



# 第九章：图像特征表达

---

中国科学技术大学  
电子工程与信息科学系

# 图像特征表达的必要性

## □ 语义目标在不同图像中像素差异大

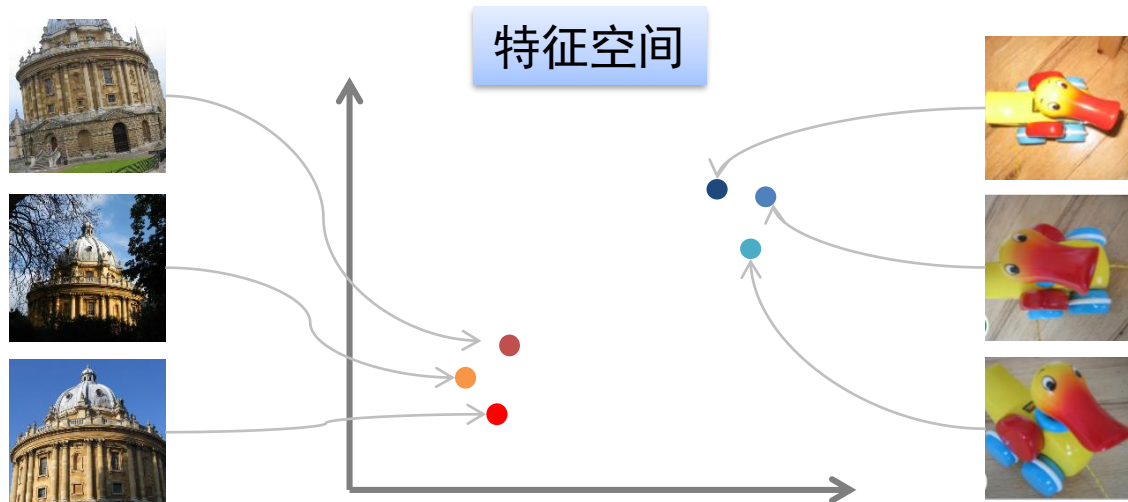
### ■ 空间变换

- ✓ 常见变换：平移，旋转，缩放
- ✓ 复杂的几何畸变

### ■ 颜色变换

## □ 图像特征特征的目标

- 以不变应万变，即对于相同语义目标的不同图像，其特征表达是相同的或相似的





# 图像特征表达

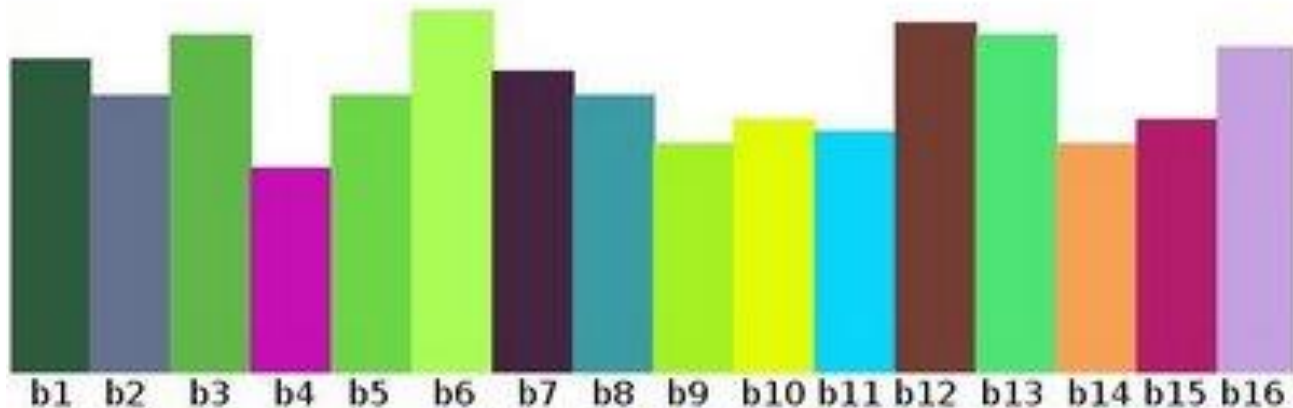
---

- 基于全局特征的图像表达
  - Histogram
  - Color Name
  - GIST
- 基于局部特征的图像表达

# 直方图 (Histogram)

## □ 什么是直方图

- 将数据的**取值空间**划分为一系列的子空间，对每个子空间进行编号(bin)，然后统计落在各个子空间的样本数量（或比例）
- 直方图就是对数据进行统计的一种方法，并且将统计值组织到一系列事先定义好的 bin 当中
- 如下图所示：bin 即直方图中的纵向条带，其数值是从数据中计算出的特征统计量
  - ✓ 这些数值可以是诸如梯度方向、色彩或任何其他特征

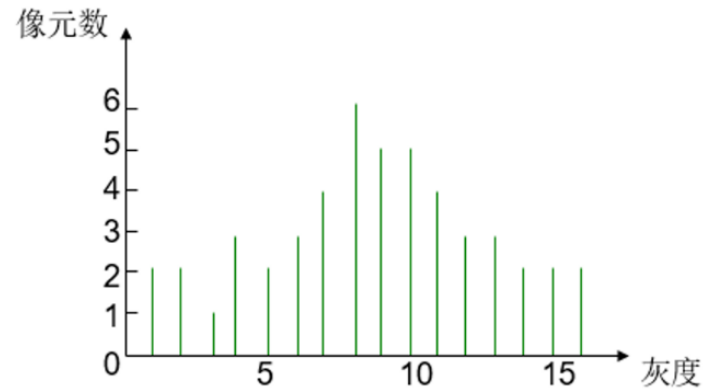


# 灰度直方图

- 一幅图像由不同灰度值的像素组成，图像中**灰度的分布**情况是该图像的一个重要特征。
- 灰度直方图对图像几何变换具有不变性

1	7	8	9	10	11	14
5	2	6	7	14	12	15
3	4	7	8	6	9	11
2	1	4	7	8	8	9
8	4	5	9	11	12	10
8	10	11	15	16	10	13
13	6	9	16	13	12	10

数字图像



灰度直方图

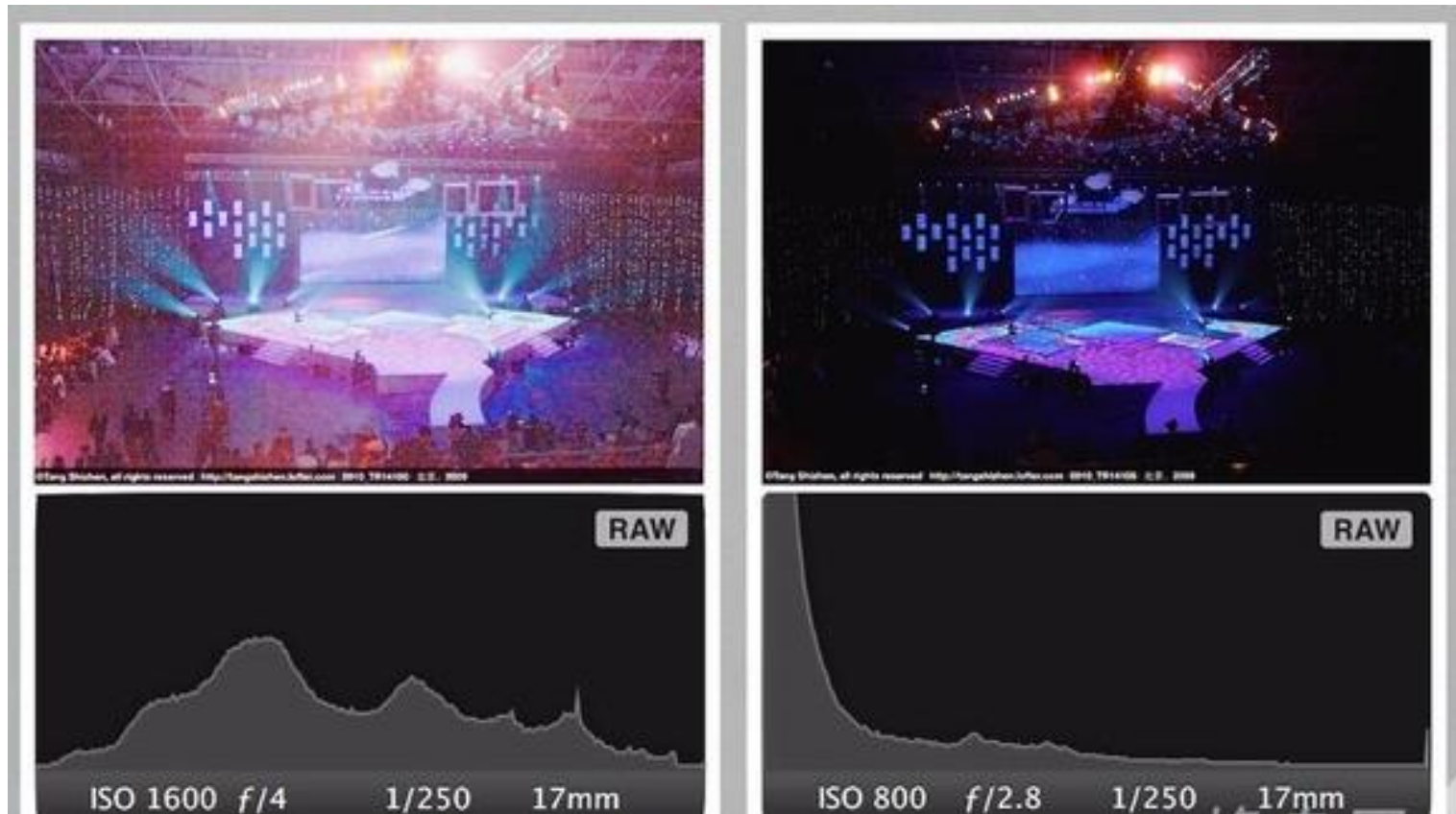


直方图统计表

灰度级	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
像元数	2	2	1	3	2	3	4	6	5	5	4	3	3	2	2	2
比例值	0.04	0.04	0.02	0.06	0.04	0.06	0.08	0.12	0.10	0.10	0.08	0.06	0.06	0.04	0.04	0.04

# 灰度直方图

- 根据直方图的形态可以大致推断图像质量的好坏



# 灰度直方图的比较计算

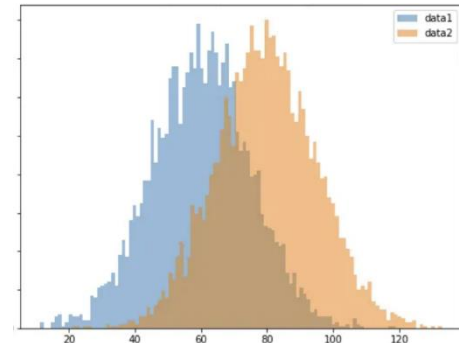
- 直方图的相关系数（事先对直方图进行  $L_2$  归一化）
  - 值越高则越匹配

$$s(H_1, H_2) = \frac{\sum_I (H_1(I) - \bar{H}_1)(H_2(I) - \bar{H}_2)}{\sqrt{\sum_I (H_1(I) - \bar{H}_1)^2 \sum_I (H_2(I) - \bar{H}_2)^2}}$$

其中  $\bar{H}_k = \frac{1}{N} \sum_J H_k(J)$ ， $N$  为直方图 bins 的个数

- 直方图相交(intersection)，度量其相似性
  - 事先对直方图进行  $L_1$  归一化

$$s(H_1, H_2) = \sum_n \min(H_1(n), H_2(n))$$





# 图像特征表达

---

- 基于全局特征的图像表达
  - Histogram
  - Color Name
  - GIST
  
- 基于局部特征的图像表达



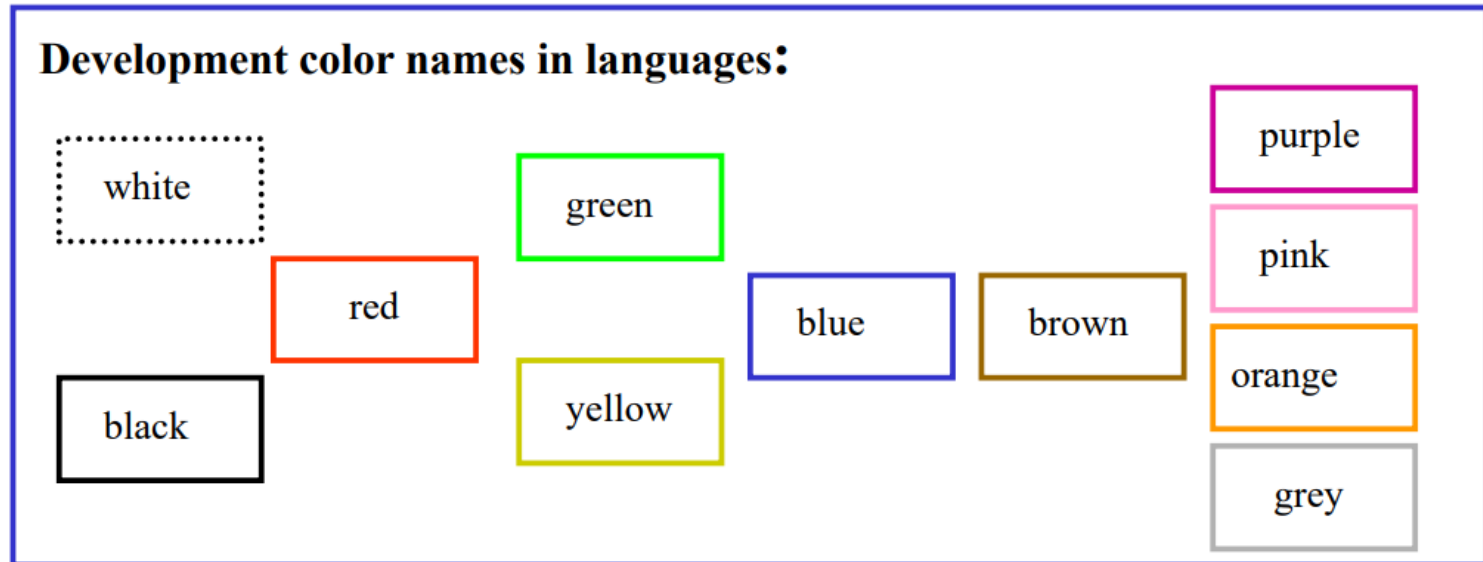


# Color Name: 一种颜色描述的方法

- 定义：人们交流时用于描述颜色的词汇
  - e.g. 蓝色、红色、绿色 ...
- Basic color terms (word)：基本颜色术语
  - 无统一可操作的定义，一般可指
    - ✓ 可应用于描述绝大部分种类的物体
    - ✓ 不包含于其它基本颜色
      - 深红(crimson)，鲜红(scarlet)不是基本颜色，同属于红色(red)
    - ✓ 该语言的绝大部分使用者对这些词汇的使用都是一致的
    - ✓ 不同的语言间基本颜色的描述通常是共享的
      - 中文里有红色，英语也有 red
    - ✓ ...
- 不同语言的 Basic color terms 的数量是不一致的
  - 原始：黑白2种
  - Russian：12种，English：11种

# Color Name

- 语言研究表明：不同语言中 Color Name 的发展都遵循一种相似的模式



- 英语中包含 11 种基本颜色术语
  - black, blue, brown, grey, green, orange, pink, purple, red, white, and yellow

# Computational Color Naming

- 学习如何将像素值映射为color name标签



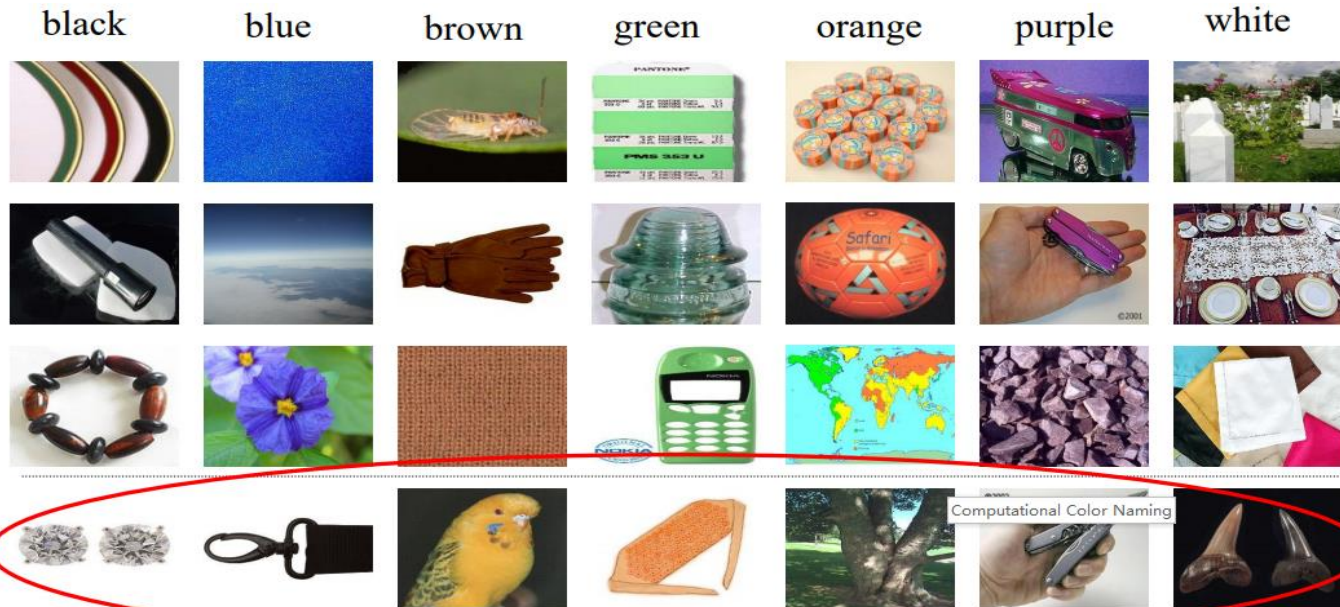
- 问题：大量图片中物体的颜色并未明确标注，是否可以自动对这些图片用 color name 进行标注？

- Find me all yellow cars?



# Color Names学习

- 使用搜索引擎收集一组弱标记图片(weakly labeled)
- 利用话题模型 (topic model, pLSA) 学习图像的color name特征表达

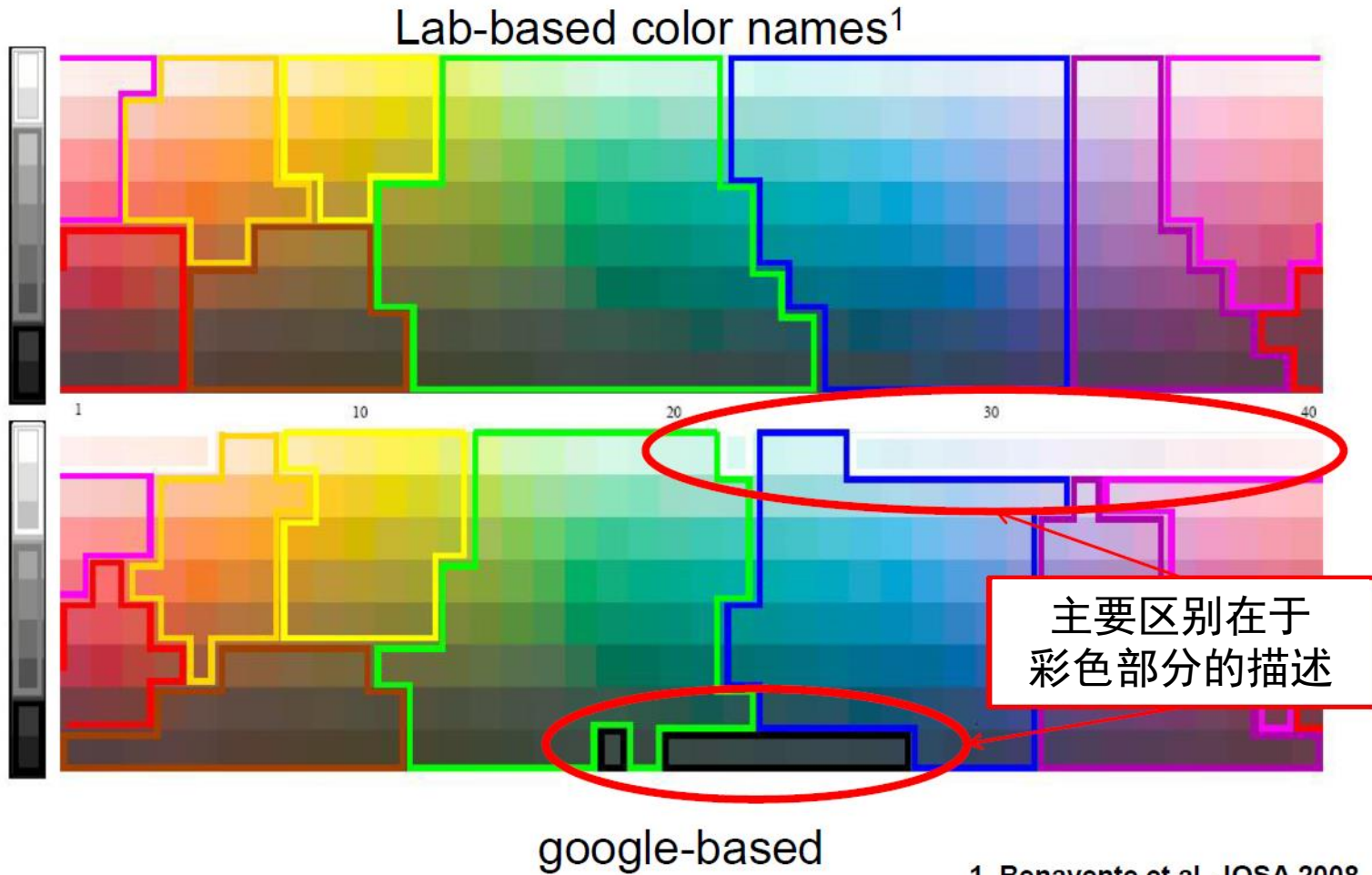


false positives

使用 Google图片 检索出的图片



# Color Names学习



1. Benavente et al. JOSA 2008

- Van De Weijer J, Schmid C, Verbeek J. Learning color names from real-world images[C]//2007 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2007: 1-8.



# 图像特征表达

---

- 基于全局特征的图像表达
  - Histogram
  - Color Name
  - GIST
- 基于局部特征的图像表达

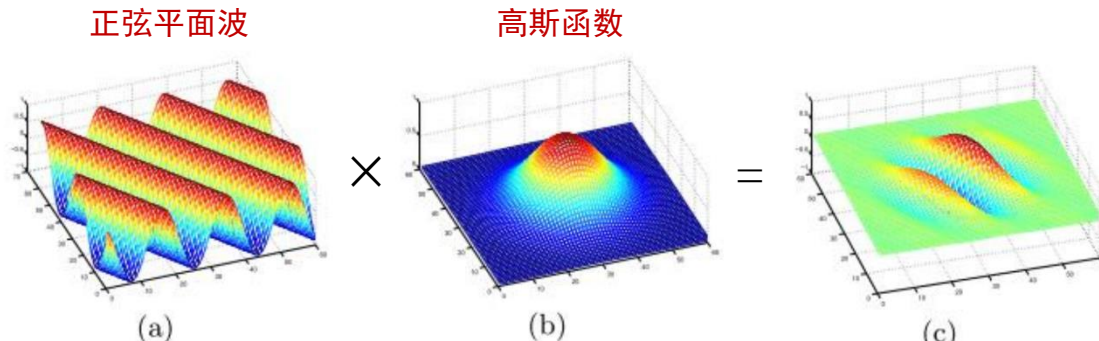
# GIST

- 在图像处理中，Gabor函数是一个用于边缘提取的线性滤波器
- 二维 Gabor 滤波器：正弦平面波调制的高斯核函数
  - 加窗短时 Fourier 变换

$$g(x, y) = s_c(x, y)w_r(x, y)$$

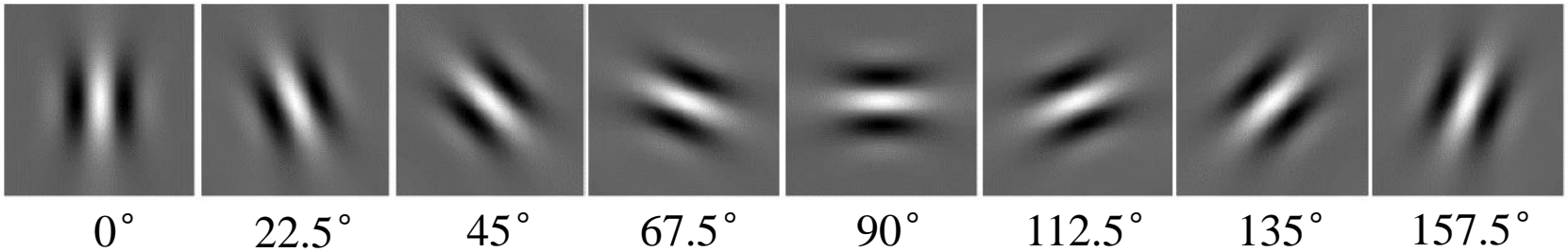
其中  $s_c(x, y) = e^{j(2\pi(u_0x+v_0y)+P)}$       $w_r(x, y) = Ke^{-\pi(a^2(x-x_0)^2+b^2(y-y_0)^2)}$

- ✓  $u_0$  和  $v_0$  分别表示空间中水平和垂直方向的频率
- ✓  $P$  是正弦函数的相位
- ✓  $x_0$  和  $y_0$  分别表示高斯函数的中心
- Gabor滤波器的频率和方向表达同人类视觉系统类似
- Gabor滤波器十分适合纹理表达和分离

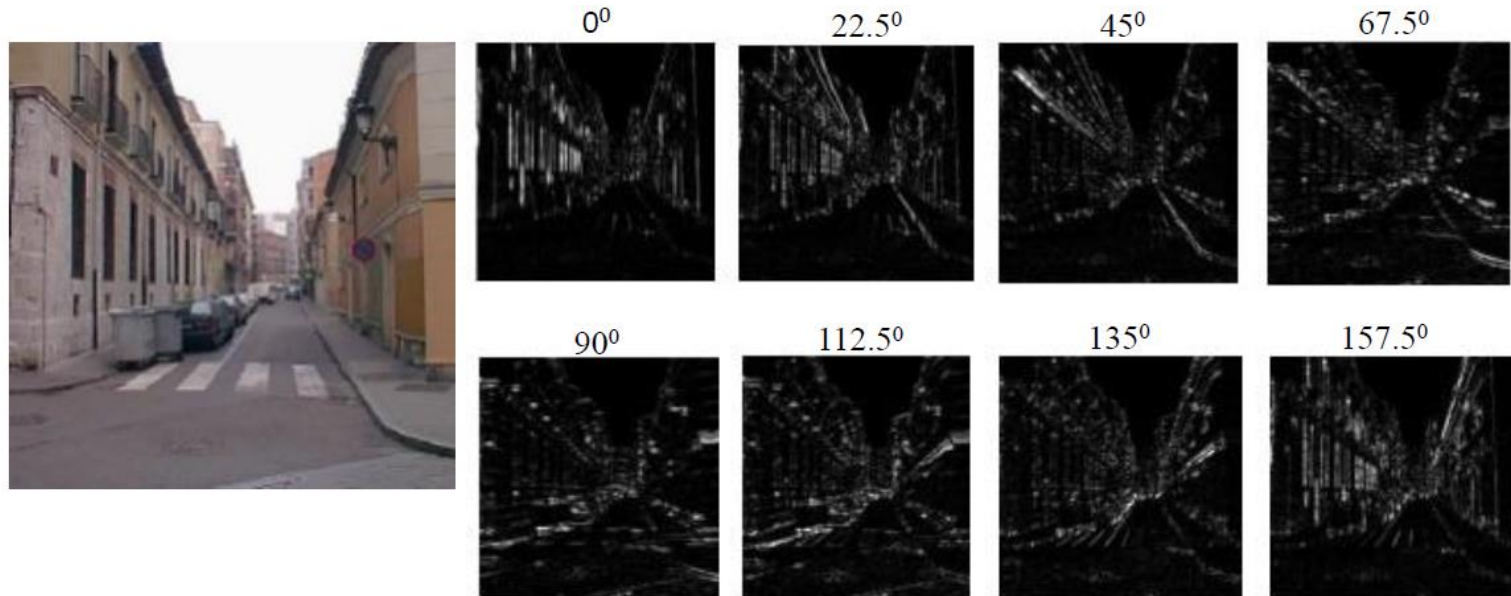


# GIST

## □ 不同方向的 Gabor 滤波器



## □ Gabor 滤波示例

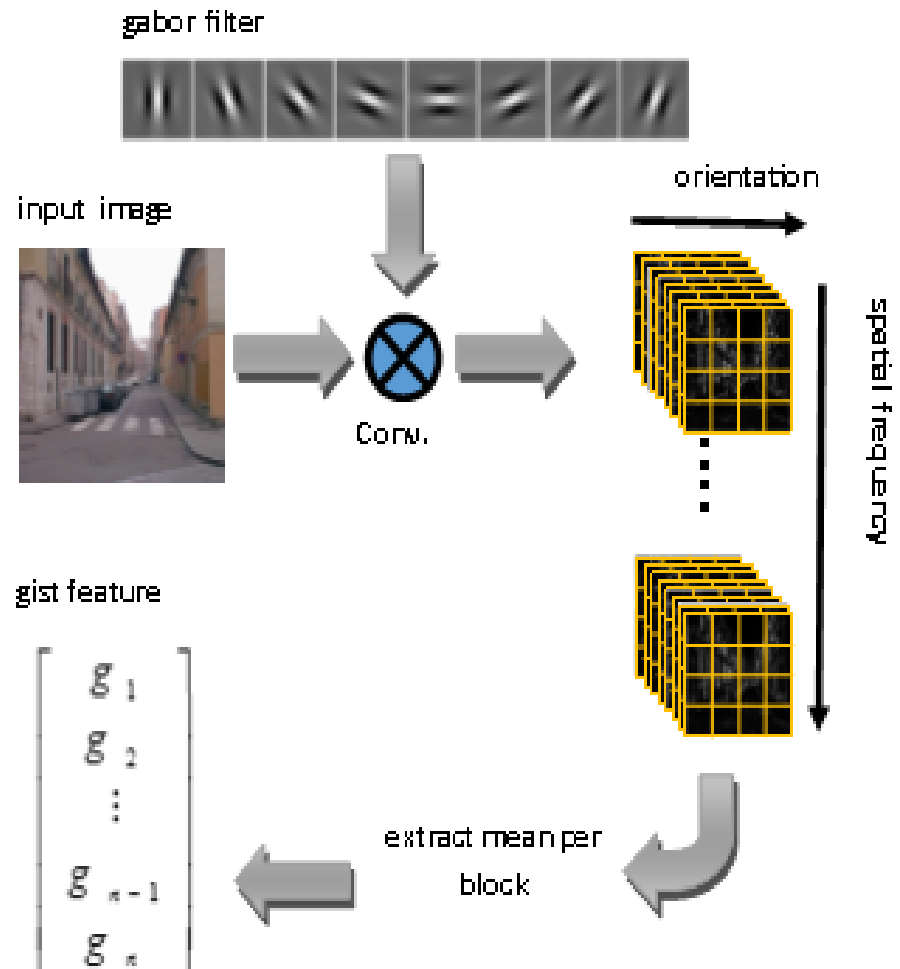




# GIST

## □ GIST特征提取一般流程

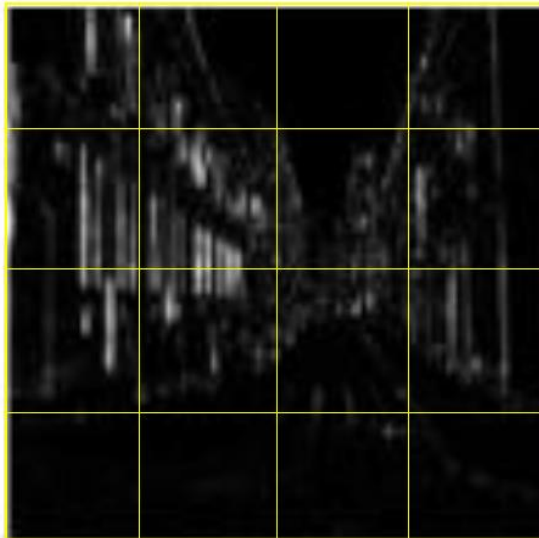
- 图象分块操作
- 计算均值进行降维



# GIST

## □ GIST-512

- 32个 gabor 滤波器：8个方向，4个尺度
  - ✓ 得到 32 个与原图等大的 feature map。
- 将每个 feature map 分为 4x4 block，计算每个block内的均值
- 将 32 个 feature map 的所有 block 的均值合并
  - ✓ 共 16x32=512 个均值，即 512 维 GIST 特征



$$\bar{x} = \frac{\sum x}{n}$$

feature map 中每个 block 计算均值

# GIST

## □ GIST 特征提取算法

---

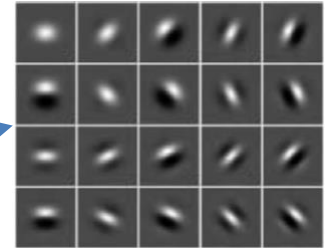
**Algorithm :** Extraction algorithm of global GIST feature

---

**Input:** Image  $I(x, y)$  with the size of  $M \times N$

**Out put:** Feature vector  $G^G$

- 1: for  $j=1, \dots, m$  do
- 2:   for  $k=1, \dots, n$  do
- 3:     Calculate  $m \times n$  Gabor filters  $g_{jk}(x, y)$
- 4:   end for
- 5: end for
- 6: Divide  $I(x, y)$  into  $n_g = n_b \times n_b$  blocks and label those different blocks with  $B_i$
- 7: for  $i=1, \dots, n_g$  do
- 8:   Calculate  $G_i^B(x, y)$
- 9:   Calculate  $\overline{G_i^B}$
- 10: end for
- 11:  $G^G = \{\overline{G_1^B}, \overline{G_2^B}, \dots, \overline{G_{n_g}^B}\}$



$$G_i^B(x, y) = \text{cat}(I(x, y) * g_{mn}(x, y)), (x, y) \in B_i$$

$$\overline{G_i^B} = \frac{1}{M' \times N'} \sum_{(x,y) \in B_i} G_i^B(x, y)$$

↓  
block size



# 图像表达

- 基于全局特征的图像表达
- 基于局部特征的图像表达
  - 简单的局部视觉特征
    - ✓ 局部二值模式 (LBP)
    - ✓ 梯度方向直方图 (HOG)
    - ✓ 形状上下文 (Shape Context)
  - 基于关键点检测的局部视觉特征
    - ✓ 图像表达基本框架
    - ✓ 局部视觉特征描述
    - ✓ 特征编码与聚合



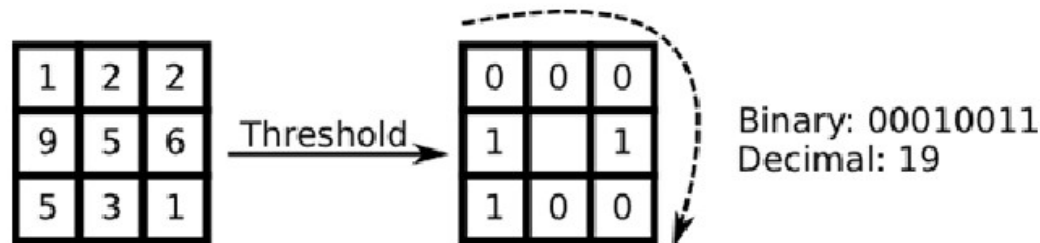
# 局部二值模式 (Local Binary Pattern)

## □ 图像特点

- 在不同光照条件下，同一场景的图像像素亮度变化显著，但局部区域的像素亮度**相对大小关系**是稳定的

## □ LBP特征描述

- 在3x3的窗口内，以中心像素为阈值，将相邻8个像素的灰度值与中心像素比较
  - ✓ 周围像素值  $\geq$  中心像素值：标记为 1
  - ✓ 周围像素值  $<$  中心像素值：标记为 0
- 邻域8个点经比较可产生8位二进制数
  - ✓ 种类：256种 ( $2^8$ )
  - ✓ 转换为十进制数：LBP值



# 局部二值模式 (Local Binary Pattern)

## □ 函数表示

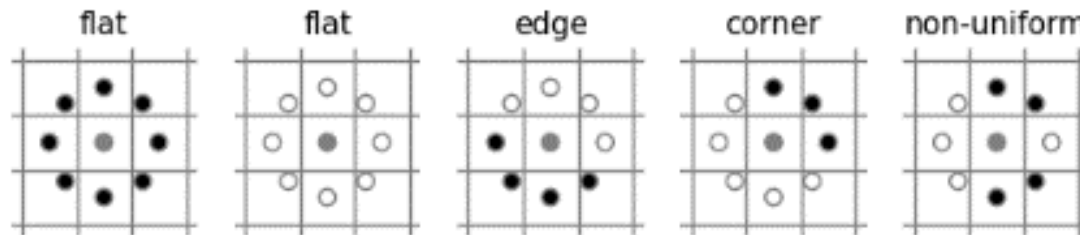
- LBP操作可定义为：

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p \cdot s(i_p - i_c)$$

其中 $(x_c, y_c)$ 为中心像素，其亮度为 $i_c$ ， $i_p$ 是周围像素亮度， $s$ 为符号函数。

## □ 小结

- LBP可以很好地捕捉图像细节，其值可反映不同的纹理

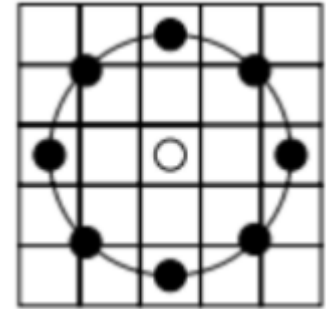


- 但由于近邻区域大小固定，不能满足不同尺寸和频率纹理需要
- 对图像的**亮度变化**具有较好的鲁棒性/不变性
  - ✓ 不变性(invariance): 以不变应万变

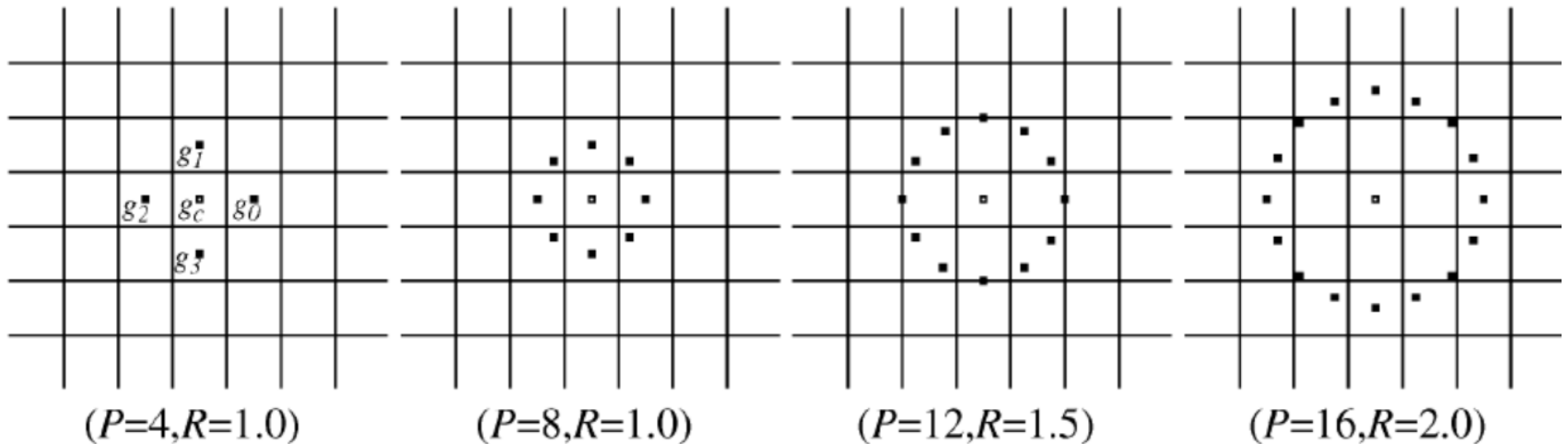
# LBP的扩展（环状LBP）

## □ 改进

- 3x3邻域扩展为任意邻域
- 圆形邻域代替正方形邻域
- 非整数点像素的灰度值可通过插值得到



## □ $LBP_P^R$ : 表示在半径为 $R$ 的圆内有 $P$ 个像素点



# LBP的扩展（环状LBP）

□ 对于一个给定的点 $(x_c, y_c)$ ，其近邻点为 $(x_p, y_p)$ ， $p \in P$

■ 可以由如下公式计算坐标

$$x_p = x_c + R \cos\left(\frac{2\pi p}{P}\right) \quad y_p = y_c - R \sin\left(\frac{2\pi p}{P}\right)$$

其中 $R$  圆的半径， $P$  为样本点的数目

■ 如果近邻点不在图像的整数坐标上，如何确定该点的像素值？

✓ 双线性插值：

$$f(x, y) \approx [1 - x \quad x] \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1 - y \\ y \end{bmatrix}$$



半径越小，纹理越细

原始LBP 圆形LBP<sub>8</sub><sup>1</sup> 圆形LBP<sub>8</sub><sup>3</sup> 圆形LBP<sub>6</sub><sup>3</sup>



# 局部二值模式：光照不变性

- 由定义可见，LBP 对光照变化非常鲁棒

原图

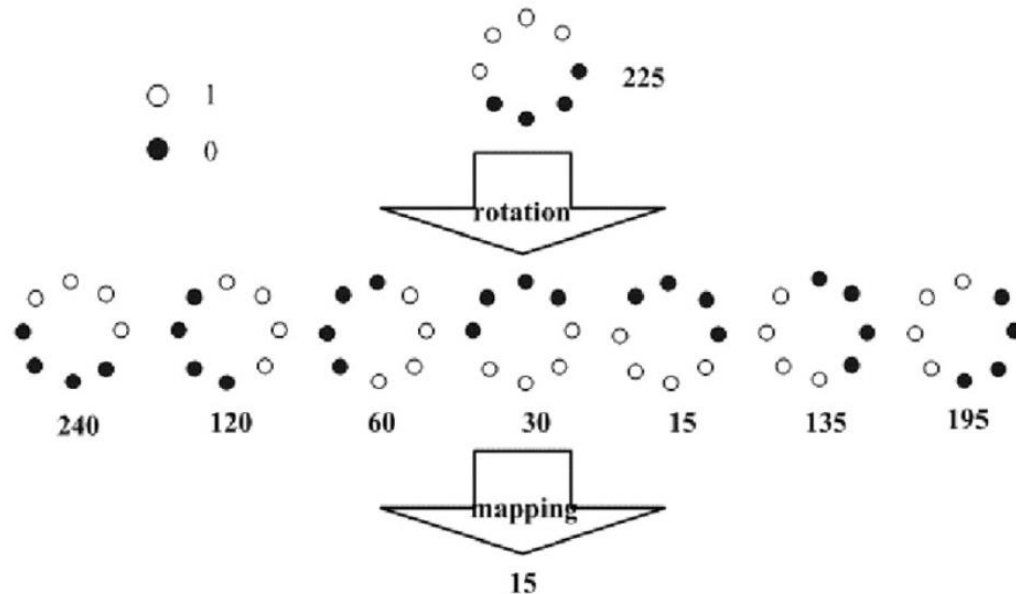


LBP图



# 局部二值模式：旋转不变性

- **旋转不变性**：图像发生旋转变换后，图像特征保持不变
- LBP本身不是旋转不变的，同一幅图像，进行旋转后，其特征会有很大的差别
- **改进**：不断旋转圆形邻域得到一系列初始定义的LBP值，取最小值作为该邻域的值
  - 下图 8 种LBP模式，对应旋转不变的LBP模式均为 00001111
  - **代价**：牺牲了特征对不同模式的区分性





# LBP等价模式 (Uniform Pattern LBP)

## □ 问题：二进制模式过多

- 一个LBP算子可以产生不同的二进制模式，对于半径为R的圆形区域内含有P个采样点的LBP算子 ( $LBP_P^R$ )将会产生  $2^P$  种模式。  
(3x3邻域内采8个点，则有  $2^8$  种模式，7x7邻域内采36个点，则有  $2^{36}$  种模式)

## □ 解决方法：等价模式

- 作用：对LBP算子的模式种类进行**归并**
- 定义：当某个LBP所对应的**循环**二进制数从0到1或从1到0**最多有两次跳变时**，该LBP所对应的二进制就称为一个等价模式类

## □ 示例

- 等价模式类：
  - ✓ 00000000 (0次跳变), 00000111 (2次跳变), 10001111 (2次跳变), ...
- 除等价模式类以外的模式均归为一类，称作混合模式类：
  - ✓ 10010111 (4次跳变)

# LBP等价模式 (Uniform Pattern LBP)

## □ 模式数量降维

- 等价模式类数量： $2^P \rightarrow P(P - 1) + 2$

## □ 示例

- 3x3邻域8个采样点
  - ✓ 总模式由 256 种减少为 59 种
  - ✓ 等价模式的值从小到大编码为 1-58，混合模式类编码为 0
- 等价模式LBP特征图整体偏暗，特征向量维数更少，还可以减少高频噪声带来的影响。



圆形  
 $LBP_8^3$

旋转不变  
 $LBP_8^3$

等价模式  
 $LBP_8^3$



# LBP的应用：LBP Histograms

## □ LBP特征统计直方图(LBPH)

- 将LBP特征与图像的空间信息结合。
- 首先，将LBP特征图像分成m个局部块，并提取每个局部块的直方图
- 然后，将这些直方图依次连接在一起形成LBP特征的统计直方图，即LBPH。

## □ 应用：OpenCV人脸识别

- 计算图像的LBP图像
- 将LBP特征图像进行分块（如分成8行8列64块区域）
- 计算每块区域特征图像的直方图，将直方图进行归一化，直方图大小为： $1 \times NumPatterns$
- 将每块区域的直方图按空间顺序依次排列成一行，形成LBP特征向量，大小为： $1 \times (NumPatterns \times 64)$
- 用机器学习的方法对LBP特征向量进行训练，用于检测和识别目标



# LBP: 小结

---

## □ 优点

- 一定程度上消除了光照变化的影响
- 通过改进，可具有旋转不变性
- 纹理特征维度低，计算速度快

## □ 缺点

- 当光照变化不均匀导致局部邻域像素间的灰度大小关系被破坏，对应的LBP结果可能也发生变化
- 通过引入旋转不变的定义，使LBP算子更具鲁棒性。但这也使得LBP结果丢失了方向信息





# 图像表达

- 基于全局特征的图像表达
- 基于局部特征的图像表达
  - 简单的局部视觉特征
    - ✓ 局部二值模式 (LBP)
    - ✓ 梯度方向直方图 (HOG)
    - ✓ 形状上下文 (Shape Context)
  - 基于关键点检测的局部视觉特征
    - ✓ 图像表达基本框架
    - ✓ 局部视觉特征描述
    - ✓ 特征编码与聚合



# 梯度方向直方图

(Histogram of Oriented Gradient, HOG)

## □ HOG 特征

- 统计图像局部区域的**梯度方向直方图**
- 主要思想
  - ✓ 在一副图像中，目标的外观和形状（appearance and shape）能够被梯度或边缘的方向分布描述。
  - ✓ 本质：梯度的统计信息，而梯度主要存在于边缘的地方

## □ 主要步骤

- 梯度计算
- 直方图统计
- 块(block)描述子



# 梯度方向直方图：梯度计算

## □ 图像预处理（可选）

- 灰度化：将彩色图像变换为灰度图像
- 伽马校正：调节图像的对比度，降低光照因素的影响

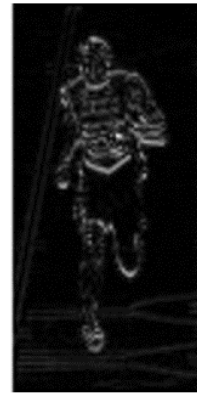
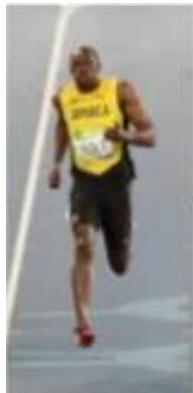
## □ 梯度计算

- 计算水平梯度  $g_x$  和竖直梯度  $g_y$
- 计算总的梯度强度和梯度方向

-1	0	1
----	---	---

-1
0
1

$$g = \sqrt{g_x^2 + g_y^2} \quad \theta = \arctan \frac{g_x}{g_y}$$



水平梯度

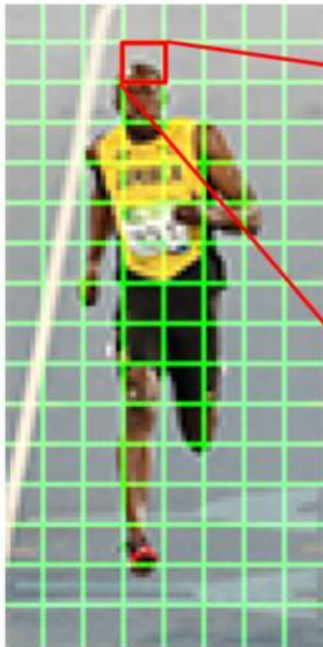
竖直梯度

梯度强度

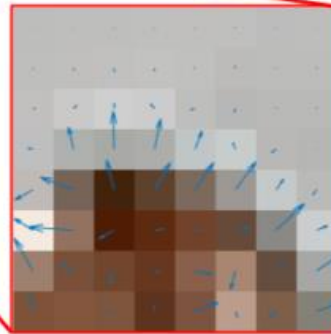
# 梯度方向直方图：直方图统计-I

## □ 对每个 cell 统计梯度直方图

- Cell size: 8x8
- 对于灰度图，每个 8x8 的 cell 包含 64 个像素，每个像素包含 2 个值（梯度大小和方向）
- 使用 9-bins 的直方图对梯度进行统计



8x8 cells of HOG



2	3	4	4	3	4	2	2
5	11	17	13	7	9	3	4
11	21	23	27	22	17	4	6
23	99	165	135	85	32	26	2
91	155	133	136	144	152	57	28
98	196	76	38	26	60	170	51
165	60	60	27	77	85	43	136
71	13	34	23	108	27	48	110

Gradient Magnitude

梯度大小

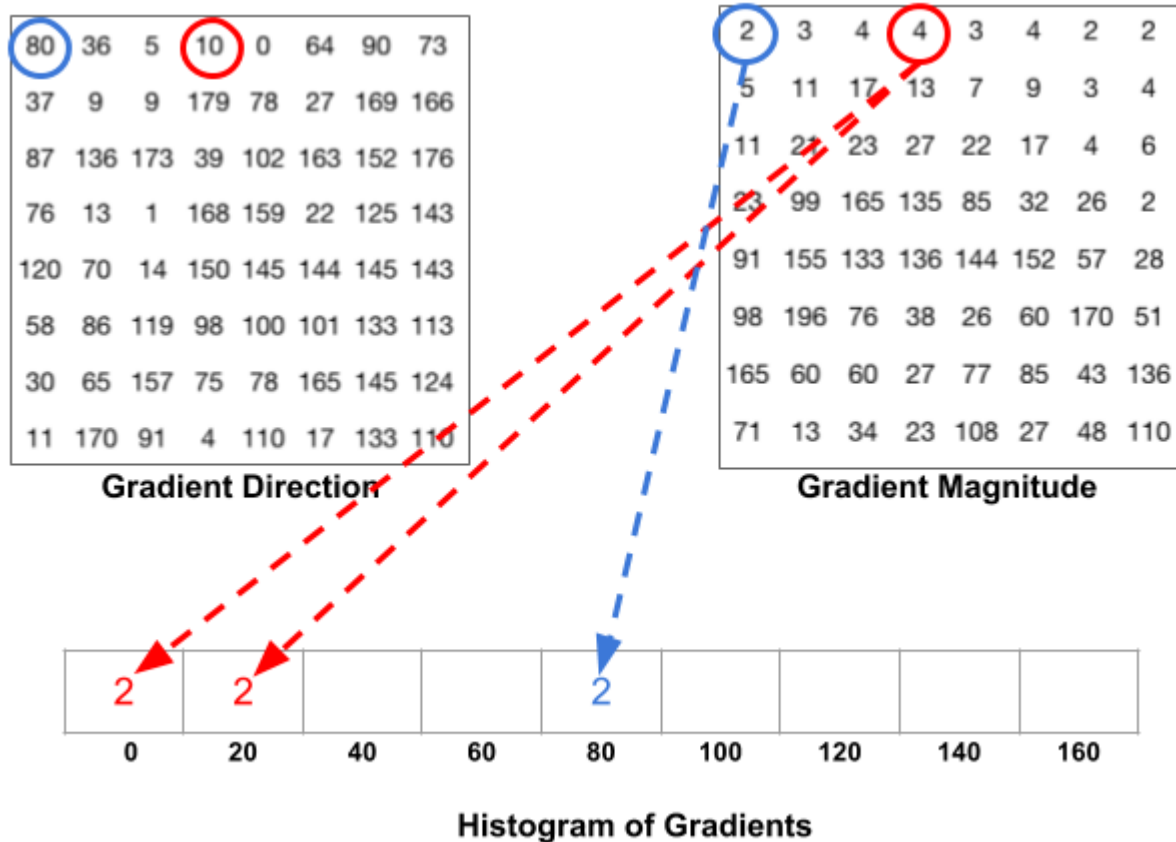
80	36	5	10	0	64	90	73
37	9	9	179	78	27	169	166
87	136	173	39	102	163	152	176
76	13	1	168	159	22	125	143
120	70	14	150	145	144	145	143
58	86	119	98	100	101	133	113
30	65	157	75	78	165	145	124
11	170	91	4	110	17	133	110

Gradient Direction

梯度方向

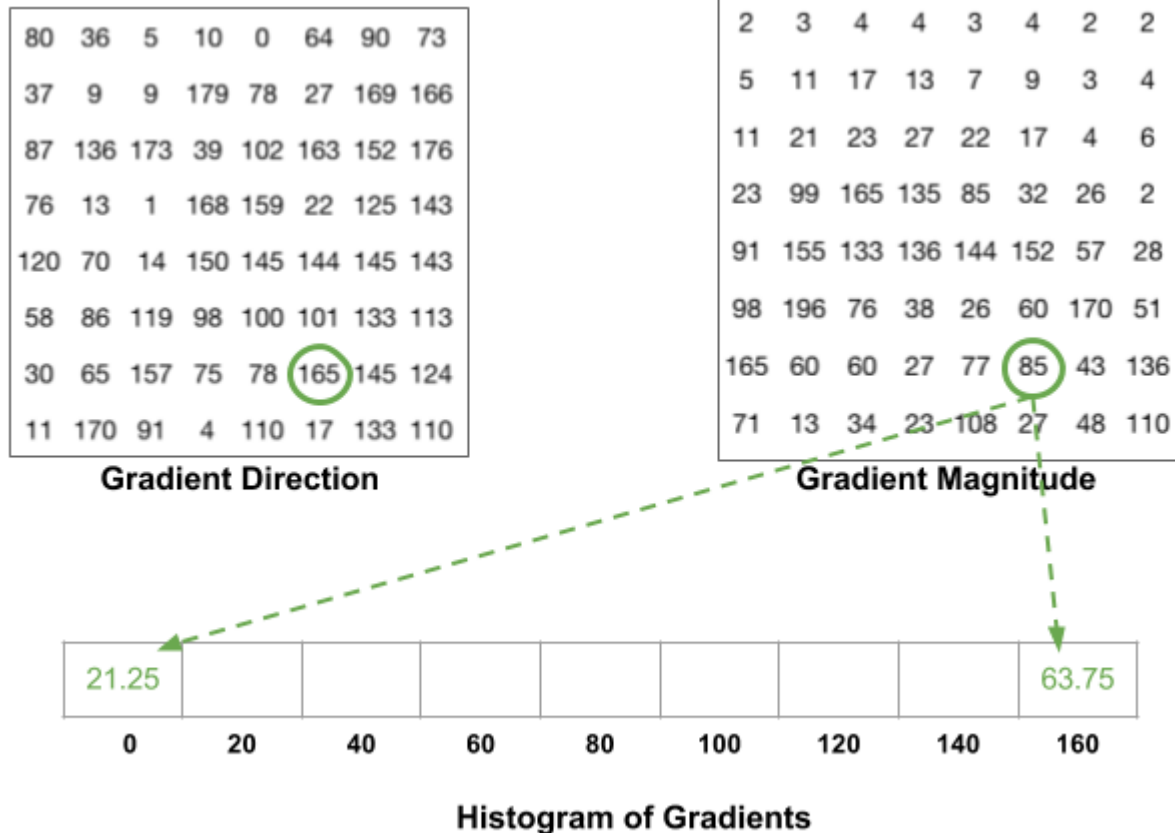
# 梯度方向直方图：直方图统计-II

- 将 0-180 度分为 9 个 bins
  - 统计：[0, 20, 40, ..., 160]
  - 将梯度幅值线性分配到与梯度方向最近的两个bin中



# 梯度方向直方图：直方图统计-III

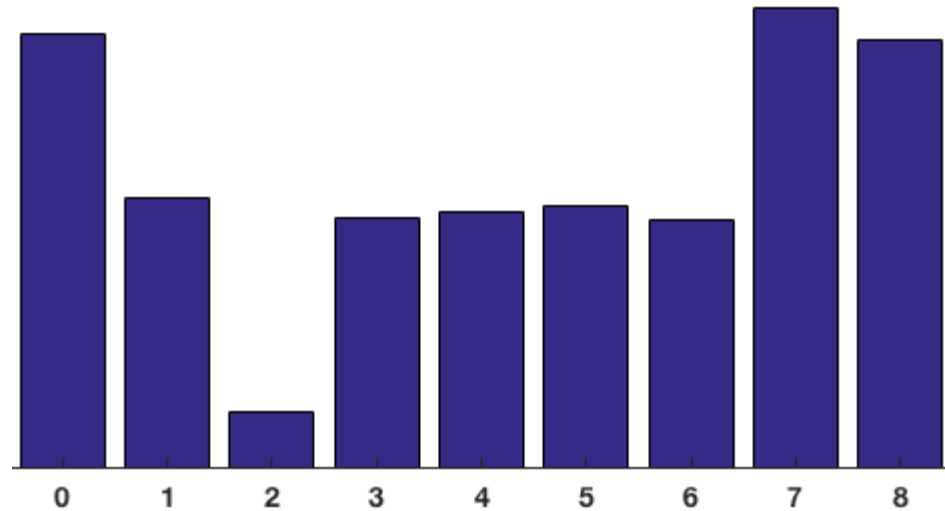
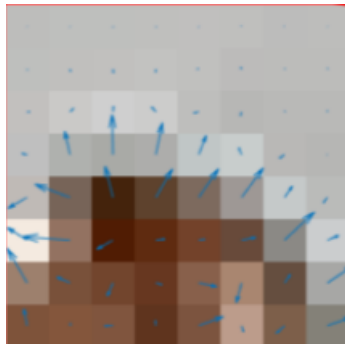
- 将 0-180 度分为 9 个 bins
  - 统计：[0, 20, 40, ..., 160]
  - 将梯度幅值线性分配到与梯度方向最近的两个bin中



# 梯度方向直方图：直方图统计-IV

## □ 每个 8x8 cell 的梯度方向直方图

- 所有像素贡献的累加
- 此处，0 度表示y轴方向，在 140-160-0(180) 度附近直方图的值较大，说明在这个 cell 中梯度方向朝上下的分量较多，有着较为明显的横向边缘。



# 块描述 (block descriptor)

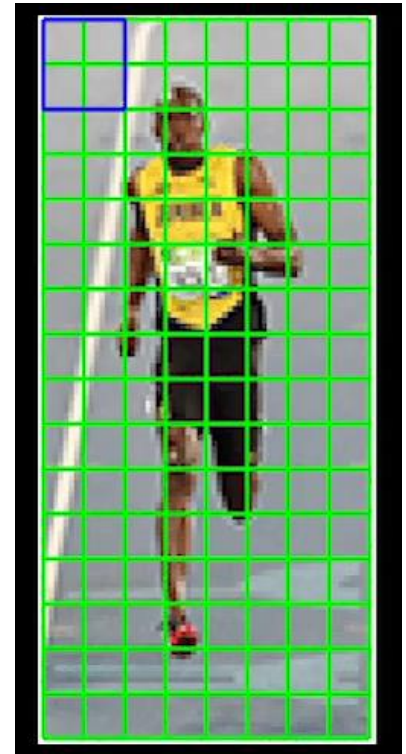
## □ block 归一化

- 大小：16x16 (包含4个cell)
- 目的：降低光照的影响
- 每个block由4个维度为9的向量组成，对block进行归一化即对该36维向量进行归一化
- block间有重叠

## □ 获取HOG特征向量

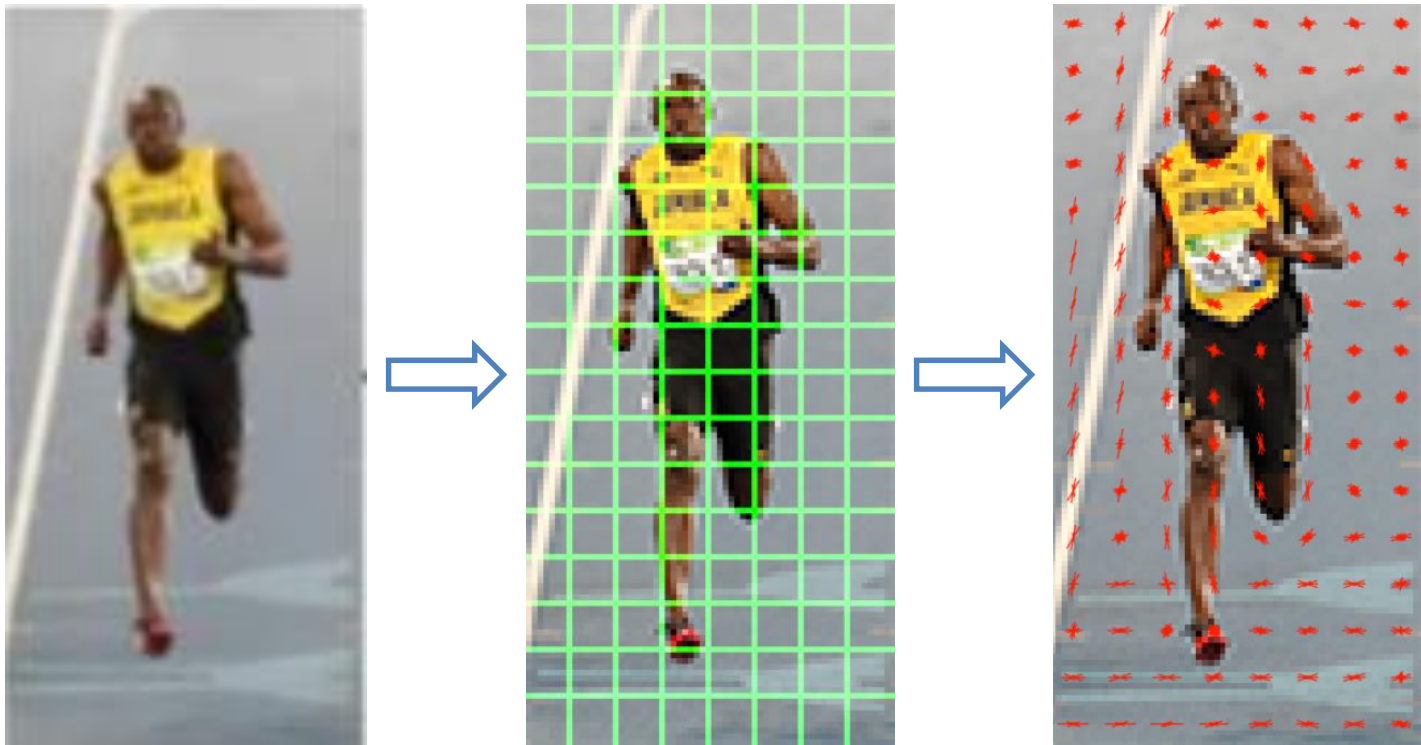
- 拼接所有block的向量即可
- 思考：图像(64x128)的HOG特征维度？
  - ✓ block 数：7x15
  - ✓ HOG特征维度：7x15x36=3780

**思考：**若一个block中所有像素的灰度值都经历相同参数  $(a, b)$  的线性变换： $f(x) = a \cdot x + b$ ，变换前后的HOG特征是否有变化？



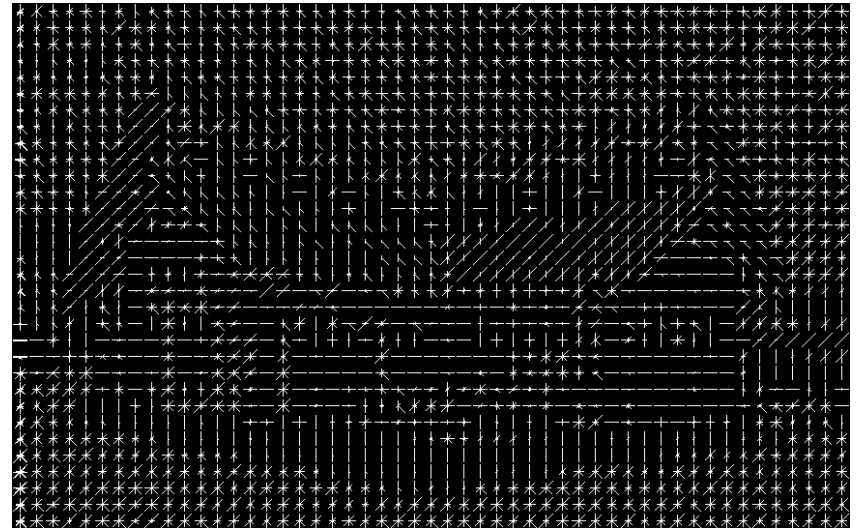
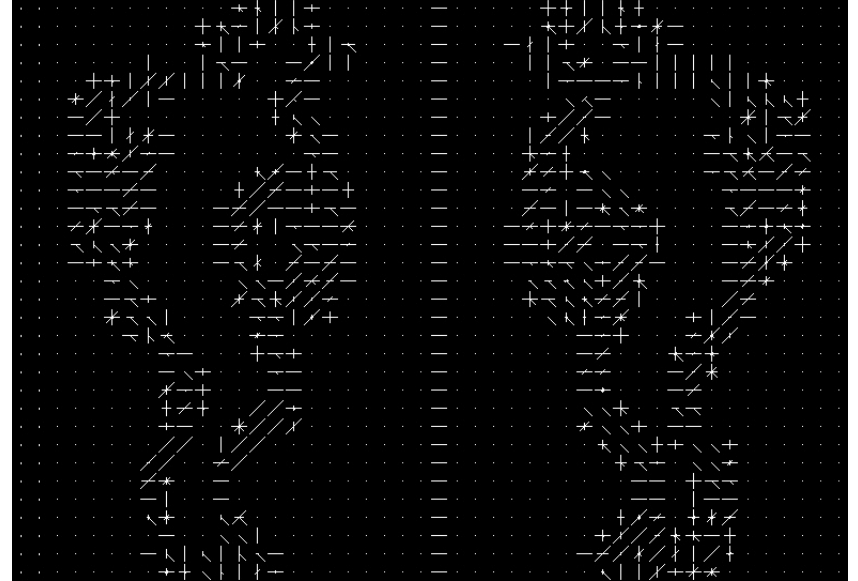
# HOG特征可视化

- 对每个 cell，画出经过block归一化后的直方图





# HOG: 示例





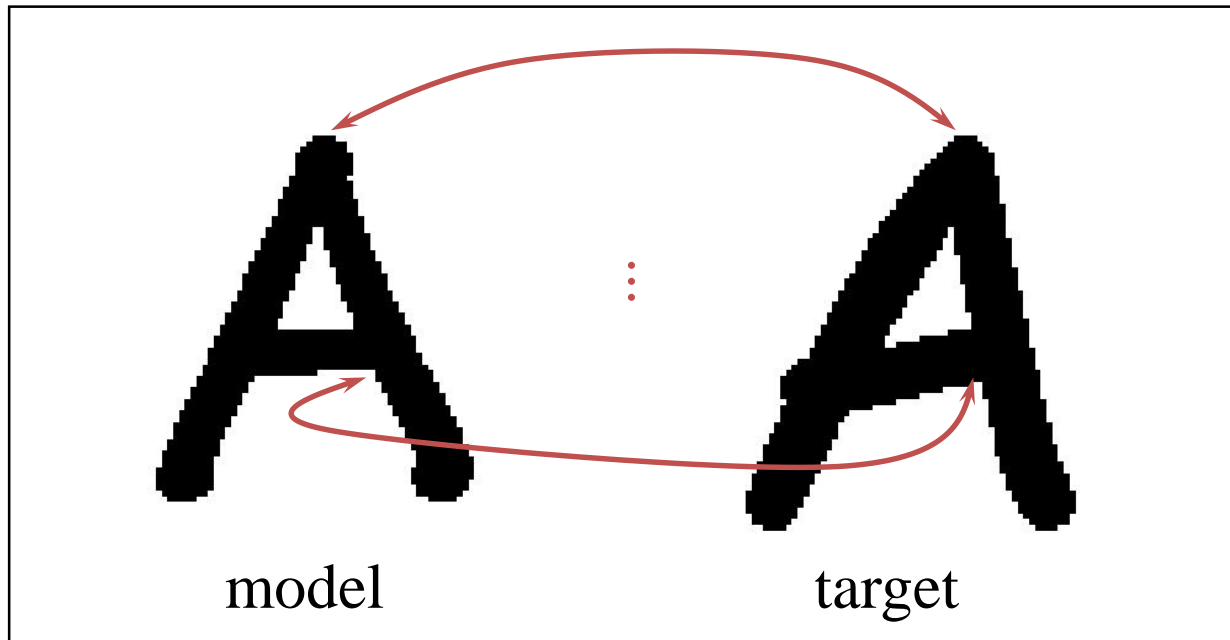


# 图像表达

- 基于全局特征的图像表达
- 基于局部特征的图像表达
  - 简单的局部视觉特征
    - ✓ 局部二值模式 (LBP)
    - ✓ 梯度方向直方图 (HOG)
    - ✓ 形状上下文 (Shape Context)
  - 基于关键点检测的局部视觉特征
    - ✓ 图像表达基本框架
    - ✓ 局部视觉特征描述
    - ✓ 特征编码与聚合

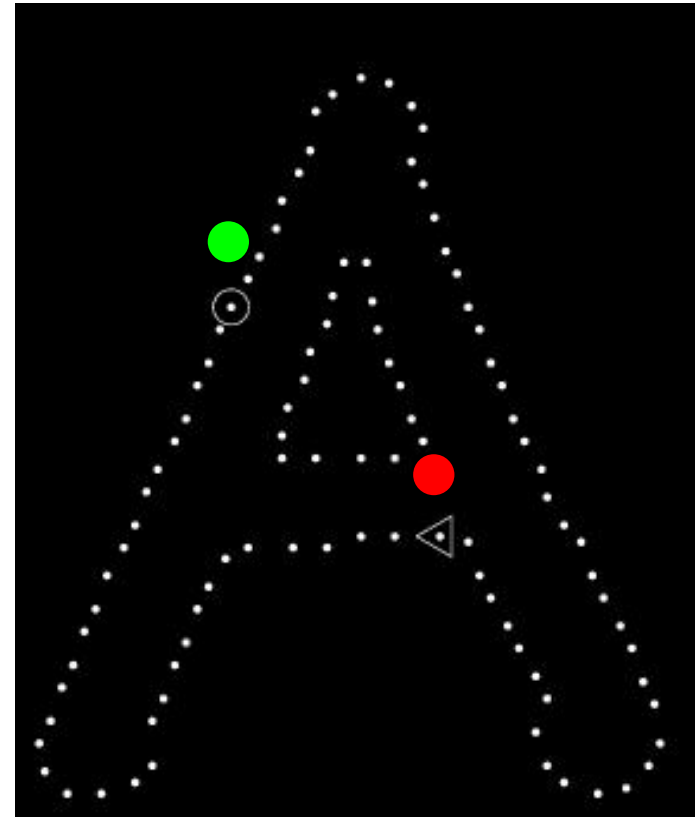
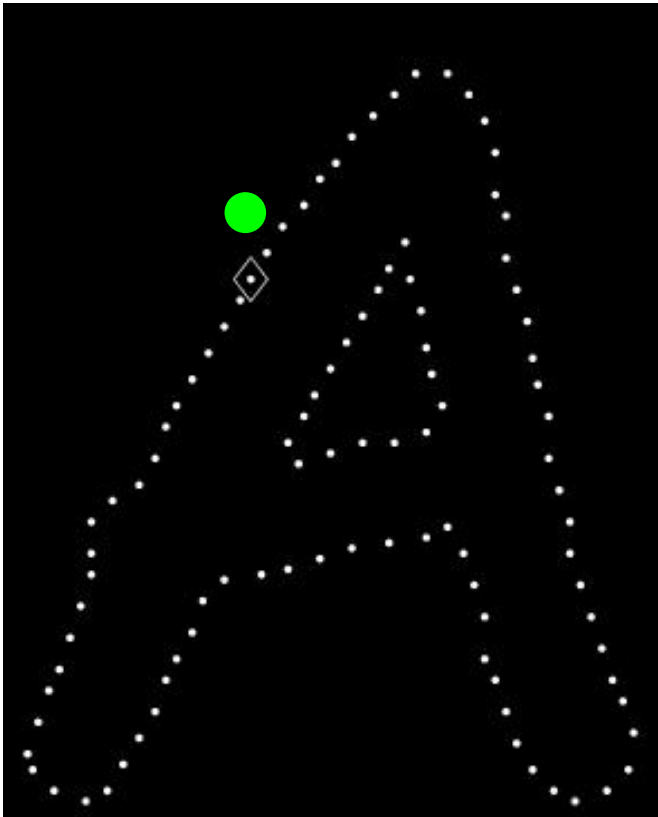
# 形状上下文 (Shape Context)

- 动机：如何度量两个二值图像中的目标相似性



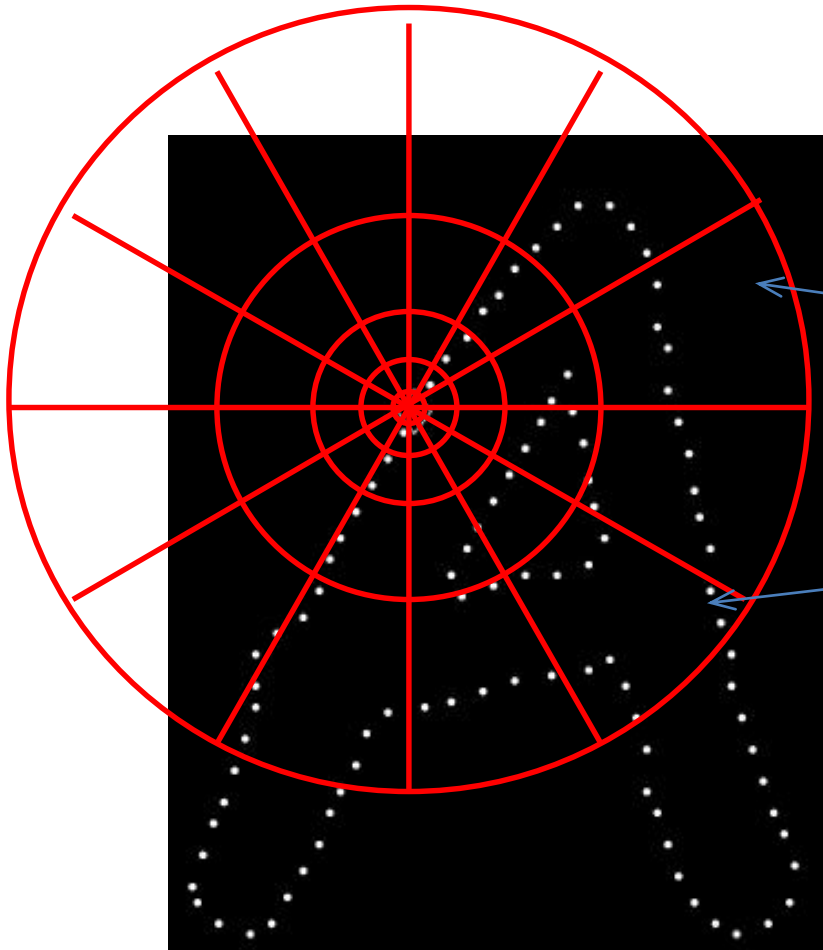
# 形状上下文：形状比较

- 提取二值图像中的物体轮廓，比较轮廓点的相似性，以及匹配关系



# 形状上下文

□ 以某一个轮廓点为参考原点，统计其他轮廓点的分布



检查落在每个bin区间中的点的数量:

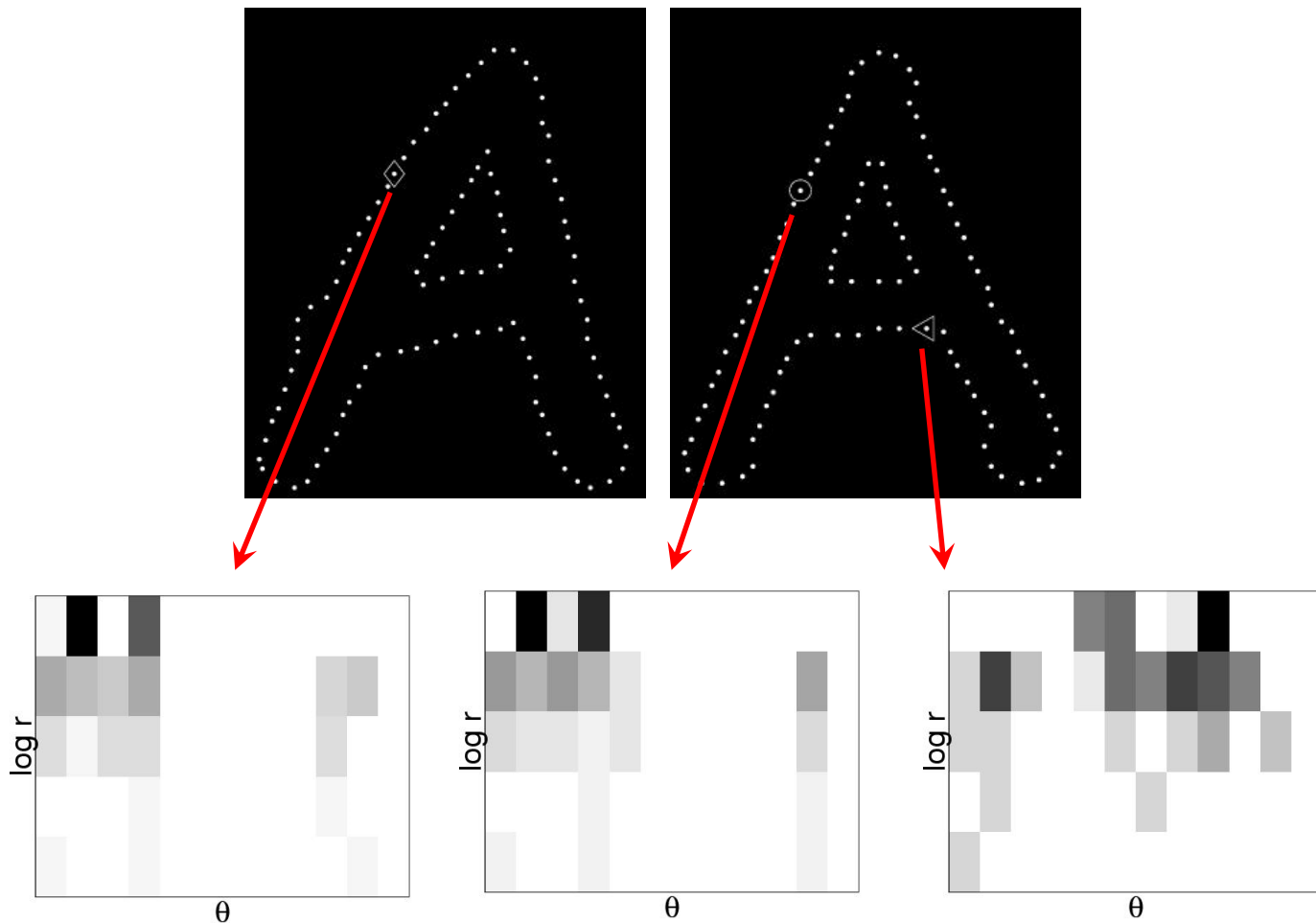
Count = 4

⋮

Count = 10

# 形状上下文

- 极坐标空间划分涉及两个变量 $r$ 和 $\theta$ ，轮廓点分布的直方图可以用二维矩阵表示



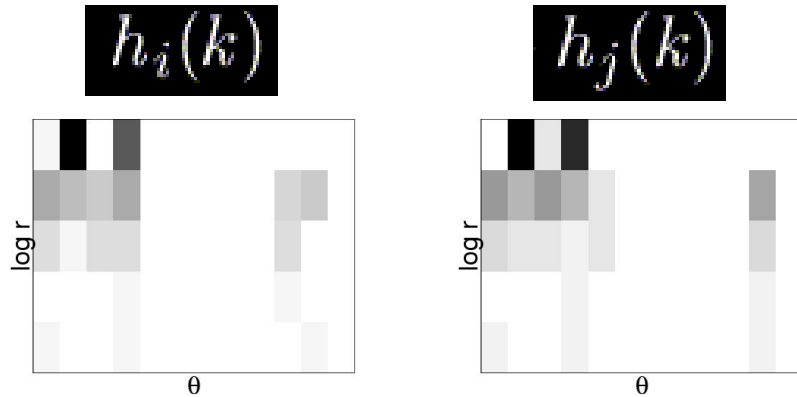


# 形状上下文：性质

---

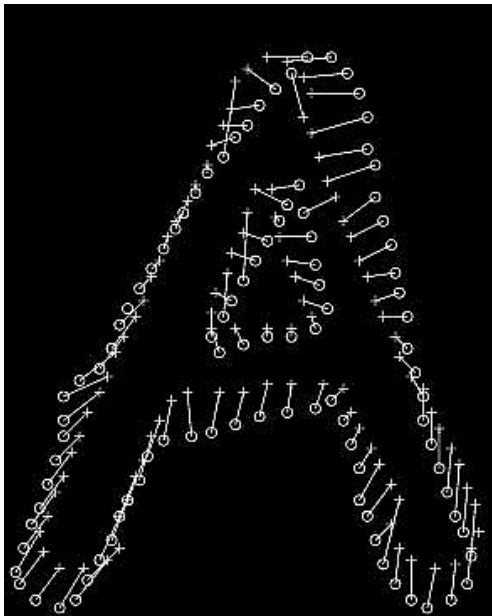
- 对平移变换具有不变性
- 通过选择合适的半径，可对缩放变换具有不变性
- 通过选择局部切向作为bin编号的参考方向，可对旋转具有不变性
- 可容忍小的仿射畸变

# 形状上下文：距离度量



基于Chi Squared distance, 计算两个轮廓的距离：

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



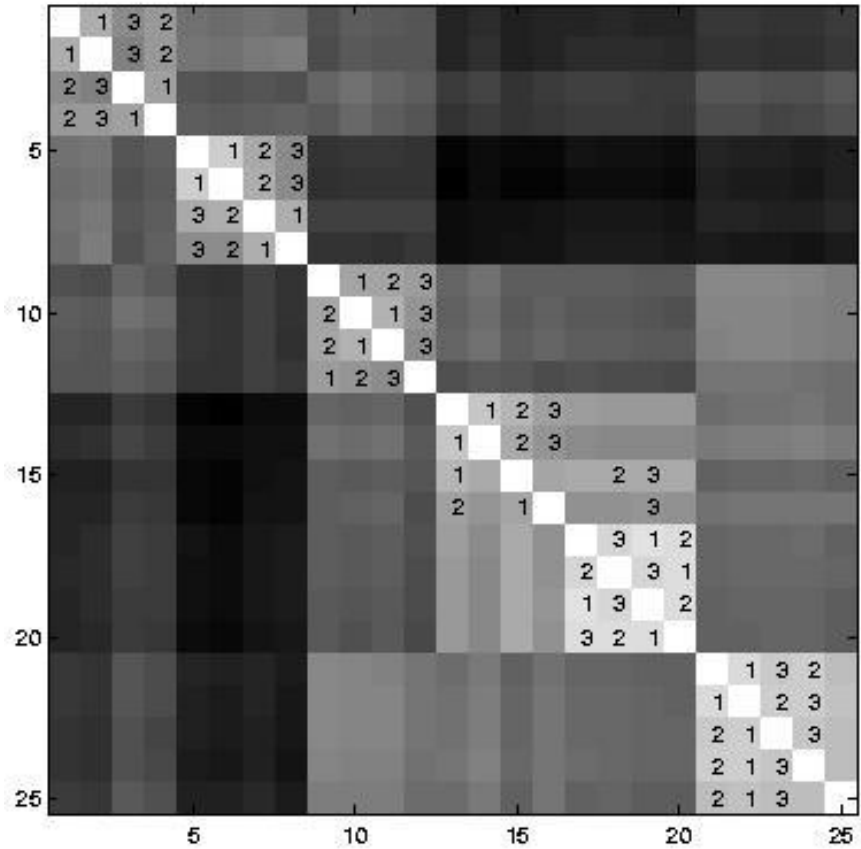
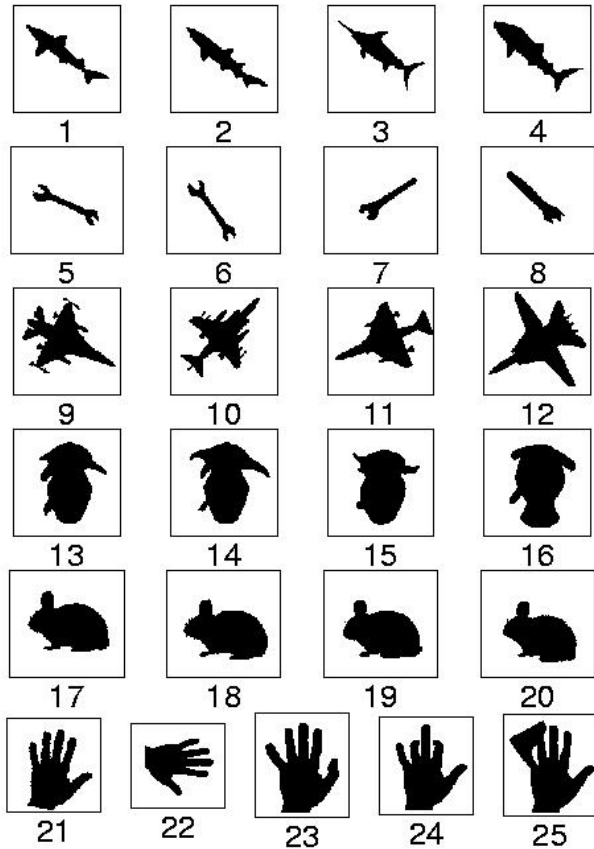
- 利用匹配代价  $C_{i,j}$ , 通过求解线性分配问题, 得到两组轮廓点的对应关系:

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

二分图匹配算法：匈牙利算法



# 形状上下文的应用-I



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

# 形状上下文的应用-II: 商标检索



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



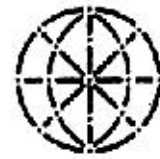
3: 0.153



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

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