

# 法律声明

---

□ 本课件包括：演示文稿，示例，代码，题库，视频和声音等，小象学院拥有完全知识产权的权利；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或机构不得盗版、复制、仿造其中的创意，我们将保留一切通过法律手段追究违反者的权利。

□ 课程详情请咨询

■ 微信公众号：小象

■ 新浪微博：ChinaHadoop



# 第10课 图像检索

---

## Image Retrieval

主讲人：张宗健

悉尼科技大学博士

主要研究方向： 计算机视觉、视觉场景理解、图像&语言、深度学习  
图像检索CbIR、Human ReID等

# 本章结构

---

## □ 相似检索

- 颜色、纹理、形状

- 局部特征点

- 词包 (Bag Of Visual Word)

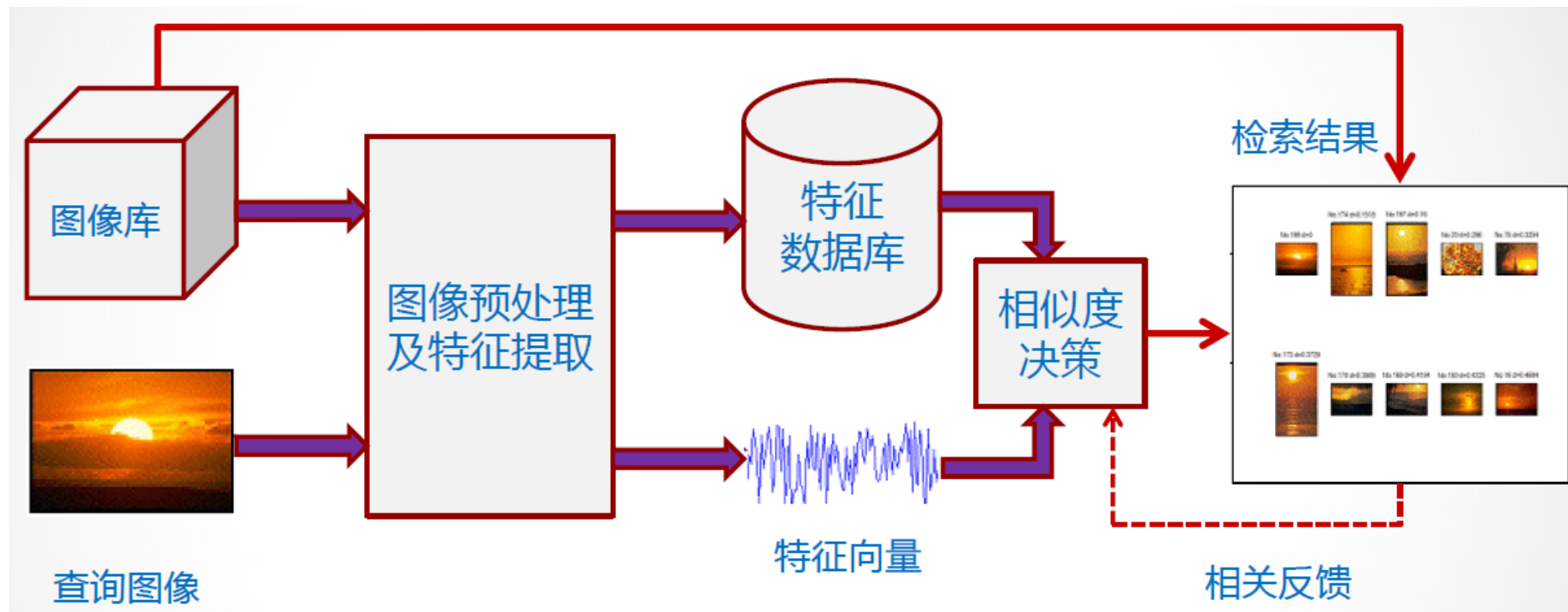
## □ 大数据集的索引加速

- KD-tree

- 局部敏感哈希 (Locality Sensitive Hash)

## □ 代码实践

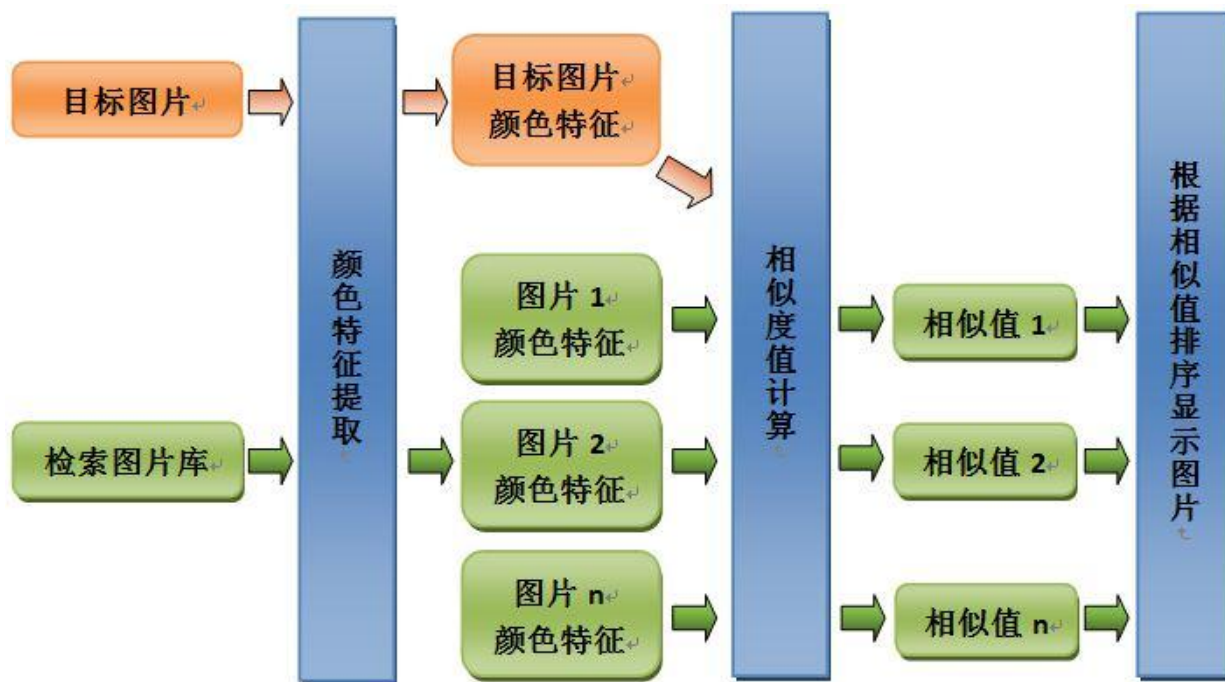
# 相似图像检索



# 相似颜色检索

## 算法结构

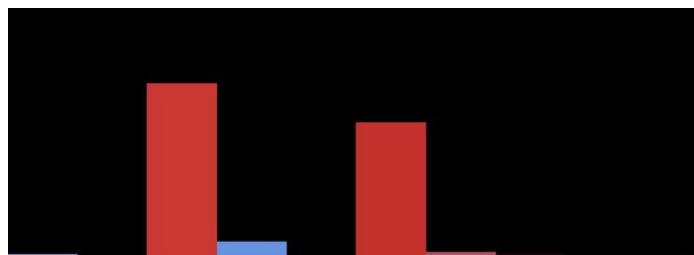
- 目标：实现基于**人类颜色感知**的相似排序
- 模块：**颜色特征提取** & **特征相似度计算**



# 相似颜色检索

## 颜色特征提取

- 目标：统计图片的**颜色成份**→**颜色聚类直方图**
- 方法：使用**K-means++**对图片**Lab**像素值进行聚类



# 相似颜色检索

---

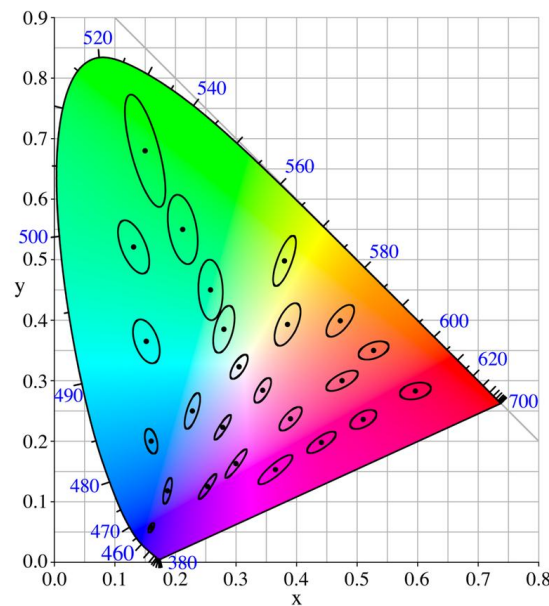
## 颜色特征相似度计算

- 颜色直方图距离
  - EMD (Earth Mover Distance)
  - 两个图片的颜色特征直方图之间的视觉相似度
  - 检索结果的排序依据
- 色差距离
  - CIEDE2000
  - Lab空间中两个颜色之间的视觉相似度
  - EMD距离的基础距离

# 相似颜色检索

## 色差距离

- 色差容忍度 (Tolerance)
  - 概念：无法感知的色差
  - 计算：色差小于JND(Just-Noticeable-Difference)阈值
  - 前提：感知均匀的色差距离
- CIE1931颜色空间
  - 容忍椭圆
  - 非感知均匀
- CIELab颜色空间
  - 视觉感知均匀的颜色模型
  - 均匀性更好的距离CIEDE2000





# 相似颜色检索

## 色差距离

- CIE制定，用于CIE-Lab空间的色差距离
- 欧式距离 → CIEDE1994 → CIEDE2000
- 改进依据：人类色差实验数据
- 改进目标：视觉均匀性
- 具体公式链接
  - [https://en.wikipedia.org/wiki/Color\\_difference#CIEDE2000](https://en.wikipedia.org/wiki/Color_difference#CIEDE2000)

$$\begin{aligned}\Delta E_{00}^{12} &= \Delta E_{00}(L_1^*, a_1^*, b_1^*; L_2^*, a_2^*, b_2^*) \\ &= \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2 + R_T \left(\frac{\Delta C'}{k_C S_C}\right) \left(\frac{\Delta H'}{k_H S_H}\right)}\end{aligned}$$

# 相似颜色检索

## EMD 距离

- 两个多维特征分布之间的非相似性度量
- 基于针对单特征的地面距离
- 传统运输问题

- 场景：多对多分配

$$\text{WORK}(P, Q, F) = \sum_{i=1}^m \sum_{j=1}^n f_{ij} d_{ij}$$

- 物资运送：多个供应商 → 多个需求客户
- 土堆搬运：多个土堆 → 多个土坑

- 约束

- 双方的节点总量相等
- 不同节点之间的成本各异

- 目标：完成分配的最小成本

$$f_{ij} \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n$$
$$\sum_{j=1}^n f_{ij} \leq w_{p_i} \quad 1 \leq i \leq m$$

$$\sum_{i=1}^m f_{ij} \leq w_{q_j} \quad 1 \leq j \leq n$$

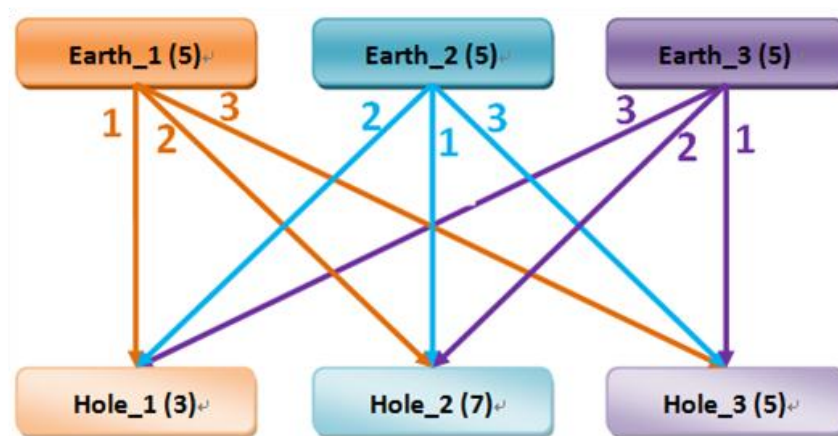
$$\sum_{i=1}^m \sum_{j=1}^n f_{ij} = \min \left( \sum_{i=1}^m w_{p_i}, \sum_{j=1}^n w_{q_j} \right),$$

# 相似颜色检索

## EMD 距离 直观解释

- 问题条件

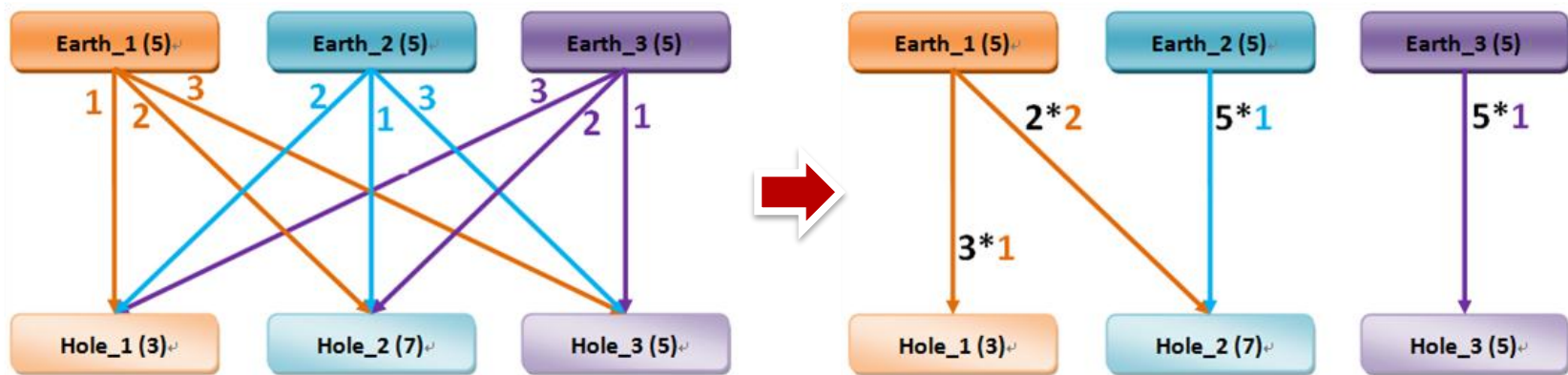
- 有三个土堆Earth\_1, Earth\_2, Earth\_3, 每个土堆都有5个单位的土量
- 有三个土坑Hole\_1, Hole\_2, Hole\_3, 每个土坑所能容纳的土量分别为3、7、5
- 不同的土堆和土坑之间的距离不同, 距离值分别是1, 2, 3
- 有一个搬运工, 一趟只能搬运1单位的土



# 相似颜色检索

## EMD 距离 直观解释

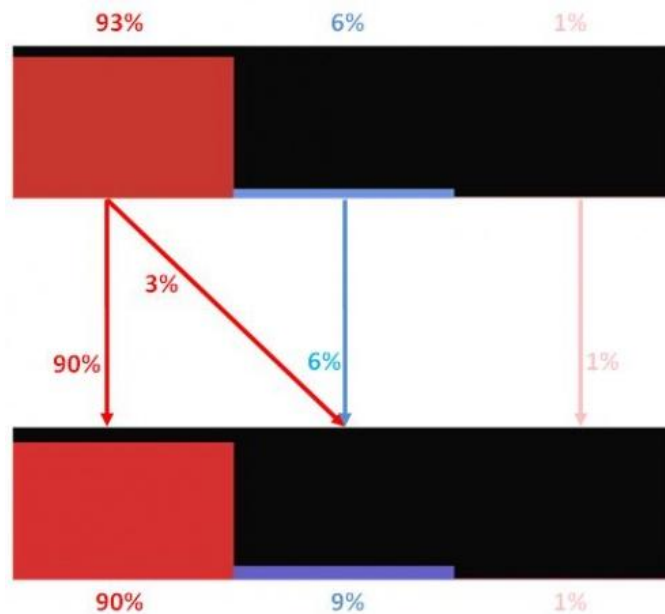
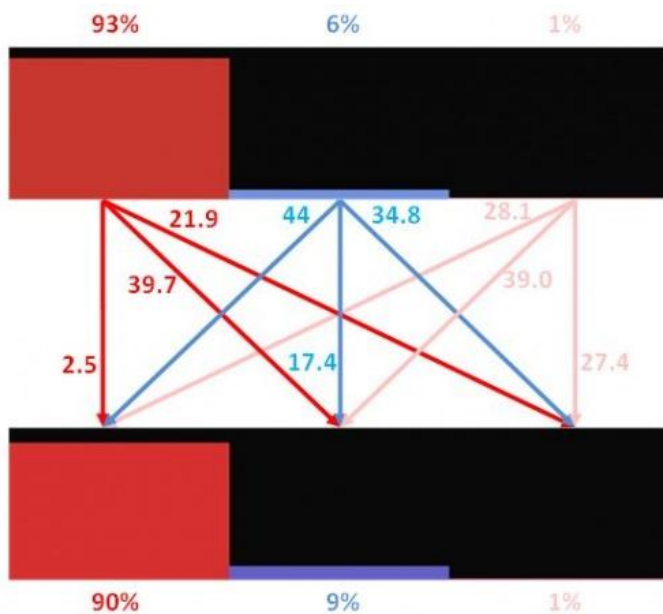
- 任务目标
  - 以最小的行走距离（EMD），将所有土堆运输到土坑处并填平
- 解决方案
  - 最小距离（EMD）： $3*1+2*2+5*1+5*1=17$ 
    - $E1 \rightarrow H1$ : 3                       $E1 \rightarrow H2$ : 2
    - $E2 \rightarrow H2$ : 5                       $E3 \rightarrow H3$ : 5



# 相似颜色检索

## EMD距离应用到颜色检索

- 多维分布→颜色特征直方图
- 地面距离→CIEDE2000

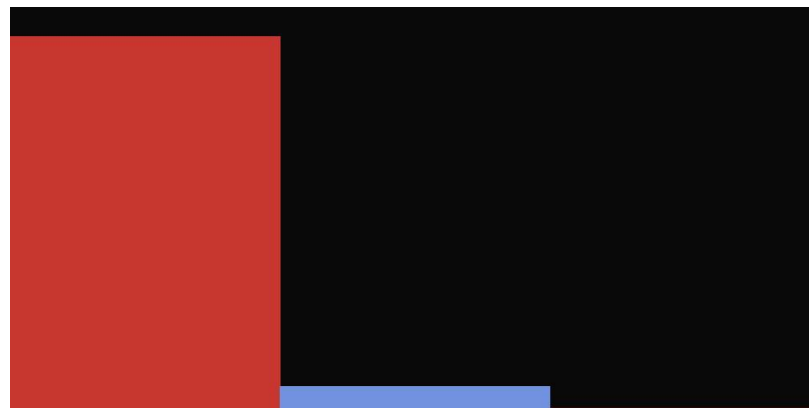


$$\text{EMD值: } 2.5 \times 90\% + 39.7 \times 3\% + 17.4 \times 6\% + 27.4 \times 1\% = 4.795$$

# 相似颜色检索

## 流程演示

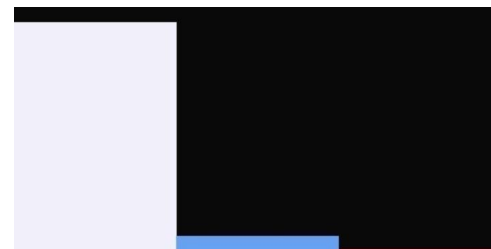
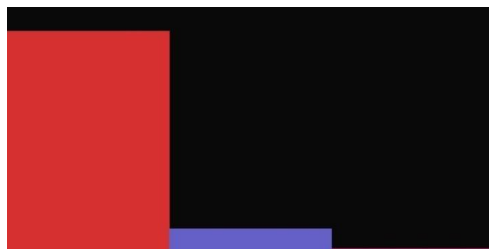
- Step1: 搜索图片的颜色特征提取



# 相似颜色检索

## 流程演示

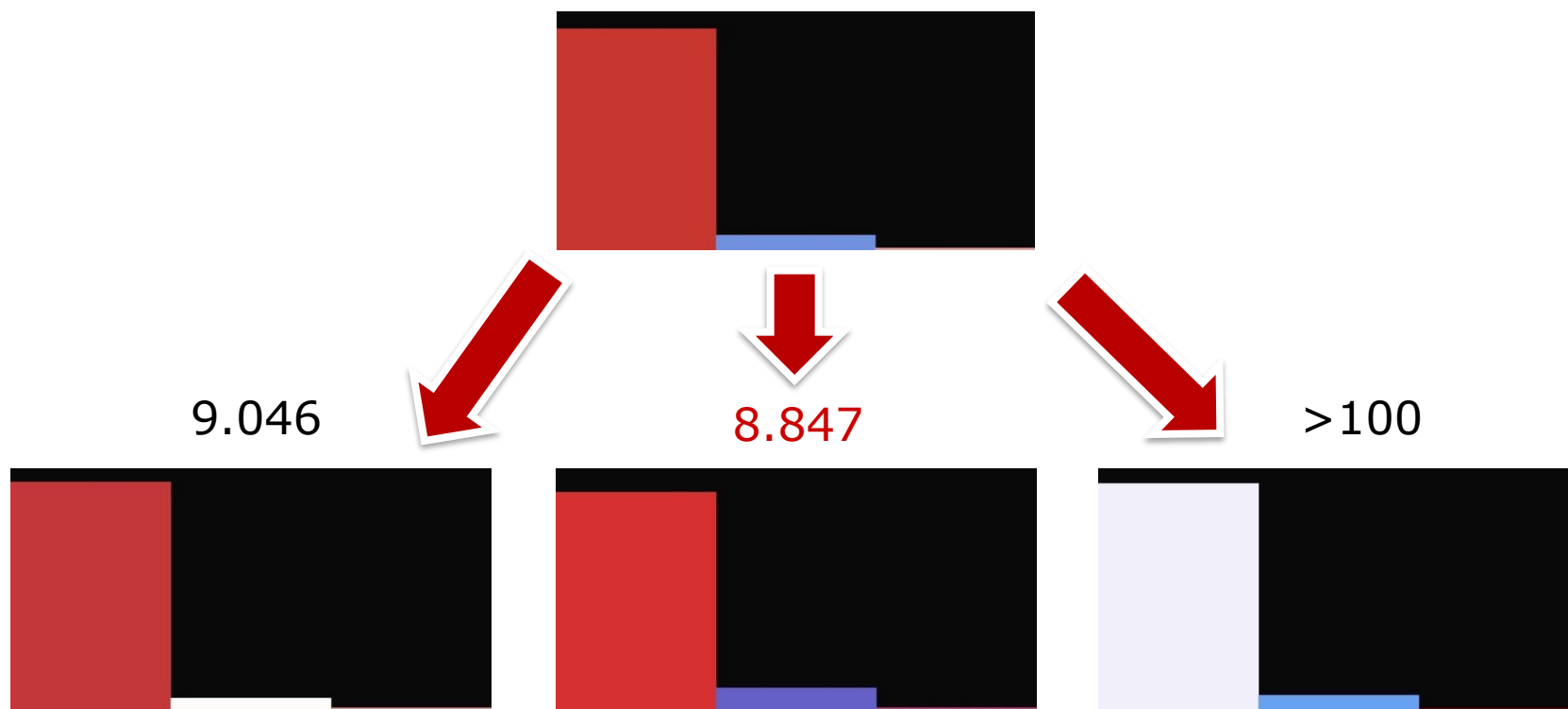
- Step2: 图片库的颜色特征提取



# 相似颜色检索

## 流程演示

- Step3: 颜色相似度(EMD)计算



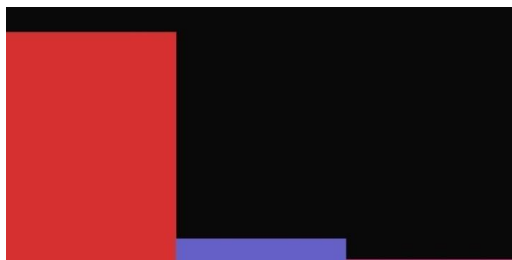


# 相似颜色检索

## 流程演示

- Step4: 根据相似度排序

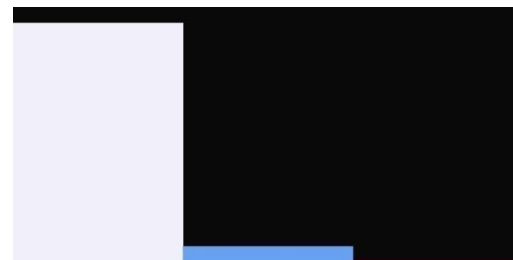
1<sup>st</sup>: 8.847



2<sup>nd</sup>: 9.046



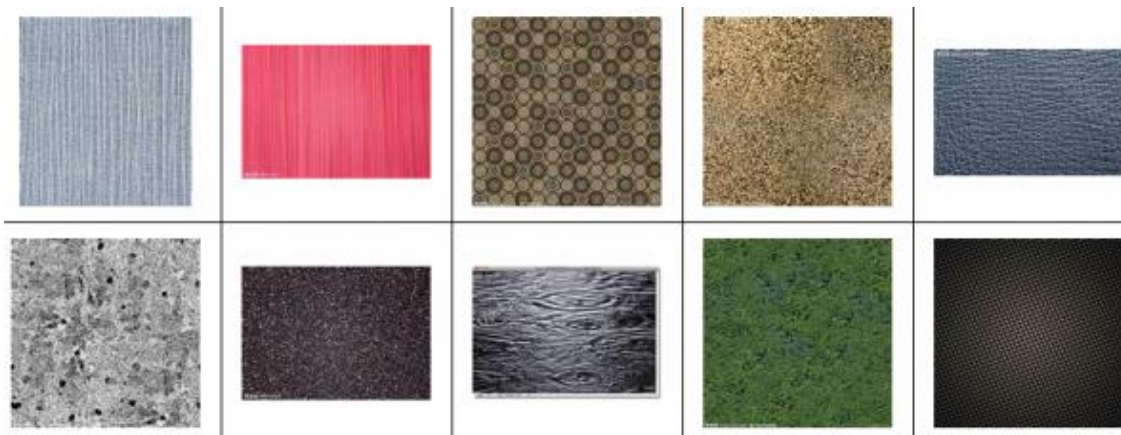
3<sup>rd</sup>: >100



# 相似纹理检索

## 纹理 (texture)

- 重复模式：元素或基元按一定规则排列
- 纹理特征
  - 反映图像中同质现象的视觉特征
  - 所有物体表面共有的内在特性
  - 包含了物体表面结构组织排列的重要信息



# 相似纹理检索

---

## 算法结构

- 目标：实现基于人类纹理感知的相似排序
- 模块（与相似颜色检索类似）
  - 纹理特征提取
    - 特征空间：多方向、多尺度Gabor滤波器组
    - 特征计算：Kmeans++聚类直方图
  - 特征相似度计算
    - 纹理聚类直方图：EMD
    - 纹理距离：L2

# 相似纹理检索

## Gabor滤波器组

- Gabor 滤波器组类似于人类的生物视觉系统
  - 多频率/尺度
  - 多方向
- Gabor 滤波器
  - 频域：属于加窗傅立叶变换
  - 空域：一个高斯核函数和正弦平面波的乘积

复数: 
$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \exp\left(i\left(2\pi\frac{x'}{\lambda} + \psi\right)\right)$$

实部: 
$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi\frac{x'}{\lambda} + \psi\right)$$

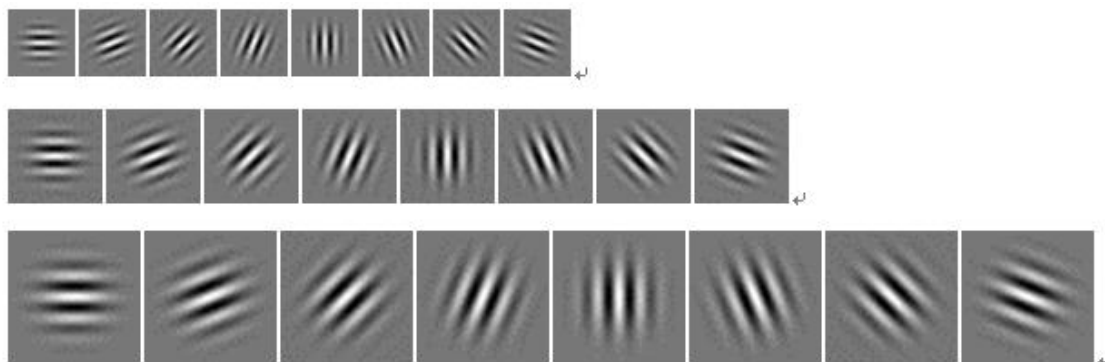
虚部: 
$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \sin\left(2\pi\frac{x'}{\lambda} + \psi\right)$$

其中: 
$$x' = x \cos \theta + y \sin \theta, y' = -x \sin \theta + y \cos \theta$$

# 相似纹理检索

## Gabor滤波器组

- 6频率（尺度）
  - 频率：1, 2, 3, 4, 5, 6
  - 尺寸：25, 35, 49, 69, 97, 137
- 8方向
  - 0, 22.5, 45, 67.5, 90, 112.5, 135, 157.5



# 相似纹理检索

---

## Gabor纹理特征提取

- 彩色图片灰度化
- 提取灰度图的Gabor滤波器特征
  - 6频率（尺度）、8方向的Gabor,
  - 48个同尺寸的特征图
  - 每个像素对应48维的Gabor特征向量
- 使用Kmeans++聚类所有像素Gabor特征
  - K值(10)根据数据集纹理复杂度而定
  - 使用KD-tree版加速

# 相似纹理检索

## Gabor纹理特征提取

- Gabor卷积操作加速
  - FFT
  - 采用"图片尺寸缩小替代Gabor模板尺寸增大"的策略

Gabor频率(尺度)	1	2	3	4	5	6
Gabor模板尺寸	25x25	35x35	49x49	69x69	97x97	137x137
原图尺寸	400x400	400x400	400x400	400x400	400x400	400x400



Gabor频率(尺度)	1	2	3	4	5	6
Gabor模板尺寸	25x25	35x35	25x25	35x35	25x25	35x35
原图尺寸	400x400	400x400	200x200	200x200	100x100	100x100

# 相似纹理检索

## Gabor纹理特征提取可视化













- 10个纹理基元

















# 相似纹理检索

## 效果图1

相似度为: 49.6532402039  相似商品推荐 经典格型情侣休闲衬衫 (女款) 苏格兰格	相似度为: 72.2671432495  相似商品推荐 法兰绒经典百搭长袖衬衫 明暗红蓝彩格	相似度为: 73.3022613525  相似商品推荐 VDF-法兰绒经典百搭休闲长袖衬衫 女款 浅卡其粉蓝格	相似度为: 73.3817672729  相似商品推荐 VDF-法兰绒经典百搭休闲长袖衬衫 女款 海蓝红白格纹	相似度为: 74.0381774902  相似商品推荐 VDF-法兰绒经典百搭休闲长袖衬衫 女款 经典灰紫格纹	相似度为: 74.7108535767  相似商品推荐 户外温暖休闲衬衫 (女款) 咖啡色格子
相似度为: 74.8854064941  相似商品推荐 VDF-法兰绒经典百搭休闲长袖衬衫 女款 朱红蓝白格纹	相似度为: 75.3982925415  相似商品推荐 户外蓄热保温格子衬衫 (情侣款) 女款 马赛克紫	相似度为: 75.5169143677  相似商品推荐 经典格型情侣休闲衬衫 (女款) 红蓝格	相似度为: 76.1359939575  相似商品推荐 [大码] 法兰绒格子休闲衬衣 (女) 红蓝小格纹	相似度为: 76.9539108276  相似商品推荐 薄法兰绒西部风尚休闲长袖衬衫 女款 墨绿玫红白格纹	相似度为: 77.1196060181  相似商品推荐 法兰绒经典百搭长袖衬衫 淡蓝灰格纹

# 相似纹理检索

## 效果图2

相似度为: 42.7042312622  相似商品推荐 名媛气质墨墨印花连衣裙 黑色印花	相似度为: 53.9294242859  相似商品推荐 Y领碎花连衣裙 黑底印花	相似度为: 54.8440284729  相似商品推荐 古典气质小碎花连衣裙 黑底印花	相似度为: 55.7197875977  相似商品推荐 复古碎花荷叶袖连衣裙 墨绿印花	相似度为: 57.1203346252  相似商品推荐 简约印花连衣裙 黑底印花	相似度为: 57.4645080566  相似商品推荐 鸽子印花淑女连衣裙 蓝色印花
相似度为: 58.3793830872  相似商品推荐 鸽子印花淑女连衣裙 粉色印花	相似度为: 58.7895545959  相似商品推荐 Sexy豹纹宽松连衣裙 绿色印花	相似度为: 59.0268325806  相似商品推荐 热带风情V领吊带连衣裙 红灰印花	相似度为: 59.0493580933  相似商品推荐 可爱小花朵V领连衣裙 黄色印花	相似度为: 59.2194747925  相似商品推荐 缤纷印花方领高腰连衣裙 白底印花	相似度为: 59.4356994629  相似商品推荐 设计师款和服连衣裙 米色

# 相似纹理检索

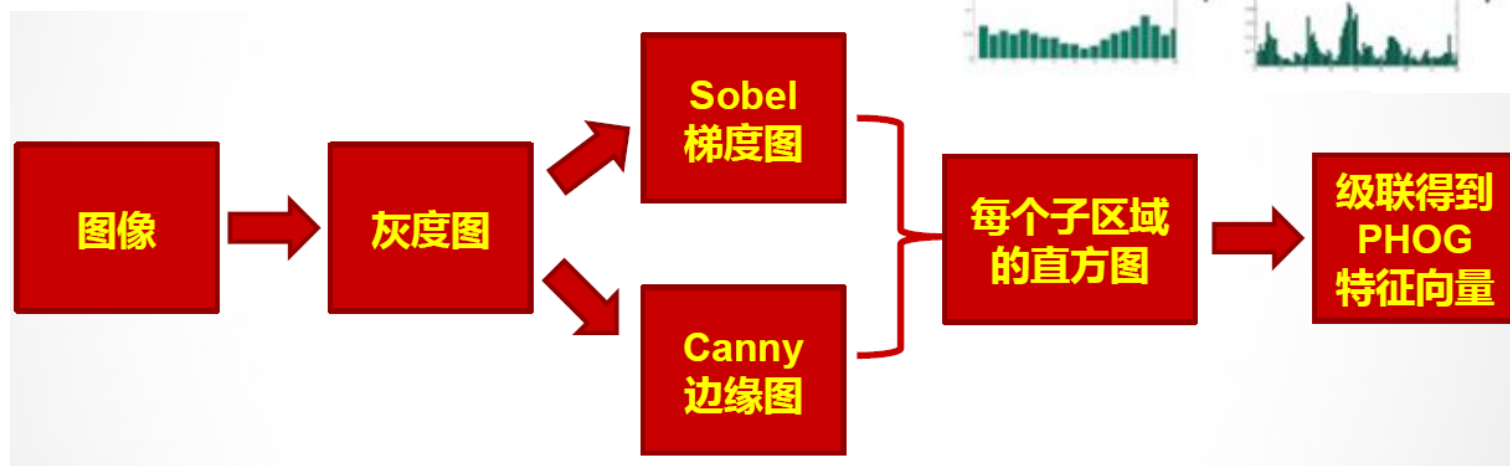
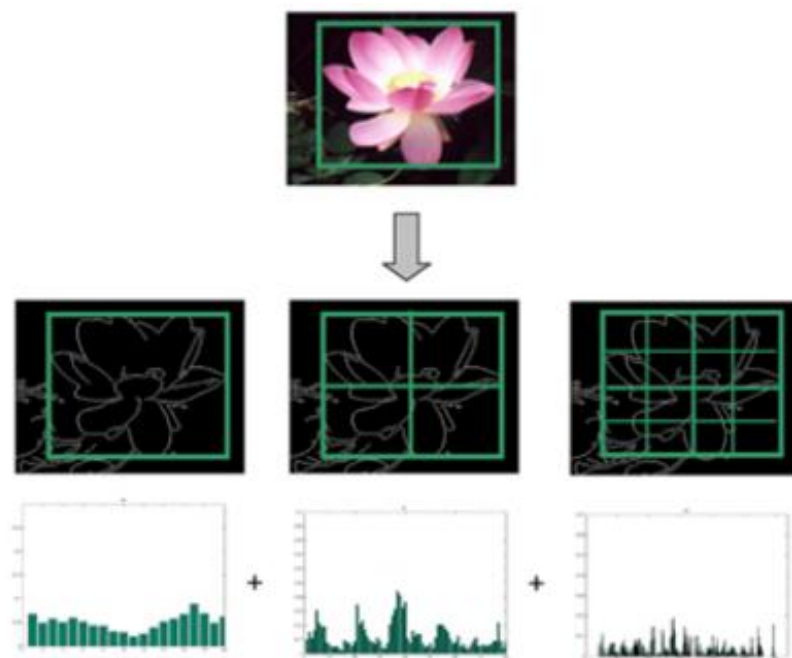
## 效果图3



# 相似形状检索

## Phog形状特征提取

- 金字塔梯度方向直方图
  - 网格：1x1, 2x2, 4x4
  - 直方图方向数量：9
  - 维数：189 = (1+4+16)x9





# 相似形状检索

## Phog形状特征相似度计算

- 标准化欧氏距离
  - $S_i$  为样本集特征中每一维对应的标准差

$$Dist(P, Q) = \sqrt{\sum_i \left(\frac{P_i - Q_i}{S_i}\right)^2}$$

- 直方图相交(Histogram Intersection)

$$Sim(P, Q) = \sum_{i=1}^{i=n} \min(P_i, Q_i)$$

# 相似形状检索

效果图1



# 相似形状检索

效果图2



# 相似局部特征检索

## 局部特征点特征提取

- 检测出所有
  - 局部特征点
  - 特征描述子
- SIFT特征点
  - SURF
  - Color SIFT
  - Affine SIFT
  - .....

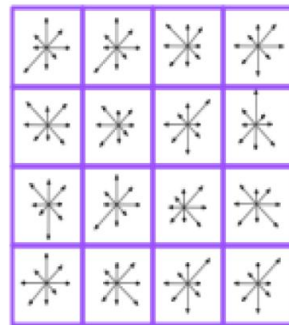
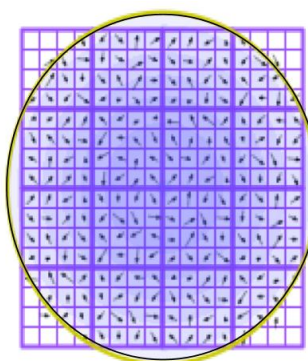
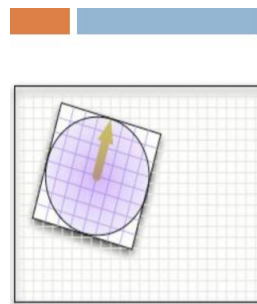
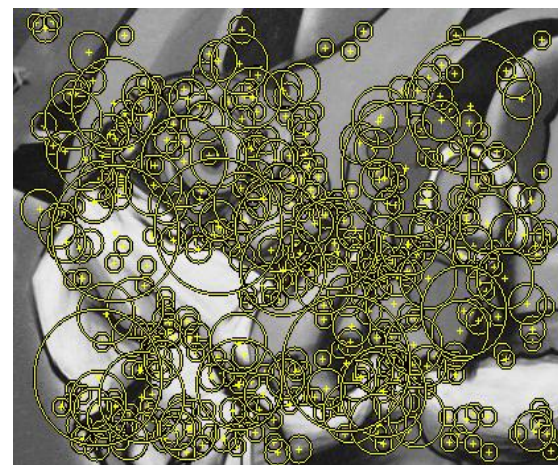


Image gradients

Keypoint descriptor

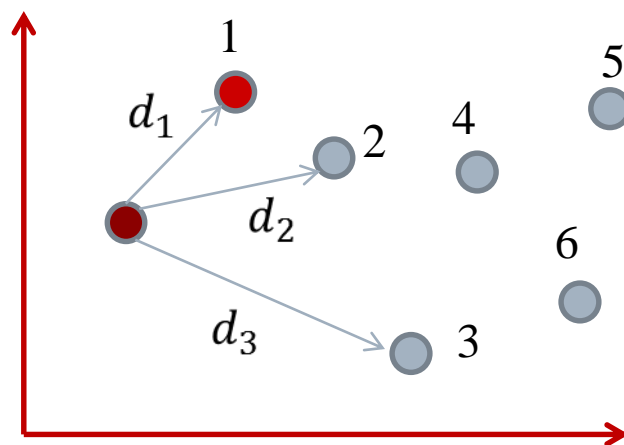
4x4个8方向梯度方向直方图 = 128维描述子



# 相似局部特征检索

## SIFT描述子之间的相似度匹配

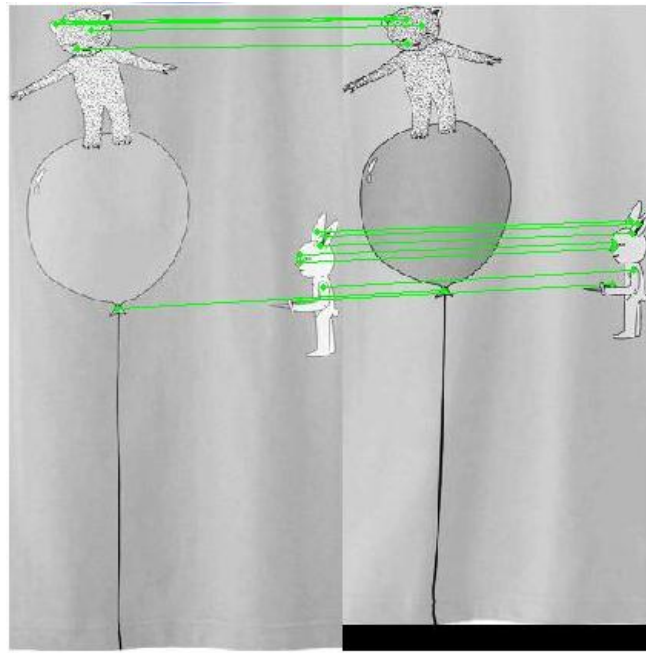
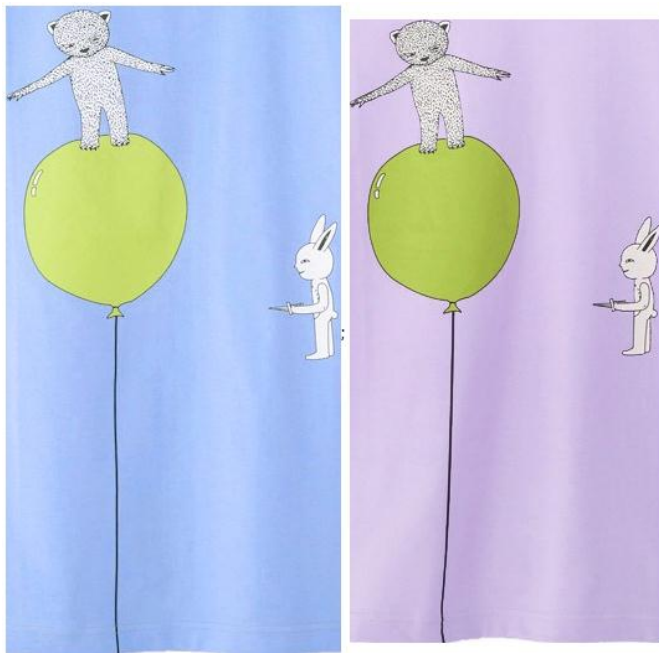
- 基于欧式距离的最邻近
  - $d_1 < d_2 < d_3 < d_4 < \dots$
- 比率条件
  - $d_1/d_2 < thresh$



# 相似局部特征检索

## 图之间的相似度匹配

- 两个图SIFT点集之间的匹配对数
- 双向匹配



# 相似局部特征检索

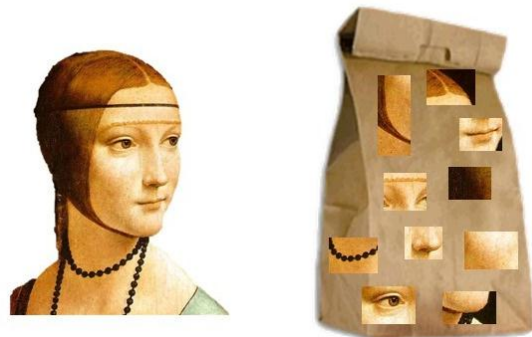
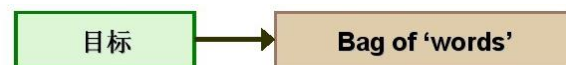
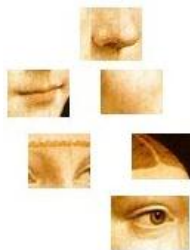
## 效果图



# 词包Bag Of Visual Word

## 视觉词汇的字典

- 由图片集的所有视觉词汇构成
- 视觉词汇的物理含义未知
- 不是现成，需要构建
  - 特征检测
    - 特征点：SIFT、SURF等
  - 特征表示
    - SIFT描述子、颜色、纹理等
  - 字典生成
    - Kmeans等聚类



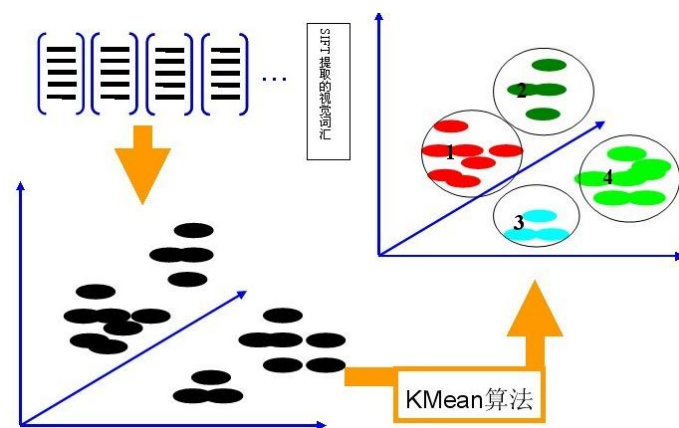
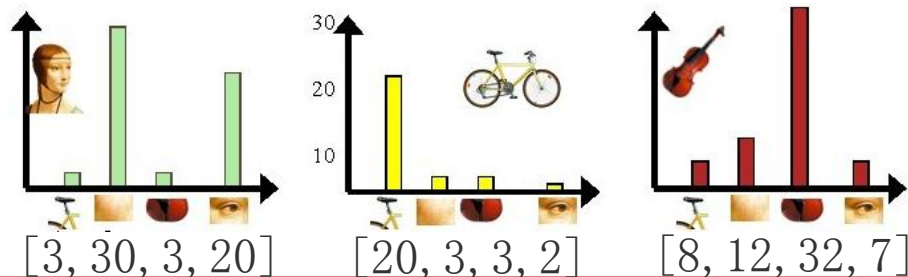
# 词包Bag Of Visual Word

## SIFT视觉词汇的字典应用

- 利用SIFT算法提取图片集中所有视觉词汇



- 利用Kmeans算法对所有词汇聚类，收缩为字典
- 基于字典编码图片特征
  - 词汇频数直方图
  - 最邻近词汇
  - 特征相似距离：L1、L2



# 大数据集的索引加速

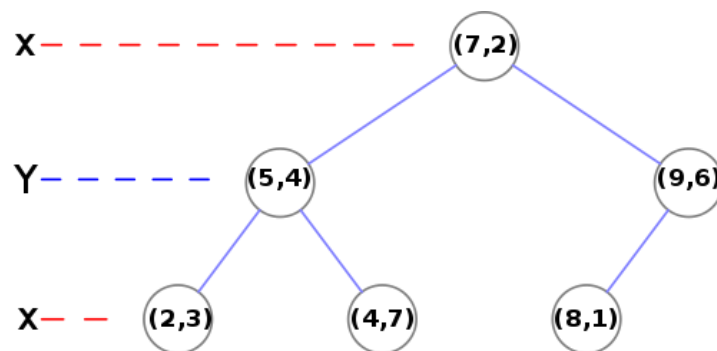
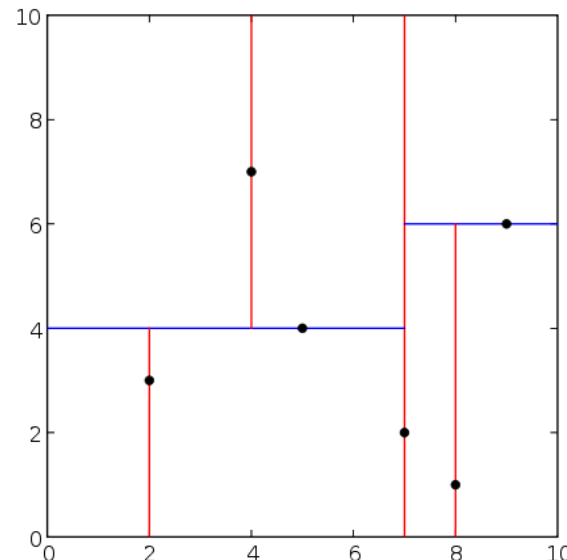
## KD-Tree

- 一种用于 **多维度检索的二叉平衡树** (1975)
  - 输入:  $N$ 个**D**维空间的数据点
  - 构建过程
    1. 确定split值: 方差最大的维度
    2. 确定分割点: 在split维度上的中值点, 首次为根节点
    3. 确定分割面: 通过分割点, 垂直split维度的超平面
    4. 确定左右子树
      - **左**子树: split维度上**小**于分割点
      - **右**子树: split维度上**大**于分割点
  - 迭代1-4, 直到空间**只包含一个数据点**

# 大数据集的索引加速

## KD-Tree

- 输入
  - $(2,3), (5,4), (4,7), (9,6), (7,2), (8,1)$
- 第一轮:
  - $\text{Var}=(6.96,5.37) \rightarrow \text{Split} = 1(x)$
  - $(7,2)$ 在x轴上是中点  $\rightarrow$  根节点
  - 左子树:  $(2,3), (5,4), (4,7)$ 
    - 第二轮:
      - $\text{Split} = 2(y)$
      - $(5,4)$ 是y轴上中点
      - $(2,3)$ 为左子树,  $(4,7)$ 为右子树
  - 右子树:  $(8,1), (9,6)$ 
    - 第二轮...



# 大数据集的索引加速

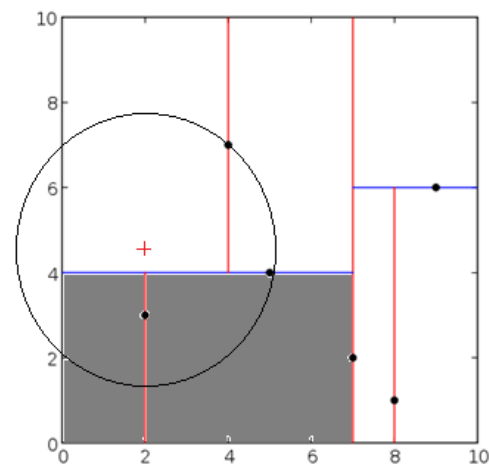
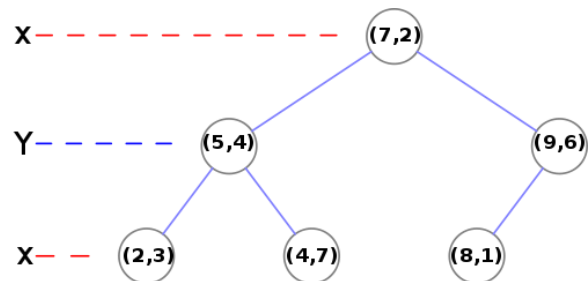
## KD-Tree

- 最邻近查询

- 路径：类似二叉树搜索，从根节点开始，根据每个维度的split维进行左右子树的查询，直到叶子节点
- 回溯：检查查询路径上节点的另一半子空间是否有距离更近的点

- 查询点(2, 4.5)

- 路径：(7, 2) → (5, 4) → (4, 7)
- 回溯：(5, 4) → (2, 3)

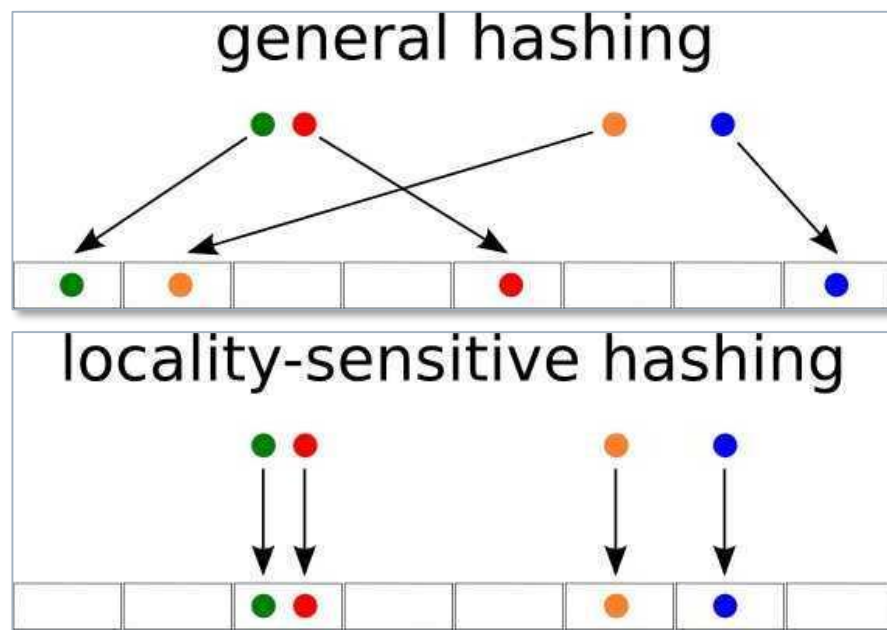




# 大数据集的索引加速

## 局部敏感哈希 (Locality Sensitivity Hashing)

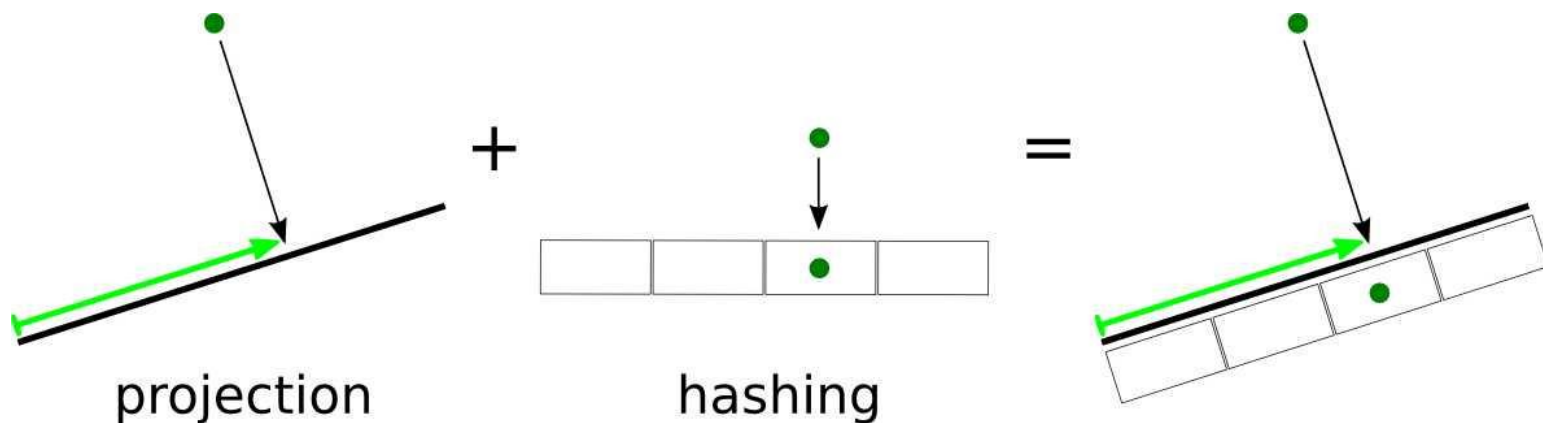
- 海量高维特征的KNN/RNN
- 利用collisions
  - 普通哈希避免collisions
- 支持距离
  - Hamming
  - Lp Norm (L1, L2)
  - Jaccard
  - Cosine
  - .....



# 大数据集的索引加速

## 局部敏感哈希 (Locality Sensitivity Hashing)

- 基于概率的高维数据降维
- 哈希思想
  - 临近的点有很高的概率会被映射到同一个桶中
  - 不临近的点，则概率低



# 大数据集的索引加速

## 局部敏感哈希 (Locality Sensitivity Hashing)

- $(R, cR, P_1, P_2)$ 局部敏感性

Definition 2.3 (Locality-sensitive hashing). A family  $\mathcal{H}$  is called  $(R, cR, P_1, P_2)$ -sensitive if for any two points  $p, q \in \mathbb{R}^d$ .

- if  $\|p - q\| \leq R$  then  $\Pr_{\mathcal{H}}[h(q) = h(p)] \geq P_1$ ,
- if  $\|p - q\| \geq cR$  then  $\Pr_{\mathcal{H}}[h(q) = h(p)] \leq P_2$ .
- $p, q$ : 任意两个原始空间的向量
- $h(p), h(q)$ : 哈希函数 $h$ 映射后的标量值
- 条件1:  $p$ 与 $q$ 的距离小于 $R$ 时,  $h(p)=h(q)$ 的概率至少为 $P_1$
- 条件2:  $p$ 与 $q$ 的距离大于 $cR$ 时,  $h(p)=h(q)$ 的概率最大为 $P_2$
- $c > 1$
- $P_1 > P_2$

# 大数据集的索引加速

## 局部敏感哈希 (Locality Sensitivity Hashing)

- 构建LSH索引

1. 重构LSH函数 $g$

- 串接 $k$ 个具有 $(R, cR, P_1, P_2)$ 局部敏感性的哈希原子函数

$\mathcal{G} = \{g : S \rightarrow U^k\}$  such that  $g(v) = (h_1(v), \dots, h_k(v))$ , where  $h_i \in \mathcal{H}$ .

2. 独立、随机选取 $L$ 个LSH函数

$g_1, \dots, g_L$  from  $\mathcal{G}$

3. 构建 $L$ 个LSH索引表 (离线执行)

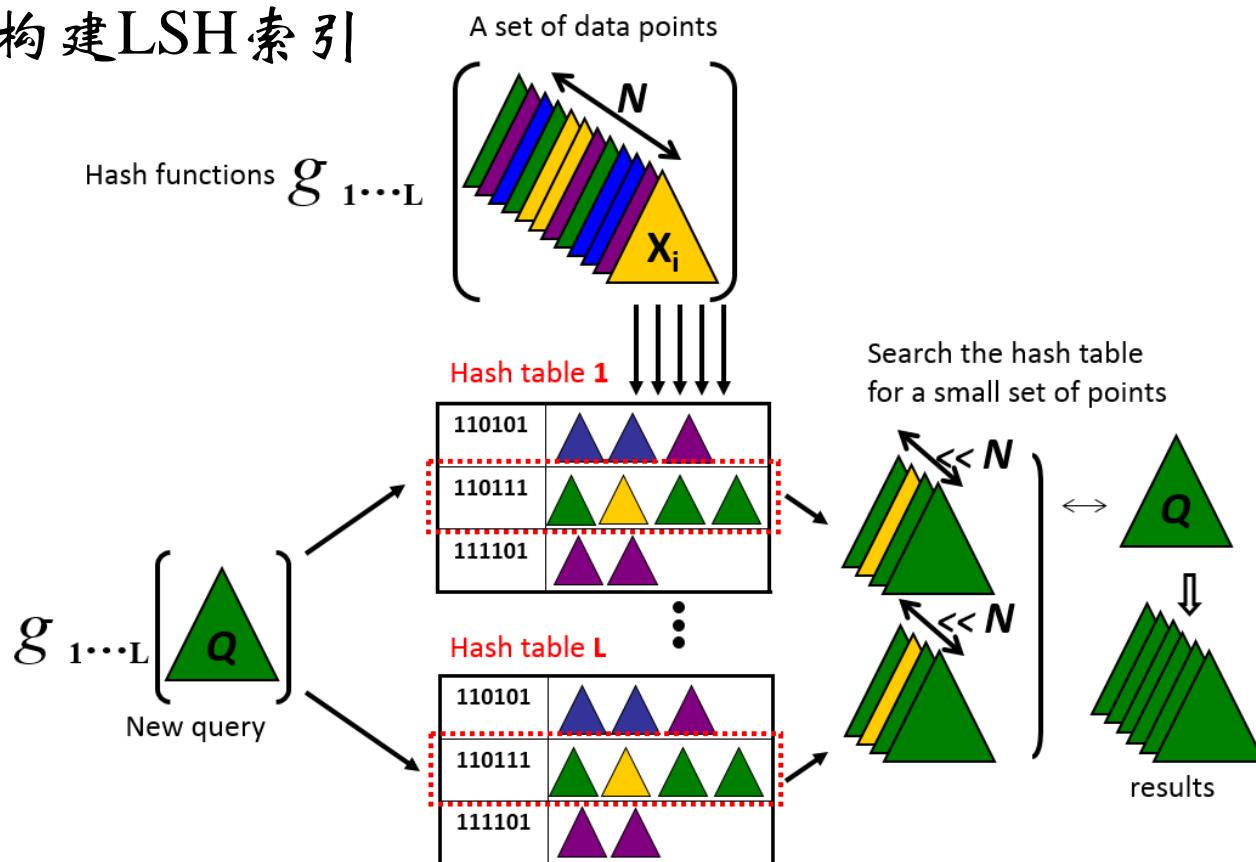
- 为所有数据点计算  $g_j(v)$ , for  $j = 1, \dots, L$ , 并存入桶中

4. 计算查询的 $L$ 个LSH值  $g_1(q), \dots, g_L(q)$ , 并在LSH桶中搜索

# 大数据集的索引加速

## 局部敏感哈希 (Locality Sensitivity Hashing)

- 构建LSH索引



# 大数据集的索引加速

## 局部敏感哈希 (Locality Sensitivity Hashing)

- 哈希函数的作用
  - 映射 (Projection) : 高维数据  $\rightarrow$  1 维
  - 多个映射  $\rightarrow$  多维 (低维)
- 不同距离使用不同的映射
  - Hamming
    - 随机坐标映射/Bit sampling
  - Cosine
    - Random hyper-plane
  - $L_p$  距离:
    - 基于  $p$ -stable 分布的随机向量/直线
    - 曼哈顿距离  $L_1$ : Cauchy (1-stable) 分布
    - 欧氏距离  $L_2$ : Gaussian (2-stable) 分布

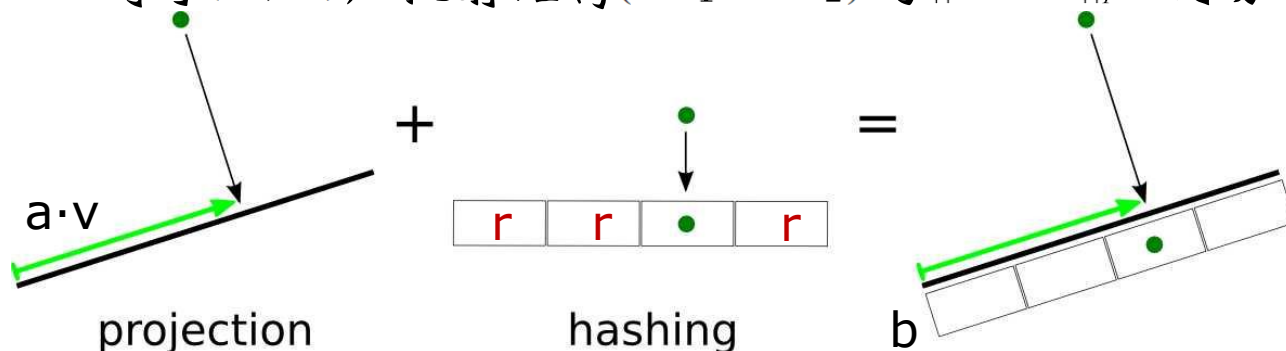
# 大数据集的索引加速

## P-stable LSH

- 原子哈希函数

$$h_{\mathbf{a},b}(\mathbf{v}) = \lfloor \frac{\mathbf{a} \cdot \mathbf{v} + b}{r} \rfloor : \mathcal{R}^d \rightarrow \mathcal{N}$$

- 把d维向量v映射为一条直线上的一个整数值
- 随机投射a: 在p-stable分布上独立、随机选取的d维向量
- 桶宽r: 映射直线上的分段宽度
- 随机偏移b: 在[0, r]上均匀随机选取的偏移值
- 对于 $(v_1, v_2)$ , 投射距离 $(a \cdot v_1 - a \cdot v_2)$ 与 $\|v_1 - v_2\|_p X$ 同分布

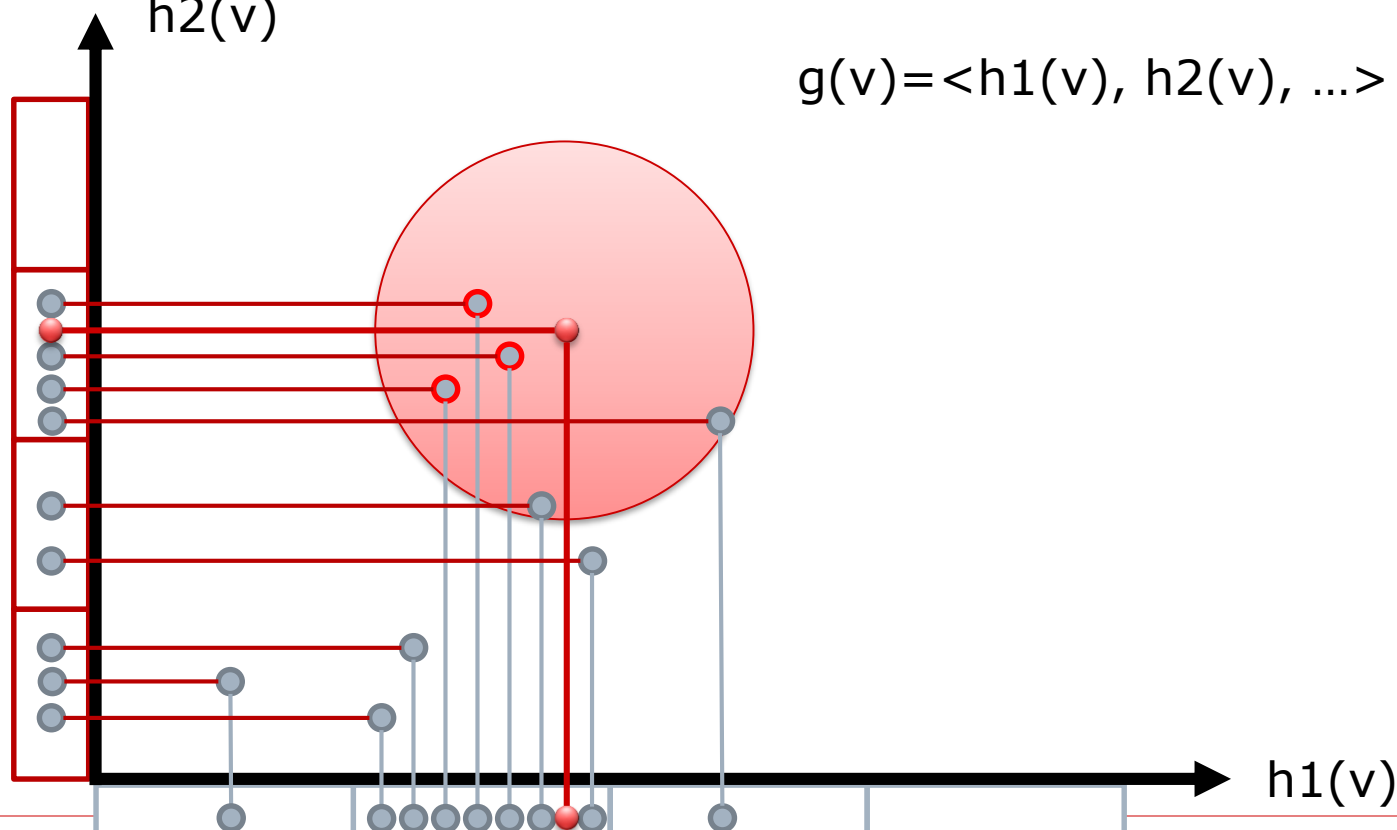


# 大数据集的索引加速

## P-stable LSH

- 增加准确率  
 $h2(v)$

$$g(v) = \langle h1(v), h2(v), \dots \rangle$$

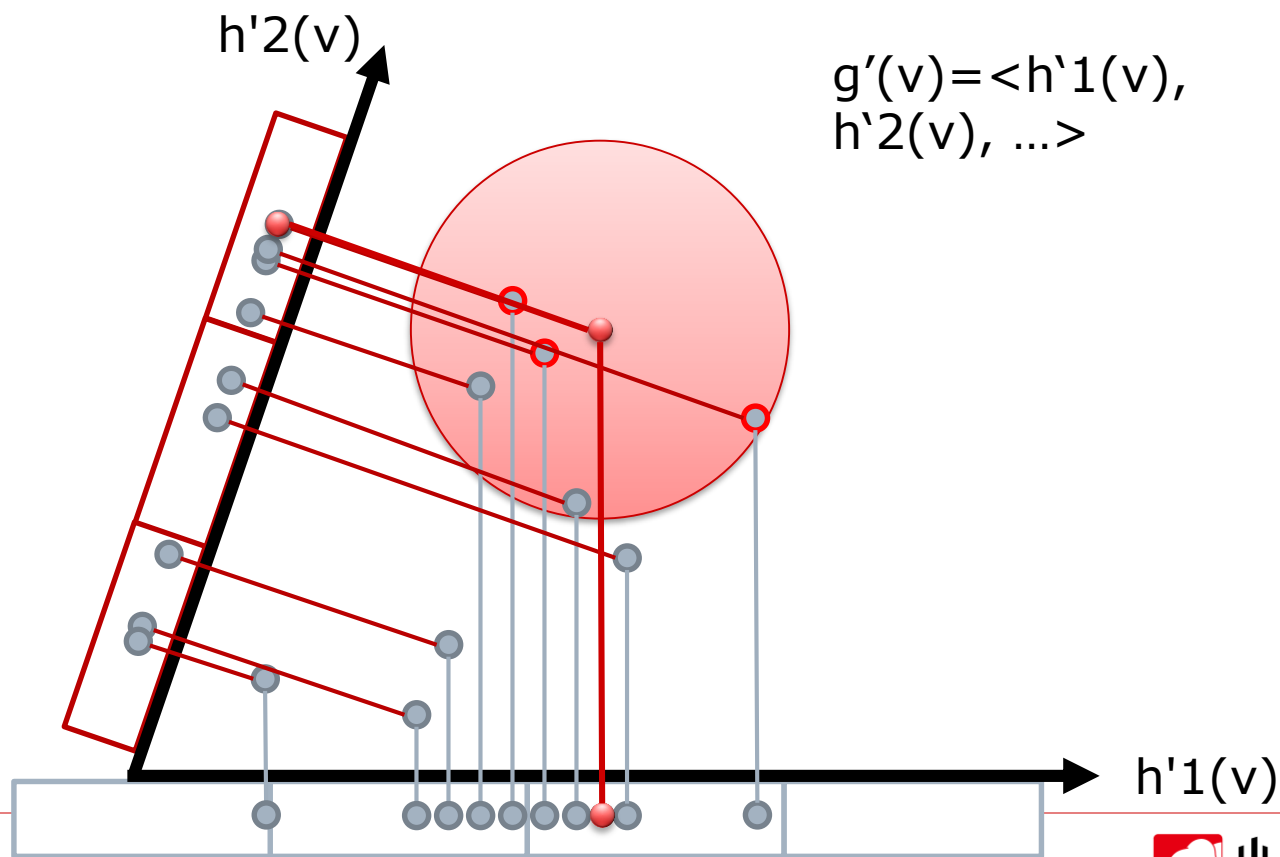




# 大数据集的索引加速

## P-stable LSH

- 增加召回率



# 大数据集的索引加速

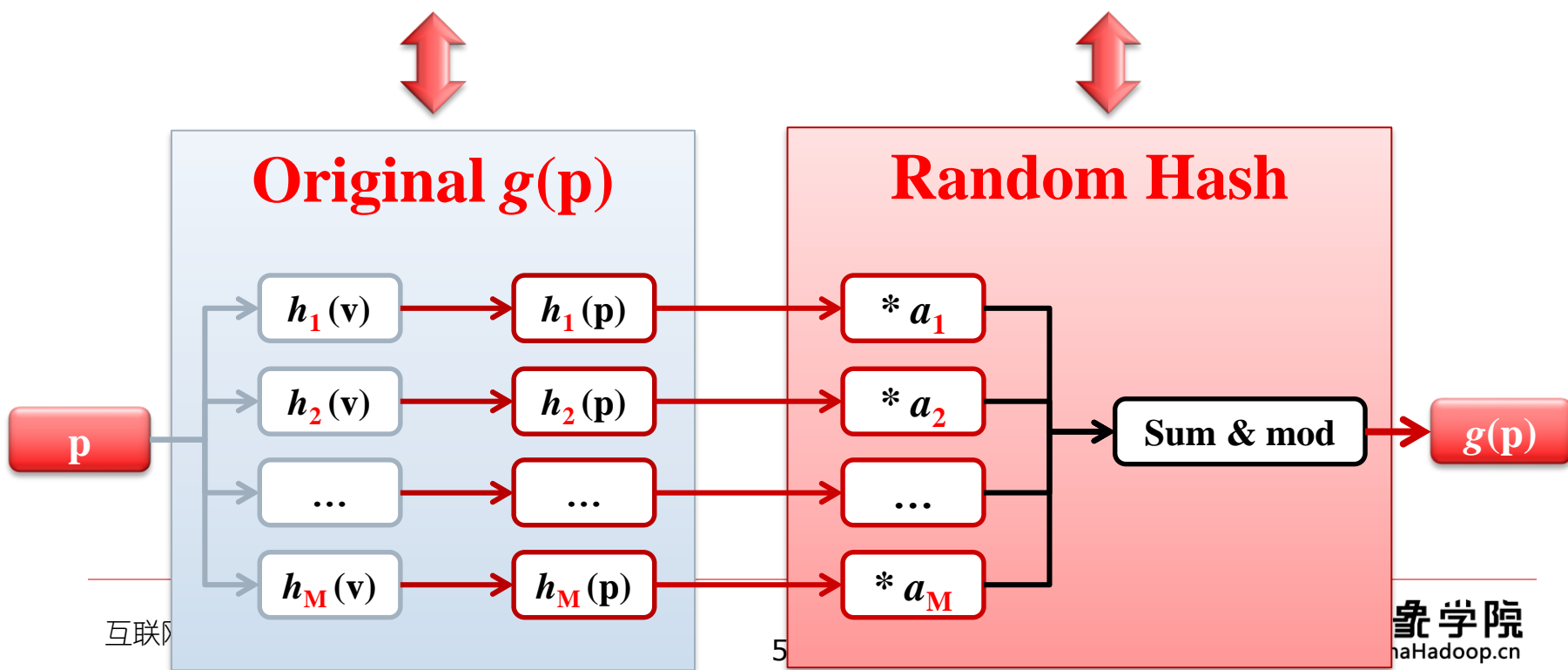
## P-stable LSH

$$g(v) = \langle h_1(v), h_2(v), \dots, h_M(v) \rangle$$

$$h_i(v) = \lfloor \frac{a_i \cdot v + b_i}{W} \rfloor, \quad i = 1, 2, \dots, M$$

计算哈希值  $g(v)$

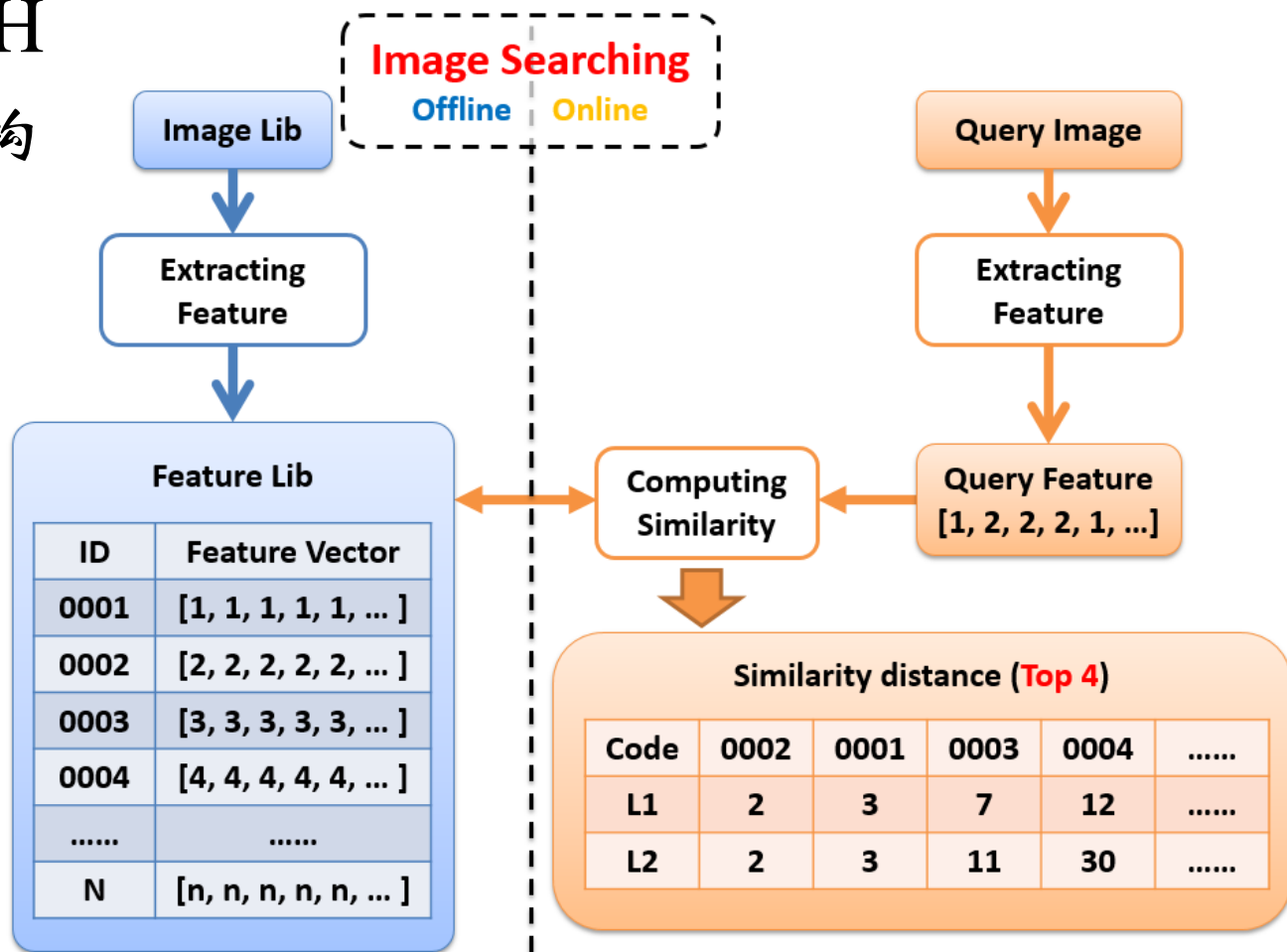
$$((a_1 * h_1 + a_2 * h_2 + \dots + a_k * h_k) \bmod P) \bmod M$$



# 大数据集的索引加速

## 如何集成LSH

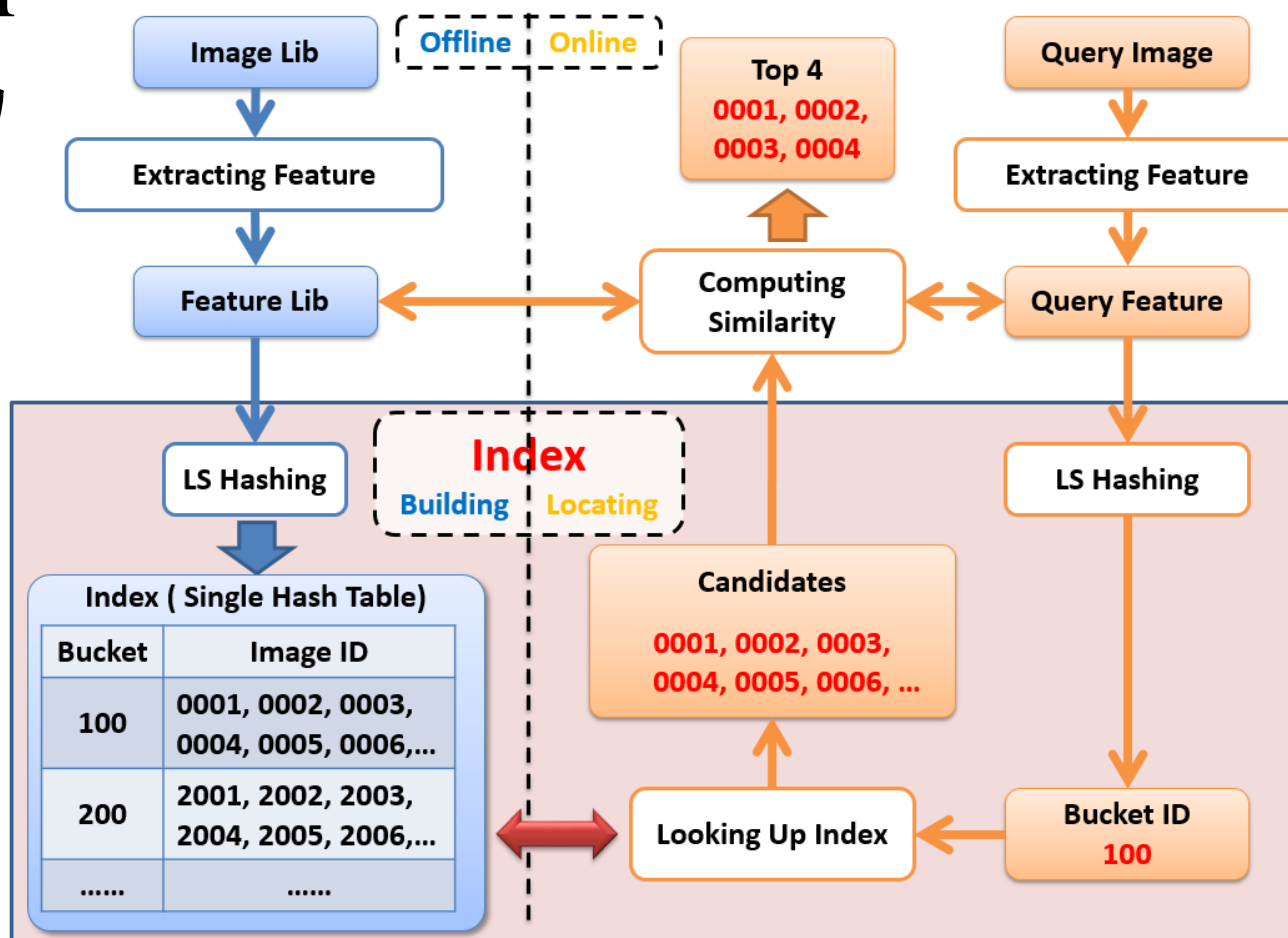
- 原始结构



# 大数据集的索引加速

## 如何集成LSH

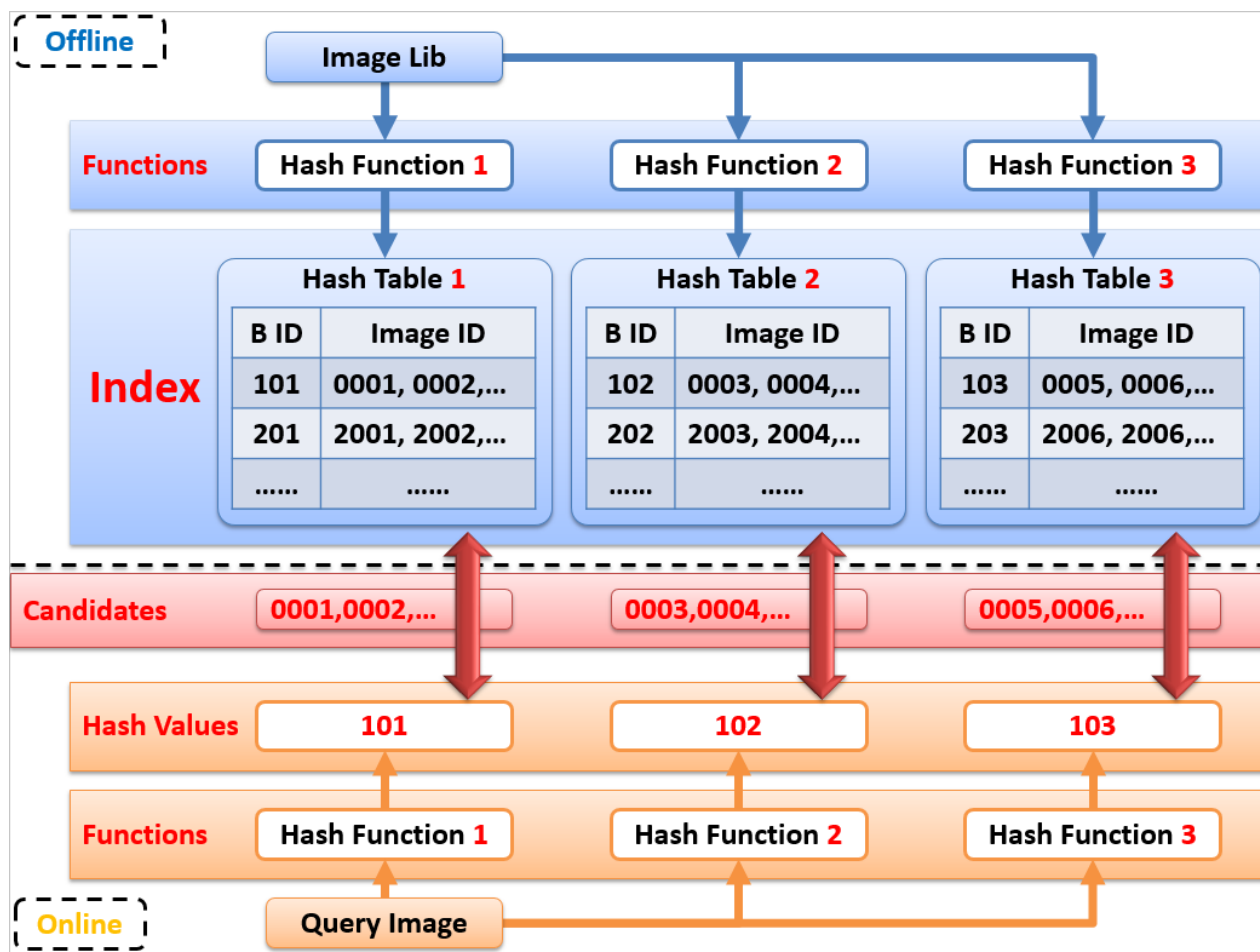
- 单表结构



# 大数据集的索引加速

## 如何集成LSH

- 多表结构



# 疑问

---

□ 问题答疑：<http://www.xxwenda.com/>

■ 可邀请老师或者其他人回复问题

## Q & A

小象账号：349zzjau

课程名：基于深度学习的计算机视觉

课后调查问卷：<http://cn.mikecrm.com/0D9JujS>

# 联系我们

---

## 小象学院：互联网新技术在线教育领航者

- 微信公众号：小象
- 新浪微博：ChinaHadoop

