n^2$
- if, for a certain k, $\sigma_k << \sigma_1$ eliminate the eigenvalues and eigenvectors above k.

Eigenface

- 假设大多数人脸图片可以用一个低维子空间表示，这个子空间由具有最大方差的 k 个方向决定
- 使用PCA来得到用于描述人脸的张成子空间的向量（Eigenface）
- 将数据集中的人脸表达为Eigenface的线性组合

Eigenface

- 训练人脸数据集

Eigenface

- Eigenface可视化效果

Principal component (eigenvector) ϕ_k

$\mu + 3\sigma_k\phi_k$

$\mu - 3\sigma_k\phi_k$

Eigenface

- 训练流程
  1. 对齐训练图片，并且拉成一维向量$x_1, x_2, \ldots, x_N$

  2. 计算人脸图片的平均值 $m = \frac{1}{N}\sum x_i$

  3. 人脸图片减去均值（the centered data matrix）

$$X_c = \begin{bmatrix} | & & | \\ x_1 & \cdots & x_n \\ | & & | \end{bmatrix} - \begin{bmatrix} | & & | \\ \mu & \cdots & \mu \\ | & & | \end{bmatrix}$$

$$= X - \mu 1^T = X - \frac{1}{n} X 1 1^T = X\left(I - \frac{1}{n} 1 1^T\right)$$

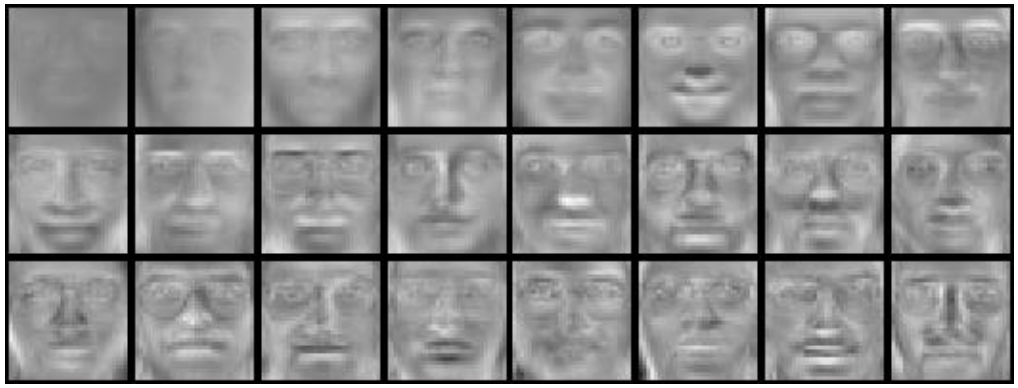
Eigenface

- 训练流程

4. 计算协方差矩阵

$$\Sigma = \frac{1}{n}\begin{bmatrix} | & & | \\ x_1^c & \cdots & x_n^c \\ | & & | \end{bmatrix}\begin{bmatrix} - & x_1^c & - \\ & \vdots & \\ - & x_n^c & - \end{bmatrix} = \frac{1}{n} X_c X_c^T$$

5. 计算协方差矩阵的特征向量，并且取对应特征值较大的前K个特征向量作为eigenface



Eigenface

- 训练流程
  6. 计算每张训练图片在eigenface上的投影

$$x_i \to \left(x_i^c \cdot f_1, x_i^c \cdot f_2, \ldots, x_i^c \cdot f_K\right) \equiv \left(a_1, a_2, \ldots, a_K\right)$$

  7. 训练图片的重建结果为 $x_i \approx m + a_1 f_1 + a_2 f_2 + \ldots + a_K f_K$ ，记录下投影权重 $(a_1, a_2, \ldots, a_K)$ 在测试时用于识别人脸

Eigenface

- 测试流程
  1. 把测试图片变成一维向量并减去前面计算得到的人脸平均值
  2. 将测试图片投影到eigenface的空间中，并且计算投影权重

$$t \rightarrow \left((t-\mu)\cdot\varphi_1,(t-\mu)\cdot\varphi_2,\cdots,(t-\mu)\cdot\varphi_K\right) \equiv \left(w_1,w_2,\cdots,w_K\right)$$

  3. 比较$(w_1,w_2,...,w_K)$与训练数据集中每个人脸的投影权重，与之投影权重差别最小的训练图片的ID即为该测试图片对应的ID

目标检测

- 简单形状的检测
- 人脸识别与检测
- 目标检测
  - 问题定义与评价标准
  - Sliding window + HOG features
  - Deformable parts model

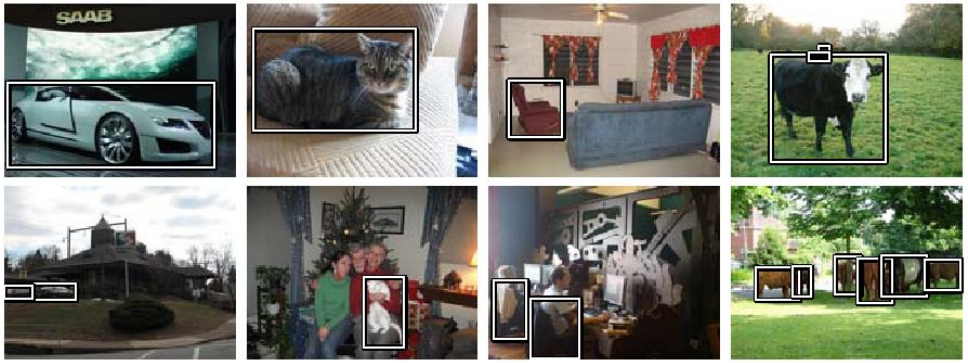
目标检测

- 问题定义
  - Detecting and localizing generic objects from various categories, such as cars, people, etc.

- 目标检测中存在的挑战
  - 光照变化
  - 视角变化
  - 物体的可变形性
  - 物体的类内差异
  - 复杂背景干扰

目标检测

- Object Detection Benchmarks
  - PASCAL VOC Challenge：20个类别



  - ImageNet Large Scale Visual Recognition Challenge (ILSVRC)：检测目标包含200个类别
  - Common Objects in Context (COCO)：80个类别

目标检测的评价标准

- 单个目标框正确与否的评判

predictions

ground truth

True positive:
- 分类正确，且与ground truth的IOU大于一定的阈值
- IOU: intersection over union

False Negative（漏检）:
- 有ground truth存在，但是没有对应的预测结果

False positive（虚警）:
- 分类错误，或者预测结果与ground truth 的IOU小于一定的阈值

目标检测的评价标准

- Precision与recall的定义

	Predicted 1	Predicted 0
True 1	true positive	false negative
True 0	false positive	true negative

	Predicted 1	Predicted 0
True 1	TP	FN
True 0	FP	TN

	Predicted 1	Predicted 0
True 1	hits	misses
True 0	false alarms	correct rejections

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

目标检测的评价标准

- 在实际检测中，我们会得到大量的预测结果，这些目标框的分数在0~1之间
  - 保留大量目标框会得到极高的recall，而precision非常低
  - 只保留预测分数很高的目标框会得到极高的precision，而recall非常低

predictions

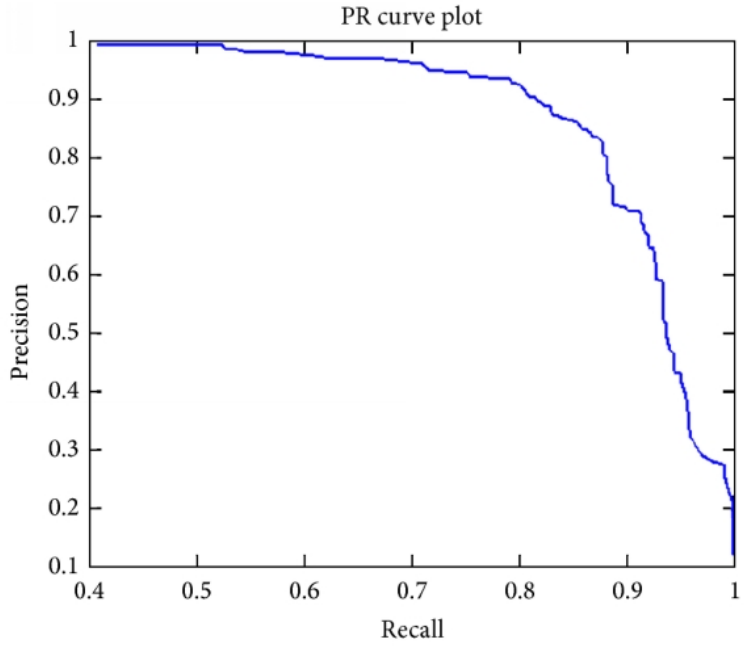
ground truth

单独依靠precision或者recall的不足以评判检测器的优劣

我们使用**mean Average Precision（mAP）**作为评价标准

目标检测的评价标准

- Average precision
  - Precision-recall 曲线

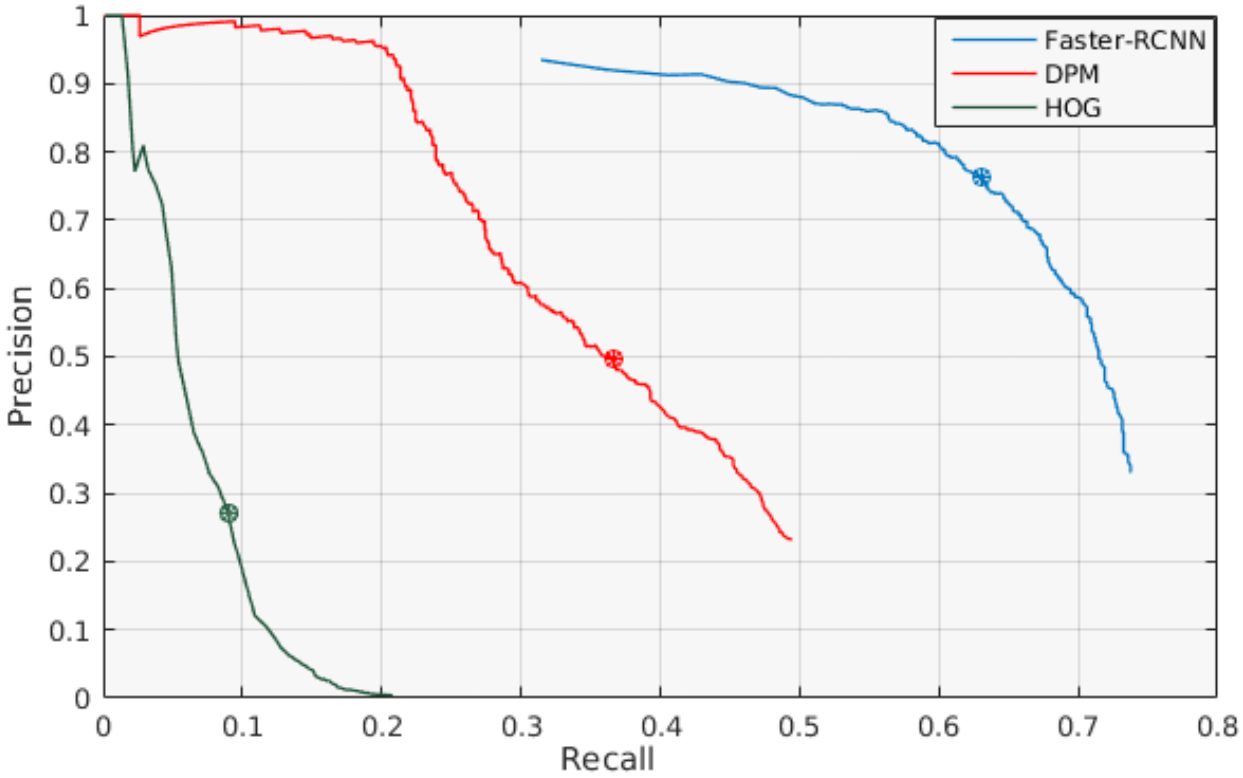


Precision-recall 曲线与坐标轴围成的面积，即为**average precision （AP）**

多个类别的average precision的平均值，即为**mean average precision（mAP）**

目标检测的评价标准

- 从precision-recall曲线以及average precision，能够很直观的比较检测方法的性能

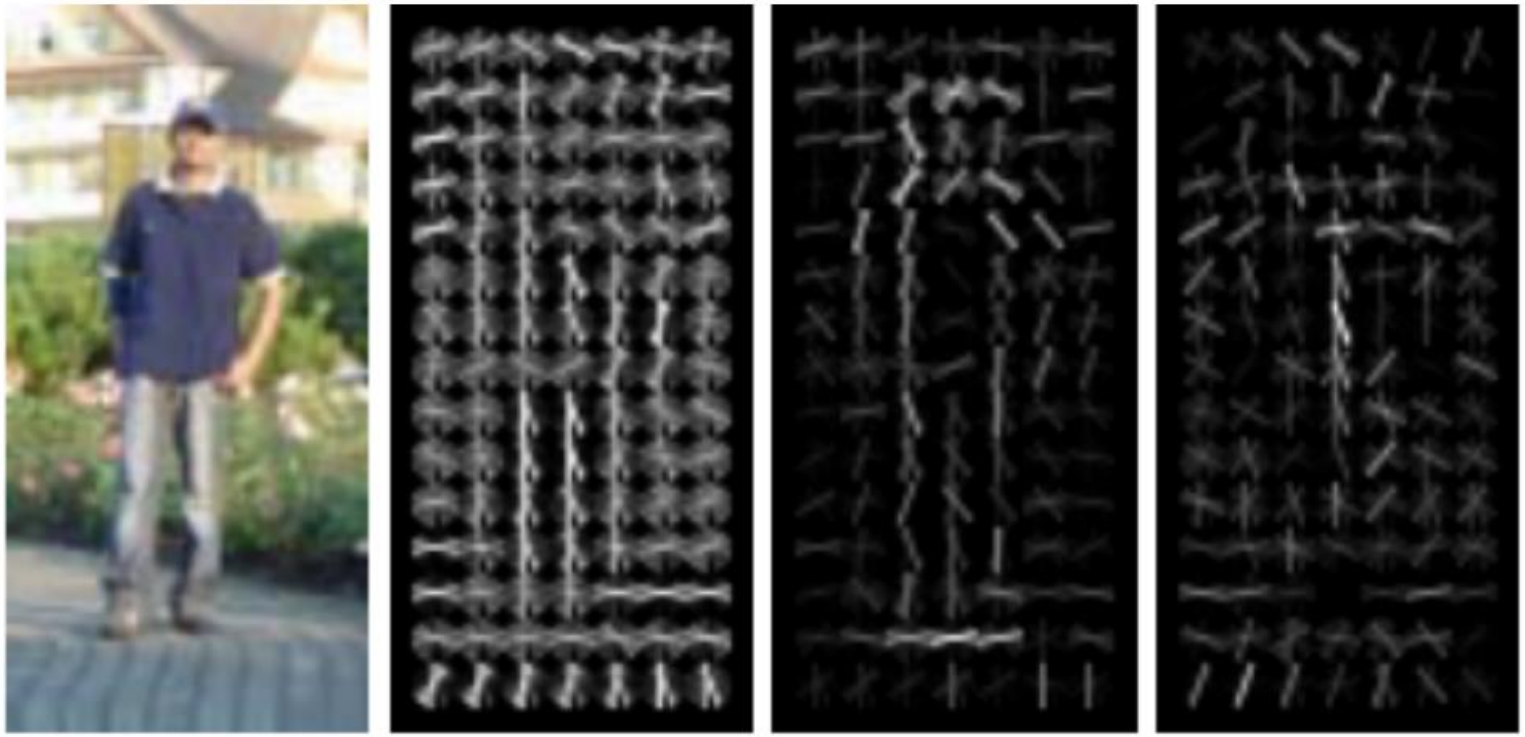


目标检测

- 简单形状的检测
- 人脸识别与检测
- 目标检测
  - 问题定义与评价标准
  - Sliding window + HOG features
  - Deformable parts model

HOG features

- 对待检测物体提取HOG template，并且将之作为filter



Sliding window + HOG features

- 用滑动窗口在整张图上搜索，判断图中是否有与HOG模板相匹配的位置

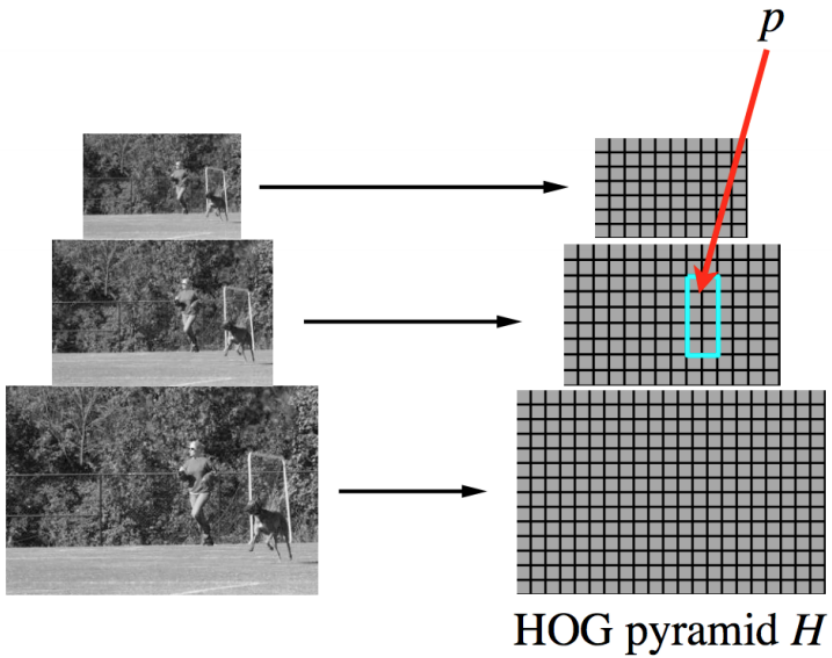
Sliding window + HOG features

- Sliding window存在的问题
  - 如何检测图中与人的尺度存在明显差异的公交车？

目标检测成功与否，与sliding window的大小选择息息相关

Sliding window + HOG features

- Multi-scale sliding window



Filter F

Score of F at position p is
$F \cdot \phi(p, H)$

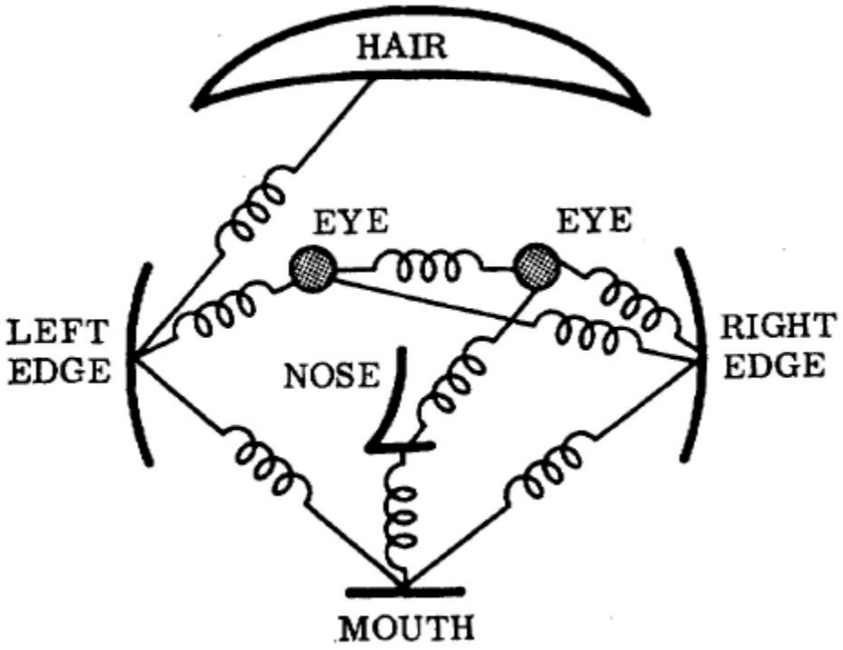
$\phi(p, H)$ = concatenation of HOG features from subwindow specified by p

目标检测

- 简单形状的检测
- 人脸识别与检测
- 目标检测
  - 问题定义与评价标准
  - Sliding window + HOG features
  - Deformable parts model

Deformable parts model

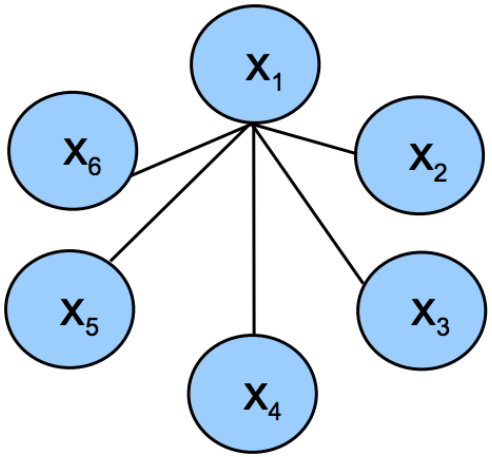
- 弹簧形变模型
  - 将物体表达为一系列可变形的部件的集合
  - 每个部件表示物体的一个局部特征
  - 部件与部件之间的呈弹簧状连接（可变形）

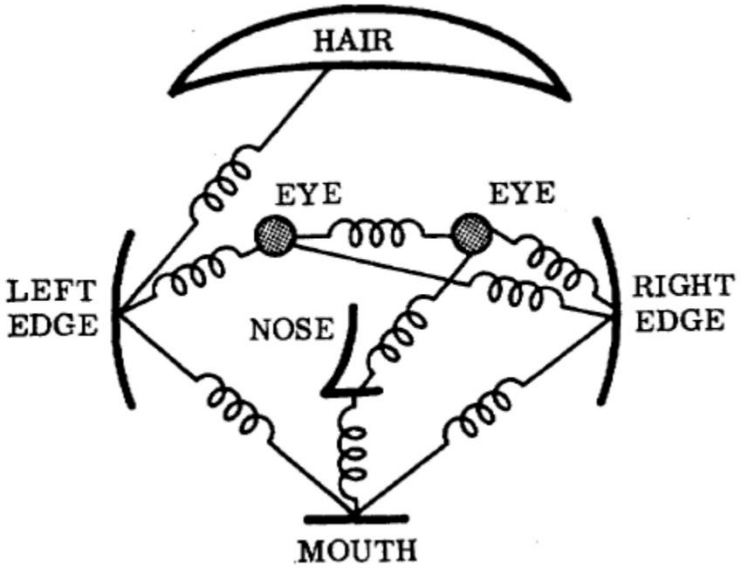


Fischler and Elschlager, Pictoral Structures, 1973

Deformable parts models

- 物体的DPM可以表示成一个star model，由一个整体模板和一系列局部模板组成

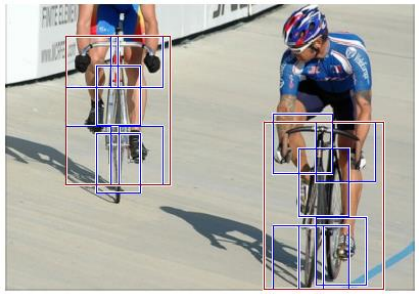


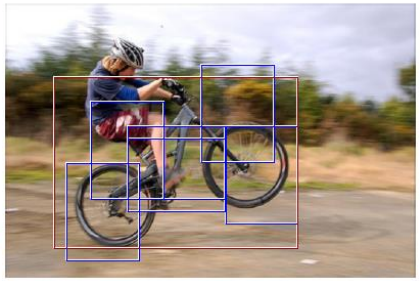


人脸模板可以表示为整个人脸检测器与各个部件（头发，左边耳朵，右边耳朵等等）的star model。人脸检测器作为root，每个部件都与root建立联系

Deformable parts model

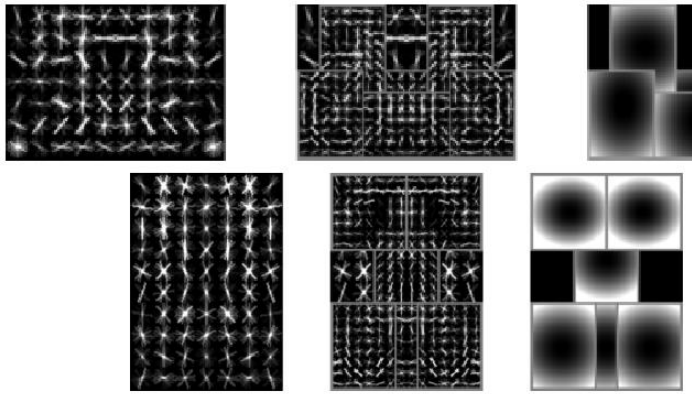
- 示例：Two-component bicycle model





“side” component

“frontal” component



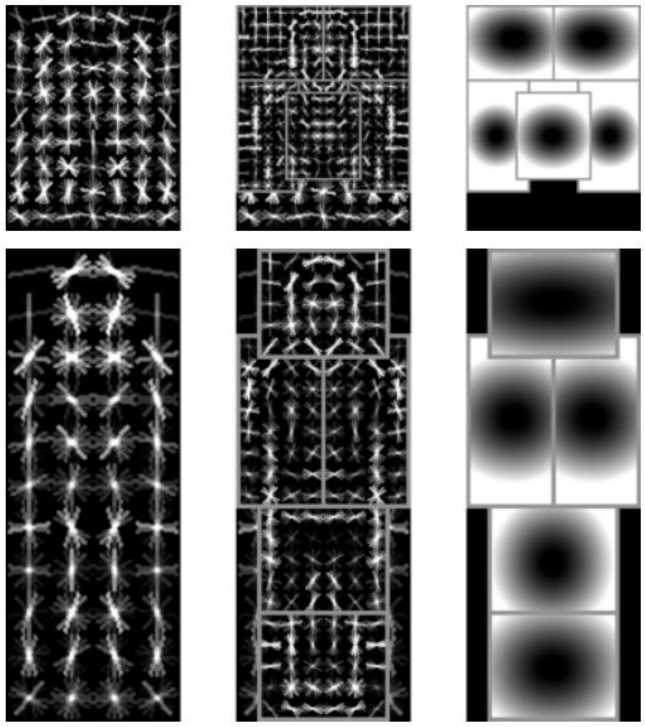
root filter　part filter　deformable cost

当part filer偏离既定位置，给出对应的惩罚

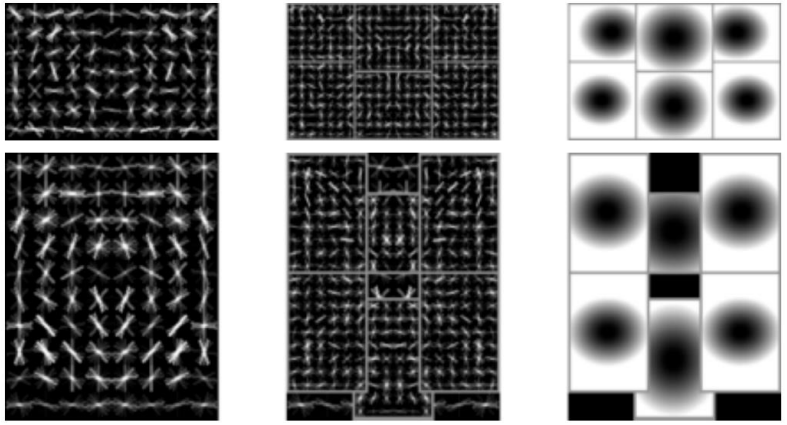
Deformable parts model

- 示例

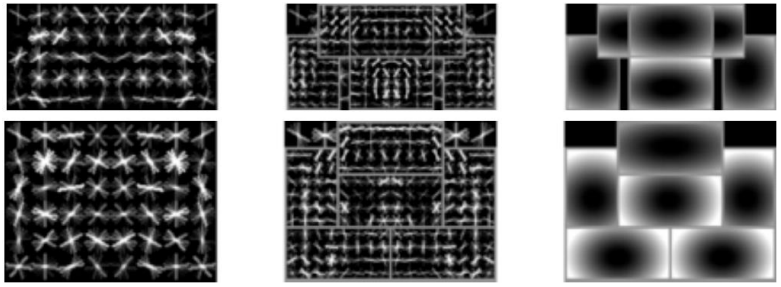
person



cat



car



Deformable parts model

- 若把物体分为可变形的n个部位，则他的DPM是一个（n+2）的元组

$$(F_0, P_1, \cdots, P_n, b)$$

Root filter　　Model for 1st part　　Bias term

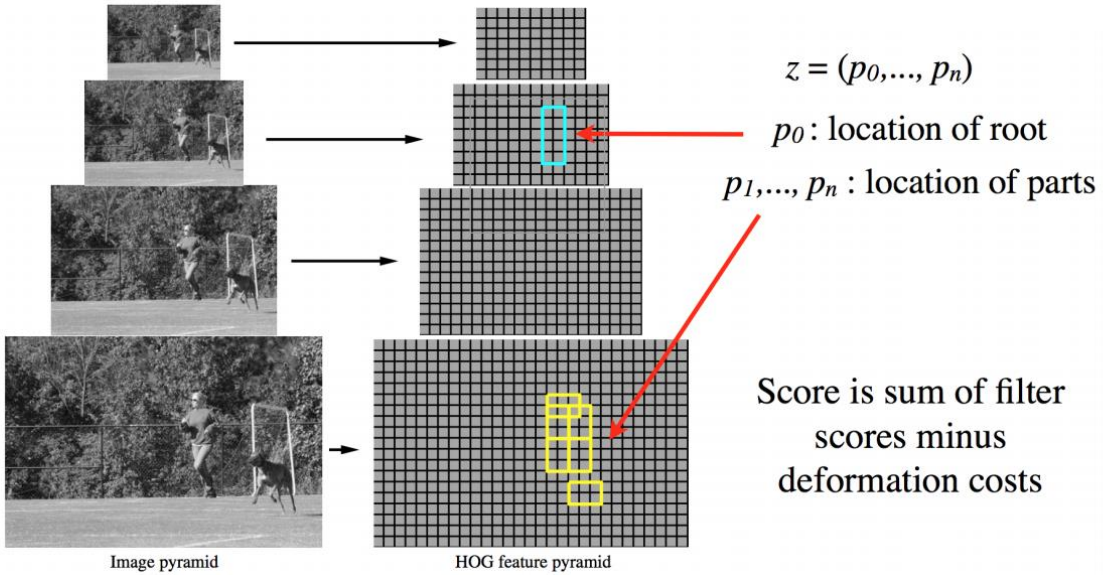
- 每个局部模型可定义为 (F_i, v_i, d_i)
  - F_i为第i个部位的filter
  - v_i为第i个部位相对于root filter的锚点位置
  - d_i为第i个部位偏移到各个位置的deformable cost

Deformable parts model

- 当使用DPM时，检测分数定义为：检测器得到的分数减去各个部位的deformable cost

$$S=\prod_{i=0}^{n} F_i \, \phi(p_i, H) - \sum_{i=1}^{n} d_i \, (dxi, dyi, dxi^2, dyi^2)$$

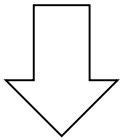
$p_i = (x_i, y_i, l_i)$ specifies the level and position of the *i*-th filter



Deformable parts model

- 基于sliding window的检测流程

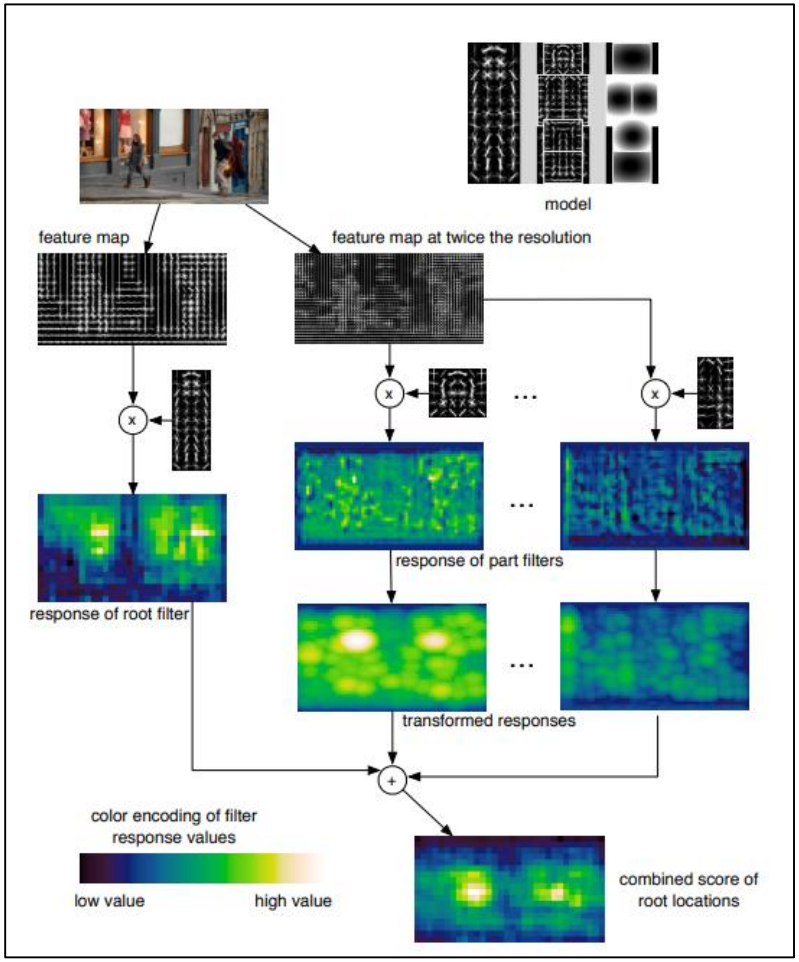
首先使用sliding window和global feature大致定位物体的位置



再使用part filters作用到物体上，与 global filter的结果相结合，得到检测的分数

Deformable parts model

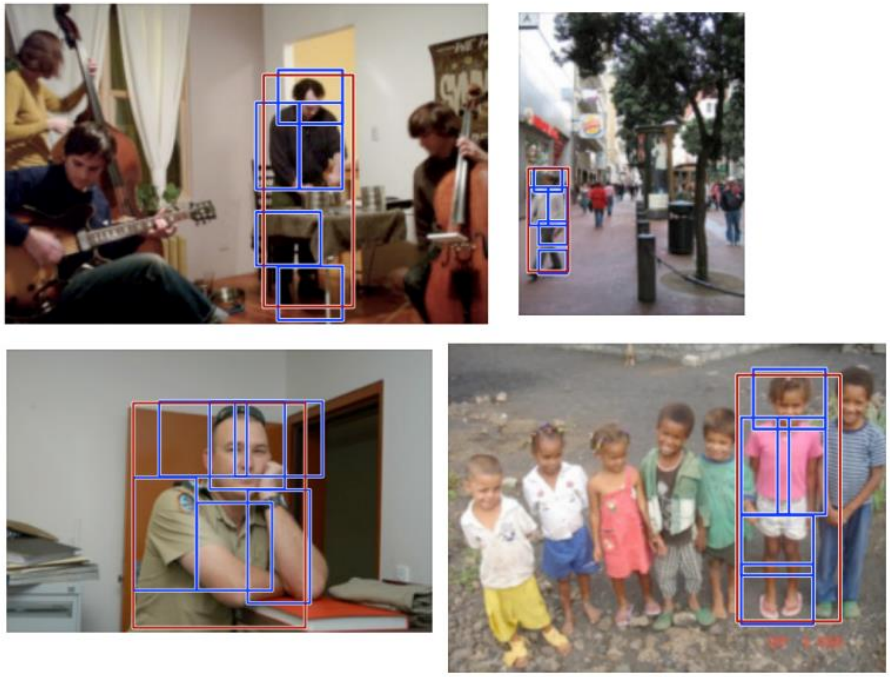
- 算法步骤
  1. 产生多个基于HOG特征的模板，包括整体模板与一系列局部模板。
  2. 用这些模板去对输入图像的HOG特征做卷积，得到响应图。
  3. 将局部模板的卷积响应图加上变形惩罚，并且和整体模板的响应图相结合，得到融合的响应图



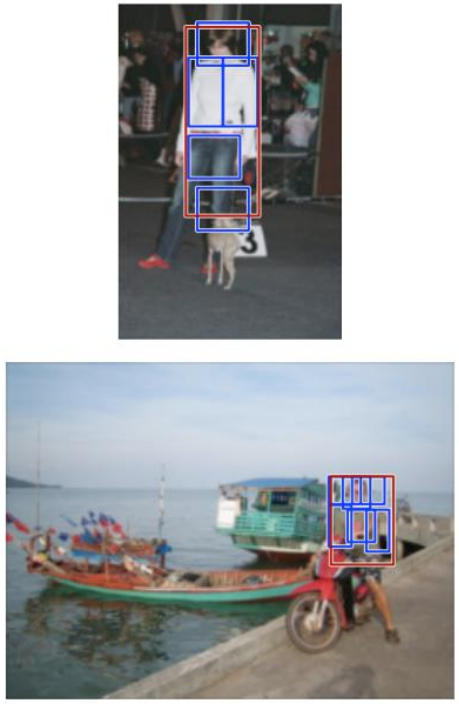
DPM检测结果

- Person

high scoring true positives



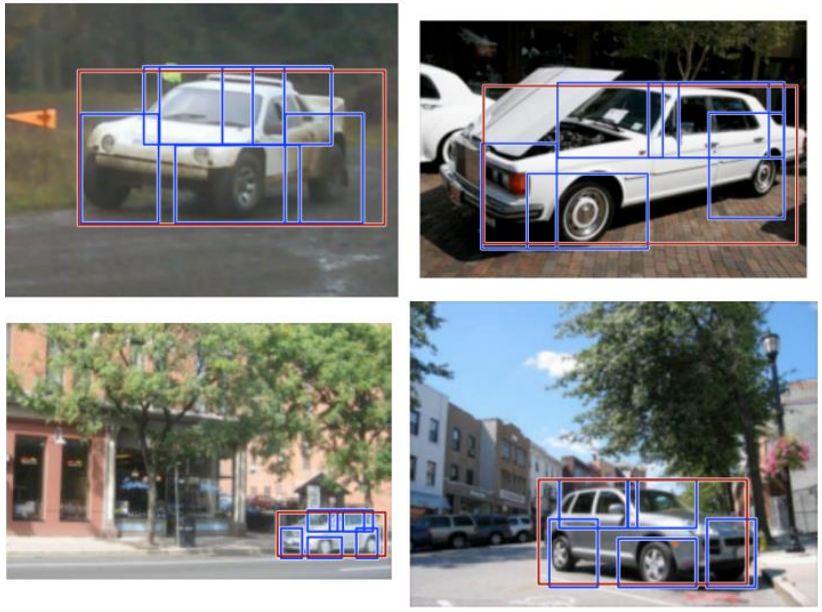
high scoring false positives
(not enough overlap)



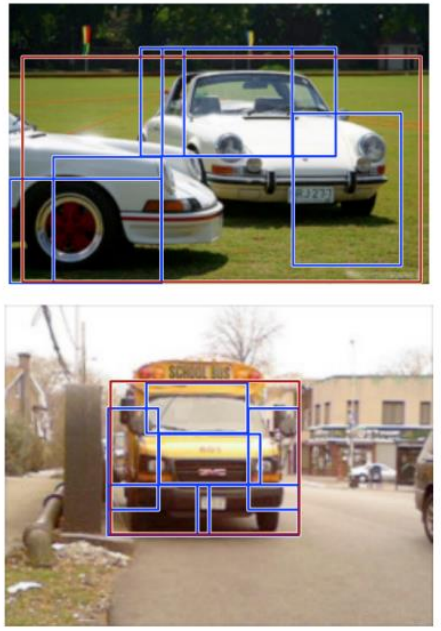
DPM检测结果

- Car

high scoring true positives



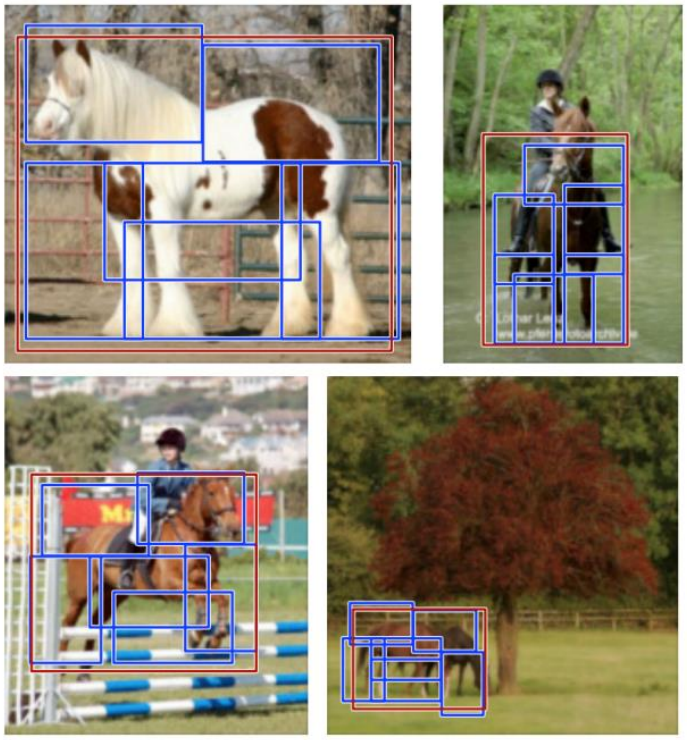
high scoring false positives



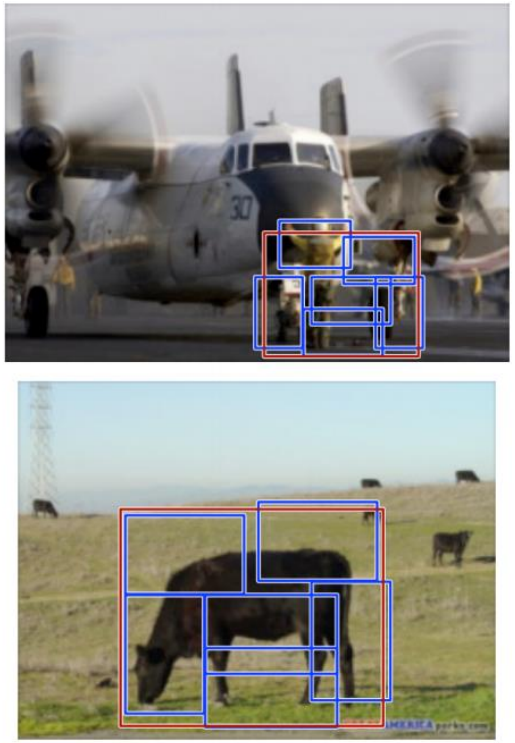
DPM检测结果

- Horse

high scoring true positives



high scoring false positives



DPM的优缺点

- Approach
  - Manually selected set of parts - specific detector trained for each part
  - Spatial model trained on part activations
  - Evaluate joint likelihood of part activations
- Advantages
  - Parts have intuitive meaning.
  - Standard detection approaches can be used for each part.
  - Works well for specific categories.
- Disadvantages
  - Parts need to be selected manually
  - Semantically motivated parts sometimes don’t have a simple appearance distribution
  - No guarantee that some important part hasn’t been missed
- When switching to another category, the model has to be rebuilt from scratch.