

内容太多，请大家提前预习

# 第7章第1讲

# 特征提取与特征选择

## Feature Extraction and Feature Selection

向世明

[smxiang@nlpr.ia.ac.cn](mailto:smxiang@nlpr.ia.ac.cn)

<https://peopleucas.ac.cn/~xiangshiming>

时空数据分析与学习课题组 (STDAL)

中科院自动化研究所 模式识别国家重点实验室

助教：张明亮([zhangmingliang2018@ia.ac.cn](mailto:zhangmingliang2018@ia.ac.cn))

程真([chengzhen2019@nlpr.ia.ac.cn](mailto:chengzhen2019@nlpr.ia.ac.cn))

张姣([zhangjiao2019@ia.ac.cn](mailto:zhangjiao2019@ia.ac.cn))

# 第一部分：特征提取

# 7.1 引言

- 模式识别

使机器具有或模拟人的模式识别能力

**模式识别：**“模式是指存在于时间和空间中可观测性、可度量性和可区分性的信息；模式识别是对模式进行分析与处理，进而实现描述、辨识、分类与解译”——谭铁牛院士在 中国科学院学部“科学与技术前沿论坛”上的报告《生物启发的模式识别》，2017年5月16日

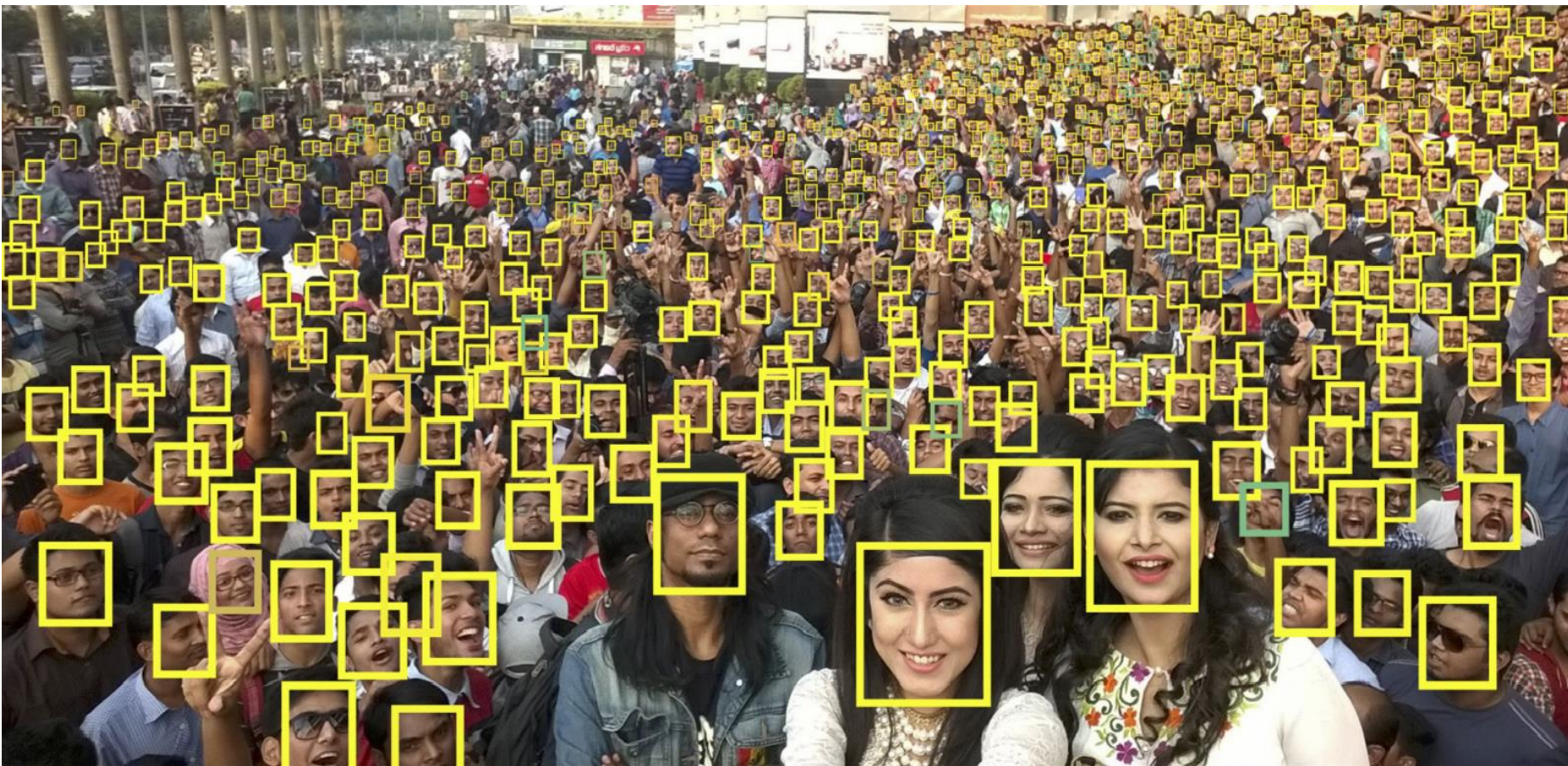
模式的直观特性包括：可观察性，可区分性，相似性。

**模式分类**是模式识别的核心研究内容，相关问题包括**模式描述、特征提取、特征选择、聚类、分类器设计**等。

取决于具体的数据对象，**模式识别的研究内容还包括**信号/图像/视频理解、视觉目标分类、图像/视频检索、文本分类等，以及面向应用的技术研究。

# 7.1 引言

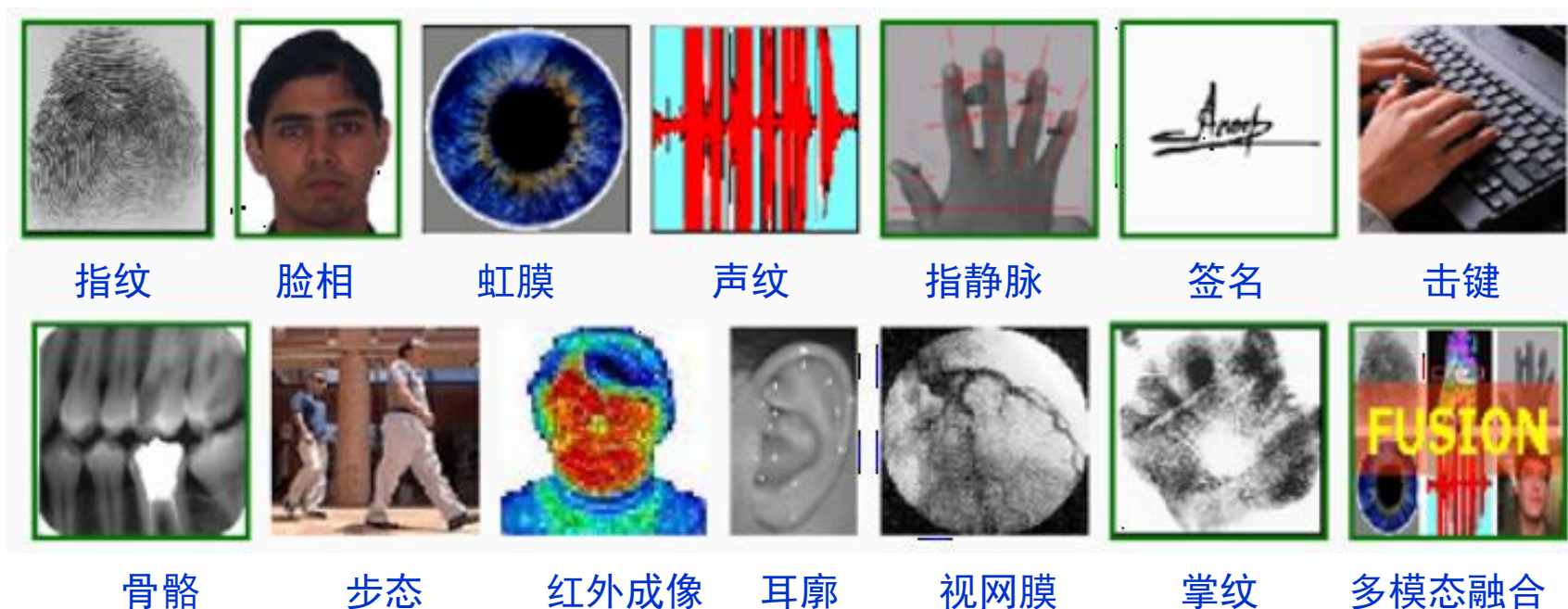
- 特征表示的重要性：以人脸识别为例





# 7.1 引言

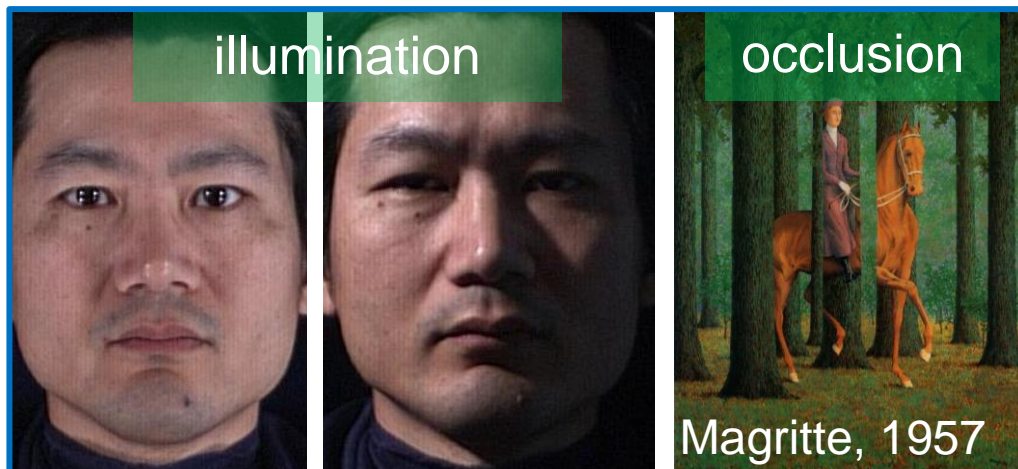
- 特征表示的重要性：以生物特征识别为例



特征应具有模式鉴别能力

# 7.1 引言

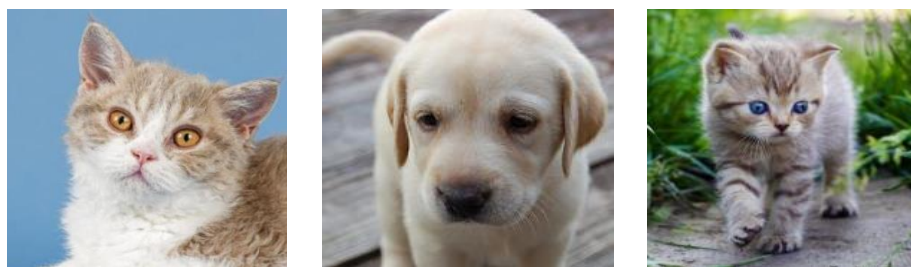
## 特征表示的重要性：自然图像中的挑战



# 7.1 引言

- 特征表示的重要性

If we want to create an algorithm to distinguish dogs from cats:



Raw input vector representation

$$\mathcal{X} = \begin{bmatrix} 23 & 19 & 20 & \dots & 18 \end{bmatrix}$$

$x_1$   $x_2$   $x_3$   $x_n$



$$L_1 \text{ distance: } d_1(I_1 - I_2) = \sum_p |I_1^p - I_2^p|$$

56	32	10	18
90	23	128	133
24	26	178	200
2	0	255	220

-

10	20	24	17
8	10	89	100
12	16	178	170
4	32	233	112

=

46	12	14	1
82	13	39	33
12	10	0	30
2	32	22	108

→ 456
 

Test image                      Train image

- 特征表示的重要性

**End-to-end Deep learning**

- AE, RBM, CNN, RNN, GNN,
- NAS

Raw Data

Feature Space

Classification

**Domain-specific Feature Representation:**

- Preprocessing
- Feature extraction
- Reducing within-class variance
- Enlarging Between-class variance

**Statistical Pattern Recognition**

- Dimension Reduction: PCA, LDA, ICA, Isomap, LLE, ...
- Feature Selection: Wrapper, Filter, Embedded, ...
- Bayesian Decision Theory: Gaussian, Parzen, KNN, Mixture...
- Neural Network: MLP, RBF, CNN, GNN
- Decision Tree: ID3, C4.5, CART, Random forests
- Kernel Method: SVM
- Ensemble Method: Bagging, Boosting
- Clustering: K-means, Hierarchical, Spectral clustering



# 7.1 引言

- **特征提取的目的**

- 减少噪声影响
- 提高稳定性
- 提取观测数据的内在特性

- **特征变换的目的**

- 降低特征空间的维度，便于分析和减少后续步骤的计算量
- 减少特征之间可能存在的相关性
- 有利于分类

# 7.1 引言

- **根据特征提取对象不同**
  - 语音特征提取
  - 文本特征提取
  - 视觉特征提取
- **根据特征提取的方式不同**
  - 局部特征提取方法：SIFT、LBP等
  - 全局特征提取方法：HoG、词袋模型等

# 7.1 引言

- 根据特征变换关系不同
  - 线性特征变换：采用线性映射将原特征变换至一个新的空间（通常维度更低）：
    - PCA、LDA、ICA
  - 非线性特征变换：采用非线性映射将原特征变换至一个新的空间（通常性能更好）：
    - KPCA、KLDA、Isomap、LLE、HLLE、LSTA、...

## 7.2 特征提取

### 7.2.1 语音特征提取

### 7.2.2 文本特征提取

### 7.2.3 视觉特征提取

- 局部二值模式 (LBP)
- Gabor特征提取
- 尺度不变特征变换 (SIFT)
- 视觉词袋 (Bag of Visual Words)
- 哈尔特征
- 梯度方向直方图 (HoG)



## 7.2.1 语音特征提取

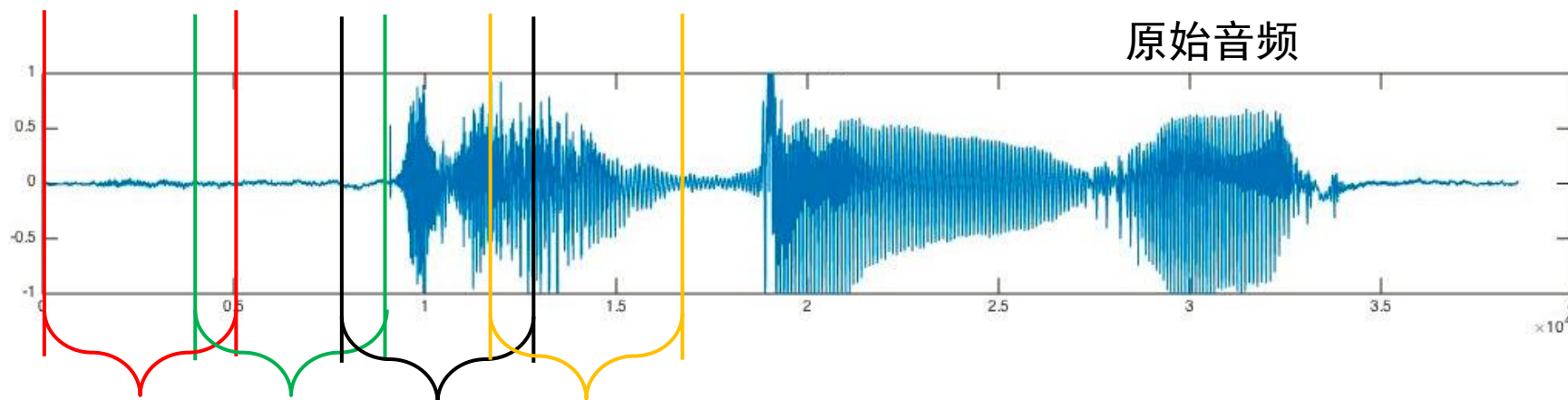
- 语音特征提取技术路线：

1. 对输入的语音信号进行预处理
2. 对语音信号进行分帧、加窗处理
3. 对每一帧的波形信号进行一些特定的数学运算，得到低维向量，作为提取的特征

- MFCCs（Mel Frequency Cepstral Coefficients，梅尔倒谱系数）：

- 一种在自动语音和说话人识别中广泛使用的特征。
- 1980年由Davis和Mermelstein提出，语音识别领域人工特征的佼佼者。
- 符合人的听觉特性。

# MFCCs特征提取



语音信号分帧：将一段语音信号，划分成若干帧

- 帧信号要加窗函数（低通滤波），使得帧两端信号平滑过渡到零
- 帧与帧之间有重叠（帧移），以免帧边缘处信号因加窗弱化而丢失

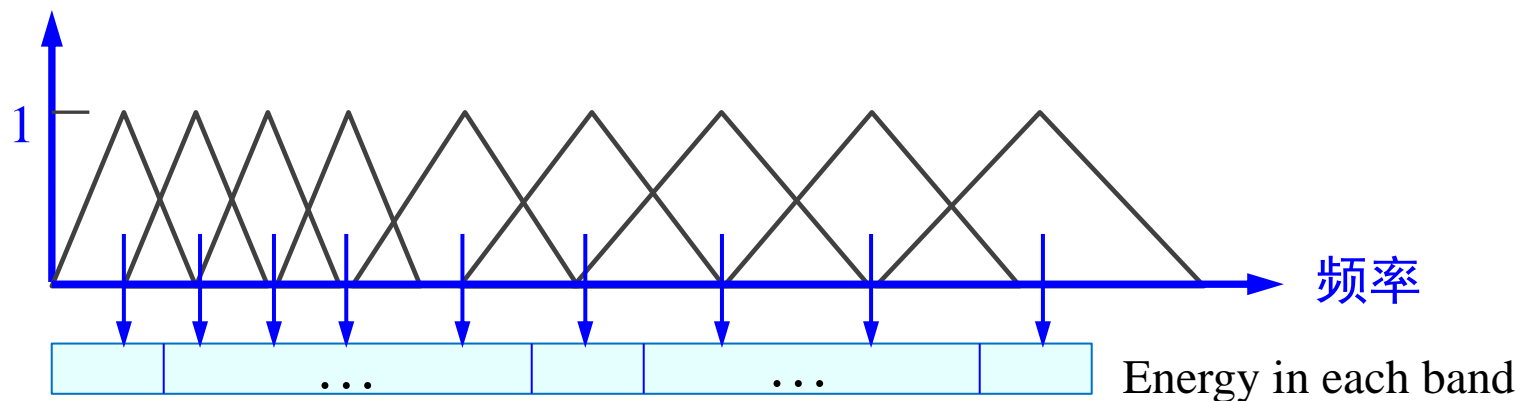
# MFCCs特征提取

## 逐帧计算MFCCs特征：

1. **傅里叶变换**：对分帧后的语音信号进行傅里叶变换，保留幅度谱，丢弃相位谱
2. **根据梅尔刻度**，利用频域三角窗对傅里叶幅度谱进行求和
3. **对求和后的幅度取对数**
4. **离散余弦变换**：对取对数后的幅度信号进行离散余弦变换，得到MFCCs特征。

# MFCCs特征提取

求取频谱在每个三角形区域内的能量总和



- 一个三角形对应一个梅尔频率带
- 低频密、高频疏，**模仿人耳听觉特性**（梅尔频率刻度）
- **减少数据量、提高稳定性**。一般取40个三角形，而傅里叶变换后的频率个数一般几百到上千。



## 7.2.1 文本特征提取

- **文档/文本**：若干词项的有序集合。
- **文本特征提取**：将文本内容转化为向量的过程。将一个文档表示成一个向量，向量的相似性反应文档的相似性。
- **主要方法**：
  - **向量空间模型（Vector Space Model）**：
    - 一个维度对应于一个词项。如果一个词项出现在一篇文档中，它在向量中的值是非零的，否则为零。
  - **TF-IDF**
  - **Word2Vec**

## 7.2.1 文本特征提取

### 词频-逆向文档频率 (TF-IDF)

- 语料库记为  $D$ ，即一个由若干文档组成的集合；文档记为  $d$ ；词语记为  $t$ 。
- 词频  $TF(t, d)$ ：在文档  $d$  中词语  $t$  出现的次数。
- 文档频率  $DF(t, D)$ ：语料库  $D$  中包含词语  $t$  的文档个数。
- 逆向文档频率  $IDF(t, D)$ ：衡量语料库  $D$  中词语  $t$  提供的信息量：

$$IDF(t, D) = \log \frac{|D| + 1}{DF(t, D) + 1}$$


- 词频-逆向文档频率：

$$TFIDF(t, d, D) = TF(t, D) \times IDF(t, D)$$

# 7.2.1 文本特征提取

## 词频-逆向文档频率 (TF-IDF)



文档特征向量：(  )<sup>T</sup>

↑                      ↑

japanese                      war

# Word2Vec

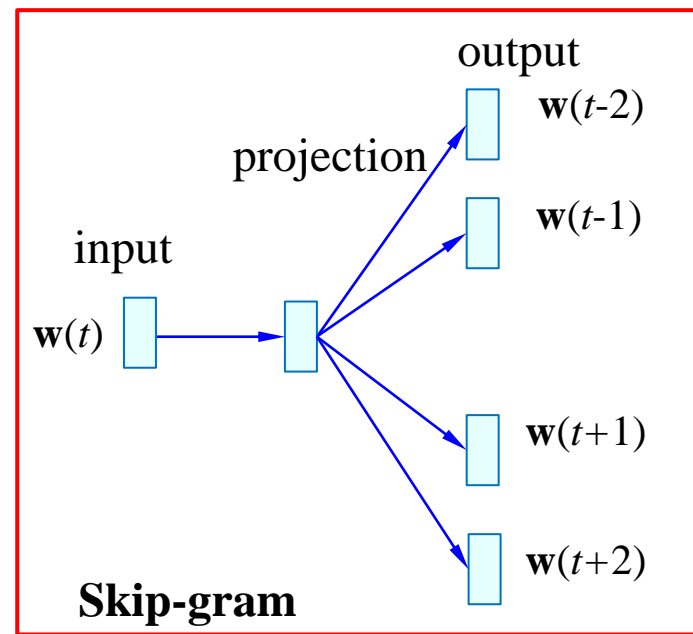
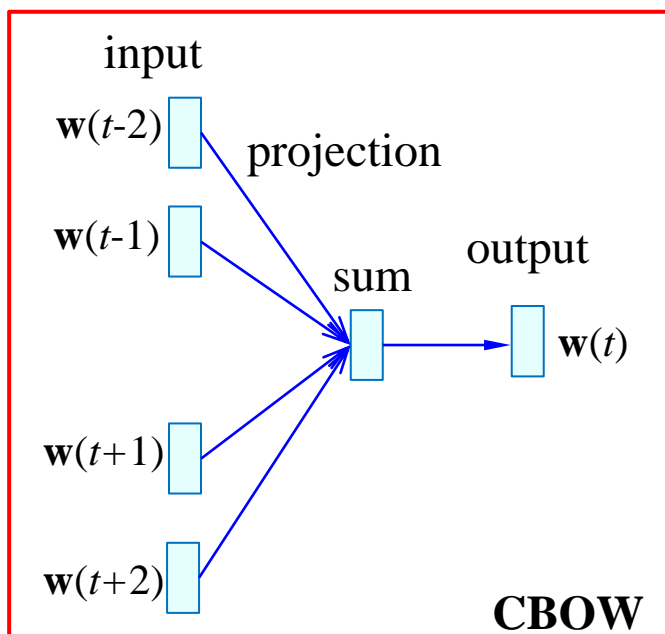
基本任务：利用一个连续向量来表示一个词项。

主要目的：相似单词具有相似的向量表示。

核心思想：利用单词预测上下文单词。

技术路线：浅层神经网络。

主要模型：[连续词袋模型](#)(Continuous Bag of Words, CBOW), [跳字模型](#)(Skip Gram)

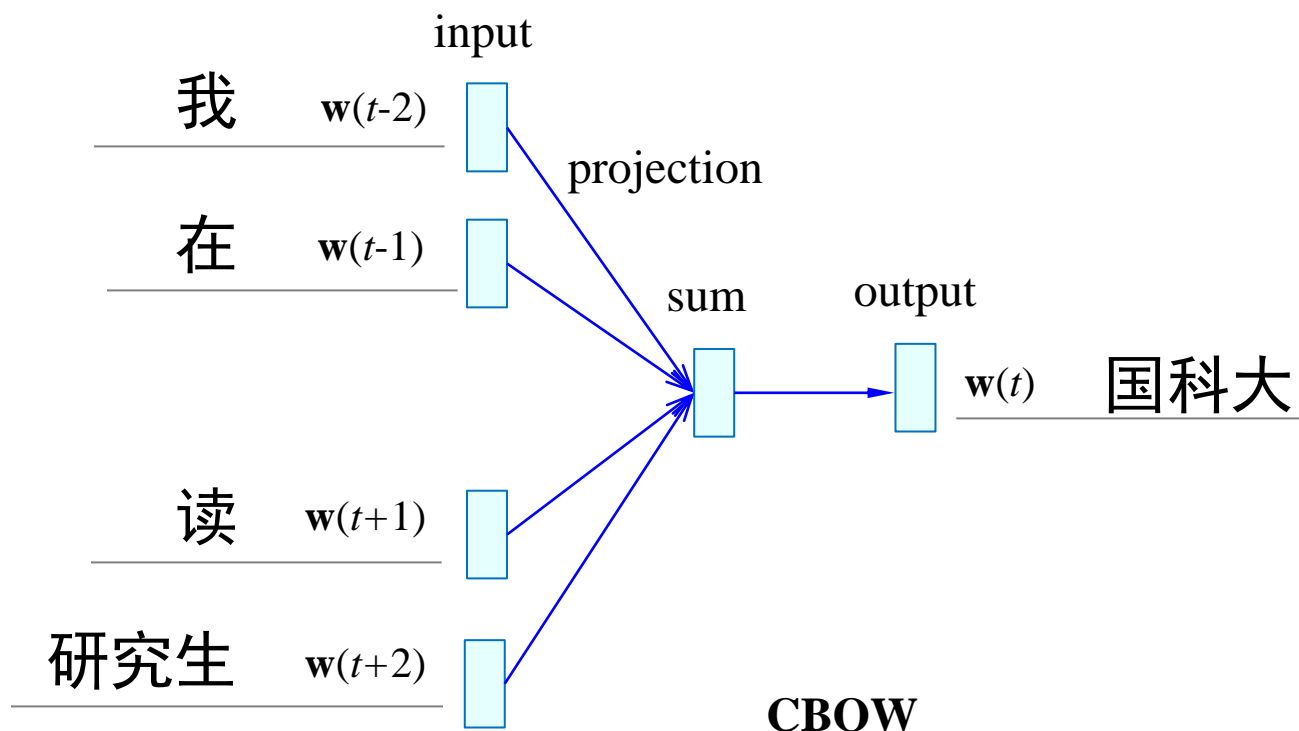


skip-gram逆转CBOW的因果关系，即已知当前词语，预测上下文



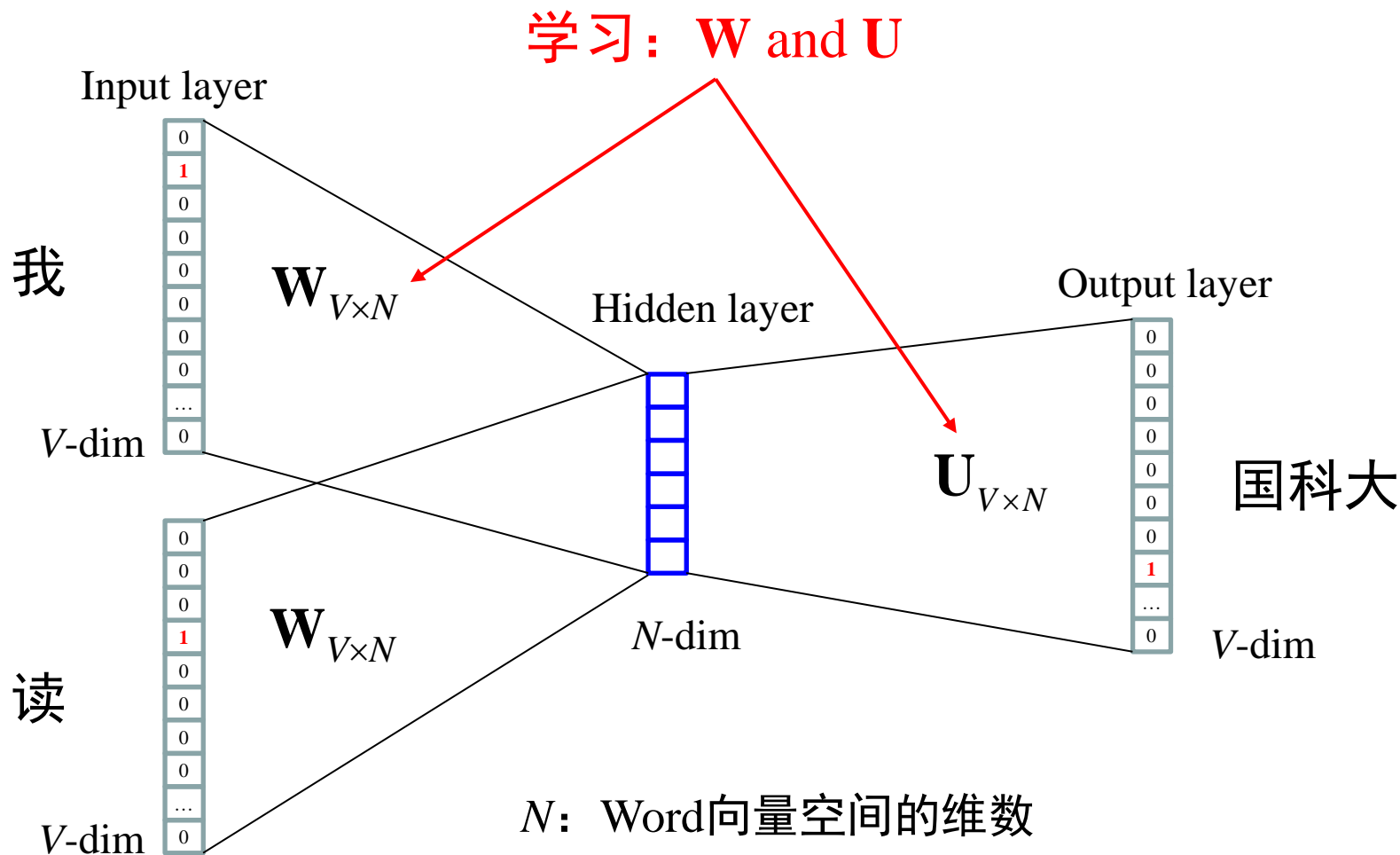
# Word2Vec

例子：我在国科大读研究生。

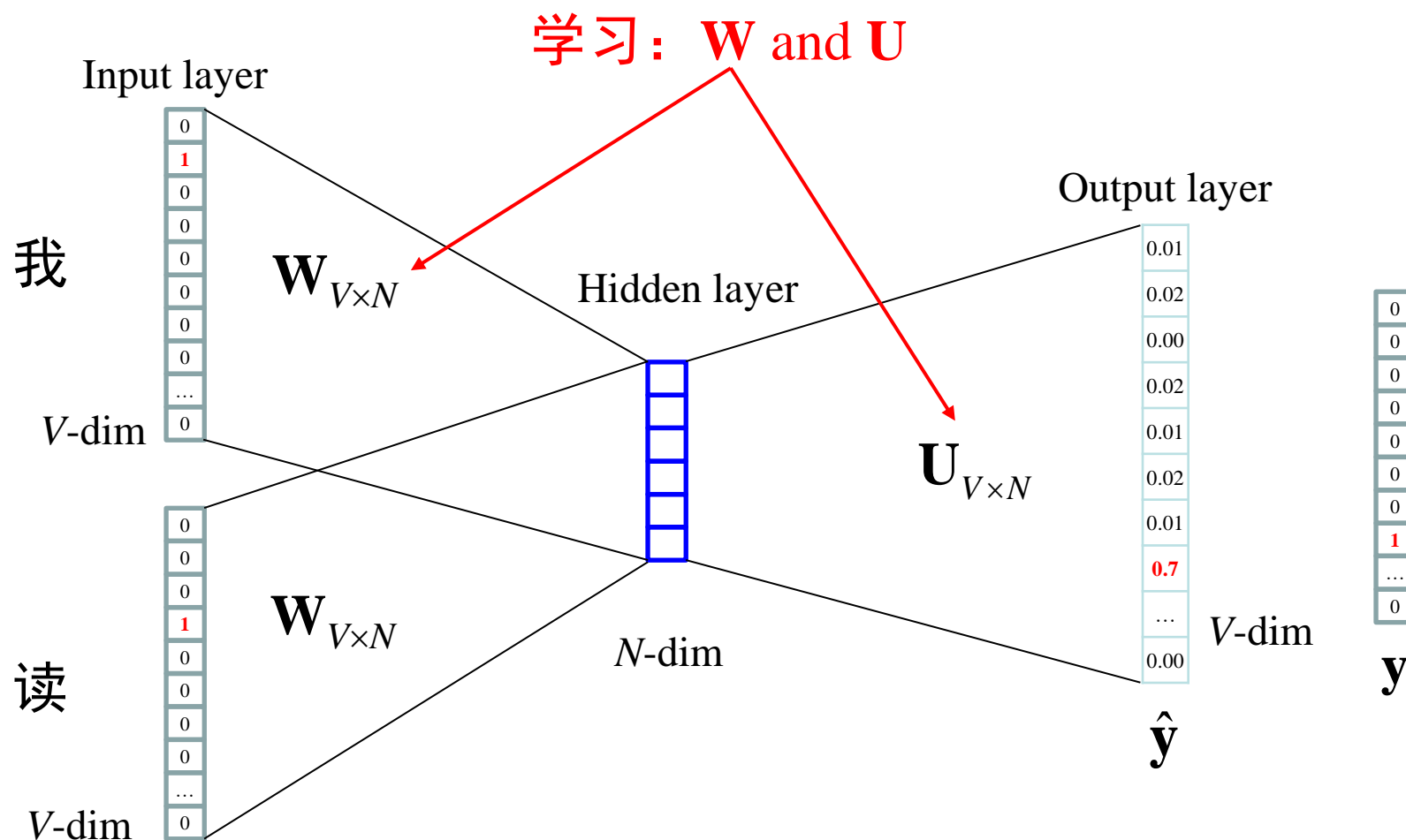


取上下文窗口为2

假定词库的大小为 $V$ ，每一个词用一个 $V$ 维one-hot向量表示，即对应的维度元素为1，其余为0。

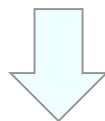


学习目标：希望预测的  $\hat{y}$  与真实的  $y$  接近 (softmax)



# Word2Vec

- ✓ **W** 对应了输入词汇表的word2vec矩阵，每一行对应一个词。
- ✓ **U** 对应了输出词汇表的word2vec矩阵，每一列对应一个词。



任取其一，或者取平均作为词汇的特征提取结果。

## 7.2.3 视觉特征提取

- 局部二值模式 (LBP)
- Gabor特征提取
- 尺度不变特征变换 (SIFT)
- 视觉词袋 (Bag of Visual Words)
- 哈尔特征
- 梯度方向直方图 (HoG)

# 7.2.3.1 视觉特征提取--LBP

## Local Binary Pattern



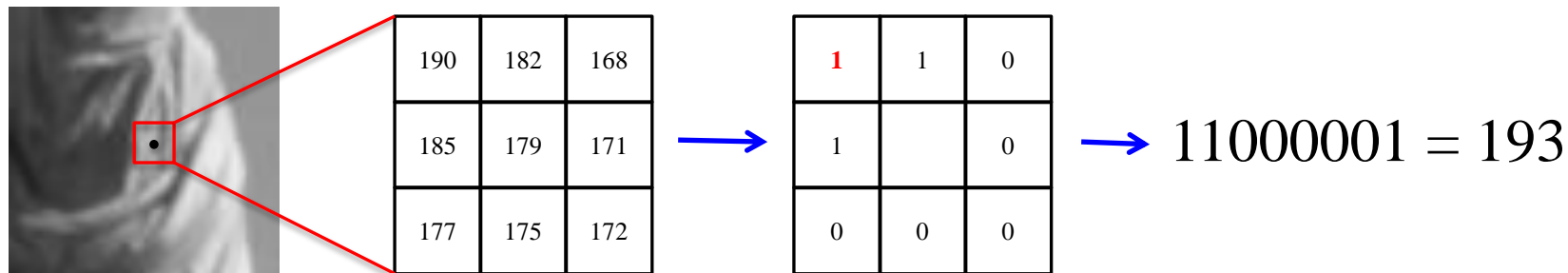
Matti Pietikäinen (芬兰奥卢大学)

T. Ojala, M. Pietikainen, T. Maenpaa. Multiresolution grayscale and rotation invariance texture classification with local binary patterns, IEEE TPAMI, 24(7), 971-987, 2002.

- 局部特征提取方法，针对每个像素点计算
- 计算简单、对于光照变化较稳定
- 广泛用于纹理分析、人脸检测、识别



# LBP特征计算过程



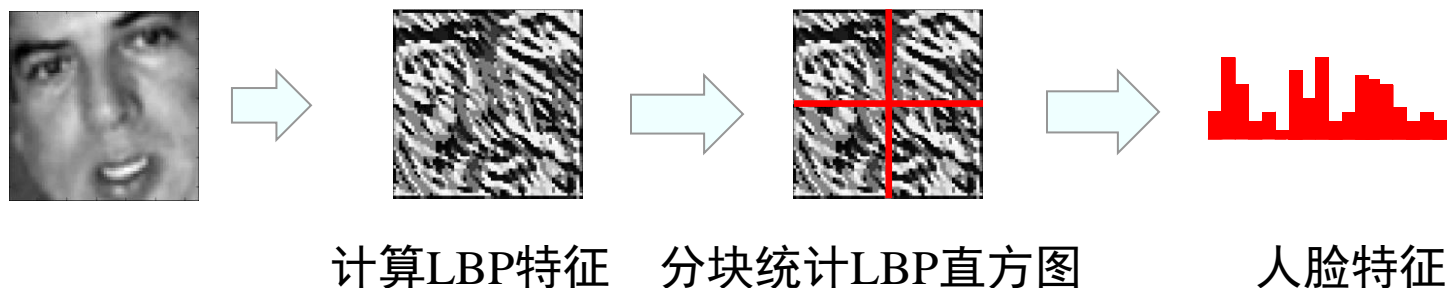
LBP特征的一般性定义：

$$LBP_{R,N}(x) = \sum_{i=0}^{N-1} \text{sign}(I(x_i) - I(x)) 2^i$$

$R$ ：像素周围邻域半径； $N$ ：像素周围区域采样点个数

若采样点 $x_i$ 不是整数（不在像素格子上）？**双线性插值**

# LBP特征应用：人脸识别



- 对人脸图像提取特征向量之后，送入人脸分类模型进行人脸识别。
- 人脸分类模型：
  - 特征变换 + k-NN分类；
  - 特征变换 + 多类SVM；
  - 特征变换 + 神经网络

## 7.2.3.2 视觉特征提取--Gabor

- **Dennis Gabor**
  - Hungarian-British , 电子工程师, 物理学家
  - 因发明了全息摄影术 (Holography) 获得1971年诺贝尔物理奖
  - Gabor变换的提出者
  - 小波变换的创始人之一



# Gabor 滤波器定义

完整复数表达:

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

实数部分:

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

虚数部分:

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

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

高斯因子的  
标准差

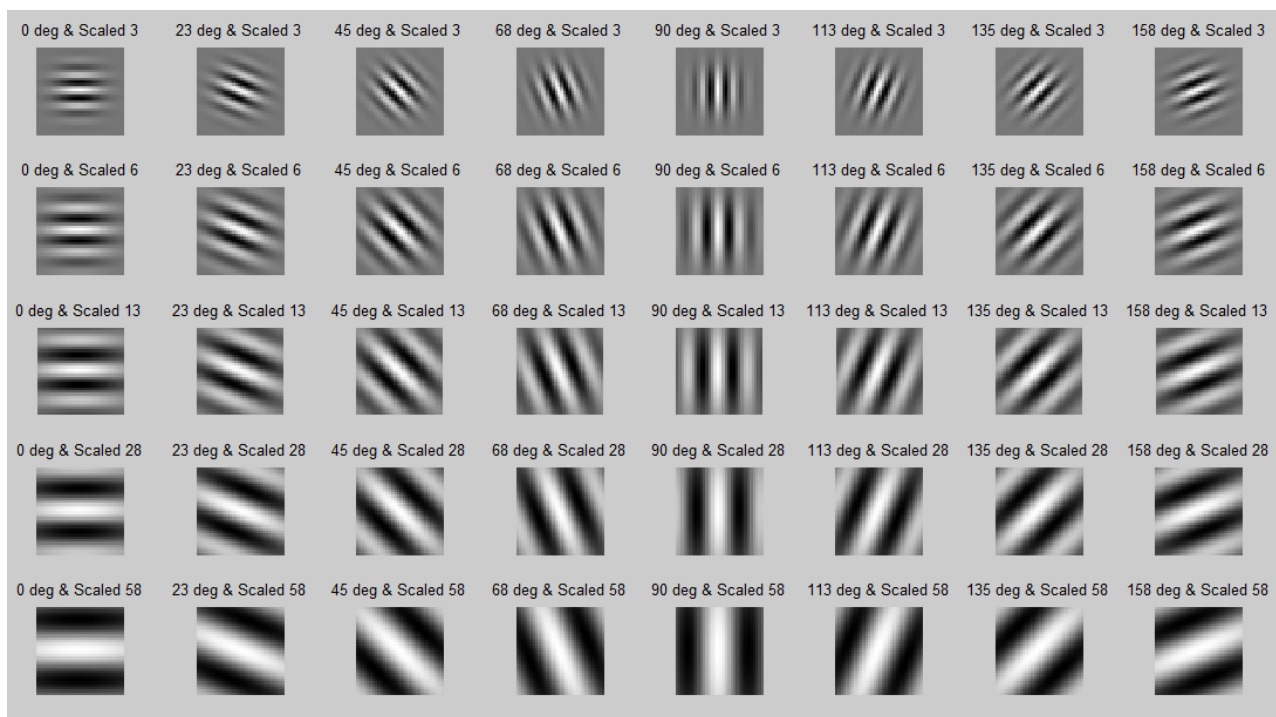
空间纵横比, 决定了Gabor函数形状的椭圆率

表示Gabor核函数中余弦函数的相位参数。

表示Gabor核函数中余弦函数的波长参数, 像素单位

表示Gabor滤波核中平行条带的方向。

## 7.2.3.2 视觉特征提取--Gabor



不同尺度和方向的Gabor滤波器

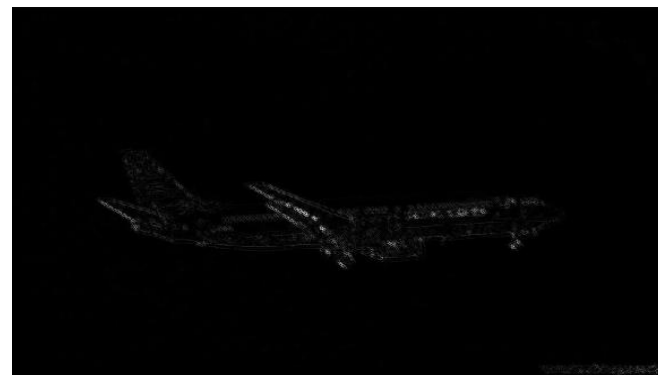
## 7.2.3.2 视觉特征提取--Gabor



0°



90



13

# 7.2.3.3 视觉特征提取--SIFT

## Scale Invariant Feature Transform

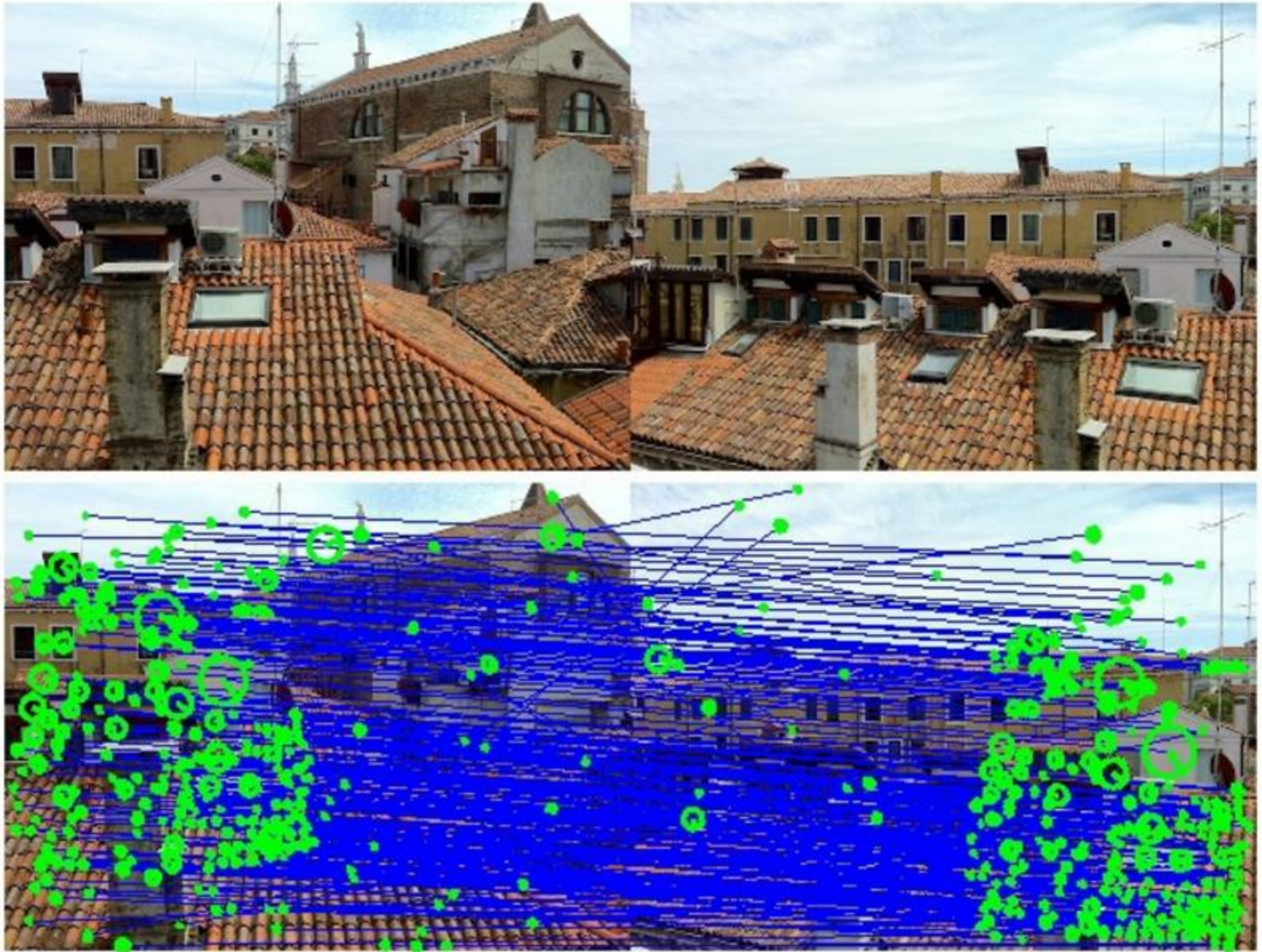


David G. Lowe 加拿大英属哥伦比亚大学  
计算机科学系教授

- D. G. Lowe. Distinctive image features from scale-invariant key-points, IJCV, 60(20, 91-110, 2004
- D. G. Lowe. Object recognition from local-invariant features, ICCV, pp.1150-1157, 1999

- 1999年ICCV上首次提出，并应用于物体识别
- 2004年IJCV上长文发表
- 视觉相关的特征中**影响力最大**者之一
- 局部方法：对图像中的局部区域进行分析
- 特征点检测+特征点描述





## 7.2.3.3 视觉特征提取--SIFT

- **包含两步：**特征点检测 + 特征点描述
- **特征点：**图像中可辨识度高的点，容易在同一物体的不同图像中重复出现
- **特征点检测要求：**尺度不变、旋转不变、对视角变化、光照变化鲁棒
- **特征描述子：**为特征点计算“个性签名”，区分是否属于同一个物理点

# 7.2.3.3 视觉特征提取--SIFT

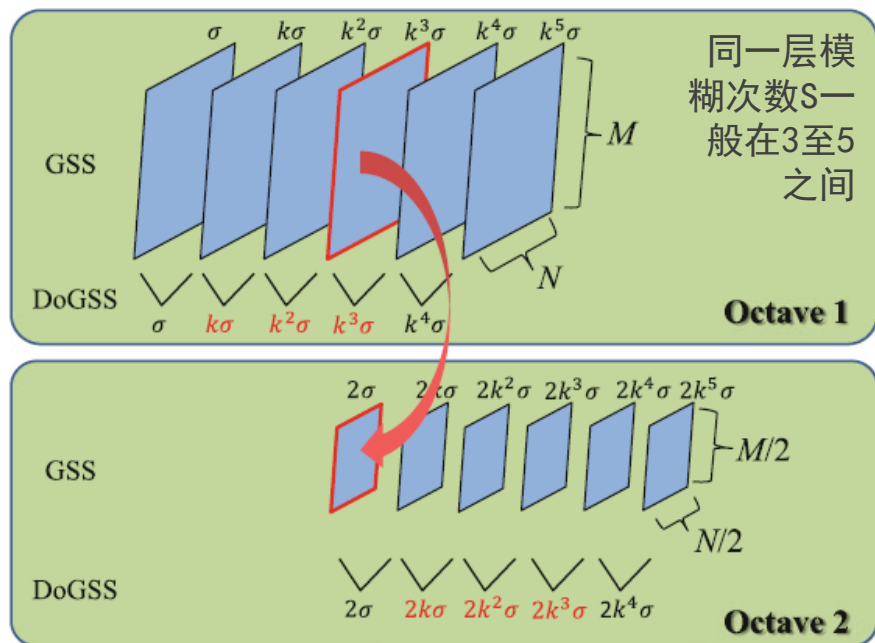
## SIFT特征提取技术路线

1. 高斯尺度空间构建 (变化尺度的高斯函数与原图像的卷积)
2. 高斯差分尺度空间构建
3. 极值点检测
4. 特征点精细定位
5. 特征点主方向计算
6. 特征描述子生成

参考：<https://blog.csdn.net/u010440456/article/details/81483145>

# 尺度空间构建与特征点检测

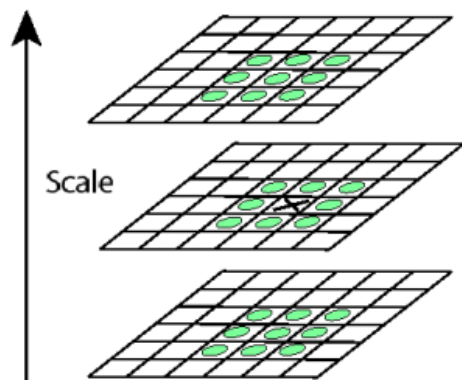
构建图像金字塔(GSS), 对同一级图像获得多尺度高斯模糊图像, 然后高斯差 (DoGSS)



**GSS:** 通过逐次与上一层图像进行高斯卷积得到图像的高斯尺度空间(GSS).

**DoGSS:** 通过GSS相邻层相减得到高斯差分尺度空间(DoGSS).

**Octave:** 采用组的概念对尺度空间进行划分, 加快计算速度



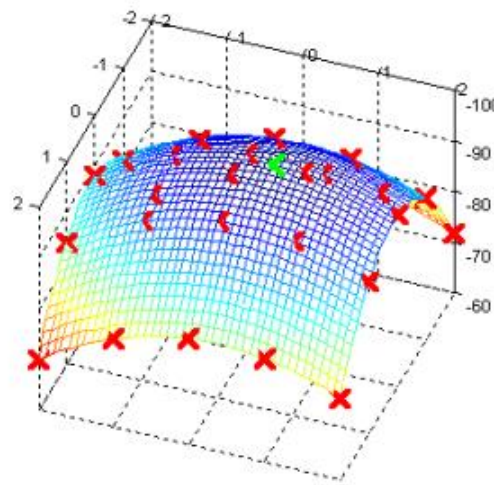
通过在DoGSS中进行极值点检测进行初始特征点定位, 包括位置(x, y)和尺度(s)



# 特征点精细定位：

## 不稳定点去除(关键点筛选)：

1. 去除在拟合后峰值点响应值较低的点(可采用拟合)
2. 去除边缘响应值较大的点 (位于边界的点)
3. 去除低对比度关键点



**精细定位：**在初始特征点周围拟合一个三维 $(x, y, s)$ 二次曲线，求得二次曲线的峰值点作为特征点精细定位结果。

**边缘点：**在边缘梯度的方向上主曲率值比较大，而沿着边缘方向则主曲率值较小；候选特征点的DoG函数的主曲率与Hessian矩阵的特征值成正比。

# Key-point localization with orientation

233x189



832

initial keypoints

729

keypoints after  
gradient threshold



536

keypoints after  
ratio threshold



低对比度的特征点

不稳定的边缘响应点

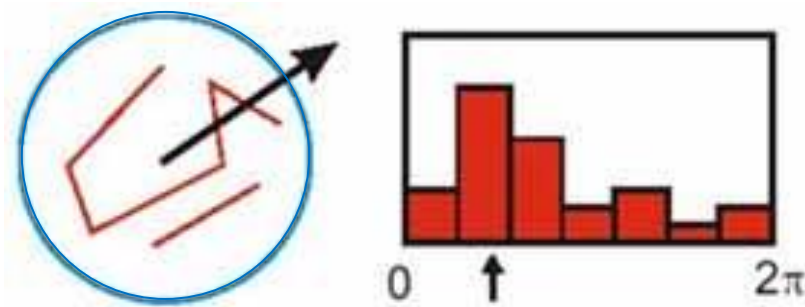
# 特征点主方向计算：

梯度幅值和幅角：

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$

$$\theta(x, y) = \alpha \tan 2((L(x+1, y) - L(x-1, y)) / (L(x, y+1) - L(x, y-1)))$$

根据检测到的极值点所在的尺度，在特征点周围区域内统计梯度方向直方图，取**直方图响应最大的方向作为主方向**

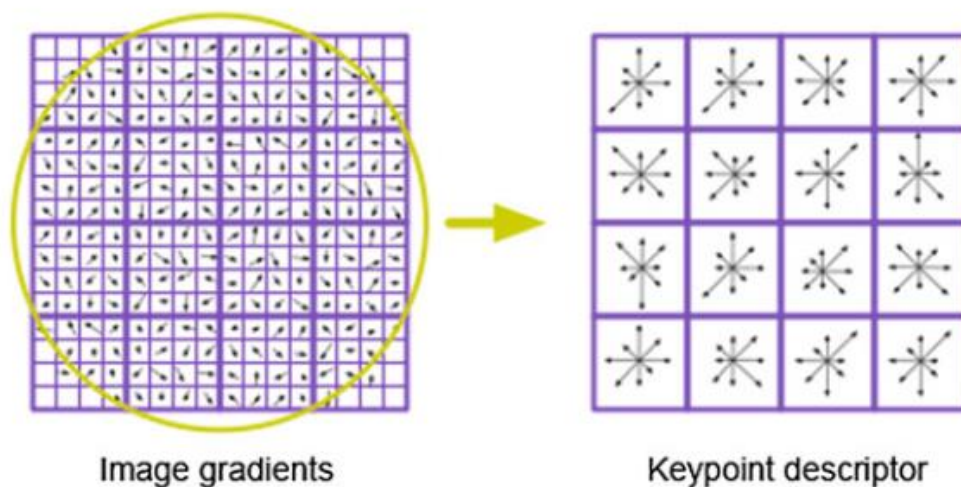


**提取多个主方向，提高匹配的稳定性：**响应值在最大值80%范围内的方向亦可作为主方向



# 特征描述子构造

1. 根据特征点的尺度，在GSS中找到尺度最接近的图像层
2. 在特征点周围进行采样，得到 $16 \times 16$ 的网格
3. 根据特征点的主方向，对网格内采样点的梯度和位置进行旋转
4. 将 $16 \times 16$ 的网格划分成 $4 \times 4$ 个子区域
5. 统计每个子区域的梯度方向直方图
6. 将子区域的梯度方向直方图串在一起，并做归一化



---

## 基于SIFT的图像匹配算法

---

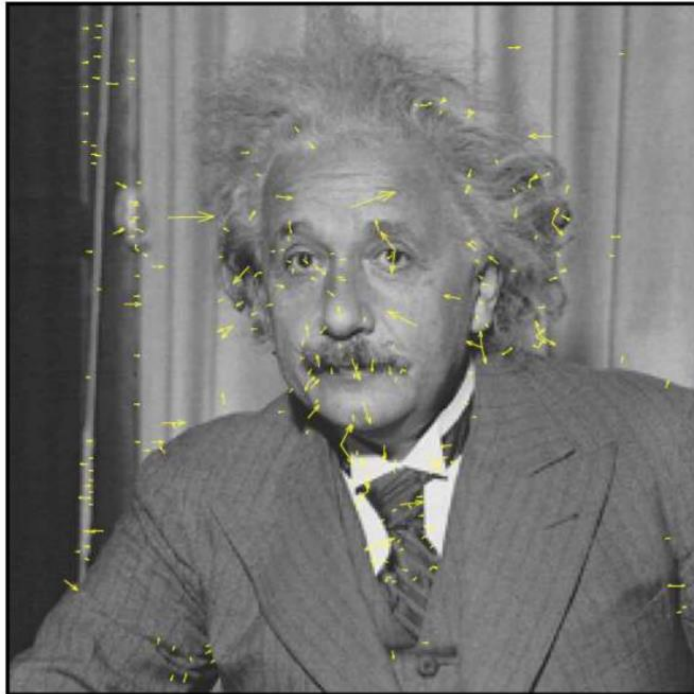
输入：待匹配图像1和图像2，特征点匹配阈值 $t$

输出：图像1和图像2之间对应点集合

---

- 1 利用SIFT算法对图像1提取特征点集合；
  - 2 利用SIFT算法对图像2提取特征点集合；
  - 3 根据特征点集合1和特征点集合2中的特征点描述子，计算它们之间的欧式距离，根据最近邻匹配原则，对特征点集合1中的每一个特征点找到集合2中对应的特征点；
  - 4 根据对应特征点的欧式距离，去除大于阈值 $t$ 的点；
  - 5 输出剩余的特征点对应集合。
-

# SIFT: Scale Invariant Feature Transform

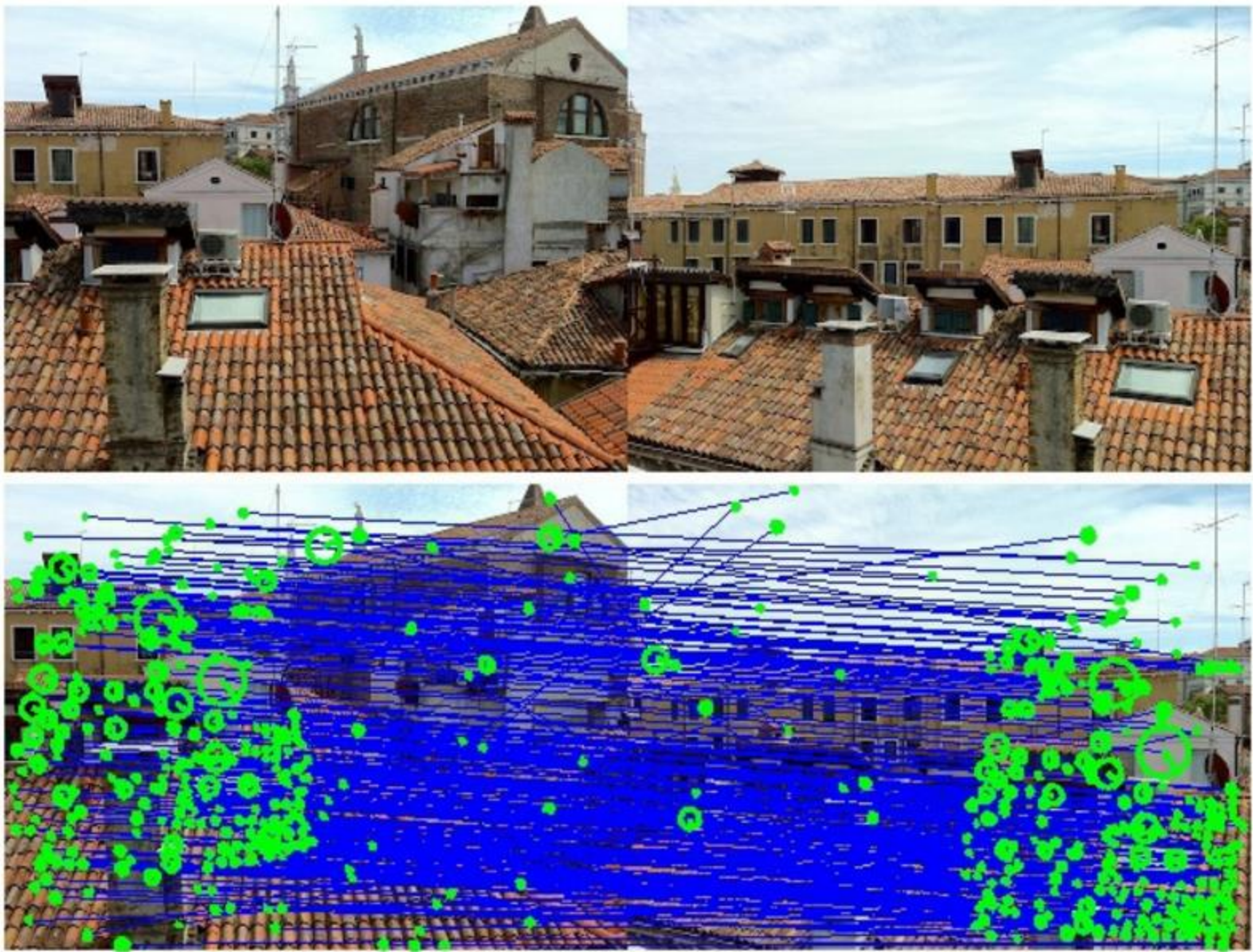


For better image matching, Lowe's goal was to develop an interest operator that is invariant to scale and rotation.



Also, Lowe aimed to create a **descriptor** that was robust to the variations corresponding to typical viewing conditions. **The descriptor is the most-used part of SIFT.**



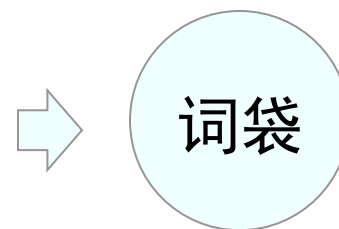


SITF & object Recognition, David Lowe, 1999

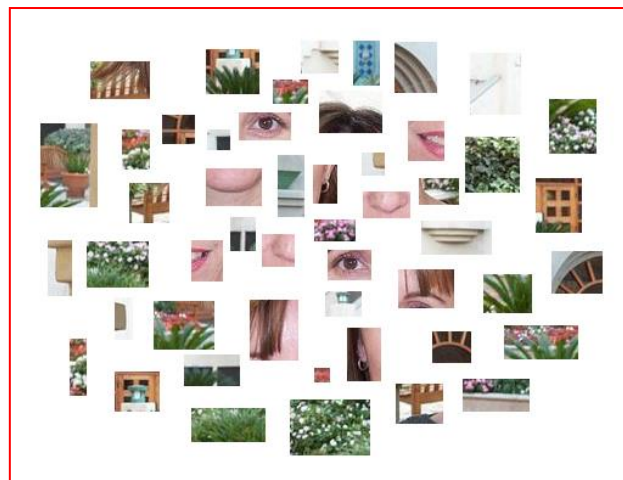
## 7.2.3.4 视觉特征提取--视觉词袋

**问题：** SIFT等局部特征提取方法，可以得到一个特征点及其描述向量的集合。**如何根据这个集合对整幅图像进行描述？**

**文档向量空间模型（Vector Space Model）：** 如果一个词项出现在一篇文档中，它在向量中的值是非零的，否则为零。



## 7.2.3.4 视觉特征提取--视觉词袋

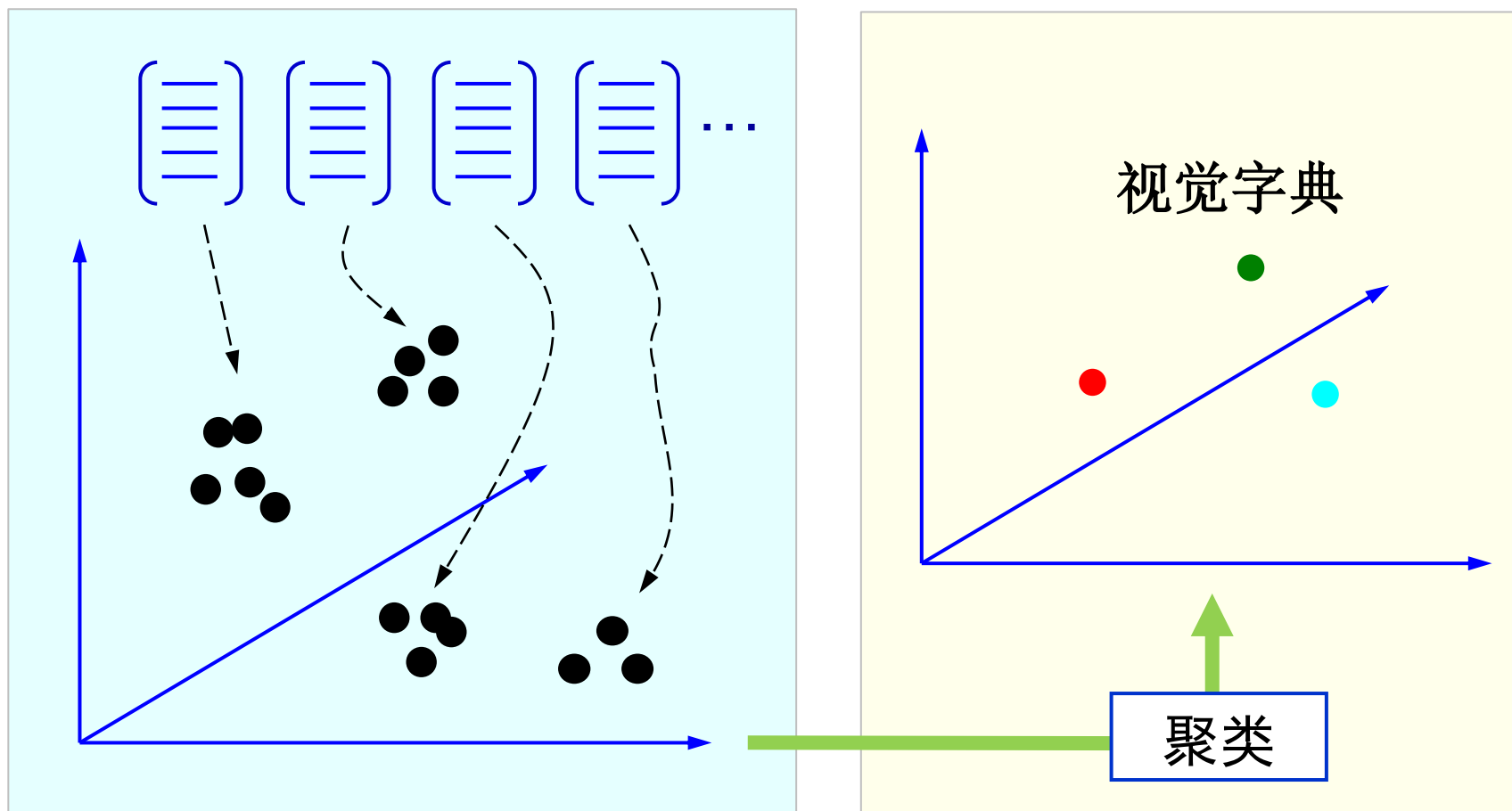


**视觉词袋模型 (bag of visual words) :**

1. 如何确定视觉字典？
2. 给定一个视觉字典，如何将一个局部视觉特征，用视觉字典进行表示？



**确定视觉字典：**收集若干图像的局部图像特征描述子，对其进行聚类，**每个聚类表示一个视觉单词**（visual word）





## 7.2.3.4 视觉特征提取--视觉词袋

- **特征编码**：给定一个视觉字典，将一个特征描述子转化成向量表示的过程。通常，表示向量的维度等于视觉字典的大小。
- **硬编码（hard assignment）**：只对向量某一维赋值。根据最近邻准则，将距离特征描述子最近的视觉单词对应的位置置为1，其余位置为0。
- **软编码（soft assignment）**：对向量所有维度赋值。根据特征描述子到各视觉单词的距离，分配相应的编码值。

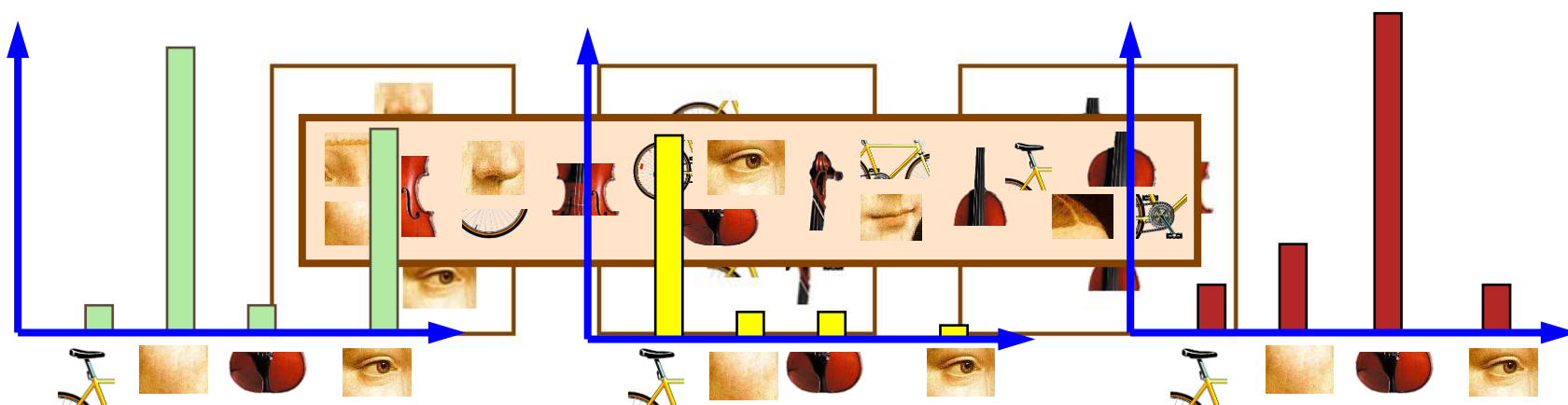
$$h_i(x) = \exp(-\|x - v_i\|^2 / \delta^2) / \sum_{i=1}^N \exp(-\|x - v_i\|^2 / \delta^2)$$

- 稀疏编码、局部线性嵌入编码、局部软编码等.....

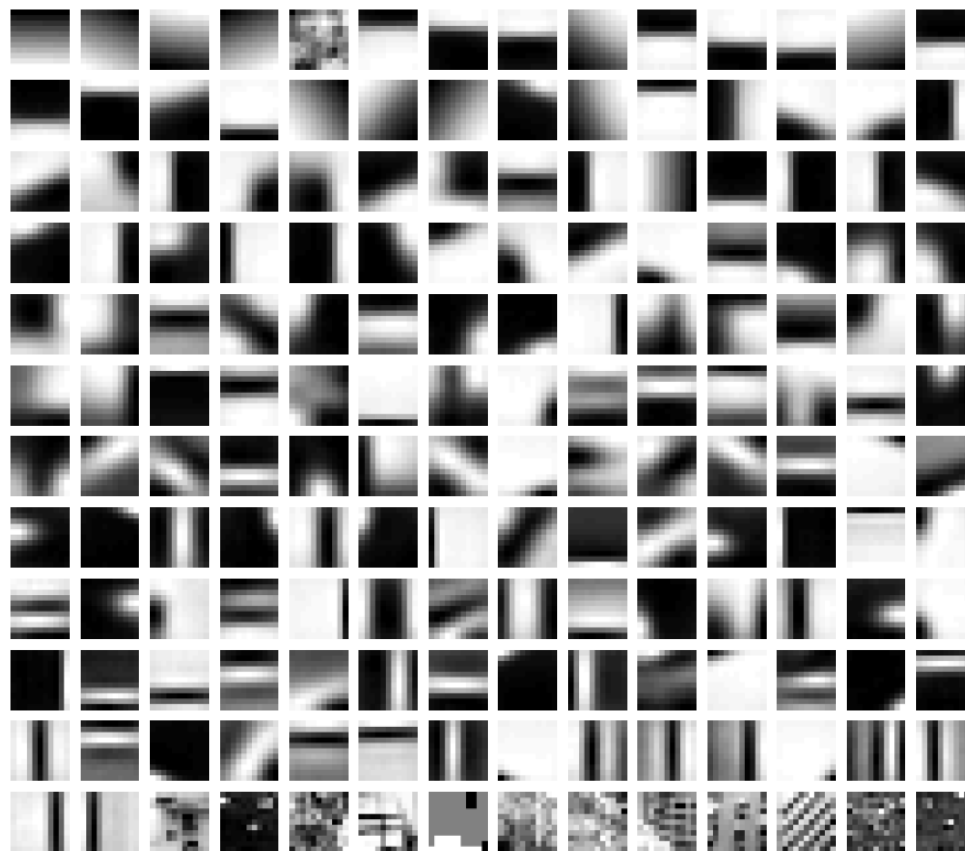
## 7.2.3.4 视觉特征提取--视觉词袋

### Technical outline for bag of features:

1. Extract features
2. Learn “visual vocabulary”
3. Quantize features using visual vocabulary
4. Represent images by frequencies of “visual words”



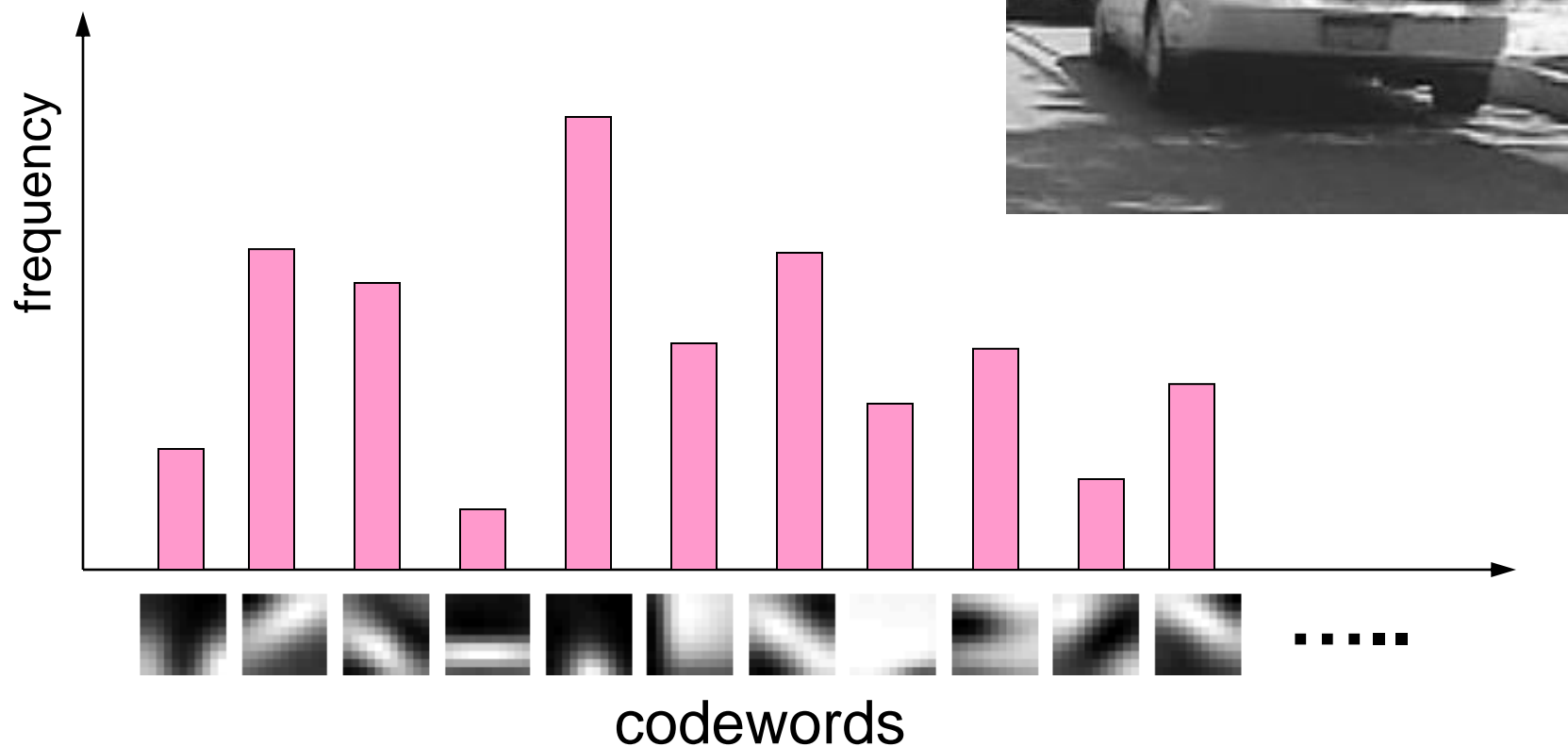
## 7.2.3.4 视觉特征提取--视觉词袋



Example Visual Vocabulary

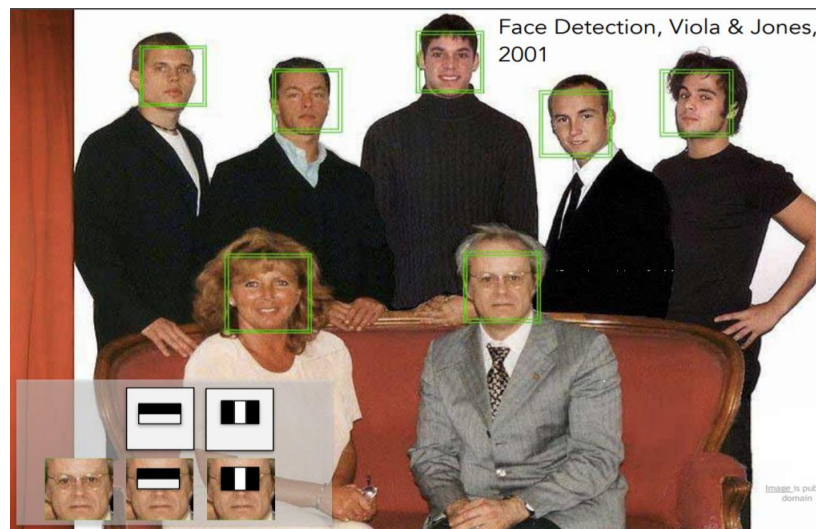
## 7.2.3.4 视觉特征提取--视觉词袋

For image representation:



## 7.2.3.5 视觉特征提取--Haar

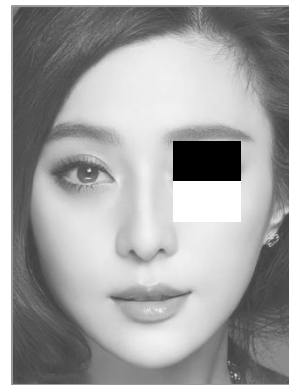
- 人脸检测领域的经典工作
- 将人脸检测带入“实时”时代
- 提出的积分图技术加快了特征计算



- ✓ P, Viola, M. Jones. Rapid object detection using a boosted cascade of simple features, CVPR, 2001
- ✓ P, Viola, M. Jones. Robust real-time face detection. IJCV, 57(2), 137-154, 2004

## 7.2.3.5 视觉特征提取--Haar

- 一个Haar特征由一组方形滤波器组成
- 滤波器响应值为**对应区域内像素值的和**
- 一个Haar特征的响应值为**白色滤波器响应值减去灰色滤波器响应值**
- 形状与Haar小波类似



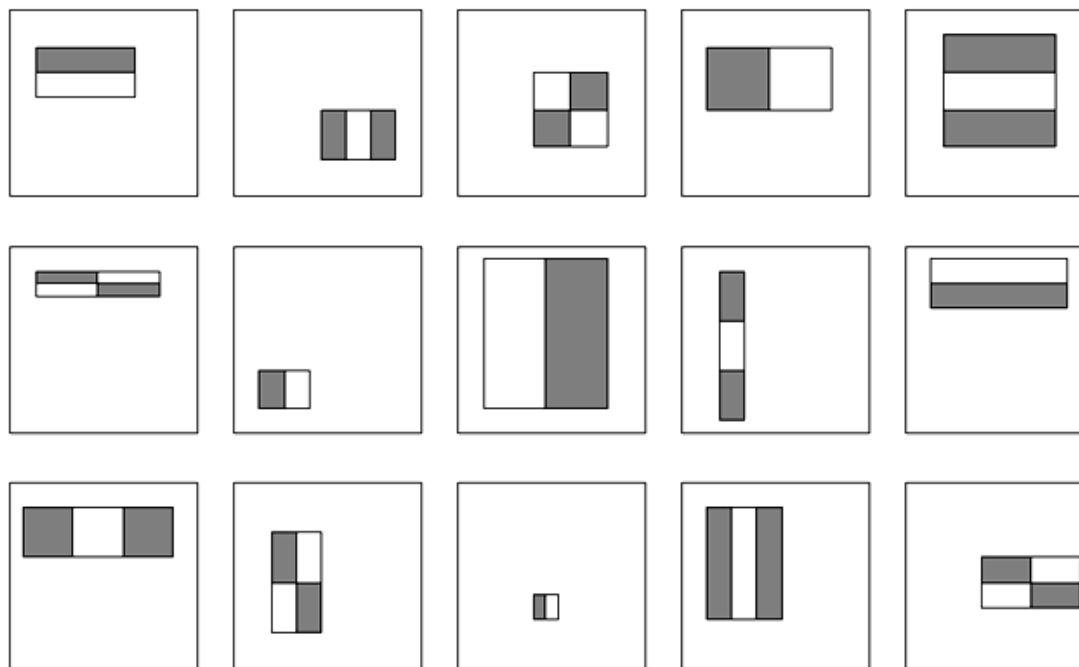
模板黑白可以翻转

$$h_t(x, y, w, h, type) = \begin{cases} 1, & \text{if Haar}(x, y, w, h, type) < \theta_t \\ 0, & \text{otherwise} \end{cases}$$

含义：通过比较Haar特征值是否超过某个阈值来判断是否为人脸

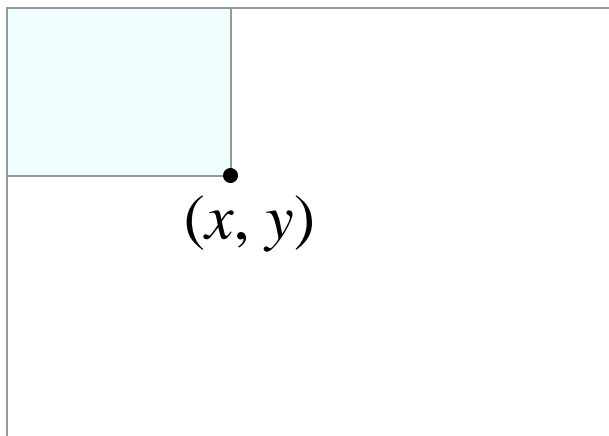
## 7.2.3.5 视觉特征提取--Haar

**直接计算复杂度高：**可能的Haar特征非常多，对于一个24x24的图像窗口，候选的Haar特征有大约160,000个



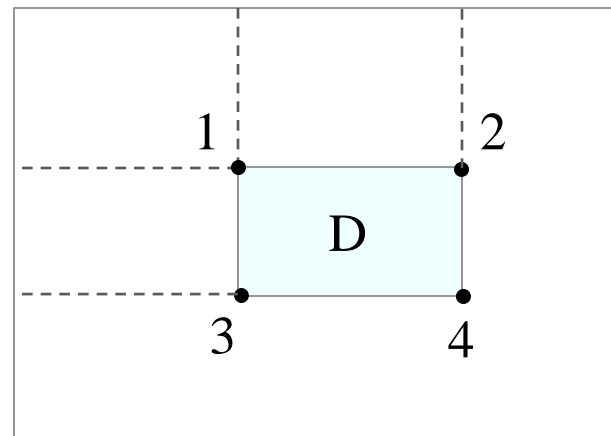
## 7.2.3.5 视觉特征提取--Haar

- 利用**积分图快速计算**方形滤波器响应值，快速计算haar特征
  - 积分图：快速计算指定区域内的像素灰度的总和，因此该概念与计算Haar特征 $\text{Haar}(x, y, w, h, \text{type})$ 有关。
  - 积分图：每个点上的值对应图像左上角区域的像素和。



$$I'(x, y) = \sum_{u \leq x, v \leq y} I(u, v)$$

积分图的计算



$$\text{sum}(D) = I'(4) + I'(1) - I'(2) - I'(3)$$

任意区域像素和的计算



## 7.2.3.5 视觉特征提取--Haar

- 只利用1个haar特征进行人脸检测，很弱
- 3个臭皮匠，超过1个诸葛亮
- 集成大量haar特征的判定结果，来判定给定图像是否人脸

Adaboost：一种强有力的特征选择和分类器集成算法。

$$G(\mathbf{x}) = \text{sign}(f(\mathbf{x})) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(\mathbf{x}) + b\right)$$

可以增加一个偏移，  
增强模型的表达能力

## 7.2.3.6 视觉特征提取--HoG

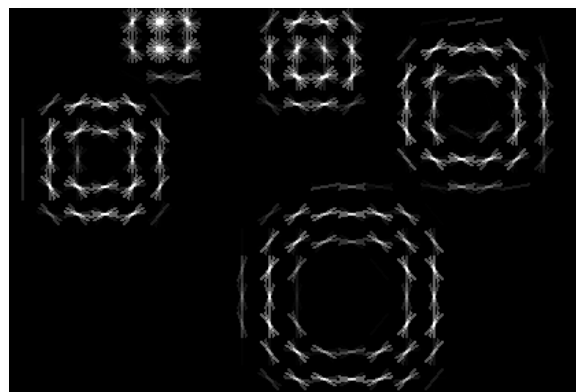
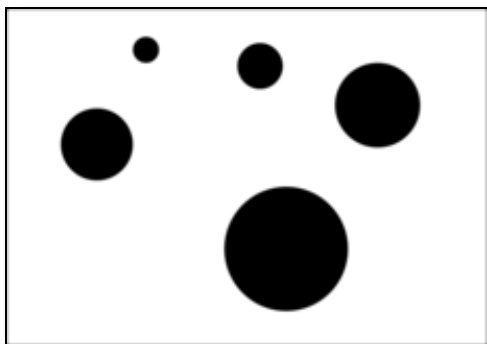
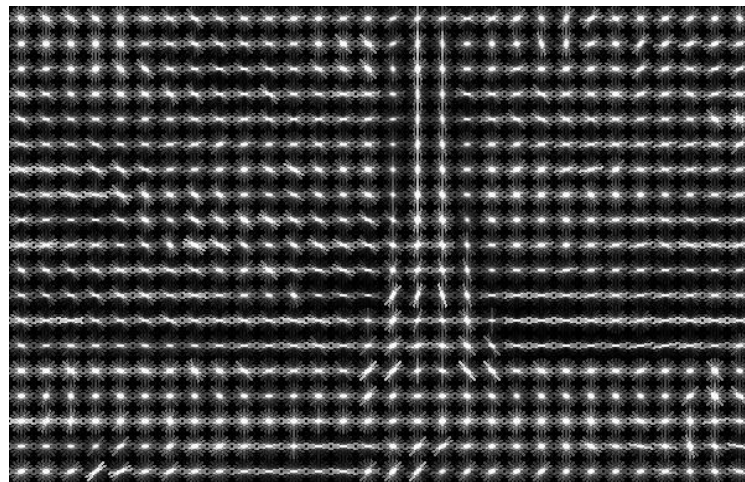
### Histogram of Oriented Gradient, HOG

- 2005年，由法国INRIA研究人员Dalal和Triggs提出，用于行人检测。
- 最早用于行人检测，后来发展成为面向一般物体检测的特征提取方法
- 引用超过20000次

N. Dalas, B. Triggs. Histogram of Oriented Gradients for human detection, CVPR, 2005.

## 7.2.3.6 视觉特征提取--HoG

- ✓ HoG计算的是图像梯度方向直方图，本质上与SIFT特征描述子一样，但空间统计直方图方式不一样。



## 7.2.3.6 视觉特征提取--HoG

### HoG特征提取技术路线：

1. 对输入图像做预处理（颜色转换、Gamma校正）；
2. 计算图像的梯度信息（幅值和方向）；
3. 将图像区域划分成若干个cell（如9x9）；
4. 统计每个cell区域内的梯度方向直方图；
5. 将相邻区域（如3x3）内的cell组成block（block之间具有重叠的cell），并将block内所有cell串联起来，进行归一化，得到block的特征。
6. 将所有block的特征串联起来组成图像HoG特征。

HoG特征维度比较高，通常结合线性SVM进行分类。

## 7.3 特征变换

- 特征提取的两层含义
  - 从数据观测获得原始特征表达的过程，例如：将一幅图像表示成一个向量；将一个文本转化成一个向量；将一段语音表示成一个向量等。
  - 从一组已有特征进行变换，得到新特征的过程（特征变换）。

线性变换：
$$\mathbf{y} = \mathbf{W}^T \mathbf{x}$$

非线性变换：
$$\mathbf{y} = \mathbf{W}(\mathbf{x})$$

特征提取的**最终形式**都是使用向量来表示数据样本，便于分析。

## 7.3.1 维数缩减

- 维数灾难

- 维数灾难最早由理查德·贝尔曼（Richard E. Bellman）在考虑优化问题时提出的，用于描述当空间维度增加时分析和组织高维空间中的数据会遇到各种问题。
  - 随着维数的增加，计算量呈指数倍增长。
  - 随着维数的增加，具有相同距离的两个样本其相似程度可以相差很远。
  - 当维度增加时，空间的体积增加得很快，可用数据变得稀疏。
  - 稀疏性对于任何要求“具有统计学意义的方法”而言都是一个問題。但是，为了获得在统计学上正确并且有可靠的结果，用来支撑这一结果所需要的数据量通常随着维数的增加而呈指数级增长。

## 7.3.1 维数缩减

- 维数缩减

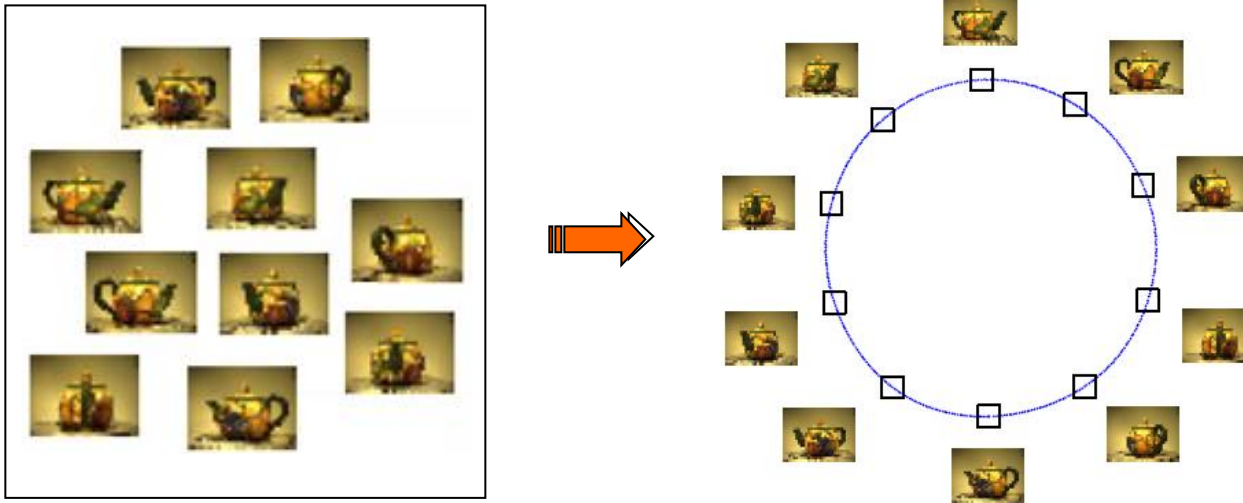
- 缓解维数灾难的一个重要途径是降维 (dimensionality reduction)，即通过某种数学变换将原始高维特征空间变换至某个低维“子空间”。在该子空间中，样本密度大幅度提高，距离计算也变得更加容易。

- 为什么能降维：

- 在很多时候，人们观测或收集到的数据虽然是高维的，但与学习任务密切相关的特征通常位于某个低维分布上，即高维空间中的一个低维“嵌入” (embedding)。
    - 感谢非均匀性祝福！

## 7.3.1 维数缩减

- 一个低维嵌入的例子





## 7.3.1 维数缩减

- 线性降维法

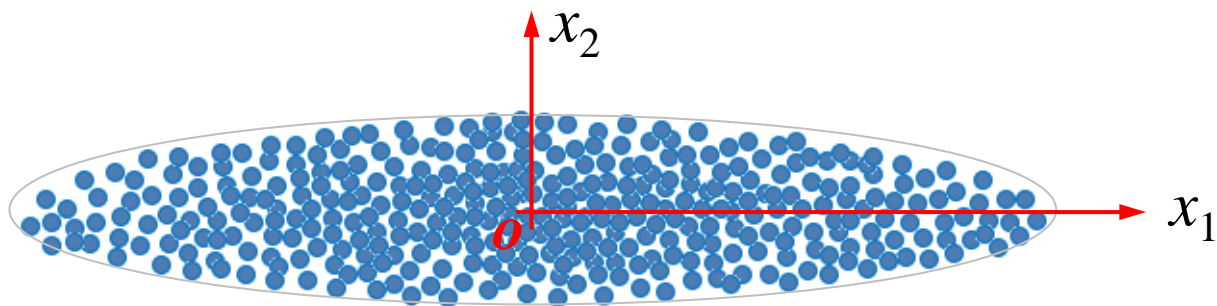
- 对高维空间中的样本 $\mathbf{x}$ 进行线性变换：

$$\mathbf{y} = \mathbf{W}^T \mathbf{x}, \quad \text{where } \mathbf{x} \in R^d, \mathbf{W} \in R^{d \times m}, \mathbf{y} \in R^m, m < d$$

- 变换矩阵 $\mathbf{W}=[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m]$ 可视为 $m$ 维空间中由 $m$ 个基向量组成的矩阵。
- $\mathbf{y}=\mathbf{W}^T \mathbf{x}$ 可视为样本 $\mathbf{x}$ 与 $m$ 个基向量分别做内积运算而得到，即 $\mathbf{x}$ 在新坐标系下的坐标。
- 新空间中的特征是原空间中特征的线性组合。
- 这就是线性降维法。
- 不同方法的差异：对低维子空间的性质有不同的要求，即对 $\mathbf{W}$ 施加不同的约束。

## 7.3.1 维数缩减

- 例子1



✓ 向 $x_1$ 轴投影：忽略每个样本的第二维，只保留 $\{x_1\}$

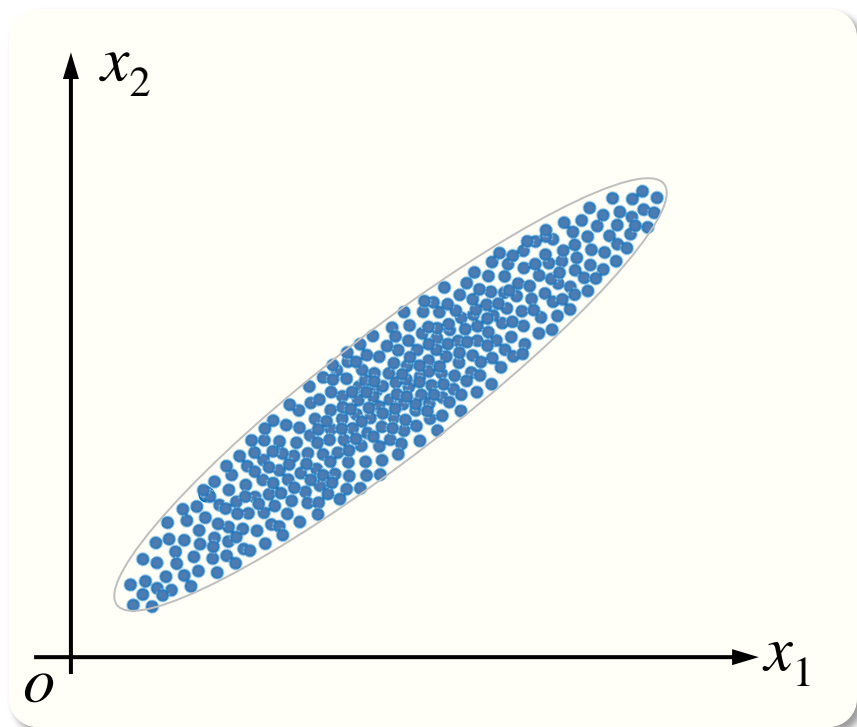
✓ 向 $x_2$ 轴投影：忽略每个样本的第一维，只保留 $\{x_2\}$

✓ 问题：

- 相对而言，对上述两种投影操作，哪一种将更多地保留原始数据集的信息？（沿 $x_2$ 轴有较小的方差）

## 7.3.1 维数缩减

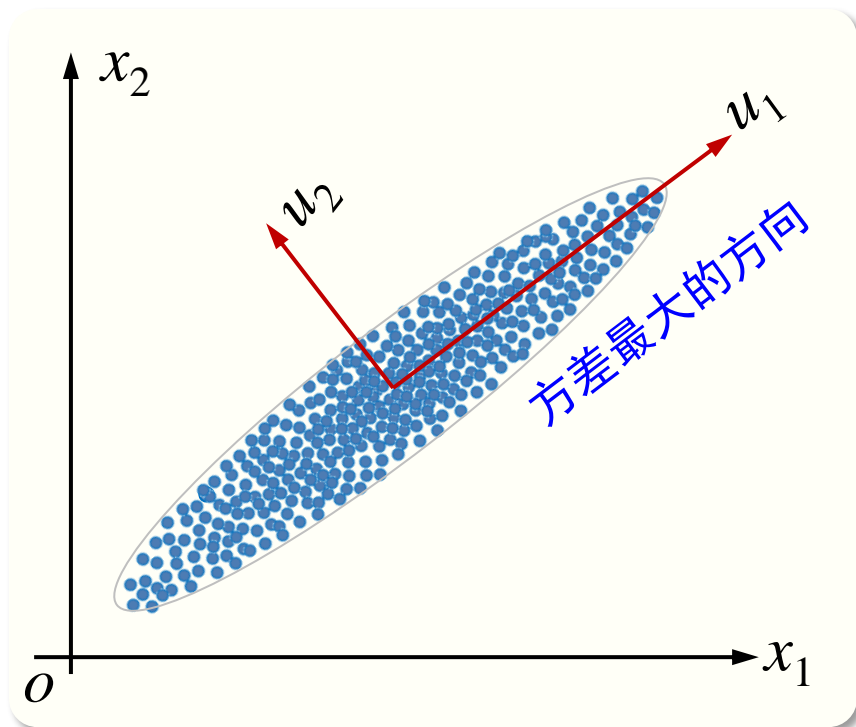
- 例子2



- ✓ 向 $x_1$ 轴投影：忽略每个样本的第二维，只保留 $\{x_1\}$
- ✓ 向 $x_2$ 轴投影：忽略每个样本的第一维，只保留 $\{x_2\}$
- ✓ 对上述两种投影操作均不能很好地保留原始数据集的信息。

## 7.3.1 维数缩减

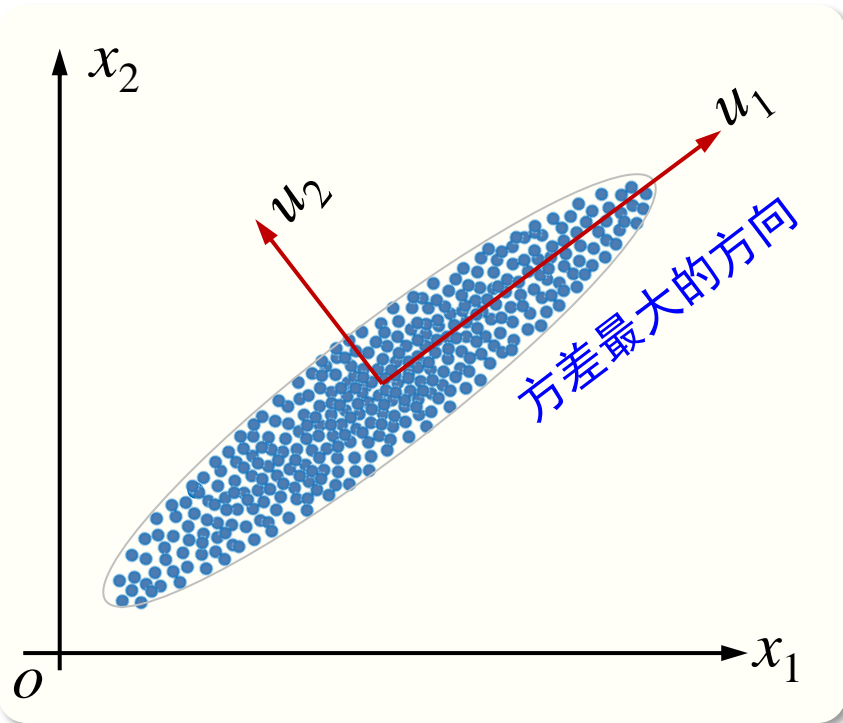
### • 例子2



- ✓ **向 $u_1$ 轴投影**：忽略每个样本的第二维，只保留 $\{u_1\}$
- ✓ **向 $u_2$ 轴投影**：忽略每个样本的第一维，只保留 $\{u_2\}$
- ✓ 在坐标系得到变换后（等价地，各个样本得到变换后），第一种投影操作仍然能够很好地保留原始数据集的信息。

## 7.3.2 主成分分析 (Principal Component Analysis , PCA)

- 动机



- ✓ 动机：寻找一组方差较大的方向，将原始数据（样本）在该方向进行投影。即将数据在新坐标系下进行表示，保留在少数方差最大的方向上的投影，达到数据变换、尽可能地保留原始数据信息和降维的目的。
- ✓ 方差较大的方向称为主成分 (Principal Component)。其中，方差最大的方向称为第一主成分，其次为第二主成分，依次类推。
- ✓ PCA: take top  $m$  PC's and project the data along those

# PCA: Finding Principal Components

- **Given:**  $n$  examples  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ , each example  $\mathbf{x}_n \in \mathbb{R}^d$ .
- **Goal:** we want to capture the maximum possible variance in the projected data, namely, projecting the data from  $d$  dimensions to  $m$  dimensions with  $m < d$ .
- Technically and algorithmically, let  $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m \in \mathbb{R}^d$  be the principal components, assumed to be:
  - **Orthogonal:**  $(\mathbf{w}_i)^T \mathbf{w}_j = 0, \forall i \neq j$ , and  $(\mathbf{w}_i)^T \mathbf{w}_i = 1, i, j = 1, 2, \dots, m$ .
- *We hope to obtain only the first  $m$  principal components.*



# PCA: Finding Principal Components

- The projection  $\mathbf{y}_i$  of a data point  $\mathbf{x}_i$  along  $\mathbf{w}_1$ :  $\mathbf{w}_1^T \mathbf{x}_i$
- The projection  $\bar{\mathbf{y}}$  of the mean  $\bar{\mathbf{x}}$  along  $\mathbf{w}_1$ :

$$\bar{\mathbf{y}} = \mathbf{w}_1^T \bar{\mathbf{x}}, \quad \text{where} \quad \bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$$

- The variance of the projected data (along projection direction  $\mathbf{w}_1$ ):

$$\text{var} = \frac{1}{n} \sum_{i=1}^n \left( \mathbf{w}_1^T \mathbf{x}_i - \mathbf{w}_1^T \bar{\mathbf{x}} \right)^2$$

- We want to obtain direction  $\mathbf{w}_1$  that maximizes the projected data variance:

$$\begin{aligned} \max \quad & \frac{1}{n} \sum_{i=1}^n \left( \mathbf{w}_1^T \mathbf{x}_i - \mathbf{w}_1^T \bar{\mathbf{x}} \right)^2 \\ \text{s.t.} \quad & \mathbf{w}_1^T \mathbf{w}_1 = 1 \end{aligned}$$

# PCA: Finding Principal Components

- Note that:

$$\begin{aligned}\text{var} &= \frac{1}{n} \sum_{i=1}^n \left( \mathbf{w}_1^T \mathbf{x}_i - \mathbf{w}_1^T \bar{\mathbf{x}} \right)^2 \\&= \frac{1}{n} \sum_{i=1}^n \left( \mathbf{w}_1^T \mathbf{x}_i - \mathbf{w}_1^T \bar{\mathbf{x}} \right) \left( \mathbf{w}_1^T \mathbf{x}_i - \mathbf{w}_1^T \bar{\mathbf{x}} \right)^T \\&= \frac{1}{n} \sum_{i=1}^n \mathbf{w}_1^T (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})^T \mathbf{w}_1 \\&= \mathbf{w}_1^T \left( \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})^T \right) \mathbf{w}_1 \\&= \mathbf{w}_1^T \mathbf{C} \mathbf{w}_1\end{aligned}$$

$$\mathbf{C} = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})^T$$

covariance matrix of data

- Then we have:

$$\begin{aligned}\max \quad & \mathbf{w}_1^T \mathbf{C} \mathbf{w}_1, \\s.t. \quad & \mathbf{w}_1^T \mathbf{w}_1 = 1\end{aligned}$$

# PCA: Finding Principal Components

- Now we introduce a Lagrange multiplier  $\lambda$  to this subject, obtaining the following objective function:

$$obj = \mathbf{w}_1^T \mathbf{C} \mathbf{w}_1 - \lambda(\mathbf{w}_1^T \mathbf{w}_1 - 1)$$

$$\frac{\partial obj}{\partial \mathbf{w}_1} = 2\mathbf{C} \mathbf{w}_1 - 2\lambda \mathbf{w}_1$$

- Taking the derivative w. r. t.  $\mathbf{w}_1$  and setting it to zero gives:

$$\mathbf{C} \mathbf{w}_1 = \lambda \mathbf{w}_1$$

- This is just an eigenvalue equation. Thus,  $\mathbf{w}_1$  must be an eigenvector of  $\mathbf{C}$ .

# PCA: Finding Principal Components

- It is seen that  $\mathbf{w}_1$  must be an eigenvector of  $\mathbf{C}$ , and  $\lambda$  is the corresponding eigenvalue.
- But, there are multiple eigenvectors of  $\mathbf{C}$ , which one is  $\mathbf{w}_1$  ?
- Consider

$$\mathbf{w}_1^T \mathbf{C} \mathbf{w}_1 = \mathbf{w}_1^T \lambda \mathbf{w}_1 = \lambda \mathbf{w}_1^T \mathbf{w}_1 = \lambda$$

- We see that the projected data variance  $\mathbf{w}_1^T \mathbf{C} \mathbf{w}_1 = \lambda$  is maximum
  - Thus,  $\lambda$  should be the largest eigenvalue, and  $\mathbf{w}_1$  is the first (largest) eigenvector of  $\mathbf{C}$  (with eigenvalue  $\lambda$ ).
  - This is just the first principal component (direction of highest variance in the data)

# PCA: The Algorithm

- 计算数据均值:  $\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$
- 计算数据的协方差矩阵:  $\mathbf{C} = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$
- 对矩阵 $\mathbf{C}$ 进行特征值分解, 并取最大的 $m$ 个特征值( $\lambda_1 \geq \lambda_2 \geq \dots \lambda_m$ )对应的特征向量 $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m$ , 组成投影矩阵 $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m] \in R^{d \times m}$
- 将每一个数据进行投影:  $\mathbf{y}_i = \mathbf{W}^T \mathbf{x}_i \in R^m, i=1, 2, \dots, n$

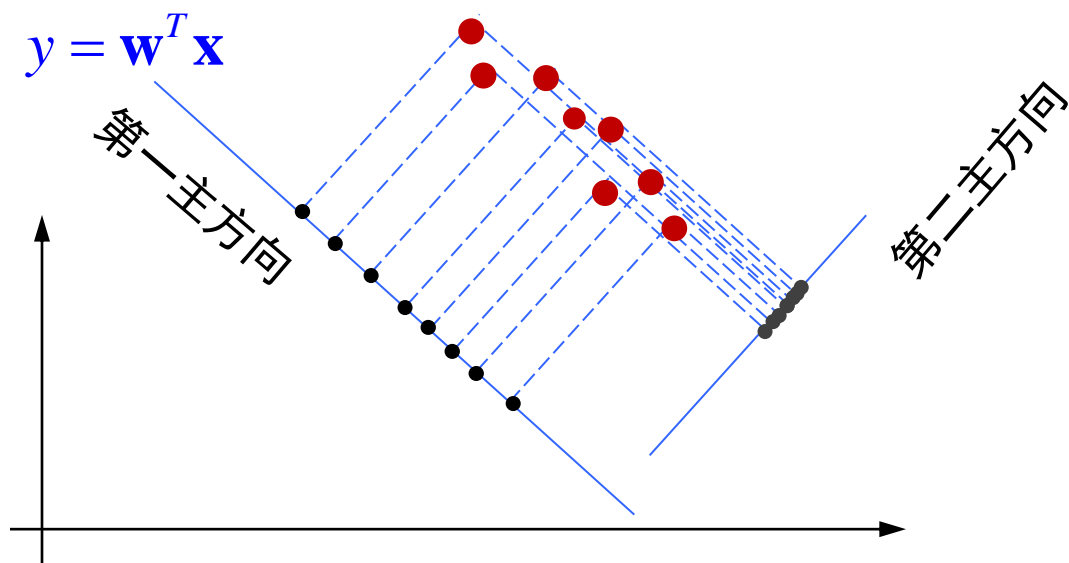
## 7.3.3 主成分分析--进一步的分析

- **PCA (Principal Component Analysis) 基本思想**
  - 再思考：如何仅用一个超平面从整体上对所有样本  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\} \subset R^d$  进行恰当表示？
  - 通常有如下两种思路：
    - **可区分性**：样本点在这个超平面上的投影能够尽可能地分开。
    - **可重构性**：样本到这个超平面的距离都足够近；



## 7.3.3 主成分分析--进一步的分析

- PCA 一采用最大可分性观点
  - 使所有样本点的投影尽可能地分开，则需最大化投影点的方差：



- PCA—采用最大可分性观点

- 采用最大可分性的观点，投影后获得的样本点为：

$$\mathbf{y}_i = \mathbf{W}^T \mathbf{x}_i \in \mathbf{R}^m, \quad i = 1, 2, \dots, n$$

由于数据点是零均值化的，则  $\sum_{i=1}^n \mathbf{y}_i = \mathbf{W}^T \sum_{i=1}^n \mathbf{x}_i = \mathbf{0}$

因此，投影后的样本点的（协）方差为

$$\sum_{i=1}^n \mathbf{W}^T \mathbf{x}_i \mathbf{x}_i^T \mathbf{W} = \mathbf{W}^T \mathbf{X} \mathbf{X}^T \mathbf{W}$$

要使数据具有最大可分性，就应该使数据尽量分散开来，因此应该使其方差最大。考虑多维情形，我们有：

$$\max_{\mathbf{W} \in \mathbf{R}^{m \times d}} \text{tr}(\mathbf{W}^T \mathbf{X} \mathbf{X}^T \mathbf{W}), \quad s.t. \quad \mathbf{W}^T \mathbf{W} = \mathbf{I}$$

## 7.3.3 主成分分析--进一步的分析

- PCA求解

- 采用拉格朗日乘子法，经过简单矩阵运算，我们有：

$$\mathbf{X}\mathbf{X}^T\mathbf{W} = \lambda\mathbf{W}$$

于是，只需要对协方差矩阵 $\mathbf{X}\mathbf{X}^T$ 进行特征值分解，并对特征值进行排序： $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \dots \geq \lambda_m$ ，取前 $m$ 个特征值对应的特征向量构成变换矩阵 $\mathbf{W}$ 。

这就是主成分分析的解。

## 7.3.3 主成分分析--进一步的分析

- PCA中主成分的概念

对协方差矩阵 $\mathbf{XX}^T$ 进行特征值分解，并对特征值进行排序： $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \dots \geq \lambda_m$ ，取前 $m$ 个特征值对应的特征向量构成变换矩阵 $\mathbf{W}$ 。

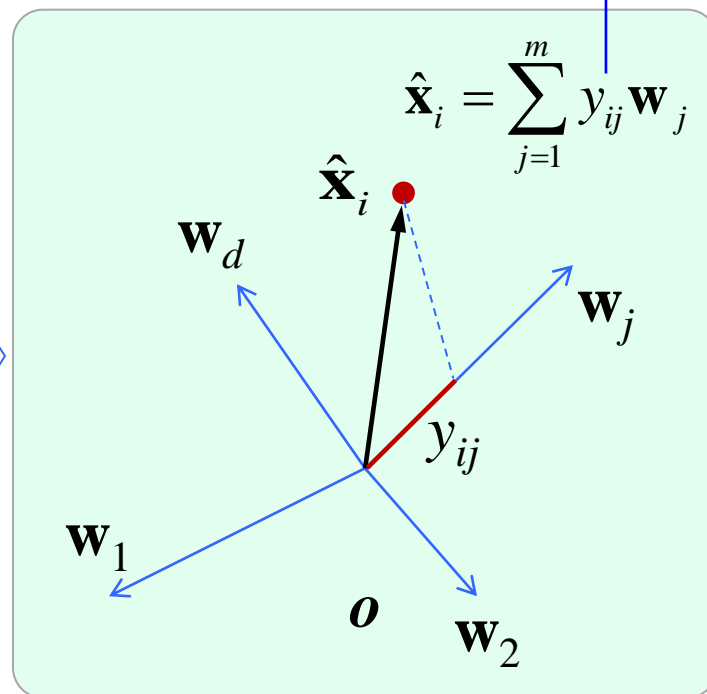
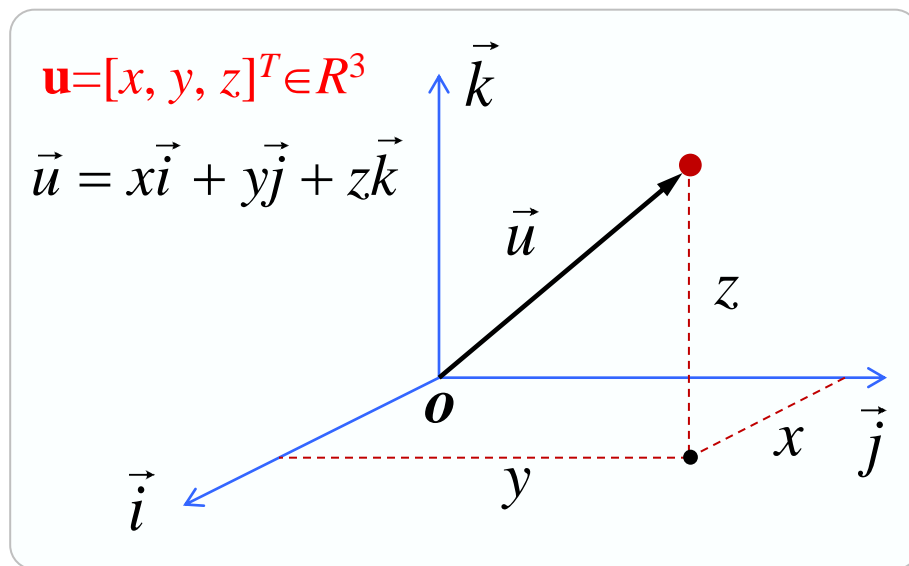
主成分(主方向、主轴)：

$\lambda_1$ 对应的特征向量称为第一主成分，其他依此类推。

## 7.3.3 主成分分析--进一步的分析

- PCA — 采用重构的观点

- 回忆向量空间中一个矢量的表示方法



$$y_{ij} = (\hat{\mathbf{x}}_i \cdot \mathbf{w}_j)$$

从三维空间到高维空间

- PCA—采用重构的观点

- 由 $\mathbf{W}$ 定义新坐标系：假定投影变换是正交变换，即新坐标系由 $\mathbf{W}=[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m]$ 来表示 ( $m < d$ )， $\mathbf{w}_i$ 的模等于1， $\mathbf{w}_i$ 与 $\mathbf{w}_j$ 两两正交。

- 设样本点 $\mathbf{x}_i$ 在新坐标系下的坐标为：

$$\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{id}]^T \in R^m$$

- 在正交坐标系下，对样本点 $\mathbf{x}_i$ ，有新坐标：

$$y_{ij} = \mathbf{w}_j^T \mathbf{x}_i, \quad \mathbf{w}_j \in R^d, \quad j = 1, 2, \dots, m$$

- 在新坐标系下，可得 $\mathbf{x}_i$ 的新表示：

$$\hat{\mathbf{x}}_i = \sum_{j=1}^m y_{ij} \mathbf{w}_j, \quad i = 1, 2, \dots, n$$

- PCA —采用重构的观点

- 重构误差:

$$\sum_{i=1}^n \|\mathbf{x} - \hat{\mathbf{x}}_i\|_2^2 = \sum_{i=1}^n \left\| \mathbf{x}_i - \sum_{j=1}^m y_{ij} \mathbf{w}_j \right\|_2^2 = \sum_{i=1}^n \|\mathbf{x}_i - \mathbf{W} \mathbf{y}_i\|_2^2$$

$$= \sum_{i=1}^n \left( (\mathbf{W} \mathbf{y}_i)^T \mathbf{W} \mathbf{y}_i - 2 \mathbf{x}_i^T \mathbf{W} \mathbf{y}_i + \mathbf{x}_i^T \mathbf{x}_i \right)$$

$$(\because \mathbf{W}^T \mathbf{W} = \mathbf{I}) = \sum_{i=1}^n \left( \mathbf{y}_i^T \mathbf{y}_i - 2 \mathbf{y}_i^T \mathbf{x}_i + \mathbf{x}_i^T \mathbf{x}_i \right)$$

$$(\because \mathbf{y}_i = \mathbf{W}^T \mathbf{x}_i) = - \sum_{i=1}^n \mathbf{y}_i^T \mathbf{x}_i + const = - \sum_{i=1}^n \left( \mathbf{W}^T \mathbf{x}_i \right)^T \left( \mathbf{W}^T \mathbf{x}_i \right) + const$$

$$= -tr \left( \sum_{i=1}^n \left( \mathbf{W}^T \mathbf{x}_i \right)^T \left( \mathbf{W}^T \mathbf{x}_i \right) \right) + const$$

$$\begin{aligned} & (\because tr(\mathbf{AB}) = tr(\mathbf{BA})) \\ (\because tr(\mathbf{A}) + tr(\mathbf{B}) = tr(\mathbf{A} + \mathbf{B})) & = -tr \left( \mathbf{W}^T \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^T \mathbf{W} \right) + const \end{aligned}$$



- PCA —采用重构的观点

- 进一步，假定数据已经零均值化，即  $\sum_{i=1}^n \mathbf{x}_i = \mathbf{0}$

令  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in R^{d \times n}$

$$\begin{aligned} \sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2 &= -tr \left( \mathbf{W}^T \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^T \mathbf{W} \right) + const \\ &= -tr \left( \mathbf{W}^T \mathbf{X} \mathbf{X}^T \mathbf{W} \right) + const \end{aligned}$$

于是，获得主成分分析的最优化模型：

$$\max_{\mathbf{W} \in R^{d \times m}} tr \left( \mathbf{W}^T \mathbf{X} \mathbf{X}^T \mathbf{W} \right), \quad s.t. \quad \mathbf{W}^T \mathbf{W} = \mathbf{I}$$

## 7.3.3 主成分分析--进一步的分析

- 讨论:

- 降低至多少维数: 
$$\frac{\sum_{i=1}^m \lambda_i}{\sum_{i=1}^d \lambda_i} \geq t \quad (\text{比如, } t=95\%)$$

- 也可以采用交叉验证, 结合最近邻分类器来选择合适的维度 $m$ 。
  - 舍弃 $m-d$ 个特征值对应的特征向量导致了维数缩减。
    - 舍弃这些信息之后能使样本的采样密度增大, 这正是降维的重要动机。
    - 另外, 当数据受到噪声影响时, 最小的特征值所对应的特征向量往往与噪声有关, 将它们舍弃可在一定程度上起到去噪的效果。

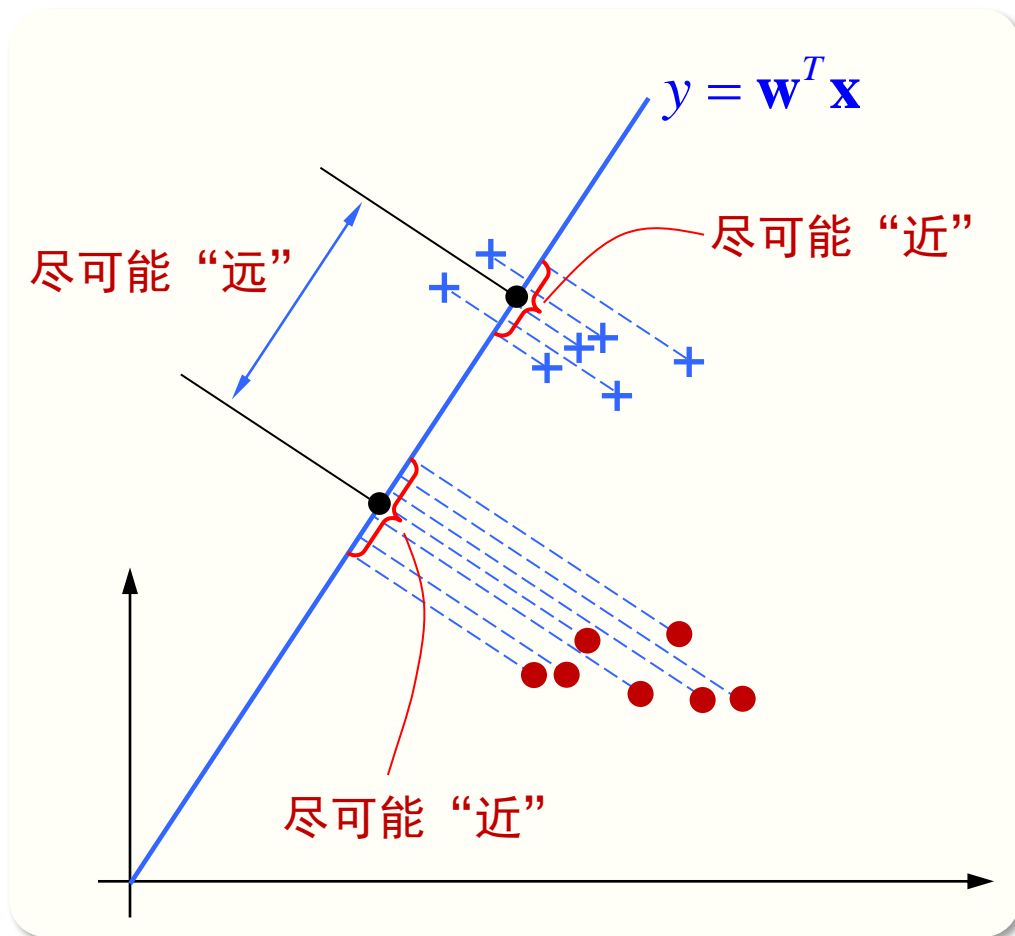
## 7.4 线性判别分析

- 算法思想

- 线性判别分析（Linear Discriminant Analysis, LDA）是一种经典的线性学习方法。
- LAD的思想较直观：对于两类分类问题，给定训练集，设法将样本投影到一条直线上，使得同类样本的投影点尽可能接近，不同类样本的投影点尽可能相互远离。
- 在对新样本进行分类时，将其投影到这条直线上，再根据投影点的位置来判断其类别。

## 7.4 线性判别分析

- 算法思想



- ✓ **动机**：寻找一组投影方向，使样本在投影之后（即在新坐标系下）类内样本点尽可能靠近，类间样本点尽可能相互远离，提升样本表示的分类鉴别能力。
- ✓ **投影方向数小于原始数据的维度**，因此投影样本即相当于将样本在子空间内进行表示，从而达到降维的目的。

## 7.4 线性判别分析

- 算法思想

- 样本集  $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$ ,  $y_i \in \{0, 1\}$
- 令  $\mathbf{X}_i$ 、 $\boldsymbol{\mu}_i$ 、 $\boldsymbol{\Sigma}_i$  分别表示第  $i \in \{0, 1\}$  类的示例集合、均值向量、协方差矩阵。
- 若将数据投影到直线（一个坐标轴）上，则两类样本的中心在直线上的投影分别为  $\mathbf{w}^T \boldsymbol{\mu}_0$  和  $\mathbf{w}^T \boldsymbol{\mu}_1$ ；两类样本的协方差分别为  $\mathbf{w}^T \boldsymbol{\Sigma}_0 \mathbf{w}$  和  $\mathbf{w}^T \boldsymbol{\Sigma}_1 \mathbf{w}$ 。
- 欲使同类样本的投影点尽可能接近，可让同类样本投影点的协方差尽可能小，即  $\mathbf{w}^T \boldsymbol{\Sigma}_0 \mathbf{w} + \mathbf{w}^T \boldsymbol{\Sigma}_1 \mathbf{w}$  尽可能小。
- 欲使异类样本的投影点尽可能远离，可让类中心点之间的距离尽可能大，即  $\|\mathbf{w}^T \boldsymbol{\mu}_0 - \mathbf{w}^T \boldsymbol{\mu}_1\|^2$  尽可能大。

## 7.4 线性判别分析

- 算法思想：最大化如下目标函数

$$J = \frac{\|\mathbf{w}^T \boldsymbol{\mu}_0 - \mathbf{w}^T \boldsymbol{\mu}_1\|_2^2}{\mathbf{w}^T \boldsymbol{\Sigma}_0 \mathbf{w} + \mathbf{w}^T \boldsymbol{\Sigma}_1 \mathbf{w}}$$

两个类的中心尽可能远

两类的类内协方差尽可能小

$$= \frac{\mathbf{w}^T (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^T \mathbf{w}}{\mathbf{w}^T (\boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1) \mathbf{w}}$$

根据上述算法思想，可定义一些量，将算法进行推广。

## 7.4 线性判别分析

- LAD算法

- 类内散度矩阵：
$$\mathbf{S}_w = \Sigma_0 + \Sigma_1$$
$$= \sum_{\mathbf{x} \in X_0} (\mathbf{x} - \boldsymbol{\mu}_0)(\mathbf{x} - \boldsymbol{\mu}_0)^T + \sum_{\mathbf{x} \in X_1} (\mathbf{x} - \boldsymbol{\mu}_1)(\mathbf{x} - \boldsymbol{\mu}_1)^T$$
- 类间散度矩阵：
$$\mathbf{S}_b = (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^T$$
- 目标函数重写为（广义Rayleigh商）：
$$J = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$$

**注意：**  $J$  的值与向量的长度无关，只与其方向有关，不失一般性可令  $\mathbf{w}$  为单位长度的向量。



## 7.4 线性判别分析

- LAD算法

- 学习目标:  $\max \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}, \quad s.t. \quad \mathbf{w}^T \mathbf{w} = 1$

- 由于目标函数值与长度无关（只与方向有关），因此可采用一种更直观的方法：令  $\mathbf{w}^T \mathbf{S}_w \mathbf{w} = 1$ :

$$\max \mathbf{w}^T \mathbf{S}_b \mathbf{w}, \quad s.t. \quad \mathbf{w}^T \mathbf{S}_w \mathbf{w} = 1$$

- 根据拉格朗日乘子法，于是有：

$$\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w} \quad \Rightarrow \quad \mathbf{S}_w^{-1} \mathbf{S}_b \mathbf{w} = \lambda \mathbf{w}$$

上式表明： $\mathbf{w}$ 为是矩阵  $\mathbf{S}_w^{-1} \mathbf{S}_b$  的特征向量。

## 7.4 线性判别分析

- LAD算法：构造性求解方法

$$\mathbf{S}_b = (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^T$$

$$\mathbf{S}_b \mathbf{w} = (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^T \mathbf{w} = s \cdot (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1), \quad \begin{matrix} \swarrow \text{标量} \\ s = (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^T \mathbf{w} \in R \end{matrix}$$

上式表明： $\mathbf{S}_b \mathbf{w}$  方向与  $\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1$  的方向相同。不妨令：

$$\mathbf{S}_b \mathbf{w} = \lambda(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

$$\Rightarrow \mathbf{w} = \mathbf{S}_w^{-1}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

- 多类LAD算法 (设类别数为 $c$ )

- 全局散度矩阵:  $\mathbf{S}_t = \mathbf{S}_w + \mathbf{S}_b = \sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T, \quad \boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$

- 类内散度矩阵:  $\mathbf{S}_w = \sum_{j=1}^c \mathbf{S}_{wj},$

其中,  $\mathbf{S}_{wj} = \sum_{\mathbf{x} \in X_j} (\mathbf{x} - \boldsymbol{\mu}_j)(\mathbf{x} - \boldsymbol{\mu}_j)^T, \quad \boldsymbol{\mu}_j = \frac{1}{n_j} \sum_{\mathbf{x} \in X_j} \mathbf{x}$

- 类间散度矩阵:

$$\mathbf{S}_b = \mathbf{S}_t - \mathbf{S}_w = \sum_{j=1}^c n_j (\boldsymbol{\mu}_j - \boldsymbol{\mu})(\boldsymbol{\mu}_j - \boldsymbol{\mu})^T$$

其中,  $n_j$  为属于第  $j$  类的样本个数。

思考题: 试证明矩阵 $\mathbf{S}_b$ 的秩小于等于 $c-1$ !

# 7.4 线性判别分析

- 多类LAD算法

**Problem 1:**  
(迹比值最大化)

$$\max \frac{\text{tr}(\mathbf{W}^T \mathbf{S}_b \mathbf{W})}{\text{tr}(\mathbf{W}^T \mathbf{S}_w \mathbf{W})}, \quad s.t. \quad \mathbf{W}^T \mathbf{W} = \mathbf{I}$$

最大化投影  
后的距离

**Problem 2:**  
(行列式比值最大化)

$$\max \frac{|\mathbf{W}^T \mathbf{S}_b \mathbf{W}|}{|\mathbf{W}^T \mathbf{S}_w \mathbf{W}|}, \quad s.t. \quad \mathbf{W}^T \mathbf{W} = \mathbf{I}$$

最大化投影后数  
据分布的体积

行列式

## 7.4 线性判别分析

- 多类LAD算法

- 需要指出的是：Problem 1与 Problem 2的解是不同的。  
Problem 2的解可以通过如下广义特征值问题求解得到：

$$\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$$

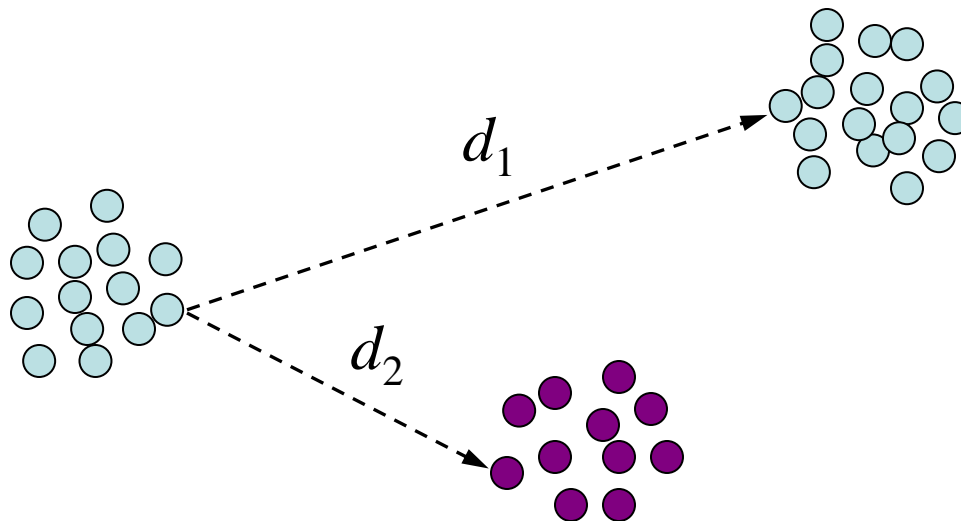
- Problem 1的求解较复杂，可以参考如下文献：

Shiming Xiang, Feiping Nie, Changshui Zhang. Learning a Mahalanobis distance metric for data clustering and classification. Pattern Recognition, 41(12), Pages 3600 - 3612, 2008

$$\max \frac{\text{tr}(\mathbf{W}^T \mathbf{S}_b \mathbf{W})}{\text{tr}(\mathbf{W}^T \mathbf{S}_w \mathbf{W})}, \quad s.t. \quad \mathbf{W}^T \mathbf{W} = \mathbf{I}$$

## 7.5 局部线性判别分析 (扩展内容, 不讲)

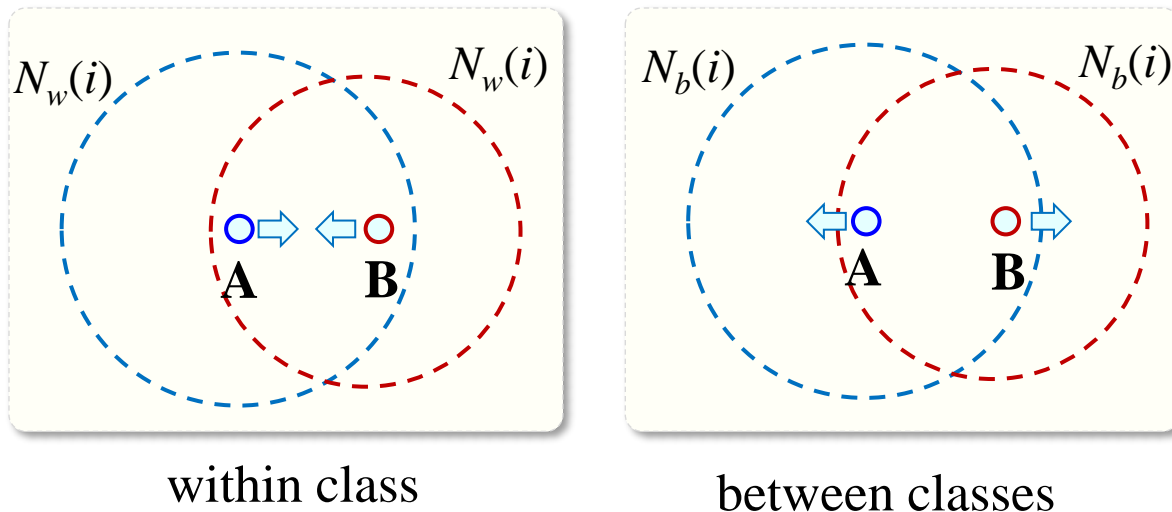
- 局限性
  - In each class, the distribution of data is Gaussian



We hope  $d_1 < d_2$ . **But difficult, or impossible!**

# 7.5 局部线性判别分析

- **Techniques of Local Analysis**
  - Neighborhood constraints (方法一)
  - Locally weighting (方法二)
    - Weighting for 1-NN
    - Local Fisher discriminant analysis
- **Basic Motivation**





# 7.5 局部线性判别分析

- **Modify  $S_w$  and  $S_b$** 
  - Neighborhood Constraints:

$$S_w = \sum_{\substack{y_i=y_j \\ \mathbf{x}_i \in N(\mathbf{x}_j), \mathbf{x}_j \in N(\mathbf{x}_i)}} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$

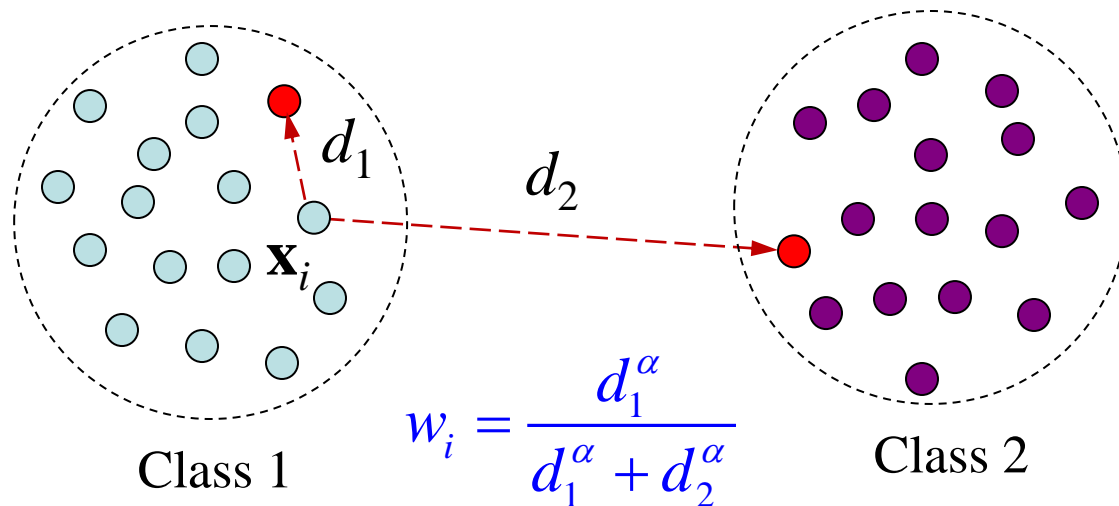
$$S_b = \sum_{\substack{y_i \neq y_j \\ \mathbf{x}_i \in N(\mathbf{x}_j), \mathbf{x}_j \in N(\mathbf{x}_i)}} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$

# 7.5 局部线性判别分析

- **Nearest Neighbor Discriminant Analysis, NNDA**

- 近邻加权

- A problem in neighborhood constraints is the selection of the number of nearest neighbors ( $k$ )

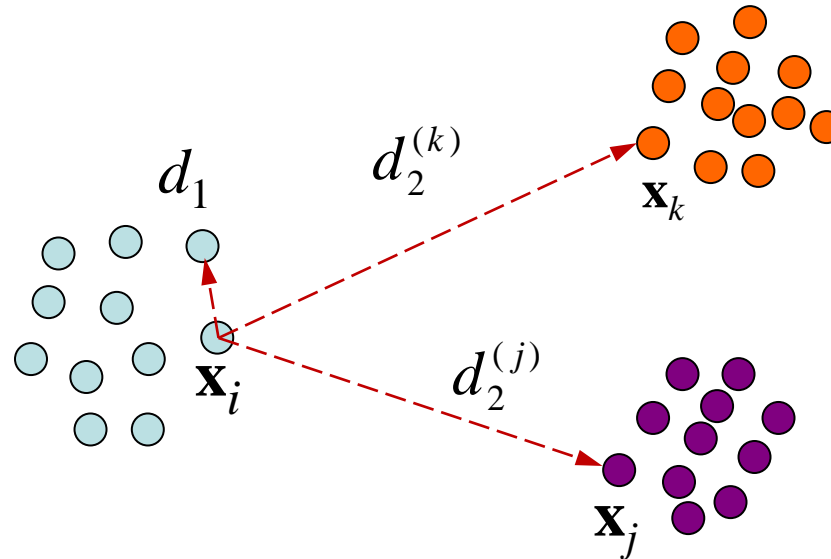


最近邻加权

# 7.5 局部线性判别分析

- **Nearest Neighbor Discriminant Analysis, NNDA**

多类最近邻加权



$$w_i = \frac{d_1^\alpha}{d_1^\alpha + (\min\{d_2^{(j)}\})^\alpha}, \quad (0 < \alpha < 1)$$

# 7.5 局部线性判别分析

- **Nearest Neighbor Discriminant Analysis, NNDA**

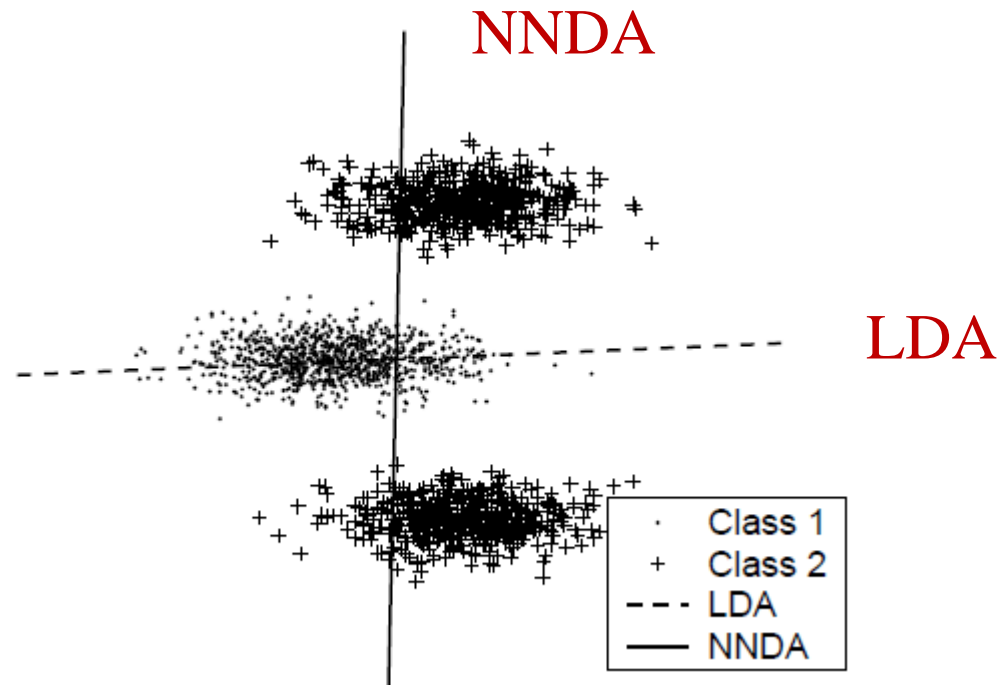
$$\mathbf{S}_w = \sum_{\substack{y_i = y_j \\ \mathbf{x}_j \in N_{1-nn}(\mathbf{x}_i), i=1,2,\dots,n}} w_i (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$

$$\mathbf{S}_b = \sum_{\substack{y_i \neq y_j \\ \mathbf{x}_j \in N_{1-nn}(\mathbf{x}_i), i=1,2,\dots,n}} w_i (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$

Xipeng Qiu, Lide Wu: Stepwise Nearest Neighbor Discriminant Analysis.  
IJCAI 2005: 829-834

## 7.5 局部线性判别分析

- **Nearest Neighbor Discriminant Analysis, NNDA**



NNDA finds the correct projection direction, but LDA failed !

# 7.5 局部线性判别分析

- **Local Fisher Discriminant Analysis, LFDA**

- Motivation

- LFDA does not impose far-apart data pairs of the same class to be close, by which local structure of the data tends to be preserved.
    - 邻域加权 (Locally Weighting)

Masashi Sugiyama, Local Fisher Discriminant Analysis for Supervised Dimensionality Reduction, ICML, 2006

## 7.5 局部线性判别分析

- Step1: Construct an affine matrix for  $n$  data points:

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{pmatrix}$$

$$A_{ij} = \begin{cases} 1, & \text{if } \mathbf{x}_j \text{ is a neighbor of } \mathbf{x}_i \\ 0, & \text{otherwise} \end{cases}, \text{ or}$$

$$A_{ij} = \begin{cases} \exp(-\|\mathbf{x}_j - \mathbf{x}_i\|^2 / (2\sigma^2)), & \text{if } \mathbf{x}_j \text{ is a neighbor of } \mathbf{x}_i \\ 0, & \text{otherwise} \end{cases}$$



## 7.5 局部线性判别分析

- Step2: Modify  $\mathbf{S}_w$  and  $\mathbf{S}_b$ :

$$\mathbf{S}_w = \frac{1}{2} \sum_{i,j} A_{ij}^{(w)} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$



$$\mathbf{S}_w = \frac{1}{2} \sum_{i,j} \bar{A}_{ij}^{(w)} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$

$$\mathbf{S}_b = \frac{1}{2} \sum_{i,j=1}^n A_{ij}^{(b)} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$



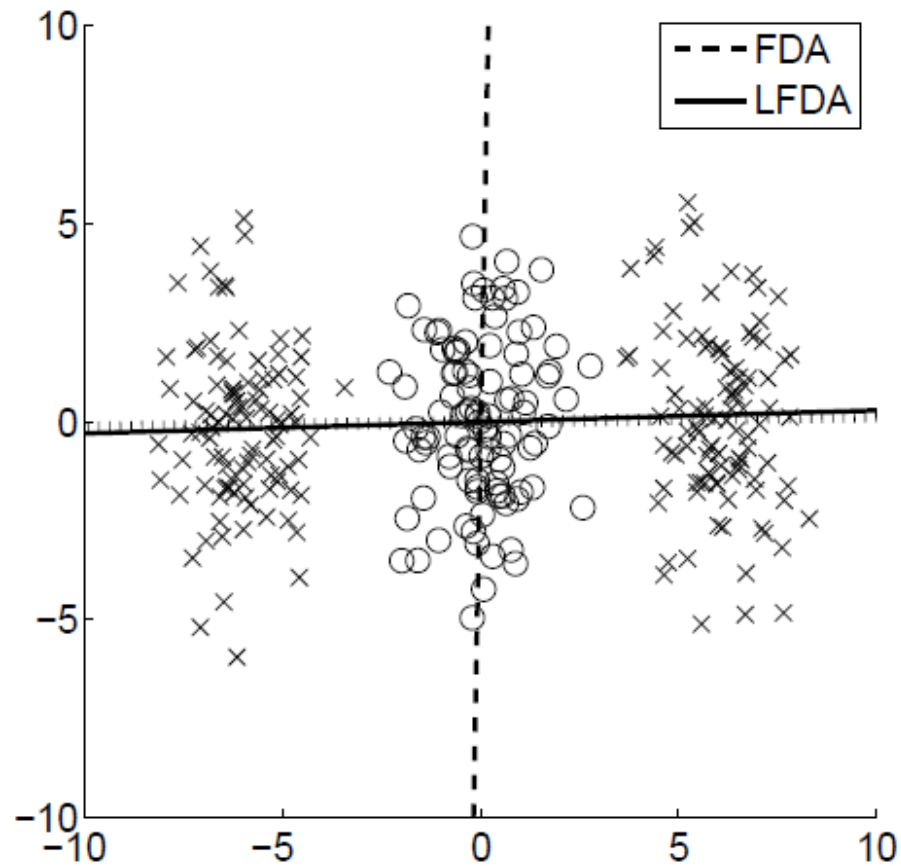
$$\mathbf{S}_b = \frac{1}{2} \sum_{i,j=1}^n \bar{A}_{ij}^{(b)} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$$

$$\bar{A}_{ij}^{(w)} = \begin{cases} \mathbf{A}_{ij} / n_c, & \text{if } y_i = y_j = c \\ 0, & \text{if } y_i \neq y_j \end{cases}$$

$$\bar{A}_{ij}^{(b)} = \begin{cases} \mathbf{A}_{ij} (1/n - 1/n_c), & \text{if } y_i = y_j = c \\ 1/n, & \text{if } y_i \neq y_j \end{cases}$$

# 7.5 局部线性判别分析

- Demo



## 7.6 其它维数缩减方法

- 经典方法
  - 独立成分分析 (Independent Component Analysis , ICA)
  - 典型关联分析(Canonical Correlation Analysis, CCA)
  - 2DPCA, 2DLDA
  - KPCA
- 流形学习方法: **LLE, Isomap, LE, LTSA**,... (下次课)
- 深度学习方法
  - PCANet
  - RBM, DBN, DBM, AutoEncoder
  - Deep CCA
  - Learning understanding Neural networks with Non-negative matrix factorization
- 应用: Eigenface, PCA-SIFT (CVPR 2004),...

# 致谢

- **Courtesy for some slides**
  - Xuyao Zhang
  - Bin Fan
  - Gaofeng Meng
  - ...

**Thank All of You!**  
**(Questions?)**

**向世明**

**smxiang@nlpr.ia.ac.cn**

**people.ucas.ac.cn/~xiangshiming**

**时空数据分析与学习课题组 (STDAL)**

**中科院自动化研究所· 模式识别国家重点实验室**